Adaptive Modeling and Simulation 2013

Edited by:

J. P. Moitinho de Almeida, P. Díez, C. Tiago and N. Parés

Adaptive Modeling and Simulation 2013

Adaptive Modeling and Simulation 2013

Proceedings of the VI International Conference
on Adaptive Modeling and Simulation (ADMOS 2013)
held in Lisbon, Portugal
3 – 5 June 2013

Edited by

J. P. Moitinho de Almeida
Instituto Superior Técnico, Portugal

P. Díez
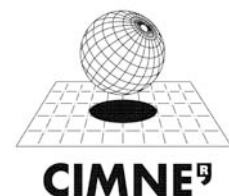Universitat Politècnica de Catalunya, Spain

C. Tiago
Instituto Superior Técnico, Portugal

N. Parés
Universitat Politècnica de Catalunya, Spain

A publication of:

CIMNE

ACKNOWLEDGEMENTS

PREFACE

This book contains the **Extended Abstracts and Full Papers** of the papers presented at ADMOS 2013, the sixth International Conference on Adaptive Modeling and Simulation, held at Instituto Superior Técnico, Technical University of Lisbon, Portugal, from June 3 to 5, 2013.

Numerical Modeling and Simulation is increasingly used as a complement to Experimental Modeling and Analysis and as a design or certification tool in engineering applications. However, after more than thirty years of worldwide research efforts around Adaptive Modeling and Simulation, the problem of properly assessing and controlling the quality of the numerical solutions is still relevant, as the design of sophisticated engineering systems requires increasingly complex and coupled modeling, which leads to increasingly time-consuming computations. Adaptive approaches, which provide reliable and cost efficient modeling and coherent coupling of different scales and mechanisms in a unique model, are more strategic and indispensable than ever.

The objective of the ADMOS 2013 conference is to provide a forum for presenting and discussing the current state-of-the-art achievements on Adaptive Modeling and Simulation, including theoretical models, numerical methods, algorithmic strategies and challenging engineering applications.

This book collects the contributions presented by the participants in the conference, which address a wide range of topics in adaptive modeling, from the classical theoretical aspects and numerical techniques to cutting edge problems and formulations, such as the determination of guaranteed error bounds and adaptive strategies for non-linear, transient or coupled problems, the application of adaptive techniques to reduced order models, the use of goal-oriented anisotropic error estimators and remeshing, as well as the application of adaptive techniques in the stochastic framework.

Advanced numerical techniques are also considered, for example XFEM, Discrete Galerkin, Meshless and Domain Decomposition, as well as the more classical methods. The domain of the applications covers a wide range of problems, from the traditional areas of structural and fluid mechanics, to quantum mechanics or biomechanics, with scales ranging from the nanoparticles, via the mesoscopic level, to those of industrial structures and civil engineering constructions, all studied with a focus on the main goal of adaptivity, seeking reliable and cost efficient modeling.

This book includes **contributions** sent directly by the authors, and the editors cannot accept responsibility for any inaccuracies, comments and opinions contained in the text. The organizers would like to take this opportunity to thank all the authors for submitting their contributions.

JP Moitinho de Almeida, Carlos Tiago
Instituto Superior Técnico,
Technical University of Lisbon,
Lisbon, Portugal

Pedro Díez, Núria Parés
Laboratory of Computational Methods
and Numerical Analysis (LaCàN)
Universitat Politècnica de Catalunya
Barcelona, Spain

SUMMARY

CONTENTS

PLENARY LECTURES

GENERAL CONTRIBUTIONS

PLENARY LECTURES

VIRTUAL CHART BASED DESIGN, INVERSE ANALYSIS AND CONTROL

F. CHINESTA[1,2]

[1] EADS Corporate Foundation International chair
GeM UMR CNRS - Ecole Centrale Nantes
1 rue de la Noe, BP 92101, F-44321 Nantes cedex 3, France
e-mail: Francisco.Chinesta@ec-nantes.fr

[2]IUF - Institut Universitaire de France

Key words: Parametric modeling, Model order reduction, PGD, Virtual charts

Abstract. Virtual chart constitutes an appealing tool for performing efficient design, real time simulations, efficient optimization, inverse analysis and control of systems and processes. When such charts are available, no more on-line simulation are needed, because everything reduces to a dialog between the designer or controller and the chart. The chart contains all the required information, that is, the solution of the physical model for each choice of the model parameters. These parameters can correspond to the applied loads, initial or boundary conditions, material or process parameters, parameters defining the geometry of the domain in which the model is defined, ... The main difficulty related to the construction of such a chart lies in the fact that the parametric space must be accurately explored in order to define an accurate chart. When the number of parameters increases, the sampling of such high-dimensional parametric space becomes unaffordable when standard discretization techniques apply for each possible scenario. Model order reduction makes possible the off-line construction of virtual charts in two ways. First, by constructing an adaptive reduced basis, from which the solution of each direct problem can be performed very fast, reaching several order of magnitude of CPU time savings. Second, by introducing the parameters as extra-coordinates and then addressing the solution of the resulting multidimensional problem by invoking the PGD, one can have access to the solution of innumerable scenarios, all those represented by the discretization of the extra-coordinates related of the model parameters. Again the question of adaptivity is crucial. From these virtual charts, simulation, optimization, inverse analysis, control, ... can be performed on-line, many times in real time and by using light computing devices like smartphones or tablets.

VI International Conference on Adaptive Modeling and Simulation
ADMOS 2013
J. P. Moitinho de Almeida, P. Díez, C. Tiago and N. Parés (Eds)

PARALLEL MESH MULTIPLICATION AND GENERATION: TOWARDS PETASCALE SIMULATIONS

ALVARO L.G.A. COUTINHO[*]

[*] Professor, High Performance Computing Center and Department of Civil Engineering
COPPE/Federal University of Rio de Janeiro
PO Box 68506, Rio de Janeiro, RJ21945-970, Brazil
e-mail: alvaro@coc.ufrj.br

Abstract. In this talk we review our current efforts in the development of enabling technologies for parallel large-scale simulations. We begin by revisiting the concept of what is large, given the current capabilities of today's high performance computers and high resolution visualisation devices. In the sequel we discuss a very efficient parallel procedure to produce high resolution models, the mesh multiplication (MM). MM recursively refines your mesh until a preset level is reached. Implementation issues, associated to mesh partition and unknown reordering are addressed. We then show a parallel implementation and performance analysis of a linear octree-based mesh generation scheme designed to create reasonable-quality, geometry-adapted unstructured hexahedral meshes automatically from triangulated surface models. We present algorithms for the construction, 2:1 balancing and meshing large linear octrees. Our scheme uses efficient computer graphics algorithms for surface detection, allowing us to represent complex geometries. We show that our implementation is able to execute the 2:1 balancing operations over 3.4 billion octants in less than 10 seconds per 1.6 million octants per CPU core. Next we examine the performance impact from tetrahedralization of non- conforming meshes generated by our parallel octree mesh generation scheme. We end our talk with a discussion of the applicability of these enabling technologies in challenging new applications in science and engineering.

VI International Conference on Adaptive Modeling and Simulation
ADMOS 2013
J. P. Moitinho de Almeida, P. Díez, C. Tiago and N. Parés (Eds)

DIRECT FEM-SIMULATION OF TURBULENT FLOW

J. HOFFMAN, J. JANSSON, N. JANSSON, R. VILELA DE ABREU and CLAES JOHNSON

Computer Science and Communication, KTH, SE-10044 Stockholm, Sweden.

Abstract. Turbulent fluid flow has been considered as the main unsolved problem of classical mechanics beyond theoretical description and also beyond computational simulation, because of thin no-slip boundary layers requiring trillions of mesh points to be resolved. In recent work we have discovered that using a slip boundary condition as a model of the small skin friction of slightly viscous turbulent flow, allows predictive simulation of mean value quantities such as drag and lift of turbulent flow with instead millions of mesh points. Basic aspects of turbulent flow from applications point of view thus show to be computable by stabilized finite element methods without turbulence modeling referred to as Direct FEM-Simulation, which opens large areas for exploration. As a key example the turbulent flow around a wing and complete airplane is computable and inspecting the solutions leads to a new theory flight essentially different from the accepted theory by Kutta-Zhukovsky-Prandtl developed 100 years ago.

REFERENCES

[1] J. Hoffman, J. Janson and C. Johnson, New Theory of Flight, submitted to J. Math Fluid Mech.

[2] J. Hoffman and C. Johnson, Computational Turbulent Incompressible Flow, Springer 2008.

[3] N. Jansson, J. Hoffman, J. Jansson, Framework for Massively Parallel Adaptive Finite Element Computational Fluid Dynamics on Tetrahedral Meshes, SIAM J. Sci. Comput., Vol. 34(1), pp. C24-C41, 2012.

[4] J. Hoffman, J. Jansson and R. Vilela De Abreu, Adaptive modeling of turbulent flow with residual based turbulent kinetic energy dissipation, Computer Methods in Applied Mechanics and Engineering, Vol.200(37-40), pp.2758-2767, 2011.

[5] J.Hoffman, J.Jansson, M.Stckli, Unified continuum modeling of fluid-structure interaction, Mathematical Models and Methods in Applied Sciences, Vol.21(3), pp.491-513, 2011.

[6] J.Hoffman and N.Jansson, A computational study of turbulent flow separation for a circular cylinder using skin friction boundary conditions, Quality in Large Eddy Simulation II, ERCOFTAC Series Vol.16, Springer, 2011.

ERROR-CONTROLLED ADAPTIVE MULTISCALE ANALYSIS FOR CRACK INITIATION AND PROPAGATION IN BRITTLE MATERIALS

ERWIN STEIN[*], TYMOFIY GERASIMOV[†], MARCUS RÜTER[†]

[*] Institute of Mechanics and Computational Mechanics (IBNM)
Leibniz Universität Hannover
Appelstraße 9A, 30167 Hannover, Germany
e-mail: stein@ibnm.uni-hannover.de

[†] Institute of Continuum Mechanics (IKM)
Leibniz Universität Hannover
Appelstraße 9A, 30167 Hannover, Germany
e-mail: gerasimov@ikm.uni-hannover.de, rueter@ibnm.uni-hannover.de

Abstract. The addressed research is based on linear elastic fracture mechanics for the macro-scale, so far, and on non-local continuum damage mechanics with linear kinematics until micro-cracking on micro-scale. The material considered is alloyed ceramics (Yttria (3Y) stabilized (with 2%-3%) tetragonal Zirconia-Oxide -- a polycrystalline ceramic 3Y-TZP). The main goals are: error-controlled adaptive modeling and numerical approximations on both scales, including error estimators for quantities of interest. This also needs energy consistent projections from micro- to macro-scale and backwards. The final objective is defined as coupled micro-macro damage and crack propagation processes for technologically interesting problems. This is scheduled in an overall adaptive scheme, trying to realize step by step verification and validation of this coupled process.

Our current results are based on new explicit and implicit residual error estimators for the eXtended Finite Element Method (XFEM), including goal-oriented error estimation [1,2]. Special features of singular enrichment functions within XFEM are also discussed. In particular, it is shown that a significant reduction of the discretization error in crack tip element is achieved by using a statically admissible asymptotic displacement field in the XFEM discretizations. Alternative to XFEM, the adaptive Singular Function Method (SFM) is considered, including new explicit residual (constant-free) error estimator for low order triangles [3], yielding very good effectivity indices between 1 and 2.

Modeling of microcrack nucleation and coalescence in ceramic specimen is realized within the framework of Continuum Damage Mechanics (CDM), in particular in terms of a non-local damage model using the enhanced gradient formulation [4]. Error estimation analysis for this coupled problem and results of adaptive mesh refinements are presented. A major point is the transition from continuous damage to equivalent micro-cracks, using energetic equivalence between damage and fracture [5].

A two-scale coupling of the above two processes is presented. The transition between the two scales is realized by the multiscale projection method.

REFERENCES

[1] T. Gerasimov, M. Rüter E. Stein: An explicit residual-type error estimator for Q1-quadrilateral XFEM in 2D LEFM. International Journal for Numerical Methods in Engineering, (90) 2012, 1118--1155.

[2] M. Rüter, T. Gerasimov, E. Stein (2012) Goal-oriented explicit residual-type error estimates in XFEM, *accepted to Computational Mechanics*

[3] T. Gerasimov, E. Stein and P. Wriggers (2012) Fully adaptive FEM/SFM for 2D fracture and damage using a new simple constant-free explicit error estimator, *in preparation*.

[4] R.H.J. Peerlings, R. de Borst, W.A.M. Brekelmans, J.H.P. de Vree: Gradient-enhanced damage for quasi-brittle materials. International Journal for Numerical Methods in Engineering, (39) 1996, 3391--3403.

[5] J. Mazars, G. Pijaudier-Cabot From damage to fracture mechanics and conversely: a combined approach, *Int J Solid Struct* 1996; 33:3327-3342.

GENERAL CONTRIBUTIONS

HIERARCHICAL MOMENT CLOSURE APPROXIMATION OF THE BOLTZMANN EQUATION

M. Abdel Malik[∗], **E.H. van Brummelen**[∗]

[∗]Multiscale Engineering Fluid Dynamics
Department of Mechanical Engineering
Eindhoven University of Technology
Den Dolech 2, 5612 AZ Eindhoven, Netherlands
e-mail: M.Abdel.Malik@tue.nl

Key words: Adaptive Modeling, Hierarchical Equations, Boltzmann, Kinetic Theory, Moment Closure

Abstract. This work discusses the application of the moment method to a generic form of kinetic equations, given by the Boltzmann equation, to simplify kinetic models of particle systems. Implicit to the method of moments is an approximation of moment closure relations to close the system of equations. The main aim is to explore the opportunities, pertaining to goal-oriented adaptive modeling, presented by the hierarchical structure of moment-closure systems.

1 INTRODUCTION

The Boltzmann equation is the classical model in the kinetic theory of (mono-atomic) fluids, describing rarefied flow by modeling deviations of the velocity distribution from a local equilibrium, thus, accounting for the transitional molecular/continuum regime. Boltzmann's equation provides an evolution equation for the one-particle marginal, viz., the probability density of particles in phase (position/velocity) space. The Boltzmann equation has several fundamental structural properties, notably, conservation of mass, momentum and energy, Galilean invariance and decay of an entropy functional (the celebrated H-theorem). These structural properties underly the connection between the Boltzmann equation and conventional continuum models: all conventional continuum models, such as the Navier-Stokes-Fourier system [1], can be derived as limits of the Boltzmann equation.

Boltzmann's equation poses a formidable challenge for numerical approximation methods, on account of its high dimensional phase-space setting: for a problem in N spatial dimensions, the single molecule phase-space is $2N$ dimensional. Away from the fluid dynamical regime numerical approximations of kinetic systems are predominantly based on

M. Abdel Malik* and E.H. van Brummelen*

particle methods, such as the Direct Simulation Monte Carlo (DSMC) method. However, the phase-space description of the system results in the prohibitive computational cost of DSMC in the fluid dynamical limit. Moreover, from an approximation perspective, DSMC can be inefficient since it is inherent to Monte-Carlo processes that the approximation error decays only as $n^{-\frac{1}{2}}$ for the number of simulation molecules n [2]. Hence, efficiently modeling gases in the transition regime between the free molecular flow and fluid dynamics remains difficult.

An alternative strategy to describe deviations from fluid dynamics is by means of moment-closure approximations [3,4]. In moment-closure approximations, the Boltzmann equation is projected onto a polynomial space, in the velocity dependence, and the system is closed by providing an approximation to the one-particle marginal based on the same polynomial space. This procedure can in fact be conceived of as a Galerkin approximation. The closure is chosen such that the structural properties of the Boltzmann equation are retained. Results by Schmeiser and Zwirchmayr [5] show that moment equations converge to linear kinetic equations as the order of moment approximation tends to infinity and to a drift-diffusion model in the macroscopic limit, i.e. as the Knudsen number tends to zero. Furthermore, from an adaptive approximation standpoint, the resulting hierarchical structure of the the moment closure system presents promising potential for rigorous model adaptivity. However, fundamental challenges remain to be addressed.

This work discusses the application of the moment method to the Boltzmann equation to derive a closed hierarchy of moment systems that retain structural features of the system in question. In addition opportunities pertaining to goal-oriented adaptive modeling provided by the hierarchical structure exhibited by the resulting closed system of moment equations will be explored. The remainder of this paper is arranged as follows, section 2 enlists the structural properties of the Boltzmann equation that are to be retained by the moment closure approximation; section 3 introduces concepts relevant to moment systems pertaining to subspace approximations as well as the consequential moment closure approximation; section 4 discusses the hierarchical structure of the resulting closed system of moment equations within a multiscale modeling framework and the oppertunities this presents for (goal-oriented) model adaptivity; finally, section 5 gives a concluding summary.

2   THE BOLTZMANN EQUATION: PROPERTIES

Consider a gas composed of a single species of identical classical particles, i.e. a monatomic gas, contained within a fixed spatial domain $\boldsymbol{\Omega} \subset \mathbb{R}^D$. Based on kinetic theory the evolution of a non-negative (phase-space) density $f = f(t, \boldsymbol{x}, \boldsymbol{v})$ over a single particle phase $\boldsymbol{\Omega} \times \mathbb{R}^D$ is governed by the (kinetic) Boltzmann equation expressed as

$$\partial_t f + v_j \partial_{x_j} f = \mathcal{C}(f) \tag{1}$$

where the collision operator $f \mapsto \mathcal{C}(f)$ acts only on the $\boldsymbol{v}$ dependence of f locally at each $(t, \boldsymbol{x})$. Let $\langle \cdot \rangle$ denote $\boldsymbol{v}-$integrations of any scalar, vector or matrix valued measurable

function over the $D-$dimensional Lesbesgue measure $d\boldsymbol{v}$. All functions considered in this work are understood to be Lebesgue measurable in all variables.

The collision operator $\mathcal{C}$ is assumed to be defined over the domain $\mathscr{D}(\mathcal{C})$ that is contained within the set of non-negative functions of $\boldsymbol{v}$. Furthermore, it is assumed that $\mathcal{C}$ has the following properties [4]:

1. *Conservation: Mass, Momentum and Energy*
   Definition A quantity γ is said to be a *collision invariant* of $\mathcal{C}$ if

$$\langle \gamma \mathcal{C}(f) \rangle = 0, \qquad \forall f \in \mathscr{D}(\mathcal{C}) \tag{2}$$

   Denote the collection of collision invariants of $\mathcal{C}$ by $\mathscr{C}$ and let the basis for an N-dimensional $\mathscr{C}$ be written as $\{e_i : 1 \leq i \leq N\}$. Relation (2) leads to N independent local conservation laws

$$\partial_t \langle e_i f \rangle + \partial_{x_j} \langle v_j e_i f \rangle = 0 \tag{3}$$

   It is assumed that the set of collision invariants is given by

$$\mathscr{C} = \mathrm{span}\{1, \boldsymbol{v}, |\boldsymbol{v}|^2\} \tag{4}$$

   where the notation in (4), adopted throughout this paper, applies to a collection of scalars, vectors and tensors, implying that the span consists of all scalar-valued linear combinations of their components. Assumption (4) implies the Boltzmann equation (1) conserves mass, momentum and energy, and has no other invariants.

2. *Entropy Dissipation: H-Theorem*
   Definition A convex function $\mathcal{H} = \mathcal{H}(f)$ over $\mathbb{R}_+$ is called an *entropy* for $\mathcal{C}$ if

$$\langle \mathcal{C}(f) \partial_f \mathcal{H}(f) \rangle \leq 0, \qquad \forall f \in \mathscr{D}(\mathcal{C}) \tag{5}$$

   and if for every $f \in \mathscr{D}(\mathcal{C})$ the following statements are equivalent

$$\begin{array}{l} \text{i. } \langle \partial_f \mathcal{H}(f) \mathcal{C}(f) \rangle = 0 \\ \text{ii. } \mathcal{C}(f) = 0 \\ \text{iii. } \partial_f \mathcal{H}(f) \in \mathscr{C} \end{array} \tag{6}$$

Relations (5) and (6) are abstractions of Boltzmann's H-theorem, where (5) assumes that $\mathcal{C}$ dissipates some entropy and (6) characterizes local equilibria of $\mathcal{C}$ by vanishing entropy dissipation. Denoting the Legendre transform of the entropy $\mathcal{H}$ by $\mathcal{H}^*$, i.e.

$$\mathcal{H}(y) + \mathcal{H}^*(z) = yz \tag{7}$$

(5) and (6) suggest that the equilibrium distribution, denoted by f_{eq}, is given by

$$f_{eq} = \partial_z \mathcal{H}^*(z) \qquad \text{for } z \in \mathscr{C} \tag{8}$$

It is assumed that the entropy of $\mathcal{C}$ exists. This assumption implies that solutions of the Boltzmann equation (1) satisfy the local dissipation law corresponding to entropy dissipation

$$\partial_t \langle \mathcal{H}(f) \rangle + \partial_{x_j} \langle v_j \mathcal{H}(f) \rangle = \langle \mathcal{C}(f) \partial_f \mathcal{H}(f) \rangle \leq 0 \tag{9}$$

where $\langle \mathcal{H}(f) \rangle$, $\langle v_j \mathcal{H}(f) \rangle$ and $\langle \partial_f \mathcal{H}(f) \mathcal{C}(f) \rangle$ are referred to as entropy density, entropy flux and entropy dissipation rate, respectively.

3. *Symmetry: Galilean Invariance*
   The operator $\mathcal{C}$ is assumed to commute with translational and orthogonal transformations. Specifically, given any $g = g(t, \boldsymbol{x}, \boldsymbol{v})$ for every vector $\boldsymbol{u} \in \mathbb{R}^D$ and for every orthogonal matrix $O \in \mathbb{R}^{D \times D}$ define transformed functions $\mathcal{T}_u f$ and $\mathcal{T}_O f$ by

$$\mathcal{T}_u g = \mathcal{T}_u g(t, \boldsymbol{x}, \boldsymbol{v}) \equiv g(t, \boldsymbol{x} - \boldsymbol{u}t, \boldsymbol{u} - \boldsymbol{v}), \quad \mathcal{T}_O g = \mathcal{T}_O g(t, \boldsymbol{x}, \boldsymbol{v}) \equiv g(t, O^\top \boldsymbol{x}, O^\top \boldsymbol{v}) \tag{10}$$

It is assumed that if $f \in \mathscr{D}(\mathcal{C})$ then so are $\mathcal{T}_u f$ and $\mathcal{T}_O f$:

$$\mathcal{T}_u \mathcal{C}(f) = \mathcal{C}(\mathcal{T}_u f), \qquad \mathcal{T}_O \mathcal{C}(f) = \mathcal{C}(\mathcal{T}_O f) \tag{11}$$

Assumption (11) implies that if f satisfies (1) so does the image of f under a Galilean group.

3 MOMENT CLOSURE

Physically, one may be more interested in functions of f (observables), from which macroscopic properties can be extracted, than in f itself. Such reasoning motivates the derivation of equations for such observables instead. That is, rather than resolving equation (1) for f, one could resolve moment systems (or weighted averages) of f, which would govern the evolution of a finite set of velocity moments of f. In resolving the moment equations instead of (1), the velocity dependence of f is replaced by a finite number of parameters, thus reducing the complexity of the problem [6].

To derive the moment equations, consider a finite linear subspace $\boldsymbol{\Theta}$ of functions of $\boldsymbol{v}$ (taken to be polynomials) with dimension θ and basis $\{\vartheta_i = \vartheta_i(\boldsymbol{v})\}_{i=1}^{\theta}$. Denote the column $\theta-$vector of these basis elements by $\boldsymbol{\vartheta} = \boldsymbol{\vartheta}(\boldsymbol{v})$, so that every $\vartheta \in \boldsymbol{\Theta}$ has a unique representation in the form $\vartheta(\boldsymbol{v}) = \boldsymbol{\alpha}^\top \boldsymbol{\vartheta}(\boldsymbol{v})$ for some $\boldsymbol{\alpha} \in \mathbb{R}^\theta$. Taking the moments, i.e. weighted average, of equation (1) over the vector $\boldsymbol{\vartheta}(\boldsymbol{v})$ yields

$$\partial_t \langle \boldsymbol{\vartheta} f \rangle + \partial_{x_j} \langle v_j \boldsymbol{\vartheta} f \rangle = \langle \boldsymbol{\vartheta} \mathcal{C}(f) \rangle \tag{12}$$

thus, a weaker form of equation (1) is formally expressed as a hierarchy of moment systems of partial differential equations in (12) in the sense that a solution of (1) would also satisfy (12). In general, it is not known whether the quantities appearing in this equation are well defined for every solution f of a given kinetic equation. Since, it has been shown that this is the case for the spatially homogenous equation [7], following Levermore [4] it shall be assumed here that these quantities are well defined.

Furthermore, it is observed that in equations (12) the flux in one equation appears as the density in the subsequent one, i.e. the expansion at some order n contains the moments at orders $n \pm 1$. Moreover, the equations contain the production terms which are related to the distribution function f through the collision term $\mathcal{C}(f)$. Therefore, in order to have a complete set of equations for the moments, constitutive relations are needed to express the densities $\langle \boldsymbol{\vartheta} f \rangle$, fluxes $\langle \boldsymbol{v} \boldsymbol{\vartheta} f \rangle$ and collisional terms $\langle \boldsymbol{\vartheta} \mathcal{C}(f) \rangle$ as a function of θ variables, thus forming a closed system. Generally, this is achieved by finding a relation between the moments and the distribution function. Deriving such a relation is called the moment closure problem.

A subspace $\boldsymbol{\Theta}$ will be called admissible if it satisfies (see [4])

i. $\mathscr{C} \subseteq \boldsymbol{\Theta}$:
   In this condition, the collection of collision invariants $\mathscr{C}$ is contained within $\boldsymbol{\Theta}$. More specifically, the constant functions are included in $\boldsymbol{\Theta}$ so that any moment closure will include the conservation law for mass. It also includes multiples of the polynomial $\boldsymbol{v}$, which gives a balance law for momentum. Multiples of $|\boldsymbol{v}|^2$ give a balance law for the energy. This is needed if any fluid dynamical approximation is to be recovered.

ii. $\boldsymbol{\Theta}$ is invariant under actions of $\mathcal{T}_u$ and $\mathcal{T}_o$:
   More specifically, this means that $\boldsymbol{\Theta}$ is unchanged when $\boldsymbol{v} \mapsto O^\top \boldsymbol{v}$ or $\boldsymbol{v} \mapsto \boldsymbol{v} - \boldsymbol{u}$, for every vector $\boldsymbol{u} \in \mathbb{R}^D$ and for every orthogonal matrix $O \in \mathbb{R}^{D \times D}$. This is a prerequisite of classical dynamics, in particular, that Galilean invariance holds.

Closure of (12) is attained if there exists a function $\mathcal{F}$ (and is made known) such that $f(t, \boldsymbol{x}, \boldsymbol{v}) = \mathcal{F}(\langle f, \boldsymbol{\vartheta} \rangle, \boldsymbol{v})$. Then the flux terms $\langle \boldsymbol{v} \boldsymbol{\vartheta} f \rangle$ and the collision terms $\langle \boldsymbol{\vartheta} \mathcal{C}(f) \rangle$ can be related to the densities $\langle \boldsymbol{\vartheta} \mathcal{F} \rangle$ to provide a closed system of the form

$$\partial_t \langle \boldsymbol{\vartheta} \mathcal{F} \rangle + \partial_{x_j} \langle v_j \boldsymbol{\vartheta} \mathcal{F} \rangle = \langle \boldsymbol{\vartheta} \mathcal{C}(\mathcal{F}) \rangle \tag{13}$$

Note that f is an element of an infinite dimensional vector space and typically cannot be expressed by any finite number of components. Therefore, any closure will require the approximation of f. The aim is to devise an approximation that, in addition to providing well-posedness of (13), maintains the aforementioned structural features of (1) listed in section 2. Moreover, the closure relation should result in a tractable system.

Remark Considering the smallest admissible subspace $\boldsymbol{\Theta} \equiv \mathscr{C}$ reduces (13) to the Euler equations for a monatomic gas.

Previous studies of moment closure approximations can be found in [3, 4, 6, 8]. Grad's moment closure [3] is based on an expansion of the one-molecule marginal using Hermite polynomials, modulated by the local equilibrium distribution. A deficiency of Grad's moment closure system is the potential occurrence of locally negative and therefore inadmissible phase-space distributions, and potential loss of hyperbolicity [9, 10]. Later, Levermore [4] developed a moment closure system based on entropy minimization, which leads to an exponential closure. However, it was subsequently shown by Junk [11] that Levermore's moment closure system suffers from a realizability problem, i.e. there exist moments for which the minimum entropy solution is undefined. On the other hand, results by Junk [11, 12], Schneider [13] and Pavan [14], show that a relaxation of the entropy minimization problem is well-posed while retaining exponential closure. However, employing the relaxed minimization problems leads to loss of the one-to-one correspondence between the entropy minimizing distribution and the moments of the single-particle phase-space densities. An additional deficiency of Levermore's entropy-based closure is the potential occurrence of singularities owing to the fact that densities describing local thermodynamic equilibrium may belong to the boundary of the set containing all degenerate densities [11, 12, 15]. Another fundamental complication pertaining to the implementation of the moment-closure systems based on exponential closure, is that the resulting formulation requires the evaluation of moments of exponentials of polynomials of, in principle, arbitrary orders. It is generally accepted that the derivation of closed-form expressions for such moments is intractable, and accurate approximation of the moments is a notoriously difficult problem [16] .

Abdel Malik and Van Brummelen [8] have recently investigated moment-closure systems based on approximations of the exponential of the form given by an even-order Taylor-series approximation about a Maxwellian. The even-order of the Taylor-series approximation ensures non-negativity of the approximation. The results in [8] convey that in this manner, it is possible to construct well-posed moment-closure systems that retain the structural features of the Boltzmann equation listed in section 2, but for which the evaluation of moments of exponentials of arbitrary polynomials is replaced by the evaluation of high-order moments of Gaussians. Such high-order moments of Gaussians can be evaluated in closed form. It has been noted that Grad's moment closure can be perceived as a first order approximation of Levermore's exponential closure [6, 17]. The results of [8] can be conceived as a refinement of Grad's moment system as it overcomes potentially negative densities and potential loss of hyperbolicity by incorporating higher order approximations.

4   MULTISCALE HIERARCHY: MODEL ADAPTIVITY

Moment-closure systems potentially offer efficient hierarchical approximations of the Boltzmann equation, by exploiting the fact that its solutions converge to functions in a particular class, the equilibrium distributions. Adaptivity between the Boltzmann equation (micro model) and the Euler equations (macro model) has been propounded by several authors, e.g., E and Engquist [18]. However, no methodology to this purpose has been developed so far. The scale disparity between the Euler equations and the Boltzmann equations is enormous, which renders direct adaptivity between these two models infeasible. The hierarchy of moment-closure systems can be conceived of as a gradual transition from the Euler equations to the Boltzmann equations. The goal-oriented error estimate provides a systematic refinement criterion.

This section aims to explore the potential opportunities provided by the hierarchical structure of the moment equations in (13) for numerical approximation of the Boltzmann equation, more specifically, for goal-oriented model adaptivity.

4.1   Towards Goal-Oriented Model Adaptivity

The general procedure for performing goal-oriented adaptivity consists of repeated application of the following operations [19]:

$$\text{SOLVE} \rightarrow \text{ESTIMATE} \rightarrow \text{MARK} \rightarrow \text{REFINE}$$

Given an initial mesh, the first step concerns the solution of the finite-element problem on that mesh. The second step is the construction of a computable a-posteriori estimate of the error in the finite-element approximation, based on the current approximation and exogenous data. In the third step, the error estimate is decomposed into element-wise contributions, and the elements which yield the largest contributions to the error in the goal functional are marked according to some marking strategy. In the final stage, the marked elements are refined.

In the context of the hierarchy of moment-closure systems, it is envisaged that the above procedure can be extended to include both mesh-adaptivity and hierarchical-rank adaptivity. That is, not only estimate the error in the numerical solution, but also the error in the moment-closure system itself. Accordingly, the adaptive procedure locally adapts the element size and the hierarchical rank to arrive at an optimal approximation. The challenges in the development of the a-posteriori error estimate and the adaptive-refinement procedure, are the construction of a computable error estimate for the moment-closure hierarchy, and the (nonstandard) decomposition of the error into contributions from the model error (i.e., the finite rank of the considered moment-closure system) and the finite-element approximation error (i.e., the finite mesh width). For the a posteriori error estimation, one could employ duality-based techniques [20–22]. Moreover, by virtue of the fact that moment-closure systems assume the form of a hierarchy of hyperbolic systems, the discontinuous-Galerkin formulation is well suited to discretize these systems, and element-wise refinement indicators can then be derived in a similar manner as in [23].

5 CONCLUSION

The enormous potential of moment-closure approximations of the Boltzmann equation in the context of numerical-approximation techniques, lies in the fact that such approximations assume the form of a hierarchy of systems of hyperbolic partial-differential equations, which puts the full arsenal of approximation techniques for this class of problems at our disposal, in particular, (goal-)adaptive finite-element methods based on a-posteriori error estimates. The hierarchical structure of the moment-closure systems, engendered by the inclusion relation acting on the polynomials, implies that the solution of each system can be regarded as an approximation to the solution to the next member in the hierarchy. Hence, by virtue of the hierarchical structure, a-posteriori error estimates can be constructed by evaluating the residual of the next member in the hierarchy. Based on this observation, we propounded an adaptive numerical approach, in which not only the mesh width and order of approximation in the finite-element method are locally adapted for one particular moment-closure system, but in which also the hierarchical rank of the moment-system is locally adapted, in accordance with an a-posteriori error estimate.

REFERENCES

[1] L. Saint-Raymond. *Hydrodynamic Limits of the Boltzmann Equation*. Number v. 1971 in Lecture Notes in Mathematics. Springer, 2009.

[2] A. Klenke. *Probability Theory: A Comprehensive Course*. Universitext (1979). Springer, 2008.

[3] H. Grad. On the kinetic theory of rarefied gases. *Communications on Pure and Applied Mathematics*, 2(4):331407, 1949.

[4] C. Levermore. Moment closure hierarchies for kinetic theories. *Journal of Statistical Physics*, 83:10211065, 1996.

[5] C. Schmeiser and A. Zwirchmayr. Convergence of Moment Methods for Linear Kinetic Equations. *SIAM Journal on Numerical Analysis*, 36(1):pp. 7488, 1998.

[6] H. Struchtrup. *Macroscopic Transport Equations for Rarefied Gas Flows: Approximation Methods in Kinetic Theory*. Interaction of Mechanics and Mathematics Series. Springer, 2005.

[7] T. Elmroth. Global boundedness of moments of solutions of the Boltzmann equation for forces of infinite range. *Archive for Rational Mechanics and Analysis*, 82:112, 1983.

[8] M.R.A. Abdel Malik and E.H. van Brummelen. Numerical approximation of the boltzmann equation: Moment closure. Technical Report 12-30, Eindhoven University of Technology, 2012.

[9] F. Brini. Hyperbolicity region in extended thermodynamics with 14 moments. *Continuum Mechanics and Thermodynamics*, 13:18, 2001.

[10] M. Torrilhon. Characteristic waves and dissipation in the 13-moment-case. *Continuum Mechanics and Thermodynamics*, 12:289301, 2000.

[11] M. Junk. Domain of Definition of Levermore's Five-Moment System. *Journal of Statistical Physics*, 93:11431167, 1998.

[12] M. Junk. Maximum entropy for reduced moment problems. *Mathematical Models and Methods in Applied Sciences*, 10:10011025, 2000.

[13] J. Schneider. Entropic approximation in kinetic theory. *ESAIM. Mathematical Modelling and Numerical Analysis*, 38:541561, 2004.

[14] V. Pavan. General Entropic Approximations for Canonical Systems Described by Kinetic Equations. *Journal of Statistical Physics*, 142:792827, 2011.

[15] M. Junk and A. Unterreiter. Maximum entropy moment systems and Galilean invariance. *Continuum Mechanics and Thermodynamics*, 14:563576, 2002.

[16] J.B. Lasserre. *Moments, Positive Polynomials and Their Applications*. Imperial College Press Optimization Series. Imperial College Press, 2010.

[17] W. Dreyer, M. Junk, and M. Kunik. On the Approximation of Kinetic Equations By Moment Systems. *Nonlinearity*, 14, 2000.

[18] E W. and B. Engquist. The heterognous multiscale methods. *Comm. Math. Sci*, 1:87132, 2003.

[19] W. Dörfler. A convergent adaptive algorithm for poisson's equation. *SIAM Journal on Numerical Analysis*, 33(3):1106–1124, 1996.

[20] M. Ainsworth and J.T. Oden. *A Posteriori Error Estimation in Finite Element Analysis*. Pure and Applied Mathematics: A Wiley Series of Texts, Monographs and Tracts. Wiley, 2011.

[21] W. Bangerth and R. Rannacher. *Adaptive Finite Element Methods for Differential Equations*. Lectures in Mathematics. Birkhuser Basel, 2003.

[22] Endre Süli. A posteriori error analysis and adaptivity for finite element approximations of hyperbolic problems. In Dietmar Krner, Mario Ohlberger, and Christian Rohde, editors, *An Introduction to Recent Developments in Theory and Numerics for Conservation Laws*, volume 5 of *Lecture Notes in Computational Science and Engineering*, page 123194. Springer Berlin Heidelberg, 1999.

[23] W. Hoitinga. *Goal Adaptive Discretization of a One-dimensional Boltzmann Equation*. PhD thesis, 2011.

VI International Conference on Adaptive Modeling and Simulation
ADMOS 2013
J. P. Moitinho de Almeida, P. Díez, C. Tiago and N. Parés (Eds)

DEIM-BASED PGD FOR PARAMETRIC NONLINEAR MODEL ORDER REDUCTION

JOSE V. AGUADO[1*], FRANCISCO CHINESTA[1], ADRIEN LEYGUE[1], ELIAS CUETO[2] AND ANTONIO HUERTA[3]

[1]GEM UMR CNRS, Ecole Centrale de Nantes
EADS Corporate Foundation International Chair
1 rue de la Noë, BP 92101, F-44321 Nantes cedex 3, France
e-mail: {jose.aguado-lopez;francisco.chinesta,adrien.leygue}@ec-nantes.fr

[2]Aragon Institute of Engineering Research (I3A), Universidad de Zaragoza
Maria de Luna 3, E-50018 Zaragoza, Spain
e-mail: ecueto@unizar.es

[3]Laboratori de Calcul Numeric (LaCaN), Universitat Politecnica de Catalunya
BarcelonaTech, 08034 Barcelona, Spain
e-mail: antonio.huerta@upc.es

Key words: Model Reduction; Nonlinear; DEIM; Proper Generalized Decomposition

Abstract. A new technique for efficiently solving parametric nonlinear reduced order models in the Proper Generalized Decomposition (PGD) framework is presented here. This technique is based on the Discrete Empirical Interpolation Method (DEIM)[1], and thus the nonlinear term is interpolated using the reduced basis instead of being fully evaluated. The DEIM has already been demonstrated to provide satisfactory results in terms of computational complexity decrease when combined with the Proper Orthogonal Decomposition (POD). However, in the POD case the reduced basis is *a posteriori* known as it comes from several pre-computed snapshots. On the contrary, the PGD is an *a priori* model reduction method. This makes the DEIM-PGD coupling rather delicate, because different choices are possible as it is analyzed in this work.

1 INTRODUCTION

The efficient resolution of complex models (in the dimensionality sense) is probably the essential objective of any model reduction method. This objective has been clearly reached for many linear models encountered in physics and engineering [2, 3]. However, model order reduction of nonlinear models, and specially, of parametric nonlinear models, remains as an open issue. Using classic linearization techniques such Newton method, both the nonlinear term and its Jacobian must be evaluated at a cost that still depends on

the dimension of the non-reduced model [1]. The Discrete Empirical Interpolation Method (DEIM), which the discrete version of the Empirical Interpolation Method (EIM) [4], proposes to overcome this difficulty by using the reduced basis to interpolate the nonlinear term. The DEIM has been used with Proper Orthogonal Decomposition (POD) [5, 1] where the reduced basis is *a priori* known as it comes from several pre-computed snapshots. In this work, we propose to use the DEIM in the Proper Generalized Decomposition (PGD) framework [5, 6], which is an *a priori* model reduction technique, and thus the nonlinear term is interpolated using the reduced basis that is being constructed during the resolution.

2 DEIM-based PGD FOR NONLINEAR MODELS

Consider a certain model in the general form:

$$\mathcal{L}(u) + \mathcal{F}_{NL}(u) = 0 \tag{1}$$

where $\mathcal{L}$ is a linear differential operator and $\mathcal{F}_{NL}$ is a nonlinear function, both applying over the unknown $u(\boldsymbol{x}), \boldsymbol{x} \in \Omega = \Omega_1 \times \ldots \times \Omega_d \subset \mathbb{R}^d$, which belongs to the appropriate functional space and respects some boundary and/or initial conditions. Using the PGD method implies constructing a basis $\mathcal{B} = \{\phi_1, \ldots, \phi_N\}$ such that the solution can be written as:

$$u(\boldsymbol{x}) \approx \sum_{i=1}^{N} \alpha_i \cdot \phi_i(\boldsymbol{x})$$

where α_i are coefficients, and

$$\phi_i(\boldsymbol{x}) = P_{i1}(x_1) \cdot \ldots \cdot P_{id}(x_d) \ , \ i = 1, \ldots, N$$

being $P_{ij}(x_j), j = 1, \ldots, d$, functions of a certain coordinate $x_j \in \Omega_j$. In the linear case, the basis $\mathcal{B}$ can be constructed sequentially by solving a nonlinear problem at each step in order to find functions P_{ij}. In the nonlinear case a linearization scheme for Eq. (1) is compulsory, but evaluating the nonlinear term is still as costly as in the non-reduced model. The DEIM method proposes to circumvent this inconvenient by performing an interpolation of the nonlinear term using the basis functions. In a POD framework, these functions come from the precomputed snapshots, but in a PGD framework these functions are constructed by using the PGD algorithm. Here we propose to proceed as follows:

1. Solve the linear problem: find u^0 such that $\mathcal{L}(u^0) = 0 \rightarrow \mathcal{B}^0 = \{\phi_1^0, \ldots, \phi_{N_0}^0\}$

2. Select a set of points $\mathcal{X}^0 = \{\boldsymbol{x}_1^0, \ldots, \boldsymbol{x}_{N_0}^0\}$. Later on we explain how to make an appropriate choice.

3. Interpolate the nonlinear term $\mathcal{F}_{NL}$ using functions $\mathcal{B}^0$ in the points $\mathcal{X}^0$. Or in other words, find the coefficients φ_i^0 such as:

$$\mathcal{F}_{NL}\left(u_m^0\right) \equiv \mathcal{F}_{NL}\left(u^0(\boldsymbol{x}_m^0)\right) = \sum_{i=1}^{N_0} \varphi_i^0 \cdot \phi_i^0(\boldsymbol{x}_m^0) \ , \ m = 1, \ldots, N_0$$

The previous equation represents a linear system which will be invertible if $\mathcal{B}^0$ is linearly independent (and it is because it comes from the solution of the linear problem) and all points of $\mathcal{X}^0$ are different.

4. Once we have computed $\{\varphi_1^0, \ldots, \varphi_{N_0}^0\}$, the interpolation of the nonlinear term reads:

$$\mathcal{F}_{NL}(u) \approx b_0 = -\sum_{i=1}^{N_0} \varphi_i^0 \cdot \phi_i^0$$

And therefore, the linearized problem writes:

$$\mathcal{L}(u) = b_0 \tag{2}$$

5. At this point, three options can be thought:

   (a) Restart the separated representation, i.e., find u^1 such that:

   $$\mathcal{L}(u^1) - b_0 = 0$$

   Applying the PGD method we will obtain a new reduced basis $\mathcal{B}^1 = \{\phi_1^1, \ldots, \phi_{N_1}^1\}$.

   (b) Reuse the solution u^0, i.e. $u^1 = u^0 + \widetilde{u}$. Then, we seek $\widetilde{u}$ such that:

   $$\mathcal{L}(\widetilde{u}) = b_0 - \mathcal{L}(u^0)$$

   We solve this problem by applying the PGD method, i.e. $\widetilde{\mathcal{B}} = \{\widetilde{\phi}_1, \ldots, \widetilde{\phi}_{\widetilde{N}}\}$ and then $\mathcal{B}^1 = \mathcal{B}^0 \oplus \widetilde{\mathcal{B}}$ and $N_1 = N_0 + \widetilde{N}$.

   (c) Reuse by projecting. In this case we consider

   $$u^{0,1}(\boldsymbol{x}) = \sum_{i=1}^{N_0} \eta_i^0 \cdot \phi_i^0(\boldsymbol{x})$$

   which introduced into Eq. (2) allows computing coefficients η_i^0. Then the approximation is enriched by considering $u^1 = u^{0,1} + \widetilde{u}$. In this case, we seek $\widetilde{u}$ such that:

   $$\mathcal{L}(\widetilde{u}) = b_0 - \mathcal{L}(u^{0,1})$$

   Once this problem is solved by applying the PGD method, i.e. $\widetilde{\mathcal{B}} = \{\widetilde{\phi}_1, \ldots, \widetilde{\phi}_{\widetilde{N}}\}$ and then $\mathcal{B}^1 = \mathcal{B}^0 \oplus \widetilde{\mathcal{B}}$ and $N_1 = N_0 + \widetilde{N}$.

6. From this point we repeat the precedent steps: let us assume that we have already computed u^k. Then select a set of points $\mathcal{X}^k = \{\boldsymbol{x}_1^k, \ldots, \boldsymbol{x}_{N_k}^k\}$, interpolate the nonlinear term using $\mathcal{B}^k$, and find u^{k+1}, until a certain convergence criteria is reached.

3   ELECTION OF THE INTERPOLATION POINTS

An open question is how to choose the points $\mathcal{X}^k = \{\boldsymbol{x}_1^k, \ldots, \boldsymbol{x}_{N_k}^k\}$. Consider that a certain computation step we know the reduced approximation basis:

$$\mathcal{B}^k = \{\phi_1^k, \ldots, \phi_{N_k}^k\}$$

Following [1, 4], we consider:

$$\boldsymbol{x}_1^k = \arg\max_{\boldsymbol{x}\in\Omega}|\phi_1^k(\boldsymbol{x})|$$

Then we compute c_1 from

$$c_1 \cdot \phi_1^k(\boldsymbol{x}_1^k) = \phi_2^k(\boldsymbol{x}_1^k)$$

which allows defining:

$$r_2(\boldsymbol{x}) = \phi_2^k(\boldsymbol{x}) - c_1 \cdot \phi_1^k(\boldsymbol{x})$$

from where we can compute the following point, $\boldsymbol{x}_2^k$ as:

$$\boldsymbol{x}_2^k = \arg\max_{\boldsymbol{x}\in\Omega}|r_2(\boldsymbol{x})|$$

As by construction $r_2(\boldsymbol{x}_1^k) = 0$, we can ensure that $\boldsymbol{x}_1^k \neq \boldsymbol{x}_2^k$. This process can be generalized and thus, if we are looking for $\boldsymbol{x}_j^k, j \leq k$, the following function can be constructed:

$$r_j(\boldsymbol{x}) = \phi_j^k(\boldsymbol{x}) - \sum_{i=1}^{j-1} c_i \cdot \phi_i^k(\boldsymbol{x})$$

where coefficients $c_i, 1 \leq i \leq j-1$, need to be computed. It can be done by imposing that:

$$r_j(\boldsymbol{x}_l^k) = 0 = \phi_j^k(\boldsymbol{x}_l^k) - \sum_{i=1}^{j-1} c_i \cdot \phi_i^k(\boldsymbol{x}_l^k) \ , \ l = 1, \ldots, j-1$$

that constitutes a linear system whose solution results the coefficients c_i. And then we compute the sought point:

$$\boldsymbol{x}_j^k = \arg\max_{\boldsymbol{x}\in\Omega}|r_j(\boldsymbol{x})| \tag{3}$$

It must be pointed out that, in principle, Eq. (3) implies reconstructing the solution, that is to say, to compute explicitly the functions $\phi_l^k, l = 2, \ldots, j-1$ from the separated functions $P_{l,s}^k(x_s)$ with $s = 1, \ldots, d$. For $l = 1$, notice that things are much simpler:

$$\boldsymbol{x}_1^k = \left(x_{1,1}^k, \ldots, x_{1,d}^k\right)$$

with

$$x_{1,s}^k = \arg\max_{x_s\in\Omega_s}|P_{1,s}^k(x_s)| \ , \ s = 1, \ldots, d$$

For $l > 1$ some simplifying procedures must be defined for avoiding the solution reconstruction and improve the performance in the multi-parametric case. The analysis of such procedures constitutes a work in progress.

4 NUMERICAL EXAMPLE

Aiming to prove the ability of the DEIM-based PGD method for solving nonlinear problems, we consider the transient heat equation with a quadratic nonlinearity:

$$\frac{\partial u}{\partial t} - \Delta u + u^2 = 0 \ , \ \ (\boldsymbol{x}, t) \in \Omega \times (0, T] \tag{4}$$

being $\Omega = [0, 1] \times [0, 1] \subset \mathbb{R}^2$. The initial condition reads $u(\boldsymbol{x}, t = 0) = 0$ and the boundary conditions are given by $u(x = 0, y = 0, t) = u(x = 1, y = 0, t) = 0$ and $\nabla u \cdot \boldsymbol{n} \, (x = 0.5, y = 1, t) = 1$. Outside these boundaries, a zero-flux condition is considered.

A space-time separated representation is sought in this case:

$$u(\boldsymbol{x}, t) = \sum_{i=1}^{N} X_i(\boldsymbol{x}) \cdot T_i(t) \tag{5}$$

We use here the reuse option, that is to say, the reduced basis is enriched without projection. Using the notation introduced in the previous section the convergence was reached after the construction of $k = 4$ reduced bases in which the nonlinear term was interpolated, for a relative error less than 1% to the reference solution. However, a relative error of 0.5% cannot be attained in spite of the number of basis enrichment. The final solution involved $N = 52$ functional products $\phi_i = X_i(\boldsymbol{x}) \cdot T_i(t)$. Fig. 1 compares the time evolution at different locations obtained with the DEIM based PGD and the exact solution. Then Fig. 2 and 3 depict the four fist space and time modes respectively. From these results we can conclude on the potentiality of the proposed technology for solving non-linear eventually multi-parametric models.

5 CONCLUSIONS

This work presents the DEIM-based PGD technique for solving efficiently reduced nonlinear models. The improvement is achieved by interpolating the nonlinear term using the reduced basis, computed as usual with the PGD method, instead of performing its complete evaluation. As the PGD is an *a priori* model reduction technique, a progressive reduced basis enrichment must be considered, and thus up to three different choices can be done: restart the reduced basis, reuse the previous reduced basis by enrichment and reuse the reduced basis by projecting. A deep analysis of the different alternatives is in progress.

REFERENCES

[1] Chaturantabut, S. and Sorensen, D.C. Nonlinear model order reduction via discrete empirical interpolation. *SIAM J. Sci. Comput.* (2010) **32**:2737--2764.

[2] Chinesta, F., Leygue, A., Bordeu, F., Aguado, J.V., Cueto, E., Gonzalez, D., Alfaro, I., Ammar, A. and Huerta, A. PGD-based computational vademecum for efficient design, optimization and control. *Arch. Comput. Methods Eng.* (2013) **20**:31--59

[3] Pruliere,E., Chinesta, F. and Ammar, A. On the deterministic solution of multidimensional parametric models by using the Proper Generalized Decomposition. *Mathematics and Computers in Simulation* (2010) **81**:791--810.

[4] Barrault, M., Maday, Y., Nguyen, N.C. and Patera, A.T. An "empirical interpolation method": application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathematique* (2004) **339/9**:667--672.

[5] Chinesta, F., Lavedeze, P. and Cueto, E. A short review in Model Order Reduction based on Proper Generalized Decomposition. *Archives of Computational Methods in Engineering* (2011) **18**:395--404, 2011.

[6] Ammar, A., Mokdad, B., Chinesta, F. and Keunings, R. A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modeling of compex fluids. Part II: Transient simulation using space-time separated representation. *Journal of Non-Newtonian Fluid Mechanics* (2007) **144**:98--121.
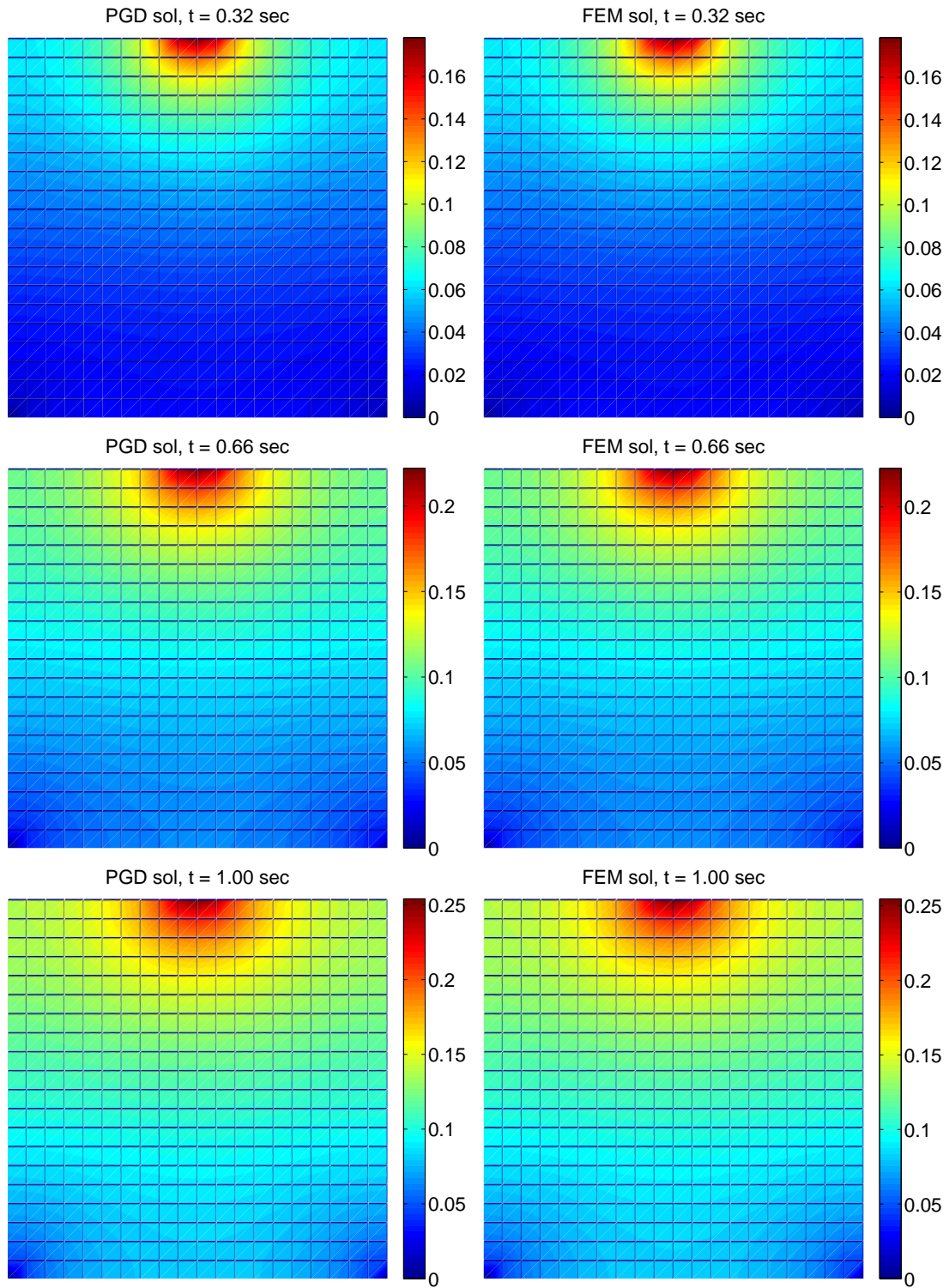
Figure 1: Comparison of the DEIM-based PGD solution (on the left) to the FEM reference solution (on the right), for three different times, $t = 0.32, 0.66, 1.00$ sec
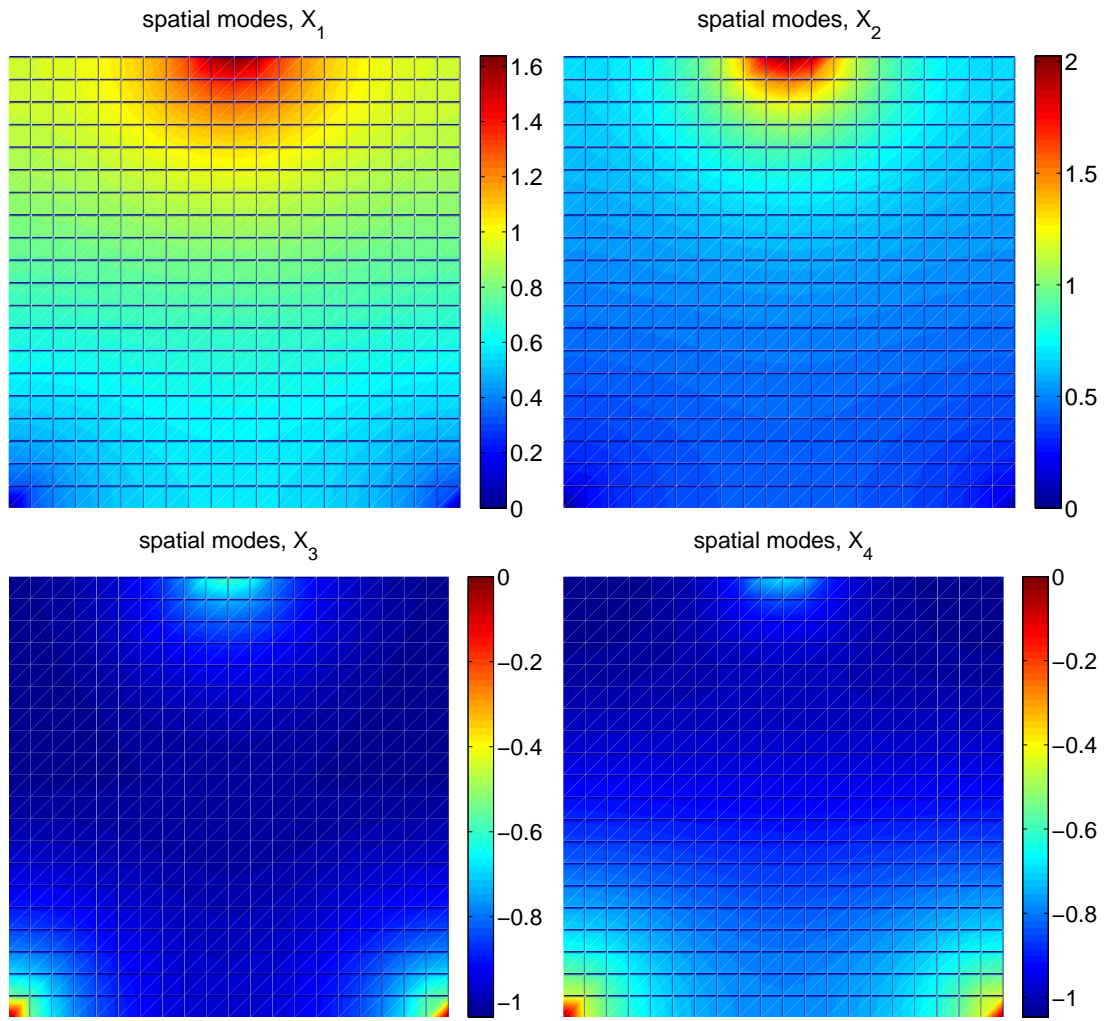
Figure 2: First four space normalized modes of the DEIM-based PGD solution
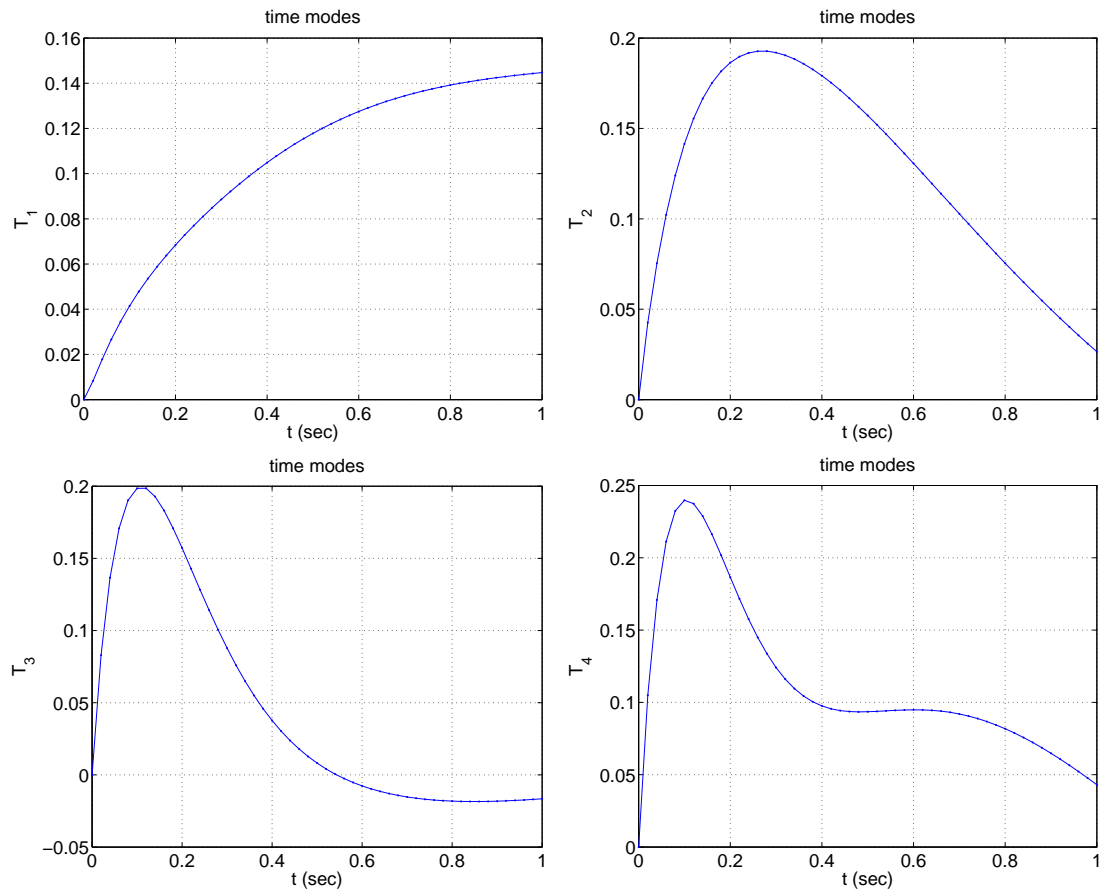
Figure 3: First four time normalized modes of the DEIM-based PGD solution

AN UNSTRUCTURED FINITE VOLUME SOLVER FOR A NEW CONSERVATION LAW IN FAST TRANSIENT DYNAMICS

M. Aguirre[*], A.J. Gil[*], J. Bonet[*] and C.H. Lee[*]

[*] Civil and Computational Engineering Centre, College of Engineering
Swansea University, Singleton Park, SA2 8PP, United Kingdom
e-mail: {m.aguirre,a.j.gil,j.bonet,c.h.lee}@swansea.ac.uk

Key words: Structural Dynamics, Conservation laws, Finite Volume, Volumetric Locking, JST

Abstract. Since the advent of computational mechanics, the numerical modelling of fast structural dynamics has been a major field of interest in industry. Traditionally, a Lagrangian formulation is employed for the numerical simulation of these problems and low order spatial interpolation is preferred for computational workload convenience. The well known second order solid dynamics formulation, where the primary variable is the displacement, is typically discretised in space by using the Finite Element Method (FEM) and discretised in the time domain by means of a Newmark (trapezoidal) time integrator. However, it has been reported that the resulting space-time discretised formulation presents a series of shortcomings. From the time discretisation point of view, the Newmark method has a tendency to introduce high frequency noise in the solution field, especially in the vicinity of sharp spatial gradients. From the space discretisation point of view, the use of isoparametric linear finite elements leads to second order convergence in displacements, but only first order convergence for stresses and strains. It is also known that these elements exhibit locking phenomena in incompressible or nearly incompressible scenarios. Recently, a new mixed methodology [1] has been developed in the form of a system of first order conservation laws, where the linear momentum and the deformation gradient tensor are regarded as the two main conservation variables. The current paper presents the discretisation of these equations by using the Jameson-Schmidt-Turkel (JST) scheme. The paper will as well focus on comparing the obtained results against the ones obtained using other methodologies such as cell centred Finite Volume or Two Step Taylor Galerkin, Stream Upwind Petrov Galerkin (SUPG).

1 INTRODUCTION

The Lagrangian mixed formulation presented in [1] is in the form of a system of first order conservation laws, where the linear momentum and the deformation gradient tensor are regarded as the two main conservation variables. The current paper presents a new implementation using the Jameson-Schmidt-Turkel (JST) scheme [3] widely know within the CFD community which is based on the use of central differences an artificial dissipation. In order to adapt the scheme to the specificities of the problem, dissipation will only be added to the first equation. This will allow satisfying the compatibility conditions of the deformation mapping. Special care will be taken in the integration of the boundary fluxes, by the use of a weighted average of nodes at the boundary faces. The spatial discretisation will be combined with a two stages Total Variation Diminishing (TVD) Runge-Kutta time integrator in order to advance the solution in time. The displacements are integrated in time using a trapezoidal rule which, combined with a Lagrange multiplier minimisation procedure, ensure the conservation of angular momentum. An additional correction to the numerical dissipation of the linear momentum is as well added in order to ensure the conservation of that variable.

2 GOVERNING EQUATIONS

A mixed system of conservation laws was presented in [1] as

$$
\begin{aligned}
\frac{\partial \boldsymbol{p}}{\partial t} - \nabla_0 \cdot \boldsymbol{P} &= \rho_0 \boldsymbol{b} \\
\frac{\partial \boldsymbol{F}}{\partial t} - \nabla_0 \cdot (\boldsymbol{v} \otimes \boldsymbol{I}) &= \boldsymbol{0} \\
\frac{\partial E_T}{\partial t} + \nabla_0 \cdot \left(\boldsymbol{Q} - \boldsymbol{P}^T \boldsymbol{v} \right) &= 0
\end{aligned}
\tag{1}
$$

where $\boldsymbol{p} = \rho_0 \boldsymbol{v}$ is the linear momentum, ρ_0 is the material density, $\boldsymbol{v}$ is the velocity field, $\boldsymbol{b}$ is the body force per unit mass, $\boldsymbol{F}$ is the deformation gradient tensor, $\boldsymbol{P}$ is the first Piola-Kirchhoff stress tensor, E_T is the total energy per unit of undeformed volume, $\boldsymbol{I}$ is the identity tensor and ∇_0 describes the material gradient operator in undeformed space. The above system of equations can be rewritten in a more compact form, describing a first order hyperbolic system as

$$
\frac{\partial \boldsymbol{\mathcal{U}}}{\partial t} + \frac{\partial \boldsymbol{\mathcal{F}}_I}{\partial X_I} = \boldsymbol{\mathcal{S}}
\tag{2}
$$

where

$$
\boldsymbol{U} = \begin{pmatrix} p_1 \\ p_2 \\ p_3 \\ F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \\ E_T \end{pmatrix}, \quad \boldsymbol{\mathcal{F}}_I = \begin{pmatrix} -P_{1I}(\boldsymbol{F}) \\ -P_{2I}(\boldsymbol{F}) \\ -P_{3I}(\boldsymbol{F}) \\ -\delta_{I1}v_1 \\ -\delta_{I2}v_1 \\ -\delta_{I3}v_1 \\ -\delta_{I1}v_2 \\ -\delta_{I2}v_2 \\ -\delta_{I3}v_2 \\ -\delta_{I1}v_3 \\ -\delta_{I2}v_3 \\ -\delta_{I3}v_3 \\ Q_I - P_{iI}v_i \end{pmatrix}, \quad \boldsymbol{S} = \begin{pmatrix} \rho_0 b_1 \\ \rho_0 b_2 \\ \rho_0 b_3 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad \forall I = 1, 2, 3 \tag{3}
$$

Additionally, the conservation law (2) has to be supplemented with a constitutive model which satisfies the objectivity requirement as well as the relevant laws of thermodynamics.

3   SPACE DISCRETISATION

The JST is a vertex centred Finite Volume Method and, as such, requires the use of a dual mesh for the definition of control volumes. In this paper, the median dual approach for triangular meshes, as presented in [4] and [5], has been chosen. This approach constructs the dual mesh by connecting edge midpoints with element centroids in two dimensions (see Figure 1) and edge midpoints with face centroids and element centroids in three dimensions. Such a configuration ensures that only one node of the initial mesh exists in each control volume. For a given edge connecting nodes e and α an area vector is then defined as

$$
\boldsymbol{C}^{e\alpha} = \sum_{k \in \Gamma_{e\alpha}} A_k \boldsymbol{N}_k \tag{4}
$$

where $\Gamma_{e\alpha}$ is the set of facets belonging to edge $e\alpha$, A_k is the area of a given facet and $\boldsymbol{N}_k$ is the normal vector of the facet (see Figure 1).

On account of the definition of the dual mesh, the defined area vector satisfy $\boldsymbol{C}^{e\alpha} = -\boldsymbol{C}^{\alpha e}$. This area vector enables a substantial reduction in the computational cost when computing the boundary integral used in the Green Gauss divergence theorem (classical in FVM), since it saves an additional loop on facets. Consider a hyperbolic system of conservation laws generally written as

$$
\frac{\partial \boldsymbol{\mathcal{U}}}{\partial t} + \frac{\partial \boldsymbol{\mathcal{F}}_I}{\partial X_I} = \boldsymbol{0} \tag{5}
$$

where $\boldsymbol{U}$ is the vector of conserved variables and $\boldsymbol{\mathcal{F}}_I$ the flux vector. This set of equations

(a) (b)
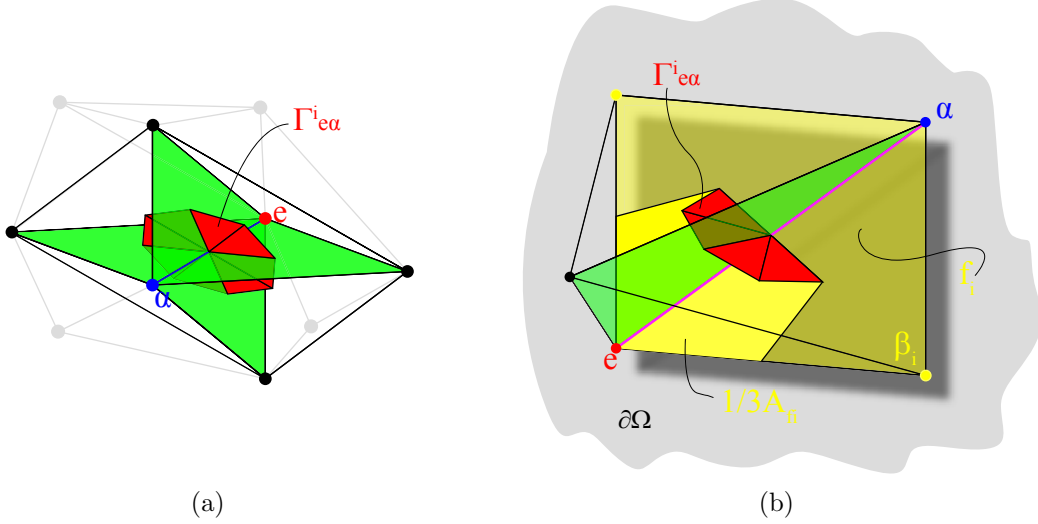
Figure 1: Set of facets related to an interior edge (left) and boundary edge (right) in three dimensions. The green surfaces correspond to the interior faces to which the edge belongs, whereas the dark yellow surfaces correspond to the boundary faces. The red surfaces are the set of interior facets $\Gamma_{e\alpha}$ corresponding to edge $e\alpha$. The bright yellow zone is the contributory area of the face $e\alpha\beta$ to node e.

can be discretised in space by using the extended JST scheme (see [4] and [5]) to give, for a given node e,

$$
\frac{d\boldsymbol{\mathcal{U}}_e}{dt} = -\frac{1}{V_e} \left(\sum_{\alpha \in \Lambda_e} \frac{1}{2}(\boldsymbol{\mathcal{F}}^e + \boldsymbol{\mathcal{F}}^\alpha) \boldsymbol{C}^{e\alpha} + \sum_{\substack{\alpha \in \Lambda_e^B \\ \beta \in (\Lambda_e^B \bigcap \Lambda_\alpha^B)}} \frac{6\boldsymbol{\mathcal{F}}^e + 6\boldsymbol{\mathcal{F}}^\alpha + 6\boldsymbol{\mathcal{F}}^\beta}{8} \boldsymbol{N}^{e\alpha\beta} A^{e\alpha\beta} \right) + \frac{1}{V_e}\boldsymbol{\mathcal{D}}(\boldsymbol{\mathcal{U}}_e)
$$

(6)

where Λ_e is the set of nodes connected to node e by an edge, Λ_e^B is the set of nodes connected to node e by a boundary edge, $\boldsymbol{\mathcal{F}}$ is a matrix gathering the flux vectors as $\boldsymbol{\mathcal{F}} = (\boldsymbol{\mathcal{F}}_1, \boldsymbol{\mathcal{F}}_2, \boldsymbol{\mathcal{F}}_3)$ and $\boldsymbol{D}(\boldsymbol{U}_e)$ is a dissipative operator. The first term of the equation is the actual Green Gauss evaluation of the cell boundary fluxes. The dissipative operator reads

$$
\boldsymbol{D}(\boldsymbol{\mathcal{U}}_e) = \sum_{\alpha \in \Lambda_e} \varepsilon_{e\alpha}^{(2)} \Psi_{e\alpha} \theta_{e\alpha} \left(\boldsymbol{\mathcal{U}}_\alpha - \boldsymbol{\mathcal{U}}_e \right) - \varepsilon_{e\alpha}^{(4)} \Psi_{e\alpha} \theta_{e\alpha} \left(\boldsymbol{L}(\boldsymbol{\mathcal{U}}_\alpha) - \boldsymbol{L}(\boldsymbol{\mathcal{U}}_e) \right)
$$

(7)

where $\varepsilon_{e\alpha}^{(2)}$ and $\varepsilon_{e\alpha}^{(4)}$ are discontinuity switches which activate second or fourth order differences operators, $\Psi_{e\alpha}$ is the spectral radius and $\theta_{e\alpha}$ denote geometrical weights. The second order differences operator is defined as

$$L(\mathcal{U}_e) = \sum_{\alpha \in \Lambda_e} \theta_{e\alpha}(\mathcal{U}_\alpha - \mathcal{U}_e) \tag{8}$$

The first term of the dissipative operator provides the second order differences operator and the second term the fourth order differences operator. The fourth order differences operator avoids the appearance of the odd-even decoupling of the solution (that would result from using averaged fluxes) whilst maintaining the second order accuracy of the scheme. The second order differences operator is introduced to smear out the solution in the vicinity of a shock whilst reducing the solution to first order locally.

4 TIME DISCRETISATION

The time discretisation is performed using a Total Variation Diminishing (TVD) Runge-Kutta time integrator as proposed by Shu and Osher [6]. For a set of equations discretised in space, but left continuous in time (method of lines) at a given node e,

$$\frac{d\mathcal{U}_e}{dt} = -\mathcal{R}_e(\mathcal{U}_e, t) \tag{9}$$

the method computes the solution at time step t^{n+1} from the solution at time step t^n as

$$
\begin{aligned}
\mathcal{U}_e^* &= \mathcal{U}_e^n - \Delta t \mathcal{R}_e(\mathcal{U}_e^n, t^n) \\
\mathcal{U}_e^{**} &= \mathcal{U}_e^* - \Delta t \mathcal{R}_e(\mathcal{U}_e^*, t^{n+1}) \\
\mathcal{U}_e^{n+1} &= \frac{1}{2}(\mathcal{U}_e^n + \mathcal{U}_e^{**})
\end{aligned}
\tag{10}
$$

In addition, the displacements are integrated in time using the trapezoidal rule, which combined with a Lagrange multiplier minimisation procedure, allows for the conservation of angular momentum.

5 NUMERICAL RESULTS

5.1 Punch test

A square flat plate of unite side length is constrained to move tangentially on the east, west and south sides, whereas it is free on the north side. The plate is subjected to an initial uniform velocity $v_{punch} = 100$ m/s on its right half side. The plate is composed of a NeoHookean rubber material with Young's modulus $E = 1.7 \times 10^7 Pa$, density $\rho_0 = 1.1 \times 10^3$ Kg/m^3 and Poissons ratio $\nu = 0.45$. The problem shows the performance of the method in nearly incompressible scenarios, with absence of volumetric locking and spurious modes (checker board) for the pressure. Figure 2 compares results obtained using Mean Dilatation technique and standard Finite Element Method (FEM) for the standard displacement based formulation and Cell Centred Finite Volume, Stream Upwind Petrov Galerkin (SUPG) and JST using the proposed formulation. It can be seen how the

FEM solution suffers from volumetric locking, while the Mean Dilatation technique is capable of circumventing it. In addition, both solutions exhibit spurious oscillations in the pressure distribution. The methodologies based in the proposed conservation law formulation alleviate both the volumetric locking and the appearance of the spurious pressure oscillations.

5.2 Two dimensional bending column

A rubber column of 1 m with and 6 m height is clamped on its bottom and subjected to an initial uniform horizontal velocity of $V_0 = 10$ m/s. The example shows the performance of the numerical technique in bending dominated scenarios. The material is set using a NeoHookean constitutive model with Young's modulus $E = 1.7 \times 10^7 Pa$, density $\rho_0 = 1.1 \times 10^3$ Kg/m^3 and Poissons ratio $\nu = 0.45$. Figure 3 shows the JST solution at different times as compared to SUPG and cell centred Finite Volume. All three solutions exhibit a smooth pressure distribution and absence of volumetric locking. Comparing the resolution of the three solutions, it can be seen that the JST method offers the most dissipative solution, whereas the SUPG gives the most accurate one.

5.3 Three dimensional bending column

This example is an extension of the two dimension column bar presented previously. The example is shown to prove the good performance of the method in three dimensional bending dominated scenarios. As in the two dimension case, a rubber material column is clamped on its bottom face ($X_3 = 0$ m), whereas is free on the rest. An initial uniform velocity $\boldsymbol{V}_0 = 10(\cos(30), \sin(30), 0)^T$ m/s is imposed and the bar is left oscillating in time. A Neohookean nearly incompressible material is chosen with Young's modulus $E = 1.7 \times 10^7 Pa$, density $\rho_0 = 1.1 \times 10^3$ Kg/m^3 and Poissons ratio $\nu = 0.45$. Figure 4 shows the evolution in time of the pressure distribution in the deformed configuration, comparing two different mesh discretisations, $h = 1/3$ m and $h = 1/12$ m. Both solutions exhibit a smooth distribution of pressure and absence of volumetric locking. Figure 5 shows the time history of the vertical displacement at point $\boldsymbol{X} = (1, 1, 6)^T$ m and pressure history at point $\boldsymbol{X} = (1/3, 1/3, 3)^T$ for three different space discretisations, $h = 1/3$ m, $h = 1/6$ m and $h = 1/12$ m . It can be seen the convergence of the solutions as the mesh is refined.

6 CONCLUSIONS

The JST scheme has been implemented for a new mixed conservation law in fast transient dynamics for triangular and tetrahedral meshes. The implementation has been carried out with special care to numerical stability, fulfillment of compatibility conditions and treatment of boundary conditions. This results in an adapted JST scheme, where the numerical dissipation is only added to the first equation (conservation of linear momentum) and the boundary conditions are treated using an external loop on faces, where
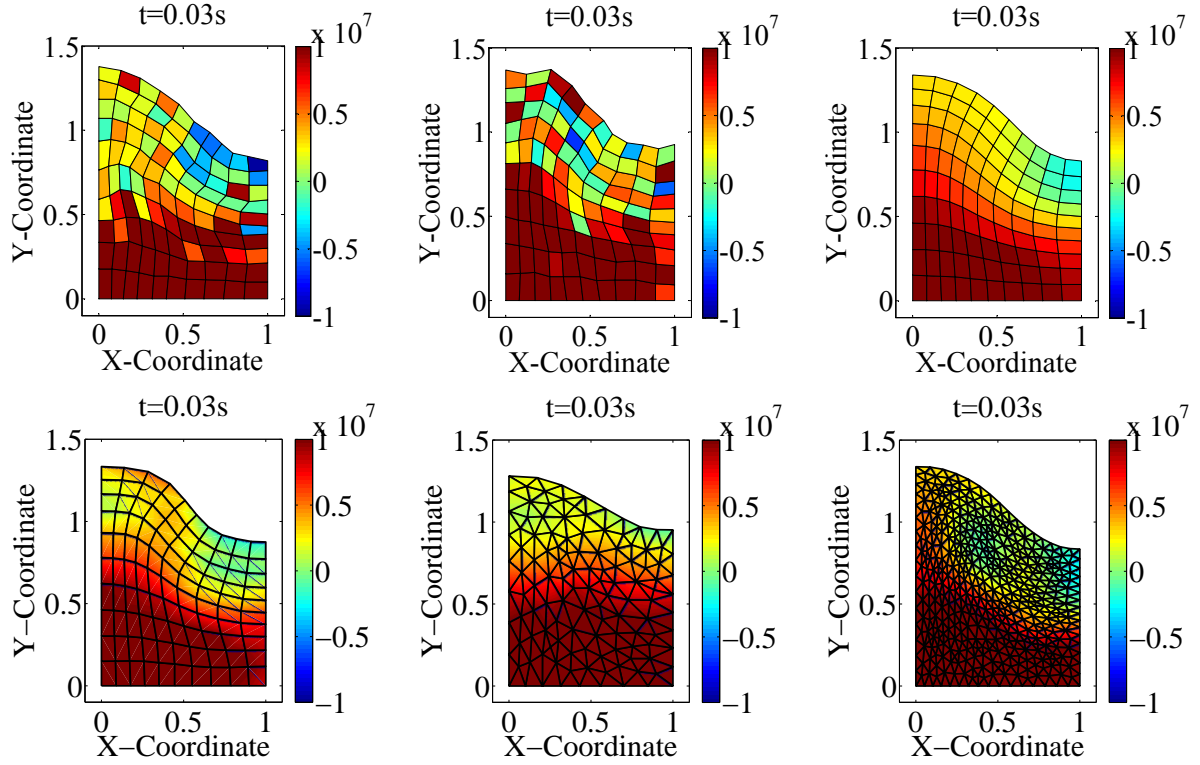
Figure 2: Numerical solution of the punch test case with an initial uniform velocity at the right hand side $v_{punch} = 100$ m/s. Material properties $E = 1.7 \times 10^7 Pa$, $\rho_0 = 1.1 \times 10^3 Kg/m^3$, $\nu = 0.45$ for a Neohookean material. The solution is shown at time $t = 0.03$s for different discretisations. From left to right and top to bottom: Mean dilatation technique, standard FEM, Cell Centred Finite Volume, SUPG and JST for the last two plots. The solution is obtained using a discretisation of 121 nodes for all the cases except for the last JST solution, where the number of nodes is doubled.
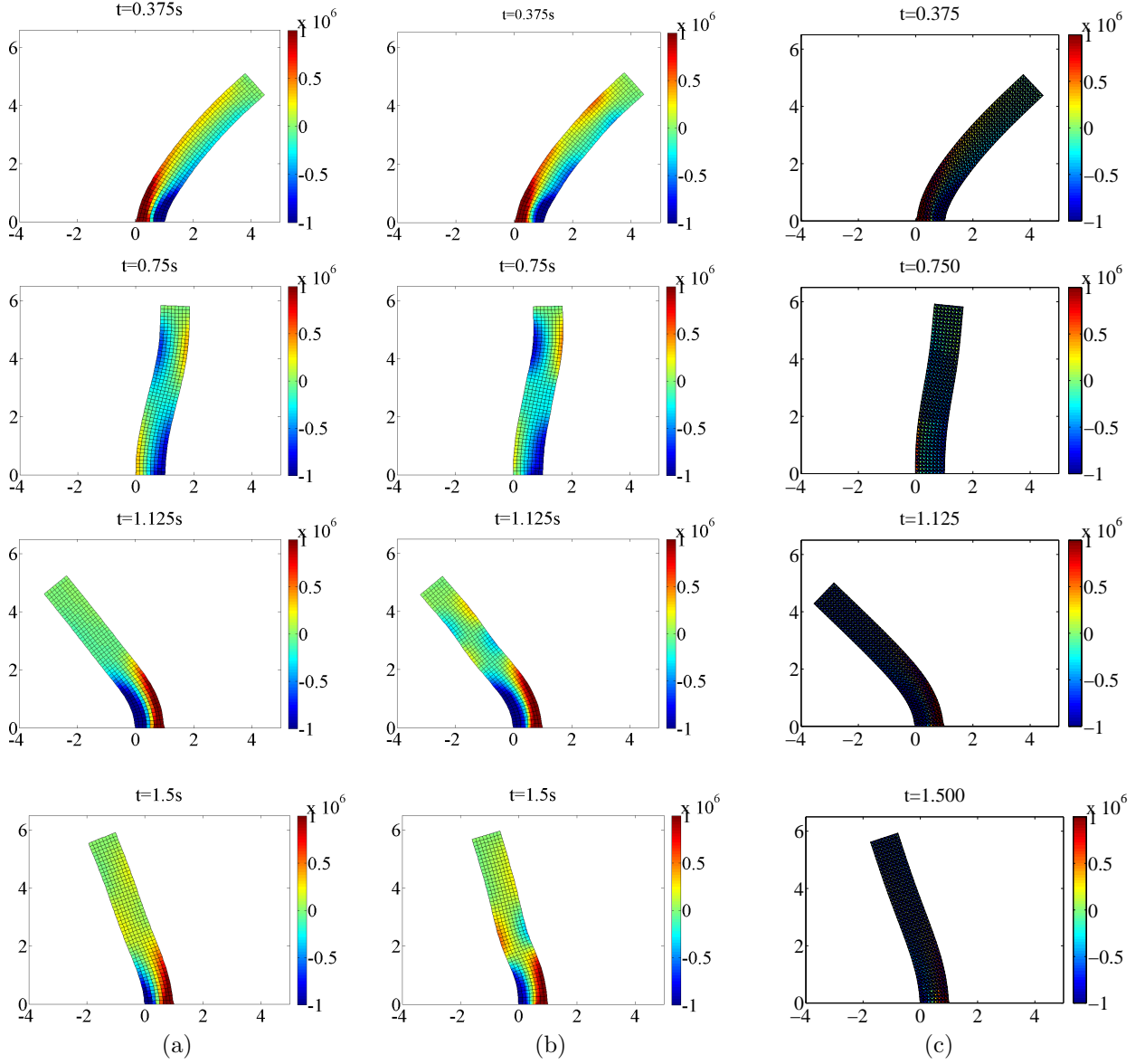
Figure 3: Cantilever Beam: Sequence of pressure distribution of deformed shapes using: (a) CCFVM imposing piecewise linear reconstruction; and (b) SUPG (consistent mass, $\tau_{\boldsymbol{F}} = \Delta t, \tau_{\boldsymbol{p}} = 0, \alpha = 0.05$) (c) JST ($\kappa^{(4)} = 1/64$). Results obtained with initial horizontal velocity $V_0 = 10$m/s. The nearly incompressible Neo-Hookean (NH) constitutive model is used such that Poisson's ratio $\nu = 0.45$, Young's modulus $E = 1.7 \times 10^7$Pa, density $\rho_0 = 1.1 \times 10^3$kg/m^3 and $\alpha_{CFL} \approx 0.4$. Discretisation of 8×48 quadrilateral elements. Time step $\Delta t = 1 \times 10^{-4}$s.
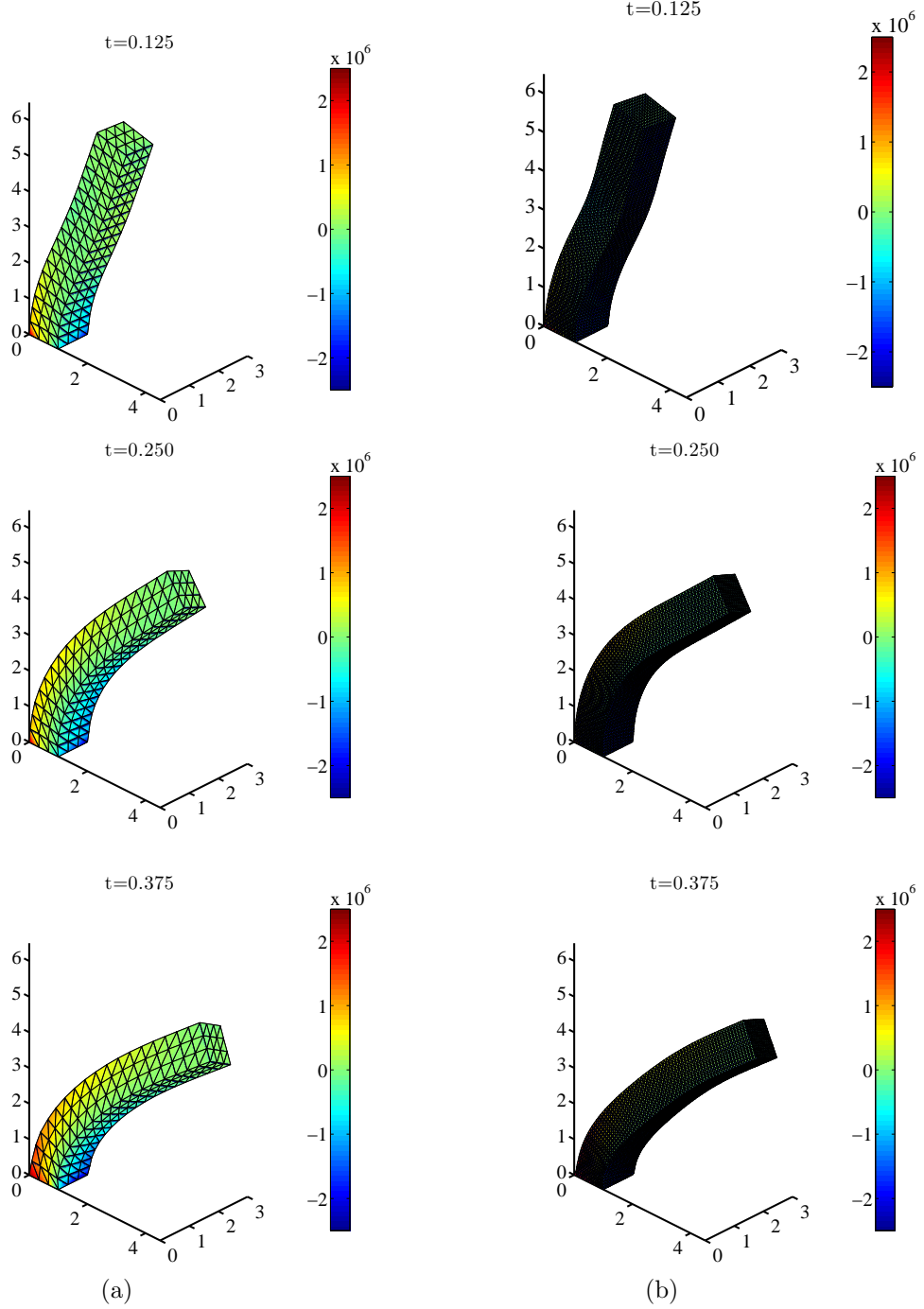
Figure 4: Three dimensional bending column. Evolution in time of the pressure distribution in the deformed configuration. Initial uniform velocity $\boldsymbol{V}_0 = 10(\cos(30), \sin(30), 0)^T$ m/s. Neohookean nearly incompressible material with Young's modulus $E = 1.7 \times 10^7 Pa$, density $\rho_0 = 1.1 \times 10^3$ Kg/m^3 and Poissons ratio $\nu = 0.45$. JST spatial discretisation with $h = 1/3$ m (a) and $h = 1/12$ m (b), $\kappa^{(4)} = 1/128$ and $\alpha_{CFL} = 0.4$.
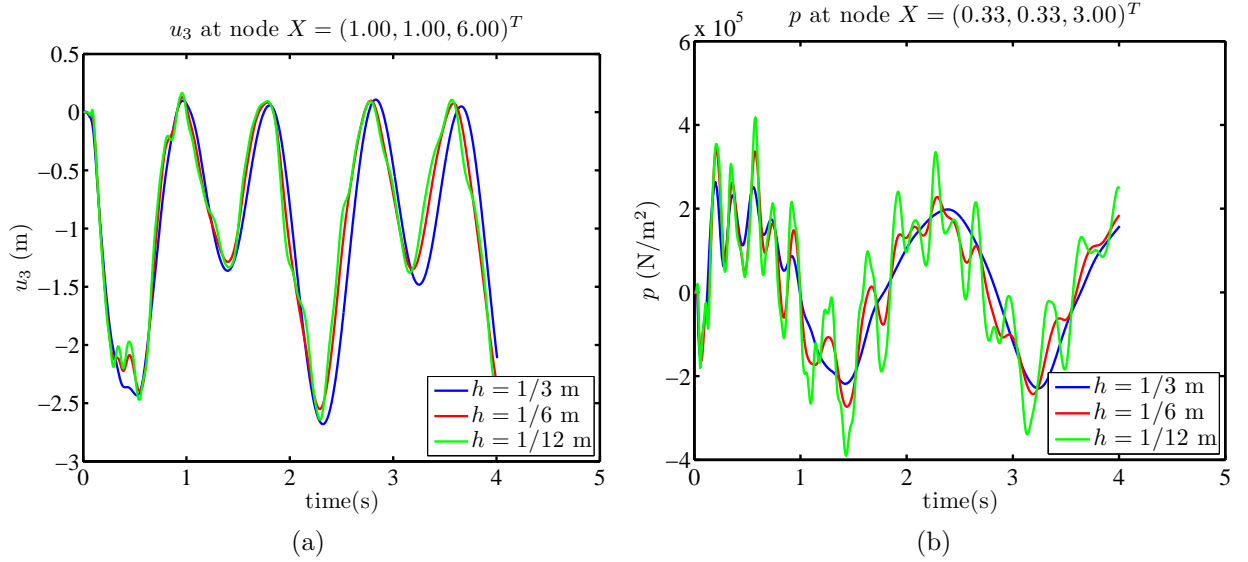
Figure 5: Three dimensional bending column. Time history of the vertical displacement at node $\boldsymbol{X} = (1, 1, 6)^T$ m (a) and time history of the pressure at node $\boldsymbol{X} = (1/3, 1/3, 3)^T$ m (b) . Initial uniform velocity $\boldsymbol{V}_0 = 10(\cos(30), \sin(30), 0)^T$ m/s. Neohookean nearly incompressible material with Young's modulus $E = 1.7 \times 10^7 Pa$, density $\rho_0 = 1.1 \times 10^3$ Kg/m^3 and Poissons ratio $\nu = 0.45$. JST spatial discretisation with $h = 1/3$ m (blue), $h = 1/6$ m (red) and $h = 1/12$ m (green), $\kappa^{(4)} = 1/128$ and $\alpha_{CFL} = 0.4$.

a weighted average of nodal values ensures accuracy and robustness of the solution. In addition, the numerical algorithm is modified to ensure preservation of linear and angular momentum. A set of numerical results has been presented both for two and three dimensions. These numerical results have proven second order convergence both for stresses and velocities. Furthermore, they circumvent the volumetric locking and spurious pressure modes as they appear in standard Finite Element (displacement based formulation) using linear elements for triangle and tetrahedra. The solutions compare well with other methodologies that discretise the proposed formulation, such as cell centred Finite Volume or Stream Upwind Petrov Galerkin. The proposed methodology allows for further research into more realistic real life problems. In fact, irreversible problems involving shocks can be easily implemented, due to the straightforward implementation of constitutive models and the built-in shock capturing term. Contact problems can as well be investigated by using Riemann-Solvers on the external faces.

REFERENCES

[1] C.H. Lee, A.J. Gil, J. Bonet *Development of a cell centred upwind finite volume algorithm for a new conservation law formulation in structural dynamics*, Computers and Structures (in press), 2012

[2] C.H. Lee, A.J. Gil, J. Bonet *Development of a New Stabilised Finite Element Methodology for the New Conservation Law Formulation in Lagrangian Fast Dynamics* Computers and Structures (under review)

[3] A. Jameson, W. Schmidt, Friedrichshafen, E. Turkel *Numerical solution of the Euler equations by Finite Volume methods using Runge-Kutta time-stepping schemes* AIAA 5th Computational Fluid Dynamics Conference 1259, 1982

[4] J. Blazek *Computational fluid dynamics: principles and applications* 2nd Edition, Elsevier, 2005

[5] K. A. Sørensen, O. Hassan, K. Morgan, N. P. Weatherill *A multigrid accelerated hybrid unstructured mesh method for 3D compressible turbulent flow* Computational Mechanics 31 101114, 2003

[6] C.-W. Shu, S. Osher *Efficient implementation of essentially non-oscillatory shock-capturing schemes* Journal of Computational Physics 77 (2) 439 471, 1988

A NEW RESIDUAL LEAST SQUARES ERROR ESTIMATOR FOR FINITE VOLUME METHODS – APPLICATIONS TO LAMINAR FLOWS

Duarte M. S. Albuquerque[*], **José M. C. Pereira**[*] **and José C. F. Pereira**[*]

[*]Instituto Superior Técnico (IST), IDMEC, LASEF
Universidade Técnica de Lisboa
Av. Rovisco Pais, 1, 1049-001, Lisbon, Portugal
e-mail: duartealbuquerque@ist.utl.pt, webpage: http://www.lasef.ist.utl.pt/

Key words: *h*-refinement, *a-posteriori* error estimation, Finite-Volume, incompressible flow, unstructured grids, least squares reconstruction

Abstract. Adaptive refinement is an important technique to reduce the computation time of flows in very refined meshes and increasing the local accuracy of the simulation.

A new *a-posteriori* error estimator, suitable for h-adaptive methods on unstructured grids, is based on the residual evaluation and a high-order polynomial reconstruction. The results are performed by the authors own Navier-Stokes code, which has been used to solve different adaptive problems [1, 2, 3].

The residual least squares (RLS) estimator is applied to different problems with a known analytic solution to study the numerical error decay with the adaptive algorithm and it is compared with the classic Taylor Series estimator [4, 5]. The proposed adaptive procedure is also applied to 3D flows around a sphere for two different types of grids.

The main goal of the present study is to perform the mesh refinement maintaining the global spatial accuracy to a desired level in the overall computational domain.

1 INTRODUCTION

The reduction of mesh generation effort and the computing time are of outmost relevance for CFD simulations of engineering fluid flow applications. Adaptive techniques reduce the time of the unstructured mesh generation and potencially the computing time, because the adaptive mesh has a smaller number of cells than the equivalent uniform mesh. Adaptive mesh refinement requires an estimator that shows the error distribution to refine locally the mesh. The information from the error estimator and the stop criterion should embody numerical accuracy and physical contrains of the numerical solution.

From the point of view of Finite Element Methods FEM the error estimators are well established and can be divided in three major groups: gradient recovery estimators,

explicit and implicit residual error estimates. A summary of different error estimators used in context of FEM was done by several authors [6, 7, 8, 9].

The Richardson Extrapolation [10] is the most popular error estimator in the finite-volume method (FVM) context and requires the solutions on two meshes with different spacings, which can be difficult to obtain in 3D industrial flow configurations. There are some attempts of single-mesh error estimators, based on energy conservation and angular moment conservation equations, see Haworth *et al.* [11]. Error estimators based on high order face interpolation was proposed by Muzaferija and Gosman [12] and later, Jasak and Gosman [4, 5] proposed an error estimator based in the Taylor series truncation error and another one based in the conservation of the second moment of the solution. Yahia *et al.* [13] has applied the Taylor series truncation to edges integral in the framework of r-adaptivity. Error estimator based on the residual error from the governing equations was investigated by Jasak and Gosman [14] and Juretic [15] extended it to a face based error estimator.

The Residual Least Squares error estimator has two main advantages, when compared to another approaches: the polynomial reconstruction made with the Least Squares method has the versatility required to be used in the case of unstructured grids which can have an arbitrary cell distribution and the Residual re-evaluation has information from the governing equations and the grid quality.

The adaptive grids are treated as unstructured grids, so the same convective and diffusive schemes are used, garantee second order error decay between the cells with different levels of refinement. In addition the new decision algorithm uses the computed information from error estimators without requiring any input parameter from the user or any previous knowledge of the numerical solution.

2 NUMERICAL METHOD

2.1 Governing Equations and Unstructured Grids Formulation

The steady isothermal flow of an incompressible fluid is governed by the mass and momentum conservation laws, being expressed by the incompressibility constraint and the Navier-Stokes equations:

$$\nabla \cdot \mathbf{u} = 0 \tag{1}$$

$$\nabla \cdot (\mathbf{u} \otimes \mathbf{u}) = \nabla \cdot \left(\nu \nabla \mathbf{u} + \nu \nabla^T \mathbf{u} \right) - \frac{1}{\rho} \nabla p \tag{2}$$

where $\mathbf{u}$ is the velocity vector, ν is the kinematic viscosity, ρ is the fluid's density and p is the fluid's pressure. The governing equations are discretized on unstructured meshes made of cells of arbitrary topology, to address the multiple faces that arise in interfaces between refined and non-refined cells. Each cell P is a polyhedron with a closed boundary ∂P which is composed by a variable number of faces $\partial P = \{S_1, S_2, ..., S_F\}$. Each face S_k is a plane polygon of arbitrary orientation which connects P and it's neighbour cell P_k,

see Figure 1. The computational points are located at the centers of each control volume corresponding to the so-called "collocated" or "non-staggered" arrangement.
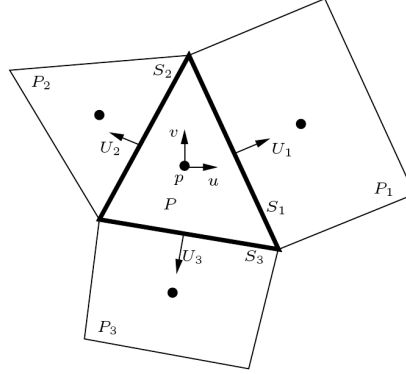
Figure 1: General 2D Polyhedral Control Volume

2.2 Pressure Velocity Coupling Algorithm

The SIMPLE algorithm [16] is used for the pressure velocity coupling. The SIMPLE algorithm starts by computing an approximate velocity field $\mathbf{u}^*$, which satisfies the momentum equations using the values from the previous iteration n. The steady equation is solved implicitly and linearization of the convection contribution is required:

$$\sum_{f=1}^{F} U_f^n \mathbf{u}_f^* - \nu \sum_{f=1}^{F} (\nabla \mathbf{u}^*)_f . \mathbf{S}_f = -\frac{V_P}{\rho} \nabla p^n \tag{3}$$

where U_f^n is the face velocity defined by $\mathbf{u}_f^n . \mathbf{S}_f$, $\mathbf{S}_f$ is the face surface vector defined by $S_f \mathbf{n}_f$, $\mathbf{n}_f$ is the normal unit vector of the face f and V_P denotes the cell P volume. A system of linear equations is assembled in this form:

$$\frac{1}{\alpha_u} a_p \mathbf{u}_p^* + \sum_{l=1}^{F} a_l \mathbf{u}_l^* = -\frac{V_P}{\rho} \nabla p^n + \frac{1-\alpha_u}{\alpha_u} a_p \mathbf{u}_p^n \tag{4}$$

being α_u the under relaxation factor for the momentum equations. The face velocity U_f^* is computed with Rhie-Chow interpolation [17]:

$$U_f^* = \overline{\mathbf{u}_p^*} . \mathbf{S}_f - \overline{\frac{\alpha_u V_P}{\rho \, a_p}} ((\nabla p^n)_f - \overline{(\nabla p^n)}) . \mathbf{S}_f \tag{5}$$

where the over-lined values are obtained by interpolation from the two cells which have the same face f and a_p are the momentum system matrix's main diagonal elements. This

face velocity is used to solve the pressure correction equation:

$$\sum_{f=1}^{F} \overline{\frac{\alpha_u V_P}{\rho\, a_p}} (\nabla p')_f . \mathbf{S}_f = \sum_{f=1}^{F} U_f^* \tag{6}$$

where p' is the pressure correction. After solving the Poisson equation 6, the velocity values are corrected with the p' new values to satisfy the continuity equation 1. The conservative face velocity is computed by:

$$U_f^{n+1} = U_f^* - \overline{\frac{\alpha_u V_P}{\rho\, a_p}} (\nabla p')_f . \mathbf{S}_f \tag{7}$$

the cell velocity is corrected by:

$$\mathbf{u}^{n+1} = \mathbf{u}^* - \overline{\frac{\alpha_u V_P}{\rho\, a_p}} (\nabla p') \tag{8}$$

and the pressure is updated:

$$p^{n+1} = p^n + \alpha_p p' \tag{9}$$

the SIMPLE algorithm requires an under relaxation factor for pressure α_p. From this point the residuals are computed, If they are lower than a prescribed value the cycle ends, if not the computation advances to the next iteration, back to equation 4.

2.3 Numerical Schemes

The author's own code SOL has the capability to make different types of regressions from the discrete cell values by solving a Weighted Least Squares (WLS) problem. Different types of polynomials and cell sets can be used in these regressions. Figure 2 shows examples of cell sets composed by different types of cell neigbours. The subfigure 2(a) shows the first and second cell's neighbours by face in a cartesian grid and the subfigure 2(b) shows the first and second cell's neighbours by vertex in a grid made by triangles.

The versatility of the regressions is suitable to compute the diffusive and convective values of the arbitrary unstructured cells and achieve second order accuracy, see Kobayashi *et al.* [18, 19] for details. Since the face regressions are centered in the face centroid, they can deal with the severe orthogonality and skewness deviations which exist in the interfaces between refined and non-refined cells, increasing the accuracy of the diffusive and convective schemes.

All the least squares regressions use a weight function W_P, given by the inverse square of the distance:

$$w_P = |x_P - x_{ref}|^{-2} \tag{10}$$

where x_P is the cell P centroid coordinates and x_{ref} is the coordinates of the regression reference.
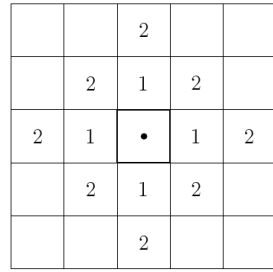
(a) Cell Neibours by Face   (b) Cell Neibours by Vertex
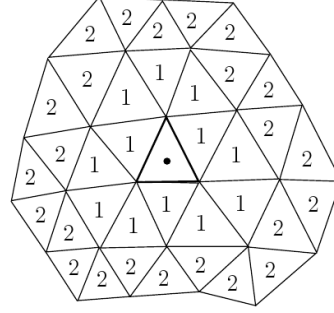
Figure 2: Different Examples of Cell Neibours

Both diffusive and convective values are computed with a single regression, using a linear polynomial and the various cells that have the face's vertices. Figure 3 shows examples of different computational molecules used in the numerical schemes. Where the cells used for each regression k of the face S_k are marked with the respective number k.
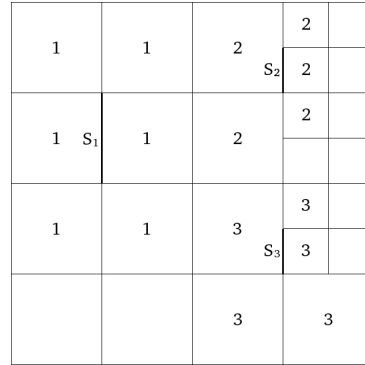
Figure 3: Possible Stencils used in the Convective and Diffusive Terms Computation

To achieve the second order integration with the finite volume method the regression is centred at the face's centroid.

The convective and diffusive schemes used in this work may originate a non positive definite matrices, so a deferred correction approach (Ferziger [20]) was used where a stable scheme is computed implicitly and the least squares scheme is computed explicitly. As stable schemes, the central differences scheme is used for the diffusive term and the convective fluxes are approximated by the first order upwind.

3 ADAPTIVE ALGORITHM AND ERROR ESTIMATORS

3.1 Adaptive Algorithm

Error estimators are required for the adaptive decision algorithm and they give a good approximation of the numerical error and its distribution in the computational domain. With this information is possible to compute an estimation of the ideal hydraulic radius h_i distribution for a desired local error E_0. For a second order method in space, the following equations are valid:

$$|E| \sim Kh^2 \tag{11}$$

$$|E_0| \sim Kh_i^2 \tag{12}$$

where E is the error estimation and K is an unknown constant. After some algebraic manipulation:

$$h_i \sim h\sqrt{E_0/E} \tag{13}$$

Ideally, the formula 13 can be used to create adaptive grids with approximately constant error if combined with an automatic grid generator.

The adaptive procedure used in this work is based in the maximum value of the error estimator. The cells with an error higher than $\lambda \, max(|E|)$ are selected for h-refinement, where λ is a factor that depends of the method's order. In the case of second order method this value λ is equal to 0.25, which is the reduction factor $(h_L/h_{L+1})^2$ of the local error, in each grid refinement.

3.2 Taylor series truncation error

The Taylor series error estimator [4] is derived by the 2nd order term of the Taylor series and it is computed by:

$$E_T = \frac{1}{2V_P} \left| \left(\frac{\partial^2 \phi}{\partial x_i \partial x_j} \right)_P \right| (M_{ij})_P \tag{14}$$

where $(M_{ij})_P$ is the inertia tensor of the cell P. The Hessian matrix values are computed from a regression made with a 2nd order polynomial from the cell's first and second neighbours. Due to the assumption of linear variation inside the computational cells, zones with lower errors will have lower values of the Hessian matrix.

3.3 Residual Least Squares

A regression is done with a 3rd order polynomial and considering the cell's first and second neighbors. New face's values and gradients are computed and compared with the values from the convection and diffusion schemes. One way to do this, is by recomputing new residual values, which indicate if the values satisfies the governing equations. The Residual Least Squares (RLS) vector is computed for each cell by the following formula:

$$\mathbf{E}_R = \frac{\sum_{f=1}^{F} U_f^n \mathbf{u}_f - \nu \sum_{f=1}^{F} (\nabla \mathbf{u})_f . \mathbf{S}_f + \dfrac{V_P}{\rho} \nabla p^n}{a_p} \qquad (15)$$

where the values $\mathbf{u}_f$ and $(\nabla \mathbf{u})_f$ are computed from the 3rd order polynomial, a_p is the matrix value used for the momentum equations, which is required to give the RLS error the same dimensions of the dependent variable. This error gives the indication for local refinement if the differences between 3rd order profile and the numerical discretization affect the governing equations. Unlike the Taylor series the RLS criteria depends on the governing equations discretization and the grid quality.

4 RESULTS

4.1 Poisson Equation in a L-Shaped Domain

For this test case the Poisson equation $\nabla^2 \phi = 0.0$ was solved in a L-shaped domain $[-1, 1]^2 \setminus ([0, 1] \times [-1, 0])$ Dirichlet boundary condition is prescribed in all boundaries and the analytic solution is given by the following equation:

$$\phi(x, y) = r^{2/3} sin(2\varphi/3) \quad with \quad (x, y) = r(cos\, \varphi, sin\, \varphi) \qquad (16)$$

The computations started with a grid of 12 triangles, three types of refinement are applied to this grid: one with uniform refinement and other two with the adaptive procedure using the classic Taylor series or the RLS as error estimators. The goal is to study the main differences between the two error estimators and evaluate their effectiveness. Figure 4 shows the mean and maximum error for the three types of grids, after 15 levels of refinement for the Taylor series and 22 levels of refinement for the RLS errors estimators:
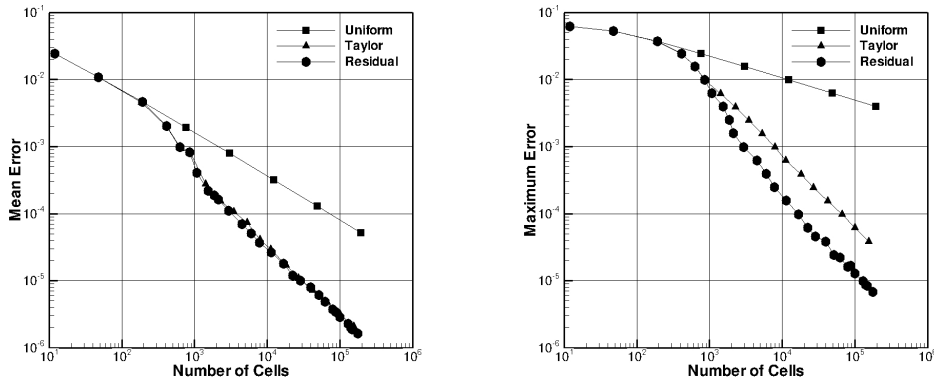
(a) Mean Error over Number of Cells   (b) Maximum Error over Number of Cells

Figure 4: Poisson Equation: Mean and Maximum Error for the Uniform and Adaptive Grids

For the uniform grid case, the mean and maximum error slope has an order of 4/3 and 2/3, respectively. Although, at the singularity point $(x, y) = (0, 0)$ the analytic solution is zero, the analytic gradient is infinite which causes the method to have an order accuracy lower than 2 for uniform grids.

For both error estimators, the mean error of the adaptive grids has second order slope, due to the difference between the slopes of the adaptive and uniform grids, the mean error from the adaptive grids is much lower than the mean error from the uniform grid.

In the subfigure 4(b), the maximum error for the three types of grids is shown. The adaptive grid with the TS estimator has an maximum error 100 times lower than the error of the uniform grid and the adaptive grid with the RLS estimator shows an maximum error 1000 times lower, showing an improvement when compared with TS estimator. The ratio between the maximum and mean error, which is a measurement of adaptivity efficiency, is 0.053 for the TS estimator and 0.2336 for the RLS estimator.

Figure 5 shows the final adaptive grids obtained with the TS and RLS error estimators. The adaptive grid with TS has more refined cells and a circular pattern in the grid interface, this happens due to the loss of accuracy of the TS error estimator after some adaptive levels. The adaptive grid with RLS has a lower number of cells and a rectangular pattern is observed, there is an increase in the error estimator accuracy as it was observed in the subfigure 4(b), there is an over estimation of the numerical error in the boundaries of the computational domain.
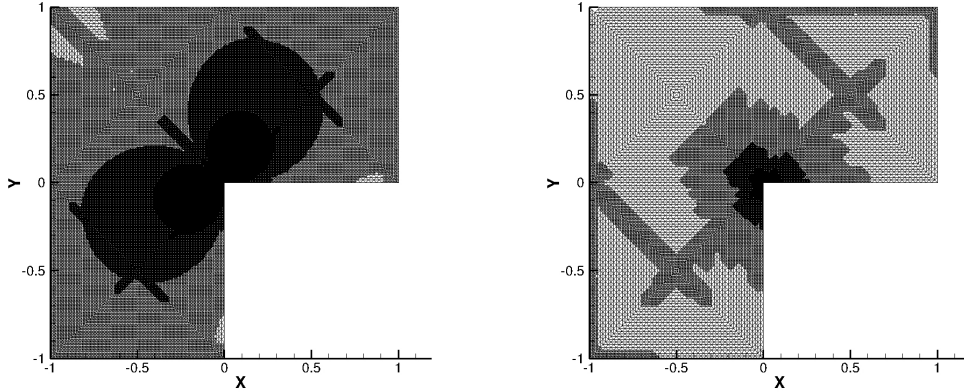
(a) Adaptive Grid using the Taylor Estimator (b) Adaptive Grid using the RLS Estimator

Figure 5: Poisson Equation: Adaptive Grids for the Taylor and RLS Estimators

4.2 Convection-Diffusion Equation - Point Source in Cross-Flow

For this test case, the convection-diffusion equation $U\dfrac{\partial\phi}{\partial x} = \Gamma\nabla^2\phi$ is solved. The selected analytical solution was used in previous works by Jasak and Gosman [4, 5] and is defined by:

$$\phi(x, y) = \frac{S}{2\pi\Gamma}K_0\left(\frac{U\sqrt{x^2 + y^2}}{2\Gamma}\right)e^{(0.5xU/\Gamma)} \qquad (17)$$

where $S = 16.67\,[\phi/s]$ is the source magnitude, $\Gamma = 0.05\,[m^2/s]$ is the diffusion coefficient, $U = 1.0\,[m/s]$ is the imposed velocity in the x axis and K_0 is the modified Bessel function of the second kind and zero order.

This problem is solved in a rectangular domain $[0.0, 4.0] \times [-0.5, 0.5]$, the line-source is located at $0.05\,m$ of the left boundary to avoid numerical problems from this singularity. Dirichlet boundary conditions are prescribed in all boundaries, except for the right boundary $(x = 4.0)$, where a null gradient is imposed.

The same refinement test were done for this solution, a Cartesian grid of 16×4 is used as initial grid and the adaptive algorithm is used until 20 levels of refinement are reached. The mean and maximum error, for the uniform and adaptive grids are shown in figure 6.
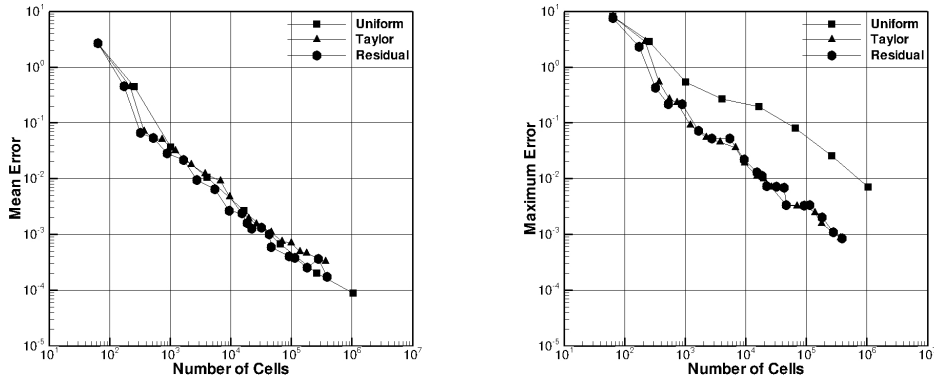
(a) Mean Error over Number of Cells (b) Maximum Error over Number of Cells

Figure 6: Line Source: Mean and Maximum Error for Uniform and Adaptive Grids

The curves of the mean error (subfigure 6(a)) show the same slope for the three grids, the adaptive grids doesn't show any improvement in the mean error when compared with the uniform refinement. For both error estimators, the maximum error of the adaptive grids is lower than for the uniform grid case. The final adaptive grid has a maximum error 10 times much lower than for the case of the uniform grid.

In this case, the error slope is not always constant due to the grid interface correction, which prevents the accumulation of the grid interfaces between different levels of

refinement, avoiding the loss of the grid quality and the solution overall accuracy.

When comparing the results from the subfigure 6(b) with the ones obtained by Jasak and Gosman [5] there is a significant improvement. This happens due to the grid interface correction and the different decision algorithm of the cells for refinement.

4.3 Flow over a Sphere

The three dimension flow over a sphere is computed as the final test of the RLS estimator and the adaptive code, two initial meshes were made one with hexahedrons and another one with tetrahedrons. The initial hexahedron grid has 46800 cells and its domain has a cylinder form, the computational domain of the initial tetrahedral grid is a squared prism with $39 \times 13 \times 13$ diameters and has 126182 cells.
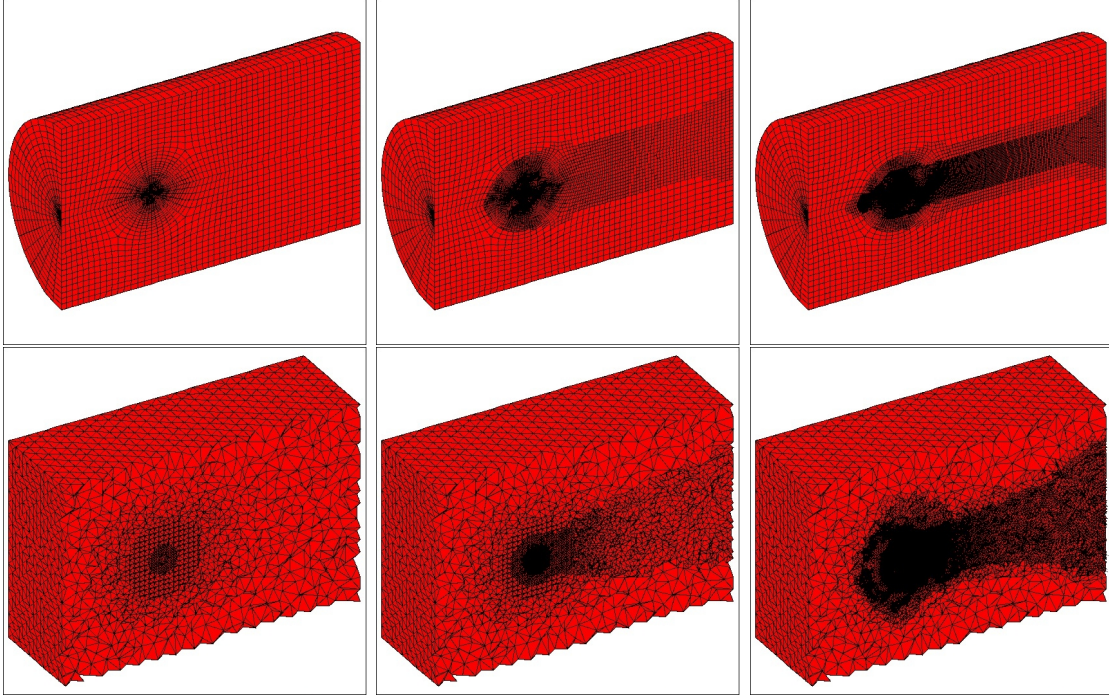
Figure 7: Example of Refinement in the Sphere Flow with Hexahedral and Tetrahedral Grids

Figure 7 shows the adaptive grids for two levels of refinement, with both the hexahedral and tetrahedral grids, for a Reynolds number of 200. The final meshes have 1331216 and 2707026 cells, respectively, which corresponds to a mesh with less 55.555% and 66.479% than compared to uniform refinement case. Both adaptive grids are refined near the sphere wall and in the flow's wake which are the primary features of this problem. The cone formed by the refined cells in the flow's wake is bigger in the tetrahedral grid, since the cells far away of the sphere have a higher hydraulic radius than in the hexahedral grid.

5   CONCLUSIONS

- The Residual Least Squares (RLS) error estimator has been shown to be suitable for adaptive refinement of Finite-Volume methods on unstructured grids. Unlike other error estimators, the RLS has information of the governing equations discretization and information of the grid quality.

- The new adaptive decision algorithm is independent on user defined parameters and can deal with the problem of the grid quality loss in the cell interface, making it more suitable than other algorithms from the literature.

ACKNOWLEDGMENTS

REFERENCES

[1] J. P. Magalhães, J. M. C. Pereira, and J. C. F. Pereira. A new refinement criterion based on regression diagnostics for finite volume methods. *Int. Conf. on Adaptive Modeling and Simulation (ADMOS), Goteborg*, 2007.

[2] D. M. S. Albuquerque, J. M. C. Pereira, and J. C. F. Pereira. Refinement least squares regression criteria applied to laminar flows. *Int. Conf. on Adaptive Modeling and Simulation (ADMOS), Paris*, 2011.

[3] J. P. Magalhães, D. M. S. Albuquerque, J. M. C. Pereira, and J. C. F. Pereira. Adaptive mesh finite-volume calculation of 2d lid-cavity corner vortices. *J. Comp Phys., Accepted for Publication*, 2013.

[4] H. Jasak and A. D. Gosman. Automatic resolution control for the finite volume method, part 1: A-posteriori error estimates. *Numer. Heat Transfer, Part B*, 38(3):237–56, 2000.

[5] H. Jasak and A. D. Gosman. Automatic resolution control for the finite volume method, part 2: Adaptive mesh refinement and coarsening. *Numer. Heat Transfer, Part B*, 38(3):257–71, 2000.

[6] M. Ainsworth and J. T. Oden. A posteriori error estimation in finite element analysis. *Comput. Methods Appl. Mech. Eng.*, 141:1–88, 1997.

[7] K. J. Fidkowski and D. L. Darmofal. Review of output-based error estimation and mesh adaptation in computational fluid dynamics. *AIAA Journal*, 49(4):673–694, 2011.

[8] T. Gratsch and K. Bathe. A posteriori error estimation techniques in practical finite element analysis. *Computers and Structures*, 83:235–265, 2005.

[9] K. Segeth. A review of some a posteriori error estimates for adaptive finite element methods. *Mathematics and Computers in Simulation*, 80:1589–1600, 2008.

[10] M. J. Berger and J. Oliger. Adaptive mesh refinement hyperbolic partial differential equations. *J. Comp Phys.*, 53:484–512, 1984.

[11] D. C. Haworth, E. L. Thary, and M. S. Huebler. A global approach to error estimation and physical diagnostics in multidimensional fluid dynamics. *Int. J. Numer. Meth. Fluids*, 17:75–97, 1993.

[12] S. Muzaferija and D. Gosman. Finite-volume CFD procedure and adaptive error control strategy for grids of arbitrary topology. *J. Comp Phys.*, 138:766–787, 1997.

[13] Djaffar Ait-Ali-Yahia, Guido Baruzzi, Wagdi G. Habashi, Michel Fortin, Julien Dompierre, and Marie Gabrielle Vallet. Anisotropic mesh adaptation: towards user-independent, mesh-independent and solver-independent CFD. part i: general principles. *International Journal of Numerical Methods in Fluids*, 32(6):725–744, 2002.

[14] H. Jasak and A. D. Gosman. Residual error estimate for the finite-volume method. *Numer. Heat Transfer, Part B*, 39(1):1–19, 2001.

[15] F. Juretic. Error analysis in finite volume CFD. *PhD thesis, Imperial College, University of London*, 2004.

[16] S. V. Patankar and D. B. Spalding. A calculation procedure for heat, mass and momentum transfer in three dimensional parabolic flows. *Int. Journal Heat and Mass Transfer*, 15:1787, 1972.

[17] C. M. Rhie and W. L. Chow. Numerical study of the turbulent flow past an airfoil with trailing edge separation. *AIAA Journal*, 21:1525–1532, 1983.

[18] M. Kobayashi, J. M. C. Pereira, and J. C. F. Pereira. A second-order upwind least-squares scheme for incompressible flows on unstructured hybrid grids. *Num. Heat Transfer B*, 34:39–60, 1998.

[19] M. Kobayashi, J. M. C. Pereira, and J. C. F. Pereira. A conservative Finite-Volume second-order accurate projection method on hybrid unstructured grids. *J. Comp. Phys.*, 150:40–75, 1999.

[20] J. H. Ferziger and M. Peric. *Computational method for Fluid Dynamics*. Springer-Verlag, Berlin/New York, 1996.

VI International Conference on Adaptive Modeling and Simulation
ADMOS 2013
J. P. Moitinho de Almeida, P. Díez, C. Tiago and N. Parés (Eds)

A CLASS OF PIOLA-KIRCHHOFF HYBRID STRESS FINITE ELEMENTS FOR ELASTODYNAMICS

J.P. MOITINHO DE ALMEIDA[*] AND HUGO A.F.A. SANTOS[†]

[*] Department of Civil Engineering, Architecture and Georesources
Instituto Superior Técnico
Av. Rovisco Pais 1, 1049 -001 Lisboa, Portugal
e-mail: moitinho@civil.ist.utl.pt

[†] Wolfson School of Mechanical and Manufacturing Engineering
Loughborough University
Loughborough, Leicestershire LE11 3TU, U.K.
email: hugofreixialsantos@gmail.com

Abstract. We introduce a novel hybrid stress finite element formulation for two-dimensional linear electrodynamics. This approach is an extension of the Piola-Kirchhoff hybrid stress formulation that we have recently proposed for linear elastostatics [1], and is applied in this communication to problems in the frequency domain. The formulation is consistent with a complementary form of the Hamiltonian variational principle, which involves, as fundamental unknown variables, the stress field components and boundary displacements. The approximate stress fields are split into two parts: a divergence-free (static) part, taken as the solution of the homogeneous momentum equations, and a dynamic part, taken as the particular solution of the momentum equations. The key ingredient of the formulation is to explicitly approximate, in the parent domain, either the second Piola-Kirchhoff stresses, the first Piola-Kirchhoff stresses, the Cauchy stresses, or rather their combination, and to enforce the divergence-free condition in the physical domain by means of a suitable projection. The main advantage of this formulation over traditional hybrid stress formulations [2] is that it allows to consider arbitrarily shaped elements without necessarily compromising static admissibility. Feasibility and effectiveness of the proposed formulation will be numerically demonstrated through the analysis of benchmark tests, which will also consider the problem of characterizing the properties of the error of these solutions.

REFERENCES

[1] Santos, H.A.F.A. and Moitinho de Almeida, J.P. A family of Piola-Kirchhoff hybrid stress finite elements for two-dimensional linear elasticity. *International Journal for Numerical Methods in Engineering* (submitted for publication), 2012.

[2] Almeida, J.P.M. and Freitas, J.A.T. Alternative approaches to the formulation of hybrid equilibrium finite elements. *Computers and Structures* (1991), **40**:1043–1047.

THREE DIMENSIONAL RE-MESHING FOR REAL TIME MODELING OF ADVANCING PROCESS IN MECHANIZED TUNNELING

Abdullah ALSAHLY, Janosch STASCHEIT and Günther MESCHKE

Institute for Structural Mechanics
Ruhr University Bochum
Universitätstr. 150, Bochum 44801, Germany
e-mail:{abdullah.alsahly; janosch.stascheit; guenther.meschke}@rub.de
webpage: http://www.sd.rub.de

Key words: Adaptive Modeling, Steering algorithm, hybrid re-meshing, mechanized tunneling

Abstract. The simulation of the advancing process for arbitrary alignments during shield tunneling requires a continuous adaption of the finite element mesh in the vicinity of the tunnel face in conjunction with a steering algorithm for the Tunnel Boring Machine (TBM) advance. Moreover, the finite element mesh should match the actual motion path of the shield machine resulting from the FE-analysis in each excavation step. For this purpose, a technique to automatize the process of mesh generation based on hybrid mesh approach is proposed in which a new computational mesh in the vicinity of the tunnel face will be automatically generated within the advancing process. This contribution is concerned with the 3D automatic mesh generation of finite element models for numerical simulations of shield driven tunneling processes.

1 INTRODUCTION

In numerical simulations of shield driven tunneling processes, the realistic modeling of both the excavation process and the advancement of the Tunnel Boring Machine (TBM) is a challenge. For a better understanding of these processes during tunnel construction, the interactions between the shield machine and the surrounding soil need to be investigated, yet this excavation process is difficult to model with existing finite element models. In addition, the simulation of the advancement of the machine as an independent body which interacts with all relevant component of the model, requires a realistic kinematics model of the shield machine which is generally not included in computational models for TBM tunneling [14]. A prototype for a process-oriented three-dimensional finite element model for simulations of shield-driven tunnels in soft, water-saturated soil has been developed and successfully used for systematic numerical studies of interactions in mechanized

tunneling [7], which has been re-formulated and extended to partially saturated soils and more advanced constitutive models for soils in the context of a integrated design support system for mechanized tunneling (see, e.g. [11, 10]). Furthermore, several finite element models have been proposed, addressing the difficulties inherent in the simulation of the excavation process. Many of these models account for excavation by removing finite elements from the excavated volume in front of the machine, and then by applying the nodal forces necessary to preserve equilibrium [1, 3]. A more realistic representation of the excavation process, based on mesh adaptation, using so-called "excavating elements" in front of the machine has been proposed by [8]. In this paper, a steering algorithm and re-meshing strategy are presented in the context of 3D modeling TBM advancement processes. The algorithm serves as an virtual guidance system which automatically determines the exact position and the driving direction of the TBM in three dimensional space. A new approach for hybrid mesh generation is proposed, which adapts the spatial discretization in the vicinity of the tunnel face according to the actual position of the TBM. This hybrid mesh attempts to combine full advantage of the numerical accuracy and practical aspects of structured hexahedra meshes, while the numerical error can be controlled by the chosen density and interpolation order of the unstructured tetrahedral mesh within the excavation region denoted as region of interest.

2 FINITE ELEMENT MODELING OF ADVANCEMENT PROCESSES

2.1 The kinematic model of the shield machine

A close to reality modeling of the advancing process and the interaction between the TBM and the surrounding environment requires a realistic kinematic model of the shield machine [4]. Therefore, a nonlinear kinematic analysis of the shield, based on the action forces imposed on the shield and on the inertial forces due to the shield, is performed. The action forces result from hydraulic jacks pushing against the machine, earth/slurry pressure at the cutting face, friction with surrounding soils, and the fluid flow of processes of the support fluid and grouting mortar, whereas the inertial forces are due to the self weight of the shield and of the equipment. Furthermore, the taper and the thickness of the shield skin are accounted for in the geometrical representation of the TBM (see figure 1), allowing for a realistic distribution of the ground reaction forces in both circumferential and longitudinal directions. A Lagrangian finite element analysis of large deformations that satisfies both the displacements and forces boundary conditions imposed by shield machine operation provides the actual TBM kinematics. Within this approach the shield machine is modeled as a deformable body using a Total Lagrangian Finite Element formulation.

2.2 Steering correction algorithm

The TBM is advanced by hydraulic jacks that are attached to the machine which push against the previously installed lining ring. The pressure exerted by these jacks must
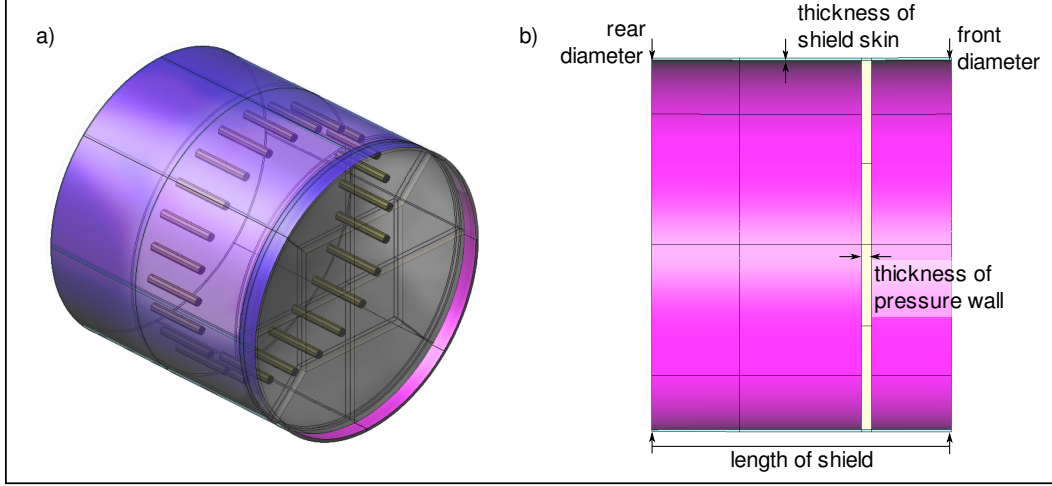
Figure 1: Geometrical model of the shield machine: a) transparent view of shield geometry and hydraulic jacks; b) dimensions of the shield

overcome the resistance generated by the surrounding soil. Moreover, since the machine is heavier at the head, the jack forces are highest in the invert and conversely the lowest in the crown. Driving the shield along curves requires additional steering forces along the sides to ensure the machine follows the intended three dimensional curve. When the steering or the so-called deviation correction is needed, the pressure at individual hydraulic jacks is adjusted to produce deflection torques in the horizontal and vertical direction. In the computational model, the shield machine is pushed forward by extending hydraulic jacks represented by CRISFIELD truss elements. These are connected to both the surface of the lining and the shield. The jack elongations are accomplished by providing initial strains which describe the desired elongation. Respective values for the jack pressures are obtained as a simulation result [13]. In accordance with tunneling practice, a reliable steering algorithm that provides the numerical model with the required information to keep the TBM on the track is developed. This TBM advancement algorithm serves as an artificial guidance system which automatically determines the exact position and the driving direction of the TBM in three dimensional space providing the vertical and horizontal deviation, shield orientation and direct input for the jacking cylinders. For this purpose, the steering algorithm provides a non-uniform jack thrusts, forcing the reference point on the shield to a given point along the alignment. The method for which the machine is advanced is as follows:

1. The geometry of the tunnel path is described by a set of coordinates in three dimensional space.

2. Each jack element J^i is defined by its end points: its connection to the lining $\boldsymbol{p}_L$ and its connection to the shield wall $\boldsymbol{p}_S$ as shown in figure 2.
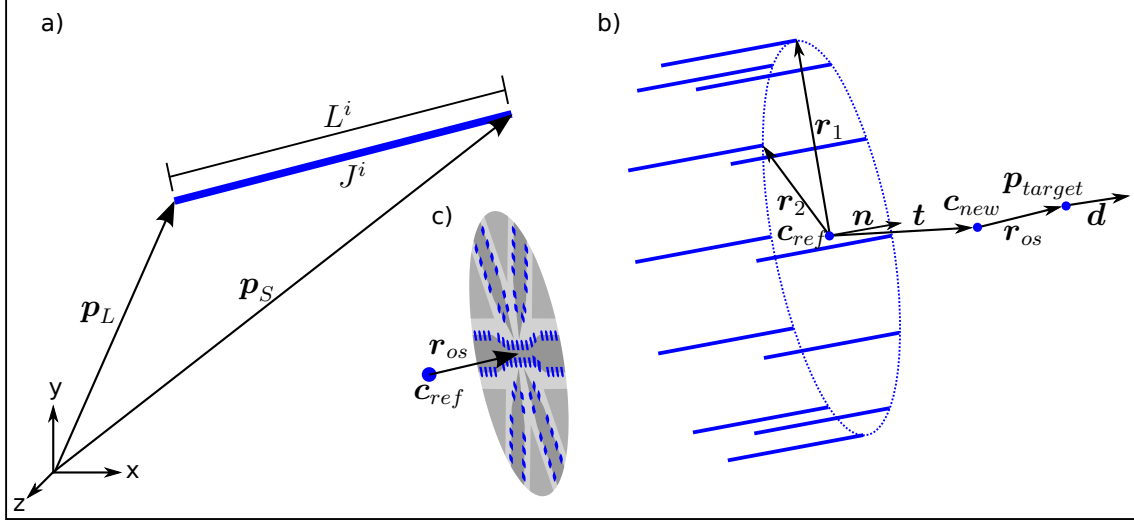
Figure 2: Definitions in the steering algorithm: a) measures related to one jack element J^i; b) measures related to the steering algorithm; c) definition of the offset vector $\boldsymbol{r}_{os}$

3. A reference point $\boldsymbol{c}_{ref}$ on the shield wall is specified to control the deviation of the machine from the intended path after each advancing step. This point is the centre of a circle formed by the tips of all n_J hydraulic jack elements on the shield end:

$$\boldsymbol{c}_{ref} = \tfrac{1}{n_J} \sum_{i}^{n_J} \boldsymbol{p}_S^i \tag{1}$$

Since this reference position differs from the centre of the cutting wheel, an offset vector $\boldsymbol{r}_{os}$ is defined that points from the reference point to the centre of the cutting wheel.

4. The new position $\boldsymbol{p}_{target}$ of the centre of the cutting wheel as well as a director $\boldsymbol{d}$ for the heading of the shield are required for advancing the shield. These are determined from the chainage of the tunnel alignment. In the kth alignment station $\boldsymbol{a}^k$, these quantities are obtained by:

$$\boldsymbol{p}_{target} = \boldsymbol{a}^{k+1} \tag{2}$$

$$\boldsymbol{d} = \frac{\boldsymbol{a}^{k+2} - \boldsymbol{a}^{k+1}}{\|\boldsymbol{a}^{k+2} - \boldsymbol{a}^{k+1}\|} \tag{3}$$

In combination with the offset vector, the new position of the reference point is obtained:

$$\boldsymbol{c}_{new} = \boldsymbol{r}_{target} - \boldsymbol{r}_{os} \tag{4}$$

From these positions, a translation vector $\boldsymbol{t}$, that defines the pure translation of the reference point from its current position to its target position, is calculated:

$$\boldsymbol{t} = \boldsymbol{c}_{new} - \boldsymbol{c}_{ref} \tag{5}$$

Subsequently, the current heading of the shield needs to be determined. For this purpose, two radial vectors $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ are defined, from which the current heading $\boldsymbol{n}$ can be derived by:

$$\boldsymbol{n} = \frac{\boldsymbol{r}_1 \times \boldsymbol{r}_2}{\|\boldsymbol{r}_1 \times \boldsymbol{r}_2\|} \tag{6}$$

5. The new position $\boldsymbol{p}^i_{S,new}$ of each jack element after steering is computed using the a standard cartesian rotation matrix $\boldsymbol{A}_{rot}$ in three dimensional space. The rotation axis is perpendicular to both $\boldsymbol{n}$ and $\boldsymbol{d}$:

$$\boldsymbol{v} = \begin{cases} \boldsymbol{n} \times \boldsymbol{d}, \text{ if } \boldsymbol{n} \neq \boldsymbol{d} \\ \{1, 0, 0\} \quad \text{else.} \end{cases} \tag{7}$$

By this choice of a rotation axis, the rotation angle α and a respective rotation matrix $\boldsymbol{A}_{rot}$ can be computed:

$$\alpha = \arccos(\boldsymbol{n} \cdot \boldsymbol{d})$$

$$\boldsymbol{A}_{rot} = \begin{pmatrix} C + v_1^2 (1 - C) & v_1 v_2 (1 - C) - v_3 S & v_1 v_3 (1 - C) + v_2 S \\ v_2 v_1 (1 - C) + v_3 S & C + v_2^2 (1 - C) & v_2 v_3 (1 - C) - v_1 S \\ v_3 v_1 (1 - C) - v_2 S & v_3 v_2 (1 - C) + v_1 S & C + v_3^2 (1 - C) \end{pmatrix} \tag{8}$$

with $C = \cos \alpha$ and $S = \sin \alpha$

6. The new position of each jack on the shield wall is determined to obtain the required jack elongations:

$$\boldsymbol{p}^i_{S,new} = \boldsymbol{c}_{new} + \boldsymbol{A}_{rot} (\boldsymbol{p}_S - \boldsymbol{c}_{ref}) \tag{9}$$

7. The new reference length and the elongations are computed and applied as an internal Green Lagrange strain for each CRISFIELD truss element:

$$L^i_{new} = \|\boldsymbol{p}_{S,new} - \boldsymbol{p}_L\| \tag{10}$$

$$\boldsymbol{E}_{11} = \frac{L^2_{new} - l^2_{ref}}{2\, l^2_{ref}} \tag{11}$$

3 AUTOMATIC MODELING OF THE EXCAVATION PROCESS

The simulation of the advancing process for arbitrary alignments by means of the proposed steering algorithm requires a continuous adaption of the finite element mesh in the vicinity of the tunnel face. Furthermore, the finite element mesh should match the actual motion path of the shield machine resulting from the FE-analysis in each excavation step. For this purpose, a re-meshing algorithm is developed in order to automate the process of mesh generation in a domain in the vicinity of the tunnel face within the advancing process.

3.1 Tunnel geometry

The two major representation schemes used to describe a solid model are Constructive Solid Geometry (CSG) schemes and Boundary Representations (B-Rep) schemes [6]. In an underground excavation the site geometry is often irregularly shaped, therefore, the B-rep scheme is seen to be the most efficient as it can easily describe the polyhedral surfaces needed to represent the tunnel geometry. A generalized cylinder is a representation of an elongated object that has a main axis (directrix or spine) and a smoothly varying cross section (generatrix) [9, 12]. A directrix and a generatrix can both be open or closed curves. In tunneling simulation , the directrix is a bounded 3D curve representing the tunnel path, and the generatrix is a closed curve representing the tunnel cross section. The 3D tunnel path and the tunnel cross section are approximated using piecewise linear line segments that are optimized to match the circular geometry of the TBM. The surface geometry of a tunnel is created by sweeping the 2D cross section polygon along the piecewise linear tunnel path, as this is a quite common approach in underground engineering structures.

3.2 The meshing algorithm

The main aim of this meshing algorithm is to describe the new geometry by generating a new mesh automatically. By using the so-called piecewise linear system (PLS), not only is the boundary of excavation path described but also the external boundary of the re-meshing domain. Figure 3 illustrates one possible representation of the excavation geometry by means of a piecewise liner system after one advancing step. The internal facet of the PLS should be automatically updated to describe the new internal boundary, where the external facets are still fixed.
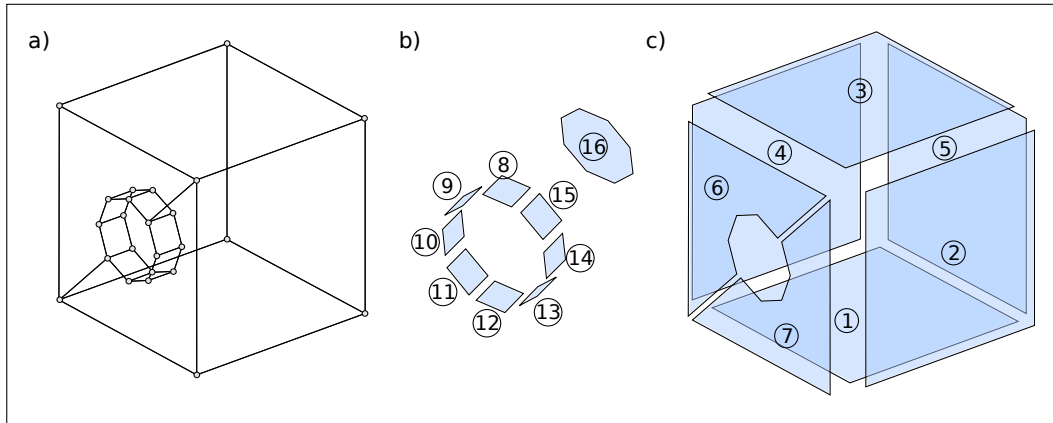
Figure 3: Representation of the excavation geometry by means of PLS: (a) Region of interest represented as PLS; (b) Internal boundary; (c) External boundary

This can be realized by sweeping the new facets representing the new excavated part and adding it to the PLS after each advancing step. The kinematic analysis of the shield

within the steering procedure provides the exact position and geometry of the new facets as well as the center of the cutting wheel and some reference points. To mesh the domain, a 3D Delaunay meshing algorithm TetGen [5] is used to generate an unstructured mesh consisting of tetrahedral elements. Delaunay-based algorithms are capable of producing quality meshes and provide control over mesh sizing throughout the domain. The major components of information required by the meshing algorithm are as follows:

- the input domain, which is a polyhedron, that defines the geometry of the problem,

- the position of the cutting head after each advancing step,

- a set of optional optimization criteria in order to control the quality of the mesh and the compatibility with neighboring domains.

3.3   Region of interest

Let V be a homogeneous and isotropic deformable body which occupies a domain $\mathbf{\Omega} \subset \mathbf{R^d}(d = 2 \, or \, 3)$ with a boundary $\mathbf{\Gamma} \subset \mathbf{R^{d-1}}$. In this proposed re-meshing technique, the domain $\mathbf{\Omega}$ of the complete simulation model is divided into 2 non-overlapping subdomains $\mathbf{\Omega_1}$ and $\mathbf{\Omega_2}$ with $\mathbf{\Omega = \Omega_1 \cup \Omega_2}$ and $\mathbf{\Omega_1 \cap \Omega_2 = 0}$. Each sub-domain $\mathbf{\Omega_i}$ might represent either an excavation domain (named as region of interest) or the surrounding soil around this region with different layers of the geological formation. The boundary $\mathbf{\Gamma_i}$ of a sub-domain $\mathbf{\Omega_i}$ can represent the interface between neighboring sub-domains or the internal boundary of the excavation domain. Here, the region of interest represented by $\mathbf{\Omega_1}$ is the region of the mesh that is continuously generated during TBM advance in the vicinity of the tunnel face. It must include at least the excavation geometry. It can be as large as the whole simulation domain or be limited to a small region around the heading face. In any case, the engineer will make the decision based on the available data and his engineering experience. In Figure 3 the boundary $\mathbf{\Gamma_1}$ of the region of interest is represented by a closed polyhedron. To enable a dynamic and efficient simulation of arbitrary TBM advancement paths, information of the target excavation path is incorporated in the definition of the region of interest.

3.4   Hybrid mesh

The hybrid mesh approach attempts to combine the advantages of both structured and unstructured mesh layouts. The hexahedral elements are used in regions where no complex geometries exist, no re-meshing is required, and the numerical accuracy and practical aspects of structured hexahedra meshes are preserved. Whereas the tetrahedral elements are generated automatically describes the complex geometry (i.e. excavation boundary), while deficiency inherent to unstructured tetrahedral meshes is introduced only in the excavation region named region of interest.

To obtain a compatible hybrid mesh we should ensure a proper connectivity such that the internal tetrahedral elements (within the region of interest) are connected properly to
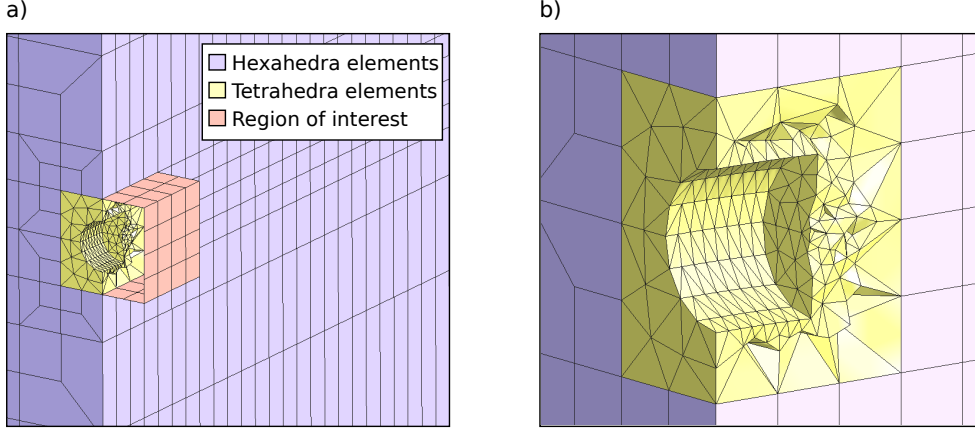
Figure 4: Hybrid mesh representation of excavation geometry; (a) Mesh conmponents (b) Compatible mesh using hexahedral and tetrahedral elements.

the external hexahedral elements (at the boundary of the region of interest). Here, 27-noded quadratic hexahedral elements and 10-noded quadratic tetrahedral elements are employed. In other words, the resultant triangular surface mesh of the tetrahedral elements match exactly the quadrilateral surface mesh of the hexahedral elements surrounding the region of interest. The resulting mesh of this hybrid approach will automatically match both the external boundary in terms of connectivity to an existing boundary mesh and, internally, the projected motion path of the shield and the heading face as shown in Figure 4.

3.5   Modeling of excavation

The re-meshing algorithm works in conjunction with the steering algorithm. The steering algorithm simulates the advancing process in a step-by-step procedure. After each advancing step the re-meshing algorithm is invoked and generates a new computational mesh describing the new excavation geometry. The re-meshing algorithm uses the results from the steering algorithm as input for generating the new mesh. The exact geometry and position of the TBM after each advancing step will be extracted and used to generate the new mesh preserving the deformed configuration of the previous excavated geometry. By doing so, the excavation and the advancement of the shield machine are numerically simulated in a continuous manner. After obtaining the new mesh, several mesh operations and optimization techniques are required as follows:

- Optimization algorithm to project all central nodes of the higher order tetrahedral elements to their correct position in order to represent the exact curve and the circular shape of the shield.

- Generation of a new surface mesh to represent the excavation boundary, tunnel face

and contact surfaces.

- Variable transfer algorithm: after the re-meshing, the values of the stresses and the internal variables on the new mesh need to be calculated from those obtained in the original deformed mesh. This is necessary because the equilibrium condition needs to be satisfied before conducting the next advancing step. An appropriate algorithm based on Superconvergent Patch Recovery (SPR) [2] for the transfer of these internal variables is adopted.

4   APPLICATION EXAMPLES

In this section, two examples are presented to demonstrate the flexibility of the re-meshing algorithm and the applicability for driving the machine along curved alignment. In the first example, the re-meshing algorithm is invoked to generate a proper mesh describes the geometry of the of the excavated path. Within this example the mesh will be regenerated automatically after each advancing step. Furthermore, the machine is to be advanced in the soil following an arbitrary path. The geometry of the excavation path is defined by means of the following parameters:

- The radius of the excavation (radius of the machine) $= 5.25m$.

- The number of the segments on the circular geometry of the machine $= 20$.

- The size of the advancing step is $= 0.5m$ in each step.

- The tunnelling path is assumed as a quadratic path described by means of a sequence of points.

After each advancing step the original mesh is replaced with a new mesh to represent the complete excavation path. The goal of this example is only to show that the new generated mesh conforms to the excavation geometry at the end of each advancing step. The generated meshes after different advancing steps are shown in 5 . The advancing is performed in x-direction whereas the steering is performed in each advancing step in the two other directions y and z simultaneously. Cross sections through the excavation path are visualized where the x-y plane and y-z plane are considered.

In this second example, the advancing process and the application of the new excavation technique are demonstrated by considering the southwestern part of the L9 tunnel "Mas Blau". It consists of a single twin-track tunnel that runs along a strongly curved path. The tunnel is characterized by a diameter (D) of 9.4 m and a cover depth of 14.0 m. The tunnel alignment and the simulation domain are shown in Figure 6. In this study, the elastic model for the lining and the shield machine with basic material properties summarized in Table 1 is adopted. For the modeling of the soil, a DRUCKER-PRAGER plasticity model is used.

Advancing step 4   Advancing step 10   Advancing step 16

Cross section through X-Z plane

Advancing step 4   Advancing step 10   Advancing step 16

Cross section through X-Y plane
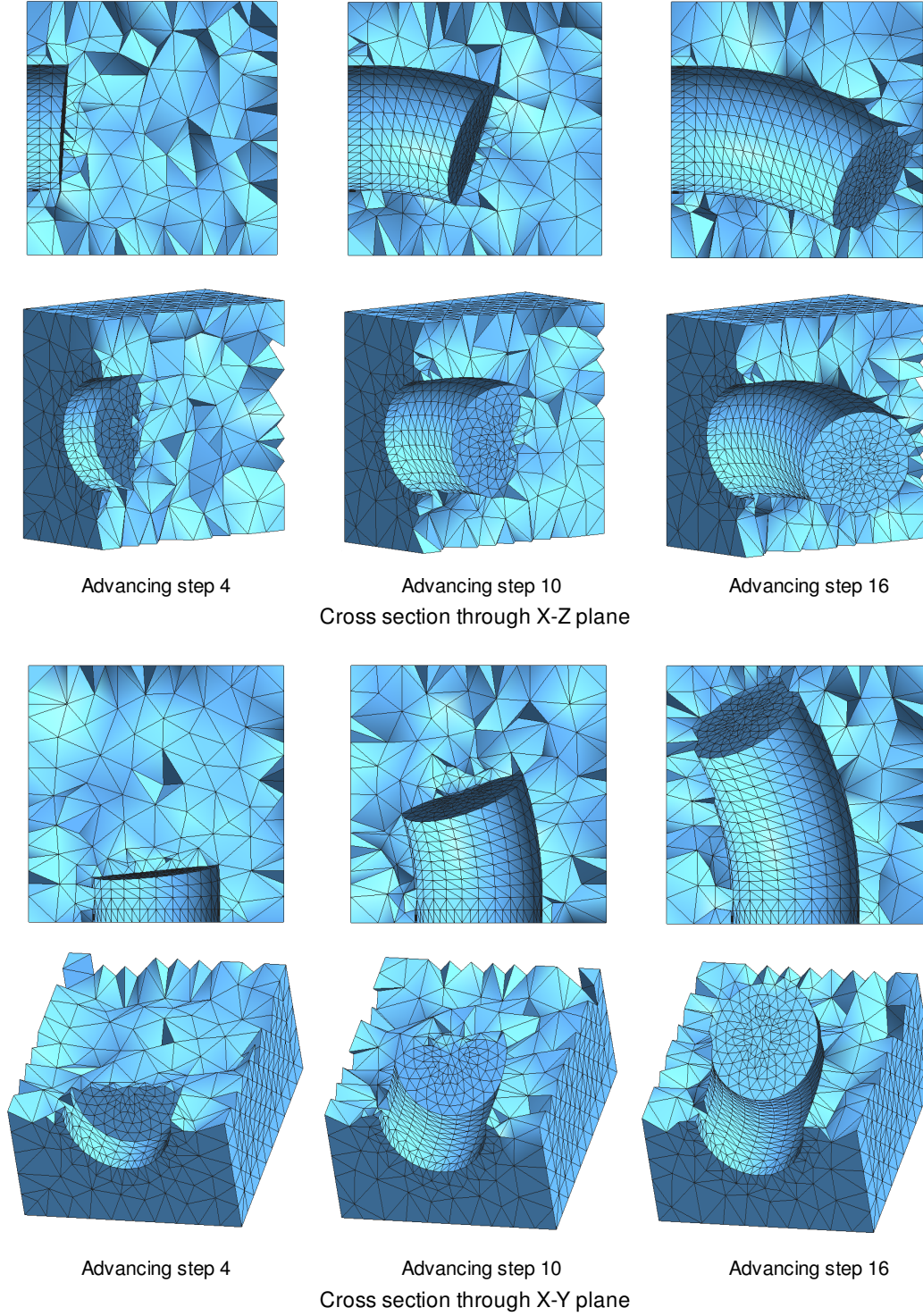
Figure 5: Conformal tetrahedral mesh for the steering of the TBM, different cross-sections through the excavation path in x-y plane and y-z plane.
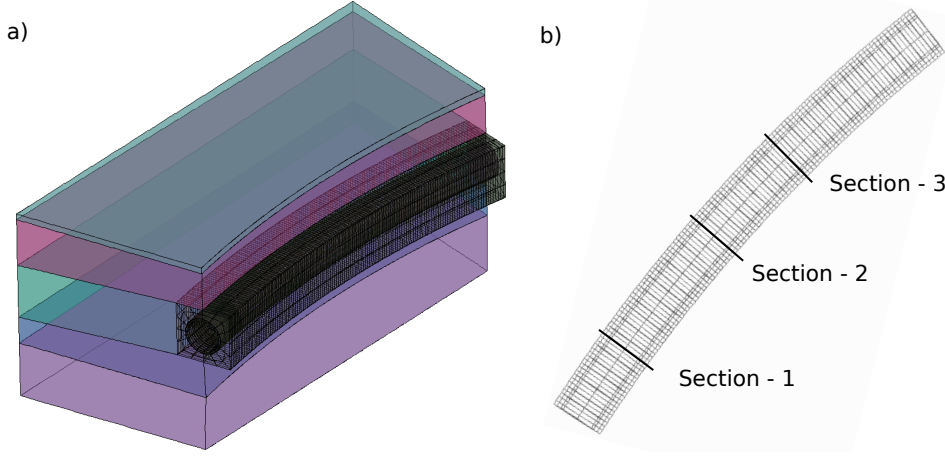
Figure 6: L9 Tunnel "Mas Blau" ; (a) representation of the simulation domain (b) tunnel alignment and the position of three cross section

Table 1: Material parameters used in the finite element model

model part	γ [kN/m^3]	φ [°]	c [MPa]	E [MPa]	ν [-]
Lining	25.0	-	-	30000	0.2
Shield	76.2	-	-	210000	0.27
Soil	28.5	29	0.11	2100	0.28

Figure 7 shows the obtained jack forces in three different positions during the simulation of the advancing process. The nonuniform jack thrust distributions illustrate the power of the steering algorithm in conjunction with the developed re-meshing algorithm and the efficiency of the algorithm that controls the TBM position and keeps it on the intended target alignment. The diagram in Figure 7 shows the change of the steering forces during the steering process in the horizontal and vertical directions. Thus, the finite element results are consistent with the actual shield advancement procedure and guidance system.

The results of the re-meshing algorithm are illustrated in Figure 8 for different advancing steps. The results obtained from this simulation demonstrate the efficiency and the high applicability of the re-meshing algorithm to capture the exact excavation path. In addition, it is possible to advance the TBM at different speed and the meshing algorithm will account for the size of the mesh automatically. In standard numerical analyses of TBM advance, where the mesh is generated a priori, this would not be possible.
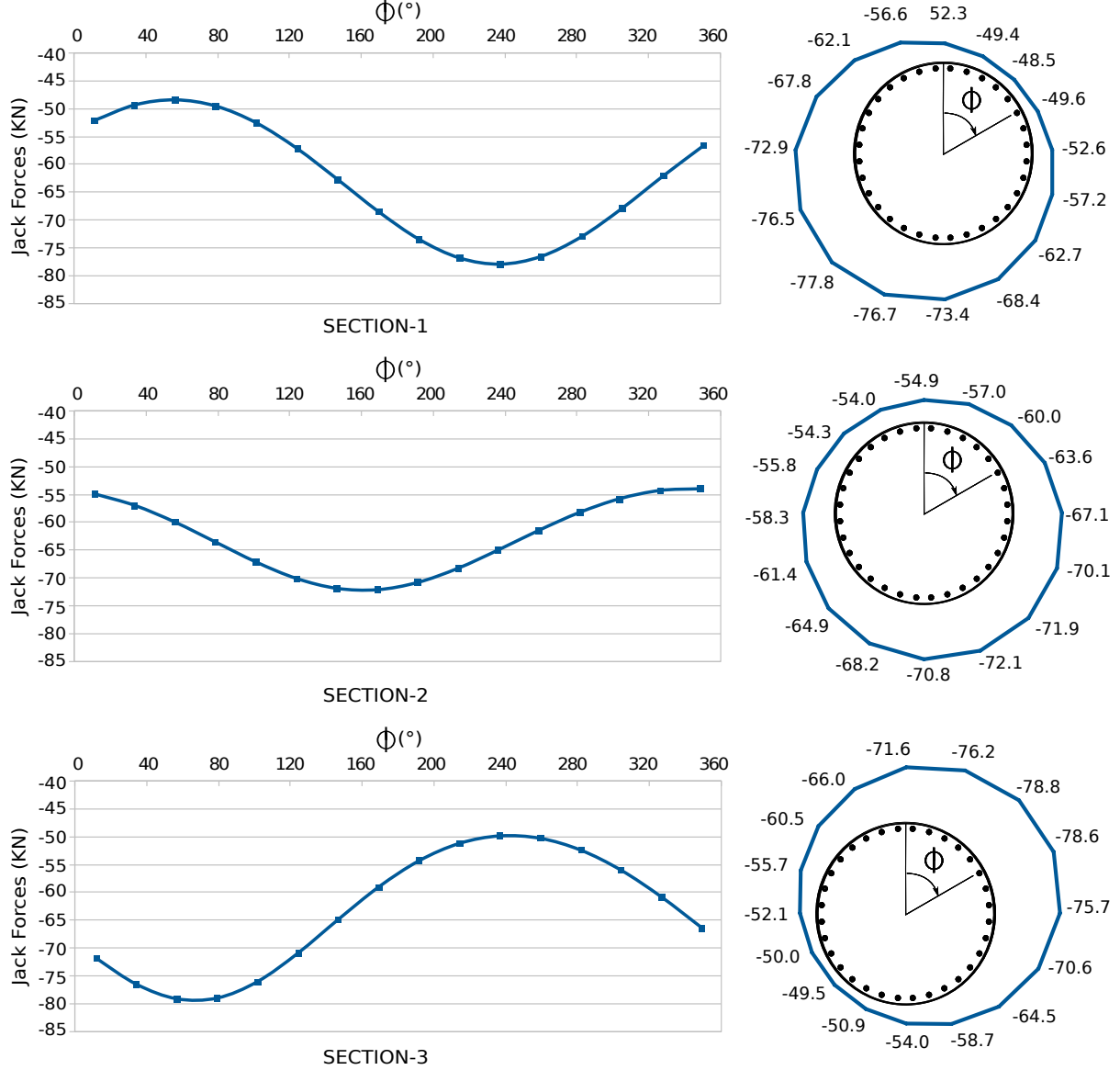
Figure 7: Distribution of the Hydraulic jack forces along the shield wall in different advancing steps (Section-1, Section-2, Section-3 in Figure 6)

5 CONCLUSIONS

A fully automatic re-meshing technique for the modeling of the advancement and the excavation process of shield tunneling has been proposed, with an efficient and automatic algorithm to simulate the advancing process along arbitrary tunnel alignments. The kinematic shield model was used to simulate the advancing process along curved tunnel alignment.The advancement process of the TBM and the soil excavation was simulated step wise using a new steering and a problem specific re-meshing algorithm. The steering

Section - 1   Section - 2   Section - 3
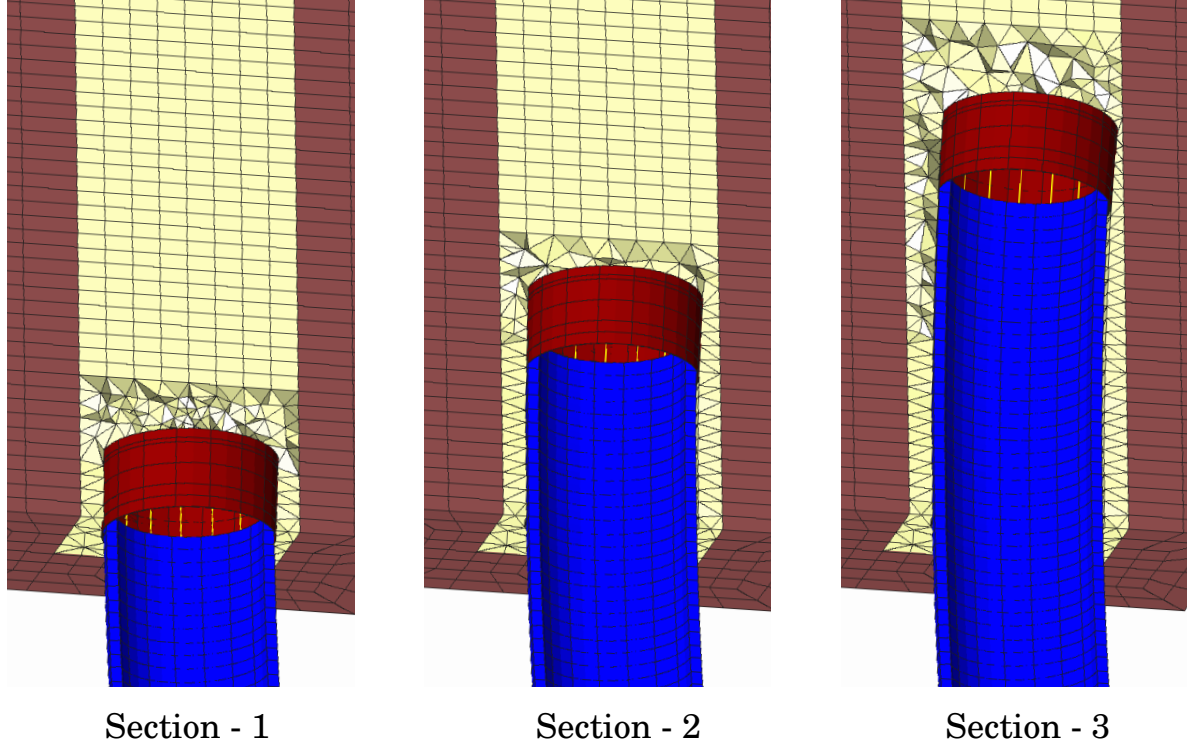
Figure 8: FE mesh representing the excavation geometry for different advancing steps (Section-1, Section-2, Section-3 in Figure 6)

algorithm in conjunction with the re-meshing algorithm are independent from any a priori generated discretization and consider the ground movement around the shield. It was shown, that the proposed computational model is able to simulate the actual motion of the TBM along arbitrary paths with minimum effort required in the preprocessing stage. Further developments of the model will focus on the influence of the discretization error during the excavation process.

6   ACKNOWLEDGEMENTS

REFERENCES

[1] S. Bernat and B. Cambou. Soil - structure interaction in shield tunnelling in soft soil. *Computers and Geotechnics*, 22(3/4):221–242, 1998.

[2] B. Boroomand and O. Zienkiewicz. Recovery by equilibrium in patches (rep). *International journal for numerical methods in engineering*, 40(1):137–164, 1997.

[3] G. Clough, B. Sweeney, and R. Finno. Measured soil response to epb shield tunneling. *Journal of Geotechnical Engineering*, 109(2):131–149, 1983.

[4] R. Finno and G. Clough. Evaluation of soil response to EPB shield tunneling. *Journal of Geotechnical Engineering*, 111(2):155–173, 1985.

[5] S. Hang. Tetgen. *A quality tetrahedral mesh generator and three-dimensional Delaunay triangulator. Version*, 1, 2007.

[6] C. M. Hoffmann. *Geometric and solid modeling: an introduction*. Morgan Kaufmann Publishers Inc., 1989.

[7] T. Kasper and G. Meschke. A 3D finite element model for TBM tunneling in soft ground. *International Journal for Numerical and Analytical Methods in Geomechanics*, 28:1441–1460, 2004.

[8] K. Komiya, K. Soga, H. Akagi, T. Hagiwara, and M. Bolton. Finite element modelling of excavation and advancement processes of a shield tunnelling machine. *Soils and Foundations*, 39(3):37–52, 1999.

[9] T. Maekawa, N. M. Patrikalakis, T. Sakkalis, and G. Yu. Analysis and applications of pipe surfaces. *Computer Aided Geometric Design*, 15(5):437–458, 1998.

[10] G. Meschke, F. Nagel, and J. Stascheit. Computational simulation of mechanized tunneling as part of an integrated decision support platform. *Journal of Geomechanics (ASCE)*, 11(6):519–528, 2011. Special Issue: Material and Computer Modeling.

[11] F. Nagel, J. Stascheit, and G. Meschke. Numerical simulation of interactions between the shield supported tunnel construction process and the response of soft, water saturated soils. *International Journal of Geomechanics (ASCE)*, 12(6):689–696, 2011.

[12] J. Pegna. *Variable sweep geometric modeling*. PhD thesis, Stanford University, 1987.

[13] J. Stascheit. *Parallelisation and model generation methods for large-scale simulations of shield tunnelling processes*. PhD thesis, Ruhr-Universität Bochum, 2010.

[14] M. Sugimoto and A. Sramoon. Theoretical model of shield behaviour during excavation. I: Theory. *Journal of Geotechnical and Geoenvironmental Engineering*, 128(2):138–155, 2002.

GLUE-CONCRETE INTERFACE OF BONDED ANCHOR – EXPERIMENT AND MODEL

JAN BARNAT[*], MIROSLAV BAJER[†]

[*] Faculty of Civil Engineering, Brno University of Technology, Institute of Metal and Timber Structures, Veveří 95, 602 00 Brno, Czech Republic, barnat.j@fce.vutbr.cz, http://www.fce.vutbr.cz

[†] Faculty of Civil Engineering, Brno University of Technology, Institute of Metal and Timber Structures, Veveří 95, 602 00 Brno, Czech Republic, bajer.m@fce.vutbr.cz, http://www.fce.vutbr.cz

Key words: Bonded anchor, Tensile load, Bond stress, Experiments, Numerical Modeling, Contact,

Abstract. This paper considers the practical application of nonlinear models in the analysis of an anchor bolt additionally placed in a concrete specimen. The model also considers epoxy surface. The results of analyses performed using the concrete model of specialized Atena 3D finite element code are presented and discussed. The mesh density and convergence stability are compared in single anchor case study. All model results are compared with experimentally obtained data. There are also experiments focused on bond stress presented which are very important for verification of model assessment. The article is focused on problems of post-installed steel bonded anchors real behaviour. Experiments and numerical models described in this paper are focused on problems of bond stress quality, on anchor joint loaded by tension force.

1 INTRODUCTION

The development of a model for the behaviour of concrete is a challenging task. Concrete is a quasi-brittle material and has different behaviour in compression and tension. The tensile strength of concrete is typically 8-15% of the compressive strength.

In compression, the stress-strain curve for concrete is linearly elastic up to about 30 percent of its maximum compressive strength. Above this point, the stress increases gradually up to its maximum compressive strength. After it reaches its maximum compressive strength σ_{cu}, the curve descends into a softening area, and eventually, crushing failure occurs at an ultimate strain ε_{cu}. In tension, the stress-strain curve for concrete is approximately linearly elastic up to its maximum tensile strength. Beyond this point, the concrete cracks and the strength decreases gradually to zero. Modelling anchor behaviour we need to take into account also a high stress gradient near anchor rod, epoxy material behaviour, possibility of failure on (concrete-epoxy, epoxy-steel) contact surfaces.

In Atena system, the constitutive model of concrete includes 20 material parameters. These parameters are specified for the problem under consideration by the user. If the parameters are not known, automatic generation can be done using the default formulas given [1]. In this

case, only the cube strength of concrete f_{cu} (nominal strength) is specified and the remaining parameters are calculated as functions of the cube strength. The formulas for these functions are taken from the CEB-FIP Model Code 90 and other research sources. Also the transfer coefficient, βt, can be calculated.

2. CONTACT - ATENA

There are two ways for failure criteria for concrete in Atena. In the rotated crack model in Atena system the direction of the principal stress coincides with the direction of the principal strain. Thus, no shear strain occurs on the crack plane and only two normal stress components must be defined. In the fixed crack model also included in Atena system the crack direction is given by the principal stress direction at the moment of the crack initiation. During further loading, this direction is fixed and represents the material axis of the orthotropy.

The problem remains in definition of glue working diagram and also the contact elements behaviour.

For contact behaviour description there are two set of stiffness in each direction defined. (Normal and tangent stiffness). The first is stiffness valid before reaching the ultimate stress value on contact and the second valid after exceeding this boundary value. For normal stiffness the boundary value of stress is defined by the value of contact tensile strength (in this case it is the mean value of concrete tensile strength). For tangential stiffness the boundary value is defined by the ultimate value of bond stress and by friction coefficient. For our model we used the mean value of ultimate bond stress from experiments and friction coefficient 0.3. For secure the solution stability the stiffness after the contact failure should not be zero. Contact model is in detail described in [2]. For concrete modelling the 3D nonlinear cementitious model was used. This concrete model is able to represent the non-linear behaviour of concrete inclusive tensile and compressive softening.

The setting of these two set of stiffness values is essential for optimisation of the model. The whole behaviour of anchor is depending on anchor bolt deflection. As it is shown further in model samples, there is a great dependence of results on mesh size. This is mainly caused by GAP elements behaviour which is defined by initial stiffness. Initial value of stiffness should be adjusted in dependence on contact elements size, as it is indicated in [3] as (1).

$$K\ [MN/m] = E,concrete\ [N/mm] / elem,size\ [m] * 10 \qquad (1)$$

The relation (1) is defined and useful for standard Atena concrete models of structures or it parts whit element sizes in range of 0,1m. Relation (1) cannot be used In case of modelling of small detail such as bonded anchor where is need to use smaller elements in range of 0,01mm. Modell with such contact model is usually unstable and mismatch the results of experiments.

2 BONDED STRESS EXPERIMENTS

Tension resistance of bonded anchor can be increased together with larger anchoring length. However due to geometric conditions this solution in not always possible. This can be problem of modern structures using high performance materials. The experiments described

in this paper are focused on verification of bonded anchor tension resistance limits determined by bond strength provided by the glue. Bond strength is an overall parameter used for description of connection quality between the steel anchor and concrete. The main principle of the bond quality test used (as depicted in fig.1) is to restrict bond failure to the anchor only. Load is applied to the anchor bolt by the loading mechanism which is itself supported by the concrete in the immediate vicinity of the installed anchor. Experiments are closely described in [2]
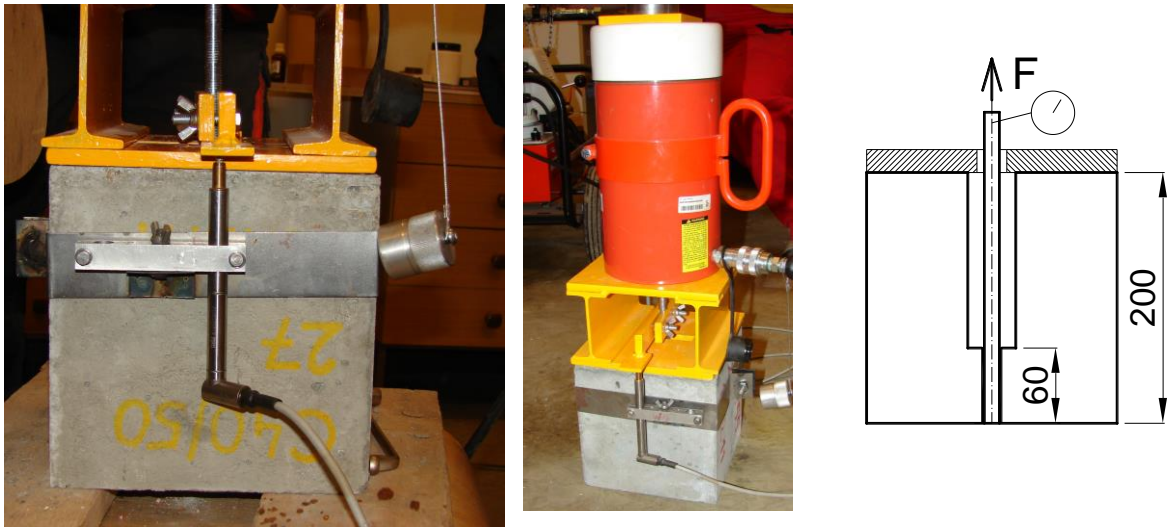
Figure 1: Bond strength experiment

Bond strength is defined as shear stresses on one of contact interfaces. It can be evaluated on interface between steel and glue or on the interface between glue and concrete, which is more suitable for description of combined concrete-bond failure mode. The experiment sample after failure is shown in fig. 2a). Fig. 2b) shows similar configuration of experiment with use of steel specimen instead of concrete. This configuration was used for determination of shear strength of hardened glue not influenced by concrete strength. Steel specimens were manufactured with internal female thread to ensure fine mechanical connection between steel and glue. The thickness of glue layer was set to 1 mm as it is usual in common bonded anchors systems.

Figure 2: a) M12 specimen after failure (epoxy resin). b) steel specimen

3 COMPARISON OF EXPERIMENTAL AND NUMERICAL RESULSTS

Folowing two graphs (fig. 3 and fig. 4) are presenting comparison of results between Atena 3d model and experiments. All data are for anchor size M12 with the length of contact 60 mm. Anchors were embedded in concrete with compressive strength 80 MPa by experimental epoxy resin based glue with shear strength 30 MPa.
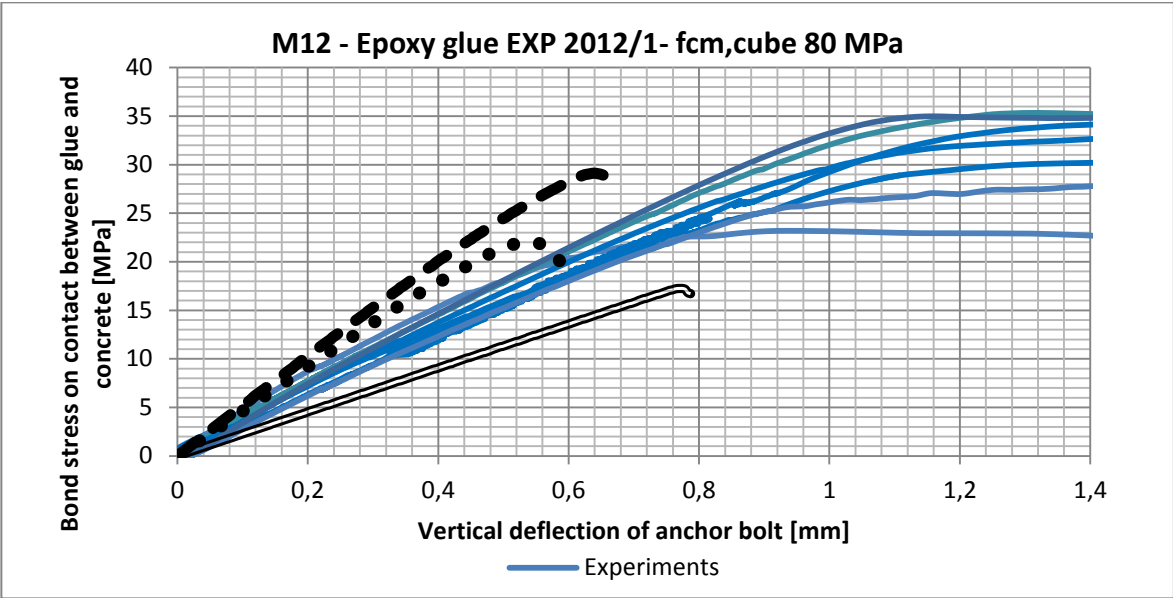
Figure 3: Model and experiment comparison (Bond strength) in dependence on initial stiffness [MN/m] for contact element size 5 mm
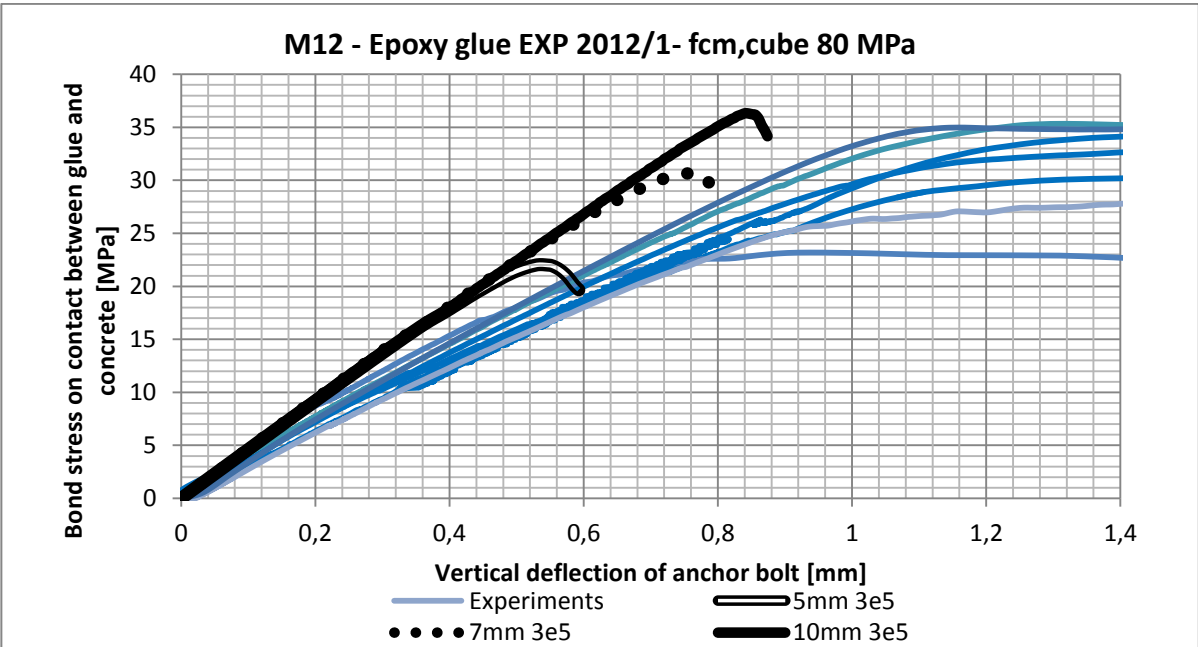
Figure 4: Model and experiment comparison (Bond strength) in dependence on contact element size and initial stiffness 3E5 [MN/m]

The contact length is 60 mm, therefore the element size in range 5-7 mm is appropriate range (in the view of cpu time and also model accuracy and stability) [5]. Fig. 3 and 4 show the dependence of model behaviour on initial contact stiffness and also element size. The stiffness of anchor in linear part of LD diagram is slightly influenced by the stiffness of whole testing system therefore smaller deformation of anchor in model cannot be regarded as inaccuracy.

Figure 5 shows results of numerical study from nine different setting of model. Experimental data in fig. 3 and 4 gives the mean value of bond strength 30 MPa. This value is also the input of cohesion parameter in contact elements. Therefore the results of model should give also value close to 30 MPa. The size of elements around the contact was set to 5, 7 and 10 mm and for all these three element sizes, the initial value of both stiffness was set to 3E4, 3E5 and 3E6 MN/m.

The black strip in fig 5a) represents the appropriate setting of initial stiffness and element size to get the result close to experiment. Sample view on model mesh with cracks propagation is shown in fig. 5b)
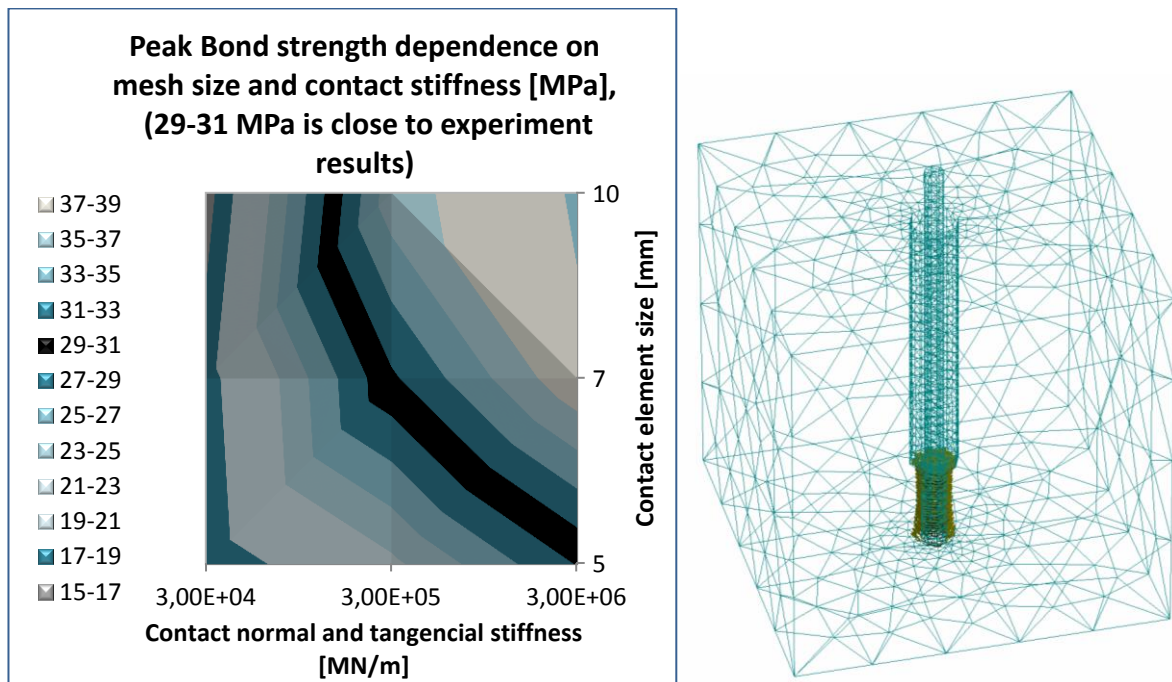
Figure 5: a) Combined dependence of results on initial stiffness and element size; b) model mesh and cracks

4   CONCLUSIONS

- Paper shows problems of modelling a small detail of composite structure, such as bonded anchor placed in high performance concrete, using Atena software for nonlinear modelling of concrete structures.
- The main problem is the mesh size dependence of model specially using GAP contact elements for simulation of glue-concrete behaviour.
- The size of elements in range of several millimetres leads together with use of typical settings of GAP elements sizes for modelling of bigger structure parts leads to model

instability and inaccuracy of results.

- Numerical study shown in fig. 5a has shown that the value initial tangent and normal stiffness of GAP elements should be optimised in dependence on element size and concrete characteristic according the approximation (2).

$$K \ [MN/m] = 500*E,concrete \ [N/mm] \ /((elem,size \ [mm])^{(-5.3)}) \qquad (2)$$
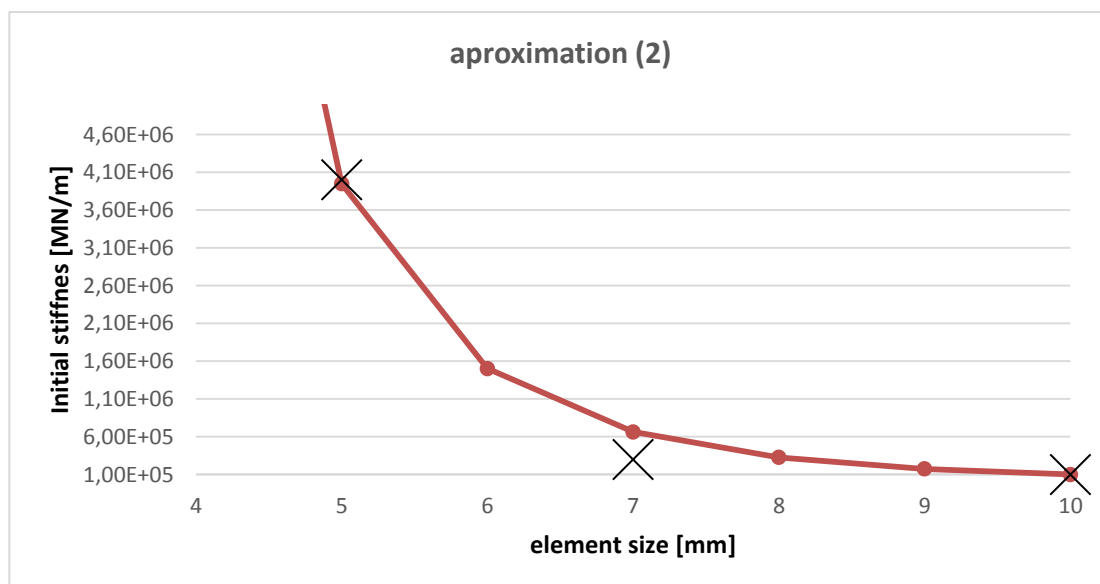
Figure 6: approximation of correlation between element size and initial stiffness

ACKNOWLEDGEMENTS

REFERENCES

[1] Červenka,V. *ATENA software documentation, part 2*, User's manual for ATENA 2D software, Červenka Consulting, Prague, Czech Republic, July 2002,

[2] Bajer, M.; Barnat, J. *The Glue-Concrete Interface of Bonded Anchors*. Construction and building materials. 2012. 34(9). p. 267 - 274. ISSN 0950-0618.

[3] Kala, J., Salajka, V., Hradil, P. Calculation of timber outlook tower with influence of behavior of "steel-timber" connection, Advanced Materials Research Vol. 428 (2012) pp 165-168, Trans Tech Publications, Switzerland, doi:10.4028/www.scientific.net/AMR.428.165, ISSN 1022-6680.

[4] Holomek, J; Bajer, M. *Experimental and Numerical Investigation of Composite Action of Steel Concrete Slab*, Procedia Engineering, Volume 40, 2012, p. 143–147, Elsevier, ISSN: 1877-7058 (1989), Vol. II, (1991).

ROBUST ERROR CONTROL FOR PHASE FIELD MODELS PAST TOPOLOGICAL CHANGES

SÖREN BARTELS

Department of Applied Mathematics,
University of Freiburg, Germany

Abstract. Phase field models are often used to describe the evolution of submanifolds, e.g., the Allen-Cahn equation approximates motion by mean curvature and more sophisticated phase field models provide regularizations of the Willmore flow and other geometric evolution problems. The models involve small regularization parameters and we discuss the dependence of a priori and a posteriori error estimates for the numerical solution of the regularized problems on this parameter. In particular, we address the question whether robust error estimation is possible past topological changes. We provide an affirmative answer for a priori error estimates assuming a logarithmic scaling law of the time averaged principal eigenvalue of the linearized Allen-Cahn or Ginzburg-Landau operator about the exact solution. This scaling law is confirmed by numerical experiments for generic topological changes. The averaged eigenvalue about the approximate solution enters a posteriori error estimates exponentially and therefore, critical scenarios are detected automatically by related adaptive finite element methods. The devised scheme extracts information about the stability of the evolution from the approximate solution and thereby allows for a rigorous a posteriori error analysis.

GOAL-ORIENTED ERROR ESTIMATION AND ADAPTIVITY FOR THE TIME-DEPENDENT LOW-MACH NAVIER-STOKES

PAUL BAUMAN, VARIS CAREY

Institute for Computational Engineering and Science
University of Texas at Austin,
e-mail: pbauman@ices.utexas.edu , varis@ices.utexas.edu

Abstract. We present a goal-oriented algorithm for error control and adaptivity, targeting the low-mach compressible Navier-Stokes equations. The algorithm, using the GRINS computational framework, is illustrated first for stationary problems and then for time-dependent problems. Issues related to stabilization and linearization are highlighted in the former case, while the interplay between storage, efficiency, and numerical accuracy of the forward and adjoint solutions is examined in the latter case.

A PRIORI BASED MESH ADAPTATION FOR VISCOUS FLOW

Anca BELME[*]**, Adrien LOSEILLE**[†]**, Hubert ALCIN**[‡]**, Frédéric ALAUZET**[†]**, Stephen WORNOM**[◇] **and Alain DERVIEUX**[‡]

[*] Institut d'Alembert, UMPC, 4 place Jussieu, 75252 Paris Cedex 05, France,
e-mail: belme@dalembert.upmc.fr

[†] INRIA, BP 105, Domaine de Voluceau, BP 105, 78153 Le Chesnay Cedex, France
e-mail: Adrien.Loseille, Frederic.Alauzet@inria.fr

[‡] INRIA, BP 93, 2004 route des Lucioles, 06902 Sophia–Antipolis Cedex, France
e-mail: Hubert.Alcin@inria.fr, Alain.Dervieux@inria.fr

[◇] LEMMA, 2000 route des Lucioles, Sophia–Antipolis, France
e-mail: Stephen.Wornom@inria.fr

Key words: CFD, viscous flow, Large Eddy Simulation, mesh adaptation

Abstract. A priori estimates are applied to the anisotropic mesh adaptation for 2D viscous flows and 3D ones including Large Eddy Simulation.

1 METHODS

Two novelties were presented [3] in the recent Eccomas conference at Vienna. First, an a priori analysis for diffusive flows allowing, through a goal-oriented criterion, the direct specification of mesh metric, *i.e.* mesh stretching and density, [6]. It is an extension of the mesh adaptation technique refered as the global unsteady fixed point algorithm. In [5] this algorithm was applied to Euler flows. This algorithm involves the following ingredients: (1) an *a priori* goal oriented error estimate based on an adjoint allowing to define an optimal metric at each time level, (2) a fixed point encapsulating a time advancing loop, a backward loop for adjoint, and the generation of a sequence of adapted meshes. Second, this method is extended to the reduction of approximation error in LES formulations [4]. We define the convergence of the LES discrete model to a continuous filtered PDE with prescribed turbulent viscous term and show that an optimal mesh can be defined according to the goal-oriented optimal metric theory. With the help of the above fixed-point algorithm, this gives an optimal mesh for a prescribed filter. This process is then equipped of an external loop for computing the filter as an LES one. These methods were

in [3] benchmarked with the 3D turbulent flow around a cylinder (Reynolds number= 3900). In complement to this very preliminary numerical experiment, we discuss here an industrial test case related to offshore platforms.

2 Continuous mesh theory

2.1 Mesh parametrization

We propose to work in the continuous mesh framework, defined in [1, 2]. The main idea of this framework is to model discrete meshes by Riemannian metric fields. It allows us to define proper differentiable optimization *i.e.,* to use a calculus of variations on continuous metrics which cannot apply on the class of discrete meshes. This framework lies in the class of metric-based methods. A continuous mesh $\mathcal{M}$ of the computational domain Ω is identified to a Riemannian metric field $\mathcal{M} = (\mathcal{M}(\mathbf{x}))_{\mathbf{x} \in \Omega}$. For all $\mathbf{x}$ of Ω, $\mathcal{M}(\mathbf{x})$ is a symmetric 3×3 matrix having $(\lambda_i(\mathbf{x}))_{i=1,3}$ as eigenvalues along the principal directions $\mathcal{R}(\mathbf{x}) = (\mathbf{v}_i(\mathbf{x}))_{i=1,3}$. Sizes along these directions are denoted $(h_i(\mathbf{x}))_{i=1,3} = (\lambda_i^{-\frac{1}{2}}(\mathbf{x}))_{i=1,3}$ and the three *anisotropy quotients* r_i are defined by: $r_i = h_i^3 (h_1 h_2 h_3)^{-1}$. The diagonalisation of $\mathcal{M}(\mathbf{x})$ writes:

$$\mathcal{M}(\mathbf{x}) = d^{\frac{2}{3}}(\mathbf{x}) \, \mathcal{R}(\mathbf{x}) \begin{pmatrix} r_1^{-\frac{2}{3}}(\mathbf{x}) & & \\ & r_2^{-\frac{2}{3}}(\mathbf{x}) & \\ & & r_3^{-\frac{2}{3}}(\mathbf{x}) \end{pmatrix} {}^t\mathcal{R}(\mathbf{x}), \tag{1}$$

The *vertex density d* is equal to: $d = (h_1 h_2 h_3)^{-1} = (\lambda_1 \lambda_2 \lambda_3)^{\frac{1}{2}} = \sqrt{\det(\mathcal{M})}$. By integrating it, we define the *total number of vertices $\mathcal{C}$*:

$$\mathcal{C}(\mathcal{M}) = \int_\Omega d(\mathbf{x}) \, \mathrm{d}\mathbf{x} = \int_\Omega \sqrt{\det(\mathcal{M}(\mathbf{x}))} \, \mathrm{d}\mathbf{x}.$$

Given a continuous mesh $\mathcal{M}$, we shall say, following [1, 2], that a discrete mesh $\mathcal{H}$ of the same domain Ω is a **unit mesh with respect to** $\mathcal{M}$, if each tetrahedron $K \in \mathcal{H}$, defined by its list of edges $(\mathbf{e}_i)_{i=1\ldots6}$, verifies:

$$\forall i \in [1,6], \quad \ell_\mathcal{M}(\mathbf{e}_i) \in \left[\frac{1}{\sqrt{2}}, \sqrt{2} \right] \quad \text{and} \quad Q_\mathcal{M}(K) \in [\alpha, 1] \quad \text{with} \quad \alpha > 0,$$

in which the length of an edge $\ell_\mathcal{M}(\mathbf{e}_i)$ and the quality of an element $Q_\mathcal{M}(K)$ are defined as follows:

$$\ell_\mathcal{M}(\mathbf{e}_i) = \int_0^1 \sqrt{{}^t\mathbf{ab} \, \mathcal{M}(\mathbf{a} + t\,\mathbf{ab}) \, \mathbf{ab}} \, \mathrm{d}t, \quad \text{with } \mathbf{e}_i = \mathbf{ab},$$

$$Q_\mathcal{M}(K) = \frac{36}{3^{\frac{1}{3}}} \frac{(\int_K \sqrt{\det(\mathcal{M}(\mathbf{x}))} \, \mathrm{d}\mathbf{x})^{\frac{2}{3}}}{\sum_{i=1}^6 \ell_\mathcal{M}^2(\mathbf{e}_i)} \in [0,1].$$

We choose a tolerance α equal to 0.8. We want to emphasize that the set of all the discrete meshes that are unit meshes with respect to a unique $\mathcal{M}$ contains an infinite number of meshes.

2.2 Continuous interpolation error

Given a smooth function u, to each unit mesh $\mathcal{H}$ with respect to $\mathcal{M}$ corresponds a local interpolation error $|u - \Pi_{\mathcal{H}} u|$. In [1, 2], it is shown that all these interpolation errors are well represented by the so-called continuous interpolation error related to $\mathcal{M}$, which is locally expressed in terms of the Hessian H_u of u as follows:

$$
\begin{aligned}
|u - \pi_{\mathcal{M}} u|(\mathbf{x}, t) &= \frac{1}{10} \text{trace}(\mathcal{M}^{-\frac{1}{2}}(\mathbf{x}) \, |H_u(\mathbf{x}, t)| \, \mathcal{M}^{-\frac{1}{2}}(\mathbf{x})) \\
&= \frac{1}{10} d(\mathbf{x})^{-\frac{2}{3}} \sum_{i=1}^{3} r_i(\mathbf{x})^{\frac{2}{3}} {}^t\mathbf{v}_i(\mathbf{x}) \, |H_u(\mathbf{x}, t)| \, \mathbf{v}_i(\mathbf{x}),
\end{aligned}
\tag{2}
$$

where $|H_u|$ is deduced from H_u by taking the absolute values of its eigenvalues and where time-dependency notations ",t)" have been added for use in next sections.

3 Mesh adaptation for laminar flow

We write in short the Navier-Stokes equations as follows:

$$
\Psi(W) = 0 \quad \text{with} \quad \Psi(W) = \frac{\partial W}{\partial t} + \nabla \cdot \mathcal{F}(W) \ + \ boundary \ conditions
\tag{3}
$$

where notation $\mathcal{F}(W)$ involves the classical inviscid and viscous fluxes. We are interested in expressing the approximation error of a functional

$$
j \ = \ (g, W)
$$

depending on the unknown state W, in terms of interpolation error for functions of the state, weighted by derivatives of the adjoint. The continuous adjoint system related to the objective functional writes:

$$
W^* \in \mathcal{V} \ , \ \forall \psi \in \mathcal{V} \ : \ \left(\frac{\partial \Psi}{\partial W}(W)\psi, W^* \right) - (g, \psi) \ = \ 0.
\tag{4}
$$

From Functional Analysis standpoint, a well-posed continuous adjoint system can be derived for any functional output as far as the linearized system is well posed. The discrete adjoint systeme writes:

$$
W_h^* \in \mathcal{V}_h \ , \ \forall \phi_h \in \mathcal{V}_h \ : \ \left(\frac{\partial \Psi_h}{\partial W}(W_h)\phi_h, W_h^* \right) - (g, \phi_h) \ = \ 0.
\tag{5}
$$

The error on functional can be written:

$$\delta j = (g, \Pi_h W - W_h) \approx (W_h^*, \Psi(W) - \Psi_h(\Pi_h W)).$$

The method proposed here involves some heuristics. Indeed, we assume that the interpolate of the adjoint is close to the discrete adjoint:

$$\Pi_h W^* \approx W_h^*. \tag{6}$$

Therefore:

$$\delta j \approx (\Pi_h W^*, \Psi(W) - \Psi_h(\Pi_h W)).$$

According to the *a priori estimate* established in [6], we have:

$$\delta j \approx \sum_{mn} \int_\Omega \ G_{m,n}(W, \nabla W_m^*, \lambda(W_m^*)) \ |S_{m,n}(W) - \Pi_h S_{m,n}(W)| \ dv.$$

where $G_{m,n}$ is a function of W, ∇W^*, and $\lambda(W^*)$, maximal eigenvalue of W^*'s Hessian, and $S_{m,n}$ depends only on W.

A continuous error model is derived by replacing the mesh by a metric $\mathcal{M}$ and the interpolation error by $1 - \pi_{\mathcal{M}}$ by the continuous analog, as introduced in Section 2:

$$\delta j \approx \sum_{mn} \int_\Omega G_{m,n}(W, \nabla W_m^*, \lambda(W_m^*))|(1 - \pi_h) \ S_{m,n}(W)| \ dv.$$

Let us define the positive symmetric matrix

$$\mathbf{H}(\mathbf{x}, t) = \sum_{m,n} G_{m,n}(W, \nabla W_m^*, \lambda(W_m^*)) \ |H_{S_{m,n}(W)}| \tag{7}$$

where $|H_{S_{m,n}(W)}|$ holds for the absolute value of the Hessian matrix of function $S_{m,n}(W)$. Then we are interested into minimising the following error model:

$$\mathbf{E}(\mathcal{M}) \ = \ \int_0^T \int_\Omega \mathrm{trace} \left(\mathcal{M}^{-\frac{1}{2}}(\mathbf{x}, t) \, \mathbf{H}(\mathbf{x}, t) \, \mathcal{M}^{-\frac{1}{2}}(\mathbf{x}, t) \right) \, \mathrm{d}\Omega \, \mathrm{d}t$$

The mesh optimization problem writes:

$$\text{Find } \mathcal{M}_{opt} = \mathrm{Argmin}_{\mathcal{M}} \, \mathbf{E}(\mathcal{M}), \tag{8}$$

under the constraint of bounded mesh fineness:

$$\mathcal{C}_{st}(\mathcal{M}) = N_{st}, \tag{9}$$

where N_{st} is a specified total number of nodes. Since we consider an unsteady problem, the space-time (st) complexity used to compute the solution takes into account the time discretization. The above constraint then imposes the total number of nodes in the time integral, that is:

$$\mathcal{C}_{st}(\mathcal{M}) = \int_0^T \tau(t)^{-1} \left(\int_\Omega d_\mathcal{M}(\mathbf{x}, t) \mathrm{d}\mathbf{x} \right) \mathrm{d}t \tag{10}$$

where $\tau(t)$ is the time step used at time t of interval $[0, T]$.

Let us assume that at time t, we seek for the optimal continuous mesh $\mathcal{M}_{go}(t)$ which minimizes the instantaneous error, *i.e.,* the spatial error for a fixed time t:

$$\tilde{\mathbf{E}}(\mathcal{M}(t)) = \int_\Omega \mathrm{trace}\left(\mathcal{M}^{-\frac{1}{2}}(\mathbf{x}, t)\, \mathbf{H}(\mathbf{x}, t)\, \mathcal{M}^{-\frac{1}{2}}(\mathbf{x}, t) \right) \mathrm{d}\mathbf{x}$$

under the constraint that the number of vertices is prescribed to

$$\mathcal{C}(\mathcal{M}(t)) = \int_\Omega d_{\mathcal{M}(t)}(\mathbf{x}, t)\, \mathrm{d}\mathbf{x} = N(t). \tag{11}$$

Solving the optimality conditions provides the *optimal goal-oriented ("go") instantaneous continuous mesh* $\mathcal{M}_{go}(t) = (\mathcal{M}_{go}(\mathbf{x}, t))_{\mathbf{x} \in \Omega}$ at time t defined by:

$$\mathcal{M}_{go}(\mathbf{x}, t) = N(t)^{\frac{2}{3}}\, \mathcal{M}_{go,1}(\mathbf{x}, t), \tag{12}$$

where $\mathcal{M}_{go,1}$ is the optimum for constraint $\int_\Omega d_\mathcal{M}(\mathbf{x}, t)\mathrm{d}\mathbf{x} = 1$:

$$\mathcal{M}_{go,1}(\mathbf{x}, t) = \left(\int_\Omega (\det \mathbf{H}(\bar{\mathbf{x}}, t))^{\frac{1}{5}} \mathrm{d}\bar{\mathbf{x}} \right)^{-\frac{2}{3}} (\det \mathbf{H}(\mathbf{x}, t))^{-\frac{1}{5}}\, \mathbf{H}(\mathbf{x}, t). \tag{13}$$

The corresponding optimal instantaneous error at time t writes:

$$\tilde{\mathbf{E}}(\mathcal{M}_{go}(t)) = 3\, N(t)^{-\frac{2}{3}} \left(\int_\Omega (\det \mathbf{H}(\mathbf{x}, t))^{\frac{1}{5}} \mathrm{d}\mathbf{x} \right)^{\frac{5}{3}} = 3\, N(t)^{-\frac{2}{3}} \mathcal{K}(t) \tag{14}$$

with $\mathcal{K}(t) = \left(\int_\Omega (\det \mathbf{H}(\mathbf{x}, t))^{\frac{1}{5}} \mathrm{d}\mathbf{x} \right)^{\frac{5}{3}}$. As in [5] (in which details can be found, the space-time problem is then solved by optimising $N(t)$ under the conditions that:

$$\int_0^T \tau(t)^{-1}\, N(t)\, \mathrm{d}t \;=\; N_{st}.$$

The unsteady optimality system (3,4,13) is solved by applying the global unsteady fixed-point adaptation algorithm introduced in [5]. Figures from an application to the impulsive start around an airfoil (Reynolds 1000) are depicted in Figure 1 which show meshes an flow density at three time levels of the adaptive calculation.

Anca Belme, Adrien Loseille, Hubert Alcin, Frédéric Alauzet, Stephen Wornom and Alain Dervieux
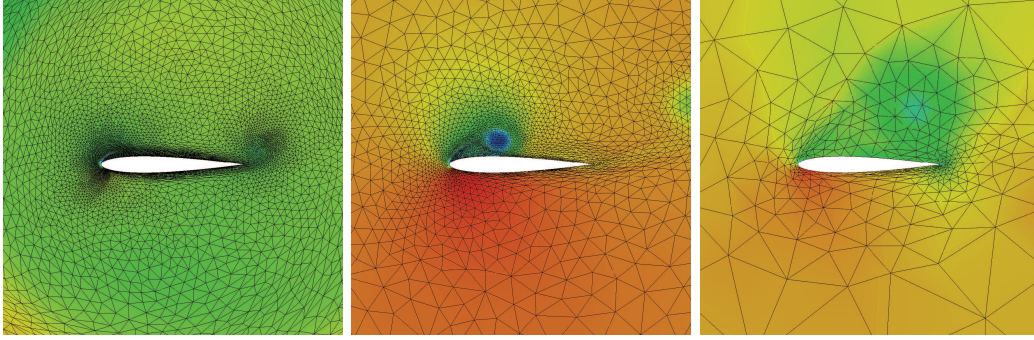
Figure 1: Impulsive start around an airfoil, density and mesh at three times

4 Mesh adaptation for turbulent flow

Convergence of LES models is a complicated issue since their filter depends of mesh size. As a consequence, when h tends to zero, LES comes closer to DNS, a positive behavior which, however does not give any chance to mesh convergence. Returning to a finite size mesh practical application, it is anticipated that the solution is made of filtered small scales and of well-resolved larger scales. In larger scales, we mean quasi-steady large scales and fluctuating intermediate scales, and both need accurate resolution. Let us consider the case where LES is obtained by adding as filter a Boussinesq term to the Navier-Stokes model. The LES model can be built on a modern model like WALE, which, in contrast to the classical Smagorinsky model, does not suffer of spuriously large dissipation in boundary layers, [8]. This can also be built by a dynamic version of the Smagorinsky model. This class of LES models can be described as discretizations of the *continuous filtered Navier-Stokes equations* for $W = ()$ built from the combination of the Navier-Stokes equation with a filter term and which we write in short:

$$\frac{\partial W}{\partial t} + \nabla \cdot \mathcal{F}(W) = -\tau^{LES}(W). \tag{15}$$

The Boussinesq term $\tau^{LES}(<W>)$ writes:

$$\tau^{LES}(W) = \nabla \cdot \ \mu_{\text{sgs}} \nabla \begin{pmatrix} 0 \\ u \\ v \\ 0 \end{pmatrix} \quad \text{with} \quad \mu_{\text{sgs}} = \rho \left(C_{\text{s}}\Delta\right)^2 \left|\widetilde{S}\right| \ , \tag{16}$$

and is weighted by a scalar field, the local filter width $\Delta = \Delta(\mathbf{x}, t)$. C_{s} is the Smagorinsky coefficient (in practice we shall use the dynamic one, which is a function of W) and

$$\left|\widetilde{S}\right| = \sqrt{2\widetilde{S_{ij}}} \quad \text{with} \quad \widetilde{S_{ij}} = \frac{1}{2}\left(\frac{\partial \tilde{u}_i}{\partial x_j} + \frac{\partial \tilde{u}_j}{\partial x_i}\right).$$

As already mentioned, a LES model is a discretization of (15) and needs to be computed on a mesh. Then the best predictivity could be classically attained when the local filter width is taken equal to local mesh size.

Instead, we keep some more time the continuous formulation, and we consider the case where the local filter size is prescribed. It is prescribed as a given continuous scalar field. We call filtered continuous model the Navier-Stokes model with the extra Boussinesq term relying on the continuous filter size:

$$\frac{\partial W}{\partial t} + \nabla \cdot \mathcal{F}(W) = -\nabla \cdot \ \mu(\Delta)\nabla \begin{pmatrix} 0 \\ u \\ v \\ 0 \end{pmatrix} \quad \text{with} \quad \mu(\Delta) = \rho \, (C_{\mathrm{s}}\Delta)^2 \left|\widetilde{S}\right|, \qquad (17)$$

the solution of which is denoted W_Δ.

Given an discrete approximation of (18) which produce a solution $W_\Delta(\mathcal{M})$ on a mesh $\mathcal{M}$, our concern is now the following problem:

For a prescribed Δ, find a mesh that is therefore independant of Δ, of a given number of nodes, which minimizes:

$$|\mathcal{E}_\Delta(\mathcal{M})| = |W_\Delta - W_\Delta(\mathcal{M})|.$$

Symbols $\mathcal{E}_\Delta(\mathcal{M})$ define the *weak error, i.e.* the deviation between the discrete LES and its continuous analog, both defined for the given (mesh independant) filter width. A basic choice for the norm is an integral on a time interval $0, T$ of the L^1 spatial norm of the instantaeous deviation.

$$|\mathcal{E}_\Delta(\mathcal{M})| = |\int_0^T \int_\Omega W_\Delta - W_\Delta(\mathcal{M}) \, d\mathbf{x} \, dt|.$$

Given now a number of nodes, the optimal mesh for reducing the weak error on a scalar output can be obtained by applying our global unsteady fixed point method with a prescribed filter size. Thanks to the global unsteady fixed point adaptation algorithm, the mesh adaptation concentrates resolution on unsteady turbulent structures.

Δ **strategies.** The above optimal mesh depends on the parameter Δ. In fact we want the best mesh $\mathcal{M}_{opt}$ for the Δ which is the local mesh size of this optimal mesh. We can solve this by an external fixed point.

5 An example

The proposed mesh adaptation method is applied to the computation of the flow around an offshore platform with a very complex geometry. This flow was accurately computed and compared with experiments in a specialized conference [7]. For the present mesh adaptive calculation, we take into account a large enough time interval and compute the

adjoint on this interval. The resulting mesh adaptation criterion can be concentrated on the generation of a single mesh, since the vortices concentrate on a region of wake which is well identified by the algorithm. See Figure 2.

6 Acknowledgements:

REFERENCES

[1] A. Loseille and F. Alauzet, "Continuous mesh framework. Part I: well-posed continuous interpolation error ". *SIAM J. Num. Anal.*, 49(1):38–60, 2011.

[2] A typical value for the Smagorinsky coefficient is $C_s = 0.1$ that is often used, especially in the presence of the mean shear. .

A. Loseille and F. Alauzet, "Continuous mesh framework. Part II: validations and applications ". *SIAM J. Num. Anal.*, 49(1):61–86, 2011.

[3] H. Alcin, A. Belme, A. Loseille, F. Alauzet, A. Dervieux, "Goal-Oriented mesh adaptation for vortex shedding flows ", ECCOMAS 2012, Vienna, Austria, 2012.

[4] H. Alcin, "Resolution d'ecoulements instationnaires et adjoints ", Thesis, university of Nice, 2012.

[5] A. Belme, A. Dervieux and F. Alauzet: "Goal-oriented anisotropic mesh adaptation for unsteady flows ". Proceeding ECCOMAS CFD Conference, Lisbon, june 2010.

[6] A. Belme, "Unsteady Aerodynamics and adjoint state method", PhD thesis, Nice, 2011.

[7] S. Sirnivas, S. Wornom, A. Dervieux, B. Koobus, O. Allain, "A study of LES models for the simulation of a turbulent flow around a truss spar geometry ", OMAE2006-92355, Proceedings of OMAE'06, 25rd International Conference on Offshore and Arctic Engineering, 4-9 June, 2006, Hamburg, Germany.

http://www-sop.inria.fr/members/Alain.Dervieux/OMAE2007.pdf

[8] F. Nicoud and F. Ducros. "Subgrid-scale stress modelling based on the square of the velocity gradient tensor". *Flow Turbulence and Combustion*, 62(3):183–200, 1999

[9] H. Ouvrard, B. Koobus, A. Dervieux and M. V. Salvetti, "Classical and variational multiscale LES of the flow around a circular cylinder on unstructured grids". *Computers and Fluids*, 39 (2010) 10831094

Anca Belme, Adrien Loseille, Hubert Alcin, Frédéric Alauzet, Stephen Wornom and Alain Dervieux
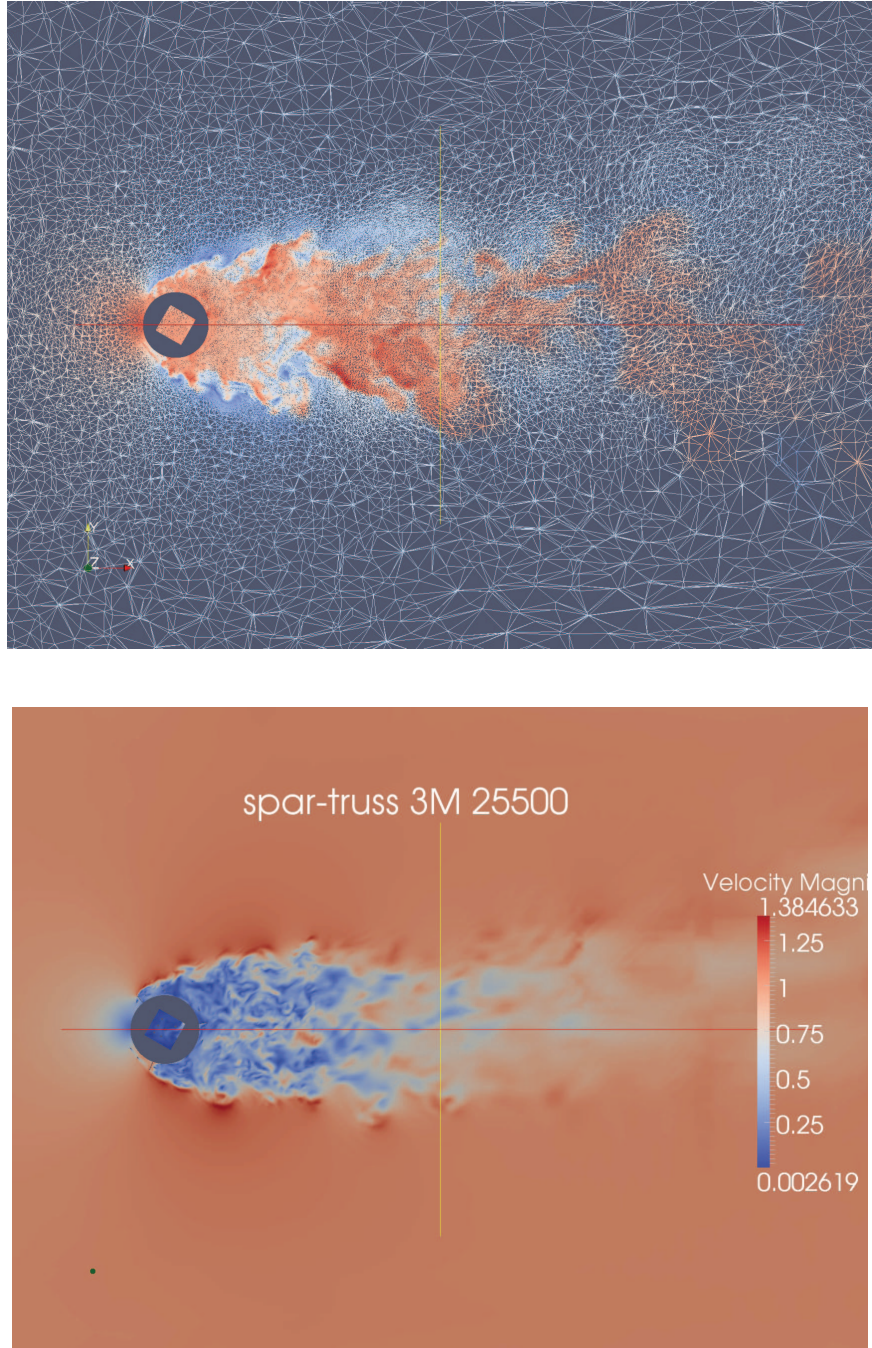
Figure 2: Mesh adaptive flow around an offshore platform: velocity module at two different times, the first one with mesh wireframe representation.

VI International Conference on Adaptive Modeling and Simulation
ADMOS 2013
J. P. Moitinho de Almeida, P. Díez, C. Tiago and N. Parés (Eds)

HIERARCHICAL MODEL (HIMOD) REDUCTION FOR ADVECTION-DIFFUSION-REACTION PROBLEMS

PABLO J. BLANCO[*], SIMONA PEROTTO[§], ALESSANDRO VENEZIANI[‡]

[*]Laboratório Nacional de Computação Científica
Av. Getúlio Vargas 333, Quitandinha, 25651-075, Petrópolis, Brazil
e-mail:pjblanco@lncc.br

[§]MOX, Dipartimento di Matematica "F. Brioschi"
Politecnico di Milano, Piazza Leonardo da Vinci 32, I-20133 Milano, Italy
e-mail:simona.perotto@polimi.it, www1.mate.polimi.it/~simona/

[‡]Department of Mathematics and Computer Science, Emory University
400 Dowman Dr., 30322, Atlanta, GA, USA
e-mail:ale@mathcs.emory.edu, www.mathcs.emory.edu/~ale/

Key words: Hierarchical Model Reduction, Geometrical Multiscale

Abstract. The effective numerical description of challenging problems arising from engineering applications demands often the selection of an appropriate reduced - aka "surrogate" - model. The latter should result from a trade-off between reliability and computational affordability (see, e.g., [1, 8]). Different approaches can be pursued to set up the reduced model. In some cases, one can take advantage of specific features of the problem at hand for devising an effective *ad-hoc* model reduction. This is the case, for instance, of problems featuring a prevalent direction in the dynamics of interest, as in the modeling of the hemodynamics in arterial trees or of the hydrodynamics in a channel network. In this context, a possible approach is represented by the so-called *geometrical multiscale*, where dimensionally heterogeneous models are advocated for describing interactions at different scales: essentially, a lower dimensional (for instance, 1D) model is locally replaced by a higher dimensional (for instance, 3D) model to include local relevant transversal dynamics. This approach has been successfully applied both in hemodynamics (see, e.g., [3, 4]) and in river dynamics (see, e.g., [6, 5]).

As an alternative to the geometrical multiscale formulation, the so-called *hierarchical modeling* has been advocated in [2, 7]. The basic idea is to perform a classical finite element discretization along the mainstream direction of the problem at hand coupled with a modal decomposition for the transversal dynamics. The rationale behind this approach is that the transversal dynamics can be suitably captured by a few degrees of modal freedom. In addition, the dimenson of the modal discretization can be suitably adapted along the main direction, according to the local features of the transversal component of

the solution. This allows to improve hierarchically and adaptively the descripton of the local transversal dynamics, still in the context of a "psychologically" 1D solver.

Comparison and coupling with the geometrical multiscale approach, effective criteria for the selection of the hierarchical modal basis relying upon a principal component analysis (and alternative to the ones introduced in [2, 7]), applications to real 3D problems (such as the circulatory system) are steps to be tackled in the development of an effective HiMod approach in engineering applications.

REFERENCES

[1] *Large Scale Inverse Problems and Quantification of Uncertainty.* Biegler, L., Biros, G., Ghattas, O., Heinkenschloss, M., Keyes, D., Mallick, B., Marzouk, Y., Tenorio, L., van Bloemen Waanders, B., Willcox, K. eds. Wiley, Chichester UK (2010).

[2] Ern, A., Perotto, S. and Veneziani, A. Hierarchical model reduction for advection-diffusion-reaction problems. In: Kunisch, K., Of, G. and Steinbach, O. eds., Numerical Mathematics and Advanced Applications, Springer-Verlag, Berlin Heidelberg, (2008) 703–710.

[3] Formaggia, L., Nobile, F., Quarteroni, A. and Veneziani, A. Multiscale modelling of the circulatory system: a preliminary analysis. *Comput. Visual. Sci.* (1999) **2**: 75–83.

[4] Leiva, J.S., Blanco, P.J., Buscaglia, G.C. Partitioned analysis for dimensionally-heterogeneous hydraulic networks. *Multiscale Model. Simul.* (2011) **9**: 872–903.

[5] Mauri, L., Perotto, S. and Veneziani, A. Adaptive geometrical multiscale modeling for hydrodynamic problems. To appear in: Cangiani, A., Davidchack, R.L., Georgoulis, E., Gorban, A.N., Levesley, J., Tretyakov, M.V. eds., Numerical Mathematics and Advanced Applications, Springer-Verlag, Berlin Heidelberg, (2013).

[6] Miglio, E., Perotto, S. and Saleri, F. Model coupling techniques for free-surface flow problems. Part I. *Nonlinear Analysis* (2005) **63**: 1885–1896.

[7] Perotto, S., Ern, A. and Veneziani, A. Hierarchical local model reduction for elliptic problems: a domain decomposition approach. *Multiscale Model. Simul.* (2010) **8**, No. 4: 1102–1127.

[8] *Model Order Reduction: Theory, Research Aspects and Applications.* Schilders, W.H., Vorst, H.A. van der, Rommes, J. eds, Mathematics in Industry, **13** Springer-Verlag, Berlin (2008).

GENERATING ASSEMBLY MODELS FOR ADAPTIVE SIMULATIONS

F. BOUSSUGE[*†], J-C. LEON[†], S. HAHMANN[†], S. PAVOT[+], E. FLORENTIN[+], L. CHAMPANEY[+]AND L. FINE[*]

[*] EADS – Innovation Works (IW)
5, quai Marcel Dassault, 92150 Suresnes, France
e-mail: flavien.boussuge@eads.net, lionel.fine@eads.net

[†] Grenoble University - INRIA
INRIA Rhône-Alpes, 655 av de l'Europe, Montbonnot, 38334 Saint-Ismier, France
email: jean-claude.leon@grenoble-inp.fr, Stefanie.hahmann@inria.fr, www.inria.imagine.fr

[+] Laboratoire de Mécanique et Technologie – ENS Cachan
61 avenue du Président Wilson, 94235 CACHAN, France
email: eric.florentin@lmt.ens-cachan.fr, laurent.champaney@lmt.ens-cachan.fr, sylvain.pavot@lmt.ens-cachan

Key words: assembly, CAD models, idealization, mesh generation, functional information, a posteriori error estimator.

Abstract. Aircraft and automotive industries face increasing needs in generating large and complex simulation models, especially at the level of assemblies, sub-systems of complex products. Starting from the digital representation of sub-systems, i.e., digital mock-ups (DMU), as available from CAD software, the major steps of the simulation model generation are described. This incorporates the geometry analysis of the DMU to derive functional information. Subsequently, this information is used to perform model simplifications and domain decomposition consistently with the simulation objectives. Given the complexity of these models, the domain decomposition is a key issue to adaptive simulations to be coupled with COFAST software as well as error estimators using the LATIN method to avoid solving large systems and to take advantage of their decoupling capabilities. An assembly of bolted components illustrates the proposed approach.

1   INTRODUCTION

Companies, especially in the aircraft and automotive industries are increasingly interested in setting up numerical simulations throughout a product development process (PDP). Because of the inherent complexity of their products, simulations are not only targeting isolated components but there is now a strong interest at studying the behavior of one or more subsystems of these products [1, 2]. The corresponding requirement is the setting of rather complex FE models that cannot be currently handled within the time scale prescribed

by an industrial product development project.

The purpose of the proposed contribution is to describe how FE simulation models can be derived from assembly CAD models and how adaptive simulations can take place with these large scale models. Consequently, the contribution focuses on the major steps of simulation model preparation and its interactions with an adaptive simulation process. The target addressed falls into the scope of the national research project, ROMMA [3], and, if all the connections between the steps are not completed yet, the paper will report the current progress in each of them.

On the one hand, domain decomposition approaches have been proposed but concentrate on the FE mesh generation process [9]. On the other hand, functional information attached to components is a current approach of design methodologies [10, 11, 12]. In this case, the top-down approach followed does not address the detailed connection with a 3D model [13]. This observation calls for new developments that address a detailed connection between 3D geometric entities and low level functions. To be able to process large assembly simulation models with an adaptive approach, the determination of components' interfaces is a key issue since it is a basic input of domain decomposition approaches. Interfaces are also part of the hypotheses of finite element analyses (FEA) to set contact with or without friction or even to merge domains representing components in accordance to simulation objectives. Assembly models as available either in or from CAD environments don't incorporate the description of interfaces between components. Contributions in this area have addressed B-Rep NURBS models as available in CAD software [2] or facetted representations [14]. However, in either configuration, the interfaces addressed reduce to contact areas, which is not the only configuration found in DMUs and no connection is initiated with basic functions of components, whereas this is a mandatory approach to produce a bottom-up approach connecting geometric information to component functions.

The paper structure is as follows. Section 2 describes how the automated enrichment of components with functional designations and interfaces can help producing simulation models with dimensional reduction of components. Section 3 briefly discusses some model requirements for adaptive simulations. Finally, section 4 describes the major steps of a domain decomposition approach taking advantage of a functionally enriched assembly model and shows how adaptive simulations can take advantage of this approach.

2 FROM CAD ASSEMBLIES TO SIMPLIFIED SIMULATION MODELS

2.1 Automated enrichment of CAD models with functional information

In a first place, studying the content and structure of an assembly model, as available in a Product Data Management System (PDMS), reveals that product assemblies or Digital Mock-Ups (DMUs) reduce to a set of components located in 3D space without geometric relationships between them. Complementarily, simulation models for assemblies strongly need geometric interfaces between components to be able to set up boundary conditions between them and/or meshing constraints, e.g. conformal mesh requirements.
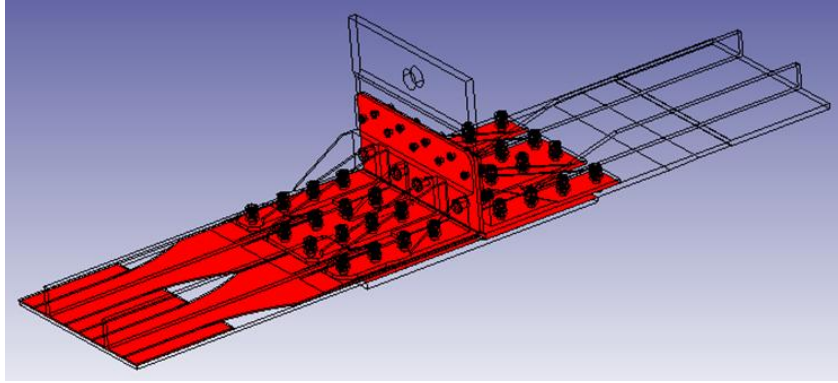
Figure 1: Interfaces extracted from a CAD assembly model. Components are represented with a wireframe setting whereas geometric interfaces are depicted in red.

Another observation derived from this analysis is the prominence of component functions as a means for specifying component simplifications/idealizations. This leads to a first step of the assembly processing scheme:

- Identification of geometric interfaces between components (see Figure 1). It has to be noticed the interfaces taking place between components in a DMU don't reduce to contacts and clearances. Indeed, components may interfere depending on their *conventional representation*. As an example, screws and nuts with threaded areas are often replaced by cylindrical ones. In this case, the screw has a cylinder diameter equal to the outer diameter of the thread. A similar setting applies to the nut. As a result, the interface between the screw and the nut becomes an interference. This observation must be taken into account when defining the shape transformation required when adapting the assembly model to simulation objectives [1];

- Assignment of functional designations to components. Currently, functional information is automatically added to some categories of components using a qualitative reasoning process. In the assembly of Figure 2, bolted junctions illustrate categories of components enriched with functional information, i.e. cap screws, nuts, locking nuts;

- As a result, components are structured geometrically, i.e. key geometric interfaces are located on the boundary of each component and in its neighborhood, as well as from a functional standpoint, i.e. functional designations of components fit into a taxonomy and set constraints over technological data describing the interfaces involved in their definitions [4]. Additionally, screws are associated to their load cycles, i.e. the set of components they tighten. This is obtained automatically from a qualitative reasoning process combining the 3D geometry of interfaces with reference states representing qualitative loading configurations.

All this information, derived qualitatively, contributes to the location of boundary

condition areas in the assembly to set up FE models. It is also key information that can be used subsequently during adaptive FE analyses incorporating a posteriori error estimators.
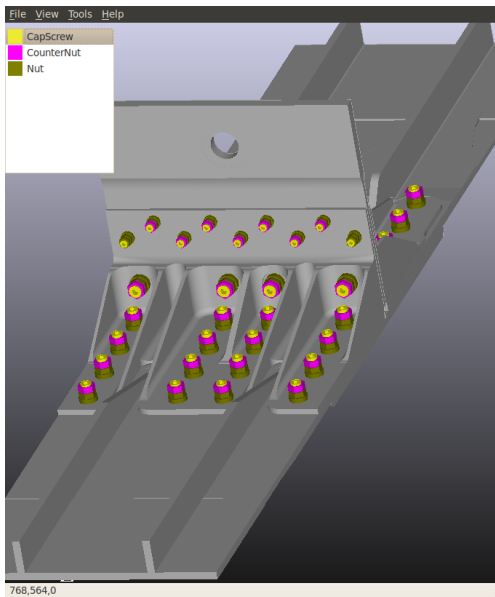
Figure 2: Assembly model with functionally enriched components, i.e. screws, nuts, counter nuts.

2.2 Simplifying assembly models using components' idealizations

Because assembly models can lead to highly complex simulation models when it focuses on car and/or aircraft models, idealizations of components are key issues of simulation model preparation since dimensional reductions is a means to generate efficient simulation models while keeping the number of Fes as low as possible. Idealization or dimensional reduction of components or sub domains is a common requirement to meet simulation objectives, especially when considering early design phases of complex structures.

To this end, component segmentation has been set up to analyze component morphology and decompose a component into sub-domains representing construction primitives. Indeed, a component is decomposed into a construction graph representing a set of non trivial construction trees. Depending on the primitive shapes, the dimensional reduction constraints help selecting the most suited tree that produce the idealized component (see Figure 3). This idealization algorithm is robust since the classical weakness of sub domain connection does not hold here. Indeed, the connections between sub domains is guided by the interactions between the primitives in the construction tree (see Figure 3d). These idealization algorithms take also into account meshing constraints, i.e. locations of stiffeners interact with the shape and size of FEs and choosing 'internal' or 'external' stiffener positions rather than mid-surface can improve the FE mesh quality.

Figure 4 illustrates the result obtained for the assembly depicted at Figures 1 and 2 after the idealization of the major components. In this model, bolts can be idealized as beams using the functional information derived in a first place.

This result is a first step to address large scale simulation models as needed for COFAST software [5] to meet simulation objectives at a rather global level.
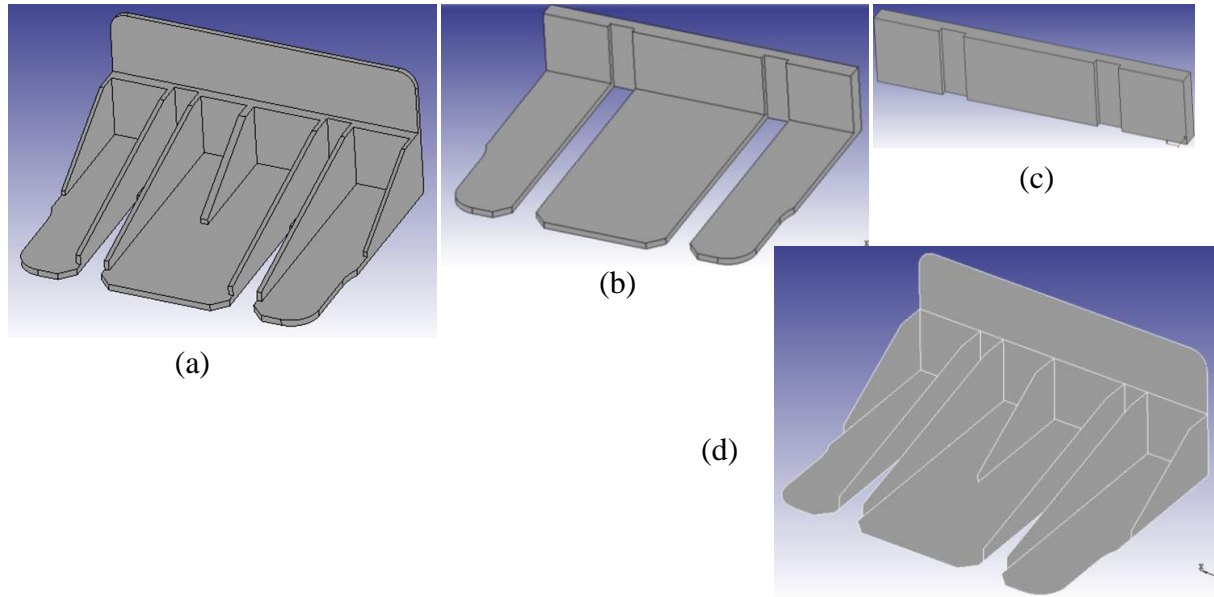
Figure 3: (a), (b), (c) construction tree derived from the construction graph as required for the dimensional reduction constraints. (d) idealized component derived from the construction tree.

3 ADAPTIVE FEA APPROACHES AND SIMULATION MODEL REQUIREMENTS

The iterative scheme that is used in COFAST, is derived from the LATIN method. The main principle is to separate the equations in order to avoid solving simultaneously a global and a nonlinear problem. The procedure searches for solutions that alternatively satisfy the global linear equations (kinematic admissibility and equilibrium on a substructure) and then, the local equations (interface equations). This leads to a decoupling of the problem. Because very few iterations of the LATIN method generate a solution over the whole time interval, the initialization overacts on the whole time interval. The solution obtained with this procedure ends up with a very low computation cost and can be parallelized to obtain a very good approximation of the solution.

Functional information becomes also important to set up simulation models dedicated to local analyses. Figure 5 illustrates how functional information can be used to simplify bolts and derive control areas around these bolts that are used to precisely meet simulation objectives. Here, the sub domains are used to model the friction phenomenon around each bolt as develops according to the Rotscher's cone [8]. Additionally, these sub domains monitor the mesh generation process so that meshing strategies can be efficiently set up when processing complex bolted assemblies.

(a)   (b)
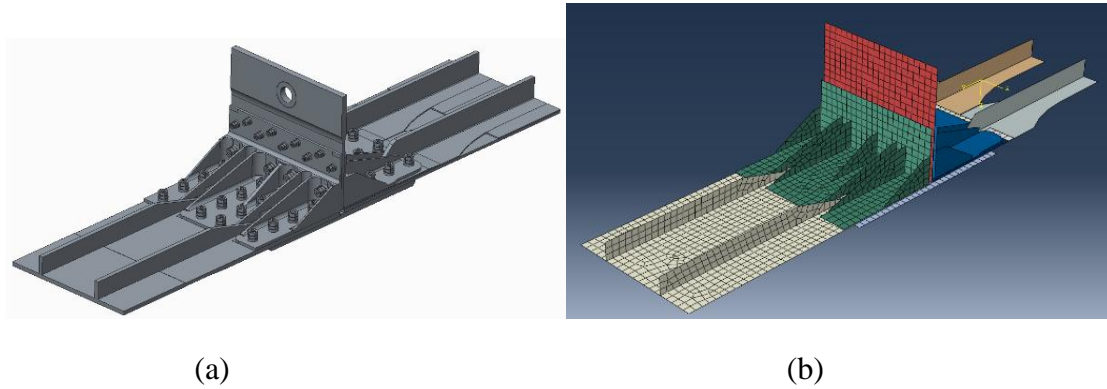
Figure 4: (a) CAD model of an assembly, (b) model demonstrating the results obtained after component segmentations, an idealization process and a mesh generation. For illustration purposes some components have been idealized but not meshed, others have been idealized and meshed.

4 SIMULATION MODEL GENERATION BASED ON DOMAIN DECOMPOSITION

4.1 Setting up a domain decomposition for adaptive simulations

Functional information derived from the DMU is of qualitative type, e.g. cylindrical fittings are not quantified but classified as 'tight fit' or 'loose fit'. However, this information acts as a template that can be used when setting up the simulation parameters required at interfaces between components. Because components interfaces are clearly identified and can be categorized from a mechanical point of view, the simulation model preparation is strengthened: the number and type of parameters needed at various interfaces can be unambiguously identified, thus avoiding inconsistencies that could arise when setting up complex simulation models.

This concept of template can be extended to include shape transformations as needed to meet simulation objectives. Let us consider a configuration where the objective is to study the stress field around bolts and take into account the friction phenomenon around each bolt. Then, the template-based approach can incorporate all the shape transformations needed for this simulation objective (see Figure 5). The transformations can be listed as follows:

- Removal of the counter nut because the targeted stress field is in the tightened components rather than the screws,

- Merging screw and nut into a single sub-domain because their interface is not of interest with respect to the simulation objective,

- Transformation of the screw heads and nuts to simplify their shape as well as their FE mesh while preserving friction areas between the bolt and the tightened components,

- Generation of a sub-domain around the screw shaft to define the friction area of interest at the interface of each tightened component and to model more precisely the stress field around the screw shaft as required in the simulation objective.
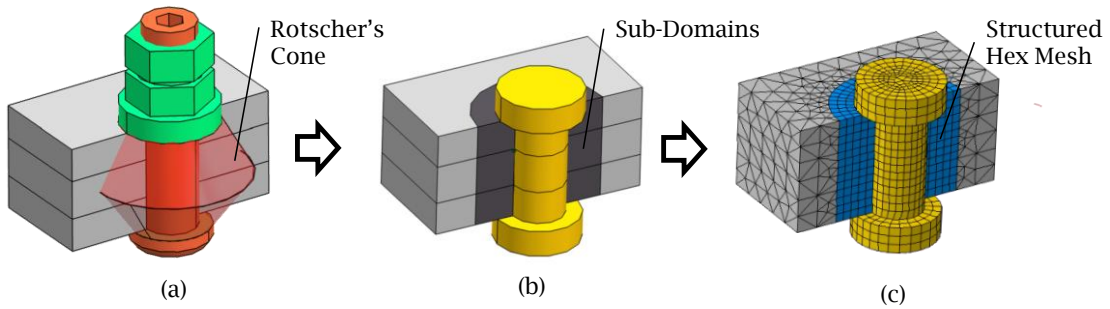
Figure 5: Template-based transformation of a bolted junction into simple mesh model with friction and contact areas definition around screw and nut.

The template-based transformation approach is parameterized with respect to the screw dimensions (diameter and length), the type of screw head (flat or hex type, …), the number of tightened components and the variants of the bolted connection (with or without locking nut, screw shaft with or without adjustement). This entirely parameterized template becomes very efficient to locate the various bolt configurations and set the appropriate interfaces with respect to the simulation objectives. Indeed, bolts can be identified from a user-specified function, e.g. bolted junction with locking function, which is a meaningful way for the user to process large assembly structures.
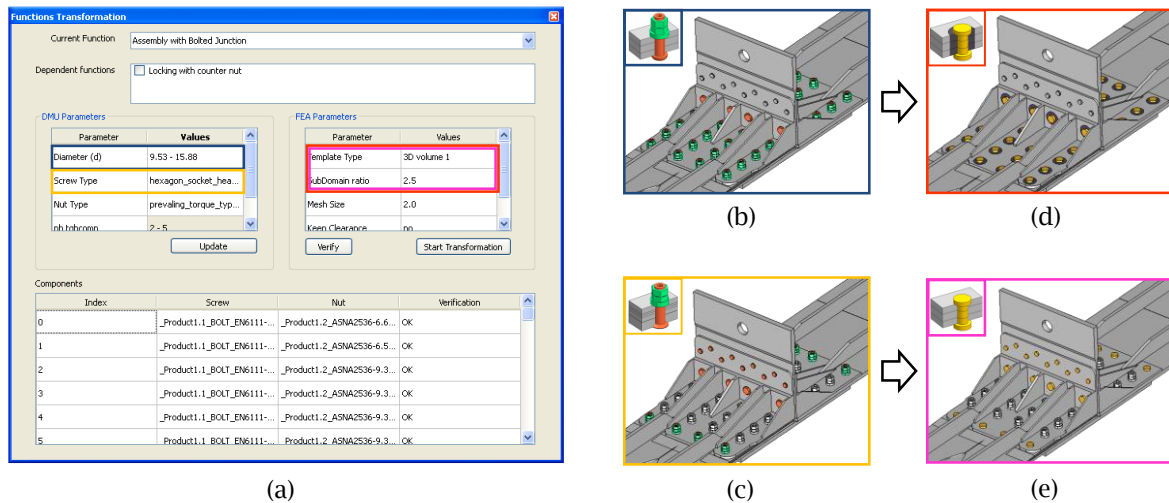
Figure 6: User interface for transformation of 'assembly Bolted Junctions' (a), filtering bolts based on diameters (b) or screw type (c), template-based transformations with (d) or without (e) sub domains.

Figure 6 illustrates the previous parameters as input parameters of the template, as seen by the user, to be able to target the proper category of bolts when generating his, resp. her, simulation model of interest. Once these transformations are performed, interface types and locations are entirely defined and can be structured to be transfered to a FE software to generate the FE mesh and set all the required boundary conditions deriving from the domain decomposition applied to the assembly model. Figure 7 shows the structure of the entities

used to generate the FE mesh and the interfaces used to specify the boundary conditions of the FE model. Figure 8 illustrates this overall process on the assembly with bolted junctions of Figure 1. Because the components identified are related to a function, sub-domains and interfaces can be assigned all the parameters required for a FEA, which significantly improves the efficiency of FE model generation and their access to adaptive simulations.
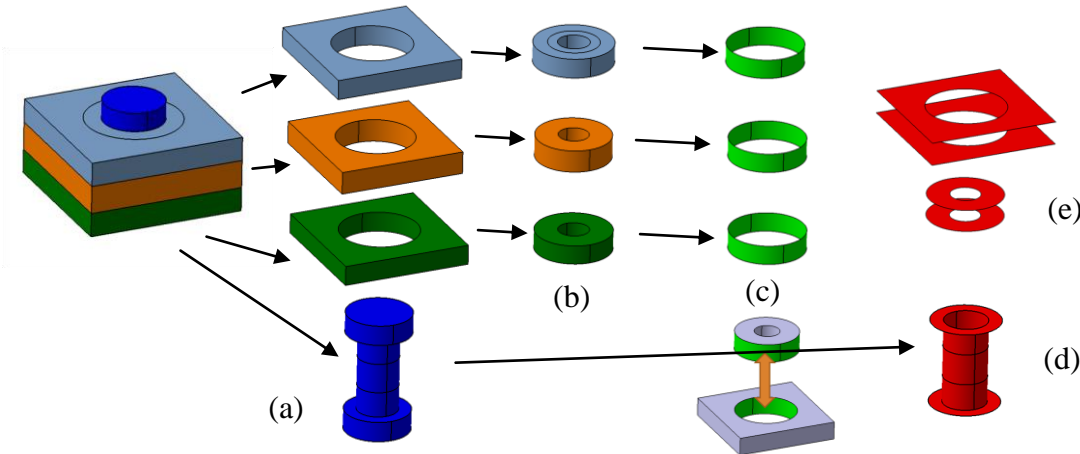
Figure 7: Domain decomposition obtained with the template-based transformation. (a) the simplified screw and nut, (b) inner domain of each tightened component containing the imprint of the screw head and nut as interfaces, (c) boundaries of inner domains, (d) interfaces around the screw and nut with an adjusted screw, (e) interfaces between tightened components.
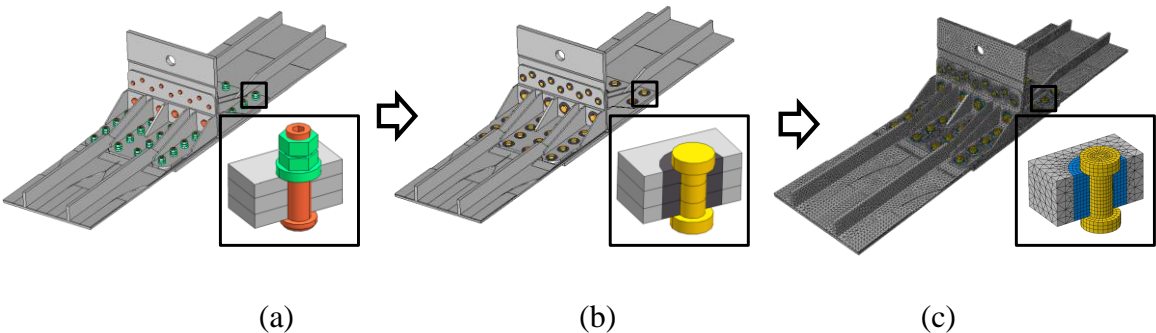
Figure 8: (a) CAD model of an assembly after the identification of functional designations of components (colors indicate the differences of functional designations), (b) model demonstrating the use of functional designations to simplify bolts and set up control volumes around them to adapt the model to the simulation objectives, (c) mesh model taking into account the interfaces obtained from the domain decomposition process.

4.2 Structuring an assembly model for adaptive simulations

Additionally, the precise location of interfaces becomes helpful for setting up input for a posteriori error estimators. Indeed, the estimator used here is based on the constitutive error relation concept. A pillar of the method is to construct admissible fields [7]. The knowledge of interfaces between substructures is then of primary importance. Nevertheless, in order to simplify the construction of admissible fields, a generic construction has been built when

some data are missing [6]. When available, the precise location of the interfaces and more generally all the knowledge about the mechanical loading can be integrated in the generic method developed and improves the quality of the computed error estimation. The first results obtained in the framework of linear elasticity have to be extended to the framework of contact with and without friction. The final objective is to obtain a tool that enables the design of a robust simulation process of assemblies through an adaptive process.

The simulation model is based on a domain decomposition of the structure. Indeed, the structure is divided into interfaces and substructures (see Figure 9).

$$\Omega_E \quad \Phi_{EE'} \quad \Omega_{E'}$$
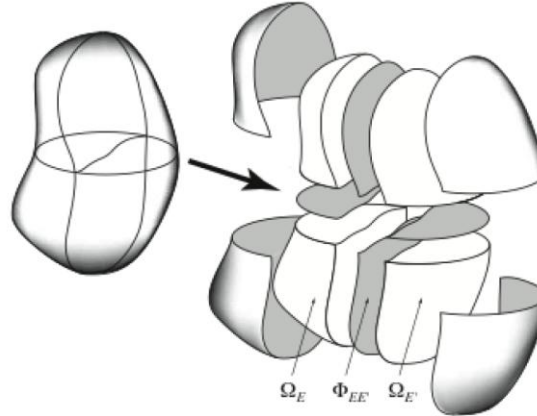
Figure 9: A structure split into sub-structures and interfaces.

Each sub-structure may correspond to a physical component. The interface may correspond to a contact with or without friction. One can also split a part into two sub-parts in order to decrease the computational cost. Another solution is to sort part by function, like for functional joints. High level mechanical contents can be inserted into some parts, e.g. non linear behavior for areas near the joint (Figure 8b).

With these partitions, the problem to solve has three sets of equations defined on the substructures and their interfaces:
- The kinematic constraints (Dirichlet's conditions and connection between interfaces and substructures);
- The equilibrium equations (Equilibrium, Neumann's condition and connection between interfaces and substructures);
- The constitutive relation equations (Classical constitutive equations and contact with friction on interfaces).

The constitutive error concept is a tool that measures the distance between the reference problem and its numerical simulation [7]. The computation of the measure in itself is easy to compute, the difficulty is to construct admissible fields. The construction of admissible fields, i.e. fields that check simultaneously kinematical constraints and equilibrium equations is a global problem. The construction of such admissible fields can lead to very high post simulations. Here, the method developed is to construct the admissible field on a sub-structure.

This local construction has a lower cost than the global method, but introduces an approximation. The first results show that for a reasonable number of substructures (up to 100), the results are very close to the one obtained by an optimization over the whole substructure (Figure 10).
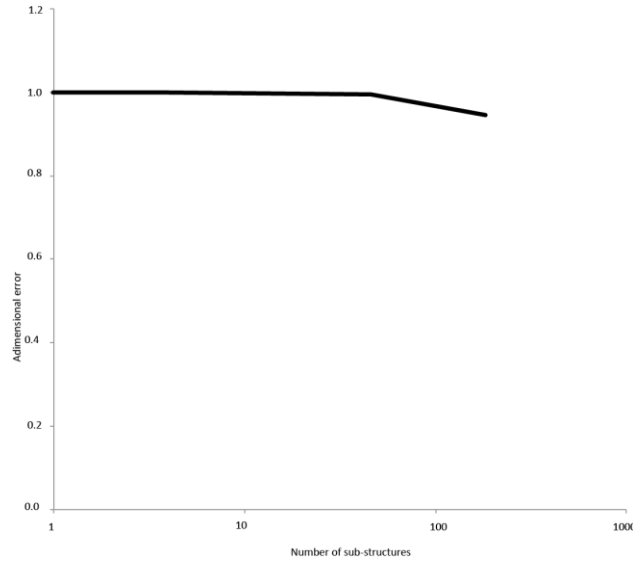
Figure 10: The quality of the estimated error vs number of substructures of interest.

Moreover, the method set up does not require a perfect description of the stress distribution and displacements at the interfaces of sub-parts to construct admissible fields. These quantities are only known in terms of generalized FE quantities. This approach introduces more flexibility and makes it possible to evaluate the error in a very simple and systematic manner on sub-parts, i.e. patches set around bolts in the example case of bolted assembly.

5 CONCLUSION

The generation of adaptive simulations for complex structures has been addressed for assembly models. Interfaces between components are a first key information that is extracted from the assembly model. From these interfaces, a qualitative reasoning process is able to derive the functional designation of some categories of components. These results have been illustrated with screws and nuts in assembly configurations. The functional designations incorporate also the location and type of interfaces between their components and their neighborhoods. This is another key information to set up adaptive analyses.

From the enriched assembly model with the functional designations of some components, it has been proposed to set up a template-based transformation that is able to produce a domain decomposition and interfaces as suited for adaptive simulations. The content of this template has been shown as fully parameterized to be able to address automatically a wide range of configurations and perform the corresponding domain decompositions.

Currently, the construction of admissible fields can be achieved efficiently with a decomposed when its interfaces are fully characterized. Current results lead to good quality

simulations as shown with the estimated error obtained.

Further work will address the extension of the current numerical simulation scheme to be able to process interfaces with contact with and without friction. Also, complementary developments will address the enlargement of the categories of components that can be identified and structured with functional designations.

REFERENCES

[1] Boussuge, F., Léon, J-C., Hahmann, S., Fine, L. An analysis of DMU transformation requirements for structural assembly simulations, *Int. Conf. ECT 2012*, Dubrovnic, 4-7 Sept., (2012).

[2] Clark, B.W., Hanks, B.W., Ernst, C.D. Conformal Assembly Meshing with Tolerant Imprinting, *Proc. of the 17th International Meshing Roundtable*, pp 267-280, (2008).

[3] ANR ROMMA project: http://romma.lmt.ens-cachan.fr/.

[4] Shahwan, A., Léon, J-C., Foucault, G., Trlin, M., Palombi, O. Qualitative behavioral reasoning from components' interfaces to components' functions for DMU adaption to FE analyses, *CAD*, **45**, pp 383-394, (2013).

[5] Champaney, L. A domain decomposition method for studying the effects of missing fasteners on the behavior of structural assemblies with contact and friction. *Computer Methods in Applied Mechanics and Engineering*, Vol 205–208, pp 121–129, (2012).

[6] Pavot, S., Florentin, E., Rougeot, P., Champaney, L. Finite element verification in the case of missing data. *Finite Elements in Analysis and Design*. Vol 64, pp 90-96, (2013).

[7] Florentin E., Guinard, S, Pasquet, P. A simple estimator for stress errors dedicated to large elastic finite element simulations: Locally reinforced stress construction. *Engineering Computations.* Vol 28 Num 1 Pages 76-92. (2011)

[8] Bickford, J.H. *Introduction to the Design and Behavior of Bolted Joints*, Fourth Edition, CRC Press, (2007), DOI:10.1201/9780849381874.fmatt.

[9] Makem, J. E., Armstrong, C. G., Robinson, T. T. "Automatic Decomposition and Efficient Semi-Structured Meshing of Complex Solids", Proceedings of the 20th International Meshing Roundtable, Paris, 199-215, (2011).

[10] Albert, A., Burkardt, N. , Ohmer, M. "Contact and Channel Model for Pairs of Working Surfaces", Advances in Design, H. A. ElMaraghy, W. H. ElMaraghy, Springer, London, 511—520, (2006).

[11] J. S. Gero, "Design prototypes: a knowledge representation schema for design", AI Magazine, 11 (4) 26–36, (1990).

[12] Roy, U.; Pramanik, N.; Sudarsan, R.; Sriram, R. D.; Lyons, K. W.: Function-to-form mapping: model, representation and applications in design synthesis, CAD, 33(10), 699-719, (2001).

[13] Roy, U., Bharadwaj, B.: Design with part behaviors: behavior model, representation and applications, CAD, 34(9), 613-636, (2002).

[14] Chouadria, R., Véron, P., "Identifying and re-meshing contact interfaces in a polyhedral assembly for digital mock-up", Eng. with Computers 22 (1), 47—58, (2006).

ADAPTIVE MODELLING AND MESHING FOR TIME DEPENDENT PROBLEMS BASED ON TIME AVERAGES

Malte Braack[*] **and Nico Taschenberger**[†]

[*]Mathematisches Seminar
Christian-Albrechts-Universität zu Kiel
Westring 383, 24118 Kiel, Germany
e-mail: braack@math.uni-kiel.de

[†]Mathematisches Seminar
Christian-Albrechts-Universität zu Kiel
Westring 383, 24118 Kiel, Germany
e-mail: taschenberger@math.uni-kiel.de

Key words: a posteriori estimator, adaptive meshing and modelling

Abstract. In this work we propose a dual weighted error estimator with respect to modelling and discretization error based on time-averages for evolutionary partial differential equations. This goal-oriented estimator measures the error of linear functionals averaged in time. It takes advantage of time averages and circumvents the solution of a nonstationary adjoint problem. We use the proposed estimator to solve convection-diffusion-reaction equations containing e.g. atmospheric chemistry models as reaction. This kind of equations are of major interest in meteorology.

1 Introduction

We present a goal-oriented duality based a posteriori error estimator and an adaptive strategy for the computation of functionals averaged in time for nonlinear time dependent problems. Error estimation with respect to a quantity of interest instead of the classical energy norm was presented in e.g. [8]. The concept of dual-weighted residual based (DWR) error estimation for the discretization error presented in [2] has entered to various fields, e.g. fluid dynamics [6] and optimization [1], [3]. Adaptvie modelling for free-surface flows was presented in [9], the concept of dual-weighted error estimation was extended to model error estimation e.g. in [5].

Functionals averaged in time are typically relevant for periodic or quasi-periodic solutions in time. Applications arise, e.g., in systems of convection-diffusion-reaction equations including a large amount of chemical reactions. In order to reduce the numerical complexity, we use simultaneously locally refined meshes and adaptive (chemical) models.

Such strategies come along with the question of how to control the discretization error and the model error.

These error parts are expressed in terms of output functionals. Hence, consideration of adjoint problems measuring the sensitivity of the functional output are needed. In contrast to the classical dual-weighted residual method we favor a fixed mesh and model strategy in time. Taking advantage of the (quasi-)periodic behaviour, only a stationary dual problem has to be solved. This implies that the computation of an evolutionary adjoint problem is circumvented. Storing the primal solution at every timestep is also not necessary. Only averaging in time is needed which is usually possible without serious problmes.

This a posteriori estimation technique is applied, e.g., to a system of convection-diffusion-reaction equations. The performance is checked by evaluating and comparing the estimated and exact errors for the mesh and the used model.

2 Variational problems and time-averages

Problem specification. We seek solutions $u \in W$ so that

$$(\partial_t u, \psi)_Q + \int_I B(u)(\psi)\,dt = (f, \psi)_Q \quad \forall \psi \in W, \tag{1}$$

for given data $f \in L^2(I, W')$ with

$$B(u)(\psi) := A(u)(\psi) + R(u)(\psi),$$

where A describes e.g. diffusion and convection and R describes a reaction model. $\Omega \subset \mathbb{R}^d$, $d \in \{2, 3\}$ is a Lipschitz domain and $I := [0, T]$ the time interval with $T > 0$. We consider the abstract variational problem in the Bochner space $W := H^1(I, V)$ with a Hilbert space V. Hence, functions in W are weakly differentiable with image in V.
By $(\cdot, \cdot)_Q$ we denote the L^2-scalar product in the time-space slab $Q := I \times \Omega$, $A : V \times V \to \mathbb{R}$ and $R : V \times V \to \mathbb{R}$ are semilinear forms supposed to be Frechét differentiable with respect to the first argument. The form R is the expensive part to solve, hence, a simplified model $R_m \approx R$ introducing less couplings is preferred.

Therefore, we seek an approximate solution $u_m \in W$ of the reduced system

$$(\partial_t u_m, \psi)_Q + \int_I B_m(u_m)(\psi)\,dt = (f, \psi)_Q \quad \forall \psi \in W, \tag{2}$$

with

$$B_m(u)(\psi) := A(u)(\psi) + R_m(u)(\psi). \tag{3}$$

By u_h and u_{hm} we denote the semidiscrete solutions corresponding to (1) and (2) in $H^1(I, V_h)$ with a conforming finite element space $V_h \subset V$, respectively.

Time-averages. We are interested in an accurate determination of a linear functional output

$$J : V \to \mathbb{R}$$

for a time averaged solution, e.g., in $J(\overline{u})$, where $\overline{u}$ denotes the time average

$$\overline{u} = \frac{1}{T} \int_0^T u \, dt.$$

The main goal of this work is the construction of an error estimator η in terms of this given functional:

$$\eta \approx J(\overline{u} - \overline{u}_{hm}).$$

Integration in time (by taking a test function constant in time) of equations (1) and (2) leads to the time averaged equations for $u, u_m \in H^1(I, W)$:

$$\sigma_T(u, \varphi) \quad + \overline{B}(u)(\varphi) \quad = (\overline{f}, \varphi) \quad \forall \varphi \in V, \tag{4}$$

$$\sigma_T(u_m, \varphi) + \overline{B}_m(u_m)(\varphi) = (\overline{f}, \varphi) \quad \forall \varphi \in V, \tag{5}$$

where we use the linear forms $\sigma : W \times V \to \mathbb{R}$ and the semilinear form $\overline{B} : W \times V \to \mathbb{R}$ defined by

$$\sigma_T(u, \varphi) := \frac{1}{T}(u(T) - u(0), \varphi),$$

$$\overline{B}(u)(\varphi) := \frac{1}{T} \int_0^T B(u)(\varphi) \, dt.$$

The time average of B_m, denoted by $\overline{B}_m$, is defined analogous to $\overline{B}$. The (time averaged) residual of the reduced problem (5) is denoted by

$$\varrho_m(u, \varphi) := (\overline{f}, \varphi) - \sigma_T(u, \varphi) - \overline{B}_m(u)(\varphi). \tag{6}$$

Using these prerequisites we can now formulate the estimator using stationary dual problems.

3   A posteriori estimation of discretization and model error

We will now present a dual-weighted error estimator. Using the time-averaged equations of the the primal problems (4) and (5) we formulate **stationary** dual problems as

$$z \in V : B'(\overline{\xi})(\varphi, z) \quad = J(\varphi) \quad \forall \varphi \in V, \tag{7}$$

$$z_m \in V : B'_m(\overline{\xi}_m)(\varphi, z_m) = J(\varphi) \quad \forall \varphi \in V. \tag{8}$$

If $B(\cdot)(\cdot)$ or $B_m(\cdot)(\cdot)$ are nonlinear in the first argument, $z = z(\xi)$ and $z_m = z_m(\xi_m)$ depend on the choices $\xi, \xi_m \in V$. The resulting error estimator is based on fluctuations in time due to possible nonlinearities of B and B_m defined by

$$
\begin{aligned}
K(u)(\varphi) &:= \overline{B}(u)(\varphi) - B(\overline{u})(\varphi), \\
K_m(u)(\varphi) &:= \overline{B}_m(u)(\varphi) - B_m(\overline{u})(\varphi),
\end{aligned}
$$

These nonlinearities in t result from nonlinearities in the reaction parts R and R_m and nonlinearities of A in u, e.g. due to nonstationary coefficients in A

Using the definitions of these dual problems we can formulate a dual weighted error estimator with respect to a linear functional $J : V \to \mathbb{R}$.

Theorem 3.1. *If B and B_m are continuously Gâteaux differentiable, the discretization and model error with respect to the linear functional J can be represented by*

$$
J(\overline{u} - \overline{u}_{hm}) = J(\overline{u} - \overline{u}_m) + J(\overline{u}_m - \overline{u}_{hm}), \tag{9}
$$

where

$$
\begin{aligned}
J(\overline{u} - \overline{u}_m) &= -\sigma_T(u - u_m, z) - K(u)(z) + K(u_m)(z) - \overline{R}(u_m)(z) + \overline{R}_m(u_m)(z), \\
J(\overline{u}_m - \overline{u}_{hm}) &= \overline{\varrho}_m(u_{hm})(z_m - i_h z_m) - \sigma_T(u_m - u_{hm}, z_m) \\
&\quad + K_m(u_{hm})(z_m) - K_m(u_m)(z_m),
\end{aligned}
$$

and $z = z(\overline{\xi}) \in V$, $z_m = z_m(\overline{\xi}_m) \in V$ are the dual solutions of (7) and (8), respectively, to the linearizations at $\overline{\xi} = \lambda \overline{u} + (1-\lambda)\overline{u}_m$ and $\overline{\xi}_m = \lambda_m \overline{u}_m + (1-\lambda_m)\overline{u}_{hm}$ with appropriate $\lambda, \lambda_m \in [0,1]$.

Proof. We split the proof into the derivation of the model error and the discretization error and start with the model error $J(\overline{u} - \overline{u}_m)$.

The mean value theorem ensures the existence of at least one $\lambda \in [0,1]$ so that for $\overline{\xi} := \lambda \overline{u} - (1-\lambda)\overline{u}_m \in V$ and $\overline{e}_m := \overline{u} - \overline{u}_m$ it holds

$$
\begin{aligned}
B'(\overline{\xi})(\overline{e}_m, \varphi) &= \int_0^1 B'(\overline{u}_m + s\overline{e}_m)(\overline{e}_m, \varphi)\, ds \\
&= B(\overline{u})(\varphi) - B(\overline{u}_m)(\varphi) \qquad \forall \varphi \in V.
\end{aligned}
$$

Let $z = z(\overline{\xi}) \in V$ be the associated dual solution of equation (7). Then it holds with (3), the time-averaged equations (4), (5) and the definition of $K(\cdot)(\cdot)$:

$$
\begin{aligned}
J(\overline{u} - \overline{u}_m) &= B'(\overline{\xi})(\overline{u} - \overline{u}_m, z) \\
&= B(\overline{u})(z) - B(\overline{u}_m)(z) \\
&= -\sigma_T(u - u_m, z) - K(u)(z) + K(u_m)(z) - \overline{R}(u_m)(z) + \overline{R}_m(u_m)(z).
\end{aligned}
$$

The discretization error $J(\overline{u}_m - \overline{u}_{hm})$ can be derived in a similar way: We use again the mean value theorem for $\overline{\xi}_m := \lambda \overline{u}_m - (1 - \lambda)\overline{u}_{hm} \in V$ and denote by $z_m = z_m(\overline{\xi}_m) \in V$ the associated dual solution of equation (8). By $i_h : V \to V_h$ we denote an arbitrary interpolation operator. Using the time averaged residual (6) it holds

$$
\begin{aligned}
J(\overline{u}_m - \overline{u}_{hm}) =& B'_m(\overline{\xi}_m)(\overline{u}_m - \overline{u}_{hm}, z_m) \\
=& B_m(\overline{u}_m)(z_m) - B_m(\overline{u}_{hm})(z_m) \\
=& (\bar{f}, z_m) - \sigma_T(u_m, z_m) - \overline{B}_m(u_{hm})(z_m) + K_m(u_{hm})(z_m) - K_m(u_m)(z_m) \\
=& \overline{\varrho}_m(u_{hm})(z_m - i_h z_m) - \sigma_T(u_m - u_{hm}, z_m) + K_m(u_{hm})(z_m) - K_m(u_m)(z_m).
\end{aligned}
$$

$\square$

4 Approximation of the estimator

Finite element approximation. We propose a discretization based on conforming finite elements for dimensions $d = 2$ or $d = 3$. According to this the mesh $\mathcal{T}_h$ of Ω consists of quadrilaterals or hexahedrals. By h_K we denote the diameter of a cell $K \in \mathcal{T}_h$ and by $Q_r(\mathcal{T}_h)$ the finite element space resulting from transformations $F_K : \hat{K} \to K$ of polynomials $\hat{\varphi}$ on a reference cell $\hat{K}$ of maximal degree $r \geq 0$ in each coordinate direction:

$$
Q_r(T_h) := \left\{ \varphi \in H^1(\Omega) : \varphi|_K \circ F_K \in \mathbb{Q}_r(\hat{K}) \quad \forall K \in \mathcal{T}_h \right\}.
$$

Hence, the finite element approximation of u_m is $u_{hm} \in V_h := Q_r(\mathcal{T}_h)^s$. The formulation for vector-valued problems is straight forward.

Assuming that $\mathcal{T}_h$ results from a globally coarser mesh $\mathcal{T}_{2h}$, we can define the higher order nodal interpolation operator to the coarser mesh $\mathcal{T}_{2h}$:

$$
i_{2h}^{(2)} : Q_r(\mathcal{T}_h) \to Q_{2r}(\mathcal{T}_{2h}). \tag{10}
$$

By $u_{2h}^{(2)}$ we denote the result of applying $i_{2h}^{(2)}$ to u_h.

In order to get an evaluable error estimator we have to approximate the exact formulation of theorem (3.1). More precisely we have to define approximations $\eta_h \approx J(\overline{u}_m - \overline{u}_{hm})$ and $\eta_m \approx J(\overline{u} - \overline{u}_m)$.

By z_h and z_{hm} we denote the solutions of the discrete problems corresponding to (7) and (8).

Definition of η_h. The numerically evaluable approximation η_h to $J(\overline{u}_m - \overline{u}_{hm})$ reads

$$
\begin{aligned}
\eta_h :=\ & \overline{\varrho}_m(u_{hm})(z_{2hm}^{(2)} - z_{hm}) - \sigma_T(u_{2hm}^{(2)} - u_{hm}, z_{hm}) \\
& + K_m(u_{hm})(z_{hm}) - K_m(u_{2hm}^{(2)})(z_{hm}).
\end{aligned}
$$

We cannot expect that $\sigma_T(u_{2hm}^{(2)} - u_{hm}, z_{hm})$ is a good approximation to $\sigma_T(u_m - u_{hm}, z_{hm})$, because $u_{2hm}^{(2)}$ at one point of time contains errors of u_{hm} accumulated over time. Neglecting this part is justified for long time averaging $(T >> 1)$ and/or periodic solutions.

To refine the mesh locally this global estimator has to be localized to get cell- or nodewise contributions to the error. Techniques to localize the different parts of the estimator can be found in [4].

Definition of η_m.   Neglecting $\sigma_T(\cdot, \cdot)$ again and by the definition of $K_{R_m}(\cdot)(\cdot)$ we propose the following approximation to $J(\overline{u} - \overline{u}_m)$:

$$\eta_m := -\overline{R}(u_{hm})(z_h) + \overline{R}_m(u_{hm})(z_h). \tag{11}$$

Solving the primal discrete problem we get u_{hm} and z_h, z_{hm} solving the discrete dual problems. To evaluate the pure model error $-K(u)(z) + K(u_m)(z)$ a solution to a model better than R_m is needed. For that the primal problem has to be solved again which is too costly. However, neglecting this part leads to pretty good estimates as we show in the numerical results.

Remark 4.1. *Computing the solutions z of problem (7) and z_m of problem (8), respectively z_h and z_{hm}, is not costly because the adjoint problems are linear and stationary. Nevertheless one may replace z_m by z to reduce numerical costs. A comparison of the two approaches can be found in the numerical examples of [7].*

5   Numerical Examples

In this section we present two numerical examples. In the first example we only measure the model error using 3 different models for atmospheric chemistry. In the second example we apply the estimator of theorem 3.1 to a simple test case and refine the mesh and adapt the model simultaneously.

We solve a system of coupled convection-diffusion-reaction equations

$$\partial_t \mathbf{u} - \nu \Delta \mathbf{u} + \beta \cdot \nabla \mathbf{u} - \mathbf{R}_m(u) = \mathbf{f},$$

with the viscosity ν, the flow field β and the reaction model R_m.

In order to switch locally between these models, we introduce a non-overlapping partitioning of Ω into subdomains Ω_i, $i = 1 \ldots M$,

$$\overline{\Omega} = \bigcup_{i=1}^{M} \overline{\Omega}_i.$$

Then in Ω_i the reduced model R^i is used and the reaction part R_m is defined by

$$R_m(u)(\varphi) := \sum_{i=1}^{M} R^i(u)(\varphi)_{|\Omega_i}.$$

To check the gentleness of the error estimator we introduce the efficiency index

$$I_{eff} = \frac{\eta}{J(\overline{u} - \overline{u}_{hm})},$$

which compares the estimated and the exact error. Hence, an efficiency index near to 1 implies that the error is estimated very well.

Model adaptivity for Atmospheric Chemistry Models. In this example we employ three different models. The used "exact" model R^1 is *RADM2*, a widely used model in atmospheric chemistry, see [11]. This atmospheric chemistry model contains 63 chemical species and 201 chemical reactions, where 5 species are *major gases*, whose concentrations are fixed. Therefore we have to solve a system of 58 coupled convection-diffusion-reaction equations. The *medium model R^2* consists of 32 reactions (see [10]) and a traditional *low model R^3* only of 3 reactions.

In this 3D example the computational domain in kilometres is $\Omega := (0, 20) \times (0, 20) \times (0, 1)$ and the computed time is one day. In order to keep the implementation as simple as possible temperature, pressure, turbulent viscosity and photolysis rates are fixed in space and time. The flow field is also fixed in space and time by $\beta := (-50, 100, 0)^T m/min$. The periodicity of the solution comes from a periodic source in the subdomain $\Omega_s := (13, 4) \times (16, 7) \times (0, 0.5)$ of NO (nitrogen oxide) and NO_2 (nitrogen dioxide). Starting with high emissions, they are reduced to zero at the middle of the day and increased again til the end of the day.

The goal functional is given by the mean value in time of O_3 (ozone) in the subdomain $\Omega_d := (13, 4) \times (16, 7) \times (0, 0.5)$:

$$J(\overline{u}) = \frac{1}{|\Omega_d|} \int_{\Omega_d} \overline{u}_{O_3} \, ds,$$

with the exact value

$$J(\overline{u}) \approx 3.144843e\text{-}02,$$

computed by using *RADM2* in the complete domain.

The adaption is chosen in a way, that the cells which sum up to 25% of the estimated error are adapted to R^1 and cells that sum up to the next 25% are adapted to the next better model; more precisely if the model of used at a cell is R^3 it is switched to R^2 otherwise from R^2 to R^1. Other adaption strategies can be found in [7].

The results can be found in table 1. We start with the cheap model R^3 in the complete domain, estimate the model error and adapt the model cellwise. As can be seen by the efficiency index I_{eff}, the error is estimated pretty good as I_{eff} is close to 1. The exact error decreases from step to step. Hence the localization of the estimated error works very well. The error of step 1 is reduced by a factor of nearly 100 by just using 20% of R^1 and 13% of R^2 at step 5. This shows that using the cheap model in the main part of the
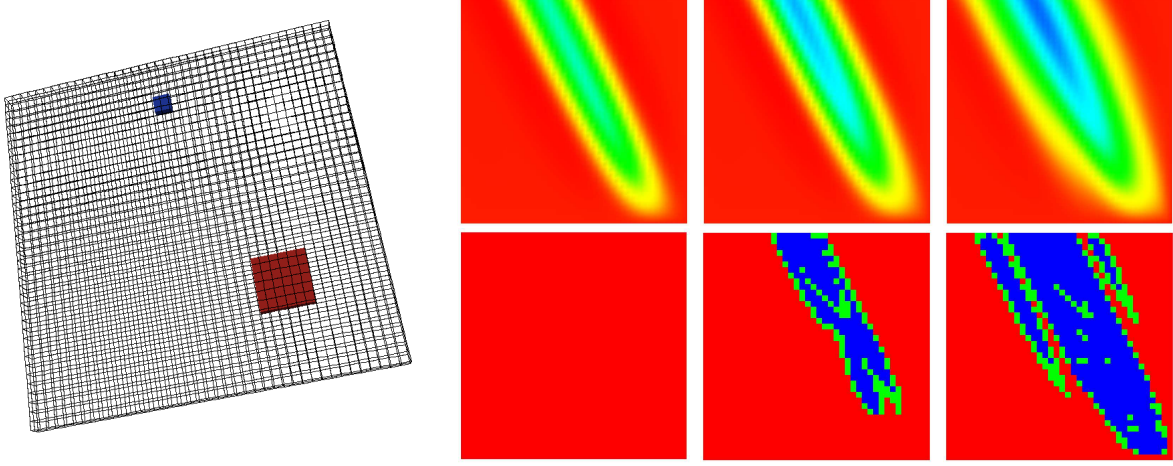
Figure 1: Left: Used mesh and area Ω_s of source term (red) and area Ω_d of goal functional (blue); Right: Mean solution O_3 at adaption steps 1,3 and 5 (above) and corresponding allocation of models R^1 (blue), R^2 (green) and R^3 (red) (below) of a cut at $250m$ height.

Table 1: Comparison of real and estimated model error

step	$\%R^1$	$\%R^2$	$\%R^3$	$J(\overline{u}_h - \overline{u}_{hm})$	η_m	I_{eff}
1	0	0	100	-5.68e-04	-6.04e-04	1.06
2	2	4	94	-1.65e-04	-1.79e-04	1.09
3	7	6	87	-9.29e-05	-1.02e-04	1.10
4	13	8	79	1.12e-05	8.61e-06	0.77
5	20	13	67	6.65e-06	5.86e-06	0.88

domain is absolutely sufficient. In figure 1 the allocation of the mean solution of ozone and the allocation of the models are pictured. As expected the model is adapted around Ω_s and Ω_d and along the route of transport according to the flow field β.

In figure 5 the development of the functional over time is shown. Although the estimator only measures time-averages, it can be seen that the functional output over time at step 5 is nearly the same as the output using *RADM2* in the complete domain.

Combined Mesh and Model Adaptivity In this paragraph we apply the complete estimator of theorem 3.1 to a test example and simultaneously refine the mesh locally and adapt the model cellwise. The estimated errors η_h and η_m are equilibrated, so that the mesh is not refined too much and/or the model is not adapted in too many cells.

We solve a system of three coupled convection-diffusion-reaction equations in the space-time slab $\Omega \times I := ((0, 40) \times (0, 10)) \times [0, 50]$ with the viscosity $\nu = 10$, homogeneous Dirichlet condition on the lower boundary, homogeneous Neumann conditions on the
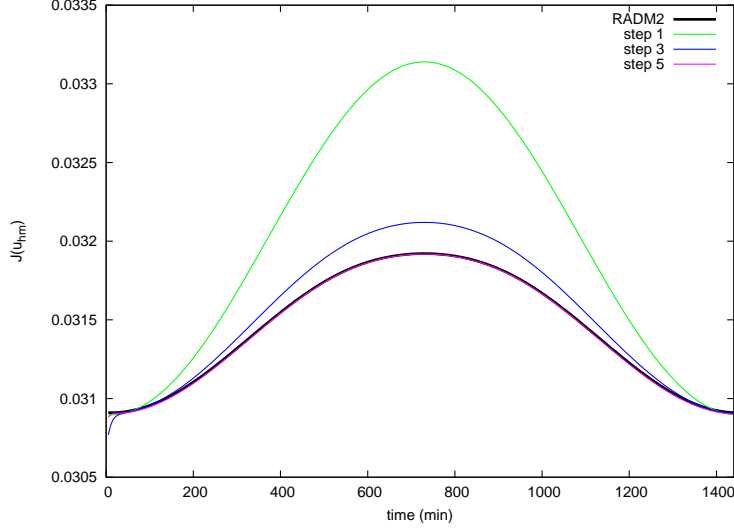
Figure 2: Comparison of computed ozone concentrations in *ppm* at adaption steps 1,3 and 5 to output of RADM2

remaining parts of $\partial\Omega$ and the periodic right-hand-side $\mathbf{f} = (f_1, f_2, 0)^T$

$$f_1(t, x) = \begin{cases} \psi(t) & \text{if } x \in \Omega_{s,1} \cup \Omega_{s,2}, \\ 0 & \text{else,} \end{cases}$$

$$f_2(t, x) = \begin{cases} \psi(t) & \text{if } x \in \Omega_{s,3}, \\ 0 & \text{else.} \end{cases}$$

with

$$\psi(t) := 1 + sin\left(\frac{3}{2}\pi + \frac{2\pi t}{10}\right),$$

and $\Omega_{s,1} := (25, 27.5) \times (0, 2.5)$, $\Omega_{s,2} := (32.5, 35) \times (5, 7.5)$, $\Omega_{s,3} := (30, 32.5) \times (2.5, 5)$. Thus the first component has sources in $\Omega_{s,1}$ and $\Omega_{s,2}$ and the second component a source in $\Omega_{s,3}$. We apply the reaction model $R^1(u) = (1, 1, -2)^T u_1 u_2$ and no reaction $R^2(u) = (0, 0, 0)^T$ and start with $R_m(u) = R^2(u)$.

The error of the mean solution of u_3 is measured with respect to the functional

$$J(\overline{u}) = \int_{\Omega_d} \overline{u}_3 \, ds,$$

with the subdomain $\Omega_d = (0, 10) \times (5, 10)$. We apply the time dependent flow field

$$\beta := (-10 - 30(1 + sin(3/2\pi + 4\pi t/10)), 10)^T,$$

so that the commutator terms $K(\cdot)(\cdot)$ do not vanish.

The exact value $J(\overline{u})$ was computed on a uniform mesh with approximately 4 million

step	#cells	$\%R_1$	$J(u - u_{hm})$	η_h	η_m	η	I_{eff}
1	4096	0	1.35e-02	0	2.01e-02	2.01e-02	1.49
2	4096	16	7.24e-03	-1.04e-05	9.87e-03	9.86e-03	1.36
3	9856	42	3.36e-03	-4.05e-06	4.88e-03	4.87e-03	1.45
4	9856	80	-6.65e-06	-6.98e-06	7.15e-07	-6.26e-06	0.94
5	28924	84	-1.68e-06	-2.11e-06	7.18e-07	-1.39e-06	0.83
6	77836	89	-7.86e-07	-9.47e-07	3.26e-07	-6.21e-07	0.79
7	186088	92	-3.31e-07	-4.11e-07	1.56e-07	-2.55e-07	0.77
8	461740	95	-1.56e-07	-1.76e-07	7.30e-08	-1.03e-07	0.66

Table 2: Development of estimated mesh and model error and comparison of estimated to real error
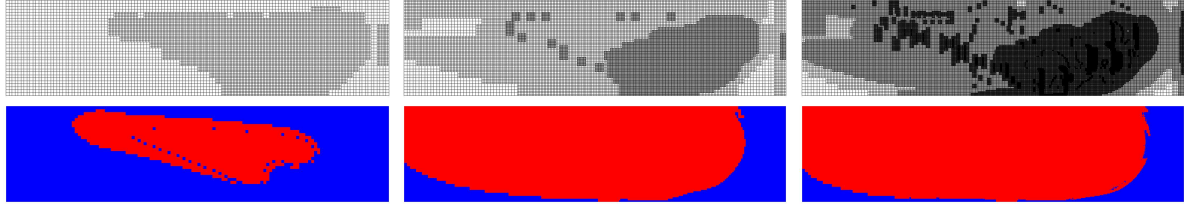
Figure 3: Local refined mesh and allocation of R_1 (red) and R_2 (blue) at steps 3, 5 and 7.

cells using Q_2 finite elements, the timestep $\Delta t = 0.0025$ and using the model $R^1(u)$ in the complete domain to

$$J(\overline{u}) \approx 1.350916847e\text{-}02.$$

In table 2 the results are presented. In the second column the number of cells of the local refined meshes are given. The mesh is not refined after the first step, because the discretization error is estimated to zero. The reason is that we start with $R_m = 0$ in the complete domain $\overline{u}_{hm,3}$ and $z_{h,1}, z_{h,2}$ are zero after the first step and from this η_h is zero. That the mesh is not refined after the third step is due to the equilibration of the error parts.

The overall estimation η gives a good approximation to the exact error as the efficiency index I_{eff} varies between 1.49 and 0.66, which shows pretty good behaviour of η. The estimated mesh and model errors decrease simultaneously. This shows that not only the estimation is reliable but rather the localization of the estimated terms lead to very good local estimates. A detailed inspection of the estimated parts compared to the estimation using higher order elements can be found in [7].

In figure 3 the local refined meshes and the model allocations are pictured. The mesh is mainly refined at the sources due to the righthandside f and the model is again adapted at the source, the area of the goal functional and the route of transport of the species. At step 5 the fine model is already used in nearly the complete domain. This is due to the problem setting, because the species are transported through large parts of the domain.

6   Conclusions

We presented a dual weighted a posteriori error estimator based on time-averages. The estimator circumvents the solution of a dual problem backward in time and uses a stationary dual problem instead. We can estimate the discretization and the model error separately and use these localized parts to equilibrate the errors. The numerical examples show a pretty good behaviour for the error estimation as well as for the used localizations.

REFERENCES

[1] R. Becker, M. Braack, D. Meidner, R. Rannacher and B. Vexler, *Adaptive finite element methods for PDE-constrained optimal control problems.* Reactive Flows, Diffusion and Transport, Springer, Berlin (2006), pp. 177–205.

[2] R. Becker and R. Rannacher, *An optimal control approach to a posteriori error estimation in finite element methods.* Acta Numer., (2001), **10**:1–102

[3] O. Bendix and B. Vexler, *A posteriori error estimation and adaptivity for elliptic optimal control problems with state controls.* Comput. Optimization and Appl., (2009), **44**:3–25.

[4] M. Braack, E. Burman and N. Taschenberger, *Duality based a posteriori error estimation for quasi periodic solutions using time averages.* SIAM J. Scientific Computing, (2011), **33**:2199–2216.

[5] M. Braack and A. Ern, *A posteriori control of modeling errors and discretization errors.* Multiscale Model. Simul., (2003), **1**:221–238.

[6] M. Braack and T. Richter, *Solutions of 3D Navier-Stokes benchmark problems with adaptive finite elements.* Computers and Fluids, (2006), **35**:372–392.

[7] M. Braack and N. Taschenberger, *A posteriori control of modelling and discretization errors for quasi periodic solutions.* (2012), submitted.

[8] J.T. Oden and S. Prudhomme, *Goal-oriented error estimation and adaptivity for the finite element method.* Computers and Mathematics with Applications, (2001), **41**:735-756.

[9] S. Perotto, *Adaptive modeling for free-surface flows.* M2AN Math. Model. Numer. Anal., (2006), **40**, no. 3, 469-499.

[10] D Poppe, *Umweltmeteorologie: Reaktionsmechanismus zur Bestimmung der Stickstoffdioxid-Konzentration.* VDI Richtlinie 3783 Blatt 19, in preparation.

[11] W.R. Stockwell, P. Middleton and J.S. Chang, *The Second Generation Regional Acid Deposition Model Chemical Mechanism for Regional Air Quality Modeling.* Journal of Geophysical Research, (1990), **95**:16343–16367.

VI International Conference on Adaptive Modeling and Simulation
ADMOS 2013
J. P. Moitinho de Almeida, P. Díez, C. Tiago and N. Parés (Eds)

STRUCTURAL OPTIMISATION AS A MOVING BOUNDARY PROBLEM USING LEVEL SET FUNCTIONS

CHRISTOPHER J. BRAMPTON[*] AND H. ALICIA KIM[*]

[*] Department of Mechanical Engineering
University of Bath
Bath, BA2 7AY, UK
e-mail: C.J.Brampton@bath.ac.uk, H.A.Kim@bath.ac.uk

Key words: Level set method, structural optimisation, topology optimisation, composites.

Abstract. This extended abstract presents topology optimisation which uses level set functions representing the moving boundaries. The level set function based approach to topology optimisation has gained much popularity in the recent years due to its numerical stabilities and clear boundary representation of the solution. One advantage of the level set representation is its inherent capability to handle topological changes such as merging and splitting boundaries. We have developed a stable hole nucleation algorithm which makes the level set formulation completely suitable for topology optimisation. We demonstrate that our level set topology optimisation, both in 2D and 3D, have good convergence properties and less dependency on the initial design. We apply this to typical structural optimisation problems as well as coupled aero-structural problems. As coupled multidisciplinary optimisation problems have multiple optima, we find that the solutions 3D level set topology optimisation produce can be quite different from the solutions from the previous element-based approaches and simplified 2D solutions, suggesting potential alternatives.

1   INTRODUCTION

The level set method is a boundary or an interface tracking method. It was first introduced by Osher and Sethian [1] and since then it has been applied a range of areas such as image processing and multiphase flows. In the field of structural optimisation, the level set method can be used to track the curves or surfaces that define structural features as they are optimised over iterations. This following sections describe the level set topology optimisation method with a stable hole creation algorithm which will be demonstrated using numerical examples.

2   TOPOLOGY OPTIMISATION

Topology optimisation is the most general form of structural optimisation; of all structural optimisation, topology optimisation finds an optimal solution that is least dependent on the initial design. A common approach to topology optimisation is to formulate the problem as a material distribution problem where the available design domain is discretised with finite elements. Optimisation then determines whether each element should or should not exist iteratively. This formulation makes the optimisation problem a large-scale binary problem

which is typically relaxed to a continuous problem with design variables bounded between 0 and 1. This enables a gradient-based optimiser to solve the problem efficiently however the solutions with design variables between 0 and 1 does not usually represent a physical and manufacturable structure as this means a structure with material properties continuous varying throughout the structure. Therefore, the solutions with non-0/1 variables are penalised. This approach has been applied to many disciplinary problems and demonstrated to work well but it is well known that a complete elimination of non-0/1 solutions can be difficult to obtain and the numerical procedure introduces various parameters to which the solutions and convergence can be highly sensitive.

An alternative approach to topology optimisation using the level set method was introduced relatively recently, [2]. Since then, there has been a flurry of activities maturing this approach. One attractive advantage is that the level set method obtains clear boundaries defining the general layout of the optimising structure at every iteration and eliminates the non-0/1 solutions completely. We will first outline our level set based topology optimisation method with a hole creation algorithm. The following sections will then show the example results to demonstrate that our method eliminates chequerboarding, a commonly known numerical instability in topology optimisation and reduced dependency on the initial solution [3]. The last example shows the application to a coupled multidisciplinary problem, aero-structural topology optimisation of an aircraft wing.

2.1 Level Set Topology Optimisation Method

The level set method defines the structural boundaries to be where the level set ϕ, is zero, (1).

$$
\begin{cases}
\phi(x) > 0, & x \in \Omega_S \\
\phi(x) = 0, & x \in \Gamma_S \\
\phi(x) < 0, & x \notin \Omega_S
\end{cases}
\tag{1}
$$

where Ω_S is the domain of the structure and Γ_S is the boundary of the structure. The compliance of the structure, $C(u, \phi)$ is minimized subject to an upper limit on structural volume:

$$
\text{Minimize}: C(u,\phi) = \int_\Omega E\varepsilon(u)\varepsilon(u)H(\phi)d\Omega
$$

$$
\text{Subject to}: \int_\Omega H(\phi)d\Omega \leq Vol^*
$$

$$
\int_\Omega E\varepsilon(u)\varepsilon(v)H(\phi)d\Omega = \int_\Omega bvH(\phi)d\Omega + \int_{\Gamma_S} fvd\Gamma_S
$$

$$
u|_{\Gamma_D} = 0 \quad \forall v \in U
\tag{2}
$$

where Ω is a domain larger than Ω_S such that $\Omega_S \subset \Omega$, Vol^* is the limit on material volume, E is the material property tensor, $\varepsilon(u)$ the strain tensor under displacement field u, U is the space

of kinematically permissible displacement fields, v is any permissible displacement field, b are body forces, f are surface tractions and $H(\phi)$ is the Heaviside function.

When applied to topology optimisation, the level set method incorporates shape sensitivity in computing the velocity function of a typical Hamilton-Jacobi equation, (3).

$$\phi_i^{k+1} = \phi_i^k - \Delta t \left| \nabla \phi_i^k \right| V_{n,i}$$

(3)

where $V_{n,i}$ is a discrete value of the velocity function acting normal to the boundary at point i, Δt is a discrete time step and k is the current iteration. In the case of (2), the velocity function also includes the Lagrangian multiplier, λ for the volume constraint, thus giving (4).

$$V_n = \lambda - E\varepsilon(u)\varepsilon(u)$$

(4)

While this *primary* level set function modifies, merges and splits existing boundaries, it is not possible to create a new boundary, i.e. a hole. We do this by introducing a *secondary* implicit level set function, $\bar{\phi}(x)$ [4]. It can be conceptually explained as the additional third dimension in the context of two-dimensional design domain, i.e. fictitious thickness. The secondary implicit level set function is initialized to an artificial thickness, $\bar{h}$, (5).

$$\bar{\phi}^0(x) = \begin{cases} +\bar{h}, & x \in \Omega_S \\ -\bar{h}, & x \notin \Omega_S \end{cases}$$

(5)

The secondary level set function is updated along with the primary level set function using (6).

$$\bar{\phi}_i^{k+1} = \bar{\phi}_i^k - \Delta t \bar{V}_{n,i}$$

(6)

A new hole is then created when $\bar{\phi}(x)$ becomes negative within the region of Ω_S and the new hole is added to the primary level set function by simply copying $\bar{\phi}(x)$ onto $\phi(x)$ within Ω_S. This inherent link between the primary and secondary level set functions forms a meaningful link between shape and topological optimisation, determining when and where to create a hole consistently.

2.2 A 2D Beam with Three Load Cases

We apply the level set topology optimisation with hole creation of Section 2.1 to a beam with three load cases shown in figure 1(a). Each load case has a magnitude of 2.0 and a weight of 1.0. The material properties are 1.0 and 0.3 for Young's modulus and Poisson's ratio, respectively. The beam is discretized using 200×50 unit sized square elements and the volume constraint is set to 40% of the design domain. Starting from the fully populated domain, the structure is optimised through figures 1(b)-(d), where figure 1(d) depicts the optimum solution coverged after 144 iterations with total compliance value of 4.67×10^2. The convergence history for this example is shown in figure 2. It is clear that the level set method

creates smooth and well-defined boundaries throughout optimisation and holes emerge as required. The hole creation does not cause a sudden discontinuity in the convergence history of figure 2, indicating that when and where the holes are created, merged and split are optimal. It is also noted that there are no chequerboarding and this numerical stability is consistent in our experience.
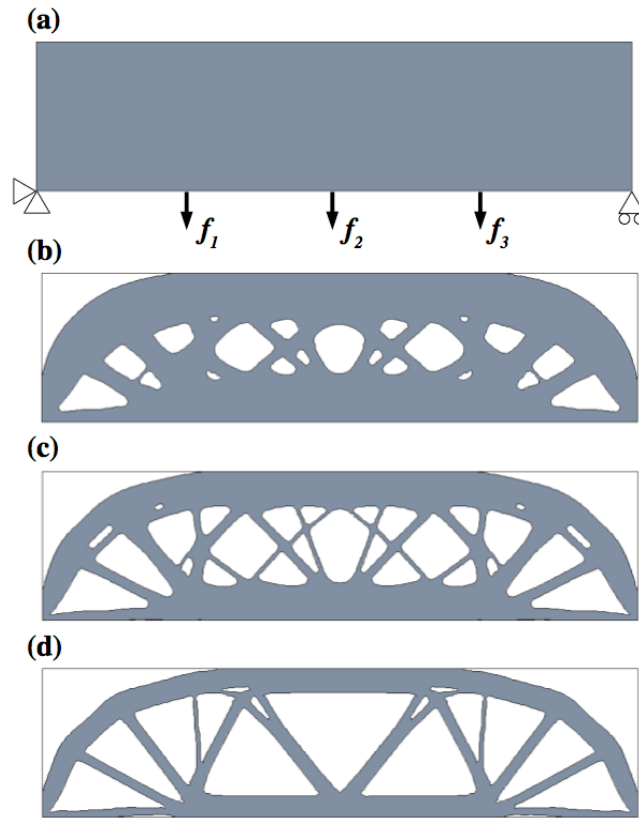
Figure 1: Beam optimisation for multiple load cases: (a) initial design; (b) 25 iterations; (c) 40 iterations; (d) solution after 144 iterations
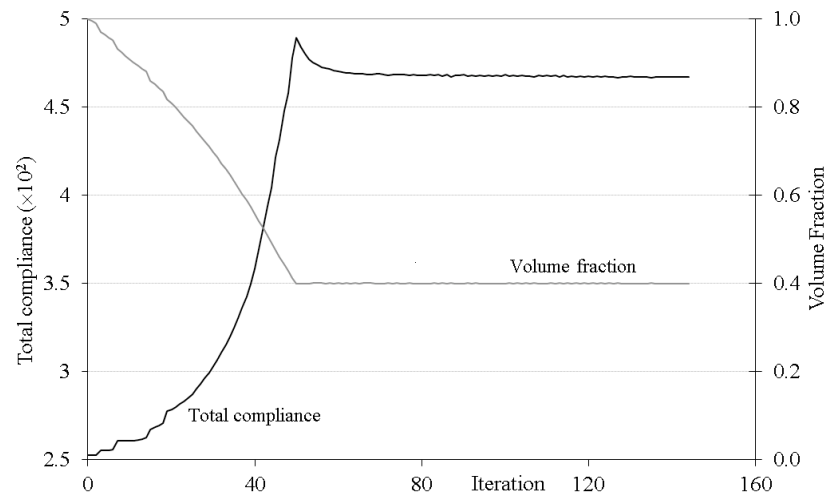
Figure 2: Convergence history of the beam optimisation

2.3 A 3D Cantilever Beam

This demonstrative example of a 3D cantilever beam is optimised for two load cases, one vertical and one horizontal loads at the centre. The other end is clamped. The beam is 45units long and the maximum cross-section is 20×20 unit^2. The volume constraint is set at 25% of the design domain.
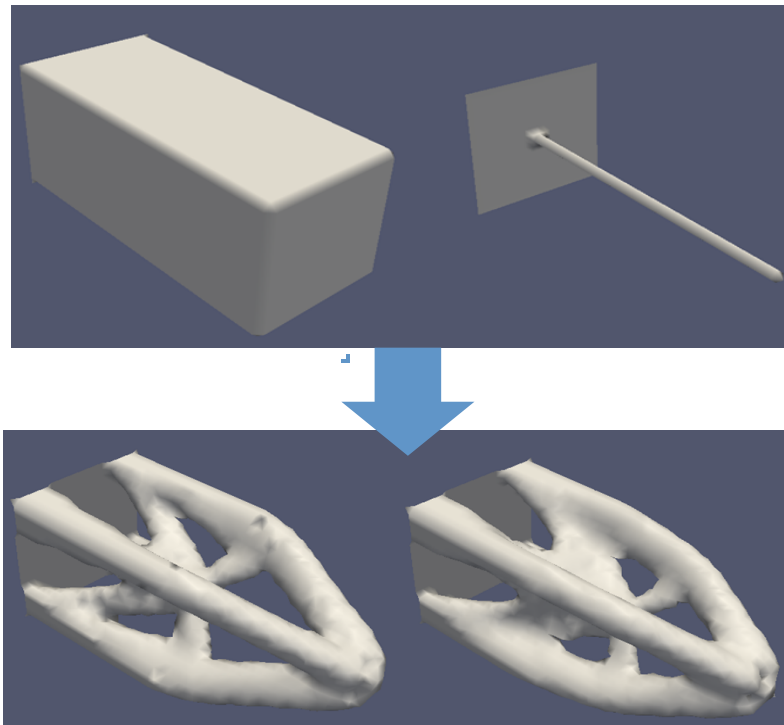
Figure 3: 3D canilever beam optimisation with two initial solutions.

We optimise this problem twice, the first time starting with the fully populated design domain and the second with the minimum structure linking the boundary and loading conditions by a thin beam, figure 3. We observe that the optimum solutions of the both runs agree favourably, with less than 0.5% difference in the compliance values between the two solutions. This shows that our level set method is robust and has reduced sensitivities to the starting solution.

2.4 Aero-Structural Wing Optimisation

We perform a preliminary study of 3D optimisation of the internal wing structure with full fluid-structure interaction used to update the aerodynamic loading during optimisation. The aerodynamic loading on the wing is calculated using the Double Lattice Method. The topology optimisation procedure is applied to the internal structure of a simple linearly tapered unswept wing box model with a $51\times20\times7$ regular fixed finite element mesh. The wing is clamped at the root under a cruise condition. The top and bottom skins are fixed and excluded from optimisation. The volume constraint is set at 35%.
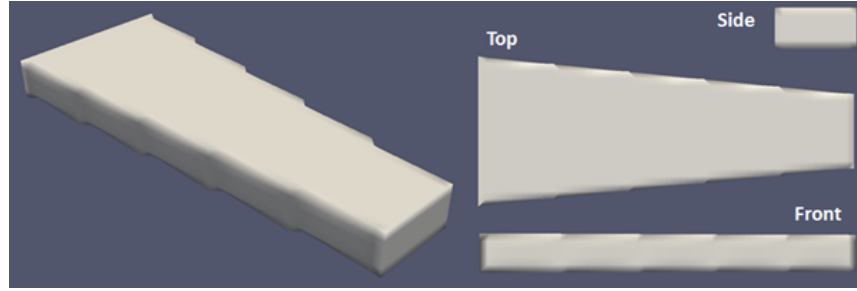
Figure 4: Geometry of a tapered unswept wing box model
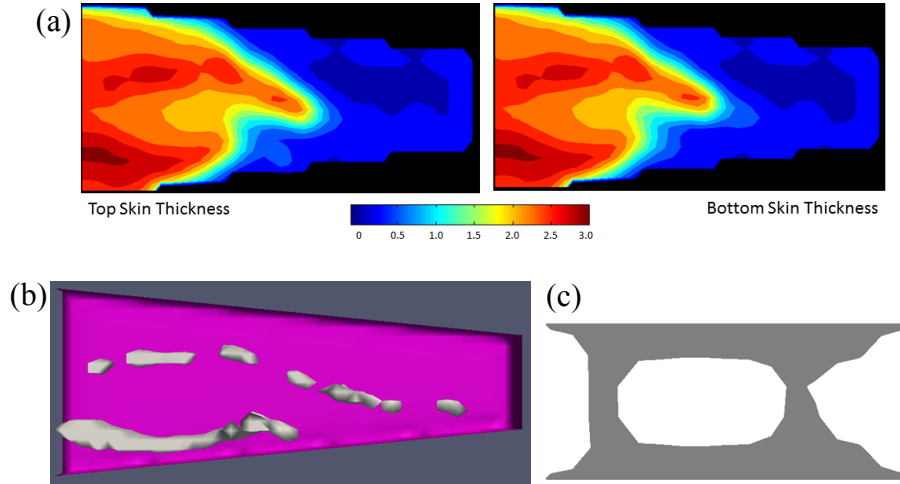
Figure 5: Optimum solution, (a) top and bottom skin thickness distribution; (b) internal column distribution; (c) cross-sectional view at span position 18

Figure 5 shows the optimum solution of the wing box. Looking at figure 5(a), the skin distribution along the top and bottom are nearly identical, with the root having the maximum thickness and gradually decreasing towards the tip. Perhaps what is the most distinctive about the optimum solution is the column like stiffeners connecting the top and the bottom skins, shown in figure 5(b). Near the root, the configuration looks somewhat reminiscent of two spar arrangements, reducing to a single "spar" arrangement towards the tip. These spar-like stiffeners are somewhat continuous near the room where the bending loads are the most significant, then they become discrete columns as the loads are reduced towards the tip. The other interesting feature to note is the skin thickness. It is significantly greatly than the typical skin thickness of the conventional wing configuration and figure 5(c) suggests a configuration similar to an I-beam. This is an intuitive characteristic as the wing is predominantly under bending during cruise. This preliminary optimisation result shows that there may be alternative configuration that may be more optimum than the conventional configuration and topology optimisation can be used to explore the potentially revolutionary optimum designs.

4 CONCLUSIONS

This extended abstract described the level set topology optimisation method and the new hole creation algorithm. Using this method, a few demonstrative examples are shown both in 2D and 3D: the chequerboarding is naturally eliminated and the method is not strongly dependent on the initial solution. Multidisciplinary topology optimisation was applied to a simple aircraft wing box under coupled aero-structural considerations. We see that the resulting structure is far from the conventional wing configuration suggesting that there is a potential for significant weight savings via revolutionary design changes. This warrants further studies.

REFERENCES

[1] Osher, S.J. and Sethian, J.A. Fronts propagating with curvature dependent speed: algorithms based on the hamilton-jacobi formulation. *J Comp Phys* (1988) **79**(1):12-49.

[2] Sethian, J.A. and Wiegmann, A. Structural boundary design via level set and immersed interface methods. *J Comp Phys* (2000) **163**(2):489-528

[3] Dunning, P.D. and Kim, H.A. Investigation and improvement of sensitivity computation using the area-fraction weighted fixed grid FEM and structural optimization. *Finite Elem Analysis Design* (2011) **47**:933-941

[4] Dunning, P.D. and Kim, H.A. A new method for creating holes in level-set function based topology optimisation. *Int J Num Meth Eng* (2013) **93**:118-134

[5] Gürdal, Z. and Olmedo, R. In-plane response of laminates with spatially varying fiber orientation: variable stiffness concept. *AIAA J* (1993) 31(4).

[6] Hyer, M.W. and Charette, R.F. The use of curvilinear fibre format in composite structure design. *AIAA J* (1991) **29**(6):1011-1015

[7] Ijsselmuiden, S.T., Abdalla, M.M., Setoodeh, S. and Gürdal, Z., Design of variable stiffness panels for maximum buckling load using lamination parameters," AIAA 2008-2123 (2008).

ROBUST ERROR ESTIMATES IN WEAK NORMS WITH APPLICATION TO IMPLICIT LARGE EDDY SIMULATION

Erik Burman[*]

[*]Department of Mathematics
University College London
Gower Street
London
WC1E 6BT

Key words: computability, stabilized FEM, Burgers' equation, passive transport, Navier-Stokes' equations, stability, a posteriori error estimates, a priori error estimates

Abstract. We discuss a posteriori and a priori error estimates of filtered quantities for solutions to some equations of fluid mechanics. For the computation of the solution we use low order finite element methods with either linear or nonlinear stabilization. The aim is to make the constants of the estimates independent of the Reynolds number, the Sobolev norm of the exact solution at time $t > 0$, or nonlinear effects such as shock formation. For the case of Burgers' equation this is possible. It follows that we obtain a complete assessment of the computability of the solution given the initial data. After a detailed description of the results in the case of the Burgers' equation we widen the scope and discuss transient convection–diffusion equations with rough data and the incompressible Navier-Stokes' equations in two space dimensions within the same paradigm.

1 INTRODUCTION

The task of designing adaptive finite element methods for flow problems reamains a challenging problem. A major bottleneck is the need to find a posteriori error estimators that are robust with respect to the Reynolds/Péclet number. In engineering practice a popular approach has been to use dual weighted residual type estimates in order to capture the stability properties of the problem at hand by solving a dual problem. This methodology however lacks theoretical underpinning, indicating when the approach is likely to work or to fail, in particular in the convection dominated regime. The aim of the present paper is to present some basic results showing that in the one dimensional case, or for special scale separated solutions in two space dimensions, robustness can be obtained for estimates of filtered quantities, provided a stabilized finite element method is used. We will here give an overview of recent results. For full proofs of the given results we refer to the recent publications [2, 3, 4].

For the discretization we use finite element methods with piecewise affine continuous approximation and linear, or nonlinear, artificial viscosity or higher order symmetric stabilization. These methods are strongly related to so called implicit large eddy simulation(ILES) methods where turbulent flows are approximated using the Navier-Stokes' equations and a discretization scheme augmented with some dissipative operator to guarantuee numerical stability, see [1].

In this framework we prove estimates for a regularized error. The interest of these estimates stems from the fact that the constant of the estimates are of moderate size and only depends on the regularity of the initial data in one space dimension, and in several dimensions the gradients of the large, energy carrying vortices. Hence there is no dependence on the Reynolds number, nor of the global regularity of the exact solution. The estimates also give a precise rate of convergence in the meshsize h, depending only on the filter width. This can be seen as a tentative theoretical explanation to the good performance of ILES methods for two dimensional flows in the absence of backscatter effects [7]. In this context our scale separation assumption (Assumption 1) acts as a sufficient condition to eliminate backscatter.

We will consider the following differential filter that sometimes is applied as a regularization in modified Navier-Stokes' systems for large eddy simulation,

$$-\delta^2 \Delta \tilde{u} + \tilde{u} = u(\cdot, T) \quad \text{on } \Omega \tag{1}$$

with $\tilde{u} = 0$ on $\partial\Omega$ and δ denoting the filter width. Let $\tilde{e} := \tilde{u} - \tilde{u}_h$, where $\tilde{u}_h$ denotes the regularized approximate solution. The a priori error estimates that we prove typically take the form

$$\|\tilde{e}(T)\|_\delta := \|\delta \nabla \tilde{e}(T)\|_\Omega + \|\tilde{e}(T)\|_\Omega \leq C(u_0, T) \exp\left(\frac{T}{\tau_F}\right) \beta^{\frac{1}{2}} \left(\frac{h}{\delta^2}\right)^{\frac{1}{2}} \tag{2}$$

where $\tilde{u}$ and $\tilde{u}_h$ are the filtered exact and computational solution respectively. The constant $C(u_0, T)$ in (2) depends only on the intial data, the mesh geometry and the final time and the coefficient β is an upper bound on the transport velocity. In some estimates length scales related to the $O(1)$ size of the domain have been omitted. The characteristic time τ_F depends on the velocity field in a nontrivial way and a key point in the below discussion is when τ_F can be expected to be $O(1)$ so that the exponential growth is moderate for moderate T. Note that the right hand side of (2) is independent of both the viscosity parameter and the Sobolev regularity of the exact solution. For previous work on error estimates for filtered solutions see [6], their estimates however are not robust in the Reynolds number.

The derivation of the estimate (2) uses:

– sharp energy stability estimates for the finite element method,

– L^∞-estimates for the finite element solution in the nonlinear cases,

– a priori stability estimates on a linearized dual problem with regularized data,

– Galerkin orthogonality and approximability.

To obtain precise control of all constants we must control the asymptotic growth of the residual and work with the exact dual adjoint, involving both the approximate and the exact solution in the nonlinear case. We will frequently use the notation $a \lesssim b$ defined by $a \leq Cb$ as well as $a \sim b$ meaning that $a \lesssim b$ and $b \lesssim a$ with C a constant independent of h, any essential physical parameters and of the exact solution. Some dependence on physical parameters may be included in the constants if it may be assumed not to change the magnitude of the constant.

2 THE BURGERS' EQUATION

Consider the simple model case of the Burgers' equation with periodic boundary conditions, on the space-time domain $Q := \Omega \times I$, with $\Omega := (0,1)$ and $I := (0,T)$ for some $T > 0$,

$$
\begin{array}{rcl}
\partial_t u + \frac{1}{2}\partial_x u^2 - \nu \partial_{xx} u & = & 0 \text{ in } Q \\
u(0,t) & = & u(1,t) \text{ for } t \in I \\
u(x,0) & = & u_0(x) \text{ for } x \in \Omega.
\end{array}
\tag{3}
$$

First we discuss the $L^\infty(I; L^2(\Omega))$ stability of the Burgers' equation and conclude that the resulting estimate includes an exponential factor of the type $\exp(\|\partial_x u\|_{L^\infty} T)$ reflecting a possible instability in the L^2-norm. Then we introduce the finite element discretization and briefly discuss the stability properties of the method. Finally we consider filtering of the final solution and show that the perturbation equation corresponding to the filtered solution has improved stability properties and the error may therefore be upper bounded independently of both the regularity of the exact solution and the physical viscosity. As we shall see, although $\|(u - u_h)(\cdot, T)\|_\Omega$, where u_h denotes the finite element approximation of (3), does not appear to allow for error estimates with moderate constants, the L^2-error of the filtered error, $\|(\tilde{u} - \tilde{u}_h)(\cdot, T)\|_\Omega$ does. Indeed, for the Burgers' equation in the high Reynolds number regime we prove the error estimate

$$
\||\tilde{u} - \tilde{u}_h\||_\delta \leq \tilde{C}(u_0, T) \exp(D_0 T) \left(\frac{h}{\delta^2} \right)^{\frac{1}{2}}
\tag{4}
$$

where $\tilde{u}$ and $\tilde{u}_h$ are the filtered exact and computational solution respectively and $D_0 \sim \sup_{x \in \Omega} |\partial_x u_0|$. We will also use the notation $U_0 \sim \sup_{x \in \Omega} |u_0(x)|$. For simplicity we assume $u_0 \in C^\infty(\Omega)$, this does not exclude the formation of sharp layers with gradients of order ν^{-1} at later times. For fixed filter width (4) results in a convergence rate of order $h^{\frac{1}{2}}$. If on the other hand the filter width is related to the mesh size $\delta \sim h^\alpha$ with $\alpha < \frac{1}{2}$ we get the convergence rate $h^{\frac{1-2\alpha}{2}}$. The parameter δ determines how strong the localization of the norm is. The choice $\delta = 1$ leads to a norm related to the H^{-1}-norm and the choice

$\delta = h$ leads to a norm similar to the L^2-norm. Clearly the estimates proposed here only makes sense for $0 \leq \alpha < \frac{1}{2}$. This indicates that no error bounds in a norm similar to the L^2 case can be obtained in this framework.

3 The Burgers' equation with viscous dissipation

The wellposedness of the equation (3) for $\nu \geq 0$ is well known it is also known that for $\nu > 0$ by parabolic regularization the solution is $C^\infty(\Omega)$. This high regularity however does not necessarily help us when approximating the solution, since we are interested in computations using a mesh-size that is much larger than the viscosity and still want the bounds to be Robust with respect to the Reynolds number.

3.1 L^2-stability of Burgers' equation

Consider a general perturbation $\eta(x)$ of the initial data of (3).

$$\begin{aligned}
\partial_t \hat{u} + \tfrac{1}{2}\partial_x \hat{u}^2 - \nu\partial_{xx}\hat{u} &= 0 \text{ in } Q \\
\hat{u}(0,t) &= \hat{u}(1,t) \text{ for } t \in I \\
\hat{u}(x,0) &= u_0(x) + \eta(x) \text{ for } x \in \Omega.
\end{aligned} \tag{5}$$

Taking the difference of (5) and (3) leads to the perturbation equation for $\hat{e} := \hat{u} - u$ with $a(u,\hat{u}) := \frac{1}{2}(u + \hat{u})$,

$$\begin{aligned}
\partial_t \hat{e} + \partial_x(a(u,\hat{u})\hat{e}) - \nu\partial_{xx}\hat{e} &= 0 \text{ in } Q, \\
\hat{e}(0,t) &= \hat{e}(1,t) \text{ for } t \in I \\
\hat{e}(x,0) &= \eta(x) \text{ for } x \in \Omega.
\end{aligned} \tag{6}$$

Multiplying equation (6) by $\hat{e}$ and integrating over Q leads to the energy equality

$$\frac{1}{2}\|\hat{e}(T)\|_\Omega^2 + \|\nu^{\frac{1}{2}}\partial_x\hat{e}\|_Q^2 = \frac{1}{2}\|\eta\|_\Omega^2 - \int_Q (\partial_x a(u,\hat{u}))\hat{e}^2.$$

We know that due to shock formation $-\partial_x a(u,\hat{u}) \sim \nu^{-1}$. Any attempt to obtain control of $\|\hat{e}(T)\|_\Omega^2$ in terms of the initial data will rely on Gronwall's lemma, leading to

$$\|\hat{e}(T)\|_\Omega^2 \leq C_a\|\eta\|_\Omega^2$$

with the exponential factor

$$C_a := \exp(\|\partial_x a(u,\hat{u})\|_{L^\infty(Q)}T) \sim \exp(T/\nu).$$

This estimate tells us that we have stability (and hence computability) only up to the formation of shocks. Using this type of argument in the analysis of the finite element method leads to error estimates useful only for solutions with moderate gradients.

3.2 Maximum principles for Burgers' equation

It is well known that the equation (3) satisfies a maximum principle on the form:

$$\sup_{(x,t)\in Q} |u(x,t)| \leq \sup_{x\in\Omega} |u_0(x)|. \tag{7}$$

For our purposes we also need some precise information on the derivative. Since the solution of (3) is smooth we may derive the equation in space to obtain the following equation for the space derivative $w := \partial_x u$:

$$\begin{aligned}
\partial_t w + u\partial_x w - \nu\partial_{xx}w &= -w^2 \text{ in } Q \\
w(0,t) &= w(1,t) \text{ for } t \in I \\
w(x,0) &= \partial_x u_0(x) \text{ for } x \in \Omega.
\end{aligned} \tag{8}$$

Assuming that w takes its maximum in some point $x \in I$ and noting that $\partial_x w(x) = 0$ and $\partial_{xx}w(x) < 0$ it follows that $\partial_t w < 0$ at the maximum and we deduce the bound:

$$\max_{(x,t)\in Q} \partial_x u \leq \max_{x\in\Omega} \partial_x u_0. \tag{9}$$

It follows by the smoothness of the initial data that the space derivative is bounded above for all times.

4 Artificial viscosity finite element method

Discretize the interval Ω with N elements and let the local mesh-size be defined by $h := 1/N$. We denote the computational nodes by $x_i := i\,h$, $i = 0,\ldots,N$, defining the elements $\Omega_j := [x_j, x_{j+1}]$, $j = 0,\ldots,N-1$. The finite element space is given by

$$V_h := \left\{ v_h \in H^1(\Omega) : v_h|_{\Omega_j} \in P_1(\Omega_j); u_h(0) = u_h(1) \right\}.$$

We define the standard L^2 inner product on $X \subset \Omega$ by $(v_h, w_h)_X := \int_X v_h w_h \,\mathrm{d}x$. The discrete form corresponding to mass-lumping reads $(v_h, w_h)_h := \sum_{i=0}^{N-1} v_h(x_i)w_h(x_i)h$. The associated norms are defined by $\|v\|_X := (v,v)_X^{\frac{1}{2}}$, for all $v \in L^2(X)$, if X coincides with Ω the subscript may be dropped, and $\|v_h\|_h := (v_h, v_h)_h^{\frac{1}{2}}$ for all $v_h \in V_h$. Note that, by norm equivalence on discrete spaces, for all $v_h \in V_h$ there holds $\|v_h\|_h \sim \|v_h\|$. Using the above notation the artificial viscosity finite element space semi-discretization of (3) writes, given $u_0 \in C^\infty(\Omega)$ find $u_h(t) \in V_h$ such that $(u_h(0), v_h)_\Omega = (u_0, v_h)_\Omega$ and

$$(\partial_t u_h, v_h)_h + \left(\partial_x \frac{u_h^2}{2}, v_h \right)_\Omega + (\hat{\nu}\partial_x u_h, \partial_x v_h)_\Omega = 0, \text{ for all } v_h \in V_h \text{ and } t > 0, \tag{10}$$

where we propose two different forms of $\hat{\nu}$:

1. linear artificial viscosity:

$$\hat{\nu} := \max(U_0 h/2, \nu); \tag{11}$$

2. nonlinear artificial viscosity:
   Let $0 \le \epsilon$ and

$$\nu_0(u_h)|_{\Omega_i} := \frac{1}{2}\|u_h\|_{L^\infty(\Omega_i)} \max_{x \in \{x_i, x_{i+1}\}} \frac{|[\![\partial_x u_h]\!]|_x|}{2\{|\partial_x u_h|\}|_x + \epsilon}, \tag{12}$$

where $[\![\partial_x u_h]\!]|_{x_i}$ denotes the jump of $\partial_x u_h$ over the node x_i and $\{|\partial_x u_h|\}|_{x_i}$ denotes the average of $|\partial_x u_h|$ over x_i. If $\epsilon = 0$ and $\{|\partial_x u_h|\}|_{x_i} = 0$ we replace the quotient $|[\![\partial_x u_h]\!]|_{x_i}|/\{|\partial_x u_h|\}|_{x_i}$ by zero.

Further let

$$\xi(u_h)|_{\Omega_i} := \begin{cases} 1 & \text{if } \partial_x u_h|_{\Omega_i} > 0, \ \partial_x u_h|_{\Omega_i} > \partial_x u_h|_{\Omega_{i+1}} > 0 \\ & \text{and } \partial_x u_h|_{\Omega_i} \ge \partial_x u_h|_{\Omega_{i-1}} > 0 \\ 0 & \text{otherwise} \end{cases}$$

$$\nu_1(u_h)|_{\Omega_i} := \xi(u_h)|_{\Omega_i} \max\left(\nu_0|_{\Omega_{i-1}} \frac{\partial_x u_h|_{\Omega_{i-1}}}{\partial_x u_h|_{I_i}}, \nu_0|_{\Omega_{i+1}} \frac{\partial_x u_h|_{\Omega_{i+1}}}{\partial_x u_h|_{I_i}}\right). \tag{13}$$

Finally define:

$$\hat{\nu}(u_h)|_{\Omega_i} := \max(\nu, h(\nu_0|_{\Omega_i} + \nu_1|_{\Omega_i})). \tag{14}$$

The rationale for the nonlinear viscosity is to add first order viscosity at local extrema of the solution u_h so that (7) holds also for the discrete solution and enough viscosity at positive extrema of $\partial_x u_h$, making (9) carry over to the discrete setting. Using the properties of the numerical viscosity we may prove the following discrete stability estimate.

The solution u_h of the formulation (10) with either the linear artificial viscosity given by (11) or the nonlinear one of (14) with $\epsilon = 0$, satisfies the upper bounds

$$\|u_h(T)\| + \|\hat{\nu}^{\frac{1}{2}}\partial_x u_h\|_Q \lesssim \|u_0\|, \quad \|\partial_t u_h\|_Q \lesssim (U_0 T^{\frac{1}{2}} h^{-\frac{1}{2}} + \nu^{\frac{1}{2}})\|\partial_x u_0\|. \tag{15}$$

4.1 Error estimates for the Burgers' equation

To derive error estimates in the norm $\|\|\cdot\|\|_\delta$ we introduce the linearized adjoint problem

$$\begin{aligned} -\partial_t \varphi + a(u, u_h)\partial_x \varphi - \nu \partial_{xx} \varphi &= 0 \text{ in } Q, \\ \varphi(0, t) &= \varphi(1, t) \text{ for } t \in I, \\ \varphi(x, T) &= \psi(x) \text{ for } x \in \Omega. \end{aligned} \tag{16}$$

The following stability estimate for (16) follows easily by standard energy methods since *both* the discrete and the continuous solutions satisfy maximum principles of the type (7) and (9),

$$\sup_{t \in (0,T)} \|\partial_x \varphi(\cdot, t)\|^2 + \nu\|\partial_{xx}\varphi\|_Q^2 \lesssim \exp(D_0 T)\|\partial_x \psi\|^2. \tag{17}$$

The rationale for the dual adjoint is the following derivation of a perturbation equation for the functional of the error $|(e(T), \psi)_\Omega|$, where $e(T) := u(T) - u_h(T)$.

$$|(e(T), \psi)_\Omega| = |(e(T), \psi)_\Omega + \int_0^T (e, -\partial_t \varphi + a(u, u_h)\partial_x \varphi - \nu\partial_{xx}\varphi)_\Omega \ dt|$$

$$= |(e(0), \varphi(0))_\Omega - \int_0^T (\partial_t u_h + u_h\partial_x u_h, \varphi)_\Omega \ dt - \int_0^T (\nu\partial_x u_h, \partial_x \varphi)_\Omega \ dt|. \quad (18)$$

This relation connects the error to the computational residual weighted with the solution to the adjoint problem and can lead both to a posteriori error estimates and to a priori error estimates, provided we have sufficient information on the stability properties of the numerical discretization methods and of the dual problem. Observing that

$$|\|\tilde{e}(T)\||_\delta^2 = (\delta\partial_x\tilde{e}(T), \partial_x\tilde{e}(T))_\Omega + (\tilde{e}(T), \tilde{e}(T))_\Omega = (e(T), \tilde{e}(T))_\Omega \quad (19)$$

we deduce that the choice $\psi = \tilde{e}(T)$ in (16) leads to an error representation for the filtered error. Using this error representation, Galerkin orthogonality and the stability of the dual solution (17) we may prove the following a posteriori error estimate. The associated a priori error estimate is a direct consequence of the a posteriori error bound, the maximum principles satisfied by the discrete solution and the bounds of (15).

Theorem 1 *Let u be the solution of* (3), *u_h be the solution of* (10). *Then the following a posteriori and a priori bounds hold:*

$$|\|\tilde{e}(T)\||_\delta \lesssim \exp(D_0 T)\left(\frac{h}{\delta^2}\right)^{\frac{1}{2}}\left(h^{\frac{1}{2}}\|(u - u_h)(0)\| + h^{\frac{1}{2}}\int_0^T \inf_{v_h \in V_h} \|v_h + u_h\partial_x u_h\| \ dt\right.$$

$$\left. + h^{\frac{3}{2}}\int_0^T \|\partial_x\partial_t u_h\| \ dt + \int_0^T \|\max(0, \hat{\nu} - \nu)^{\frac{1}{2}}\partial_x u_h\| \ dt + h\left(\int_0^T \nu\|[\![\partial_x u_h]\!]\|_N^2 \ dt\right)^{\frac{1}{2}}\right), \quad (20)$$

where $\|[\![\partial_x u_h]\!]\|_N := \left(\sum_{i=0}^{N-1}(\partial_x u_h(x_i)|_{\Omega_{i+1}} - \partial_x u_h(x_i)|_{\Omega_i})^2\right)^{\frac{1}{2}}$, *with Ω_N identified ith Ω_0 by periodicity.*

$$|\|\tilde{e}\||_\delta \quad \lesssim \quad \exp(D_0 T)\left(\frac{h}{\delta^2}\right)^{\frac{1}{2}}\left(\left(h^{\frac{1}{2}} + U_0^{\frac{1}{2}}\sqrt{T}\right)\|u_0\| + (TU_0 + h^{\frac{1}{2}}\nu^{\frac{1}{2}})\|\partial_x u_0\|\right). \quad (21)$$

5 EXTENSION TO FLOW IN HIGHER DIMENSION

In higher dimension the difficulty compared to the Burgers equation, is that the gradient tensor of the velocity can not be expected to have any sign, even when the flow is incompressible. If strong vortices or separation is present in the flow the diverging streamlines may cause exponential growth of perturbations with factor proportional to the maximum velocity gradient in the energy estimates. This reflects that two particles

that initally are close may be separated very quickly by the flow, hence giving rise to sensitivity to perturbations. Below we will discuss how the idea of estimating filtered quantities can be used for the derivation of robust error estimates, first for passive transport with rough data and then for the two-dimensional Navier-Stokes equation. A key assumption in the below argument is a large eddy hypothesis, stating that the velocity field allows for an a priori decomposition where the main energy is carried by large eddies with moderate gradients and that remaining component can have arbitrary oscillation, but energy comparable to the diffusive/viscous dissipation, as made precise in this assumption.

Assumption 1 *(Large eddy scale separation)*
Let $\boldsymbol{\beta} \in [W^{1,\infty}(\Omega)]^2$. Given $\mu \in \mathbb{R}^+$, assume that there exists a decomposition of the velocity field,

$$\boldsymbol{\beta} = \overline{\boldsymbol{\beta}} + \boldsymbol{\beta}',$$

where, for all t, $\|\overline{\boldsymbol{\beta}}\|_{W^{1,\infty}(\Omega)} \sim 1$ and $\|\boldsymbol{\beta}'\|^2_{L^{\infty}(\Omega)} \sim \mu$.

Under this assumption we may define a global timescale for the flow relating to both the coarse scale spatial variation and the fine scale amplitude,

$$\tau_F := \min(\|\overline{\boldsymbol{\beta}}\|^{-1}_{W^{1,\infty}(\Omega)}, \mu/\|\boldsymbol{\beta}'\|^2_{L^{\infty}(\Omega)}) \sim 1. \tag{22}$$

Of course for any given $\boldsymbol{\beta}$ and viscosity μ one can find the optimal decomposition $\overline{\boldsymbol{\beta}} + \boldsymbol{\beta}'$ that maximizes τ_F, which gives a measure of the computability of that particular flow problem. Essentially we assume that the velocity vectorfield can be decomposed in a coarse scale, responsible for transport, that is slowly varying in space and a fine scale, responsible for mixing, that has small amplitude but may have very strong spatial variation. Expressed in Péclet numbers this means that the coarse scale Péclet number may be arbitrarily high, whereas the fine scale Péclet number must be of order one.

The Assumption 1 may now be used to derive a posteriori and a priori error estimate that are robust in the multidimensional case. We will briefly review the cases of passive transport and two dimensional Navier-Stokes' below.

5.1 Transient convection–diffusion equations

The problem that we will consider takes the following form. Let Ω be an open polygonal/polyhedral subset of $\mathbb{R}^d$, with boundary $\partial\Omega$, $u_0, f \in L^2(\Omega)$ and let $\boldsymbol{\beta} \in [C_0(I; W^{1,\infty}(\Omega))]^d$, $\mu \in \mathbb{R}^+$, then formally we may write, for $t > 0$ find $u \in H^1_0(\Omega)$ such that $u(x,0) = u_0(x)$ in Ω and

$$\partial_t u + \boldsymbol{\beta} \cdot \nabla u - \mu\Delta u = f, \quad \text{in } \Omega. \tag{23}$$

For the boundary conditions let $u|_{\partial\Omega} = 0$ and assume that the velocity field satisfies non-penetration boundary conditions $\boldsymbol{\beta} \cdot n_{\partial\Omega}|_{\partial\Omega} = 0$. We also consider the associated dual

problem, for $t > 0$ find $\varphi \in H_0^1(\Omega)$ such that

$$
\begin{aligned}
-\partial_t \varphi - \boldsymbol{\beta} \cdot \nabla \varphi - \mu \Delta \varphi &= 0 \text{ in } \Omega \\
\varphi &= 0 \text{ on } \partial\Omega \\
\varphi(\cdot, T) &= \psi(\cdot) \text{ in } \Omega.
\end{aligned}
\tag{24}
$$

Using energy methods and the Assumption 1 we may prove the following stability estimate for the dual solution

$$
\sup_{t \in I} |||\varphi(\cdot, t)|||_\delta + T^{-1} \|\delta^{1/2} \nabla \varphi\|_Q + T^{-1} \|\delta^{1/2} \partial_t \varphi\|_Q + \|(\delta\mu)^{1/2} \Delta \varphi\|_Q \lesssim C_{\tau_F, T} |||\psi|||_\delta, \tag{25}
$$

with $C_{\tau_F, T} \sim e^{\left(\frac{T}{\tau_F}\right)}$, where τ_F is given by (22).

5.1.1 Finite element discretization

Let $\{\mathcal{T}_h\}_h$ be a family of nonoverlapping conforming, quasi uniform triangulations, $\mathcal{T}_h := \{K\}_h$ where the triangles K have diameter h_K and that is indexed by $h := \max h_K$. We let the set of interior faces $\{F\}_h$ of a triangulation $\mathcal{T}_h$ be denoted by $\mathcal{F}$.

We will consider a standard finite element space of piecewise affine, continuous functions $V_h := \{v_h \in H^1(\Omega) : v_h|_K \in P_1(K), \quad \forall K \in \mathcal{T}_h\}$, where $P_1(K)$ denotes the set of affine polynomials on K also let $V_h^0 := V_h \cap H_0^1(\Omega)$.

For $t > 0$ find $u_h \in V_h^0$ such that $u_h(x, 0) = \pi_h u_0(x)$ and

$$
(\partial_t u_h, v_h) + a(u_h, v_h) + s_h(u_h, v_h) = (f, v_h), \quad \forall v_h \in V_h^0, \tag{26}
$$

where $a(\cdot, \cdot)$ is defined by:

$$
a(u, v) := (\beta \cdot \nabla u, v) + (\mu \nabla u, \nabla v)
$$

and

$$
s_h(u_h, v_h) := \gamma \sum_{F \in \mathcal{F}} \left\langle h_F^2 \|\boldsymbol{\beta} \cdot n_F\|_{L^\infty(F)} [\![\nabla u_h \cdot n_F]\!], [\![\nabla v_h \cdot n_F]\!] \right\rangle_F. \tag{27}
$$

The finite element method (26) satisfies the estimate

$$
\sup_{t \in I} \|u_h(t)\|_\Omega + \|\mu \nabla u_h\|_Q + \left(\int_0^T s_h(u_h, u_h) \, dt\right)^{\frac{1}{2}} \lesssim \int_0^T \|f\|_\Omega \, dt + \|u_0\|_\Omega. \tag{28}
$$

Theorem 2 (*A posteriori error estimate*) *Let* $\tilde{e} := \tilde{u} - \tilde{u}_h$. *Then there holds*

$$
\begin{aligned}
|||\tilde{e}|||_\delta \lesssim C_{\tau_F, T} \left(\frac{h}{\delta^2}\right)^{1/2} \Bigg(&\int_I \inf_{v_h \in V_h} \|h^{1/2}(\boldsymbol{\beta} \cdot \nabla u_h - v_h)\|_\Omega \, dt \\
&+ \int_I \sum_{F \in \mathcal{F}} \left(\|\mu [\![\nabla u_h]\!]\|_F^2\right)^{1/2} dt \\
&+ \int_I s_h(u_h, u_h)^{\frac{1}{2}} dt + h^{1/2} \int_I \|f - \pi_h f\|_\Omega \, dt + h^{1/2} \|u_0 - \pi_h u_0\|_\Omega \Bigg),
\end{aligned}
\tag{29}
$$

where we recall that $C_{\tau_F,T} \sim e^{\left(\frac{T}{\tau_F}\right)}$.

Theorem 3 *(A priori error estimate) Assume that* $\frac{\|\overline{\boldsymbol{\beta}}\|_{L^\infty(Q)}h}{\mu} > 1$, *with* $\|\overline{\boldsymbol{\beta}}\|_{L^\infty(Q)} \sim 1$, *then there holds*

$$\||\tilde{e}|\|_\delta \lesssim C_{\tau_F,T} \left(\frac{h}{\delta^2}\right)^{1/2} (h^{1/2} + T^{\frac{1}{2}} \left(\int_0^T \|f\|_\Omega \, dt + \|u_0\|_\Omega\right). \tag{30}$$

The right hand side of (30) is independent of μ and Sobolev norms of the solution. It only depends on the L^2-norm of data, showing that even for cases with rough source terms and initial data, such as those encountered in environmental flows, this estimate holds.

Note that the stability of the dual problem holds regardless of the numerical method used. The stabilization in the numerical method allows us to control the first residual in the a posteriori error estimate, by using the discrete stability estimate (28). If no stabilization is present there is no control of the streamline derivative, making it impossible to obtain uniformity in μ. If the domain is convex so that elliptic regularity can be used one may prove an optimal estimate valid also in the low Reynolds number regime

5.2 The Navier-Stokes' equations in two space dimensions

We will consider the Navier-Stokes' equations written on vorticity-velocity form. Let Ω be the unit square and assume that the boundary conditions are periodic in both cartesian directions. The equations then writes, $\omega(x,0) = \omega_0(x)$ and

$$\begin{aligned}
\partial_t \omega + \nabla \cdot (u\omega) - \nu\Delta\omega &= 0, \text{ in } Q, \\
-\Delta\Psi &= \omega \text{ in } Q, \\
u &= \text{rot } \Psi \text{ in } Q.
\end{aligned} \tag{31}$$

Let $L_* := \{q \in L^2(\Omega); \int_\Omega q = 0\}$. The associated weak formulation takes the form for $t > 0$, find $(\omega, \Psi) \in H^1(\Omega) \times H^1(\Omega) \cap L_*(\Omega)$, with $\omega(x,0) = \omega_0(x)$ and such that for $t > 0$ and $\forall(v, \Phi) \in H^1(\Omega) \times H^1(\Omega) \cap L_*(\Omega)$,

$$\begin{aligned}
(\partial_t \omega, v) + (\nabla \cdot (u\omega), v) + (\nu\nabla\omega, \nabla v) &= 0, \\
(\nabla\Psi, \nabla\Phi) &= (\omega, \Phi), \\
u &= \text{rot } \Psi \text{ in } Q.
\end{aligned} \tag{32}$$

6 Finite element discretization

Define V_h to be the standard space of piecewise affine, continuous periodic functions. Let $V_* := V_h \cap L_*$. We consider continuous finite elements with equal-order to discretize in space the vorticity ω and the stream function Ψ. The discrete velocity is given by

$u_h|_K := \mathrm{rot}\ \Psi := \{\partial_y \Psi, -\partial_x \Psi\}$. Note that using this definition $\nabla \cdot u_h = 0$ in Ω, i.e. the discrete velocity is globally divergence free. For $t > 0$ find $\omega_h, \Psi_h \in V_h \times V_*$ such that

$$
\begin{aligned}
(\partial_t \omega_h, v_h)_M + (\nabla \cdot (u_h \omega_h), v_h) + (\nu \nabla \omega_h, \nabla v_h) + s(u_h; \omega_h, v_h) &= 0 \\
(\nabla \Psi_h, \nabla \Phi_h) - (\omega_h, \Phi_h) &= 0 \\
u_h - \mathrm{rot}\ \Psi_h &= 0, \quad \forall v_h, \Phi_h \in V_h \times V_*.
\end{aligned}
\tag{33}
$$

Here $s(\cdot; \cdot, \cdot)$ denotes a stabilization operator that is linear in its last argument and $(\partial_t \omega_h, v_h)_M$ denotes the bilinear form defining the mass matrix, this operator either coincides with $(\cdot, \cdot)_\Omega$ or is defined as the scalar product $(\cdot, \cdot)_\Omega$ approximated using nodal quadrature, i.e. so called mass lumping. We will assume the stabilization term satisfies the bounds

$$
\|h[\![u_h \cdot \nabla \omega_h]\!]\|_{\mathcal{F}} \lesssim s(u_h, \omega_h; \omega_h)^{\frac{1}{2}} \lesssim h^{\frac{1}{2}}(U_0 + \|u_h\|_{L^\infty(\Omega)})\|\nabla \omega_h\|,
$$

$$
s(u_h, \omega_h; v_h) \lesssim h^{\frac{1}{2}}(U_0^{\frac{1}{2}} + \|u_h\|_{L^\infty(\Omega)}^{\frac{1}{2}}) s(u_h, \omega_h; \omega_h)^{\frac{1}{2}} \|\nabla v_h\|.
$$

This typically holds for (27) or for standard linear artificial viscosity with coefficient (11). The dual adjoint problem associated to the perturbation equation of (32) and (33) takes the form

$$
\begin{aligned}
-\partial_t \varphi_1 - u \cdot \nabla \varphi_1 - \varphi_2 - \nu \Delta \varphi_1 &= 0 \text{ in } Q, \\
-\Delta \varphi_2 - \nabla \omega_h \cdot \mathrm{rot}\ \varphi_1 &= 0 \text{ in } Q, \\
\varphi_1(x, T) &= \psi(x) \text{ in } \Omega.
\end{aligned}
\tag{34}
$$

A key result for the present analysis is the following stability estimate for the dual adjoint solution.

Proposition 1 *Assume that the exact velocity u satisfy the Assumption 1 with $\mu = \nu$. Then there holds for the solution (φ_1, φ_2) of (34),*

$$
\sup_{t \in I} \|\nabla \varphi_1(\cdot, t)\| + \|\nu^{\frac{1}{2}} D^2 \varphi_1\|_Q \lesssim C_{\tau_F, T} \|\nabla \psi\|
\tag{35}
$$

$$
\int_I \|\nabla \varphi_2(\cdot, t)\|\ dt \leq C_{\tau_F, T} \int_I \|\omega_h\|_{L^\infty(\Omega)}\ dt\ \|\nabla \psi\|.
\tag{36}
$$

Using the dual problem with $\psi = \tilde{\omega} - \tilde{\omega}_h$ we may prove the following a posteriori estimate,

Theorem 4 *(A posteriori error estimates)*

$$
\begin{aligned}
\|\|\tilde{\omega} - \tilde{\omega}_h\|\|_\delta &\lesssim e^{\frac{T}{\tau_F}} \left(\frac{h}{\delta^2}\right)^{\frac{1}{2}} \Big(\|(\omega - \omega_h)(\cdot, 0)\| + \int_0^T \|h[\![u_h \cdot \nabla \omega_h]\!]\|_{\mathcal{F}}\ dt \\
&\quad + \int_0^T \|\nu^{\frac{1}{2}}[\![n_F \cdot \nabla \omega_h]\!]\|_{\mathcal{F}}\ dt + h^{\frac{1}{2}} \sup_{t \in I} \|\omega_h(\cdot, t)\| \int_0^T \|\omega_h(\cdot, t)\|_{L^\infty(\Omega)}\ dt \\
&\quad + \left(h^{\frac{3}{2}} \int_0^T \|\partial_t \nabla \omega_h\|\ dt\right)^* + \int_I s(u_h; \omega_h, \omega_h)^{\frac{1}{2}}\ dt \Big)
\end{aligned}
\tag{37}
$$

where the term marked with a $*$ is omitted if the consistent mass matrix is used. For the velocities we have the estimate

$$\|(u - u_h)(\cdot, T)\| \leq \left(\|h^{\frac{1}{2}} [\![n_F \cdot \nabla \Psi_h(\cdot, T)]\!] \|_{\mathcal{F}} + \| \| (\tilde{\omega} - \tilde{\omega}_h)(\cdot, T) \| \|_1 \right) \tag{38}$$

where $\| \| (\tilde{\omega} - \tilde{\omega}_h)(\cdot, T) \| \|_1$ may be a posteriori bounded by taking $\delta = 1$ in (37).

If we assume that $s_h(u_h, \omega_h, v_h)$ is strong enough so that $\|\omega_h\|_{L^\infty(Q)} \lesssim \|\omega_h(\cdot, 0)\|_{L^\infty(\Omega)}$ then Theorem 4 together with the stability properties of the finite element method leads to the following a priori error estimates, that are independent of the Reynolds number and Sobolev norms of the exact solution,

$$\| \| (\tilde{\omega} - \tilde{\omega}_h)(T) \| \|_\delta \lesssim e^{\frac{T}{\tau_F}} \left(\frac{h}{\delta^2} \right)^{\frac{1}{2}} \text{ and } \|(u - u_h)(\cdot, T)\| \lesssim e^{\frac{T}{\tau_F}} h^{\frac{1}{2}}.$$

This can be achieved for instance using a linear artificial viscosity, similar to (11), or nonlinear diffusion of shock-capturing type (see [5] for precise definitions) on meshes for which the Laplacian produces an M-matrix, as detailed in [4].

REFERENCES

[1] J. P. Boris. *On large eddy simulation using subgrid turbulence models comment 1.* In J. L. Lumley, editor, *Whither Turbulence? Turbulence at the Crossroads*, Lecture Notes in Physics, page 344353. Berlin Springer Verlag, 1990.

[2] E. Burman. *Computability of filtered quantities for the Burgers' equation.* arxiv:1111.1182. (2012).

[3] E. Burman. *Robust error estimates in weak norms for advection dominated transport problems with rough data.* arXiv:1303.1964 (2013).

[4] E. Burman. *Robust error estimates for stabilized finite element approximations of the two dimensional Navier-Stokes equations with application to implicit large eddy simulation.* arXiv:1303.1996 (2013).

[5] E. Burman and A. Ern. *Stabilized Galerkin approximation of convection-diffusion-reaction equations: discrete maximum principle and convergence. Math. Comp.,* 74(252):1637–1652 (electronic), 2005.

[6] A. Dunca and V. John. *Finite element error analysis of space averaged flow fields defined by a differential filter. Math. Models Methods Appl. Sci.,* 14(4):603–618, 2004.

[7] J. Kent, J. Thuburn, and N. Wood. *Assessing implicit large eddy simulation for two-dimensional flow. Quarterly Journal of the Royal Meteorological Society,* 138(663):365–376, 2012.

NON-CONFORMING FINITE ELEMENT METHODS FOR THE OBSTACLE PROBLEM

C. Carstensen[*,†] and K. Köhler[*]

[*]Institut für Mathematik, Humboldt-Universität zu Berlin, Unter den Linden 6, D-10099 Berlin, Germany

[†]Department of CSE, Yonsei University, Seoul, Korea

Key words: obstacle problem, a priori error estimate, a posteriori error estimate, efficiency

Abstract. In an obstacle problems with an affine obstacle, homogeneous Dirichlet boundary conditions, and standard regularity assumptions, the Crouzeix-Raviart non-conforming finite element method (FEM) allows for linear convergence as the maximal mesh-size approaches zero. The residual-based a posteriori error analysis leads to reliable and efficient control over the error with explicit constants. It involves the design of a new discrete Lagrange multiplier and allows for the computation of a guaranteed upper error bound. A novel energy control for non-conforming FEMs lead to a computable guaranteed lower bound for the minimal energy. The paper presents numerical experiments to investigate the theoretical results empirically and so to explore the possibilities of the non-conforming finite element method with respect to adaptive mesh refinement in practice.

1 INTRODUCTION

Given a bounded polygonal Lipschitz domain $\Omega \subset \mathbb{R}^2$ with boundary $\partial\Omega$, the energy product $a : H^1(\Omega) \times H^1(\Omega) \to \mathbb{R}$ on the Hilbert space $H^1(\Omega)$ reads

$$a(u, v) = \int_\Omega \nabla u \cdot \nabla v \, dx \quad \text{for all } u, v \in H^1(\Omega)$$

and induces the energy semi-norm $\|\|\cdot\|\| := a(\cdot, \cdot)^{1/2}$, which is a norm on the vector space $V := H_0^1(\Omega) := \{v \in H^1(\Omega) | \, v = 0 \text{ on } \partial\Omega\}$. Given some source term $f \in L^2(\Omega)$ set $F \in L^2(\Omega)^*$ by

$$F(v) := \int_\Omega f v \, dx \quad \text{for all } v \in L^2(\Omega).$$

The obstacle $\chi \in H^2(\Omega) \cap W^{1,\infty}(\Omega)$ satisfies $\chi \leq 0$ along $\partial\Omega$ in order to ensure that the closed and convex subset

$$K := \{v \in H_0^1(\Omega) \,|\, \chi \leq v \text{ a.e.}\} \quad \text{of } H_0^1(\Omega)$$

is non-empty. The well established weak formulation of the obstacle problem seeks $u \in K$ such that

$$F(v - u) \le a(u, v - u) \quad \text{for all } v \in K. \tag{1.1}$$

It is well known [KS80], that a unique weak solution u of (1.1) exists. The a priori convergence analysis of [Fal74] provides linear convergence of the error in the H^1 semi-norm $\||\bullet\||$ for $u \in H^2(\Omega)$ approximated by a P_1 conforming finite element method. The more recent analysis of [Wan03] for a non-conforming P_1 FEM requires $u \in W^{s,p}(\Omega)$ for some $2 < p$ and $2 < s < 2 + 1/p$.

The non-conforming finite element method seeks some approximation in the set K_{NC} where the obstacle condition is tested at the midpoints of the edges in a regular triangulation of the polygonal domain into triangles. Hence the term non-conforming refers to the fact that the discrete solution is not a Sobolev function as well as to the additional fact that the discrete solution u_{CR} does not satisfy the obstacle condition almost everywhere in the domain.

This paper announces some theoretical results which guarantee linear convergence for the error in the discrete energy norm for any weak solution u in $H^2(\Omega)$ which is in parallel analogy to the classical result [Fal74] for conforming FEMs. The adaptive mesh-refinement is based on some a posteriori analysis and the first reliable and efficient error estimators are introduced and tested in this paper; cf. [BC04], [Vee01], [CM10], and [Bra05] for conforming first-order methods. Three computational benchmarks are revisited to empirically verify the theoretical predictions. The aim is to provide numerical evidence for the guaranteed error control and for the superiority of adaptive over uniform mesh-refinements.

The rather technical proofs for the underlying theoretical statements utilise the medius analysis in that they combine arguments from the a priori and a posteriori error analysis and will appear elsewhere.

The remaining parts of this paper are organised as follows. Section 2 introduces the discretisation of the obstacle problem. Section 3 presents a new a priori error analysis under minimal regularity assumptions and an a posteriori error result. The paper concerns three computational benchmark examples in Section 4. The first example discusses a typical corner singularity on an L-shaped domain. The second concerns a smooth obstacle on a square domain and the third has a piecewise affine obstacle also on a square domain.

Throughout this paper, the standard notation for Lebesgue and Sobolev spaces and their norms $\|\bullet\|_{L^2(\Omega)}$, $\||\bullet\|| = \|\nabla \bullet\|_{L^2(\Omega)}$ and $\||\bullet\||_{NC} := \|\nabla_{NC} \bullet\|_{L^2(\Omega)}$ and their local variants are used. Moreover $A \lesssim B$ abbreviates $A \le CB$ for some generic constant C and $A \approx B$ abbreviates $A \lesssim B \lesssim A$.

2 Preliminaries

2.1 Discretisation

Let $\Omega \subset \mathbb{R}^2$ be a bounded polygonal Lipschitz domain partitioned in a shape-regular triangulation $\mathcal{T}$ into triangles with the set of edges $\mathcal{E}$ and interior edges $\mathcal{E}(\Omega)$. Any edge $E \in \mathcal{E}$ has length $|E|$, the midpoint $\text{mid}(E)$, the unit normal ν_E and the tangent τ_E; $\text{mid}(\mathcal{E}) := \{\text{mid}(E) | E \in \mathcal{E}\}$ denotes the set of all midpoints. The subdivision of each triangle $T \in \mathcal{T}$ into four congruent

sub-triangles by straight lines through the edges midpoints results in the red-refined triangulation $red(\mathcal{T})$. For any $k \in \mathbb{N}_0$, set

$$P_k(T) := \{v_k : T \to \mathbb{R} \mid v_k \text{ is a polynomial of degree} \leq k\},$$

$$P_k(\mathcal{T}) := \{v_k \in L^2(\Omega) \mid \forall T \in \mathcal{T}, \; v_k|_T \in P_k(T)\},$$

$$\mathrm{CR}^1(\mathcal{T}) := \{v_{\mathrm{CR}} \in P_1(T) \mid v_{\mathrm{CR}} \text{ continuous at } \mathrm{mid}(\mathcal{E})\},$$

$$\mathrm{CR}^1_0(\mathcal{T}) := \{v_{\mathrm{CR}} \in \mathrm{CR}^1(\mathcal{T}) \mid \forall E \in \mathcal{E}(\partial\Omega), \; v_{\mathrm{CR}}(\mathrm{mid}(E)) = 0\},$$

$$\mathrm{K}_{\mathrm{NC}} := \{v_{\mathrm{CR}} \in \mathrm{CR}^1_0(\mathcal{T}) \mid \forall E \in \mathcal{E}(\Omega), \; \fint_E \chi ds \leq v_{\mathrm{CR}}(\mathrm{mid}(E))\}.$$

The triangulation $\mathcal{T}$ is shape regular in the sense that any interior angle of any triangle is bounded from below by some universal positive constant ω_0 and all the generic constants hidden in the notation $\lesssim$ may depend on $\omega_0 > 0$. The triangulation $\mathcal{T}$ is regular in the sense that any two distinct triangles in $\mathcal{T}$ with non-empty intersection are either identical or share exactly one common node or one common edge. For any triangulation $\mathcal{T}$, define the (local) mesh-size $h_{\mathcal{T}} \in P_0(\mathcal{T})$ and L^2-projection $\Pi_0 : L^2(\Omega) \to P_0(\Omega)$ by $h_{\mathcal{T}}|_T := h_T := diam(T)$ and $\Pi_0|_T f := \fint_T f dx$ for all $T \in \mathcal{T}$ and $f \in L^2(\Omega)$, with the integral mean $\fint_T \bullet \, dx := \int_T \bullet \, dx/|T|$. With the piecewise gradient $\nabla_{\mathrm{NC}} v_{\mathrm{CR}}$ of any discrete function $v_{\mathrm{CR}} \in \mathrm{CR}^1(\mathcal{T})$, the discrete energy product $a_{\mathrm{NC}} : \mathrm{CR}^1(\mathcal{T}) \times \mathrm{CR}^1(\mathcal{T}) \to \mathbb{R}$ reads

$$a_{\mathrm{NC}}(u_{\mathrm{CR}}, v_{\mathrm{CR}}) := \int_\Omega \nabla_{\mathrm{NC}} u_{\mathrm{CR}} \cdot \nabla_{\mathrm{NC}} v_{\mathrm{CR}} dx \quad \text{for all } u_{\mathrm{CR}}, v_{\mathrm{CR}} \in \mathrm{CR}^1(\mathcal{T})$$

and induces the discrete energy semi-norm $\||\cdot|\|_{\mathrm{NC}} := a_{\mathrm{NC}}(\cdot, \cdot)^{1/2}$ in $\mathrm{CR}^1(\mathcal{T})$. Owing to the discrete Friedrichs inequality $\|v_{\mathrm{CR}}\|_{L^2(\Omega)} \lesssim \||v_{\mathrm{CR}}|\|_{\mathrm{NC}}$ for all $v_{\mathrm{CR}} \in \mathrm{CR}^1(\mathcal{T})$ (cf. [BS08]) this is a norm in $\mathrm{CR}^1_0(\mathcal{T})$.

The discrete analogue to the variational inequality (1.1) seeks $u_{\mathrm{CR}} \in \mathrm{K}_{\mathrm{NC}}$ with

$$F(v_{\mathrm{CR}} - u_{\mathrm{CR}}) \leq a_{\mathrm{NC}}(u_{\mathrm{CR}}, v_{\mathrm{CR}} - u_{\mathrm{CR}}) \quad \text{for all } v_{\mathrm{CR}} \in \mathrm{K}_{\mathrm{NC}}. \tag{2.1}$$

The abstract results on variational inequalities in the Hilbert space $\left(\mathrm{CR}^1(\mathcal{T}), a_{\mathrm{NC}}\right)$ guarantee the unique existence of a discrete solution u_{CR}. Each edge $E \in \mathcal{E}(\Omega)$ is associated with its edge-oriented basis function $\psi_E \in \mathrm{CR}^1(\mathcal{T})$ such that $\psi_E \equiv 1$ along E while $\psi_E(\mathrm{mid}(F)) = 0$ for any other edge $F \in \mathcal{E}\backslash\{E\}$, and its support $\overline{\omega_E} := \cup\{T \in \mathcal{T} \mid E \in \mathcal{E}(T)\}$. For each edge $E \in \mathcal{E}(\Omega)$, the solution u_{CR} to the discrete variational inequality (2.1) satisfies the discrete consistency condition

$$0 \leq u_{\mathrm{CR}}(\mathrm{mid}(E)) - \fint_E \chi ds \perp F(\psi_E) - a_{\mathrm{NC}}(u_{\mathrm{CR}}, \psi_E) \leq 0. \tag{2.2}$$

This follows from direct considerations with the degrees of freedom in (2.1) and is the discrete analogue of the well known (continuous) consistency condition [KS80] for $u \in H^2_{loc}(\Omega)$ which satisfies

$$0 \leq u - \chi \perp f + \Delta u \leq 0 \quad \text{almost everywhere in } \Omega. \tag{2.3}$$

3 Error Analysis

This section provides an a priori and a posteriori error estimate for the error $\||u - u_{CR}\||_{NC}$ for the solutions u and u_{CR} of the continuous and discrete obstacle problem (1.1) and (2.1) as well as lower bounds of the minimal energy $E(u)$ based on the discrete energy $E_{NC}(u_{CR})$.

Theorem 3.1 (a priori error estimate) *The continuous and discrete solutions $u \in K$ and $u_{CR} \in K_{NC}$ to the obstacle problem with $u \in H^2(\Omega)$ satisfy*

$$\||u - u_{CR}\||_{NC} \lesssim \|h_{\mathcal{T}} f\|_{L^2(\Omega)} + \left\|h_{\mathcal{T}} D^2 u\right\|_{L^2(\Omega)}$$
$$+ \|\chi - I_{NC}\chi\|_{L^\infty(\Omega)} + \|h_{\mathcal{T}} \nabla(\chi - I_{NC}\chi)\|_{L^\infty(T)}. \qquad \square$$

Given the discrete Crouzeix-Raviart solution $u_{CR} \in K_{NC}$, define some function

$$\lambda_{CR} := \sum_{E \in \mathcal{E}(\Omega)} \rho_E \frac{\psi_E}{\|\psi_E\|^2_{L^2(\Omega)}} \text{ with } \rho_E := F(\psi_E) - a_{NC}(u_{CR}, \psi_E) \qquad (3.1)$$

for the edge-oriented basis function $\psi_E \in CR^1(\mathcal{T})$ associated to the edge $E \in \mathcal{E}(\Omega)$. It holds

$$\Lambda_{CR}(v_{CR}) = \int_\Omega \lambda_{CR} v_{CR} dx \text{ for all } v_{CR} \in CR_0^1(\mathcal{T}).$$

In the sequel, $\Lambda_{CR}(v)$ always denotes the L^2 scalar product of any Lebesgue function $v \in L^2(\Omega)$ with the above $\lambda_{CR} \in CR_0^1(\mathcal{T})$. The following a posteriori error estimate involves the continuous Lagrange multiplier

$$\Lambda := F - a(u, \bullet) \in V^*$$

with the L^2 representation $\lambda = f + \Delta u$. Define $\||\Lambda - \Lambda_{CR}\||_*$ by

$$\||\Lambda - \Lambda_{CR}\||_* := \sup_{v \in V \setminus \{0\}} \int_\Omega (\lambda - \lambda_{CR})(v) dx / \||v\||.$$

Theorems 3.2-3.3 utilise the subset $\mathcal{T}' := \left\{ T \in \mathcal{T} \,\middle|\, 0 < |\{x \in T | \lambda_{CR}(x) > 0\}| \right\}$ of $\mathcal{T}$ with the 2D Lebesgue measure $|\bullet|$ and the oscillations of a function g given by

$$\text{osc}(g, \mathcal{T}) := \sqrt{\sum_{T \in \mathcal{T}} h_T^2 \|g - \Pi_0 g\|^2_{L^2(T)}}.$$

Theorem 3.2 (guaranteed upper error bound) *Any $v \in K$ satisfies*

$$ⓐ \; 1/2\||u - u_{CR}\||^2_{NC} + \Lambda(u - v) + \int_{\mathcal{T}'} (\chi - u)\Pi_0 \lambda_{CR} dx + \int_{\mathcal{T} \setminus \mathcal{T}'} (\chi - u)\lambda_{CR} dx$$
$$\leq 1/2 \left(\kappa_{CR} \|h_{\mathcal{T}}(f - \lambda_{CR})\|_{L^2(\Omega)} + \text{osc}(\lambda_{CR}, \mathcal{T}')/j_{1,1} \right)^2 + 1/2\||v - u_{CR}\||^2_{NC}$$
$$+ \int_{\mathcal{T}'} (\chi - v)\Pi_0 \lambda_{CR} dx + \int_{\mathcal{T} \setminus \mathcal{T}'} (u_{CR} - v)\lambda_{CR} dx;$$
$$ⓑ \; \||\Lambda - \Lambda_{CR}\||_* \leq \||u - u_{CR}\||_{NC} + \text{osc}(f - \lambda_{CR}, \mathcal{T})/j_{1,1}$$
$$+ 1/2 \|\Pi_0(f - \lambda_{CR})(\bullet - \text{mid}(\mathcal{T}))\|_{L^2(\Omega)} + \||u_{CR} - v\||_{NC}. \qquad \square$$

The universal constant $\kappa_{CR} \leq 0.298217419$ is derived from an interpolation error estimate for the non-conforming interpolant I_{NC} as in [CGR12]. The lower bound for the exact energy $E(u)$ is given in the following theorem.

Theorem 3.3 (lower bound for the minimal energy) *The discrete solution u_{CR} and the continuous solution u to the obstacle problem satisfy*

$$\textcircled{a}\ E_{NC}(u_{CR}) - \frac{\kappa_{CR}^2}{2} \|h_{\mathcal{T}} f\|_{L^2(\Omega)}^2 \leq E(u);$$

$$\textcircled{b}\ E_{NC}(u_{CR}) - \left(\kappa_{CR} \|h_{\mathcal{T}}(f - \lambda_{CR})\|_{L^2(\Omega)} + \mathrm{osc}(\lambda_{CR}, \mathcal{T}')\right)^2 / 2$$
$$- \int_{\mathcal{T}'} (\chi - u_{CR})\Pi_0 \lambda_{CR} dx + \int_{\mathcal{T} \setminus \mathcal{T}'} (I_{NC}\chi - \chi)\lambda_{CR} dx \leq E(u). \qquad \square$$

For any $v \in K$, the a posteriori error estimate of Theorem 3.2 leads to a computable global upper bound $GUB(v)$ of the five non-negative error terms in $LHS(v)$

$$LHS(v) := \|u - u_{CR}\|_{NC} + \Lambda(u - v)^{1/2} + \left(\int_{\mathcal{T}'} \lambda_{CR}\Pi_0(\chi - u)dx\right)^{1/2}$$
$$+ \left(\int_{\mathcal{T} \setminus \mathcal{T}'} (I_{NC}\chi - u)\lambda_{CR} dx\right)^{1/2} + \|\Lambda - \Lambda_{CR}\|_* \lesssim GUB(v)$$
$$GUB(v) := \|h_{\mathcal{T}}(f - \lambda_{CR})\|_{L^2(\Omega)} + \mathrm{osc}(\lambda_{CR}, \mathcal{T}') + \|v - u_{CR}\|_{NC}$$
$$+ \left(\int_{\mathcal{T}'} \lambda_{CR}\Pi_0(\chi - v)dx\right)^{1/2} + \left(\int_{\mathcal{T} \setminus \mathcal{T}'} \lambda_{CR}(u_{CR} - v)dx\right)^{1/2}$$
$$+ \mathrm{osc}(f - \lambda_{CR}, \mathcal{T}) + \|\Pi_0(f - \lambda_{CR})(\bullet - \mathrm{mid}(\mathcal{T}))\|_{L^2(\Omega)}.$$

This reliable error bound is efficient in the sense that the converse inequality holds up to some generic factor hidden in the notation $\lesssim$ and up to data oscillations.

Theorem 3.4 (efficiency) *Any function $v \in K$ with $\|u - v\| \lesssim \|u - u_{CR}\|_{NC}$ satisfies*

$$GUB(v) \lesssim LHS(v) + \mathrm{osc}(f, \mathcal{T}) + \mathrm{osc}(\lambda, \mathcal{T}). \qquad \square$$

4 Computational Benchmarks

This section is devoted to the presentation of a novel adaptive mesh-refinement algorithm and the empirical investigation of the superiority of adaptive over uniform meshes, the computational comparison of conforming and non-conforming first-order FEMs and the verification of the guaranteed error and energy bounds in practice.

4.1 Numerical Realisation

Adaptive Algorithm. INPUT is a coarse mesh $\mathcal{T}_0$, and a parameter $0 < \theta \leq 1$.
LOOP For level $\ell = 0, 1, 2, \ldots$ until termination do

COMPUTE the discrete solution u_{CR} on $\mathcal{T}_\ell$ with `ndof` many unknowns with code similar to [ACF].

ESTIMATE the error $\||u - u_{CR}\||^2_{NC}$ with any of the estimators $\eta_{P1red}, \eta_{Energy}, \eta_{J2}$ defined below. Theorem 3.2 leads to an estimator

$$\eta_v^2(T) := \left(\kappa_{CR} \|h_{\mathcal{T}}(f - \lambda_{CR})\|_{L^2(T)} + Osc(\lambda_{CR}, T)/j_{1,1}\right)^2 + \||v - u_{CR}\||^2_{NC(T)} + 2I(T)$$

where

$$Osc(\lambda_{CR}, T) := \begin{cases} osc(\lambda_{CR}, T) \text{ for } T \in \mathcal{T}', \\ 0 \text{ for } T \in \mathcal{T} \setminus \mathcal{T}', \end{cases} \qquad I(T) := \begin{cases} \int_{T'} (\chi - v)\Pi_0\lambda_{CR}dx \text{ for } T \in \mathcal{T}', \\ \int_{T'} (u_{CR} - v)\lambda_{CR}dx \text{ for } T \in \mathcal{T} \setminus \mathcal{T}'. \end{cases}$$

The estimator depends on a function $v \in K$. Three different possibilities Ⓐ – Ⓒ for $v \in P_1(red(\mathcal{T})) \cap C_0(\Omega) \cap K$ are presented in this paper all of which follow from linear interpolation once the values at the nodes are defined.

Ⓐ v_{P1red} is computed in two steps. In a first step a function w_2 is defined to equal $u_{CR}(mid(\mathcal{E}))$ at the edges' midpoints and the values at a node $z \in \mathcal{N}(\Omega)$ are chosen such that on the patch ω_z^* w_2 minimises $1/2\||w - u_{CR}\||^2_{NC(\omega_z^*)} + \int_{\omega_z^*} (\chi - w)\Pi_0\lambda_{CR}dx + \int_{\omega_z^*} (u_{CR} - w)\lambda_{CR}dx$ over all function $w \in P_1(red(\mathcal{T})) \cap C_0(\Omega)$. This is a one dimensional minimisation problem. In a second step set $v := P_K(w_2)$ where P_K is the projection onto the set of admissible function with respect to the energy norm.

Ⓑ v_{Energy} is computed in two steps. In a first step a function w_2 is defined which equals $u_{CR}(mid(\mathcal{E}))$ at the edges' midpoints and the values at a node $z \in \mathcal{N}(\Omega)$ are chosen such that on the patch ω_z^* w_2 minimises $E(w) - E_{NC}(u_{CR})$ locally over all function $w \in P_1(red(\mathcal{T})) \cap C_0(\Omega)$. This is a one dimensional minimisation problem. In a second step set $v := P_K(w_2)$ where P_K is the projection onto the set of admissible function with respect to the energy norm.

Ⓒ v_{J2} is set to the arithmetic mean of the different values of u_{CR} at the nodes. The values at the edges' midpoints are chosen such that $\fint_E v_{J2}\,ds = u_{CR}(mid(E))$ along any edge $E \in \mathcal{E}$.

Those three functions lead to the error estimators $\eta_{P1red}, \eta_{Energy}, \eta_{J2}$.

With Theorem 3.3 estimate the lower bounds $\mu_j, (j = 1, 2)$ for the energy defined as

$$\mu_1 := E_{NC}(u_{CR}) - \frac{\kappa_{CR}^2}{2} \|h_{\mathcal{T}} f\|^2_{L^2(\Omega)}$$

$$\mu_2 := E_{NC}(u_{CR}) - \left(\kappa_{CR} \|h_{\mathcal{T}}(f - \lambda_{CR})\|_{L^2(\Omega)} + osc(\lambda_{CR}, \mathcal{T}')\right)^2/2$$
$$- \int_{\mathcal{T}'} (\chi - u_{CR})\Pi_0\lambda_{CR}dx + \int_{\mathcal{T} \setminus \mathcal{T}'} (I_{NC}\chi - \chi)\lambda_{CR}dx.$$

MARK the minimal set $\mathcal{M}_\ell \subseteq \mathcal{T}_\ell$ such that

$$\theta \sum_{T \in \mathcal{T}_\ell} \eta(T) \leq \sum_{T \in \mathcal{M}_\ell} \eta(T).$$

REFINE by red-refinement of elements in $\mathcal{M}_\ell$ and red-green-blue-refinement of further elements to avoid hanging nodes and compute $\mathcal{T}_{\ell+1}$. od

OUTPUT efficiency indices $\sqrt{\sum_{T \in \mathcal{T}} \eta_v^2(T)}/\||u - u_{CR}\||_{NC}$.

4.2 L-Shaped Domain

Figure 4.1: Convergence history of the exact errors for the non-conforming and conforming FEM on uniform and adaptive meshes (left) and efficiency indices (right) of the three different error estimators for the non-conforming scheme as functions of the number of unknowns on adaptive and uniform meshes for the Example 1 with the error estimators from Theorem 3.2.
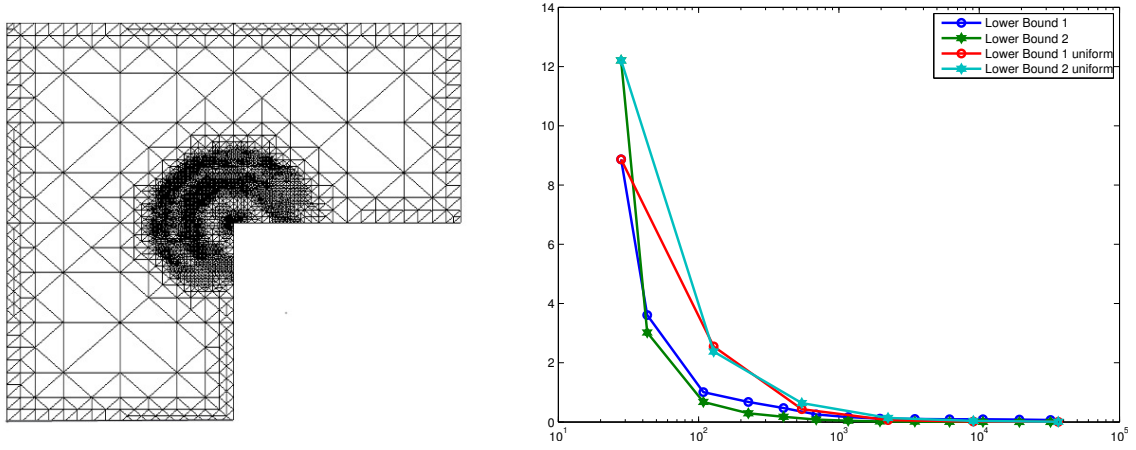
Figure 4.2: Adaptive mesh with refinement indicator η_{P1red} (left) and $E(u)-$lower bound for the lower bounds μ_1 and μ_2 of the exact energy $E(u)$ on uniform and adaptive meshes for Example 1.

The first example from [BC04] involves the L-shaped domain $\Omega := (-2, 2)^2 \setminus ([0, 2] \times [-2, 0])$, the obstacle $\chi := 0$, the Dirichlet data $u_D := 0$, and the source term

$$f(r, \varphi) := -r^{2/3} \sin(2\varphi/3)(7/3(\partial g/\partial r)(r)/r + (\partial^2 g/\partial r^2)(r)) - H(r - 5/4)$$
$$g(r) := \max\{0, \min\{1, -6s^2 + 15s^4 - 10s^3 + 1\}\} \text{ for } s := 2(r - 1/4)$$

156

with the Heaviside function H. The exact singular solution reads

$$u(r, \varphi) = r^{2/3} g(r) \sin(2\varphi/3)$$

and has a typical corner singularity at the re-entrant corner. The experiments on uniform meshes show an experimental convergence rate of approximately -0.44 in terms of the number of degrees of freedom which appears suboptimal when compared with the optimal rate $-1/2$ for linear convergence for the conforming and the non-conforming finite element method. The non-conforming FEM leads to efficiency indices between 1.6 and 3 as shown in Figure 4.1 on the right. For the calculation of the efficiency index it needs to be taken into account, that the error estimator does not only estimate $\||u - u_{\mathrm{CR}}\||_{\mathrm{NC}}$ but also the terms

$$\||u - u_{\mathrm{CR}}\||_{\mathrm{NC}} + \Lambda(u - v)^{1/2} + \left(\int_{\mathcal{T}'} \lambda_{CR} \Pi_0 (\chi - u) dx \right)^{1/2}$$

$$+ \left(\int_{\mathcal{T} \setminus \mathcal{T}'} (\mathrm{I}_{\mathrm{NC}} \chi - u) \lambda_{CR} dx \right)^{1/2} + \||\Lambda - \Lambda_{CR}\||.$$

The computation of the entire exact error will lead to even better efficiency indices. The adaptive algorithm for the non-conforming method, with η_{P1red} as the refinement indicator, leads to an improved convergence rate of approximately -0.5. An adaptive algorithm for the conforming scheme shows the same behaviour (see Figure 4.1 on the left). This indicates, that the error estimators yield good results on unstructured grids as well as on uniformly refined meshes. The efficiency indices for the non-conforming method on an adaptive mesh are comparable to the efficiency indices on uniform meshes (see Figure 4.1). Furthermore the mesh displayed in Figure 4.2 (left) shows, that the contact zone is less refined by the refinement indicator η_{P1red} then the area around the re-entering corner at the point $(0, 0)$, although the boundary of the contact zone is well refined and clearly visible. The lower bounds for the minimal energy $E(u)$ show very similar behaviour. Both lower bounds converge slightly faster for the adaptive algorithm with η_{P1red} as a refinement indicator, than on uniform meshes as demonstrated in Figure 4.2 on the right.

4.3 Smooth Obstacle

This example from [GK09] on the square domain $\Omega := (-1, 1)^2$ involves the smooth obstacle $\chi(x, y) := -(x^2 - 1)(y^2 - 1)$, the homogeneous Dirichlet data $u_D|_{\partial\Omega} := 0$ and the source term $f := \Delta\chi$. The exact solution to this problem reads $u = \chi$. On uniformly refined meshes both the conforming and non-conforming finite element method lead to an experimental convergence rate of -0.5. Both methods converge with the same convergence rate for an adaptively refined mesh; cf. Figure 4.3 on the left. The non-conforming scheme leads to good efficiency indices. Again it needs to be taken into account, that more terms of the exact error are estimated. Figure 4.3 right shows the efficiency indices both for a uniform mesh and an adaptive mesh. For this example the adaptive algorithm leads to an almost uniform refinement of the entire domain,
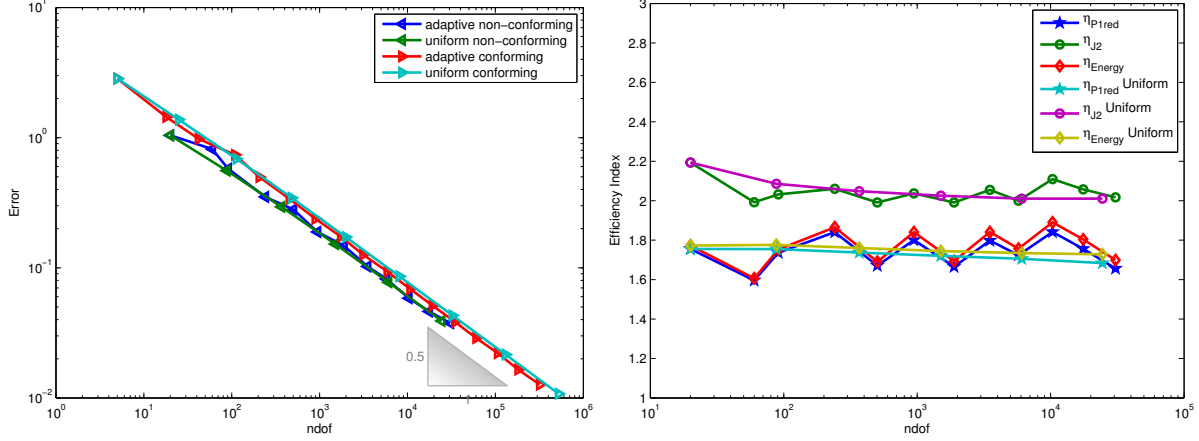
Figure 4.3: Convergence history of the exact errors for the non-conforming and conforming FEM on uniform and adaptive meshes (left) and efficiency indices (right) of the three different error estimators for the non-conforming scheme as functions of the number of unknowns on adaptive and uniform meshes for the Example 2 with the error estimators from Theorem 3.2.

although the central part and the corners are refined more strongly as can be seen in Figure 4.4 on the left. The adaptive refinement does not indicate the contact zone for this problem. The lower bounds for the minimal energy are comparable for this example as well. They converge with a very similar behaviour both on uniform meshes and on an adaptively refined mesh with the refinement indicator η_{P1red} although the adaptive algorithm leads to slightly better results (see Figure 4.4 on the right).

4.4 Pyramid Problem

This example from [BC04] has an unknown exact solution u. The experiment is conducted on the square domain $\Omega := (-1, 1)^2$ and involves the pyramidal obstacle $\chi(x, y) := \text{dist}(x, y, \partial\Omega)$. This experiment has homogeneous Dirichlet data and the constant source term $f := 1$. The exact solution is approximated by solving the discrete problem after two additional red refinements in each step. Both the conforming and the non-conforming scheme lead to the experimental convergence rate of -0.5 on uniform meshes, as can be seen in Figure 4.5 on the right. The adaptive algorithm does not show this convergence rate but rather has very bad convergence. On uniform meshes additional undocumented experiments show that the error estimator for the non-conforming finite element method only converge with a convergence rate of -0.3. The efficiency indices in Figure 4.5 on the right confirm this, as they do not tend to a constant value but continue to rise. This does not contradict Theorem 3.2 as the error estimate is still reliable. The efficiency in Theorem 3.4 is shown for all the error term $\||u - u_{\text{CR}}\||_{\text{NC}} + \||\Lambda - \Lambda_{CR}\||_* + \Lambda(u - v) + \int_\Omega \lambda_{CR}\Pi_0(\chi - u)dx$. The numerical experiments only considers the error term $\||u - u_{\text{CR}}\||_{\text{NC}}$ and hence indicate that the remaining terms have more impact on the overall error then in the other examples. Furthermore, the approximation of u by two additional red refinements might not be a good approximation and can lead to errors. The mesh created by the adaptive algorithm
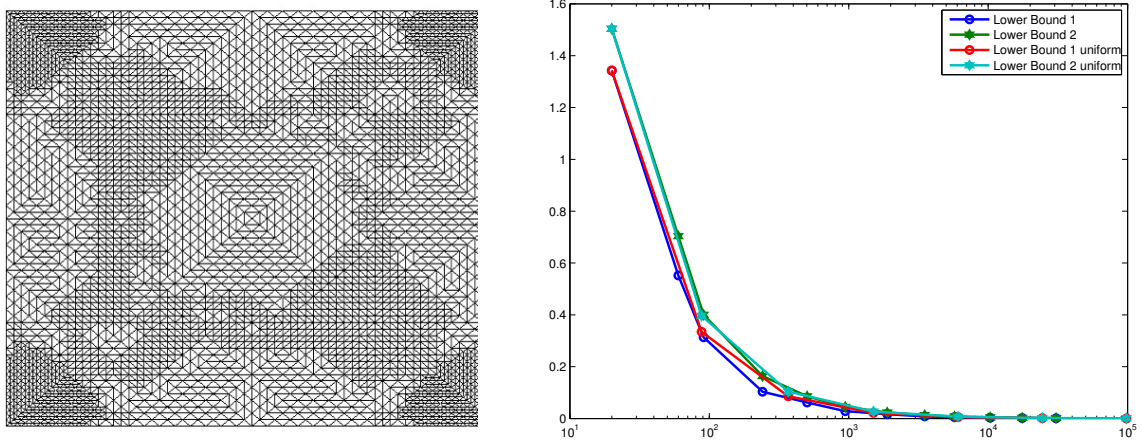
Figure 4.4: Adaptive mesh with refinement indicator η_{P1red} (left) and $E(u)-$lower bound for the lower bounds μ_1 and μ_2 of the exact energy $E(u)$ on uniform and adaptive meshes for Example 2.

depicted in Figure 4.6 on the left demonstrates very clearly the one-dimensional contact zone, which is the union of the two diagonals. The area towards the diagonals is much more refined. Figure 4.6 on the right shows the quality of the lower bounds of the exact energy. As in the examples before the lower bound μ_1 shows the better convergence. This holds true, both for the adaptive algorithm with the refinement indicator η_{P1red} and for the calculation on uniform meshes. In this example the initial mesh is aligned with the obstacle and hence it does not make a big difference whether the adaptive or the uniform mesh design is employed.

5 Conclusions

The numerical experiments confirm the theoretical results from Section 3. It is clearly shown that guaranteed upper error bounds are possible even for a non-conforming discretisation of the non-linear obstacle problem. The accuracy of the non-conforming method differs from those of the conforming scheme only by a multiplicative constant, but overall they show the same convergence rate in terms of the number of degrees of freedom. This is in contrast to the statement on page 111 in [Bra07]: In the numerical experiments of this paper, even for a singular solution, the convergence rate is comparable for conforming and non-conforming FEMs; cf. [CPS12] and the website [Bra].

The first two benchmark examples show that an adaptive algorithm leads to the optimal convergence rate of -0.5 whereas, in Example 1, the uniform algorithm only leads to a convergence rate of -0.44. The third benchmark example does not show this improved behaviour with respect to the convergence rate but, nonetheless, the adaptive algorithm leads to an improvement of the efficiency indices. For uniform refinement the error control for this example is not efficient, but this proves to be the case for adaptive refinement.

The lower bounds of the minimal energy show that the bound μ_1 is preferable to the lower
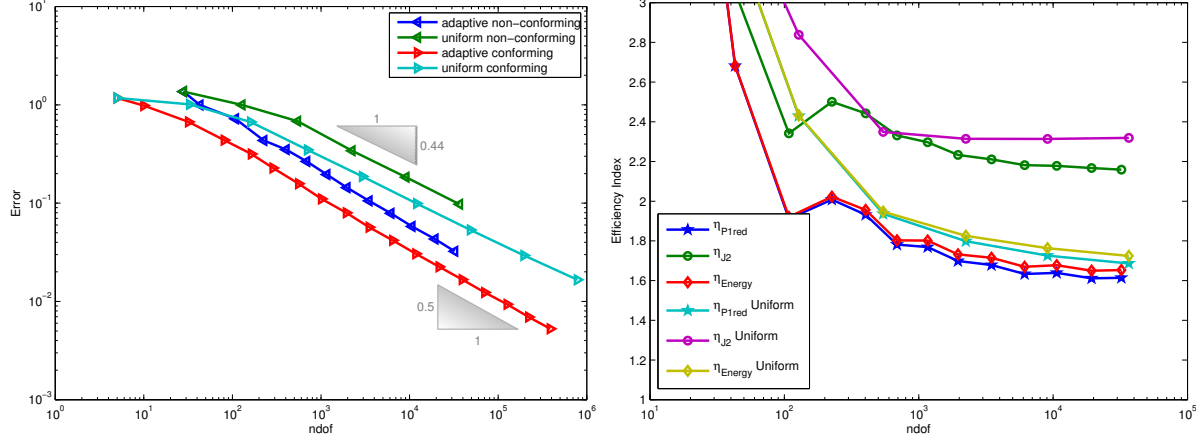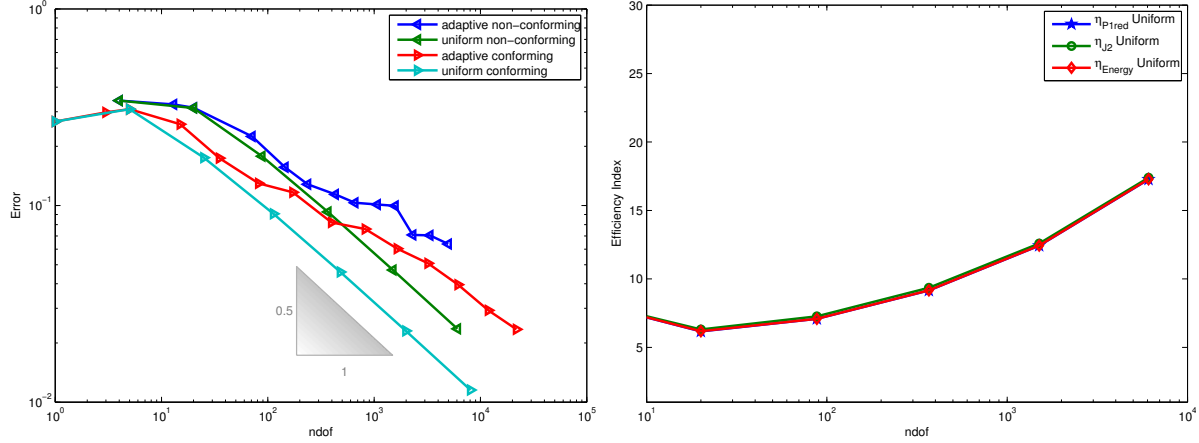
Figure 4.5: Convergence history of the exact errors for the non-conforming and conforming FEM on uniform and adaptive meshes (left) and efficiency indices (right) of the three different error estimators for the non-conforming scheme as functions of the number of unknowns on adaptive and uniform meshes for the Example 3 with the error estimators from Theorem 3.2.

bound μ_2, although in the examples at hand, the difference between the two estimates, as well as between uniform and adaptive mesh refinement is marginal. All three experiments conducted for this paper show that an adaptively refined mesh also leads to better lower bounds for the energies.

All the adaptive refinements were done with the error estimator η_{P1red} as a refinement indicator. Undocumented experiments show that the same adaptive algorithm with either η_{J2} or η_{Energy} as refinement indicators establish comparable results.

Acknowledgements

REFERENCES

[ACF]   J. Alberty, C. Carstensen, and S.A. Funken. Remarks aroung 50 lines of matlab: short finite element implementation. *Numerical Algorithms*, 20.

[BC04]   S. Bartels and C. Carstensen. Averaging techniques yield reliable a posteriori finite element error control for obstacle problems. *Numer. Math*, 99(2):225–249, 2004.

[Bra]   D. Braess. Finite elements. theory, fast solvers and applications in solid mechanics. http://homepage.ruhr-uni-bochum.de/dietrich.braess/FEM_corr_eng3.htm.

[Bra05] Dietrich Braess. A posteriori error estimators for obstacle problems—another look. *Numer. Math.*, 101(3), 2005.

[Bra07] Dietrich Braess. *Finite elements: Theory, fast solvers, and applications in solid mechanics*. Cambridge University Press, 2007.
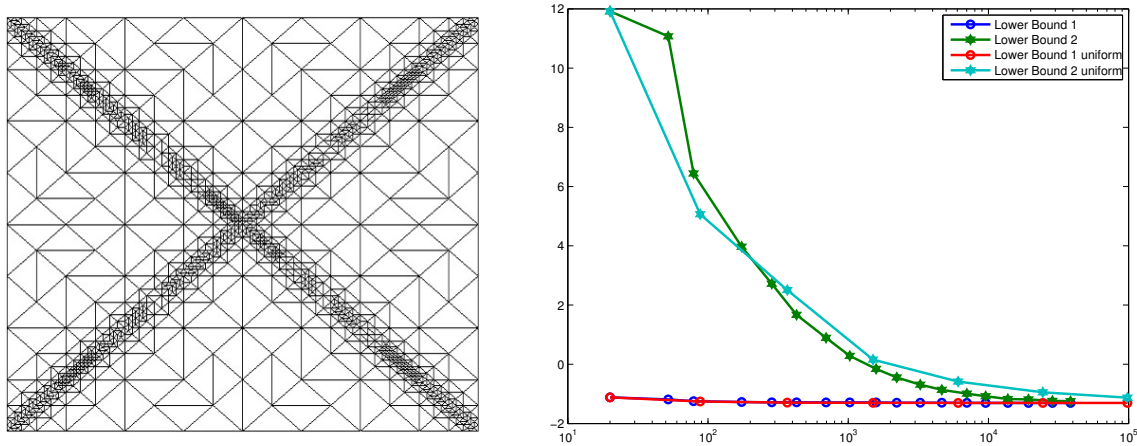
Figure 4.6: Adaptive mesh with refinement indicator η_{P1red} (left) and $E(u)$−lower bound for the lower bounds μ_1 and μ_2 of the exact energy $E(u)$ on uniform and adaptive meshes for Example 3.

[BS08]   S. Brenner and L.R. Scott. *The Mathematical Theory of Finite Element Methods*. Springer, 2008.

[CGR12] Carsten Carstensen, Joscha Gedicke, and Donsub Rim. Explicit error estimates for courant, crouzeix-raviart and raviart- thomas finite element methods. *Jounal of Computational Mathematics*, 30(4):337–353, July 2012.

[CM10]   C. Carstensen and C. Merdon. Estimator competition for Poisson problems. *J. Comput. Math.*, 28(3):309–330, 2010.

[CPS12] C. Carstensen, D. Peterseim, and M. Schedensack. Comparison results of finite element methods for the poisson model problem. *SIAM J. Numerical Analysis*, 50(6):2803–2823, 2012.

[Fal74]   R Falk. Error estimates for the approximation of a class of variational inequalities. *Mathematics of Computation*, 28(128):963–971, October 1974.

[GK09]   C. Gräser and R. Kornhuber. Multigrid methods for obstacle problems. *J. Comput. Math.*, 27(1):1–44, 2009.

[KS80]   D. Kinderlehrer and G. Stampacchia. *An Introduction to Variational Inequalities and Their Applications*. Harcourt Brace Jovanovich, 1980.

[Vee01]   A. Veeser. Efficient and reliable a posteriori error estimators for elliptic obstacle problems. *SIAM J. Numer. Anal.*, 39(1), 2001.

[Wan03] Lie-heng Wang. On the error estimate of nonconforming finite element approximation to the obstacle problem. *J. Comput. Math.*, 21(4):481–490, 2003.

VASCULAR RECONSTRUCTION MODELLING OF LUMEN-ADAPTED ARTERIES WITH STIFFENED GRAFTS

CATARINA F. CASTRO[*], CARLOS C. ANTÓNIO[*] AND LUISA C. SOUSA[*]

[*] Instituto de Engenharia Mecânica (IDMEC) e Departamento de Engenharia Mecânica (DEMec)
Faculdade de Engenharia da Universidade do Porto
Rua Dr. Roberto Frias, 4200-465 Porto, Portugal
e-mail: ccastro@fe.up.pt

Key words: Shape Optimization, Genetic Algorithms, Biomedical Engineering, ANN Applications.

Abstract. Optimization of prosthetic graft configuration with regard to blood dynamics is the major target of this research. Hemodynamic simulations of idealized arterial bypass systems are acquired using a finite element arterial blood simulator exhibiting hemodynamic flow changes due to compliance differences of a stiff graft and an elastic arterial wall. An artificial neural network simulating hemodynamic specific conditions is developed in order to reduce computational time. Optimal graft configurations are searched using a multi-objective genetic algorithm. An optimal set of solutions are presented and analyzed.

1 INTRODUCTION

Vascular grafts are special tubes that serve as replacements for damaged blood vessels. When suitable autologous veins are unavailable, prosthetic graft materials are used for peripheral arterial revascularisations. Research studies over the last three decades have established that hemodynamic interactions with the vascular surface as well as surgical injury are inciting mechanisms capable of eliciting distal anastomotic intimal hyperplasia (IH) and ultimate bypass graft failure. Compliance and calibre mismatch between native vessel and graft contributes towards poor long term patency [1].

The ideal vascular bypass graft would replicate the mechanical properties of native artery perfectly. Research study of dynamic arterial wall properties of large arteries such as the carotid and femoral arteries is becoming more common. Using non-invasive techniques the maximum and minimum arterial diameters and the intima-media thickness (IMT) at the point of maximum diameter and minimum diameter have been determined over the cardiac cycle.

The pressure wave following the ejection of blood by the heart is gradually conveyed to the periphery. Close to the heart the wave velocity is of the order of 5 m/s and gradually increases towards the periphery [2]. The arterial diameter and intima-media thickness values can be used together with the blood pressure measurements to calculate several standard arterial stiffness indices [3]. Nevertheless, in vitro and in vivo experiments have demonstrated that the diameter-pressure relationship exhibits an exponential characteristic [4], which can be accounted for by replacing the pulse pressure in the distensibility coefficient by the logarithm

of the ratio of systolic and diastolic pressure [5].

Computational approaches have been used simulating blood flow through idealized bypass models [6, 7, 8]. They exhibit particular patterns characterized by the presence of recirculation zones and secondary flows in certain regions. Pulsatile simulations of artery-graft systems show elevated and negative wall shear stresses at the toe, heel and hood regions of the anastomosis. The region where the wall shear stress (WSS) is negative corresponds to recirculation regions within the artery. Regions of reverse flow are usually associated with local deposit of particles, which results in a blockage of the artery. Therefore, potential artery-graft anastomosis design improvements that reduce the amount of wall shear stress and recirculation zones may have to be performed in order to increase the clinical success of vascular bypass grafts.

In this project a developed multi-objective genetic algorithm [6] is considered in order to reach optimal graft geometries for idealized arterial bypass systems of fully occluded host arteries. Genetic algorithms require a large number of computer simulations. So, an artificial neural network (ANN) is developed to efficiently calculate specific outputs associated with blood flow for predefined graft geometries. Input and target data have been acquired using a modified version of a finite element (FE) arterial blood simulator previously developed and tested considering fully unsteady incompressible Navier-Stokes equations and a three-dimensional geometry [9, 10].

2 MULTI-OBJECTIVE GENETIC ALGORITHM

Geometry plays the key role in determining the nature of hemodynamic patterns. This investigation will address rigid sinusoidal grafts with walls drawn by sine curves. In order to understand the dependence of the bypass blood flow on arterial wall variability, finite element model simulations were performed using a modified version of a previously developed code [9, 10]. Figure 1 presents the deformable model that includes both the proximal and distal bypass sections in order to analyze the flow development along the entire bypass.
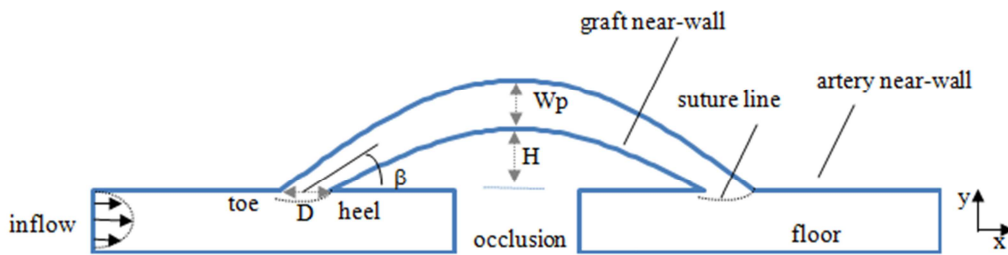
Figure 1: Anastomotic configuration and nomenclature of the graft/artery geometry

For modeling purpose the simplified arterial graft prosthesis is a tubular vessel disposed around a longitudinal axis as described in the bypass model given in Figure 1. Each design vector b has four geometric components $b = (H, \beta, Wp, D)$ as displayed in Figure 1: the distance from the near wall of the graft to the near wall of the artery H, the junction angle β, the width of the prosthesis at its longitudinal symmetric line Wp and the suture line dimension D. The host artery is assumed to be a fully stenosed conduit, simulated using two cylindrical tubes of 6 millimeter diameter, the proximal host artery before the obstruction and the distal

host artery after obstruction. The graft is symmetric and meets the host artery with a side-to-end proximal anastomosis and an end-to-side distal anastomosis. As usually adopted by most previous investigations, vessels are assumed to be impermeable tubes.

A general multi-objective optimization seeks to optimize the components of a vector-valued objective function. For the shape optimization application presented here, the genetic algorithm (GA) begins by randomly setting an initial population of possible individuals, where individuals represent different graft geometries. The successive populations maintain the same number of individuals as it evolves throughout successive generations. Each individual is referred as a chromosome containing design variable values referred as genes of the chromosome over which genetic operators are applied. Operators such as selection/crossover, mutation and elimination supported by an elitist strategy are considered to ensure that fitness of the forthcoming generations is always improved [6, 9].

For the study presented here two functional values qualifying and quantifying the graft local hemodynamics are considered. Multiple hemodynamic factors capable of eliciting a hyperplastic response at the cellular level indicate the potential significance of platelet-wall interactions coinciding with regions of low WSS in the development of intima hyperplasia (IH). The first functional to be considered is:

$$f_1(\boldsymbol{b}) = \sum_1^{N_\Gamma} \|WSS\|^2 \tag{1}$$

where WSS are the smoothed wall shear stress values at the floor of the distal artery-graft junction. Sites of significant particle interactions with the vascular surface have been identified by functions associated to long near-wall residence times capturing a significant IH formation. A second component is associated with the fluid velocity distribution at the cross-section of the distal graft-artery junction:

$$f_2(\boldsymbol{b}) = \sum_{\Omega_{g-a}^*(\boldsymbol{b})} \mathbf{v_x}^2 \tag{2}$$

For the same inlet velocity profile, values of this function $f_2(\boldsymbol{b})$ will be larger for disturbed longitudinal velocity distributions along the distal graft-artery junction and smaller for smooth and parabolic distributions.

In the project reported here, an artificial neural network (ANN) is developed to efficiently simulate blood flow for specific graft geometries. A set of randomly generated 500 input vectors $\boldsymbol{b} = (H, \beta, Wp, D)$ and target vectors $\boldsymbol{f} = (f_1(\boldsymbol{b}), f_2(\boldsymbol{b}))$ has been collected using the FE code within the design space given as follows

$$\begin{aligned} 10 \leq H \leq 30 \; mm \\ 0.15 \leq \beta \leq 0.785 \; rad \\ 6 \leq Wp \leq 10 \; mm \\ 6 \leq D \leq 10 \; mm \end{aligned} \tag{3}$$

The ANN analysis was performed using MATLAB with the Neural Network Application Toolbox (The MathWorks Inc., MA, USA). The multilayer feed-forward neural network of the software is well suited for function fitting problems. Since the initial weights and biases are randomly set more than one trial was done. The regression analysis was considered using the R value as an indication of the relationship between the outputs and targets. For this example, the training data indicates a good fit. The validation and test results showed R values greater than 0.99. Once the ANN has demonstrated acceptable pattern recognition skills, its

creation has been achieved being ready for use.

3 EFFECT OF LUMEN-ADAPTED ARTERY

The influence of wall variability on blood flow behavior has been studied and published either in vascular replicas as well as in real vascular anatomies. By their elastic nature, major conduit arteries should be able to store blood volume temporarily during systole and release it during diastole. This reduces the systolic blood pressure required for the flow of a given volume quantity and gradually suppresses the pulsatile flow pattern. The repetitive stretching of the wall (strains of up to 10 per cent) may cause fragmentation of the elastic fibers in the wall, modifying wall elasticity. To maintain wall stress the elastic arteries respond with a diameter increase in combination with an increase of arterial wall thickness. In recent years various methods have been developed to assess and monitor the above interaction. Most of these methods are based on ultrasound techniques because of its wide availability and its non-invasive and non-traumatic nature. Presently these techniques enable the assessment of wall thickness, diastolic diameter, distension waveform, i.e. the time-dependent change in diameter, the relative pulsatile increase in diameter, and pulse wave velocity, for elastic and muscular arteries in humans. For the carotid artery, comparison between the computed and experimentally measured wall movement during a cardiac cycle showed that the pressure waveform plays the main role in driving the wall movement while the pressure gradient resulting from the flow only has a secondary influence [11].

A basic problem for the assessment and evaluation of mechanical parameters is to acquire the local time dependent blood pressure. The artery responds to the pulsatile change in transmural pressure during the cardiac cycle with a pulsatile change in cross-sectional area. The transient wall motion can be divided into three phases: rapid dilation, rapid partial contraction, and slow contraction. The minimum artery diameter occurs during the low-pressure end-diastolic cardiac phase and the maximum artery diameter during the peak systolic phase. Evaluation of arterial vessel diameter variability along the cardiac cycle has been addressed by several authors. Using computerized edge detection-sequential multi-frame image processing Selzer et al. presented a typical plot of the continuous measurement of carotid arterial diameter over 2 cardiac cycles [3].

In this work, the material of the graft wall is assumed to be rigid and the material of the artery wall is assumed to be incrementally linear elastic. For the 6mm diameter artery, the basic geometrical parameters considered here are the initial (end-diastolic) diameter, the change in diameter and the wall thickness. As expected, intima-media thickness (IMT) varies according to the arterial dimension. When the arterial diameter is at its minimum, IMT is at its thickest point and when the arterial diameter is at its maximum, IMT is at its thinnest point. Using Selzer et al. measurements [3] polynomial curves were fitted defining both the arterial diameter and IMT over the cardiac cycle. The shape of the pressure waveform is highly site-dependent and is modified not only by proximal and distal bifurcations, curvatures and artery tapering, but also by the site-dependent mechanical characteristics of the artery itself. In the numerical simulation, a flow waveform boundary is specified at the model inlet. A pulsatile velocity curve along the cardiac cycle has been adapted by Carneiro [12] from Taylor and Draney [13].

In order to realize the necessity to introduce the arterial wall elastic behavior, FEM

simulations were performed considering the same geometric bypass defined by parameters $b = (H = 18.67mm, \beta = 0.782rad, Wp = 6.11mm, D = 7.40mm)$ at the end of the diastolic phase. In the computations, the non-Newtonian behavior of blood will follow the Casson model as used in the software validation by Sousa et al. [9]. This model shows both yield stress and shear-thinning non-Newtonian viscosity, broadly used to describe the shear thinning behavior of blood [14]. Comparison between the Newtonian and non-Newtonian fluid models has demonstrated that the velocity profile of the non-Newtonian fluid is somewhat flattened, due to its shear-thinning behavior.
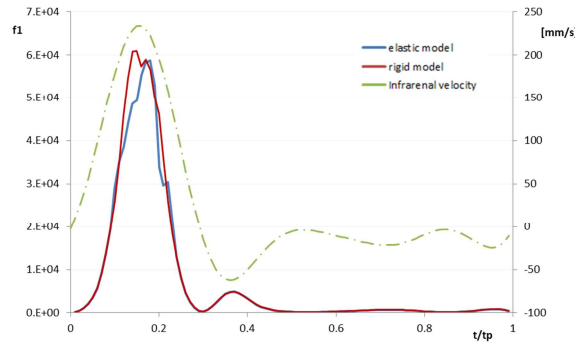
Figure 2: Magnitude variability of the functional associated to wall shear stress for elastic and rigid artery model along the cardiac cycle
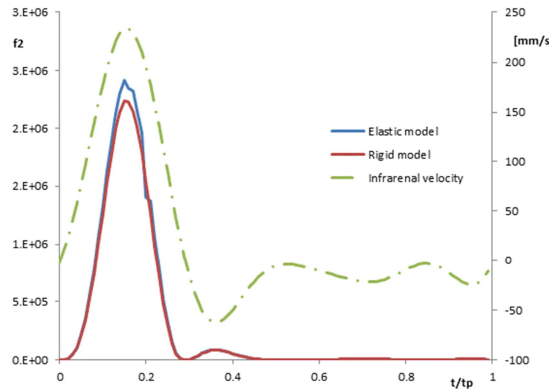
Figure 3: Magnitude variability of the functional associated to velocity distribution at the distal graft-artery junction for elastic and rigid artery model along the cardiac cycle

The introduction of arterial diameter variability along the pulsatile blood velocity curve induces 0.4mm diameter variability so the simulated diameter artery goes from 6mm at the end-diastolic phase up to 6.4mm at the systolic peak. Simulated values for the objective functions $f_1(b)$ and $f_2(b)$ considering different artery conditions are given bellow. Figure 2 presents a comparison of the magnitude variability of functional $f_1(b)$ associated to wall shear stress between the elastic and the rigid artery model together with the pulsatile velocity waveform. As expected high WSS values correspond to the systolic phase. The difference between the rigid and the elastic model is that higher and earlier values are observed for the

rigid model.

Figure 3 presents a comparison of the magnitude variability of functional $f_2(\boldsymbol{b})$ associated to fluid velocity distribution at the cross-section of the distal graft-artery junction between the elastic and the rigid artery model. Contrarily to the previous functional, the rigid model presents higher values at the peak systolic phase as compared to the elastic model. This result can be explained due to lack of compliance between artery and prosthetic graft. Mismatched biomechanical properties between the graft and native surrounding tissue are commonly cited as a cause of graft failure.

4 RESULTS AND CONCLUSIONS

Pareto optimality is a concept that formalizes the trade-off between a given set of possible contradicting objectives. By only one time global search procedure all the Pareto optimal solutions are found managing the drawing of the Pareto front and then extracting optimal solutions according to selected preferences. As a compromise between computer time and population diversity, parameters for the genetic algorithm were taken as 12 and 5 for the population and elite group size, respectively. The number of bits in binary codifying for each design variable is 5. The GA termination has been defined by fixing the total number of generations as 300. For each generation 6 new individuals (five from crossover plus one from mutation) are created. Simulation results for the optimized bypass graft are presented in Figures 4 and 5.
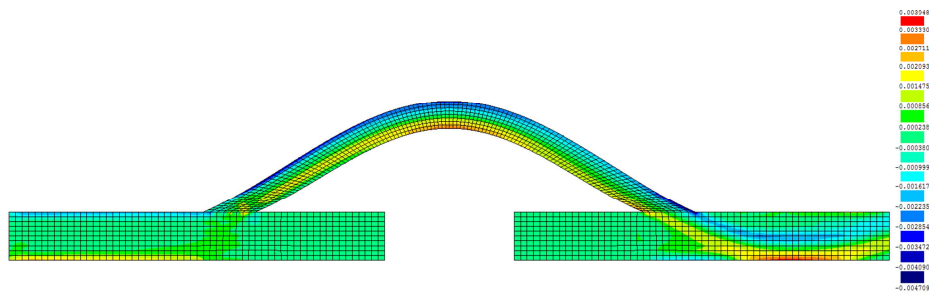
Figure 4: WSS distribution for the optimal elastic artery model at peak systolic phase
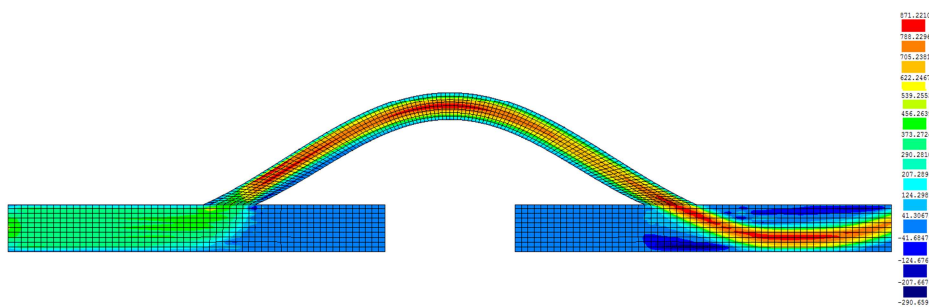
Figure 5: Velocity distribution for the optimal elastic artery model at peak systolic phase

Qualitatively, the distributions of wall shear stress as well as velocity in the elastic artery model do not change significantly compared to the rigid artery model, despite the quite large wall variations. However, less flow separation and reversed flow is observed in the elastic

model. At peak systolic phase (t/tp=0.16), Figure 4 shows regions of high WSS appear around stenosis at the distal corner of the proximal bifurcation and at the toe of the distal anastomosis suggesting that both the proximal and distal regions are responsible for early bypass graft failure.

The velocity values are given in Figure 5 (t/tp=0.16) demonstrating a very good quality of the finite element simulation being capable of capturing the flow acceleration as it emerges from the graft to the artery and the flow recirculation at the floor of the host artery, consistently with the expectations. The abrupt connection between artery and graft induces large velocity variations. Long residence times usually observable immediately after the toe of the distal anastomosis are quite undetectable. The importance of designing optimized bypass graft cannot be neglected.

Transient hemodynamic flow induces minimum artery diameter during the low-pressure end-diastolic cardiac phase and the maximum carotid artery diameter during the peak systolic phase and the IMT has an opposite behavior [3]. On the other hand, the prosthetic graft is stiff and the hemodynamic flow changes due to compliance differences across an anastomosis cause increased shear stress to damage endothelial cells and also reduced shear stress leading to areas of relative stagnation and increasing interaction between platelets and vessel wall. The aim of geometry design is to minimize these disruptive flow characteristics. The optimization process manages to achieve geometries presenting wall shear stress values with the expected variability for the blood behavior in the systemic arterial tree.

The problem reported here is related with both shape design and flow control that are involved in the simulation of the bypass system. Improving blood flow dynamics in the graft system is an important element for the long-term success of bypass surgeries. Comparison between the rigid and compliant models illustrates a considerable reduction of reversed flow in the distensible model [15]. This is because when the graft flow rate decreases during the deceleration phase, in order to satisfy the fluid's mass conservation in the rigid wall model, this decrease can be compensated for, only by means of reducing the flow velocity. However, in the compliant model, the vessel's contraction partially compensates the flow rate drop by reducing the cross sectional area; consequently, less would be left to be compensated for, by reducing the velocity. As a result, less reversed flow is present in the distensible-wall models.

Further investigation on the hemodynamic benefits of the blood flow in graft artery compliance and configurations and in more realistic bio-mechanical conditions will be addressed by the authors.

Acknowledgments

REFERENCES

[1] Sarkar, S., Salacinski, H.J., Hamilton G. and Seifalian, A.M. The Mechanical Properties of Infrainguinal Vascular Bypass Grafts: Their Role in Influencing Patency. *Eur J Vasc Endovasc Surg* (2006) **31**:627-636.

[2] Hoeks, A.P.G., Brands, P.J., Willigers, J.M. and Reneman, R.S. Non-Invasive Measurement of Mechanical Properties of Arteries in Health and Disease. *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine* (1999) **213**:195-202.

[3] Selzer, R.H., Mack, W.J., Lee, P.L., Kwong-Fu, H. and Hodis, H.N. Improved common carotid elasticity and intima-media thickness measurements from computer analysis of sequential ultrasound frames. *Atherosclerosis* (2001) **154**:185-193.

[4] Powalowski, T. and Pensko, B. A noninvasive ultrasonic method for the elasticity evaluation of the carotid arteries and its application in the diagnosis of the cerebrovascular system. *Arch. Acoustics* (1988) **13**: 109-126.

[5] Langewouters, G.J., Wesseling, K.H. and Goedhard, J.A. The static elastic properties of 45 human thoracic and 20 abdominal aortas in vitro and the parameters of a new model. *Journal of Biomechanics* (1984) **17**: 425-435.

[6] Castro, C.F., António, C.C. and Sousa, L.C. Multi-objective optimization of bypass grafts in arteries. *TMSi - Sixth International Conference on Technology and Medical Sciences*, Porto, Portugal (2010).

[7] Su, C.M., Lee, D., Tran-Son-Tay, R. and Shyy, W. Fluid flow structure in arterial bypass anastomosis. *J Biomech Eng.* (2005) **127**(4): 611-618.

[8] Probst, M., Lülfesmann, M., Bücker, H.M., Behr, M. and Bischof, C.H. Sensitivity of shear rate in artificial grafts using automatic differentiation. *Int. J. Numer. Meth. Fluids* (2010) **62**:1047-1062.

[9] Sousa, L., Castro, C.F., Antonio, C.A. and Chaves, R. Computational Techniques and Validation of Blood Flow Simulation. *WEAS Transactions on Biology and Biomedicine, Included in ISI/SCI Web of Science and Web of Knowledge* (2011) **8**(4):145-155.

[10] Sousa, L.C., Castro, C.F., António, C.C. and Chaves, R. Blood flow simulation and vascular reconstruction. *Journal of Biomechanics* (2012) **45**: 2549-2555.

[11] Zhao, S.Z., Xu, X.Y., Hughes, A.D., Thom, S. A., Stanton, A. V., Ariff, B. and Long, Q. Blood flow and vessel mechanics in a physiologically realistic model of a human carotid arterial bifurcation. *Journal of Biomechanics* (2000) **33**: 975-984.

[12] Carneiro, A.F.G.C. *Influência do ciclo cardíaco no fluxo sanguíneo na vizinhança da bifurcação ilíaca*. Escola de Engenharia da Universidade do Minho, Portugal, PhD Thesis (2009).

[13] Taylor, C.A. and Draney, M. Experimental and Computational Methods in Cardio vascular and Fluid Mechanics. *Annual Review of Fluid Mechanics* (2004) **36:** 197-231.

[14] Ouared, R. and Chopard, B. Lattice Boltzmann simulations of blood flow: non-Newtonian reology ans clotting processes. *J. Stat. Phys.* (2005) **121**: 209-221.

[15] Kabinejadian, F. and Ghista, D.N. Compliant model of a coupled sequential coronary arterial bypass graft: Effects of vessel wall elasticity and non-Newtonian rheology on blood flow regime and hemodynamic parameters distribution. *Medical Engineering and Physics* (2012) **34**(7):860-872.

RECENT ADVANCES IN THE CONTROL OF PGD-BASED APPROXIMATIONS

LUDOVIC CHAMOIN*, PIERRE LADEVEZE*, AND FLORENT PLED*

*Laboratoire de Mécanique et Technologie (LMT-Cachan)
Ecole Normale Supérieure de Cachan
61 Avenue du Président Wilson, 94230 Cachan, France
e-mail: {chamoin,ladeveze,pled}@lmt.ens-cachan.fr, www.lmt.ens-cachan.fr

Key words: Model reduction, Verification, Proper Generalized Decomposition, Constitutive relation error, Adjoint problem, Handbook techniques

Abstract. In this work, we define a verification procedure that enables to build guaranteed PGD-reduced models for linear elliptic or parabolic problems depending on many parameters. It is based on the general concept of constitutive relation error and provides for strict bounds on both global error and error on outputs of interest. Furthermore, it helps driving adaptive strategies by assessing contributions of various error sources. Consequently, virtual charts that may be constructed from the PGD approximate solution can be certified. Technicalities and performances of the control approach, in particular when dealing with a large set of model parameters, are detailed on a transient thermal problem.

1 INTRODUCTION

Due to the continuous advances in both modeling and computing resources, numerical simulation has become a common tool in science and engineering activities. Nowadays, it is numerically possible to deal with very complex models that aim at giving an accurate representation of the real world. However, due to an overwhelming computational effort, this practice remains difficult and often impossible when considering mathematical models with various fluctuating parameters. This case is for instance encountered when tackling stochastic or optimization problems in which a large amount of scenarios need to be considered. For such multi-parameter models, numerical simulation faces the so-called *curse of dimensionality* that leads to a huge number of degrees of freedom when using classical brute force (i.e. grid-based) approximation methods. Therefore, alternative computing approaches are necessary in this context.
During the last decade, model reduction techniques have been the object of a growing interest both in research and industry. They exploit the fact that the response of complex

models can often be approximated with a reasonable precision by the response of a surrogate model, seen as the projection of the initial model on a low-dimensional functional basis. Model reduction techniques, that distinguish themselves by the way of defining and constructing the reduced basis, are thus an appropriate computing tool for addressing multi-parameter models. In particular, an appealing model reduction technique called Proper Generalized Decomposition (PGD) has recently emerged and is currently the topic of various research works [1]. It is based on separation of variables within a spectral resolution approach, and contrary to the well-known POD, no partial knowledge on the solution is required which is a major asset. PGD basis functions (or modes) are computed on the fly, once for all and in an offline process, by solving a series of manageable mono-parameter problems. The obtained PGD approximation, that explicitly depends on all model parameters, can then be used in an online optimization process.

Performances of PGD have been shown in many applications for which variations in loading, boundary or initial conditions, material behavior, geometry,...were taken into account as additional model coordinates [2, 3, 4, 5] . However, a major difficulty for its transfer and intensive use in industry is the control of the PGD-reduced model. Indeed, certifying the accuracy of the PGD solution is a fundamental issue in order to perform robust and reliable design. This control requires mastering the number of PGD modes which are computed, but also the numerical methods which are employed in these computations.

There are actually very few works which have addressed the control of PGD-based approximations until now. Basic results on a priori error estimation for representations with separation of variables are given in [6], whereas a pioneering work mostly devoted to adaptivity can be found in [7]. A first robust approach for PGD verification, using the concept of Constitutive Relation Error (CRE) [8, 9], was proposed recently [10]. It applies to linear elliptic or parabolic problems depending on parameters, and provides for guaranteed PGD-reduced models for both global error and error on specific outputs of interest [11, 12]. Furthermore, the approach enables to assess contributions of various error sources (space and time discretizations, truncation of the PGD decomposition, etc.), which can help driving adaptive strategies.

In previous works [10, 11, 12], performances were shown with only few parameters. Here, we present new advances which have been performed in this PGD-verification method. We particularly focus on cases with numerous model parameters. We also use a non-intrusive procedure for the solution of the adjoint problem, in order to limit implementation issues. Therefore, virtual charts associated with quantities of interest and computed from PGD models can now fully benefit from the verification method to satisfy a prescribed accuracy. Numerical experiments on a transient thermal model with fluctuating material parameters are conducted to illustrate the proposed verification approach and its performances.

2   REFERENCE PROBLEM AND NOTATIONS

We consider a transient diffusion problem defined on an open bounded domain $\Omega \subset R^d$ ($d = 1, 2, 3$), with boundary $\partial\Omega$, over a time interval $\mathcal{I} = [0, T]$. We assume that a prescribed zero temperature is applied on part $\partial_u\Omega \neq \emptyset$ of $\partial\Omega$ and that the domain is subjected to a time-dependent thermal loading that consists of : (i) a given thermal flux $r_d(\mathbf{x}, t)$ on $\partial_q\Omega \subset \partial\Omega$, with $\partial_u\Omega \cap \partial_q\Omega = \emptyset$ and $\overline{\partial_u\Omega \cup \partial_q\Omega} = \partial\Omega$; (ii) a source term $f_d(\mathbf{x}, t)$ in Ω.
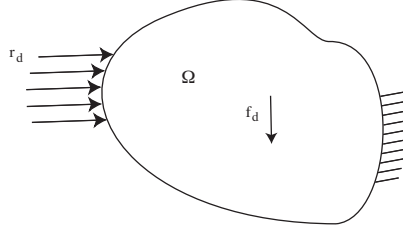
Figure 1: Representation of the reference problem.

For the sake of simplicity, we consider that initial conditions are set to zero. The material that composes Ω is assumed to be isotropic but heterogeneous and partially unknown. Therefore, diffusion coefficient k and thermal capacity c depend on space variable $\mathbf{x}$ but also on a set of N parameters $\mathbf{p} = [p_1, p_2, \ldots, p_N]$ belonging to a given bounded domain $\Theta = \Theta_1 \times \Theta_2 \times \cdots \times \Theta_N$.

The associated mathematical problem consists of finding the temperature-flux pair $(u(\mathbf{x}, t, \mathbf{p}), \boldsymbol{\varphi}(\mathbf{x}, t, \mathbf{p}))$, with $(\mathbf{x}, t, \mathbf{p}) \in \Omega \times \mathcal{I} \times \Theta$, that verifies :

- the thermal constraints :
$$u = 0 \quad \text{in } \partial_u\Omega \times \mathcal{I} \times \Theta \tag{1}$$

- the equilibrium equations :
$$c\frac{\partial u}{\partial t} = -\boldsymbol{\nabla} \cdot \boldsymbol{\varphi} + f_d \quad \text{in } \Omega \times \mathcal{I} \times \Theta \quad ; \quad \boldsymbol{\varphi} \cdot \mathbf{n} = r_d \quad \text{in } \partial_q\Omega \times \mathcal{I} \times \Theta \tag{2}$$

- the constitutive relation :
$$\boldsymbol{\varphi} = -k\boldsymbol{\nabla}u \quad \forall(\mathbf{x}, t, \mathbf{p}) \in \Omega \times \mathcal{I} \times \Theta \tag{3}$$

- the initial conditions :
$$u_{|t=0^+} = 0 \quad \forall(\mathbf{x}, \mathbf{p}) \in \Omega \times \Theta \tag{4}$$

$\mathbf{n}$ denotes the outgoing normal to Ω. In the following, in order to be consistent with other linear problems encountered in Mechanics (linear elasticity for instance), we carry out the change of variable $\boldsymbol{\varphi} \rightarrow -\boldsymbol{\varphi}$ which leads, in particular, to the new constitutive relation

$\varphi = k\boldsymbol{\nabla} u$.

Defining $\mathcal{V} = H_0^1(\Omega) = \{v \in H^1(\Omega), v_{|\partial_u\Omega} = 0\}$, the weak formulation in space of the previous problem reads for all $(t, \mathbf{p}) \in \mathcal{I} \times \Theta$:

$$\text{Find } u(\mathbf{x}, t, \mathbf{p}) \in \mathcal{V} \text{ such that } b(u, v) = l(v) \quad \forall v \in \mathcal{V} \tag{5}$$

with $u_{|t=0^+} = 0$. Bilinear form $b(\bullet, \bullet)$ and linear form $l(\bullet)$ are defined as :

$$b(u, v) = \int_\Omega \left\{ c\frac{\partial u}{\partial t}v + k\boldsymbol{\nabla} u \cdot \boldsymbol{\nabla} v \right\} \mathrm{d}\Omega \quad ; \quad l(v) = \int_\Omega f_d v \mathrm{d}\Omega - \int_{\partial_q\Omega} r_d v \mathrm{d}S \tag{6}$$

As regards the full weak formulation, we introduce the functional spaces $\mathcal{T} = L^2(\mathcal{I})$, $\mathcal{P}_i = L^2(\Theta_i)$, and $L^2(\mathcal{I}, \Theta; \mathcal{V}) = \mathcal{V} \otimes \mathcal{T} \otimes_{n=1}^N \mathcal{P}_n$. We therefore search solution $u \in L^2(\mathcal{I}, \Theta; \mathcal{V})$, with $\dfrac{\partial u}{\partial t} \in L^2(\mathcal{I}, \Theta; L^2(\Omega))$, such that :

$$B(u, v) = L(v) \quad \forall v \in L^2(\mathcal{I}, \Theta; \mathcal{V}) \tag{7}$$

with

$$\begin{aligned} B(u, v) &= \int_\Theta \left[\int_\mathcal{I} b(u, v)\mathrm{d}t + \int_\Omega cu(\mathbf{x}, 0^+)v(\mathbf{x}, 0^+)\mathrm{d}\Omega \right] \mathrm{d}\mathbf{p} \\ L(v) &= \int_\Theta \int_\mathcal{I} l(v)\mathrm{d}t\mathrm{d}\mathbf{p} \end{aligned} \tag{8}$$

The exact solution of (7), which is usually out of reach, is denoted u_{ex} (and $\boldsymbol{\varphi}_{ex} = k\boldsymbol{\nabla} u_{ex}$). It is classically approximated using the FEM in space associated with a time integration scheme and a given grid in the parameter space Θ.

Remark : in the steady-state case, we merely consider :

$$b(u, v) = \int_\Omega k\boldsymbol{\nabla} u \cdot \boldsymbol{\nabla} v d\Omega \quad ; \quad B(u, v) = \int_\Theta b(u, v) d\mathbf{p} \quad ; \quad L(v) = \int_\Theta l(v) d\mathbf{p} \tag{9}$$

3   CONSTRUCTION OF THE PGD APPROXIMATION

We now introduce the recently called _Proper Generalized Decomposition_ (PGD) technique [1] which constitutes an _a priori_ construction of a separated variables representation of the solution of (7). The approximate PGD solution is searched under the form :

$$u(\mathbf{x}, t, \mathbf{p}) \approx u_m(\mathbf{x}, t, \mathbf{p}) \equiv \sum_{i=1}^m \psi_i(\mathbf{x})\lambda_i(t)\Gamma_i(\mathbf{p}) \quad \text{with } \Gamma_i(\mathbf{p}) = \prod_{n=1}^N \gamma_{i,n}(p_n) \tag{10}$$

m is the order (i.e. number of modes) of the representation, whereas space functions $\psi_i(\mathbf{x})$, time functions $\lambda_i(t)$, and parameter functions $\gamma_{i,n}(p_n)$ respectively belong to $\mathcal{V}$, $\mathcal{T}$, and

$\mathcal{P}_n$. An attractive feature of the PGD technique is that the construction of modes does not require any knowledge on u. Neither $\psi_i(\mathbf{x})$ nor $\lambda_i(t)$ nor $\gamma_{i,n}(p_n)$ are initially given; these are computed on the fly. In the following, we give a classical version of the PGD technique, called *progressive Galerkin-based PGD* and inspired from classical fixed-point algorithms used to solve eigenvalue problems.

We assume that a PGD approximation of order $m-1$ has been computed. The PGD approximation of order m is then defined as:

$$u_m(\mathbf{x},t,\mathbf{p}) = u_{m-1}(\mathbf{x},t,\mathbf{p}) + \psi(\mathbf{x})\lambda(t)\Gamma(\mathbf{p}) \quad \text{with } \Gamma(\mathbf{p}) = \prod_{n=1}^{N} \gamma_n(p_n) \tag{11}$$

ψ, λ, and γ_n ($n=1,\ldots,N]$) are a priori unknown functions belonging respectively to discretized subsets $\mathcal{V}_h$, $\mathcal{T}_h$, and $\mathcal{P}_{nh}$; we assume $\mathcal{V}_h$ and $\mathcal{T}_h$ respect kinematic constraints and initial conditions, respectively. Starting from an initialization $\psi^{(0)}(\mathbf{x})\lambda^{(0)}(t)\Gamma^{(0)}(\mathbf{p})$, one builds a new mode representation $\psi^{(1)}(\mathbf{x})\lambda^{(1)}(t)\Gamma^{(1)}(\mathbf{p})$ thanks to the following sub-iteration:

– determine $\lambda^{(1)} \in \mathcal{T}_h$ such that:

$$B(u_{m-1} + \psi^{(0)}\lambda^{(1)}\Gamma^{(0)}, \psi^{(0)}\lambda^*\Gamma^{(0)}) = L(\psi^{(0)}\lambda^*\Gamma^{(0)}) \quad \forall \lambda^* \in \mathcal{T}_h \tag{12}$$

– for $n_0 = 1,\ldots,N$, determine $\gamma_{n_0}^{(1)} \in \mathcal{P}_{n_0 h}$ such that:

$$B(u_{m-1} + \psi^{(0)}\lambda^{(1)}\gamma_{n_0}^{(1)}\Gamma_{/n_0}^{(1,0)}, \psi^{(0)}\lambda^{(1)}\gamma^*\Gamma_{/n_0}^{(1,0)}) = L(\psi^{(0)}\lambda^{(1)}\gamma^*\Gamma_{/n_0}^{(1,0)}) \quad \forall \gamma^* \in \mathcal{P}_{n_0 h} \tag{13}$$

with $\Gamma_{/n_0}^{(1,0)} = \prod_{n=1}^{n_0-1} \gamma_n^{(1)} \times \prod_{n=n_0+1}^{N} \gamma_n^{(0)}$;

– determine $\psi^{(1)} \in \mathcal{V}_h$ such that:

$$B(u_{m-1} + \psi^{(1)}\lambda^{(1)}\Gamma^{(1)}, \psi^*\lambda^{(1)}\Gamma^{(1)}) = L(\psi^*\lambda^{(1)}\Gamma^{(1)}) \quad \forall \psi^* \in \mathcal{V}_h \tag{14}$$

Few sub-iterations are performed in practice; in the following numerical results, the process has been stopped after 4 sub-iterations. Furthermore, time function $\lambda^{(j)}(t)$ and parameter functions $\gamma_n^{(j)}(p_n)$ are normalized at each sub-iteration j.

4 GLOBAL ERROR ESTIMATION IN THE PGD FRAMEWORK

4.1 The Constitutive Relation Error method - Principle

The verification strategy we propose uses the concept of Constitutive Relation Error (CRE) (see [8] for full details). Let $(\hat{u}, \hat{\boldsymbol{\varphi}})$ be an admissible solution of the problem, i.e. verifying (1), (2), and (4). The CRE measure, that depends on $\mathbf{p}$, then reads:

$$E_{CRE}^2(\mathbf{p}) = \frac{1}{2}\int_{\mathcal{I}}\int_{\Omega}\frac{1}{k}[\hat{\boldsymbol{\varphi}} - k\boldsymbol{\nabla}\hat{u}]\cdot[\hat{\boldsymbol{\varphi}} - k\boldsymbol{\nabla}\hat{u}]d\Omega dt \equiv \frac{1}{2}|||\hat{\boldsymbol{\varphi}} - k\boldsymbol{\nabla}\hat{u}|||^2 \tag{15}$$

where $||| \bullet |||$ is an energy norm in the space-time domain, and one has the extension of the Prager-Synge theorem :

$$|||\boldsymbol{\varphi}_{ex} - \hat{\boldsymbol{\varphi}}^*|||^2 + \frac{1}{2}\int_\Omega c(u_{ex} - \hat{u})^2_{|T}\mathrm{d}\Omega = \frac{1}{2}E^2_{CRE} \tag{16}$$

with $\hat{\boldsymbol{\varphi}}^* = \frac{1}{2}[\hat{\boldsymbol{\varphi}} + k\boldsymbol{\nabla}\hat{u}]$. The value of E_{CRE} can be used as an estimate of the global error between u_{ex} and u_m.

<u>Remark</u> : *Again, in the steady-state case, we would consider :*

$$E^2_{CRE}(\mathbf{p}) = \frac{1}{2}\int_\Omega \frac{1}{k}[\hat{\boldsymbol{\varphi}} - k\boldsymbol{\nabla}\hat{u}] \cdot [\hat{\boldsymbol{\varphi}} - k\boldsymbol{\nabla}\hat{u}]\mathrm{d}\Omega \equiv \frac{1}{2}||\hat{\boldsymbol{\varphi}} - k\boldsymbol{\nabla}\hat{u}||^2 \tag{17}$$

$$||\boldsymbol{\varphi}_{ex} - \hat{\boldsymbol{\varphi}}^*||^2 = \frac{1}{2}E^2_{CRE} \tag{18}$$

4.2   Construction of an admissible solution

We now explain how an admissible pair $(\hat{u}_m, \hat{\boldsymbol{\varphi}}_m)$ can be obtained as a post-processing of all information available from the computation of the PGD solution u_m. Constructing the kinematically admissible field $\hat{u}_m(\mathbf{x}, t, \mathbf{p})$ is rather simple, and one usually takes $\hat{u}_m = u_m$. Getting $\hat{\boldsymbol{\varphi}}_m(\mathbf{x}, t, \mathbf{p})$ is more difficult and technical. In order to use classical tools that enable to compute equilibrated fluxes (in particular the prolongation condition, see [8, 13]), one should first construct a field $\boldsymbol{\varphi}_m(\mathbf{x}, t, \mathbf{p})$ which satisfies the following FE equilibrium for all $(t, \mathbf{p}) \in \mathcal{I} \times \Theta$:

$$\int_\Omega \boldsymbol{\varphi}_m \cdot \boldsymbol{\nabla}u^*\mathrm{d}\Omega = \int_\Omega (f_d - c\frac{\partial\hat{u}_m}{\partial t})u^*\mathrm{d}\Omega - \int_{\partial_q\Omega} r_d u^*\mathrm{d}S \quad \forall u^* \in \mathcal{V}_h \tag{19}$$

For the sake of simplicity, let us suppose that the loading can be written under the radial form :

$$(f_d(\mathbf{x}, t), r_d(\mathbf{x}, t)) = \sum_{j=1}^J \alpha_j(t)\left(f_d^j(\mathbf{x}), r_d^j(\mathbf{x})\right) \tag{20}$$

We thus compute, for each couple (f_d^j, r_d^j), a field $\boldsymbol{\varphi}_d^j(\mathbf{x})$ verifying the FE equilibrium :

$$\int_\Omega \boldsymbol{\varphi}_d^j \cdot \boldsymbol{\nabla}u^*\mathrm{d}\Omega = \int_\Omega f_d^j u^*\mathrm{d}\Omega - \int_{\partial_q\Omega} r_d^j u^*\mathrm{d}S \quad \forall u^* \in \mathcal{V}_h \tag{21}$$

It is in practice obtained using the finite element method in displacement, i.e. by searching $v^j \in \mathcal{V}_h$ with $\boldsymbol{\varphi}_d^j = \boldsymbol{\nabla}v^j$. It follows that $\boldsymbol{\varphi}_d = \sum_{j=1}^J \alpha_j(t)\boldsymbol{\varphi}_d^j(\mathbf{x})$ can be introduced in the calculation of $\boldsymbol{\varphi}_m$, which should then verify for all $(t, \mathbf{p}) \in \mathcal{I} \times \Theta$:

$$\int_\Omega (\boldsymbol{\varphi}_m - \boldsymbol{\varphi}_d) \cdot \boldsymbol{\nabla}u^*\mathrm{d}\Omega = -\int_\Omega c\frac{\partial\hat{u}_m}{\partial t}u^*\mathrm{d}\Omega = -\sum_{i=1}^m c\dot{\lambda}_i\Gamma_i\int_\Omega \psi_i u^*\mathrm{d}\Omega \quad \forall u^* \in \mathcal{V}_h \tag{22}$$

Noticing that at the end of sub-iterations to compute each PGD mode $m_0 \in [1, m]$, condition (14) yields :

$$B(u_{m_0}, \psi^* \lambda_{m_0} \Gamma_{m_0}) = L(\psi^* \lambda_{m_0} \Gamma_{m_0}) \quad \forall \psi^* \in \mathcal{V}_h \tag{23}$$

We thus get :

$$\int_\Omega \left[\iint_\Theta \int_\mathcal{I} \lambda_{m_0} \Gamma_{m_0} (k \boldsymbol{\nabla} u_{m_0} - \boldsymbol{\varphi}_d) \mathrm{dt d\mathbf{p}} \right] \boldsymbol{\nabla} \psi^* \mathrm{d}\Omega$$

$$= -\int_\Omega \left[\iint_\Theta \int_\mathcal{I} c \lambda_{m_0} \Gamma_{m_0} \frac{\partial u_{m_0}}{\partial t} \mathrm{dt d\mathbf{p}} \right] \psi^* \mathrm{d}\Omega \quad \forall \psi^* \in \mathcal{V}_h \tag{24}$$

$$= -\int_\Omega \sum_{i=1}^{m_0} \left[\iint_\Theta \int_\mathcal{I} c \lambda_{m_0} \Gamma_{m_0} \dot{\lambda}_i \Gamma_i \mathrm{dt d\mathbf{p}} \right] \psi_i \psi^* \mathrm{d}\Omega \quad \forall \psi^* \in \mathcal{V}_h$$

It follows that for $m_0 \in [1, m]$, term :

$$\mathbf{Q}_{m_0} \equiv \iint_\Theta \int_\mathcal{I} \lambda_{m_0} \Gamma_{m_0} (\boldsymbol{\varphi}_d - k \boldsymbol{\nabla} u_{m_0}) \mathrm{dt d\mathbf{p}} \tag{25}$$

equilibrates $\sum_{i=1}^{m_0} \left[\iint_\Theta \int_\mathcal{I} c \lambda_{m_0} \Gamma_{m_0} \dot{\lambda}_i \Gamma_i \mathrm{dt d\mathbf{p}} \right] \psi_i$ in a FE sense. By a simple inversion of the

system, one obtains that a term of the form $\sum_{j=1}^{m} R_{ij} \mathbf{Q}_j$ equilibrates ψ_i in the FE sense

$(i = 1, \ldots, m)$. Consequently,

$$\boldsymbol{\varphi}_m = \boldsymbol{\varphi}_d - \sum_{i=1}^{m} \sum_{j=1}^{m} c \dot{\lambda}_i \Gamma_i R_{ij} \mathbf{Q}_j \tag{26}$$

satisfies FE equilibration (19) (or (22)).

Then, usual techniques [8, 13] can be used to build, from $\boldsymbol{\varphi}_m$, a flux $\hat{\boldsymbol{\varphi}}_m$ that verifies the full equilibrium :

$$\int_\Omega \hat{\boldsymbol{\varphi}}_m \cdot \boldsymbol{\nabla} u^* \mathrm{d}\Omega = \int_\Omega (f_d - c \frac{\partial \hat{u}_m}{\partial t}) u^* \mathrm{d}\Omega - \int_{\partial_q \Omega} r_d u^* \mathrm{d}S \quad \forall u^* \in \mathcal{V} \tag{27}$$

This flux reads $\hat{\boldsymbol{\varphi}}_m = \hat{\boldsymbol{\varphi}}_d - \sum_{i=1}^{m} \sum_{j=1}^{m} c \dot{\lambda}_i \Gamma_i R_{ij} \hat{\mathbf{Q}}_j$ where $\hat{\boldsymbol{\varphi}}_d$ and $\hat{\mathbf{Q}}_j$ are computed solving local problems on each element or patch of elements.

<u>Remark</u> : *for the steady-state case, (23) reads :*

$$B(u_{m_0}, \psi^* \Gamma_{m_0}) = L(\psi^* \Gamma_{m_0}) \quad \forall \psi^* \in \mathcal{V}_h \tag{28}$$

and we get :

$$\int_\Omega \left[\int_\Theta \Gamma_{m_0}(k\boldsymbol{\nabla} u_{m_0} - \boldsymbol{\varphi}_d) d\mathbf{p} \right] \boldsymbol{\nabla}\psi^* d\Omega = 0 \quad \forall \psi^* \in \mathcal{V}_h \tag{29}$$

Therefore, $\mathbf{Q}_{m_0} \equiv \int_\Theta \Gamma_{m_0}(\boldsymbol{\varphi}_d - k\boldsymbol{\nabla} u_{m_0}) d\mathbf{p}$ *is auto-equilibrated (in a FE sense), and* $\boldsymbol{\varphi}_m$ *and* $\hat{\boldsymbol{\varphi}}_m$ *can be defined as :*

$$\boldsymbol{\varphi}_m(\mathbf{x}, \mathbf{p}) = \boldsymbol{\varphi}_d(\mathbf{x}) + \sum_{m_0=1}^m \beta_{m_0}(\mathbf{p})\mathbf{Q}_{m_0}(\mathbf{x}) \quad ; \quad \hat{\boldsymbol{\varphi}}_m(\mathbf{x}, \mathbf{p}) = \hat{\boldsymbol{\varphi}}_d(\mathbf{x}) + \sum_{m_0=1}^m \beta_{m_0}(\mathbf{p})\hat{\mathbf{Q}}_{m_0}(\mathbf{x})$$

$$\tag{30}$$

where β_{m_0} *are coefficients, depending on* $\mathbf{p}$, *which are explicitly obtained by minimizing* $\int_\Theta E_{CRE}^2(\mathbf{p}) d\mathbf{p}$.

5 GOAL-ORIENTED ERROR ESTIMATION

5.1 Guaranteed bounding

Let I be an output of interest defined by the extraction pair $(\boldsymbol{\varphi}_\Sigma, \mathbf{f}_\Sigma)$:

$$I(\mathbf{p}) = \int_\mathcal{I} \int_\Omega \{\boldsymbol{\varphi}_\Sigma \cdot \boldsymbol{\nabla} u + \mathbf{f}_\Sigma \cdot u\} \, d\Omega dt \tag{31}$$

$\boldsymbol{\varphi}_\Sigma(\mathbf{x}, t)$ and $\mathbf{f}_\Sigma(\mathbf{x}, t)$ can possibly be Dirac distributions. We therefore introduce the associated adjoint problem, and compute an approximate (resp. admissible) PGD solution $(\tilde{u}_{\tilde{m}}, \tilde{\boldsymbol{\varphi}}_{\tilde{m}})$ (resp. $(\hat{\tilde{u}}_{\tilde{m}}, \hat{\tilde{\boldsymbol{\varphi}}}_{\tilde{m}})$) for this problem. In practice, the PGD solution of the adjoint problem is performed using an order $\tilde{m}$ possibly different from m, as well as introducing local enrichment functions in the vicinity of the space-time region where I is defined [14]. The fundamental result for linear elliptic and parabolic problems then reads (see [9]) :

$$|I_{ex} - I_m - I_{corr}| \leq E_{CRE}\tilde{E}_{CRE} \tag{32}$$

where $I_{ex}(\mathbf{p})$ (resp. $I_m(\mathbf{p})$) is the exact (resp. approximated by PGD) value of the output of interest, $I_{corr}(\mathbf{p})$ is a correction term computed from approximate solutions of both reference and adjoint problems, and $E_{CRE}(\mathbf{p})$ (resp. $\tilde{E}_{CRE}(\mathbf{p})$) is the constitutive relation error of the reference (resp. adjoint) problem. Let us note that this bounding result does not use any Galerkin orthogonality property, but only properties of admissible solutions.

Consequently, strict bounds on the local error $I_{ex} - I_m$ (or directly on I_{ex}) can be obtained for any value $\mathbf{p}$ of material parameters.

5.2 Specific indicators on error sources

In the problem we consider, the error $I_{ex} - I_m$ comes from two main sources : (i) the truncation of the sum in the PGD representation (10) at a given order m ; (ii) discretizations used to compute modes. Indeed, the error reads :

$$I_{ex} - I_m = (I_{ex} - I_{dis}) + (I_{dis} - I_m) = \mathcal{E}_{dis} + \mathcal{E}_{PGD} \tag{33}$$

where I_{dis} is the value of the output of interest obtained after discretization of the reference problem inside all parameter domains, and $\mathcal{E}_{dis}$ (resp. $\mathcal{E}_{PGD}$) is part of the error due to discretization (resp. to truncation of the PGD representation). Furthermore, error due to discretization can also be split between contributions coming from the discretization of each parameter domain (space, time,...).
In order to control the computation process and lead adaptive strategies, we thus introduce an indicator for each error source. This is performed using verification tools presented previously, but considering additional (intermediate) reference problems obtained from a partial numerical method applied to the initial reference problem. For instance, assessing part $\mathcal{E}_{PGD}$ of the error only due to truncation in the PGD representation can be done considering the discretized problem (i.e. the one providing for I_{dis}) as the reference problem. It is of the form :

$$\mathbf{U}_h^1 = \mathbf{0} \quad ; \quad \mathbb{M}\frac{\mathbf{U}_h^{p+1} - \mathbf{U}_h^p}{\Delta t} + \mathbb{K}\mathbf{U}_h^p = \mathbf{F}_h^p \quad \forall p \in [1, P-1] \tag{34}$$

where Δt is the time step size, P is the number of time steps, whereas $\mathbb{M}$ and $\mathbb{K}$ are classical matrices deriving from the FEM. Admissible pairs are then defined with respect to the new reference problems, and are in practice computed as a direct post-processing of available information.

6 NUMERICAL RESULTS

As a simple example, we consider the 2D structure of Figure 2 which is a section presenting two rectangular holes in which a fluid circulates. It is subjected to a given source term $f_d(x, y) = 200xy$, a given flux $r_d(t) = -1$ on the holes boundaries, and a zero temperature is imposed on other boundaries. Using symmetries, we only keep the upper right quarter of the structure that we denote Ω.
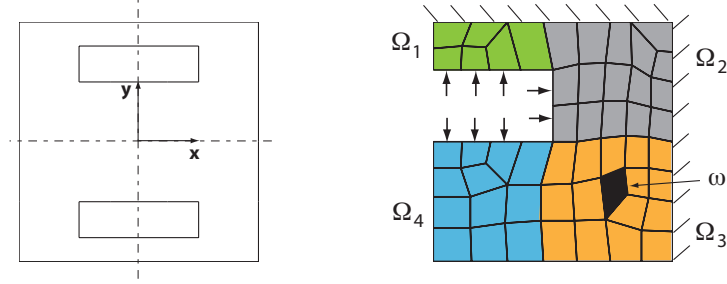
Figure 2: The 2D reference problem.

We consider that the diffusion coefficient k is fluctuating but remains piecewise homogeneous, i.e. it is homogeneous in each of the four non-overlapping subdomains Ω_i ($i = 1, 2, 3, 4$) defined in Figure 2 and such that $\overline{\Omega_1 \cup \Omega_2 \cup \Omega_3 \cup \Omega_4} = \Omega$. Furthermore, the thermal capacity c is supposed homogeneous in the whole domain Ω. In the following, the two material coefficients are defined as :

$$k(\mathbf{x}, \theta_i) = 1 + \sum_{i=1}^{4} g_i I_{\Omega_i}(\mathbf{x})\theta_i \qquad c(\mathbf{x}, \theta_5) = 1 + 0.2\,\theta_5 \qquad (35)$$

with $[g_1, g_2, g_3, g_4] = [0.1, 0.1, 0.2, 0.05]$, $\theta_i \in [-2, 2]$, and $I_{\Omega_i}(\mathbf{x})$ denoting the indicatrix function of subdomain Ω_i.

The resulting solution $u(\mathbf{x}, t, \theta_1, \theta_2, \theta_3, \theta_4, \theta_5)$ is searched using the PGD technique, with four-nodes quadrangular elements in space and a forward Euler time scheme.

Figure 3 gives the evolution of the constitutive relation error (seen as a global error estimate) with respect to the number m of PGD modes taken in the approximation; this estimate is computed for $\theta_i = 0$ ($i = 1, \ldots, 5$). We observe that after 5 modes, the error reaches an asymptotic value that corresponds to the discretization error.
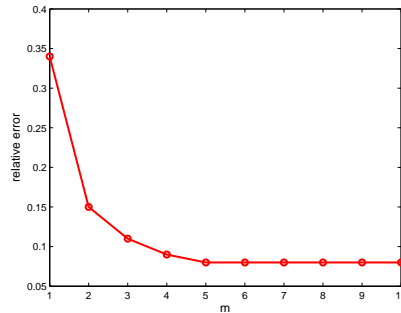
Figure 3: Relative error estimate with respect to the number m of PGD modes.

Assuming that θ_i are (truncated) centered reduced normal variables, and considering a given zone $\omega \subset \Omega$ (see Figure 2), we study two quantities of interest :

– the mathematical expectation (in the probabilistic sense) of the mean value of u inside ω at final time T :

$$I_1 = \mathrm{E}\left[\frac{1}{|\omega|}\int_\omega u_{|T}\mathrm{d}\Omega\right] \tag{36}$$

– the maximal value of the mean value of u inside ω at final time T :

$$I_2 = \sup_{\theta_i}\frac{1}{|\omega|}\int_\omega u_{|T}\mathrm{d}\Omega \tag{37}$$

We choose an order $m = 3$ for the approximate PGD solution of the reference problem. For both I_1 and I_2, obtained normalized upper bounds on $I_{ex} - I_m - I_{corr}$ as well as specific error indicators are given in Figure 4 with respect to the number $\tilde{m}$ of computed PGD modes for the adjoint solution.
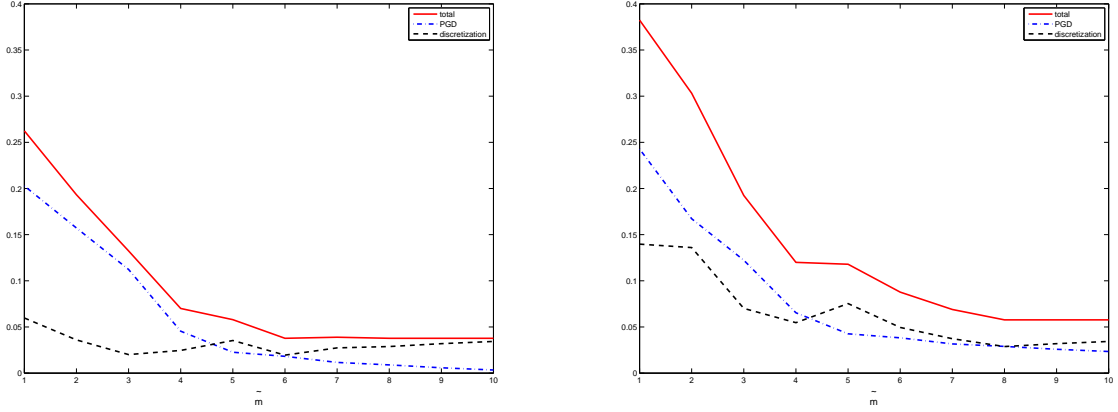
FIGURE 4: Normalized upper error bound and error indicators with respect to the number $\tilde{m}$ of PGD modes used for the adjoint solution : I_1 (left), I_2 (right).

We observe that the correcting term I_{corr} enables to assess I_{ex} very effectively. The remaining asymptotic error could be decreased by improving discretizations used to compute PGD modes.

7 CONCLUSIONS

PGD-reduced models are a promising tool for solving complex engineering problems. However, a central and main question is to guarantee their accuracy. The verification method described here is a first attempt to address this challenge for elliptic and parabolic problems. It can be applied in the case of numerous parameters (such as stochastic problems), even though optimizations when performing numerical integrations should be investigated in that case to decrease computational effort. It can also be directly extended to cases where the loading is defined with parameters.

REFERENCES

[1] Chinesta, F., Ladevèze, P. and Cueto, E. A short review on model reduction based on Proper Generalized Decomposition. *Archives of Computational Methods in Engineering* (2011) **18**:395–404.

[2] Nouy, A. A generalized spectral decomposition technique to solve a class of linear stochastic partial differential equations. *Computer Methods in Applied Mechanics and Engineering* (2007) **196**(45-48):4521–4537.

[3] Nouy, A. A priori model reduction through Proper Generalized Decomposition for solving time-dependent partial differential equations. *Computer Methods in Applied Mechanics and Engineering* (2010) **199**(23-24):1603–1626.

[4] Chinesta, F., Ammar, A. and Cueto, E. Recent advances and new challenges in the use of the Proper Generalized Decomposition for solving multidimensional models. *Archives of Computational Methods in Engineering* (2010) **17**(4):327–350.

[5] Niroomandi, S., Alfaro, I. and Cueto, E. Accounting for large deformations in real-time simulations of soft tissues based on reduced-order models, *Computer Methods and Programs in Biomedicine* (2012) **105**(1):1–12.

[6] Ladevèze, P. *Nonlinear Computational Structural Mechanics: New Approaches and Non-Incremental Methods of Calculation.* Springer (1998).

[7] Ammar, A., Chinesta, F., Diez, P. and Huerta, A. An error estimator for separated representations of highly multidimensional models. *Computer Methods in Applied Mechanics and Engineering* (2010) **199**:1872–1880.

[8] Ladevèze, P. and Pelle, J.P. *Mastering Calculations in Linear and Nonlinear Mechanics.* Springer NY (2004).

[9] Ladevèze, P. Strict upper error bounds for calculated outputs of interest in computational structural mechanics. *Computational Mechanics* (2008) **42**(2):271–286.

[10] Ladevèze, P. and Chamoin, L. On the verification of model reduction methods based on the Proper Generalized Decomposition. *Computer Methods in Applied Mechanics and Engineering* (2011) **200**:2032–2047.

[11] Chamoin, L, Ladevèze, P. Robust control of PGD-based numerical simulations. *European Journal of Computational Mechanics* (2012) **21**(3-6):195–207.

[12] Ladevèze, P. and Chamoin, L. *Toward guaranteed PGD-reduced models.* Bytes and Science, G. Zavarise & D.P. Boso (Eds.), CIMNE (2013).

[13] Pled, F., Chamoin, L. and Ladevèze, P. On the techniques for constructing admissible stress fields in model verification: performances on engineering examples. *International Journal for Numerical Methods in Engineering* (2011) **88**(5):409–441.

[14] Ladevèze, P. and Chamoin, L. Calculation of strict bounds for finite element approximations of non-linear pointwise quantities of interest. *International Journal for Numerical Methods in Engineering* (2010) **84**:1638–1664.

ADAPTIVE MESHLESS ANALYSIS OF THIN SHELLS WITH THE AID OF THE INTERIOR PENALTY METHOD

JORGE C. COSTA AND PAULO M. PIMENTA

Polytechnic School at University of São Paulo
e-mail: jorgecosta@usp.br; ppimenta@usp.br

Abstract. Meshless approximations provide a great resource in the analysis of structures as the desired continuity in the approximated fields can be achieved. This feature is well suited for thin structures like shells, as stresses can be obtained as smoothly as desired. However, the non-interpolatory characteristic of such approximants makes the imposition of essential boundary and interface conditions not straightforward. For instance, in the classical Element-Free Galerkin Method (EFG), Lagrange multipliers are used to enforce such conditions. Recently, an alternative has been revisited: the Interior Penalty Method, usually referred to as Nietsche's Method, which identifies the Lagrange multiplier with the flux at the essential or interface boundary and introduces a penalty parameter, which warrants the convergence rate of the approximation. In the elasticity case, the reaction tractions are the same as the stress normal to the boundary. The present work aims at developing the study of this method in the linear elastic analysis of shells, firstly for the imposition of boundary displacements and latter for multi-region problems. In the former case, its advantage over Lagrange Multipliers is that no additional degrees of freedom are introduced and there is no need to introduce a new approximation space (which would have to obey an inf-sup condition), in the latter, refinement over one portion of the domain can be performed without affecting other regions, even maintaining their stiffness matrices, and different regions can be discretized with different approximants, *e.g.*, finite elements.

DEVELOPMENT OF A 3D NAVIER-STOKES DG SOLVER FOR ADAPTIVE SCHEME AND MODELLING

VINCENT COUAILLIER, FLORENT RENAC, MARTA DE LA LLAVE PLATA EMERIC MARTIN, JEAN-BAPTISTE CHAPELIER, MARIE-CLAIRE LE PAPE

Onera The French Aerospace Lab
F-92322 Châtillon, France
vincent.couaillier@onera.fr

Keywords. Discontinuous Galerkin method, hybrid mesh, turbulent flow, High-order discretization methods

Abstract. Over the years, the development of new and increasingly powerful CFD simulation tools has helped manufacturers in the aerospace industry gain a greater understanding of the operating performance of their products. This has allowed them to progress through the design life cycle in a more timely and cost-effective manner by supplementing or replacing experimental testing with CFD computations.

The industrial demand for CFD predictions at an ever-increasing level of detail is the driving force for the development of highly accurate simulation techniques able to predict not only overall flow features, but also local values of the quantities of interest. This will allow engineers to expand the range of flow conditions to which CFD can be applied.

Nevertheless, most of the industrial CFD codes used today are based on second-order methods, which appear not to be sufficiently accurate to reach these goals. With the aim of overcoming the limitations of second-order approaches, Onera has launched the development of a DG solver called Aghora [1], [2]. The main goal is to develop a new demonstrator able to integrate efficient high-order schemes based on Discontinuous Galerkin methods using hybrid type meshes (tetrahedral, hexahedra, prisms and pyramids) for the simulation of turbulent flows using different levels of modelling, i.e. RANS [2], LES, hybrid RANS/LES and DNS. Adaptive techniques based on local HPM methods (H for grid, P for accuracy of shape function, M for model) will be used in order to represent accurately the flow physics.

However, these methods require the solution of very large discrete systems. This leads to long execution times and high memory requirements. Consequently, in order to tackle such challenges, the project focuses on the implementation of efficient algorithms for modern multi-core architectures with highly-scalable parallel strategies. The paper will present the status of the modal DG schemes implemented in Aghora as well as representative test cases illustrating the adaptability capacity of DG methods.

In order to illustrate the interest of DG approach with high order polynomial degrees, a convergence analysis in terms of number of Degrees of Freedom (DOFs) is presented hereafter. The calculations are performed with the compressible Navier-Stokes equations for the simulation of a manufactured solution of a Poiseuille flow (M=0.1). Fig. 1 shows the evolution of the L^2 norm of the error vs. the characteristic size of the elements h, represented here by the Degrees of Freedom. The slopes obtained by the computations, compared to the theoretical slopes,

demonstrate the effective accuracy of the implemented Navier-Stokes DG scheme in Aghora.
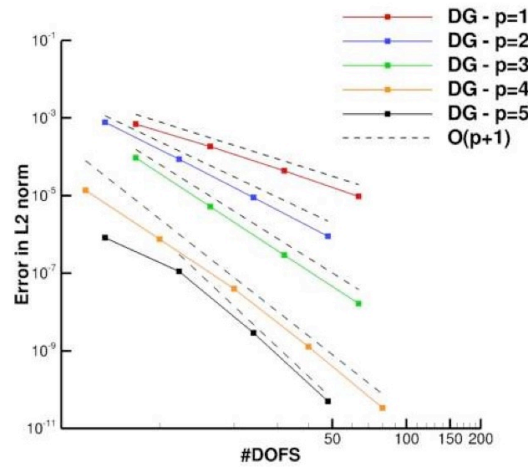
Figure 1: Convergence analysis in DoFs – Manufactured solutions for Navier-Stokes equations Aghora (laminar flow) - L2-norm on the error between numerical and exact solutions

Mesh convergence analysis (h) and polynomial degree convergence analysis (p) for turbulent flow computations will be presented. Fig. 2 illustrates the solution of a transonic flow around the Onera M6 wing for a DGP1 computation performed with the Kω Wilcox model.
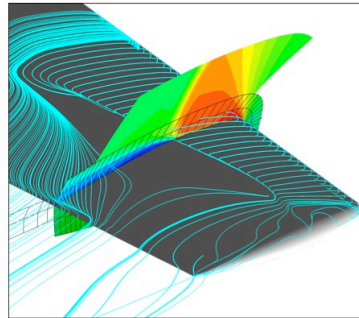
Figure 2: Turbulent flow around the Onera M6 wing – DGP1 computation with a RANS/Kω model

REFERENCES

[1] F. Bassi and S. Rebay, Numerical evaluation of two discontinuous Galerkin methods for the compressible Navier-Stokes equations, Int. J. Num.. Methods Fluids 40 (2002), pp. 197– 202.
[2] F. Renac, C. Marmignon and F. Coquel, Time implicit high-order discontinuous Galerkin method with reduced evaluation cost, to appear in SIAM J. Sci. Comput.
[3] F. Renac, C. Marmignon and F. Coquel, Fast time implicit-explicit discontinuous Galerkin method for convection dominated flow problems, Math. Sci., 10 (2012), pp.~1161--1172.
[4] V. Couaillier, F. Renac, S. Gerald, E. Martin, M. de la Llave Plata, Discontinuous Galerkin method on hybrid meshes for internal and external flow configurations, ECCOMAS 2012, September 10-14, 2012, Vienna, Austria
[5] F. Renac, A moment limiter for a high-order Runge-Kutta discontinuous Galerkin method on unstructured meshes, ECCOMAS 2012, September 10-14, 2012, Vienna, Austria

VI International Conference on Adaptive Modeling and Simulation
ADMOS 2013
J. P. Moitinho de Almeida, P. Díez, C. Tiago and N. Parés (Eds)

ANISOTROPIC A-PRIORI ERROR ESTIMATES ON SURFACES

F. Dassi, S. Perotto and L. Formaggia*

* Politecnico di Milano, MOX, Department of Mathematics "F.Brioschi",
Piazza Leonardo da Vinci, 32, 20133, Milano, ITALY
franco.dassi@mail.polimi.it
{simona.perotto,luca.formaggia}@polimi.it

Key words: anisotropic meshes, a-priori error estimators, finite elements

Abstract.

A lot of practical problems are related to the resolution of Partial Differential Equations (PDEs) defined on surfaces embedded in a three dimensional space. In such cases the classical differential operators have to be suitably modified to recover tangential information (see [1]); likewise, the derivation of error estimators is usually not a trivial task, essentially due to the fact that these estimators should include the error due to the finite element approximation as well as to the fitting of the computational domain (see [3]).

Moving from the theory proved in [2], we propose an *anisotropic a-priori error estimator* to control the L^2-norm of the interpolation error associated with linear finite elements defined on surfaces. This new error estimator consists of two different contributions:

- an *almost best-approximation term*, typical of a finite element discretization;

- a *geometric error term*, related to the discretization of the surface.

Moving from to this estimator, we settle a metric-based anisotropic mesh adaptation procedure which essentially employs local operations (node smoothing, edge collapsing, edge splitting, edge swapping) to adapt the mesh. Since an anisotropic estimator takes into account the directional features of the solution at hand, we obtain adapted meshes whose elements are suitably oriented to match the intrinsic directionality of the function defined on the surface, and of the surface itself.

As expected the employment of anisotropic meshes leads to a remarkable improvement of the mesh adaptation procedure in terms of computational costs.

REFERENCES

[1] Dziuk G. Finite elements for the Beltrami operator on arbitrary surfaces. *Partial Differential Equation Calc. Var.*, (1988), **1357**:142–155.

[2] Formaggia L. and Perotto S. New anisotropic a priori error estimates. *Numer. Math.*, (2001), **89**:641–667.

[3] Mordin P. and Nochetto R. AFEM for the Laplace-Beltrami operator on graphs: design and conditional contraction property. *Math. Comp.*, (2011) **80**:625–648.

ON GLOBAL ERROR ESTIMATION AND CONTROL OF FINITE DIFFERENCE SOLUTIONS FOR PARABOLIC EQUATIONS

KRISTIAN DEBRABANT[*] AND JENS LANG[†]

[*]University of Southern Denmark
Department of Mathematics and Computer Science
Campusvej 55, 5230 Odense M, Denmark
e-mail: debrabant@imada.sdu.dk, www.imada.sdu.dk/~debrabant

[†]Technische Universität Darmstadt
Department of Mathematics
Dolivostr. 15, 64293 Darmstadt, Germany
e-mail: lang@mathematik.tu-darmstadt.de, www.mathematik.tu-darmstadt.de/~lang/

Key words: Numerical integration of PDEs, method of lines, global error estimation, global error control, defects and local errors, tolerance proportionality

Abstract. The aim of this paper is to extend the global error estimation and control addressed in Lang and Verwer [SIAM J. Sci. Comput. 29, 2007] for initial value problems to finite difference solutions of parabolic partial differential equations. The classical ODE approach based on the first variational equation is combined with an estimation of the PDE spatial truncation error to estimate the overall error in the computed solution. Control in a discrete L_2-norm is achieved through tolerance proportionality and mesh refinement. A numerical example illustrates the reliability of the estimation and control strategies.

1 Introduction

We consider initial boundary value problems of parabolic type, which can be written as

$$\partial_t u(t,x) = f(t,x,u(t,x),\partial_x u(t,x),\partial_{xx} u(t,x))\,, \quad t \in (0,T]\,,\ x \in \Omega \subset \mathbb{R}^d\,, \qquad (1)$$

equipped with an appropriate system of boundary conditions and with the initial condition

$$u(0,x) = u_0(x)\,, \quad x \in \overline{\Omega}. \qquad (2)$$

The PDE is assumed to be well posed and to have a unique continuous solution $u(t,x)$ which has sufficient regularity.

The method of lines is used to solve (1) numerically. We first discretize the PDE in space by means of finite differences on a (possibly non-uniform) spatial mesh Ω_h and solve the resulting system of ODEs using existing time integrators. For simplicity, we shall assume that this system of time-dependent ODEs can be written in the general form

$$
\begin{aligned}
U_h'(t) &= F_h(t, U_h(t)), \qquad t \in (0, T], \\
U_h(0) &= U_{h,0},
\end{aligned}
\tag{3}
$$

with a unique solution vector $U_h(t)$ being a grid function on Ω_h. Let

$$
R_h : u(t, \cdot) \to R_h u(t)
\tag{4}
$$

be the usual restriction operator defined by $R_h u(t) = (u(t, x_1), \ldots, u(t, x_N))^T$, where $x_i \in \Omega_h$ and N is the number of all mesh points. Then we take as initial condition $U_{h,0} = R_h u(0)$.

To solve the initial value problem (3), we apply a numerical integration method at a certain time grid

$$
0 = t_0 < t_1 < \cdots < t_n < \cdots < t_{M-1} < t_M = T,
\tag{5}
$$

using local control of accuracy. This yields approximations $V_h(t_n)$ to $U_h(t_n)$, which may be calculated for other values of t by using a suitable interpolation method provided by the integrator. The global time error is then defined by

$$
e_h(t) = V_h(t) - U_h(t).
\tag{6}
$$

Numerical experiments in [5] for ODE systems have shown that classical global error estimation based on the first variational equation is remarkably reliable. In addition, having the property of tolerance proportionality, that is, there exists a linear relationship between the global time error and the local accuracy tolerance, $e_h(t)$ can be successfully controlled by a second run with an adjusted local tolerance. Numerous techniques to estimate global errors are described in [9].

In order for the method of lines to be used efficiently, it is necessary to take also into account the spatial discretization error. Defining the spatial discretization error by

$$
\eta_h(t) = U_h(t) - R_h u(t),
\tag{7}
$$

the vector of overall global errors $E_h(t) = V_h(t) - R_h u(t)$ may be written as sum of the global time and spatial error, that is,

$$
E_h(t) = e_h(t) + \eta_h(t).
\tag{8}
$$

It is the purpose of this paper to present a new error control strategy for the global errors $E_h(t)$. We will mainly focus on reliability. So our aim is to provide error estimates

$\tilde{E}_h(t) \approx E_h(t)$ which are not only asymptotically exact, but also work reliably for moderate tolerances, that is for relatively coarse discretizations.

The global errors are measured in discrete L_2-norms. A priori bounds for the global error in such norms are well known, see e.g. [6, 10]. However, reliable a posteriori error estimation and efficient control of the accuracy of the solution numerically computed to an imposed tolerance level are still challenging. We achieve global error control by iteratively improving the temporal and spatial discretizations according to estimates of $e_h(t)$ and $\eta_h(t)$. The global time error is estimated and controlled along the way fully described in [5]. To estimate the global spatial error, we follow an approach proposed in [1] (see also [7]) and use Richardson extrapolation to set up a linearised error transport equation.

2 Spatial and time error

By making use of the restriction operator R_h, the spatial truncation error is defined by

$$\alpha_h(t) = (R_h u)'(t) - F_h(t, R_h u(t)). \tag{9}$$

From (3) and (9), it follows that the global spatial error $\eta_h(t)$ representing the accumulation of the spatial discretization error is the solution of the initial value problem

$$\begin{aligned} \eta_h'(t) &= F_h(t, U_h(t)) - F_h(t, R_h u(t)) - \alpha_h(t), & t \in (0, T], \\ \eta_h(0) &= 0. \end{aligned} \tag{10}$$

Assuming F_h to be continuously differentiable, the mean value theorem for vector functions yields

$$\begin{aligned} \eta_h'(t) &= \partial_{U_h} F_h(t, U_h(t))\, \eta_h(t) - \alpha_h(t) + \mathcal{O}(\eta_h(t)^2), & t \in (0, T], \\ \eta_h(0) &= 0. \end{aligned} \tag{11}$$

With $V_h(t)$ being the continuous extension of the numerical approximation to (3), the residual time error is defined by

$$r_h(t) = V_h'(t) - F_h(t, V_h(t)). \tag{12}$$

Thus the global time error $e_h(t)$ fulfills the initial value problem

$$\begin{aligned} e_h'(t) &= F_h(t, V_h(t)) - F_h(t, U_h(t)) + r_h(t), & t \in (0, T], \\ e_h(0) &= 0. \end{aligned} \tag{13}$$

Again, the mean value theorem yields

$$\begin{aligned} e_h'(t) &= \partial_{U_h} F_h(t, V_h(t))\, e_h(t) + r_h(t) + \mathcal{O}(e_h(t)^2), & t \in (0, T], \\ e_h(0) &= 0. \end{aligned} \tag{14}$$

Apparently, by implementing proper choices of the defects $\alpha_h(t)$ and $r_h(t)$, solving (11) and (14) will in leading order provide approximations to the true global error. The issue of how to approximate the spatial truncation error and the residual time error will be discussed in the next sections.

3   Estimation of the residual time error

We assume that the time integration method used to approximate the general ODE system (3) is of order $p \leq 3$. Following the approach proposed in [5] we define the interpolated solution $V_h(t)$ by piecewise cubic Hermite interpolation. Let $V_{h,n} = V_h(t_n)$ and $F_{h,n} = F_h(t_n, V_{h,n})$ for all $n = 0, 1, \ldots, M$. Then at every subinterval $[t_n, t_{n+1}]$ we form

$$V_h(t) = V_{h,n} + A_n(t - t_n) + B_n(t - t_n)^2 + C_n(t - t_n)^3, \qquad t_n \leq t \leq t_{n+1}, \quad (15)$$

and choose the coefficients such that $V_h'(t_n) = F_{h,n}$ and $V_h'(t_{n+1}) = F_{h,n+1}$. This gives

$$V_h(t_n + \theta\tau_n) = v_0(\theta)V_{h,n} + v_1(\theta)V_{h,n+1} + \tau_n w_0(\theta)F_{h,n} + \tau_n w_1(\theta)F_{h,n+1} \quad (16)$$

with $0 \leq \theta \leq 1$, $\tau_n = t_{n+1} - t_n$, and

$$v_0(\theta) = (1-\theta)^2(1+2\theta), \; v_1(\theta) = \theta^2(3-2\theta), \; w_0(\theta) = (1-\theta)^2\theta, \; w_1(\theta) = \theta^2(\theta-1). \quad (17)$$

Now let $Y_h(t)$ be the (sufficiently smooth) solution of the ODE (3) with initial value $Y(t_n) = V_{h,n}$. Then the local error of the time integration method at time t_{n+1} is given by

$$le_{n+1} = V_{h,n+1} - Y_h(t_{n+1}) = \mathcal{O}(\tau_n^{p+1}). \quad (18)$$

Combining (16) and (18) and applying a Taylor expansion gives

$$V_h(t_n + \theta\tau_n) - Y_h(t_n + \theta\tau_n) = v_1(\theta)le_{n+1} + \frac{1}{24}(2\theta^3 - \theta^2 - \theta^4)\tau_n^4 Y_h^{(4)}(t_n) + \mathcal{O}(\tau_n^{p+2}). \quad (19)$$

Recalling $Y_h'(t) = F_h(t, Y_h(t))$ for $t \in (t_n, t_{n+1}]$ and rewriting the residual time error as

$$r_h(t) = V_h'(t_n + \theta\tau_n) - Y_h'(t_n + \theta\tau_n) + F_h(t, Y_h(t)) - F_h(t, V_h(t)), \quad (20)$$

with $\theta = (t - t_n)/\tau_n$, we find by differentiating the right hand side of (19)

$$r_h(t_n + \theta\tau_n) = 6(\theta - \theta^2)\frac{le_{n+1}}{\tau_n} + \frac{1}{12}(3\theta^2 - \theta - 2\theta^3)\tau_n^3 Y_h^{(4)}(t_n) + \mathcal{O}(\tau_n^{p+1}). \quad (21)$$

Setting $\theta = 1/2$ in (21) will reveal

$$r_h(t_{n+1/2}) = \frac{3}{2}\frac{le_{n+1}}{\tau_n} + \mathcal{O}(\tau_n^{p+1}). \quad (22)$$

Thus the cubic Hermite defect halfway the step interval can be used to retrieve in leading order the local error of any one-step method of order $1 \leq p \leq 3$ (see also [5], Section 2.2). Following the arguments given in [5], Section 2.1, we consider instead of (14) the step size frozen version

$$
\begin{aligned}
\tilde{e}'_h(t) &= \partial_{U_h} F_h(t_n, V_{h,n})\, \tilde{e}_h(t) + \tfrac{2}{3} r_h(t_{n+\frac{1}{2}}),\ t \in (t_n, t_{n+1}],\ n = 0, \ldots, M-1, \\
\tilde{e}_h(0) &= 0
\end{aligned}
\tag{23}
$$

to approximate the global time error $e_h(t)$. Using

$$
V_h(t_{n+1/2}) = \frac{1}{2}(V_{h,n} + V_{h,n+1}) + \frac{\tau}{8}(F_{h,n} - F_{h,n+1})
\tag{24}
$$

and

$$
V'_h(t_{n+1/2}) = \frac{3}{2\tau}(V_{h,n+1} - V_{h,n}) - \frac{1}{4}(F_{h,n} + F_{h,n+1})
\tag{25}
$$

we can compute the residual time error halfway the step interval from (12)

$$
\begin{aligned}
r_h(t_{n+1/2}) &= \tfrac{3}{2\tau}(V_{h,n+1} - V_{h,n}) - \tfrac{1}{4}(F_{h,n} + F_{h,n+1}) \\
&\quad - F_h\left(t_{n+\frac{1}{2}}, \tfrac{1}{2}(V_{h,n} + V_{h,n+1}) + \tfrac{\tau}{8}(F_{h,n} - F_{h,n+1})\right).
\end{aligned}
\tag{26}
$$

Remark 3.1 From (21) we deduce

$$
\frac{1}{\tau_n} \int_{t_n}^{t_{n+1}} r_h(t)\, dt = \frac{le_{n+1}}{\tau_n} + \mathcal{O}(\tau_n^{p+1}),
\tag{27}
$$

showing, in the light of (22), that $\tfrac{2}{3} r_h(t_{n+1/2})$ is in leading order equal to the time-averaged residual. Thus, we can justify the use of the error equation (23) without the link to the first variational equation. $\diamond$

4 Estimation of the spatial truncation error

An efficient strategy to estimate the spatial truncation error by Richardson extrapolation is proposed in [1]. We will adopt this approach to our setting.

Suppose we are given a second semi-discretization of the PDE system (1), now with doubled local mesh sizes $2h$,

$$
\begin{aligned}
U'_{2h}(t) &= F_{2h}(t, U_{2h}(t)), & t &\in (0, T], \\
U_{2h}(0) &= U_{2h,0}.
\end{aligned}
\tag{28}
$$

In practice, one first chooses Ω_{2h} and constructs then Ω_h through uniform refinement. The following two assumptions will be needed. (i) The solution $U_{2h}(t)$ to the discretized PDE on the coarse mesh Ω_{2h} exists and is unique. (ii) The spatial discretization error

$\eta_h(t)$ is of order q with respect to h. We define the restriction operator R_{2h}^h from the fine grid Ω_h to the coarse grid Ω_{2h} by the identity $R_{2h} = R_{2h}^h R_h$ and set

$$\eta_h^c(t) = R_{2h}^h \eta_h(t), \quad U_h^c(t) = R_{2h}^h U_h(t), \quad V_h^c(t) = R_{2h}^h V_h(t). \tag{29}$$

From the second assumption it follows that

$$\eta_h^c(t) = 2^{-q} \eta_{2h}(t) + \mathcal{O}(h^{q+1}) \tag{30}$$

and therefore

$$R_{2h}u(t) = \frac{2^q}{2^q - 1} U_h^c(t) - \frac{1}{2^q - 1} U_{2h}(t) + \mathcal{O}(h^{q+1}). \tag{31}$$

The relation $U_h^c(t) - U_{2h}(t) = \eta_h^c(t) - \eta_{2h}(t)$ together with (30) gives

$$U_h^c(t) - U_{2h}(t) = \frac{1 - 2^q}{2^q} \eta_{2h}(t) + \mathcal{O}(h^{q+1}). \tag{32}$$

The spatial truncation error on the coarse mesh Ω_{2h} is analogously defined to (9) as

$$\alpha_{2h}(t) = (R_{2h}u)'(t) - F_{2h}(t, R_{2h}u(t)). \tag{33}$$

Substituting $R_{2h}u(t)$ from (31) into the right-hand side, using the ODE system (28) to replace $U_{2h}'(t)$, and manipulating the expressions with (32) we get after Taylor expansion

$$\alpha_{2h}(t) = \frac{2^q}{2^q - 1} \Big((U_h^c)'(t) - F_{2h}(t, U_h^c(t)) \Big) + \mathcal{O}(h^{q+1}). \tag{34}$$

Analogously to (6), we set $e_h^c(t) = V_h^c(t) - U_h^c(t)$. Substituting $(U_h^c)'(t)$ by $R_{2h}^h F_h(t, U_h(t))$ and using again Taylor expansion it follows that

$$\begin{aligned}
\alpha_{2h}(t) &= \frac{2^q}{2^q-1} \Big(R_{2h}^h F_h(t, V_h(t)) - F_{2h}(t, V_h^c(t)) \Big) + \mathcal{O}(h^{q+1}) \\
&\quad - \frac{2^q}{2^q-1} \Big(R_{2h}^h \big(\partial_{U_h} F_h(t, V_h(t)) e_h(t) \big) - \partial_{U_h} F_{2h}(t, V_h^c(t)) e_h^c(t) \Big) + \mathcal{O}(e_h(t)^2).
\end{aligned} \tag{35}$$

Assuming the term on the right-hand side involving the global time error to be sufficiently small, we can use

$$\tilde{\alpha}_{2h}(t) = \frac{2^q}{2^q - 1} \Big(R_{2h}^h F_h(t, V_h(t)) - F_{2h}(t, V_h^c(t)) \Big) \tag{36}$$

as approximation for the spatial truncation error on the coarse mesh. To guarantee a suitable quality of the estimate (36) we shall first control the global time error for attempting that afterwards the overall error is dominated by the spatial truncation error (see Section 6).

An approximation $\tilde{\alpha}_h(t)$ of the spatial truncation error on the (original) fine mesh is obtained by interpolation respecting the order of accuracy (see Section 5). Thus, to approximate the global spatial error $\eta_h(t)$ we consider instead of (11) the step-size frozen version

$$\begin{aligned}
\tilde{\eta}_h'(t) &= \partial_{U_h} F_h(t_n, V_{h,n}) \, \tilde{\eta}_h(t) - \tilde{\alpha}_h(t), \quad t \in (t_n, t_{n+1}], \; n = 0, \dots, M-1, \\
\tilde{\eta}_h(0) &= 0.
\end{aligned} \tag{37}$$

5   The example discretization formulas

In order to keep the illustration as simple as possible we restrict ourselves to one space dimension. For the spatial discretization of (1) we use standard second-order finite differences. Hence we have $q = 2$. The discrete L_2-norm on a non-uniform mesh

$$x_0 < x_1 < \ldots < x_N < x_{N+1}\,, \quad h_i = x_i - x_{i-1}\,, \quad i = 1, \ldots, N+1\,, \tag{38}$$

for a vector $y = (y_1, \ldots, y_N)^T \in \mathbb{R}^N$ is defined through

$$\|y\|^2 = \sum_{i=1}^{N} \frac{h_i + h_{i+1}}{2}\, y_i^2\,. \tag{39}$$

Here, the components y_0 and y_{N+1} which are given by the boundary values are not considered.

The example time integration formulas are taken from [5]. For the sake of completeness we shall give a short summary of the implementation used. To generate the time grid (5) we use as an example integrator the 3rd-order, A-stable Runge-Kutta-Rosenbrock scheme ROS3P, see [3, 4] for more details. The property of tolerance proportionality [8] is asymptotically ensured through working for the local residual with

$$Est = \frac{2}{3}\left(I_h - \gamma\tau_n A_{h,n}\right)^{-1} r_h(t_{n+1/2})\,, \qquad A_{h,n} = \partial_{U_h} F_h(t_n, V_{h,n})\,, \tag{40}$$

where γ is the stability coefficient of ROS3P. The common filter $(I_h - \gamma\tau_n A_{h,n})$ serves to damp spurious stiff components which would otherwise be amplified through the F_h-evaluations within $r_h(t_{n+1/2})$.

Let $D_n = \|Est\|$ and $Tol_n = Tol_A + Tol_R\|V_{h,n}\|$ with Tol_A and Tol_R given local tolerances. If $D_n > Tol_n$ the step is rejected and redone. Otherwise the step is accepted and we advance in time. In both cases the new step size is determined by

$$\tau_{new} = \min\bigl(1.5, \max(2/3, 0.9\, r)\bigr)\tau_n\,, \qquad r = (Tol_n/D_n)^{1/3}\,. \tag{41}$$

After each step size change we adjust τ_{new} to $\tau_{n+1} = (T - t_n)/\lfloor(1 + (T - t_n)/\tau_{new})\rfloor$ so as to guarantee to reach the end point T with a step of averaged normal length. The initial step size τ_0 is prescribed and is adjusted similarly.

The linear error transport equations (23) and (37) are simultaneously solved by means of the implicit midpoint rule, which gives approximations $\tilde{e}_{h,n}$ and $\tilde{\eta}_{h,n}$ to the global time and spatial error at time $t = t_n$. We use the implementations

$$\left(I_h - \frac{1}{2}\tau_n A_{h,n}\right)\delta e_{n+1} = 2\tilde{e}_{h,n} + \frac{2}{3}\tau_n r(t_{n+1/2})\,, \qquad \tilde{e}_{h,n+1} = \delta e_{n+1} - \tilde{e}_{h,n}\,, \tag{42}$$

and

$$\left(I_h - \frac{1}{2}\tau_n A_{h,n}\right)\delta\eta_{n+1} = 2\tilde{\eta}_{h,n} - \tau_n\tilde{\alpha}_h(t_{n+1/2})\,, \qquad \tilde{\eta}_{h,n+1} = \delta\eta_{n+1} - \tilde{\eta}_{h,n}\,. \tag{43}$$

Clearly, the matrices $A_{h,n}$ already computed within ROS3P can be reused. The spatial truncation error $\tilde{\alpha}_{2h}(t)$ at $t=t_{n+1/2}$ is given by

$$\tilde{\alpha}_{2h}(t_{n+1/2}) = \frac{4}{3} \left(R_{2h}^h F_h \left(t_{n+1/2}, V_h(t_{n+1/2}) \right) - F_{2h} \left(t_{n+1/2}, R_{2h}^h V_h(t_{n+1/2}) \right) \right) . \tag{44}$$

Since $V_h(t_{n+1/2})$ and $F_h(t_{n+1/2}, V_h(t_{n+1/2}))$ are available from the computation of $r_h(t_{n+1/2})$ in (26), this requires only one function evaluation on the coarse grid. The vector $\tilde{\alpha}_{2h}(t_{n+1/2})$ on the coarse mesh is prolongated to the fine mesh and is then divided by $2^q = 4$ if the neighboring fine grid points are equidistant, otherwise it is divided by $2^{q-1} = 2$. The remaining $\tilde{\alpha}_h(t_{n+1/2})$ on the fine mesh are computed by interpolation respecting the order of the neighboring spatial truncation errors.

Due to freezing the coefficients in each time step, the second-order midpoint rule is a first-order method when interpreted for solving the linearised equations (14) (or likewise the first variational equation) and (11). Thus if all is going well, we asymptotically have $\tilde{e}_{h,n} = e_h(t_n) + \mathcal{O}(\tau_{max}^4)$ and $\tilde{\eta}_{h,n} = \eta_h(t_n) + \mathcal{O}(\tau_{max} h_{max}^q) + \mathcal{O}(h_{max}^{q+1})$.

After computing the spatial truncation errors we can solve the discretized error transport equations (43) for all $\tilde{\eta}_{h,n}$. We restrict here to globally uniform refinement. A locally adaptive refinement strategy can be found in [2]. Although the uniform strategy may be less efficient, it is very easy to implement and therefore of special practical interest if software packages which do no allow dynamic adaptive mesh refinement are used.

Let Tol be a given tolerance. Then our aim is to guarantee $\|\eta_h(T)\| \le Tol$. From (43), we get an approximate value $\tilde{\eta}_{h,M}$ for the spatial discretization error at T. If the desired accuracy is still not satisfied, i.e., $\|\tilde{\eta}_{h,M}\| > Tol$, we choose a new (uniform) spatial resolution

$$h_{new} = \sqrt[q]{\frac{Tol}{\|\tilde{\eta}_{h,M}\|}} \, h \tag{45}$$

to account for achieving $\|\eta_{h_{new}}(T)\| \approx Tol$. From h_{new} we determine a new number of mesh points. The whole computation is redone with the new spatial mesh.

6   The control rules

Like for the ODE case studied in [5] our aim is to provide global error estimates and to control the accuracy of the solution numerically computed to the imposed tolerance level. Let $GTol_A$ and $GTol_R$ be the global tolerances. Then we start with the local tolerances $Tol_A = GTol_A$ and $Tol_R = GTol_R$.

Suppose the numerical schemes have delivered an approximate solution $V_{h,M}$ and global estimates $\tilde{e}_{h,M}$ and $\tilde{\eta}_{h,M}$ for the time and spatial error at time $t_M = T$. We then verify whether

$$\|\tilde{e}_{h,M}\| \le C_T C_{control} Tol_M, \quad Tol_M = GTol_A + GTol_R \|V_{h,M}\|, \tag{46}$$

where $C_{control} \approx 1$, typically > 1, and $C_T \in (0,1)$ denotes the fraction desired for the global time error with respect to the tolerance Tol_M. If (46) does not hold, the whole

computation is redone over $[0, T]$ with the same initial step τ_0 and the adjusted local tolerances

$$Tol_A = Tol_A \cdot fac, \quad Tol_R = Tol_R \cdot fac, \quad fac = C_T Tol_M / \|\tilde{e}_{h,M}\|. \tag{47}$$

Based on tolerance proportionality, reducing the local error estimates with the factor fac will reduce $e_h(T)$ by fac [8].

If (46) holds, we check whether

$$\|\tilde{e}_{h,M} + \tilde{\eta}_{h,M}\| \leq C_{control} Tol_M. \tag{48}$$

If it is true, the overall error $E_h(T) = V_h(T) - R_h u(T) = e_h(T) + \eta_h(T)$ is considered small enough relative to the chosen tolerance and $V_{h,M}$ is accepted. Otherwise, the whole computation is redone with the (already) adjusted tolerances (47) and an improved spatial resolution.

We use the new mesh size computed from (45) with $Tol = (1 - C_T)Tol_M$. To check the convergence behaviour in space and therefore also the quality of the approximation of the spatial truncation error, we additionally compute the numerically observed order

$$q_{num} = \log\left(\frac{\|\tilde{\eta}_{h,M}\|}{\|\tilde{\eta}_{h_{new},M}\|}\right) \Big/ \log\left(\frac{h}{h_{new}}\right). \tag{49}$$

If q_{num} computed for the final run is not close to the expected value q used for our Richardson extrapolation, we reason that the approximation of the spatial truncation errors has failed due to a dominating global time error, which happens, e.g., if the initial spatial mesh is already too fine. Consequently, we coarsen the initial mesh by a factor two and start again. If the control approach stops without a mesh refinement, we perform an additional control run on the coarse mesh and compute q_{num} from (49) with $h_{new} = 2h$. It turns out that this simple strategy works quite robustly, provided that the meshes used are able to resolve the basic behaviour of the solution.

Summarizing, the first check (46) and the possibly second control computation serve to significantly reduce the global time error. This enables us to make use of the approximation (36) for the spatial truncation error, which otherwise could not be trusted. The second step based on suitable spatial mesh improvement attempts to bring the overall error down to the imposed tolerance. Using the sum of the approximate global time and spatial error inside the norm in (48), we take advantage of favourable effects of error cancellation. These two steps are successively repeated until the second check is successful. Additionally, we take into account the numerically observed order in space to assess the approximation of the spatial truncation error.

7 Numerical illustration

To illustrate the performance of the global error estimators and the control strategy, we consider the Allen-Cahn equation modelling a diffusion-reaction problem. For results on further test problems and adaptive mesh refinement, see [2].

The bi-stable Allen-Cahn equation is defined by

$$\partial_t u = 10^{-2}\, \partial_{xx} u + 100 u\left(1 - u^2\right), \quad 0 < x < 2.5\,, \qquad 0 < t \le T = 0.5\,, \qquad (50)$$

with the initial function and Dirichlet boundary values taken from the exact wave front solution $u(x,t) = (1 + e^{\lambda(x - \alpha t)})^{-1}$, $\lambda = 50\sqrt{2}$, $\alpha = 1.5\sqrt{2}$. This problem was also used in [5].

We set $GTol_A = GTol_R = GTol$ for $GTol = 10^{-l}, l = 2, \ldots, 7$ and start with one and the same initial step size $\tau_0 = 10^{-5}$. Equally spaced meshes of 25, 51, 103, 207, 415, 831, and 1663 points are used as initial mesh. The control parameters introduced above for the control rules are $C_T = 1/3$ and $C_{control} = 1.2$. All runs were performed, but for convenience we only select a representative set of them for our presentation, which can be found in Table 1.

Table 1: Selected data for the Allen-Cahn problem.

Tol	N	Tol_M	$\|\tilde{E}_{h,M}\|$	$\|\tilde{e}_{h,M}\|$	$\|\tilde{\eta}_{h,M}\|$	Θ_{est}	Θ_{ctr}	q_{num}
1.00e-2	103	2.05e-2	1.84e-0	1.45e-1	1.98e-0	9.89	0.11	
4.69e-4	103	2.05e-2	5.78e-1	1.26e-3	5.79e-1	2.69	0.10	
4.69e-4	677	2.02e-2	6.04e-3	1.11e-3	7.15e-3	1.19	3.98	2.34
1.00e-2	415	2.02e-2	7.69e-2	1.44e-1	6.73e-2	3.05	0.80	
4.66e-4	415	2.02e-2	1.86e-2	1.11e-3	1.97e-2	1.23	1.34	
4.66e-4	207	2.03e-2	9.17e-2	1.15e-3	9.29e-2	1.47	0.32	2.24
1.00e-3	207	2.03e-3	9.82e-2	2.97e-3	1.01e-1	1.60	0.03	
2.27e-4	207	2.03e-3	8.80e-2	4.93e-4	8.85e-2	1.39	0.03	
2.27e-4	1683	2.02e-3	6.14e-4	4.71e-4	1.09e-3	1.11	3.67	2.10
1.00e-3	831	2.02e-3	2.26e-3	2.87e-3	5.12e-3	1.33	1.19	
2.35e-4	831	2.02e-3	4.01e-3	4.91e-4	4.50e-3	1.12	0.57	
2.35e-4	1521	2.02e-3	8.42e-4	4.90e-4	1.33e-3	1.12	2.68	2.02
1.00e-4	1663	2.02e-4	8.89e-4	1.86e-4	1.08e-3	1.07	0.24	
3.63e-5	1663	2.02e-4	9.88e-4	6.14e-5	1.05e-3	1.05	0.21	
3.63e-5	4643	2.02e-4	7.30e-5	6.14e-5	1.34e-4	1.04	2.89	2.00

Table 1 contains the following quantities, $Tol = Tol_A = Tol_R$ from (47), the number of mesh points N, $Tol_M = GTol\,(1 + \|V_{h,M}\|)$ from (46), the estimated global error $\tilde{E}_{h,M} = \tilde{e}_{h,M} + \tilde{\eta}_{h,M}$ at time $t = T$, the estimated time error $\tilde{e}_{h,M}$, and the estimated spatial truncation error $\tilde{\eta}_{h,M}$. Note that we always start with $Tol = GTol$ in the first run.

The indicators $\Theta_{est} = \|\tilde{E}_{h,M}\|/\|E_h(T)\|$ for the ratio of the estimated global error and the true global error, and $\Theta_{ctr} = Tol_M/\|E_h(T)\|$ for the ratio of the desired tolerance and

the true global error serve to illustrate the quality of the global error estimation and the control. $\Theta_{ctr} \geq 1/C_{control} = 5/6$ indicates control of the true global error.

The numerically observed order q_{num} for the spatial error is also given. From the table one can see whether a tolerance-adapted run to control the global time error, a spatial mesh adaptation step or an additional control run on a coarser grid was necessary. Especially, the latter is marked by a dashed line.

Table 1 reveals a high quality of the global error estimation and also the control process works quite well.

Let us pick one exemplary run out to explain the overall control strategy in more detail. Starting with $GTol = Tol = 10^{-3}$ and 831 mesh points, which corresponds to the fourth simulation, the numerical scheme delivers global error estimates $\|\tilde{e}_{h,M}\| = 2.87 \times 10^{-3}$ and $\|\tilde{\eta}_{h,M}\| = 5.12 \times 10^{-3}$ for the time and spatial error of the approximate solution $V_{h,M}$ at the final time $t_M = T$. The first check for the time error estimate $\|\tilde{e}_{h,M}\| \leq C_T C_{control} Tol_M = 8.08 \times 10^{-4}$ fails and we adjust the local tolerances by a factor $fac = C_T Tol_M / \|\tilde{e}_{h,M}\| = 2.35 \times 10^{-1}$, which yields the new $Tol = 2.35 \times 10^{-4}$. The computation is then redone. Due to the tolerance proportionality, in the second run the time error is significantly reduced and the inequality $\|\tilde{e}_{h,M}\| \leq 8.08 \times 10^{-4}$ is now valid. We proceed with checking $\|\tilde{E}_{h,M}\| \leq C_{control} Tol_M = 2.42 \times 10^{-3}$, which is still not true. From (45), we compute a new number of spatial mesh points $N = 1521$. Finally, the third run is successful and with the numerically observed spatial order $q_{num} = 2.02$ the numerical solution is accepted.

The ratios for $\Theta_{est} = \|\tilde{E}_{h,M}\| / \|E_h(T)\|$ lie between 1.04 and 1.23, after the control runs. Control of the global error, that is $\|E_h(T)\| \leq C_{control} Tol_M$, is in general achieved after two steps (one step to adjust the time grid and one step to control the space discretization), whereas the efficiency index $\Theta_{ctr} = Tol_M / \|E_h(T)\|$ is close to three. This results from a systematic cancellation effect between the global time and spatial error, which is not taken into account when computing h_{new} from (45).

8 Summary

We have developed an error control strategy for finite difference solutions of parabolic equations, involving both temporal and spatial discretization errors. The global time error strategy discussed in [5] appears to provide an excellent starting point for the development of such an algorithm. The classical ODE approach based on the first variational equation and the principle of tolerance proportionality is combined with an efficient estimation of the spatial error and mesh adaptation to control the overall global error. Inspired by [1], we have used Richardson extrapolation to approximate the spatial truncation error within the method of lines. Our control strategy aims at balancing the spatial and temporal discretization error in order to achieve an accuracy imposed by the user.

The key ingredients are: (i) linearised error transport equations equipped with sufficiently accurate defects to approximate the global time error and global spatial error and (ii) uniform or adaptive (see [2]) mesh refinement and local error control in time

based on tolerance proportionality to achieve global error control. For illustration of the performance and effectiveness of our approach, we have implemented second-order finite differences in one space dimension and the example integrator ROS3P [4]. On the basis of the test problem in this article and two other test problems in [2] we could observe that our approach is very reliable, both with respect to estimation and control.

REFERENCES

[1] M. Berzins (1988), *Global error estimation in the method of lines for parabolic equations*, SIAM J. Sci. Stat. Comput. 9, pp. 687–703.

[2] K. Debrabant, J. Lang, *On Global Error Estimation and Control of Finite DifferenceSolutions for Parabolic Equations*, Preprint, arXiv:0911.2656v1 [math.NA], Nov. 2009.

[3] J. Lang (2000), *Adaptive Multilevel Solution of Nonlinear Parabolic PDE Systems. Theory, Algorithm and Applications*, Lecture Notes in Computational Science and Engineering, Vol. 16, Springer.

[4] J. Lang, J.G. Verwer (2001), *ROS3P – An accurate third-order Rosenbrock solver designed for parabolic problems*, BIT 41, pp. 731–738.

[5] J. Lang, J.G. Verwer (2007), *On global error estimation and control for initial value problems*, SIAM J. Sci. Comput. 29, pp. 1460–1475.

[6] S. Larsson, V. Thomée (2005), *Partial Differential Equations with Numerical Methods*, Texts in Applied Mathematics, Vol. 45, 2nd printing, Springer.

[7] L. Lawson, M. Berzins, P.M. Dew (1991), *Balancing space and time errors in the method of lines for parabolic equations*, SIAM J. Sci. Stat. Comput. 12, pp. 573–594.

[8] L.F. Shampine (1994), *Numerical Solution of Ordinary Differential Equations*, Chapman & Hall, New York.

[9] R.D. Skeel (1986), *Thirteen ways to estimate global error*, Numer. Math. 48, pp. 1–20.

[10] J.W. Thomas (1995), *Numerical Partial Differential Equations. Finite Difference Methods*, Text in Applied Mathematics 22, Springer.

MASSIVELY PARALLEL COMPUTATION ON ANISOTROPIC MESHES

H. DIGONNET[*], L. SILVA[*] AND T. COUPEZ[*]

[*]CEMEF - MINES ParisTech
Rue Claude Daunesse
BP 207
06904 Sophia Antipolis cedex
France

Key words: Massively parallel computation, Anisotropic mesh adaptation, Multigrid

Abstract. In this paper, we present developments done to obtain efficient parallel computations on supercomputers up to 8192 cores. While most massively parallel computation are shown using regular grid it is less common to see massively parallel computation using anisotropic adapted unstructured meshes. We will present here two mains components done to reach very large scale calculation up to 10 billions unknowns using a muligrid method over unstructured mesh running on 8192 cores. We firstly focus on the strategy used to generate computational meshes and in particular anisotropic ones adapted to capture a quite complicated test function. Then we will briefly describe a parallel multigrid method. Performance test over a large range of cores from 512 to 8192 cores is then presented using the French national supercomputers Jade and Curie. The last section will present a calculation done on smallest number of cores on our own cluster, but using more realistic data obtain directly from experimentation. The goal is to be able to realize such kind of simulation on really complex micro structure obtain by tomography at a larger scale.

1 INTRODUCTION

From the last years computers power increase mainly by cores multiplication (rather than clock rate rise) inside each CPU (classically 8 or 16 cores in 2012) and also for the supercomputers of the top500 that contain several thousand to more than one million cores. this context impose to develop fully parallel softwares to at least be able to take advantage of new computer hardware. If we look at numerical simulation, computation times are always to important and has soon as they decrease the user want to have more precise computation by introducing more physical properties or by looking in computing lower scale calculation to improve the material behavior. One of the mains way to reduce

computation time is to use parallel computers and in the ideal case divide the CPU time by the number of cores used. For that it is then necessary to develop algorithms that could run on massively parallel computer containing hundred, thousand or even more cores like ones present in the top500 list of supercomputers. We present here some algorithms done to allow our application CimLib to run over massively parallel supercomputer using up to 8192 cores. The first section will focus on parallel mesh adaptation with anisotropic elements taht lead to generate smaller meshes for a given precision. A second part will briefly present parallel multigrid method implemented to reduce the complexity of the algorithm used to solve linear systems and fully utilize the power and memory given on a computer. We present parallel performance on solving incompressible Stokes equation using from 8 to 8192 cores and enable resolution of a linear system containing more that 10 billions unknowns by using 8192 cores. The last section will present more reasonable simulation but on a more realistically and complex case. It consists in computing a flow through a micro structure given directly from a real one using tomography.

2   parallel anisotropic mesh adaptation

2.1   parallel mesh adaptation strategy

We briefly present here the methodology used to parallelize our unstructured and non hierarchic mesher, MTC [1]. At the beginning of this work, the sequential mesher already exist and is likely to evolve (we recently add anisotropic mesh size [2]), so we want to keep it as it was, by not including parallel instructions inside. So, we have not parallelized directly it, but only give the oprtunity to use it inside a parallel context. The strategy is then to combine local remeshing (inside each processor domain, freezing the interfaces between the partitions) [3] and parallel mesh repartitioning [4] to move unremeshed interfaces inside domains for the next step, in order to be able to apply the remeshing procedure on theses zones during the next phase. Figure 1 illustrate this strategy on a simple geometry using seven processors.

2.1.1   optimization and performance

This strategy of parallelization leads to several iterations (depending of the space dimension) between the mesher and the repartitioner, but the work to be done at each iteration decreases quickly. For example, in a 2d space : the first remeshing step is proportional to a surface, the second to a line and the last one to a point. For the repartitionner, as we only need to move bad quality zones inside the domain in order to remesh it, the proportion of elements and nodes to be migrate across the processors decreases in the same way. For that reason the cost of using a multi steps strategy must be really close to a single remeshing step: the global work done by the mesher is approximately the same, and only some few mesh migrations are done during repartitioning steps. For that purpose an optimization of the update procedures in case of small changes has been done (the time spent to remesh or migrate 10 elements is then close to zero). This optimization
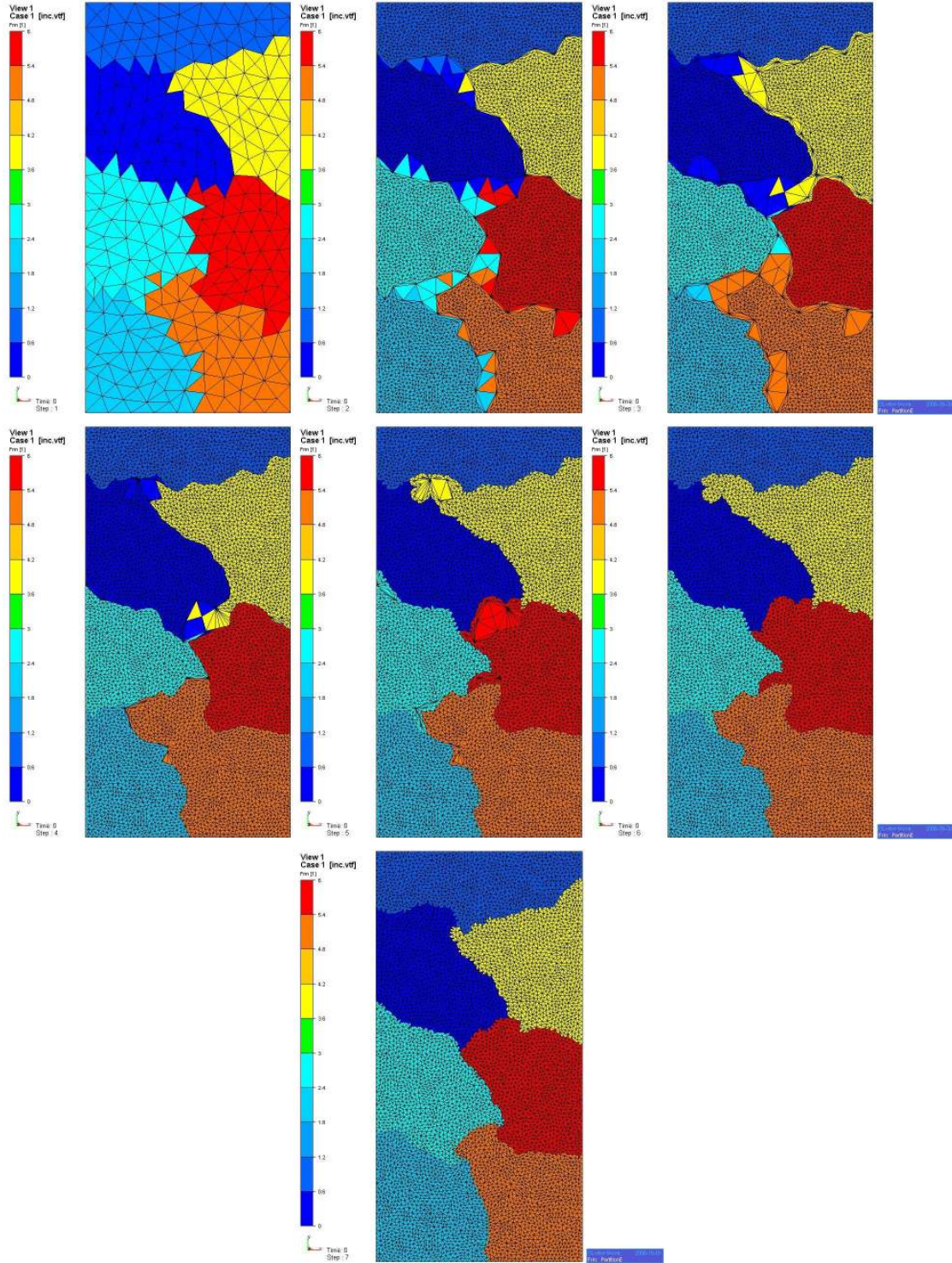
Figure 1: Illustration of the parallelization strategy used for the mesher in a 2d case with a refinement by 4, over 7 processors. Each image represent one step of the iterative methodology consisting in successively call parallel remeshing with frozen interfaces and parallel repartitioning, until convergence to good quality mesh. The last repartitionning phase is done to load balance the work by considering finite element resolution to come.

is based on a permute-copy-past algorithm that leads to reduce the complexity from N (the data size) to m (the moving size) with $m \ll N$. Permutation removes any copy of all the data by making instead some few permutations. After that, work is performed only in the cutting zone and pasted back after being done. This optimization was essential to maintain the iterative strategy costless, and so provide a good parallel efficiency to the mesher.

2.2 anisotropic error estimator

Anisotropic meshes are used to keep the same accuracy of isotropic meshes but with a smaller number of nodes and elements by allowing different mesh size depending of the direction. The use of such anisotropic meshes could reduce drastically the size of the mesh and in particular in 3d where a uniform refinement of a factor 2 increase the number of nodes by a factor 8 that has to be compared to only a factor 2 if the refinement is only needed in one direction. For anisotropic mesh adaptation we need use a mesh size definied by a scalar for isotropic mesh and a symmetric positive defined tensor (named metric) for anisotropic one. In order to enable mesh adaptation we need to provide to the mesher a continuous field of mesh size one simple solution is to give a P1 interpolate field where a local mesh size is defined at each mesh nodes. The way to compute this field could be a average of the elements metrics containing this node with the difficulty of choosing a good average value for tensors. A more direct way to defined this metric consist in using the distribution tensor describe in [2] and given by :

$$\mathbb{X}_i = \frac{1}{|\Gamma(i)|} \sum_{j \in \Gamma(i)} X^{ij} \otimes X^{ij}$$

where X^{ij} is the vector $\overrightarrow{X_i X_j}$ between two neighbor nodes i and j and $\Gamma(i)$ the neighbors nodes of i.

This tensor give the anisotropic size of the envelope containing of the elements belonging to node i, and the natural metric for the mesh is then simply defined by $\mathbb{M}_i = \mathbb{X}_i^{-1}$ to insure that using this metric this envelope has an unit length.

The anisotropic error estimator is well presented in [2] and we will here only present the mains ideas. To be able to build a anisotropic error estimator we first need to be able to define an anisotropic error and for that we will use the same technique of the distribution tensor. For a given Level Set we will compute the error over nodes edges as :

$$e_{ij} = \left| G^{ij} X^{ij} \right|$$

where G^{ij} is the gradient variation of the level set function across the edge ij.

With this anisotropic error and given a number of nodes mesh we could define (for demonstration refer to paper [2]) a metric tensor to generate an adapted mesh that will uniformly distribute the error across its edges. The metric is given by :

$$\mathbb{M}_i = \frac{1}{|\Gamma(i)|} \left(\sum_{j \in \Gamma(i)} s_{ij}^2 X^{ij} \otimes X^{ij} \right)^{-1}$$

with

$$p \in [1, d]$$

$$\lambda = \left(\frac{\left(\sum_i \sum_{j \in \Gamma(i)} e_{ij}^{\frac{p}{p+2}} \right)}{A} \right)^{\frac{p+2}{2}}$$

where d is the space dimension, A is the global number of edges in the mesh and p is the power of the power law estimation for the number of edges generated in the mesh when we want to divide one edge. For example: for a one direction anisotropic mesh divide an edges in that direction by a factor 2 will only multiply by 2 the global number of edges in the mesh and then $p = 1$; in case of an isotropic mesh divide an edge by two will generate eight times more edges so $p = 3$.

2.3   application test case

We present in this section an application test case adapted to massively parallel computation. It consist in computing a adapted mesh to allow a good representation of a complicated function defined by :

$$f(x) = a \circ a(x - x_0) + a \circ a(x - x_1)$$

where

$$a(x) = tanh\left(Esin\left(\frac{4*N+1}{2} \Pi \|x\| \right) \right)$$

and

$$x_0 = (0, .., 0) \qquad x = (1, ..., 1)$$

E and N are two parameters that can be adjusted. They respectively influence the thickness and the complexity of the details present in the function: if E increase the size of the detail decrease and if N increase then each detail contain become more and more complex. The figure 2 represent the test function on an uniform square mesh containing around 50000 nodes for E,N parameters equal to 1, 2 and 4. We notice that for E=4

and N=4 the uniform mesh start to be to small to obtain a good representation of the function f.

As soon as these parameters increase the function become really difficult to capture and we need to activate both the anisotropic mesh adaptation and parallel computation to generate meshes fine enough to represent the test function. The bottom of figure 3represent the 23 millions nodes anisotropic mesh generate to represent the test function with E=16 and n=6 on the unit square. It clearly illustrate the benefit of being able to execute anisotropic mesh adaptation in parallel, here using 256 cores.

The figure 4 represent the same test case in a 3d context using smaller parameters $N = 1$ and $E = 4$. E and N parameters. The 3d adapted mesh contain 2 millions nodes and have been generate using 50 mesh adaptation iterations using 64 cores of Jade supercomputer in 15.5 hours .

3   Multigrid solver

Using powerful computers containing a large numbers of cores could lead using very big meshes with several millions nodes. Solving physical equation using for example the finite element method will ends in solving very large linear systems. Using iterative methods like Krylov ones will parallel preconditioner give good parallel performance but the algorithm complexity (around $\mathcal{O}(n^{3/2})$ in 3d) become a bottleneck as soon as the number of unknowns start to be important (taking a problem 2 times bigger and using 2 times more cores the time spent to solve will be greater than 2). To over come this difficulty we have implemented a parallel multigrid solver using the framework give by PETSc [5]. To do this we need to provide for each level the system to be solve and also the interpolation/restriction matrix operator. Of course all these matrices must be build and store in a distributed way to allow using a large number of cores. More details are presented in [6] but a particular attention have been given to reduce the communication during the construction of the interpolation operator using a parallel octree localization algorithm and pixel mask filter distributed over processors to limit false positive external nodes detection. The figure 5 present parallel performance of the multigrid resolution for Stokes equation over a 2d square mesh of 800 millions nodes and using 512 to 4096 cores. Time spent to solve the system decrease from 96.7 seconds with 512 cores to 17.7 seconds with 4096 cores, in the same time the number of multigrid iteration only increase from 11 to 13 and leads to a speed-up of 5.46.

Even if a good parallel performance of the multigrid solver is important, the main interest is the scalability of this approach that make us able to solve very very large systems. An ideal scalability will be give by a algorithm that will use exactly the same computing resources for a given per core problem size independently of the number of cores uses. A scalability test have been done from 8 to 8192 cores on the multigrid solver. The results are very interesting : the memory used per core during the process stay nearly constant (from 1.94 to 2.14 GB) as well as the time spent to assemble all levels systems (from 9.7 to 11 seconds) and the multigrid iterations (from 13 to 11); only the resolution
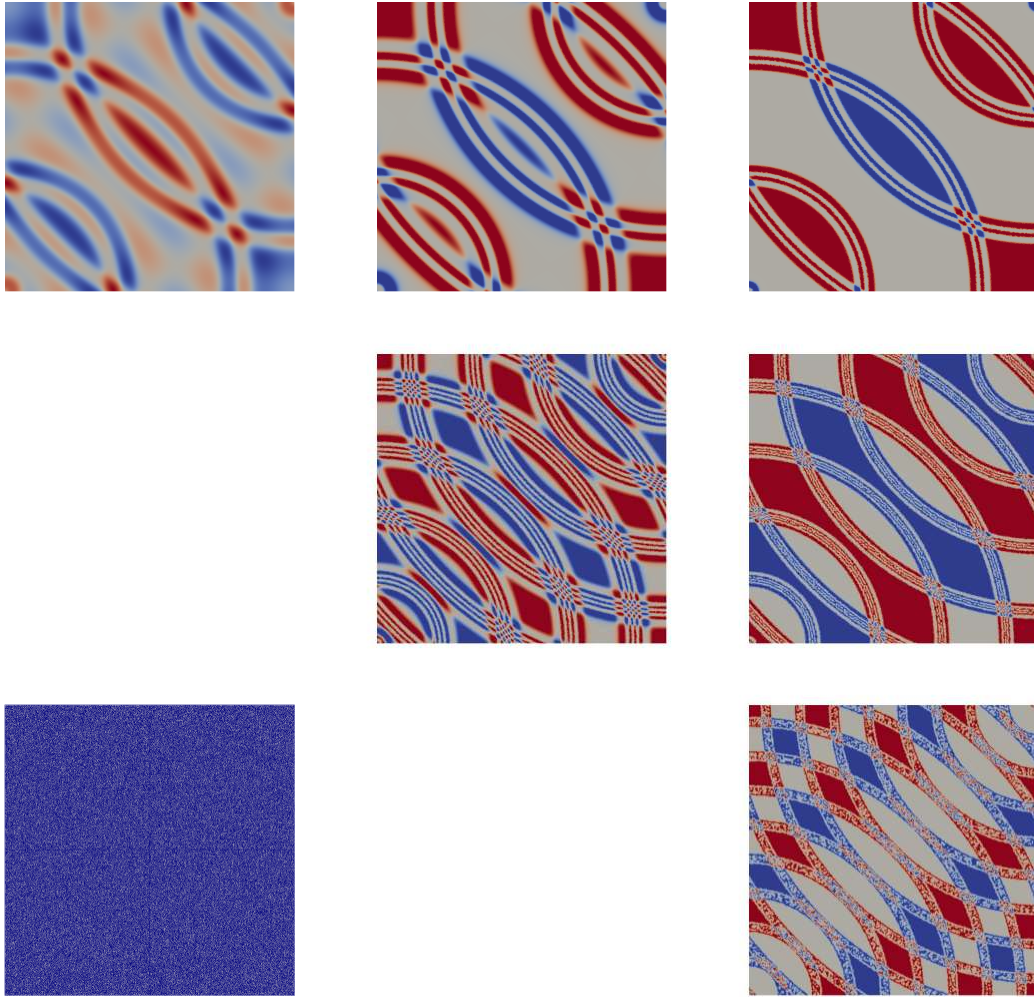
Figure 2: Influence of the two parameters N and E on the complexity of the test function. This test function is plot on an uniform 2d mesh containing 50 000 nodes. From the top to the bottom the parameter $N = 1, 2, 4$ and from the left to the right the parameter $E = 1, 2, 4$. On the left bottom the uniform mesh used to plot the test function.
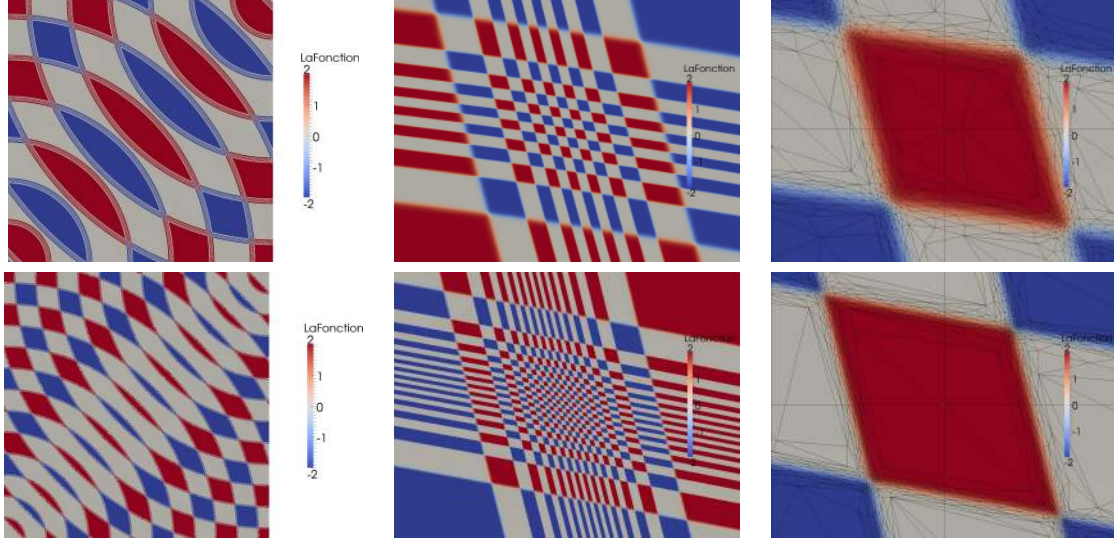
Figure 3: Anisotropic mesh adaptation around the test function $f(x) = g \circ g(\|x - 0\| + g \circ g(\|x - 1\|)$ with $g(x) = tanh(Esin(\frac{4*N+1}{2}\pi x))$ with 2 sets of parameters : on the top N=3, E=16 and at the bottom $N = 6$, $E = 16$. For these two sets we present : the function on the unit square, one cross detail and a zoom on the mesh for a deep detail inside the cross detail. The 2d adapted meshes containing respectively 1 million and 23 millions nodes. They where build in respectively 576 seconds over 128 cores and 3 hours on 256 cores of Jade supercomputer. The smallest mesh size is around receptively 10^{-5} and 10^{-6}.
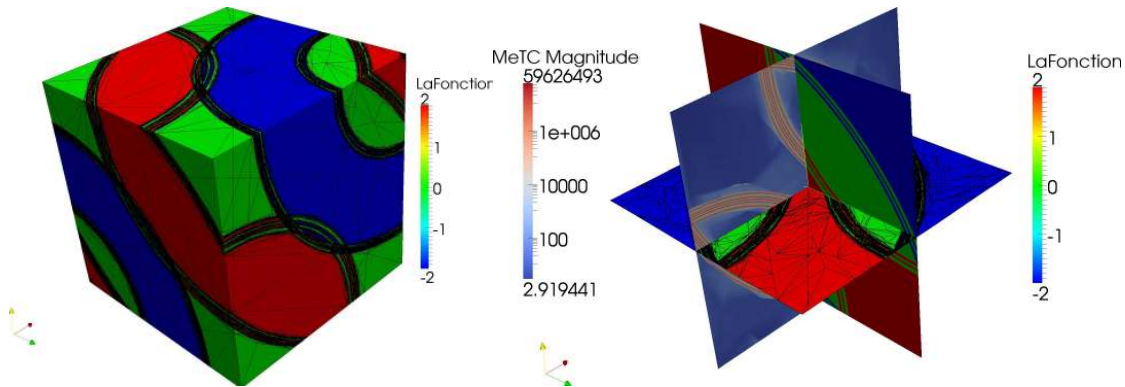
Figure 4: Anisotropic mesh adaptation in 3d around the test function with $N = 1$ and $E = 8$ for a final mesh containing 2 millions nodes.
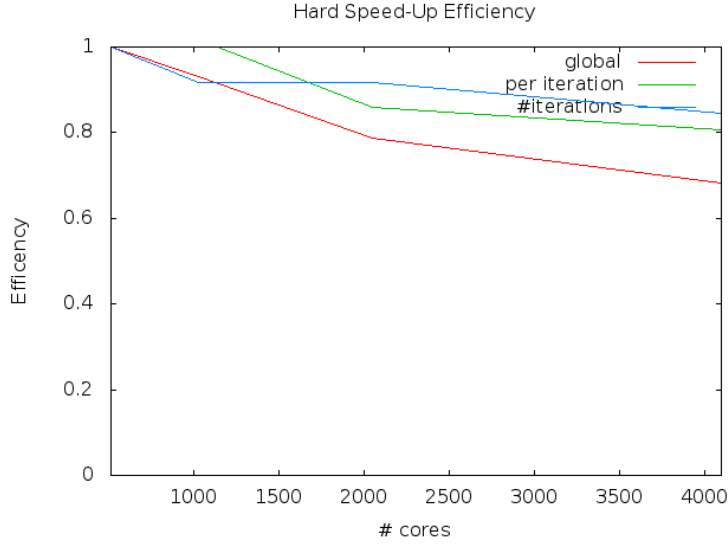
Figure 5: Parallel efficiency for the multigrid resolution using 512 to 4096 cores. The number of multigrid iteration needed to converge stay almost constant (efficiency close to one) has well as the parallel efficiency per iteration. At the global point of view the parallel efficiency is a bit worse due to the combination of per iteration parallel performance and increasing number of iteration but still very good (close to one) has the problem size per core decrease as number of cores used increase.

time increase a beat more from 90 to 148 seconds. This augmentation of 50% of the time spent to solve the problem between 8 and 8192 cores is still reasonable and may also be reduce using more multigrid level over 8192 cores as the coarse level system size may become to big (around 600 000 unknowns) to be solve in a efficient way. To conclude this multigrid solver implementation have been able to solve a global system (based on the Stokes equation) containing more than 10 billions unknowns using 8 levels in 158 seconds using 8192 cores and 17.5 PB of memory.

4 From real to virtual

In this section we present some work done to enable making simulation using real micro structure. Today tomography could produce some really nice (well defined but also heavy) image of real material as shown on the left of figure 6 [7]. This image is constituted of a large number of gray level voxel (equivalent to pixel in 2d), the back ones represent the polymer matrix and white one solid fibers. From this image we extract the an isosurface mesh (shown on the right of figure 6). Depending of the image definition the number of voxel and thus the number of faces in the isosurface mesh could become really important and may need to be executed in parallel using some small enough part of the whole domain using some similar technique as octree. Once we have the isosurface mesh, it could be immerse into the computational mesh by computing for all the mesh nodes the distance to the isosurface that give a level set representation of the micro structure [8]. To improve the
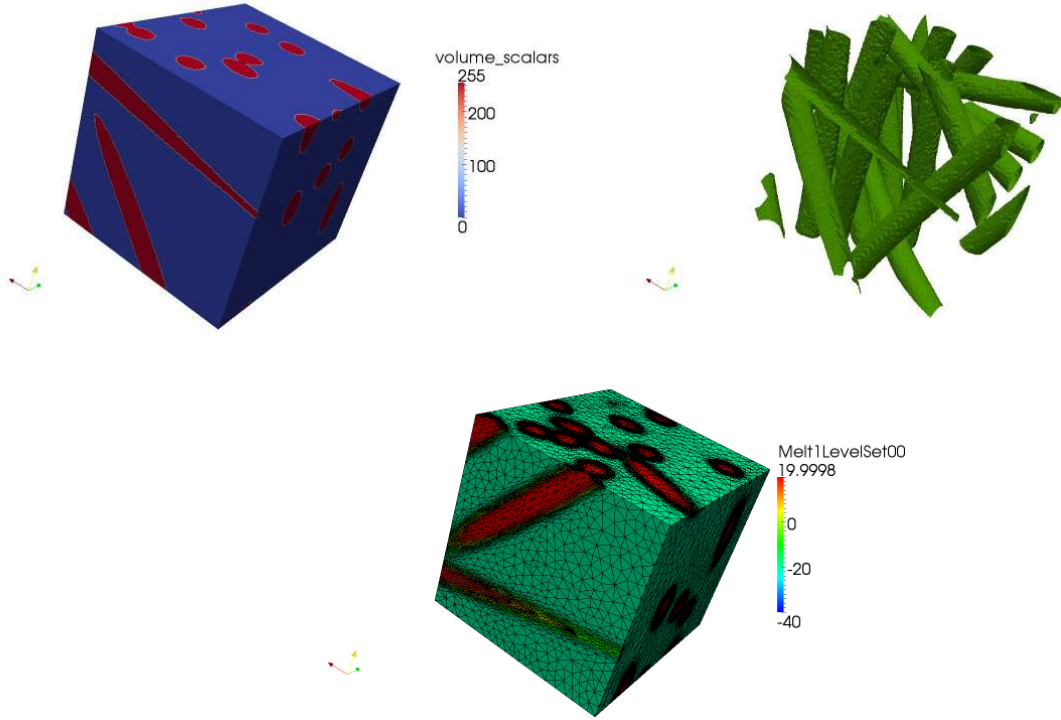
Figure 6: From real to virtual: a large 3d voxel image of the microstructure obtain by tomography and treated with a segmentation algorithm, its surface mesh given by the isosurface 127, and finally the computational domain with an anisotropic adapted mesh.

immersion of the microstrure we could lauch some anisotropic mesh adaptation procedures that will provide the computational adapted mesh as shown in bottom of figure 6.

If the previous figures 6were done to illustrate the mesh adaptation to the real microstructure using anisotropic mesh, using bigger image for the microstructure generate very big mesh and justify using massively parallel computer for both anisotropic mesh adaptation but also for computing the solution. Figure 7 represent the computed flow using Stokes equation across the immerged microstucture. Computation was done using 96 cores on a 10 millions nodes adapted mesh to a microsuture image containing 900x900x220 voxels. This is this type of computation than we plan to do in a close future but over much more larger image with around 4000x4000x4000 voxels given by new tomographic acquisition. For being able to do that all the computational chain need to be run in parallel from the image generation to the visualization of the computed results.

5 CONCLUSIONS

In this paper, we have presented the parallelization strategy for the mesher that consist in iterate between a remeshing step with frozen interfaces and repartitionning step used to migrate interfaces. In that way the mesher engine stay sequential and do not contain any
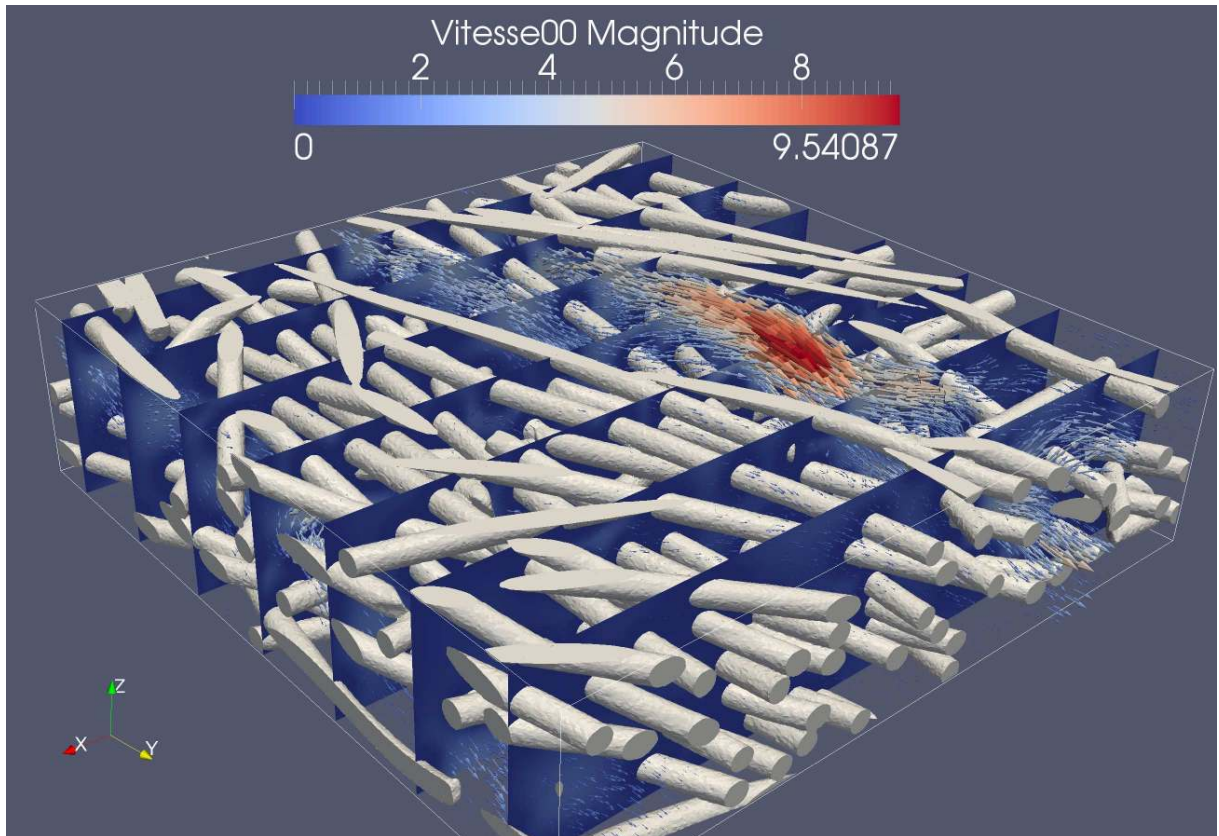
Figure 7: Large numerical simulation of the 3d flow across a microstructure done using a big 900x900x220 tomographic image. Flow is compute over a 10 millions nodes adapted mesh using 96 cores. The isosurface represent fibers microstructure, vector the velocity field.

parallel instruction so it could still be developed (introduce anisotropic mesh) without any specific knowledge. The implementation of a permutation cut and past optimization gives excellent parallel performance as well as a better quality control for meshes generated. Some examples of anisotropic distributed meshes adapted to well represent a complicated test function are give and clearly show the interest of using anisotropic mesh rather than isotropic ones : a 23 millions nodes anisotropic mesh adapted using 128 cores can capture the test function with parameters N=6 and E=16 leading to a very complex function, a equivalent representation will need a picture containing around one thousand billions pixels.

A massively parallel multigrid method working on unstructured meshes have been also presented. The construction of interpolation/restriction operators between very large distributed meshes and using a large number of cores need a powerful localization algorithm used to determine which element contain a node. For that we use a parallel octree localization method improve by a domain filter define by a pixel mask to reduce false detection of non local nodes. By using this filter we have a important reduction of communication even when domains are unrelated???. Very good parallel performance have been presented for the multigrid approach use to solve Stokes equation on very fine meshes over a large range of cores from 8 up to 8192 cores. The multigrid method developed allow to better use all the resources of massively parallel computers (CPU and memory). Over 8192 cores a global system containing more than 10 billions nodes have been solved with an 8 levels multigrid solver in 148 seconds, using a total of 17,5Po of memory. It is important to notice that for such a resolution we have been oblige to remove the 32 bits integer limit in our code.

Finally we presented some more realistic simulation done on real micro structure obtained by tomography. Tomography of a real material gives a 3d image (voxels) segmented to represent the micro structure. This image is then used to build a Level-Set function. A anisotropic adapted computational mesh is build in parallel to fit the micro structure. Once this adaptation is done we could compute the flow through the micro structure. We have presented here one simulation done over a 10 millions nodes 3d adapted mesh using a micro structure image with 900x900x200 voxels. Visualization of the results has been done using a parallel version of the Paraview [9]software, coupled with distributed results files and using python scripts [10].

Future work we be done to improve and validate our algorithms and be able to scale over tens to one hundred of thousands cores as the 32 bits integer limits have already been removed. We will also look to simulate non linear materials on a larger VER given by new tomographic acquisition leading to 4000x4000x4000 voxels images.

6 Acknowledgments

REFERENCES

[1] T. Coupez, Génération de maillage et adaptation de maillage par optimisation locale, *Revue Européenne des Éléments* **9**, 403 (2000).

[2] T. Coupez, Metric construction by length distribution tensor and edge based error for anisotropic adaptive meshing, *Journal of Computational Physics"* **230**, 2391(apr 2011).

[3] T. Coupez, H. Digonnet and R. Ducloux, Parallel meshing and remeshing, *Applied Mathematical Modelling* **25**, 153 (2000).

[4] A. Basermann, J. Clinckemaillie, T. Coupez, J. Fingberg, H. Digonnet, R. Ducloux, J.-M. Gratien, U. Hartmann, G. Lonsdale, B. Maerten, D. Roose and C. Walshaw, Dynamic load-balancing of finite element applications with the DRAMA library, *Applied Mathematical Modelling* **25**, 83 (2000).

[5] S. Balay, J. Brown, , K. Buschelman, V. Eijkhout, W. D. Gropp, D. Kaushik, M. G. Knepley, L. C. McInnes, B. F. Smith and H. Zhang, *PETSc Users Manual*, Tech. Rep. ANL-95/11 - Revision 3.3, Argonne National Laboratory (2012).

[6] H. Digonnet, Multigrid using Adaptive Unstructured Meshes for Massively Parallel Computation, in *Proceedings of the Third International Conference on Parallel, Distributed, Grid and Cloud Computing for Engineering*, eds. P. Iványi and B. ToppingCivil-Comp Proceedings (Civil-Comp Press, Stirlingshire, UK, Pécs, Hongary, 2013).

[7] L. Orgéas, P. Dumont, J.-P. Vassal, O. Guiraud, V. Michaud and D. Favier, In-plane conduction of polymer composite plates reinforced with architectured networks of copper fibres, *Journal of Materials Science* **47**, 2932 (2012).

[8] P. Laure, G. Puaux, L. Silva and M. Vincent, Permeability computation on a REV with an immersed finite element method, *AIP Conference Proceedings* **1353**, Pages 978(May 2011), The 14th International Esaform Conference on Material Forming: ESAFORM 2011- Queen's University Belfast, UK, 27-29 April, 2011.

[9] A. Henderson, *The ParaView Guide: A Parallel Visualization Application* (Kitware, November 2004).

[10] H. Digonnet, Making Massively Parallel Computations Available for End Users, in *Proceedings of the Second International Conference on Parallel, Distributed, Grid and Cloud Computing for Engineering*, eds. P. Iványi and B. Topping, Civil-Comp Proceedings, Vol. 95 (Civil-Comp Press, Stirlingshire, UK, Paper 61, Ajaccio, France, 2011).

A FRAMEWORK FOR ROBUST A POSTERIORI ERROR CONTROL IN UNSTEADY NONLINEAR ADVECTION-DIFFUSION PROBLEMS

VÍT DOLEJŠÍ*, ALEXANDRE ERN† AND MARTIN VOHRALÍK‡

*Department of Numerical Mathematics
Charles University in Prague
Sokolovská 83, 186 75 Praha 8, Czech Republic
e-mail: dolejsi@karlin.mff.cuni.cz

†Université Paris-Est, CERMICS
Ecole des Ponts ParisTech
6 & 8, avenue B. Pascal, 77455 Marne-la-Vallée, France
e-mail: ern@cermics.enpc.fr

‡INRIA Paris-Rocquencourt, POMDAPI Project
B.P. 105, 78153 Le Chesnay, France
e-mail: martin.vohralik@inria.fr

Key words: Unsteady nonlinear advection-diffusion problem, a posteriori estimate, dual norm, flux reconstruction, flux equilibration, unified framework, robustness, discontinuous Galerkin method

Abstract. We derive a framework for a posteriori error estimates in unsteady, nonlinear, possibly degenerate, advection-diffusion problems. Our estimators are based on a space-time equilibrated flux reconstruction and are locally computable. They are derived for the error measured in a space-time mesh-dependent dual norm stemming from the problem and meshes at hand augmented by a jump seminorm measuring possible nonconformities in space. Owing to this choice, a guaranteed and globally efficient upper bound is achieved, as well as robustness with respect to nonlinearities, advection dominance, domain size, final time, and absolute and relative size of space and time steps. Local-in-time and in-space efficiency is also shown for a localized upper bound of the error measure. In order to apply the framework to a given numerical method, two simple conditions, local space-time mass conservation and an approximation property of the reconstructed fluxes, need to be verified. We show how to do this for the interior-penalty discontinuous Galerkin method in space and the Crank–Nicolson scheme in time. Numerical experiments illustrate the theory. More details on the analysis and results can be found in [1].

REFERENCES

[1] V. DOLEJŠÍ, A. ERN AND M. VOHRALÍK, *A framework for robust a posteriori error control in unsteady nonlinear advection-diffusion problems. SIAM J. Numer. Anal.*, in press, 2013; see also HAL Preprint 00652979 v2, 2012.

NS-IBM: A PARALLEL INCOMPRESSIBLE NAVIER-STOKES SOLVER ON UNSTRUCTURED CARTESIAN MESH WITH AUTOMATIC MESH REFINEMENT

D. DURRENGERGER AND Y. HOARAU

Institut de Mécanique des Fluides et des Solides de Strasbourg
Équipe Mécanique et Environnement
Groupe « Instabilité, Turbulence et Diphasique »
CNRS - Université de Strasbourg - ENGEES - INSA
2 Rue Boussingault, Strasbourg, France
e-mail: durrenberger@math.unistra.fr, www-imfs.u-strasbg.fr

A common feature of the various groups of the team "mécanique et environnement" of the Institute of Fluid Mechanics and Solid Strasbourg is the numerical modeling of various flows in complex geometries: turbulent flows, free surface flows, fluid-structure interaction, flows in airways. For this purpose, we use very different solvers, either free solvers or commercial solvers or solvers developed at IMFS developed. This raises, among others, the problem of continuity of knowledge.

Aware that a Navier-Stokes can not claim universality and potential to deal with any possible issues, we have developed a flexible and versatile tool for solving incompressible Navier-Stokes equations on Cartesian unstructured meshes. While the heart of a Navier-Stokes solver can be based on a relatively small selection of proven methods, the geometry processing for codes claiming geometric flexibility represents up to 90% of the investment in terms of development effort. The numerical method proposed is based on a finite volume conservative discretization and a unstructured and non-conforming grid designed to circumvent the problem of mesh generation by the use of automatic mesh refinement. The immersed boundary method is used handle the geometry.
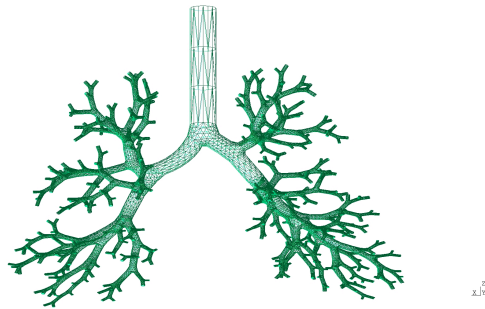
Figure 1Geometry of human airways

The solver is fully parallelized with MPI and so far it has been extensively validated on the test cases of the driven cavity, the cylinder and the square cylinder before it application on the simulation of flow in human airways. So far the AMR is only available as a pre-processing tool. In this conference we will present the concept of this solver, its validation and its application to the simulation of unsteady flows in a Human airways model under realistic breathing conditions.

P-ADAPTION FOR COMPRESSIBLE FLOWS

D. EKELSCHOT, C. BIOTTO, J. PEIRO, S. SHERWIN, D. MOXEY

Department of Aeronautics
Imperial College London
South Kensington Campus
London SW7 2AZ United Kingdom
e-mail: d.ekelschot12@imperial.ac.uk

Key words: Compressible flow, Discontinuous Galerkin method, p-adaption, Adaptive Modeling, Simulation

Abstract. We present a p-adaptive method which takes advantage of the ability of a discontinuity sensor used to quantify the difference between the actual solution and a projected reduced order solution in order to vary the polynomial resolution in an element.

1 INTRODUCTION

High-order methods have become increasingly more attractive in the field of aerodynamics due to their ability to increase the accuracy locally, their minimal numerical diffusion and dispersion properties, and the possibility to employ high-order meshes to better describe the geometry. The present work focuses on a spectral/hp element method using the Discontinuous Galerkin (DG) formulation [4] that is implemented using the open source library $Nektar^{++}$. The main advantages of the DG method range from its high accuracy to being highly flexible (allows for higher order meshes and h/p refinement) and its efficiency since it is easy to parallelise due to its block diagonal mass matrix structure. Although the DG formulation has numerous advantages, its main disadvantage is that it is computationally costly. A second limitation for the DG method is related to the treatment of flow discontinuities which, if approximated by a polynomial of high degree, leads to oscillations in the solution. As a result, an automatic polynomial adaptive procedure (p-adaption) is proposed in the present work.

The p-adaptive process can be applied to both inviscid and viscous flows and lead to a reduction in the computational cost of the simulation that could be significant, without loss of accuracy. A similar dynamic p-adaptive method is described in [1] and it is applied to the shallow water equations in [3]. The procedure described in these articles is based on a sensor that reconstructs the gradient of the solution and updates the polynomial degree by checking whether the magnitude of the sensor is higher or lower than a certain

threshold value. The method is applied at each time step and the adaption strategy is limited to use either $p = 1, 2$ or $p = 2, 3$ in each simulation.

The local support of the DG discretisation allows for the application of different polynomials and different number of quadrature points in different zones of the domain. Furthermore it is also possible to define the polynomial degree and the number of points of each element of the domain independently from each other. This property is intrinsic in the discontinuous features of the DG method. Since information is propagated between two elements only through their interface, the expansion within an element depends only upon its own values and the interface values of adjacent elements.

2   SENSOR-BASED P-ADAPTION STRATEGY

This study proposes an alternative strategy and the adaption procedure is applied after the current spatial distribution of polynomial degree, p, has converged to a steady solution. Moreover, the maximum degree is not imposed, but each element is free to assume any degree and the automatic p-adaptive strategy stops when a stable spatial p-distribution is reached. This method has been developed for modelling steady problems but it may be extended to time-dependent problems provided that an efficient method to vary the polynomial degree at each time step is implemented.

This procedure takes advantage of the ability of a discontinuity sensor used to quantify the difference between the actual solution (p) and the projected reduced one $(p-1)$ in order to vary the polynomial resolution in an element. The value of the sensor in an element is defined in the same way as described in [6]:

$$S_e = \frac{||\rho_e^p - \rho_e^{p-1}||_{L_2}}{||\rho_e^p||_{L_2}} \tag{1}$$

where ρ_e^p and ρ_e^{p-1} are the average solutions of degree p and $p-1$ respectively on the same element. The polynomial degree is decreased when a discontinuity is present in order to avoid oscillations and increased when a high gradient is identified to improve the accuracy. This procedure allows the simulation to adapt to the flowfield, increasing the accuracy of the solution only where needed and, as a consequence, reducing the computational cost required for solving the problem. Furthermore, this sensor is used to locally add an extra diffusion term to enable shock capturing as described in [5, 6].

Initially, a converged linear solution is obtained after which the sensor in each element is calculated. Based on the determined sensor value and the pre-defined sensor thresholds, the degree of the polynomial approximation in each element is increased, reduced or maintained and a new converged solution is obtained. The sensor distribution is divided into four zones:

$$
p_e = \begin{cases} p_e - 1 & \text{if } s_e > s_{ds} \\ p_e + 1 & \text{if } s_{sm} < s_e < s_{ds} \\ p_e & \text{if } s_{fl} < s_e < s_{sm} \\ p_e - 1 & \text{if } s_e < s_{fl} \end{cases} \tag{2}
$$

where s_{ds}, s_{sm} and s_{fl} are the threshold values to identify discontinuities, smooth and flat solutions respectively. This procedure is carried out iteratively.

In order to determine the solution at $p - 1$ and threat the numerical fluxes at the edges of the element accordingly, the solution at polynomial order p, determined using a modified basis, has to be projected onto a hierarchical orthogonal expansion basis. Hence, using a more general formulation, the solution of a variable u is expressed as:

$$
u^0 = u \tag{3}
$$

where u and u^0 represent the general solution obtained using a modified basis ($\mathbf{B}$) and orthogonal ($\mathbf{B}^0$) respectively, hence:

$$
\mathbf{B}^0 \hat{u}^0 = \mathbf{B}\hat{u} \rightarrow \hat{u}^0 = \left[\mathbf{B}^0\right]^{-1} \mathbf{B}\hat{u} \tag{4}
$$

where $\hat{u}$ represents the vector of coefficients at polynomial p^+. Since a hierarchical basis is used, it is possible to lower the polynomial order to p^- where $p^+ > p^-$. Since the coefficients are not coupled in the hierarchical orthogonal basis and the information about the mean is contained only in the first coefficient, it is possible to apply a cut-off filter to the orthogonal coefficient vector. This cut-off filter sets all the coefficients that are higher than p^- equal to zero. The information contained in the high frequency components is removed without altering the mean value. The orthogonal coefficients represent the solution at p^- using an orthogonal basis, hence the following transformation has to be applied to obtain the modified filtered coefficients of the lower polynomial degree:

$$
\hat{u}_f = \left[\mathbf{B}^{-1}\right] \mathbf{B}^0 \hat{u}_f^0 \tag{5}
$$

The solution ρ_e^{p-1}, is obtained from ρ_e^p using this post processing step.

When dealing with different polynomial degrees, it is important to ensure an adequate treatment of the two following operations: the change of the polynomial degree of the solution in one element and the computation of the numerical flux on the interface. Hence, after the sensor is applied and the polynomial order of the element is changed, a similar filtering procedure is performed to compute the advective numerical fluxes on the interface of two elements with different expansions since the appropriate number of quadrature points has to be used. The number of quadrature points has to be equal to the number used by the highest polynomial degree of the two adjacent elements to avoid numerical

instabilities [2]. To ensure conservation and stability, the continuity of the total flux is required and therefore:

$$\int_{\Gamma_{f-}} \mathbf{F}_-^u d\Gamma_f = \int_{\Gamma_{f+}} \mathbf{F}_+^u d\Gamma_f \tag{6}$$

Where $\mathbf{F}_-^u$ and $\mathbf{F}_+^u$ represent the numerical flux on the edge between two elements with a lower and a higher polynomial order respectively. If the order or the quadrature points is different, the coefficients are copied directly on the higher resolved side, but fewer coefficients have to be set on the other side. The interface flux is then projected on the space of orthogonal polynomials and then filtered to delete the high-order frequencies. Once the degree of the orthogonal expansion is decreased to the lower degree, a reverse projection is carried out and modified coefficients are found.

3 RESULTS

The performance of the p-adaptive method is illustrated for the solution of the transonic flow ($M = 0.8$) over a NACA0012 aerosol under an angle of attack of $\alpha = 1.25°$. Two shocks are generated on the aerofoil: a strong shock on the top, at about $x = 0.6$ and a weaker shock on the bottom of the aerofoil at $x = 0.3$. The reference C_p distribution used for comparison is taken from [8], in which the numerical solution is obtained with a finite volume method, the aerofoil wall is discretised by 320 cells and the farfield boundary is placed at 25 chords. Figure 1 depicts the density and Mach distribution around the aerofoil, the final spatial p distribution and the sensor distribution. Even though the grid is very coarse, the shock is well resolved, it is captured in only one cell and it does not create oscillations in the neighbour cells. The discontinuity sensor is active only at the shock waves and the rest of the flow field is diffusion free. The p-adaptive procedure increases the polynomial degree of the discretisation close to the aerofoil and maintains $p = 3$ on the shock in order to avoid oscillations of the solution. Since the mesh is very coarse, most of the error in the C_p calculation is introduced at the shock position on the top of the aerofoil. Since the p-adaptive procedure does not increase the polynomial degree in the elements where the shock is present.

The efficiency of the p-adaptive procedure comes from the reduction in the number of operations required to solve the equations, but also the initial condition of each $1 \leq p \leq p_{max}$ simulation is a converged solution obtained with a lower degree ($1 \leq p \leq p_{max} - 1$). However, the smaller CFL time restriction associated with p_{max} has to be imposed over all the domain, thus reducing the time step also in the regions with lower polynomial order. A possible improvement could be the application of a domain decomposition technique or variable timestepping to deal with different values ofΔ t through the domain.

Current work is performed on the extension of the illustrated p-adaption technique to time-dependent 3D problems in $Nektar^{++}$ provided that an efficient method is developed

to vary the polynomial degree at each time step. Furthermore, ongoing investigation is performed on the topic of shock capturing in 3D compressible flow and results of both topics will be discussed during the ADMOS conference.

(a)

(b)

(c)

(d)
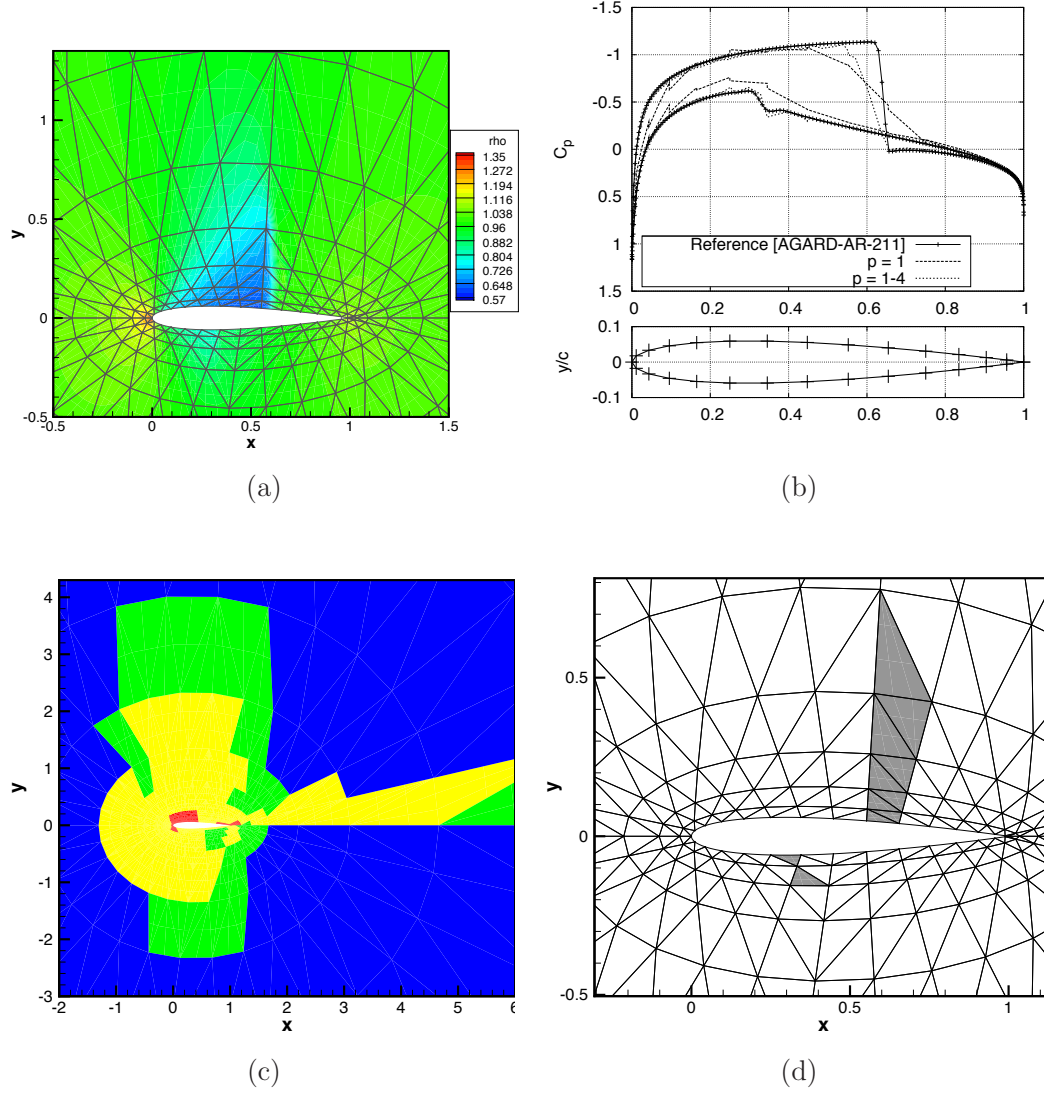
Figure 1: Solution of the inviscid transonic flow past a NACA0012 aerosol. (a) Density distribution; (b) Mach Distribution; (c) Polynomial degree distribution: blue: p = 1; green: p = ; yellow: p = 3; red: p = 4; (d) Sensor distribution.

REFERENCES

[1] Burbeau, A. and Sagaut P. *A dynamic p-adaptive Discontinuous Galerkin method for viscous flows with shocks.* Computers and Fluids, 34:401-417, (2005)

[2] Sherwin, S. *A high order Fourier/unstructured discontinuous Galerkin method for hyperbolic conservation laws.* Seventh International Conference on Hyperbolic Problems, Züric, 9-13 Februari (1998)

[3] Kubatko, B. and Bunya, S. and Dawson, C. and Westerink, J.J. *Dynamic p-adaptive Runge–Kutta discontinuous Galerkin methods for the shallow water equations.* Comput. Methods Appl. Mech. Engrg. 198:1766-1774, (2009)

[4] , Kardiadakis, G.E. and Sherwin, S. *Spectral/hp element methods for computational fluid dynamics.* Oxford Science Publications, (2005)

[5] Klöckner, A. and Warburton, T. and Hesthaven, J.S. *Viscous shock capturing in a time-explicit discontinuous galerkin method.* Math. Model. Nat. Phenom. 10:1-27, (2011)

[6] Persson, P-O. and Peraire, J. *Sub-Cell Shock Capturing for Discontinuous Galerkin Methods.* American Institure of Aeronautics and Astronautics, Paper 2006-0112, (2006)

[7] Biotto, C. *A Discontinuous Galerkin Method for the Solution of Compressible Flows.* Ph.D. Thesis, Imperial College London, (2011)

[8] Norstrud, H. and Boerstoel, J.W. and Jones, D.J.and Viviand, H. *Test cases for inviscid flow field methods.* AGARD Report AR-211, (1985)

MULTIOBJECTIVE AND MULTIPOINT OPTIMIZATION OF A HEAVY CLASS HELICOPTER ENGINE INSTALLATION USING EVOLUTIONARY ALGORITHMS

A. FABBRIS[1], A. GARAVELLO[2], M. RUSSO[3], R. PONZA[4], E. BENINI[5]

[1] Research Engineer , HIT09 S.r.l., Galleria Storione, 8 – 35100 Padova, Italy
e-mail: a.fabbris@hit09.com, www.hit09.com

[2] Research Engineer, HIT09 S.r.l., Galleria Storione, 8 – 35100 Padova, Italy
e-mail: a.garavello@hit09.com, www.hit09.com

[3] Research Engineer , HIT09 S.r.l., Galleria Storione, 8 – 35100 Padova, Italy
e-mail: m.russo@hit09.com, www.hit09.com

[4] Senior Research Engineer, HIT09 S.r.l., Galleria Storione, 8 – 35100 Padova, Italy
e-mail: r.ponza@hit09.com, www.hit09.com

[5] Associate Professor, Dept. of Industrial Engineering
University of Padova, Via Venezia, 1 – 35131 Padova, Italy
e-mail: ernesto.benini@unipd.it, www.unipd.it

Key words: Model Validation, CFD Coupled Solvers, Optimization Algorithms, Optimization Strategies, Industrial Application, Computing Methods.

Abstract. Aerodynamic design and optimization of engine installation is a pivotal part of the helicopter design process. To this purpose an adaptive, problem-independent and reliable optimization methodology would be particularly valuable in reaching such goal. The application of advanced evolutionary algorithms coupled with CFD solvers for the accurate flow solution of validated numerical models represents a very powerful tool for the parametric design and optimization of engine installation components. Within the JTI Clean Sky FP7 project "HeavyCopter" the consortium constituted by the University of Padova (UNIPD) and the spin-off company HIT09 developed an automatic optimization loop based on the home made genetic algorithm GeDEA, and applied it to engine installation design of a heavy-class helicopter, as well as to aircraft components optimization problems. This paper illustrates the application of the GeDEA-based optimization loop both at forward and hover reference flight conditions for such helicopter. The algorithm pursues the minimization of the total pressure losses at the inlets while keeping the flow distortion at the engine inlet at the lowest level; regarding the exhaust, the back-pressure is minimized in order to increase the power output of the engine while preserving the entrainment ratio. The results highlight significant improved performance margins in all the components.

1   INTRODUCTION

Within the Clean Sky framework, the joint technology initiative funded by the European Commission and industry, a specific activity is dedicated to the study of the engine installation of the AgustaWestland AW101 heavy helicopter. A consortium constituted by the University of Padova and the companies HIT09 and MDA submitted the HEAVYcOPTer project proposal in response to a specific Call on the subject [1]. The Call pertained a contribution and the supporting in accomplishing the aerodynamic optimisation of the intake and exhaust of the AW101 helicopter.

Efficient aerodynamic design of air intakes is a challenging objective for airframe manufacturers: inlet flow typically develops in adverse pressure gradient conditions, which leads to boundary layer instability and possible flow separation. Therefore inlet cross sectional area distribution along the central line should be optimized in order to minimize boundary layer "loading" and avoid separation [1]. In addition, for helicopter intake applications, an S-shaped duct is usually required to channel the air to the engine face; this is due to the presence of the engine shaft and the requirement for short and compact duct layout. From the fluid-dynamic point of view, a curved duct induces a secondary flow pattern, which essentially sets up pockets of swirling flow at the duct exit [2] and determines engines performance degradation [3]. In severe situations, these pockets can produce rotating stall instability of the compressor rotor [4]. Therefore, the internal shape of the curved duct should embody proper strategies in order to minimize total pressure loss and flow distortions at the engine face [5]. Finally, stability of boundary layer in turboprop and helicopter inlets may also be remarkably affected by the aircraft operating conditions and flight speed [6], [8], [8]. In such a context, CFD is a powerful tool which can be used to accurately evaluate the complex flow behaviour within inlet ducts: [10] and [11] are remarkable examples of CFD application to intake aerodynamics. When coupled with geometry parameterization techniques, CFD provides an effective automatic design methodology for inlet ducts.

Within the HEAVYcOPTer framework, the baseline intake CFD model has been built up and validated by means of a comparison against the available wind tunnel experimental data, starting from the existing AW101 engine installation geometry provided by AgustaWestland Ltd. via CATIA® CAD models. CFD analysis has been carried out for the nominal hover and forward flight cruise conditions; then, results have been analyzed in terms of total pressure losses, flow distortions, flow separations and all those aspects that affect the efficiency of the helicopter intake system. This analysis allowed to properly understand the aerodynamic behaviour of the actual design and to identify the most appropriate parametric changes to be applied to the geometry during the optimisation phase.

The baseline CFD solution and its associated parametric geometrical model are then the main inputs for the optimisation procedure selected, which involves the application of the GeDEA [12]. The GeDEA is the University of Padova home-made genetic algorithm able to perform multi-objective optimisation analysis with the general approach of the Pareto frontier search; it has been compared to others state of the art genetic algorithm with excellent results and, interfaced with flow solvers, it has been successfully used in several fluid-dynamics applications; in particular, within the clean-sky GRC2 research program [13], the GeDEA

based optimisation loop has been successfully applied to several fuselage and engine installation components of the European tilt rotor ERICA [13].

The results obtained by the application of the above mentioned optimisation chain on the AW101 engine installation are presented in this paper, with focus on the air intakes and exhaust number one optimisation outcomes.

2 DESCRIPTION OF THE OPTIMISATION METHOD

The aerodynamic optimisation procedure which has been implemented and used for the project HEAVYcOPTer is structured in three phases as follows:

- Baseline model preparation and simulation phase;

- Automatic optimisation phase;

- Post-processing and optimized CAD model reconstruction phase.

2.1 Baseline model simulation

Typically the starting point is represented by the CAD model of the baseline configuration. Starting from the geometrical model, the procedure moves into the "baseline simulation block" (see Figure 1), where the baseline configuration of the component under consideration is analyzed via CFD in terms of aerodynamic performance in the most relevant operating conditions. The assessment of the baseline solution allows the designer to proper understand the flow field characteristics of the object under analysis, gives fundamental indications for the optimisation objectives and constraints identification and make it possible to setting up the geometrical parametric model.
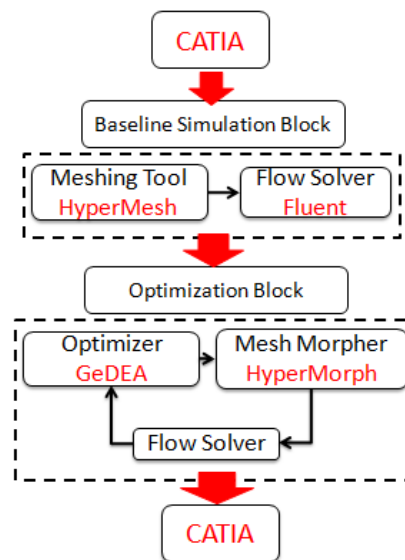
Figure 1: Optimisation method flow-chart

223

2.2 Automatic optimisation execution

When the preliminary operations have been completed, the optimisation can be carried out by means of the automatic optimisation loop in Figure 1: it is constituted by the following components:

- GeDEA (Genetic Diversity Evolutionary Algorithm): it is an advanced multi-objective optimisation algorithm developed at the University of Padova [12]. It is the selected optimisation engine;
- Altair HyperMorph®: it makes possible to convert the design parameters coming out from GeDEA into morphed CFD cases, suitable for the objective function evaluation;
- Ansys Fluent®: the selected flow solver; it takes in input the morphed CFD cases coming from HyperMorph® and gives back to GeDEA the corresponding values of the chosen objective functions.

During the optimisation process, GeDEA lets a population of individuals "evolve" (each one corresponding to a different set of design variables and so to a different geometrical configuration) until the convergence to the Pareto optimal frontier has been reached, being the Pareto frontier the set of non-inferior solutions, which represents the solution of a multi-objective optimisation problem; a non-inferior solution, also called Pareto optimal or non-dominated solution, is one in which an improvement in one objective requires the degradation of another [15].

2.3 Post-processing

The Pareto frontier in output from the automatic optimisation loop represents a multiple set of solutions equally optimal according to the Pareto concept but of course different form the aerodynamic and engineering point of view. In fact each solution over the Pareto frontier may present advantages and drawbacks with respect the other solutions. In order to choose among the optimal set the most appropriate solution a post-processing is necessary. Thanks to the intrinsic multi-objective approach adopted, the designer is allowed to select, among the Pareto optimal set, the solution which is more suitable for his needs: for example, choosing to privilege the improvement of one objective with respect to the other or even including other considerations such as non-aerodynamic requirements. The strength of the selected approach is that the designer can choose the proper trade-off between the objectives when the optimisation work has been completed and he is not forced to introduce his arbitrariness in the problem set up, as commonly happens using traditional optimisation approaches.

3   AW101 ENGINE INSTALLATION DESCRIPTION

The AW101 engine intake system is constituted by three side intake ducts feeding the three helicopter engines; from now on, we will refer to air intakes as "intake#1" and "intake#3" for the two symmetrical intakes on the fuselage sides, and "intake#2 for the intake placed at the top of the fuselage roof (Figure 2).

224
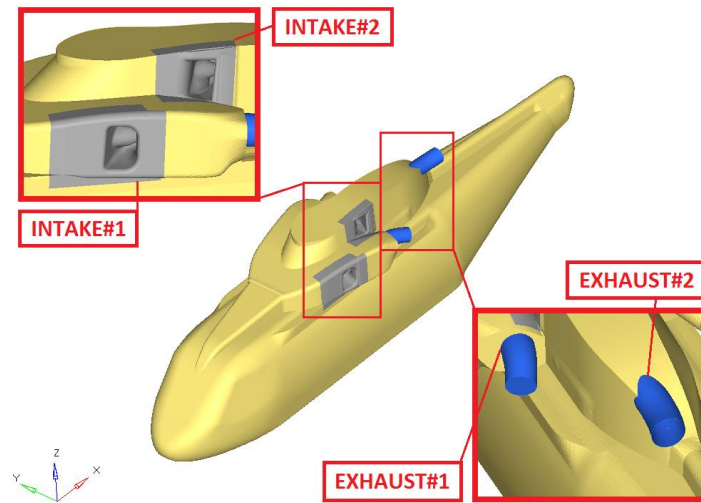
Figure 2: View of the engine installation on the AW101 CAD model.

An intake duct is an S-shaped duct connecting the side entry section with the engine compressor face, commonly referred as *Aerodynamic Interface Plane* (AIP). CAD layouts of engine#1 and engine#2 bay internal components were provided by AgustaWestland Ltd in order to allow the definition of geometrical modifications of the duct surfaces so as to be compliant with the installation architectural constraints (Figure 3).
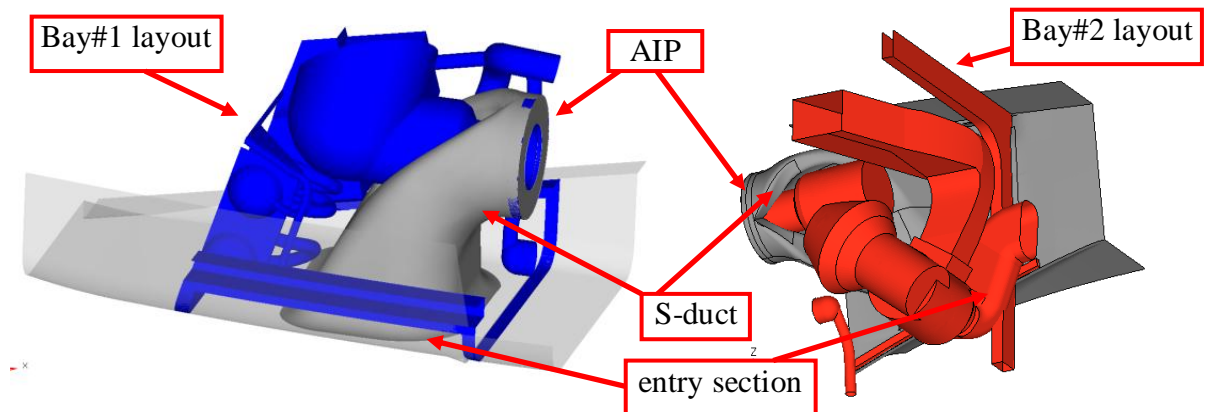
Figure 3: Internal view of intake S-duct and bay internal components layout for engine#1 and engine#2.

AW101 engine bay and exhaust are replaced by a simplified rig configuration, which was tested at the AWL wind tunnel facility in order to get cold flow data on current engine system and to validate CFD models. The CAD layout of the exhaust internal components was provided by AgustaWestland Ltd in order to allow the definition of geometrical modifications of the daisy nozzle and central body surfaces so as to be compliant with the installation architectural constraints (Figure 4). The swirl generator blades are replaced by a flat surface

normal to the main flow direction, where a FAN boundary conditions is applied in order to reduce mesh size and to make the model more representative of the real exhaust installation. Only the primary nozzle and the central body installed on Engine#2 is optimised.
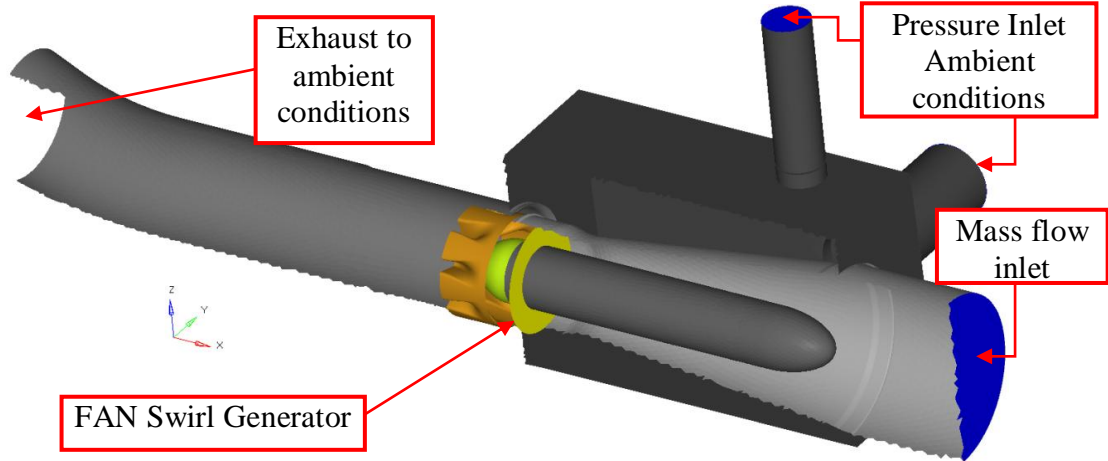
Figure 4: Boundary Conditions applied to full scale Rig model.

4 SET UP OF THE PARAMETRIC MODEL

Once the main geometrical features characterising the baseline designs are identified, design parameters are generated for the complete geometrical control of the intake#1 and intake #2 duct shapes and of the exhaust daisy nozzle and central body shapes. Those parametric shapes are generated using the Altair software HyperMesh® by means of the mesh morphing and parameterisation techniques available within the morphing toolbox HyperMorph®, utilizing a combination of different free form techniques available within the tool. When applied, the nodes displacements can be saved as perturbation vectors and then be reapplied to the baseline model with any given scaling factor. Shape scaling factors become then the design variables for the optimisation problem; the morphed geometry results therefore from the linear combination of the user defined shapes multiplied by their own scaling factors:

$$v = \sum_{i=1}^{n} \alpha_i Sh_i \qquad (1)$$

where:

- v is the global displacement vector;
- Sh_i are the ith basic shapes defined within HyperMorph®.
- α_i is the ith shape scaling factor generated by GeDEA.
- n is the number of parameters for the current application.

During the automatic optimisation process, the scaling factor α_i represents the set of design parameters controlled by the genetic algorithm.

Some examples of shapes are shown in Figure 5 and Figure 6.
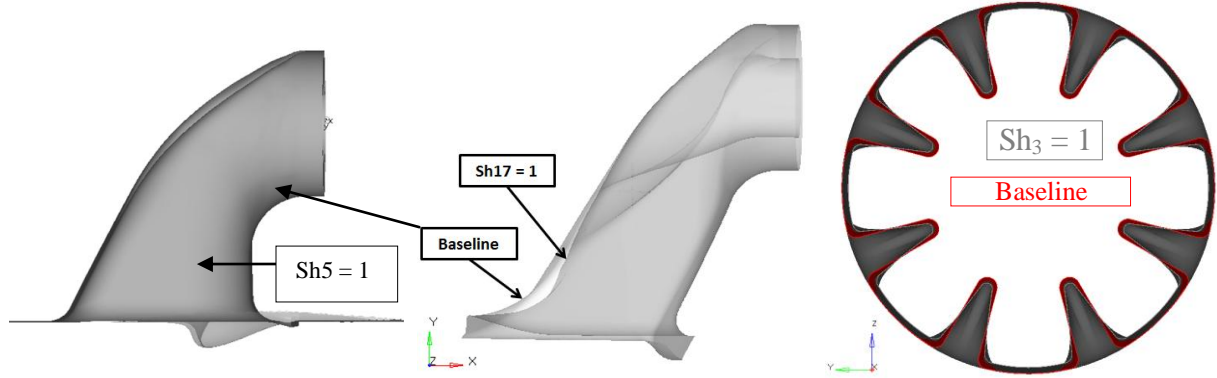
Figure 5: Examples of shapes respectively for intake#1, intake#2 and exhaust#2
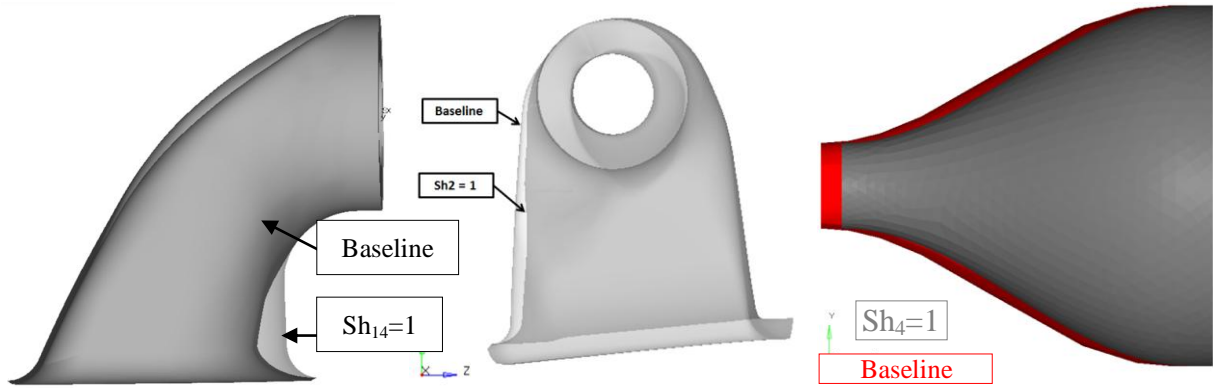
Figure 6: Examples of shapes respectively for intake#1, intake#2 and exhaust#2

5 FORMULATION OF THE OPTIMISATION PROBLEM

The GeDEA-based optimization loop has already been successfully applied to several fuselage and engine installation components of the European tilt rotor ERICA [14]: the interested reader can find an extensive description of the main achievements of the Clean Sky GRC2 projects CODETilt [16] and TILTOp [17] in the conference papers [18], [19] and [19]. Two different objective functions are formulated to pursue the HEAVyCOPTer optimization, one for the air intakes and one for the exhausts, respectively.

The former is a two-objective and two-component vector function; it is obtained from the sum of the total pressure loss term and a *penalty function* term. It can be formally expressed as:

$$minimize\ \{G(x) = [F(x) + PF(x)]\} \tag{2}$$

Where $F(x)$ accounts for aerodynamic total pressure loss (ΔP_T) within the intakes at the two reference flight conditions:

$$F(x) = \begin{bmatrix} \Delta P_T(x)\,|\, @\, hover \\ \Delta P_T(x)\,|\, @\, cruise \end{bmatrix} \tag{3}$$

The penalty function $PF(x)$ introduces a functional constraint on the flow distortions at the engine inlet by worsening the score of a new configuration with an additional term that is proportional to the DC60 factor difference with respect to the baseline configuration:

$$PF(x) = \begin{bmatrix} 0 & \text{if } DC60(x) \le DC60_{baseline} \\ \beta \left| \dfrac{DC60(x) - DC60_{baseline}}{DC60_{baseline}} \right|^{\gamma} & \text{if } DC60(x) > DC60_{baseline} \end{bmatrix} \tag{4}$$

where the coefficients β and γ control the intensity and shape of the penalty function respectively. Again, this term is evaluated at the two flight conditions.

The design variables vector, $\mathbf{x}$, is given by the set of scaling factors, subject to the variable bounds which will differ between Intake#1 and Intake#2:

$$\mathbf{x} = [\ \alpha_1 ... \alpha_n] \tag{5}$$

The exhausts objective function $E(\mathbf{x})$ is a bi-objective two component vector function, evaluated at the forward flight condition only:

$$\text{minimize} \left\{ E(x) = \begin{bmatrix} BP(x)\,|\, @\, cruise \\ |ER(x) - ER_{baseline}|\,|\, @\, cruise \end{bmatrix} \right\} \tag{6}$$

where $BP(\mathbf{x})$ and $ER(\mathbf{x})$ represents the back-pressure and the entrainment ratio respectively:

$$BP(x) = \frac{P_{T,exhaust-inlet} - P_s}{P_s} \,;\, ER(x) = \frac{\dot{m}_{cold}(x)}{\dot{m}_{hot}} \tag{7}$$

where P_s represents the free-stream static pressure.

As mentioned before, $G(\mathbf{x})$ and $E(\mathbf{x})$ are evaluated and passed to the algorithm by means of CFD simulations of the individual $\mathbf{x}$.

6 SUMMARY OF THE OPTIMISATION RESULTS

The both intakes and exhaust#2 optimisation results are discussed within in this section: the optimization loop had completed 7, 5 and 5 evolutionary generations on Intake#1, Intake#2 and Exhaust#2 respectively. Remarkable improvements on the objective functions are achieved. Figure 7, Figure 8 and Figure 9 show the final Pareto frontiers calculated by the GeDEA algorithm for Intake#1, Intake#2 and Exhaust#2 respectively: despite the number of generations is relatively small, significant improvements in both hover and forward flight objective functions can be observed.
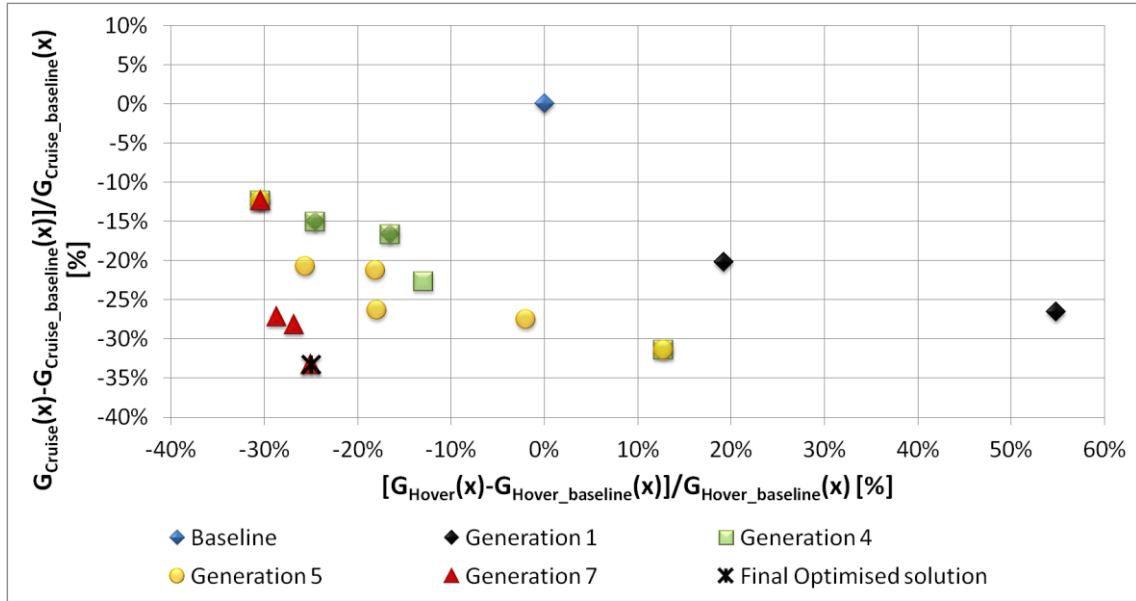
Figure 7: Intake#1 GeDEA Pareto frontier, 7th generation and evolution of the Pareto front through the generations; the selected optimal individual is highlighted.
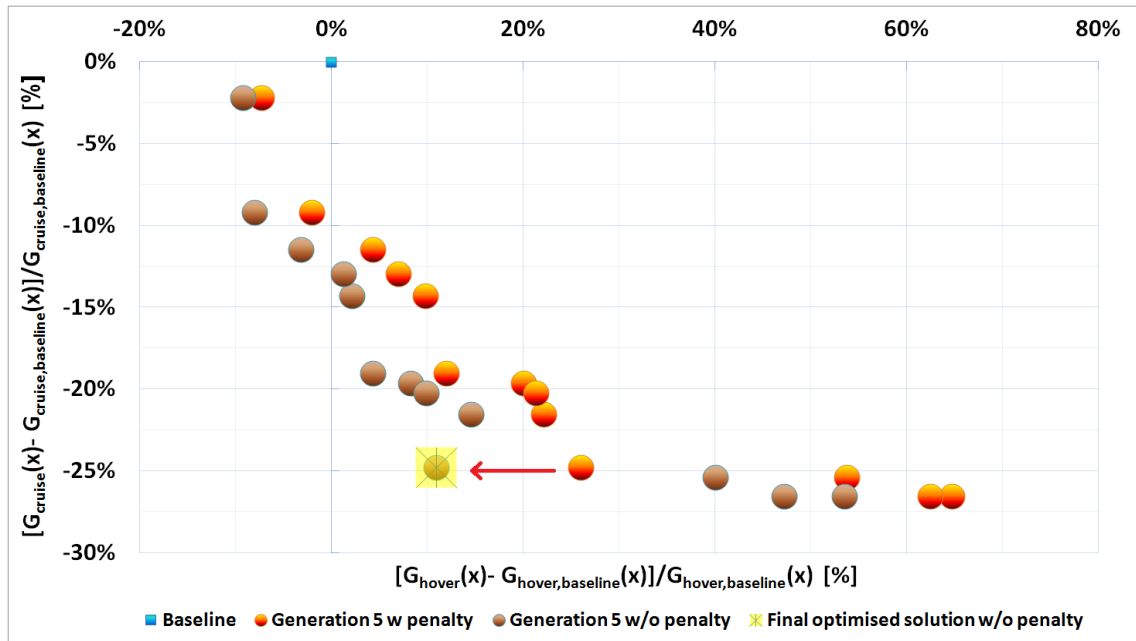
Figure 8: 5th generation Intake#2 GeDEA Pareto frontier, including (orange dots) and excluding (brown dots) the penalty function (**Error! Reference source not found.**) in the computation of the fitness value. The no-penalty fitness score of the optimised solution is highlighted in yellow.
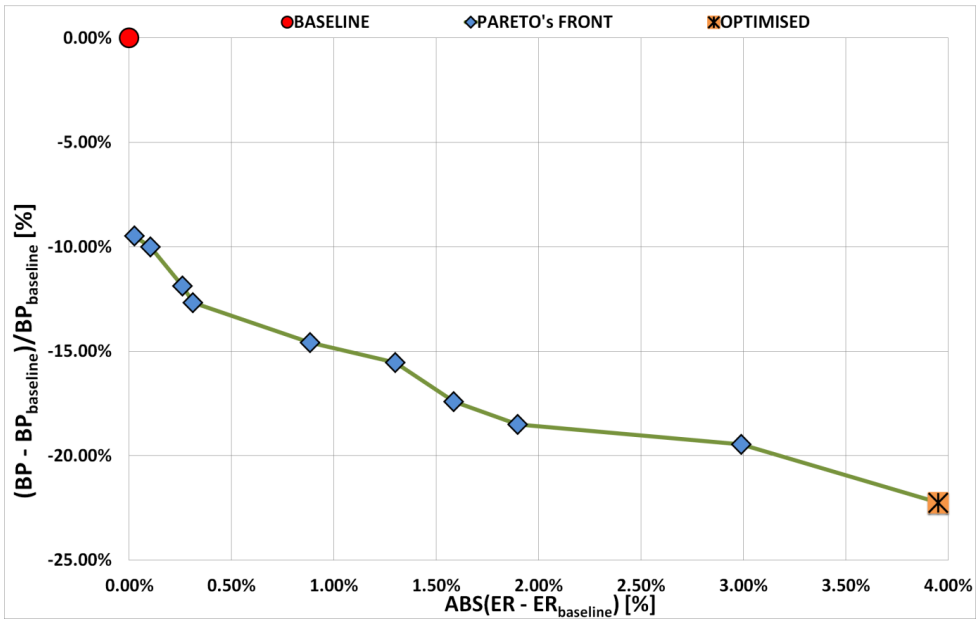
Figure 9: 5th generation Exhaust#2 GeDEA Pareto frontier; the selected optimal individual is highlighted.

Hover: total pressure distribution along the duct

Hover: AIP total pressure distribution
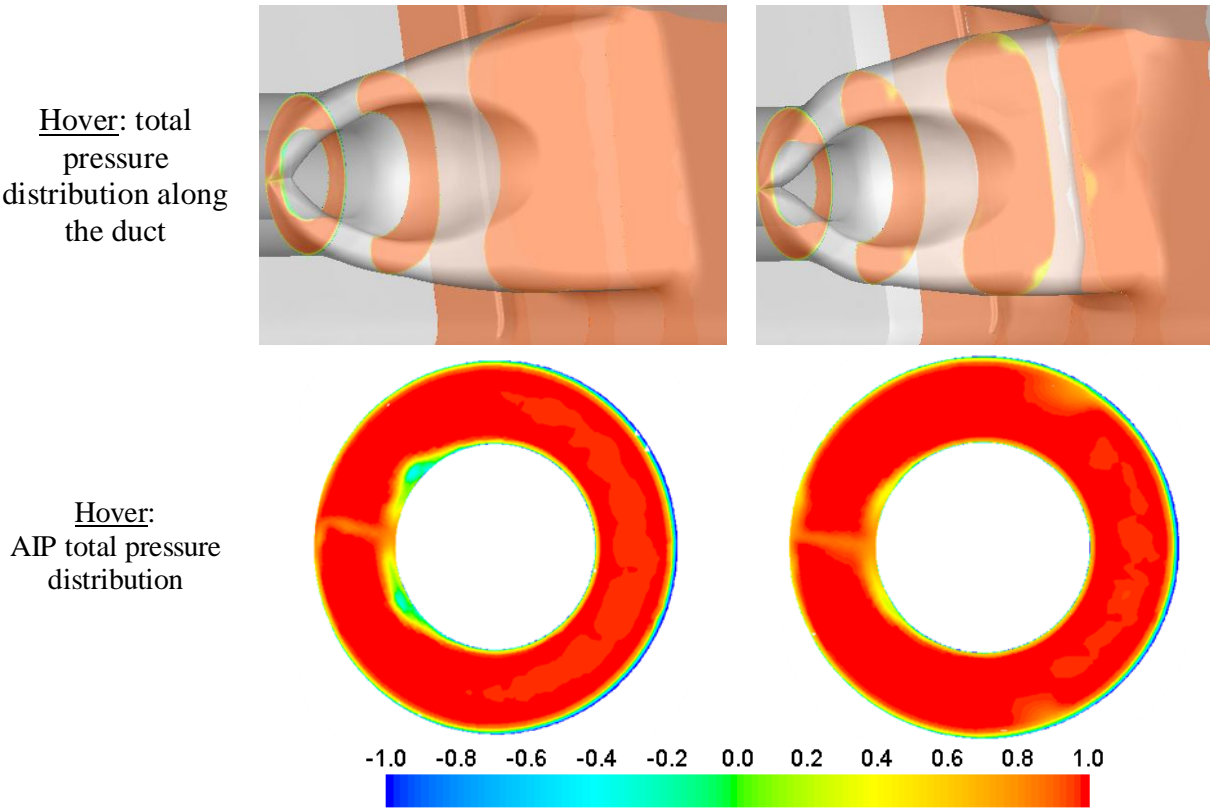
Figure 10: Hover AIP Total Pressure distribution comparison (normalised by free stream total pressure value) for intake#1; baseline (left) and optimal solution (right).

Forward flight: total pressure distribution along the duct

Forward flight: AIP total pressure distribution
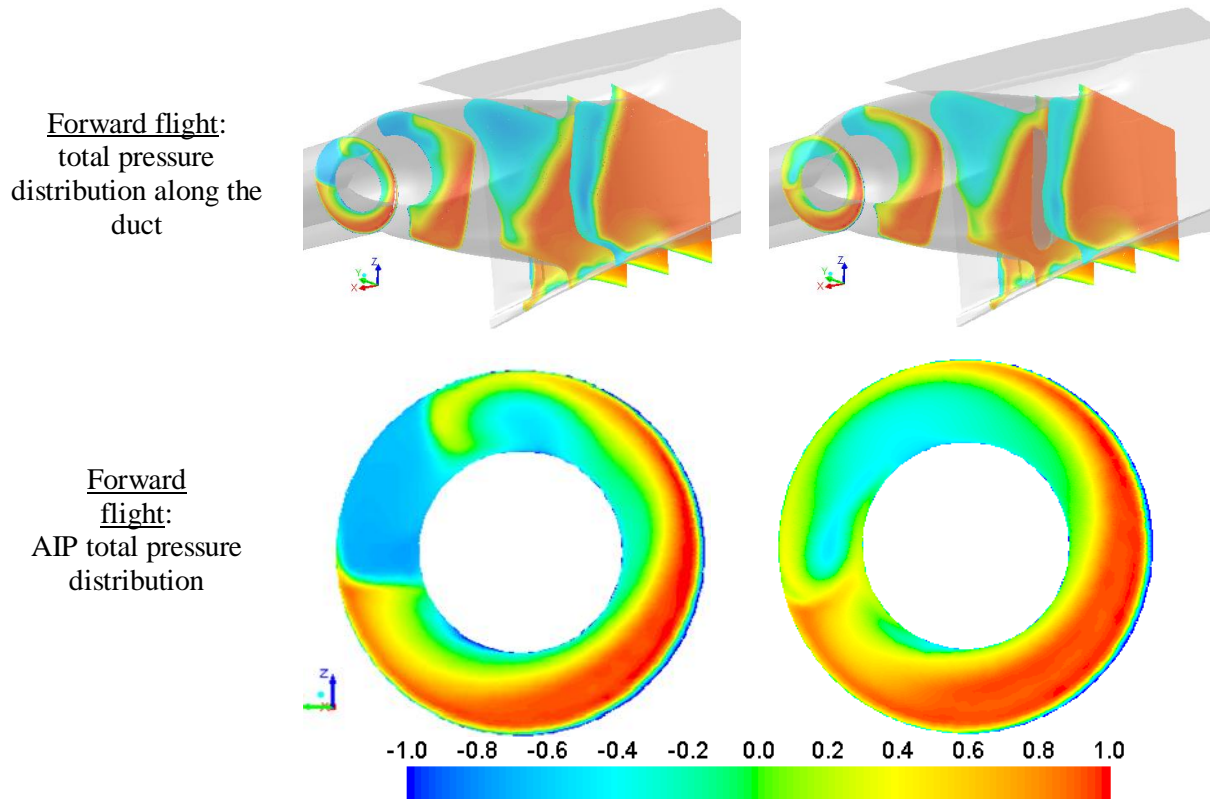
Figure 11: Forward flight AIP Total Pressure distribution comparison (normalised by free stream total pressure value) for intake#2; baseline (left) and optimal solution (right).
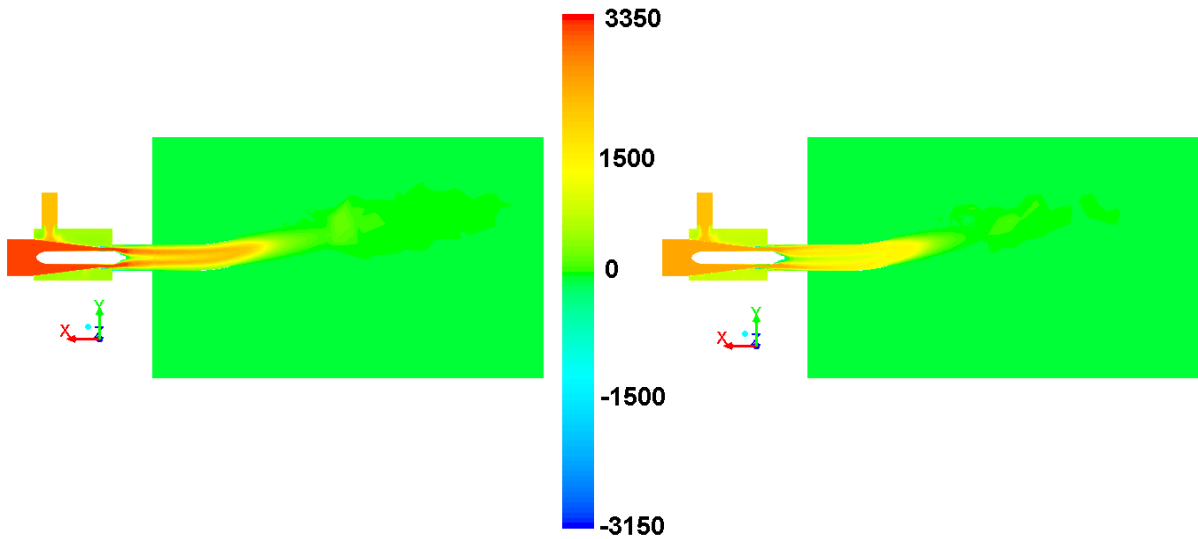
Figure 12: Forward flight Total Pressure [Pa] contours over x-y plane section comparison for exhaust#2; baseline (left) and optimal solution (right).
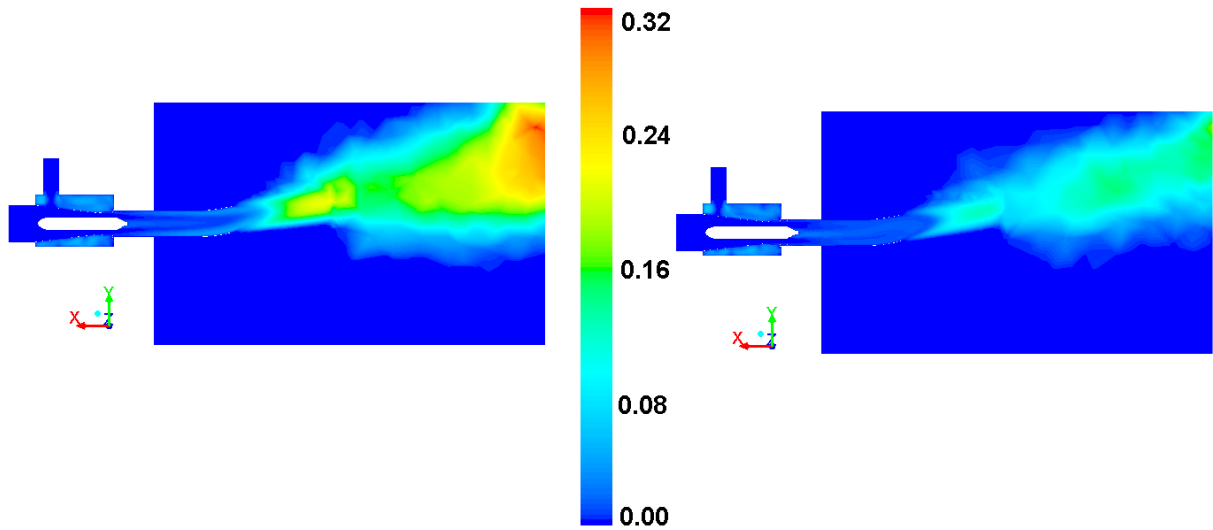
Figure 13: Forward flight Turbulent Viscosity [kg/s] over x-y plane section comparison for exhaust#2; baseline (left) and optimal solution (right).

7 CONCLUSIONS

The selected optimised geometry for Intake#1 displays a -25/-33% total pressure loss reduction from the baseline in hover/forward flight respectively, without any detrimental flow distortion effect.

The cruise-optimized geometry of Intake#2 does worsen the hovering performances to a limited extent, while significantly improving the forward flight efficiency: a +10/-25% total pressure loss drop is obtained (hovering/cruise).

Back-pressure on Exhaust#2 was lowered by the 22% from the baseline, while maintaining the entrainment ratio (which increases by the 4%).

The paper demonstrate the strength of the parametric approach chosen: the genetic algorithm GeDEA provides an efficient search procedure for alternative designs and optimal solutions while the morphing technology adopted allows solution compatibility with feasibility considerations and industrial constraints.

ACKNOWLEDGMENTS

REFERENCES

[1] CleanSky JTI call for proposal JTI-CS-2010-4-GRC-02-003, *Contribution to optimisation of heavy helicopter engine installation design.*, available on http://www.cleansky.eu/content/homepage/calls, Call SP1-JTI-CS-2010-04, pag. 47.

[2] Seddon J. and Goldsmith L. *Intake Aerodynamics*. Blackwell Science, 2 edition, 1999.

[3] Seddon, J., "*Understanding and countering the swirl in S-ducts: tests on the sensitivity of swirl to fences*," Aeronautical Journal, R.Ae.Soc., 1984.

[4] Aulehla, F., Schmitz, D.M., "*Intake swirl and simplified methods for dynamic pressure distortion assessment*," in Intake Aerodynamics, VKI Lecture Series, 1988.

[5] Farokhi, S., *Aircraft propulsion*, John Wiley & Sons, Inc. 2009.

[6] Guo, R. W. and Seddon, J., "*Some unsteady flow characteristics of two S-shaped intake models tested at high incidence*," University of Bristol, Department of Aeronautical Engineering, Report No. RWG/JS/3/82, 1982.

[7] Anderson, B.H., "*Inlets, Ducts and Nozzles*," Proceedings of Aeropropulsion '87, NASA CP-3049, 1987.

[8] Sinha, R., "T*ransient Inlet Nacelle Code*," AIAA Journal of Propulsion and Power, 4, pp. 252-255, 1988.

[9] Saha, K., Singh, S.N. and Seshadri, V. (2007*). Computational Analysis on Flow Through Transition S-Diffusers: Effect of Inlet Shape,* AIAA, Journal of Aircraft, Vol. 44, No.

[10] Robichaud, M. P., Di Bartolomeo, Heikurinen, W., Habashi, W.G. (1997). *Turboprop air intake design using 3-D viscous analysis,* AIAA, Aerospace Sciences Meeting & Exhibit, 35th, Reno, NV.

[11] Zhang, W. L., Knight, D. D., Smith, D. (2000*). Automated Design of a Three-Dimensional Subsonic Diffuser,* Journal Of Propulsion And Power, Vol. 16, No. 6.

[12] Benini, E. and Toffolo, A. *Genetic diversity as an objective in multi-objective evolutionary algorithms*. Evolutionary Computation, MIT press journal, 11(2):151–167, 2003.

[13] Clean Sky website: http://www.cleansky.eu/content/page/reduced-drag-airframe-non-lifting-rotating-parts-grc2.

[14] Fabio Nannoni, Giorgio Giancamilli, and Marco Cicalè. *Erica: the European advance tilt-rotor*. 27th European Rotorcraft Forum, Moscow, Russia, 11th – 14th September 2001.

[15] Deb K. *Multi-Objective Optimisation Using Evolutionary Algorithms*. Wiley, 2001

[16] CleanSky JTI call for proposal JTI-CS-2010-1-GRC-02-004, *Contribution to design optimization of tiltrotor for drag reduction (fuselage/wing junction, nose and landing gear sponsons, empennage)*, available on http://www.cleansky.eu/content/homepage/calls, SP1-JTI-CS-2010-01, pag. 57.

[17] CleanSky JTI call for proposal JTI-CS-2009-2-GRC-02-001, *Contribution to the study of the air intake and exhaust integration into a tiltrotor nacelle*, available on http://www.cleansky.eu/content/homepage/calls, SP1-JTI-CS-2009-02, pag. 34.

[18] Garavello, A., Benini, E., Ponza, R., Scandroglio, A., Saporiti, A., *Aerodynamic Optimization of the ERICA Tilt-Rotor Intake and Exhaust System*, Proceeding of the 37th European Rotorcraft Forum, September 13th –15th, 2011, Ticino Park, Italy, Aerodynamic Session, paper ID 191.

[19]   Garavello A., Benini E., Ponza R., "*Multi-Objective Aerodynamic Design of Tilt-Rotor Airframe Components by means of Genetic Algorithms and CFD*", 68th American Helicopter Society Forum, May 1th – 13th, 2012, Fort Worth, Texas (USA).

[20]   Miste', G., Nibale, T., Garavello, A., Benini,E. *Assessment of the Engine Installation Performance of a Redesigned Tiltrotor Intake/Exhaust System,* 68th American Helicopter Society Forum, May 1th – 13th, 2012, Fort Worth, Texas (USA).

A FULLY COUPLED FINITE VOLUME SOLVER FOR THE SOLUTION OF INCOMPRESSIBLE FLOWS ON LOCALLY REFINED NON-MATCHING BLOCK-STRUCTURED GRIDS

Ulrich Falk, Michael Schäfer

Institute of Numerical Methods in Mechanical Engineering
Technical University Darmstadt
Dolivostraße 15, 64293 Darmstadt, Germany
e-mail: falk/schaefer@fnb.tu-darmstadt.de, www.fnb.tu-darmstadt.de

Key words: computational fluid dynamics, finite volume method, pressure-based coupled solver, block-structured grid, non-matching block interface

Abstract. A fully coupled solver for the solution of steady laminar incompressible flow problems on locally refined non-matching block-structured grids that promises improved convergence properties is presented. For this a coupled velocity-pressure algorithm developed by Darwish [1] that solves the momentum and pressure equations simultaneously is extended correspondingly. The spatial finite-volume discretisation applied is of second-order accuracy. All blocks are implicitly coupled and the method is fully conservative.
The newly developed method is verified via comparisons with manufactured solutions. Its performance is evaluated by systematic comparisons with standard segregated pressure-correction solution techniques for representative test cases.

1 INTRODUCTION

Incompressible flows of Newtonian fluids can be described mathematically by the non-linear Navier-Stokes equations. To handle these equations in the present approach numerically, they are discretised with a finite volume method on collocated block-structured grids.
Geometrically complicated flow configurations demand highly adapted grids in order to achieve the required numerical accuracy. In general the grid should be very fine in regions with larger variations of the dependent variables. Choosing the grid resolution of a block-structured grid such as large variations of the dependent variables are sufficiently resolved, can lead to high densities of grid points in regions where they are not required. One possibility to avoid regions with too high grid resolution are local grid refinement approaches. Various strategies for local grid refinement have been proposed in literature [3].

The present approach constitutes a h-method where the grid cells are obtained by dividing the control volumes (CVs) into four subcells in two dimensions and eight subcells in three dimensions. The approach can be easily extended to an arbitrary refinement of grid cells. A block-wise refinement of the block-structured grid is performed.

There are mainly two strategies to couple the local refinement region with the non-refined regions: The first one treats the refinement region as a different grid level and the result on a coarse level serves as a boundary condition and an initial guess to the computation of a finer level. In case of a rather complex multigrid datastructure is already used such approaches are fairly easy to implement due to the preservation of the grid structure and straightforward treatment of the coarse-fine interface. Based on the idea of Berger and Collela [4] Quirk [5] successfully implemented an hierarchical adaptive grid refinement approach for compressible flows.

The second strategy couples refined and non-refined regions at the same level, computing the whole grid simultaneously [6, 7]. Special discretisation schemes are required for the coarse-fine grid interface to ensure a proper coupling of the subdomains. The presented algorithm follows this second strategy.

There are several ways by which blocks can be interconnected. Arbitrary overlapping blocks connected to each other can be generated during a grid generation process (so-called Chimera grids). With these Chimera grids it is difficult to ensure conservation. Interpolation between overlapping regions may have to be constructed problem dependent, restricting the generality of the algorithm.

In the presented approach the internal block boundaries are patched together, making them share a common interface line, but allowing a different point distribution for each block. Compared to Chimera grids redundant regions are avoided. These patched grids are also called zonal grids or block adaptive grids.

Besides the choice of the grid the velocity-pressure coupling algorithm is an essential part with respect to the efficiency and robustness of a solution algorithm for the Navier-Stokes equations. There are mainly two strategies to perform the velocity-pressure coupling on collocated grids, either a segregated or a coupled approach. In the segregated approach, the system of equations for all variables are solved sequentially using fixed values from the last iteration of other dependent variables. A well known representative is the SIMPLE algorithm [8].

In the coupled approach all discretised equations of all dependent variables are solved in one system. Pressure based coupled solvers can be divided into two groups. In the first group the Navier-Stokes equations are discretised in a straightforward manner, i.e., no pressure variable in the mass-conservation equation is introduced. An example for this group is Vankas [10] symmetric coupled Gauss Seidel algorithm. These approaches lead to an ill-conditioned system of equations because of the present zeros in the main diagonal of the continuity equation. The solution of these stiff algebraic equation systems is rather difficult.

In the second group, which includes the approach presented here, a pressure equation is

derived in the same way as in the SIMPLE algorithm. Examples of this group are Lonsdale's [11] control-volume finite element method and the method of Webster [12]. In the presented approach the velocity-pressure coupling on collocated structured grids developed by Darwish [1] is extended to non-matching block-structured grids. The extention allows the specific adaptation of the numerical grid for each flow configuration.

The local block refinement (LBR) method described in this paper is aimed at increasing accuracy and efficiency in the computation of flow problems. The general discretisation procedure for the coupled approach is first outlined in the next section, followed by a description of the refinement method in section 3. In section 4 the LBR method is verified via Manufactured Solutions. In section 5 the performance is evaluated by systematic comparison with standard segregated pressure-correction solution techniques for representative test cases.

2 DISCRETISATION PROCEDURE

Laminar incompressible steady flows of Newtonian fluids can be described by the following nonlinear partial differential equations describing conservation of mass and momentum:

$$\frac{\partial u_i}{\partial x_i} = 0 \tag{1}$$

$$\frac{\partial(\rho u_i\, u_j)}{\partial x_j} = \frac{\partial}{\partial x_j}\left[\mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)\right] - \frac{\partial p}{\partial x_i} + \rho f_i \tag{2}$$

where u_i denotes the components of the velocity vector, x_i is the vector of Cartesian coordinates, μ the dynamic viscosity, p the pressure and f_i the outer force vector.

The governing equations (1) and (2) are solved via the block-structured finite volume method, whereby the flow domain is divided into blocks and each block is discretised by a structured grid. Each control volume is associated with a main grid point at its geometric center. To calculate the balance of the conserved quantities, the governing equations are integrated over each CV. Through the use of the divergence theorem the volume integrals of the convection, diffusion and pressure gradient terms are transferred to surface integrals, which, with the application of the continuity equation and application of approximation schemes of second order accuracy for the surface and volume integrals (mid-point rule), results in:

$$\sum_{f=nb(P)} \rho_f u_{i,f} n_{i,f} S_f = 0 \tag{3}$$

$$\sum_{f=nb(P)} \left[\rho_f\, u_{i,f}\, u_{j,f}\, n_{j,f} - \mu\left(\frac{\partial u_i}{\partial x_j}\right)_f n_{j,f} + p_f\, n_{i,f}\right] S_f = \rho\, f_{i,P}\, V_P \tag{4}$$

where variables with subscript f are CV-face variables, variables with subscript P are CV-center variables, S_f denotes the face area, n_f the unit cell-face normal vector and V_P

the cell volume.

To yield an algebraic equation system for the momentum equations the variation of the dependent variables and its derivatives have to be expressed in terms of grid-point center values. The diffusive fluxes are discretised by a central differential scheme (CDS), the convective fluxes are treated with a so called deferred correction approach, i.e. a first-order-upwind approximation (UDS) is used to calculate the elements of the coefficient matrix while the explicitly calculated difference between the UDS and CDS approximation is added on the right hand side of the equation system. The overall approximation order (formal order of accuracy) is therefore second order.

Starting point for the derivation of the pressure equation is the equation of mass conservation. The mass flow through a CV-face emerges from the product of face-velocity, density and the face-area. The aim is to express the face-velocity by means of pressure and center-point-velocities. This can be achieved by the construction of a pseudo-momentum equation at the CV-face. Therefore two momentum equations at, e.g. CV-mid-point P and the adjacent CV-mid-point F, are Rhie-Chow interpolated [13]. The face-velocity is obtained as

$$u_{f,i} = \underbrace{\overline{u_{f,i}}}_{\text{linear interpolated velocity}} - \overline{D_f} \underbrace{\left[\left(\frac{\partial p}{\partial x_i} \right)_f - \left(\overline{\frac{\partial p}{\partial x_i}} \right)_f \right]}_{\text{correction term}} \tag{5}$$

where $\overline{D_f}$ is the quotient of the cell volume and the corresponding coefficient of the main diagnonal of the discretised momentum equation (a_p^u, a_p^v, a_p^w). The variables with overbar in equation (5) are linearly interpolated cell face values from the neighboring cell centers. A detailed explanation of the derivation of the pressure equation can be found in [1]. The continuity equation then becomes

$$\sum_{f=nb(P)} \rho_f \left[\overline{u_{f,i}} - \overline{D_f} \left(\frac{\partial p}{\partial x_i} \right)_f \right] n_{i,f} S_f = - \sum_{f=nb(P)} \rho_f \overline{D_f} \left(\overline{\frac{\partial p}{\partial x_i}} \right)_f n_{i,f} S_f. \tag{6}$$

The variables on the left hand side of equation (4) and (6) are treated implicitly. This is the cornerstone of the coupled algorithm and can accelerate the convergence.

The discretised momentum equations (4) are linearized using values for mass fluxes from the previous iteration. The PETSc linear algebra library [14] (preconditioned Generalized Minimal Residual Krylov method) is used to solve the resulting sparse linear system.

3 LOCAL BLOCK REFINEMENT

At the interior of each block each CV has four neighbors in two dimensions or six neighbors in three dimensions which share a common face. At non-matching block interfaces this is not necessarily the case (see Figure 3). The aim is now to solve the conservation equations on a global grid, while the non-matching block interfaces should be treated
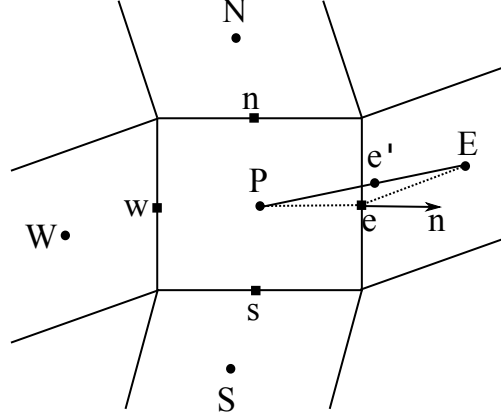
Figure 1: A typical CV and notation used.

implicitly, i.e. not by means of boundary conditions. The non-matching block interface treatment described below is based on Lilek's work [6] and was adjusted accordingly for the coupled approach of Darwish [1]. Since both surface and volume integrals are approx-
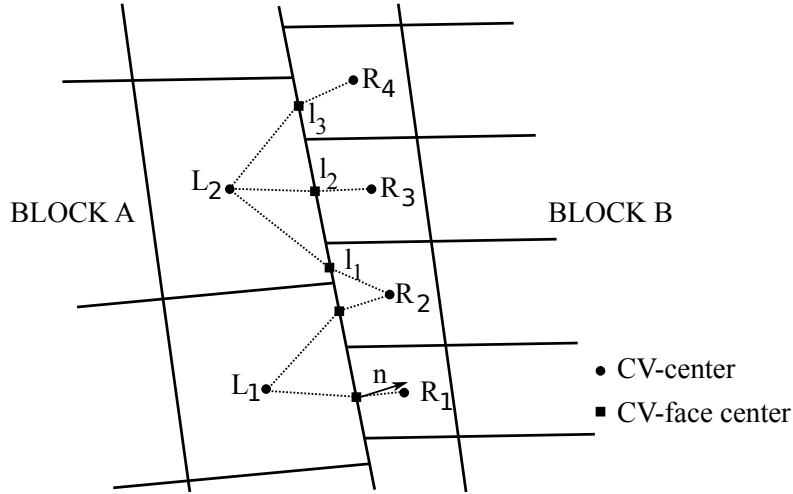
Figure 2: A typical non-matching block interface in two dimensions.

imated using the mid-point rule, nothing has to be changed for the calculation of volume integrals when non-matching block interfaces are present. However some adaptions have to be performed in order to approximate the surface integrals. When non-matching block interfaces are present a CV can have several adjacent neighboring CVs (see Figure 3, L2 has R2, R3 and R4 as neighbor) at the interface. According to the number of adjacent CVs, face value approximations (see Figure 3, l_1, l_2, l_3) for the convective, diffusive and pressure terms in the equations (4) and (6) have to be calculated. Details on various options to calculate these approximations can be found in Ferziger and Peric [15]. Here

only methods used in the present approximations will be described for the east side. The other CV-faces are treated accordingly. CV-face variables (u,v,w,p) are approximated using linear interpolation with a correction term to restore second-order accuracy on skewed grids (see Figure 1):

$$\sigma_e \approx \sigma_E \, \gamma_e \, + \, \sigma_P \, (1 - \gamma_e) \, + \, (\text{grad} \, \sigma)_{e'} \cdot (r_e - r_{e'}) \tag{7}$$

with the linear interpolation factor γ_e, the linearly interpolated CV-face gradient in the CV-center $(\text{grad} \, \sigma)_{e'}$ and the position vectors r_e and $r_{e'}$. The CV-center gradient can be calculated explicitly using the midpoint-rule approximation based on the Gauss theorem:

$$\left(\frac{\partial \sigma}{\partial x_i} \right)_P \approx \frac{\sum_k \sigma_k \, S_k^i}{V} \quad \text{with} \quad (k = e, w, n, s, \dots) \tag{8}$$

where σ_k is calculated in the same way as in equation (7). For the diffusive term the following second order accurate approximation is applied:

$$(\text{grad} \, \sigma)_e \cdot n_e \approx \frac{\sigma_E - \sigma_P}{|r_E - r_P|} - \overline{(\text{grad} \, \sigma)_e}^{old} \left(\frac{r_E - r_P}{|r_E - r_P|} - n_e \right) \tag{9}$$

The underlined term is calculated using prevailing values of the variables. The explicitly calculated gradient at the CV face (denoted by the overbar) is obtained by linear interpolation of the CV-center gradients.

Correspondingly to the finite volume discretisation these approximations are multiplied by the overlapping CV area of the adjacent CVs, so that the conservativity is fulfilled.

4 VERIFICATION
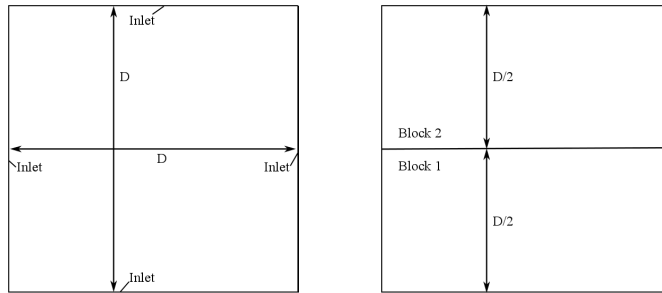
Figure 3: Manufactured Solution left: geometry and boundary conditions, right: applied blocking.

In this section it is demonstrated, that (i) the LBR method is correctly integrated in the coupled approach, (ii) the implemented LBR approach is stable for a higher refinement rate than two, (iii) it is possible to improve the solution accuracy with the implemented

LBR method and (iv) the obtained solution accuracy is identical to a segregated LBR approach presented in [6]. These mentioned points can be examined in a simple way using the Method of Manufactured Solutions (MMS) in a code verification process. The MMS is used to obtain exact solutions for the governing equations to determine whether the calculated solution is converging and the discretisation error is reduced at the expected rate (observed order equals formal order of accuracy). The form of the Manufactured Solution was chosen to be infinitely differentiable:
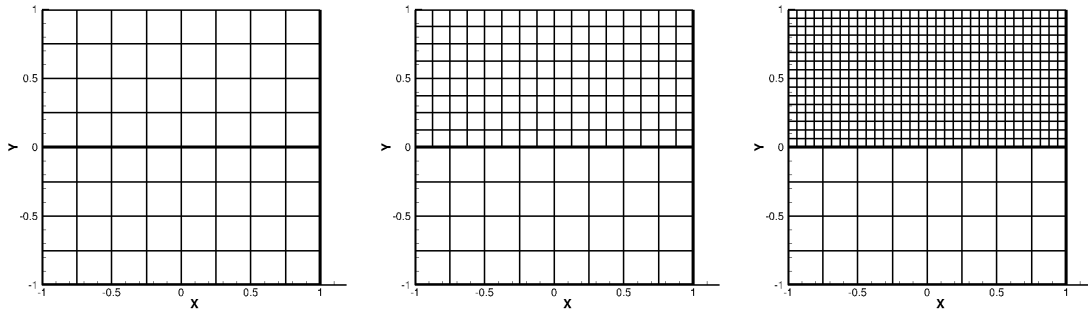
Figure 4: Examples of used grids in verification process; left: 2 blocks grid spacing 1/4, middle: 2 blocks grid spacing 1/4 & 1/8, right: 2 blocks grid spacing 1/4 & 1/16

$$
\begin{aligned}
u(x,y) &= -\cos(2\pi x)\sin(2\pi y) \\
v(x,y) &= \sin(2\pi x)\cos(2\pi y) \\
p(x,y) &= -\frac{1}{4}\left(\cos(4\pi x) + \cos(4\pi y)\right)
\end{aligned}
\tag{10}
$$

with x, y $\in [-1, 1]^2$. Additional source terms after applying the Manufactured Solutions to the governing equations with density ρ and viscosity μ set to one are:

$$
\begin{aligned}
su(x,y) &= -8\pi^2 \cos(2\pi x)\sin(2\pi y) \\
sv(x,y) &= 8\pi^2 \sin(2\pi x)\cos(2\pi y)
\end{aligned}
\tag{11}
$$

where $su(x,y)$ is the source term for the u-momentum equation and $sv(x,y)$ is the source term for the v-momentum equation. The flow geometry, boundary conditions and the applied blocking are shown in Figure 3. Examples of applied grids are shown in Figure 4. Results obtained with different sequences of meshes are given in Tables 1 and 2. Due to the symmetry of the velocity components and therefore identical results concerning error and observed order of accuracy in the following the v-velocity component is omitted. The observed order of accurary is calculated by the following equation:

$$
\text{observed order} = \frac{\log\left(\varepsilon_{N/2}/\varepsilon_N\right)}{\log(2)}
\tag{12}
$$

241

Table 1: Solution error and observed order (OO) of accuracy of u-velocity.

grid spacing	ε_N coupled	OO coupled	ε_N segregated	OO segregated
1/16	0.7643E-02	-	0.7646E-02	-
1/16 & 1/32	0.3840E-02	-	0.3842E-02	-
1/16 & 1/64	0.2001E-02	-	0.2015E-02	-
1/32	0.1906E-02	2.01	0.1906E-02	2.00
1/32 & 1/64	0.9621E-03	2.00	0.9596E-03	2.00
1/32 & 1/128	0.5043E-03	1.99	0.5069E-03	1.99
1/64	0.4761E-03	2.00	0.4761E-03	2.00
1/64 & 1/128	0.2395E-03	2.00	0.2399E-03	2.00
1/64 & 1/256	0.1265E-03	2.00	0.1272E-03	1.99

where N is the number of grid points and ε is the error defined by

$$\varepsilon_N = \sqrt{\frac{\sum_{i=1,NT}\left(\sigma_i - \sigma_{i,exact}\right)^2}{NT}}. \tag{13}$$

NT is the total number of grid cells and σ stands for the velocity-components and pressure. The results confirm, that (i) the approach is correctly implemented, (ii) the observed order

Table 2: Solution error and observed order (OO) of accuracy of pressure.

grid spacing	ε_N coupled	OO coupled	ε_N segregated	OO segregated
1/16	0.6261E-01	-	0.8658E-01	-
1/16 & 1/32	0.5287E-01	-	0.6019E-01	-
1/16 & 1/64	0.6575E-01	-	0.6789E-01	-
1/32	0.2015E-01	1.63	0.2062E-01	2.06
1/32 & 1/64	0.1421E-01	1.89	0.1541E-01	1.96
1/32 & 1/128	0.1398E-01	2.23	0.1470E-01	2.20
1/64	0.5287E-02	1.93	0.5317E-02	1.95
1/64 & 1/128	0.3687E-02	1.94	0.3808E-02	2.01
1/64 & 1/256	0.3344E-02	2.06	0.3844E-02	1.93

and formal order of accuracy agree independent of the refinement rate, (iii) it is possible to improve the numerical accuracy by local block-refinement and (iv) the solution error of the coupled and segregated approach is almost identical independent of the refinement rate.

5   APPLICATION

Separated flows behind a backward facing step have become an important test for CFD code developers. We use such a configuration to investigate possible performance gains through the application of LBR.

The geometry and applied blocking of the two-dimensional backward facing step flow is shown in Figure 5, where all length scales are given relative to the inflow height $D = 1\ m$. The kinematic viscosity $\nu = 10^{-3}\ m^2 s^{-1}$ and density of $\rho = 1.0\ kg\ m^{-3}$ is prescribed. At the inlet boundary a parabolic velocity profile is defined resulting in a steady flow at Re=200. Examples of applied grids are shown in Figure 6.

The physical quantity for comparison is the reattachment length X_{RL} of the flow behind
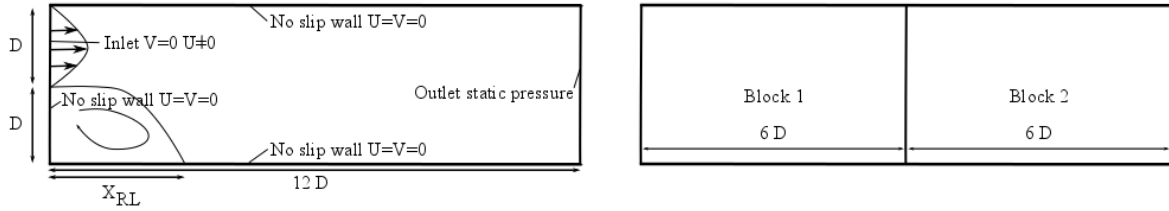
Figure 5: Backward facing step flow configuration left: geometry and boundary conditions, right: applied blocking.
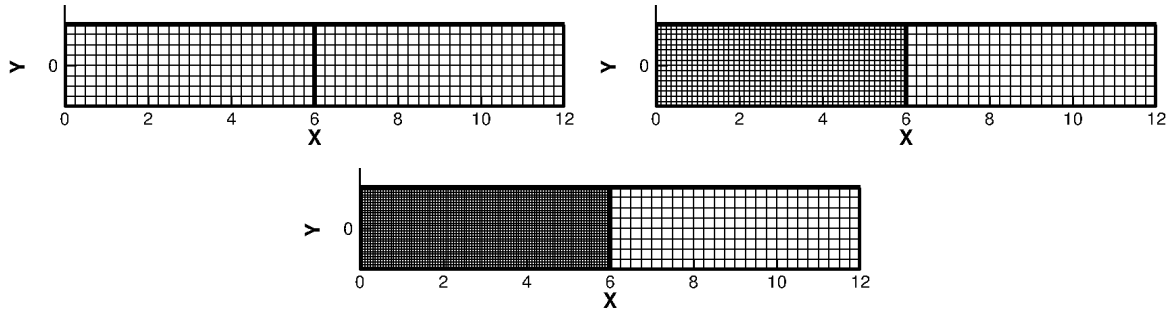
Figure 6: Examples of used grids backward facing step flow configuration left top: 2 blocks grid spacing 1/4, right top: 2 blocks grid spacing 1/8 & 1/4, bottom: 2 blocks grid spacing 1/16 & 1/4

the backward facing step. The results are summarized in Table 3 for various grids, where the coupled solver is called CP and the segregated solver is called SG. LBR indicates the application of local block refinement. Besides the reattachment length, the memory requirement, the total computing time and the computing time per CV are listed in Table 3 to allow a realistic comparison of the performance of both algorithms and the benefits through LBR. All computations were carried out on a workstation with Intel Core i7-960 CPU, 3.2 GHz and 5979 Mbyte memory. Since no exact solution for the stated flow problem is available, no quantitative statement about the error in the reattachment length can

Table 3: Results for backward facing step flow.

Code	Grid spacing	X_{RL} [m]	Memory [MB]	CPU time [sec]	CPU time/CV [sec]
CP	1/8	4.8597	16.71	1.20	0.0007839
	1/16	5.1599	48.06	4.50	0.0007325
	1/32	5.2585	189.76	22.78	0.0009273
	1/64	5.3003	822.35	130.67	0.0013293
CP LBR	1/16 & 1/8	5.1589	32.33	3.29	0.0008574
	1/32 & 1/16	5.2584	119.15	14.38	0.0009360
	1/64 & 1/32	5.2983	509.41	75.97	0.0012365
CP LBR	1/16 & 1/4	5.1515	29.44	4.42	0.0013543
	1/32 & 1/8	5.2572	105.82	15.27	0.0011701
	1/64 & 1/16	5.2989	448.52	74.30	0.0014228
SG	1/8	4.8613	3.14	2.15	0.0013985
	1/16	5.1612	4.21	23.54	0.0038309
	1/32	5.2601	8.56	308.12	0.0125375
	1/64	5.3008	25.41	4155.18	0.0422687
SG LBR	1/16 & 1/8	5.1593	3.70	12.49	0.0032512
	1/32 & 1/16	5.2595	6.82	151.28	0.0098488
	1/64 & 1/32	5.3008	18.92	2149.76	0.0349901
SG LBR	1/16 & 1/4	5.1521	3.56	9.51	0.0029156
	1/32 & 1/8	5.2572	5.77	123.47	0.0094570
	1/64 & 1/16	5.3002	14.81	1711.89	0.0327798

be given. However, the results show a good convergence behaviour. To access the benefit through LBR block one (see Figure 6) is always refined. Ideally, the reattachment length should be the same as if the fine grid spacing is applied for the entire domain.

The results in Table 3 demonstrate that the reattachment length X_{RL} is nearly independent of the grid refinement rate at the block interface. That means the reattachment length X_{RL} with the application of LBR is nearly the same as if the fine grid spacing is applied for the entire domain. This represents a saving of at most 46 per cent in the total number of CVs for a refinement rate of four. Therefore, the memory requirement, which can be a limiting factor for the application of the coupled approach, can be almost halved. Due to the smaller number of CVs and hence smaller computational effort needed per iteration, the computational time is reduced to 57 per cent of the computational time for a conventional grid. Therefore, the performance of the already highly efficient coupled approach could be increased significantly through the integration of the LBR approach. According to the results in Table 3 for the coupled approach a refinement rate greater two

offers no great benefit with respect to the computation time. It seems that an excessive non-orthogonality at the block interface for refinement rates greater two reduces the linear solver convergence to an extend that it cancels the advantage of the saving of CVs.

For completeness the results for the segregated solver are presented in Table 3 also. Generally speaking, the results are consistent with those of the coupled approach. The memory requirement and computing time can be reduced drastically through LBR in contrast to the coupled approach even for higher refinement rates.

The superiority of the coupled approach compared to the segregated approach could also be demonstrated for the integrated LBR approach. The results show that the CPU time per control volume are at least one magnitude shorter compared to the segregated approach.

6   CONCLUSIONS

The coupled approach introduced by Darwish [1] was successfully extended by a local grid refinement procedure. By means of Manufactured Solutions it was exemplified that the method proved to preserve second order accuracy of the underlying numerical scheme. Moreover, the local grid refinement was found to be robust even for higher refinement rates and provided substantial gains in performance and reduction of memory requirements. In the considered backward facing step case, computing time gains of 43 per cent and reduction of memory requirements up to 45 per cent could be verified.

Although the implementation and results presented correspond to the two-dimensional version of the code, the extension of the procedure to three dimensions is straight forward. For the three-dimensional local block refinement procedure even larger gains in performance and reduction of memory requirements are expected.

REFERENCES

[1] Darwish M.; Sraj, I. and Moukalled, F., A coupled incompressible flow solver on structured grids. *Numerical Heat Transfer, Part B* (2007) **52**:352-371.

[2] M. Schäfer, *Computational Engineering - Introduction to numerical methods*, Springer, Berlin, 2006.

[3] Lange C. F.; Schäfer M. and Durst F., Local refinement with a multigrid flow solver. *International Journal for Numerical Methods in Fluids* (2002) **38**:21-41.

[4] Berger M.J. and Collela P., Local adaptive mesh refinement for shock hydrodynamics. *Journal of Computational Physics* (1989) **82**:64-84.

[5] Quirk J.J., A cartesian grid approach with hierarchical refinement for compressible flows. *Proceedings of the Second European Computational Fluid Dynamic Conference, Stuttgart* (1994) Wiley: Chichester:200-209.

[6] Lilek Z.; Muzaferija S. and Peric M., An implicit finite-volume method using non-matching blocks of structured grid. *Numerical Heat Transfer, Part B: Fundamentals* (1997) **32,4**:385-401.

[7] Coelho P.; Pereira J.C.F. and Carvalho M.G., Calculation of laminar recirculating flows using local non-staggered grid refinement system. *International Journal for Numerical Methods in Fluids* (1991) **12**:535-557.

[8] Patankar S.V. and Spalding D.B., A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows. *International Journal Heat and Mass Transfer* (1972) **15**:1787-1806.

[9] Moukalled F. and Darwish M., A unified formulation of the segregated class of algorithms for fluid flow at all speeds. *Numerical Heat Transfer B* (2000) **37**:103-139.

[10] Vanka S.P., Fully coupled calculation of fluid flows with limited use of computer storage. *Argonne National Laboratory Technical Report* (1983) **ANL**:83-87.

[11] Lonsdale R.D., An algebraic multigrid scheme for solving the Navier-Stokes equations on unstructured meshes. *Proc. 7th International Conference on Numerical Methods in Turbulent abd Laminar Flows* (1991) Stanford,CA 1432-1442.

[12] Webster R., An algebraic multigrid solver for Navier-Stokes problems. *International Journal Numerical Methods Fluids* (1994) **18**:761-780.

[13] Rhie C.M. and Chow W.L., A numerical study of the turbulent flow past an isolated airfoil with trailing edge separation *AIAA* (1983) **21**:1525-1532.

[14] Balay S.; Brown J.; Buschelman K.; Gropp W.D.; Kaushik D.; Knepley M.G.; McInnes L.C.; Smith B.F. and Zhang H., PETSc Web page, (2012).

[15] Ferziger J.H. and Peric M., Computational Methods for Fluid Dynamics, Springer, Berlin, 1996.

NON INTRUSIVE STOCHASTIC SIMULATIONS USING A GOAL ORIENTED ADAPTIVE STRATEGY.

Eric FLORENTIN*, Pedro DIEZ†

*LMT-Cachan (ENS Cachan / CNRS / UPMC / PRES UniverSud Paris)
61, avenue du Président Wilson, F-94235 Cachan, France
e-mail: eric.florentin@ens-cachan.fr

†Laboratori de Càlcul Numric (LaCàN),
Universitat Politècnica de Catalunya, C2 Campus Nord UPC, E-08034 Barcelona, Spain
e-mail: pedro.diez@upc.edu

Key words: Reduced basis, Adaptivity, Stochastic modeling, Goal-oriented error assessment

Abstract. The paper presents a goal-oriented strategy in the framework of stochastic non-intrusive Monte Carlo finite element simulations. The methods consists in a successive enrichment of a reduced basis. This enrichment is performed on the fly, during the Monte Carlo process. The error made by the representation on the reduced basis is assessed introducing a dual problem associated to the quantity of interest. The efficiency of the proposed approach is illustrated in numerical examples. In particular, an extension of the work developed in [6] is introduced. It consists in introducing a reduced basis for solving the dual problem in an efficient way. Different variant are tested for the successive enrichment of the dual reduced basis.

1   INTRODUCTION

Stochastic Finite Element Methods (FEM) are currently an essential tool for the quantitative prediction of the response of mechanical models that include randomness. Both for the research and industrial players, a key issue is to reduce the computational cost in order to afford dealing with large scale applications. A state of the art for stochastic methods can be found for example in [1, 2, 3].

In [4, 5], the authors introduced a reduced basis methodology to reduce the cost of Monte Carlo simulations, offering an attractive framework for solving stochastic problems with a large number of parameters. The idea is simple and effective because the different Monte Carlo shots lead to similar FE problems and therefore the reduced basis approach is highly performant.

This paper describes a new approach to generate a reduced basis in the context of Monte Carlo strategies for stochastic modeling. The reduced basis is constructed automatically, ensuring a prescribed level of accuracy for the output of interest. The proposed methodology uses standard elements in goal-oriented error assessment and adaptivity [6],. Here, the error which is assessed is the approximation introduced by the Reduced Basis with respect to the complete FE solution, for a given mesh. That differs from the usual practice in the Verification framework, in which the error introduced by the mesh is evaluated. Numerical tests demonstrate the efficiency and robustness of the proposed strategy. The use of this method for 3D massive industrial examples, where the cost is of primary importance, is going to be the object of further research.

2   PROBLEM STATEMENT

Let Ω be a bounded domain and $\partial\Omega$ its boundary which is divided in two parts $\partial_D\Omega$ and $\partial_N\Omega$ such that $\overline{\partial_D\Omega} \cup \overline{\partial_N\Omega} = \partial\Omega$, $\partial_D\Omega \neq \varnothing$ and $\partial_D\Omega \cap \partial_N\Omega = \varnothing$. Displacement $\mathbf{u}_d$ is imposed on $\partial_D\Omega$ and a traction $\mathbf{g}_d$ is applied on $\partial_N\Omega$ and a body force field $\mathbf{f}_d$ is applied in Ω.

The material is linear elastic, and $\mathbf{K}(\mathbf{x}, \theta)$ is the Hooke tensor random field, where $\mathbf{x} \in \Omega$ is the position and $\theta \in \Theta$ denotes the randomness. Θ is the set of possible outcomes of θ.

The problem reads: find the unknown displacement field $\mathbf{u}(\mathbf{x}, \theta)$ such that

$$\mathbf{div}\left(\mathbf{K}(\mathbf{x}, \theta)\varepsilon\left[\mathbf{u}(\mathbf{x}, \theta)\right]\right) + \mathbf{f}_d(\mathbf{x}, \theta) = \mathbf{0} \qquad \text{in } \Omega \tag{1a}$$

$$\mathbf{K}(\mathbf{x}, \theta)\varepsilon\left[\mathbf{u}(\mathbf{x}, \theta)\right].\mathbf{n} = \mathbf{g}_d(\mathbf{x}, \theta) \qquad \text{on } \partial_N\Omega \tag{1b}$$

$$\mathbf{u}(\mathbf{x}, \theta) = \mathbf{u}_d(\mathbf{x}, \theta) \qquad \text{on } \partial_D\Omega \tag{1c}$$

The corresponding standard weak form reads as follows: find $\mathbf{u}(\mathbf{x}, \theta)$ such that

$$a(\mathbf{u}(\mathbf{x}, \theta), \mathbf{w}(\mathbf{x})) = \ell(\mathbf{w}(\mathbf{x})) \quad \forall \mathbf{w}(\mathbf{x}) \in \mathcal{U} \tag{2}$$

where $a(\cdot, \cdot)$ is a bilinear form , $\ell(\cdot)$ is a linear form and $\mathcal{U}$ the set of admissible displacements, satisfying (1c).

3   MONTE CARLO NON INTRUSIVE SOLVING SCHEME

The non-intrusive approach decouples the discretization of the physical space and the stochastic space, represented here by Ω and Θ. This can be described in two steps.

$\square$ **Step 1:** finite element discretization.

In this first phase, the problem is considered as deterministic (*for a given value of θ.*) The discretisation of the space is characterized by the standard finite element functions $N_i(\mathbf{x})$, $i = 1, 2 \ldots N_{FE}$ $\mathcal{U}_h \subset \mathcal{U}$

$$\mathcal{U}_h = \text{span}\left\{N_1, N_2, \ldots N_{N_{FE}}\right\} \tag{3}$$

The numerical approximation in the space defined in (3) is $\mathbf{u}_h$ such that

$$\mathbf{u}(\mathbf{x}, \theta) \approx \mathbf{u}_h(\mathbf{x}, \theta) = \sum_{i=1}^{N_{FE}} u_i(\theta) N_i(\mathbf{x}) = \mathbf{N}^T(\mathbf{x}) \mathbf{U}(\theta), \tag{4}$$

with $\mathbf{U} = [u_1 \, u_2 \, \cdots \, u_{N_{FE}}]^T$ and $\mathbf{N}(\mathbf{x}) = [N_1(\mathbf{x}) \, N_2(\mathbf{x}) \, \cdots \, N_{N_{FE}}(\mathbf{x})]^T$. The corresponding discretized form of (2) is the linear system of equations

$$\mathbb{K}(\theta) \mathbf{U}(\theta) = \mathbf{F}(\theta), \tag{5}$$

where $\mathbb{K}$ is the classical finite element stiffness matrix and classical $\mathbf{F}$ nodal forces.

☐ **Step 2:** Monte Carlo simulation.

The Monte Carlo technique consists in generating a number N_{MC} of realizations of θ. Note that these realizations are generated using the actual Probability Density Function (PDF) of θ. This is equivalent to determine N_{MC} realizations of $\mathbb{K}(\theta)$ and therefore, solving N_{MC} linear systems of equations (5), obtaining N_{MC} realizations of $\mathbf{U}(\theta)$.

Thus, the PDF of $\mathbf{U}(\theta)$ or some specific Quantity of Interest (linearly dependent with $\mathbf{U}$) is approximated from these realizations.

This non-intrusive strategy is extremely simple because it decouples the approximation of the stochastic behavior and the solution of the deterministic mechanical model. The main drawback is that the numerical cost is can be very large.

4 METHOD DEVELOPPED

4.1 Reduced Basis

The reduced basis method allows to reduce computational costs when solving a large number of problems, but introduce an error linked with the size of the reduced basis size. Monte Carlo sampling requires solving many instances of problem (5) and Reduced Basis Strategy can help. A collection of N_{RB} linearly independent solutions creates a approximated subset of the FE space : $\{u_{h(1)}, u_{h(2)}, \ldots, u_{h(N_{RB})}\}$ described by the corresponding vectors of nodal values $\{\mathbf{U}_1, \mathbf{U}_2, \ldots, \mathbf{U}_{N_{RB}}\}$,

$$\mathcal{U}_{RB} := \operatorname{span}\left\{ u_{h(1)}, u_{h(2)}, \ldots, u_{h(N_{RB})} \right\} \subset \mathcal{U}_h \tag{6}$$

Then, the solution of a new instance of (5) is seek in $\mathcal{U}_{RB}$ instead of in $\mathcal{U}_h$ as a linear combination of the elements of the reduced basis:

$$\mathbf{U}_{RB} = \sum_{i=1}^{N_{RB}} a_i \mathbf{U}_i = \mathbb{U}_{RB} \, \mathbf{a} \tag{7}$$

where the matrix $\mathbb{U}_{RB} = [\mathbf{U}_1 \, \mathbf{U}_2 \cdots \mathbf{U}_{N_{RB}}]$ (with N_{FE} rows and N_{RB} columns) describes the change of basis and $\mathbf{a}^T = [a_1 \, a_2 \cdots a_{N_{RB}}]$ is the vector of unknowns.

The solution $\mathbf{a}$ can be found as the solution of a linear system :

$$\left(\mathbb{U}_{RB}^T \mathbb{K}(\theta) \mathbb{U}_{RB} \right) \mathbf{a} = \left(\mathbb{U}_{RB}^T \mathbf{F}(\theta) \right) \text{ readily rewritten as } \mathbb{K}_{RB}(\theta) \mathbf{a}(\theta) = \mathbf{F}_{RB}(\theta) \tag{8}$$

4.2 Goal oriented error

At every instance θ_k of the Monte Carlo process, the error introduced in the reduced basis phase is measured by

$$\mathbf{E}_{RB} := \mathbf{U}_{RB} - \mathbf{U}, \tag{9}$$

This error vector, expressed in the standard FE basis, corresponds to the approximation introduced by the reduced basis with respect to the complete FE solution. It does not represent the error due to the Finite Element discretization, but the error uniquely do to the reduced basis process.

The squared norm of the residual, $\mathbf{R}_{RB}^T \mathbf{R}_{RB}$ is in fact a good error indicator for the energy norm of the error with

$$\mathbf{R}_{RB} := \mathbb{K}\mathbf{U}_{RB} - \mathbf{F} \tag{10}$$

If interested by a QoI denoted Q_{RB} associated with the reduced basis solution $\mathbf{U}_{RB}$,

$$Q_{RB} = \mathbf{G}^T \mathbf{U}_{RB} \tag{11}$$

it is classical to introduce a dual problem and its solution $\mathbf{V}$ and where $\mathbf{G}$ defines the quantity of interest. Using the same spatial discretisation for dual problem as the direct problem (5), it results a similar system for $\mathbf{V}$

$$\mathbb{K}(\theta)\mathbf{V}(\theta) = \mathbf{G}(\theta), \tag{12}$$

This dual solution $\mathbf{V}$ can be used to assess the error in the evaluation of the QoI associated with the reduced basis. The error in the QoI associated with the reduced basis approach writes:

$$e_{RB}^Q := \mathbf{V}^T \mathbf{R}_{RB}. \tag{13}$$

5 Adaptive Strategies

5.1 Algorithms

Note that if $\mathbf{V}$ is known, the error in the QoI associated with the reduced basis is computed explicitly using the right-hand side term of (13), once the reduced basis solution is available. The problem is that solving (12) leads to a similar computational cost that for solving 5. The local error 13 can be used to control the enrichment of an adapted reduced basis. Indeed, this allows assessing the error committed in every shot with a low computational cost and decide on the fly if the reduced basis is rich enough or if it has to be enriched further.

Two adaptation strategies, corresponding to different level of randomness are proposed.
□ **Algo 1:** Dual unique resolution

From the practical viewpoint it is assumed that the variation of $\mathbf{V}$ with the randomness is small in such a way that $\mathbf{V}$ is kept constant in order to estimate the error in the QoI.

Thus, the solution of the dual system (12) is performed just once and leads to $\mathbf{V_0}$. The criterion used to estimate error is then:

$$e_{RB1}^Q = \mathbf{V_0}^T \mathbf{R}_{RB} \tag{14}$$

This algorithm is presented in [6].

□ **Algo 2:** Dual Reduced Basis Resolution

For problem with higher level of randomness, the solution $\mathbf{V_0}$ can be very different of the actual value $\mathbf{V}$. Then, the idea correspond to a resolution of the dual problem using a reduced basis representation.

$$\mathbf{V}_{RB} = \sum_{i=1}^{N_{RBD}} b_i \mathbf{V}_i = \mathbb{V}_{RB}\, \mathbf{b} \tag{15}$$

where the matrix $\mathbb{V}_{RB} = [\mathbf{V}_1\, \mathbf{V}_2 \cdots \mathbf{V}_{N_{RB}}]$ (with N_{FE} rows and N_{RBD} columns) describes the change of basis and $\mathbf{b}^T = [b_1\, b_2 \cdots B_{N_{RBD}}]$ is the vector of unknowns.

Then the solution $\mathbf{b}$ can be found as the solution of a linear system :

$$\left(\mathbb{V}_{RB}^T \mathbb{K}(\theta)\mathbb{V}_{RB}\right) \mathbf{b} = \left(\mathbb{V}_{RB}^T \mathbf{G}(\theta)\right) \ \text{readily rewritten as} \ \mathbb{K}_{RB}(\theta)\mathbf{b}(\theta) = \mathbf{G}_{RB}(\theta) \tag{16}$$

The criterion used to estimate error is then:

$$e_{RB2}^Q = (\mathbb{V}_{RB}\, \mathbf{b})^T \mathbf{R}_{RB} \tag{17}$$

Remark 1: The choice of a unique vector in the basis $\mathbb{V}_{RB} = V_0$ correspond exactly to algorithm 1. In this sense, this resolution correspond to a generalization of the algorithm 1.

Remark 2: A particular case correspond to the same reduced basis to obtain the approximated solution of both primal an dual problem.

$$\mathbb{U}_{RB} = \mathbb{V}_{RB} \tag{18}$$

Remark 3: Different variants that correspond to the decision to enrich independently $\mathbb{U}_{RB}$ and $\mathbb{V}_{RB}$ or not can be performed. The decision to enrich $\mathbb{V}_{RB}$ can be based on a dual error criterion :

$$e_{RB2}^{QD} = (\mathbb{U}_{RB}\, \mathbf{a})^T (\mathbb{K}\mathbb{V}_{RB} - \mathbf{G}) \tag{19}$$

5.2 Results

Results of algorithm 1 and algorithm 2 are compared to a full Monte Carlo simulation which is considered as the reference. In [6] the results illustrates the capabilities of algorithm 1 for a 2D mechanical problem. This work focuses on the different possibilities to solve the problem using algorithm 2 which is an extension of the work presented in [6] and the results illustrate the algorithm 2 developed here. Results are given in Table 1,

Table 1: Results for algorithm 2

ϵ_0	N_{RB}	N_{RB}	ϵ
10^2	1	1	1.3E+1
10	8	6	2.4E+0
1	18	16	2.0E-1
10^{-1}	38	38	1.9E-2
10^{-2}	93	79	2.5E-3
10^{-3}	139	122	2.2E-4

for a given level of prescribed local error ϵ_0 on the QoI, the actual error ϵ is given and is to be compared to the prescribed one ϵ_0. The size of the Reduced basis N_{RB} and N_{RBD} is also given. The results are showing that the algorithm makes it possible to respect a given level of error on the quantity of interest. The employed reduced basis for solving primal and dual problems are automatically adapted by enrichment on the fly.

6   CONCLUSIONS

Next step is to implement this algorithm and its variant on 3D large scale examples where the problem of computational costs are of primary importance.

REFERENCES

[1] G. I. Schuller. A state-of-the-art report on computational stochastic mechanics. A State-of-the-Art Report on Computational Stochastic Mechanics. *Probabilistic Engineering Mechanics*, 12(4):197 – 321, (1997).

[2] Anthony Nouy. Recent developments in spectral stochastic methods forthenumerical solution of stochastic partial differential equations. *Archives of Computational Methods in Engineering*, 16:251–285, (2009).

[3] George Stefanou. The stochastic finite element method: Past, present and future. *Computer Methods in Applied Mechanics and Engineering*, 198(9-12):1031–1051, (2009).

[4] S. Boyaval, C. Le Bris, T. Lelièvre, Y. Maday, N. C. Nguyen, and A. T. Patera. Reduced basis techniques for stochastic problems. *Archives of Computational Methods in Engineering*, 17(4):435–454, (2010).

[5] S. Boyaval, C. Le Bris, Y. Maday, NC. Nguyen, and Anthony T. Patera. A reduced basis approach for variational problems with stochastic parameters: Application to heat conduction with variable robin coefficient. *Computer Methods in Applied Mechanics and Engineering*, 198(41-44):3187 – 3206, (2009).

[6] E. Florentin and P. Díez. Adaptive reduced basis strategy based on goal oriented error assessment for stochastic problems *Computer Methods in Applied Mechanics and Engineering*, Vol. 225-228. 116–127 (2012)

MULTI-BLOCK DECOMPOSITION USING CROSS-FIELDS

HAROLD J. FOGG, CECIL G. ARMSTRONG AND TREVOR T. ROBINSON

School of Mechanical and Aerospace Engineering,
Queen's University, Belfast BT9 5AH, N. Ireland.
e-mail: hfogg01@qub.ac.uk

Key words: Quadrilateral Mesh, Cross-Field, Mesh Singularities, Multi-block decomposition

Abstract. A method is presented for automatically generating cross-fields from direction constraints on arbitrary surfaces meshed with triangular elements. A cross-field describes the directionality of a quad mesh and the developed solver produces a cross-field abiding to prescribed target element sizes and directions. Additionally, a simple method of generating a multi-block decomposition suitable for all-quad meshing by tracing critical streamlines of the cross-field is explained and illustrated.

1 INTRODUCTION

1.1 Related work

Computer graphics visual effects methods such as anisotropic shading, morphing, texture synthesis, and global parametrisation of surfaces have been the driving force for the development of many vector field design methods [1, 2, 3, 4]. The vector field purpose is to control a bijective mapping onto the surface and the design objective is to minimise the angular distortion and/or stretching of the mapping. Vector fields with four-way symmetry have been found to be particularly useful for these applications. The concept has been described by many authors with different terminology, such as *cross-field* [5, 6], *4-way symmetry field (4-RoSy field)* [7, 8], and *frame field* [9]. Quad mesh generation is a somewhat equivalent problem where the mapping of an isotropic square mesh onto the surface is sought. In recent years cross-fields have begun to emerge as an important common feature to many new advanced mesh generation methods.

The QuadCover algorithm [9] produces a globally continuous parametrisation whose isoparametric curves describe a global closed quadrilateral mesh from a provided cross field. It was strongly inspired by Ray *et al.* [10] whose Periodic Global Parametrisation method produces comparable results by similar means. The global parametrisation method for generating quad meshes from input cross-fields has been used in other

works [5, 11] with modifications. Progress has also been made on the 3-D equivalent problem [12, 13, 14].

Various approaches have been taken for the construction of smooth cross-fields on surfaces. The curvature orientated cross-field is a popular option and used in [15, 16, 9]. It produces appealing cross-fields coupled to inherent surface properties and a piece-wise linear approximation aligned with the principal directions is an optimal representation of the smooth underlying surface [17, 18] (as cited in [16]). However, where curvature tensors are symmetrical the principal directions are ill-defined which results in a discontinuous cross-field. Smoothing or optimisation schemes are needed to put order on the randomly orientated crosses in these regions.

For other approaches, a typical first step is to define a measure of the smoothness of a cross-field based on the angular deviations, variably called the smoothness energy, error or energy functional, and then set about minimising its integral over the surface area. Wei [2] and Turk [3] took a pragmatic approach and developed mixtures of relaxation and interpolation procedures to numerically converge to a vector field result.

Bunin's *continuum theory of unstructured mesh generation* [6] clarifies how singularities are the crucial characteristic features of cross-fields and quad meshes. A rigorous theory is presented that relates the size variation of an infinitesimal quad mesh to the directional variation of a cross-field. The scalar size field, acting as the continuum description of the mesh, is governed by a stationary heat equation where singularities appear as point sources and sinks. A quad mesh singularity corresponds to a node which is not attached to a regular array of quad elements, *e.g.* where three or five quad elements are attached to a given node in the interior of the mesh. Singularities necessarily occur from particular combinations of mesh alignment constraints or significant total Gaussian curvature of the surfaces. Once a valid arrangement of singularities has been identified, the scalar field can be easily solved numerically. In a follow-up paper [19], Bunin describes a method to tackle the *inverse Poisson* type problem of placing mesh singularities on planar surfaces. Ben-Chen *et al.* [20] proposed a method for identifying suitable locations for cone singularities of a conformally related metric to the surface by considering the Gaussian curvatures.

Palacios *et al.* [8], describe a design system in which cross-fields can be created and modified on surfaces from a set of prescribed singularities using interpolation and relaxation algorithms. A vector-based representation of an N-RoSy field is used which is globally continuous over planar surfaces but not for curved surfaces. A *parallel transport* scheme similar to that described by Zhang [4] is proposed to describe the field continuously over local regions.

The same vector-based representation of a cross-field is used in the recent work by Kowalski *et al.* [21] for decomposing planar surfaces for quad meshing. Boundary alignment constraints are applied to the cross-field and it is solved in the interior by a two stage algorithm: The first solves a stationary heat equation for the coordinates of the representation vector. The second normalises the vector solution of the previous step by an optimisation routine. The final cross-field is smooth with a small number of singularities

appearing where the neighbourhood boundary alignment constraints are incompatible.

A method for constructing a smooth cross-field from a sparse set of directional constraints is described by Bommes *et al.* [5]. Direction constraints are extracted from the triangulated surface representation along curves where the estimated principal curvatures are significantly anisotropic. An involved optimisation algorithm called a *greedy mixed-integer* solver is designed for minimising the smoothness energy of a cross-field that is locally constant on each tri element. It iteratively solves for a smooth cross-field containing well-positioned singularities between the constrained directions.

With a similar discretised cross-field representation and smoothness energy functional, Liu *et al.* [11] use a non-linear optimisation method to solve for a cross-field from direction constraints along user-specified strokes. A properly initialised cross-field is imperative for the procedure. A simple method is used to propagate the cross-field from the crosses at specified elements to unspecified ones by iterating over an unordered queue of elements. Crosses for elements without specified crosses that neighbour elements with specified crosses are computed by a method mimicking a parallel-transport operation and then they are removed from the queue. This is repeated until the queue is empty. Liu comments that by using a randomly generated initialisation instead of the propagation based initialisation, an optimised cross-field contains many spurious features and singularities are effectively determined by the specifics of the arbitrary initialisation.

The problem of generating smooth *3-D cross-frame fields* for hexahedral mesh generation has been addressed in recent works. As for 2-D, the proposed 3-D methods involve using optimisation schemes to minimise the smoothness energy of a cross-frame field. Huang *et al.* [22, 13] proposed the use of a spherical harmonic functions to describe cross-frames which have convenient properties for measuring field smoothness and for specifying alignments to cross-frames. Before the non-linear optimisation solver is used to minimise the discretised smoothness energy integral, the system is initialised by solving a stationary heat type problem for the spherical harmonic coefficients. The fundamentals of the procedure are like those used by Kowalski [21] to calculate cross-fields in 2-D. Li *et al.* [14] describe an equivalent method to measure the smoothness energy of a cross-frame field. A boundary aligned cross-frame field is solved for by non-linear optimisation similarly. Their initialisation procedure is summarised as propagating the boundary frames into the interior of the tetrahedral mesh of the volume so that for any interior tet, its frame is assigned to be the same as that of its nearest boundary tet.

1.2 Contributions

The work presented in this paper makes the following contributions:

- A methodology is developed for generating smooth cross-fields on triangulated surfaces from an arbitrary number of alignment constraints using an efficient *fast-marching* algorithm to propagate the field into regions between constraints. It improves on the initialisation process described by Liu [11]. A similar fast marching

algorithm is used by Lai [23] to parallel-transport a vector over a surface.

- The method is reasoned in the terms of Bunin's *continuum theory* and a smoothness energy functional is developed which is minimised locally in each propagation step. Additional terms for penalising deviations from target element sizes and directions are included in the functional so that the generated cross-fields have improved suitability with respect to prescribed size fields.

- A simple method is described for generating multi-block decompositions of surfaces by tracing critical streamlines of the cross-fields similar to that shown in [21].

2 THEORY

2.1 Continuum theory of unstructured mesh generation

In Bunin's theory [6] a conformal (angle-preserving) transformation from an arbitrary surface of interest, S, to a locally flat or developable surface, $\tilde{S}$, is searched for. A field of equally spaced parallel geodesics and their orthogonal trajectories represent a uniform mesh on $\tilde{S}$ and its image on S describes the quad mesh solution. The elements are ideally shaped squares on $\tilde{S}$ and as the spacing reduces to zero the elements are also squares on S. The key results of Bunin's paper are:

1. The scalar variable ϕ involved in the conformal factor between the metric tensors of S and $\tilde{S}$,

$$\tilde{g}_{ij} = e^{2\phi} g_{ij} \,, \quad \phi \in \mathbb{R}, \tag{1}$$

obeys the Poisson equation,

$$\Delta_s \phi = K + \sum_{i=1}^{N} k_i \frac{\pi}{2} \delta_{P_i} \,, \quad k_i \in \mathbb{Z} \geq -4, \tag{2}$$

where Δ_s is the Laplace-Beltrami operator, K is the Gaussian curvature of S and the δ_{P_i} terms are weighted delta Dirac functions describing discrete *cone* points of $\tilde{S}$ where a discrete total curvature of multiples of $\pi/2$ occurs at the cone vertex. The cone points correspond to singularities in the quad mesh with their characteristic types given by the integers k_i. From the metric scaling relationship (Eqn. (1)), it follows that

$$\phi = -\ln h, \tag{3}$$

where h is the element size.

2. The geodesic curvature, κ_g, of a mesh edge is directly related to local variation of the ϕ-field by

$$\frac{\partial \phi}{\partial e} = \kappa_g \equiv \frac{\partial \theta}{\partial t}. \tag{4}$$

The tangent vector $\mathbf{t}$, intrinsic normal $\mathbf{e}$, and surface normal $\mathbf{n}$ make a right hand orthonormal basis. The directional derivatives in the directions of $\mathbf{t}$ and $\mathbf{e}$ are represented by $\frac{\partial}{\partial t}$ and $\frac{\partial}{\partial e}$. θ is defined as the angle of the tangent vector from a reference direction on the tangent plane that is parallel transported over the surface (see [6, Section 2]). It follows that

$$\nabla\theta = R(-\tfrac{\pi}{2}\mathbf{n})\nabla\phi, \tag{5}$$

where $R(-\frac{\pi}{2}\mathbf{n})$ represents a $-\pi/2$ rotation about $\mathbf{n}$. Hence, $\nabla\theta$ is also governed by Eqn. (2).

3. For boundary conforming solutions, boundary conditions are applied to curved boundary edges and corners with angles not multiples of $\pi/2$. Neumann boundary conditions are applied to curved boundary edges according to Eqn. (4). Point sources with strengths dependent on the corner angle are applied at boundary corners.

These concepts are illustrated by the example shown in Figure 1.
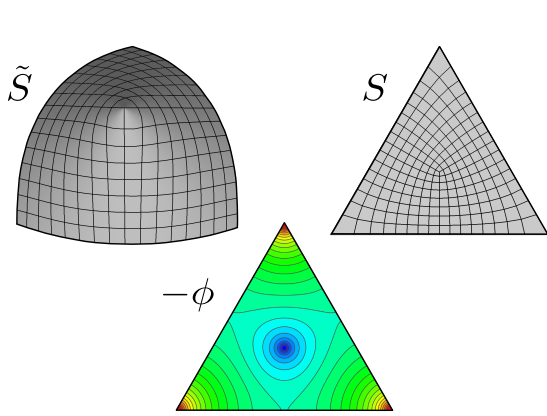
Figure 1: Example of Bunin's *continuum theory* for a flat triangular surface S. The conformally related surface $\tilde{S}$ is conical. The $-\phi$-field describing the logarithm of the element size on S is solved as a temperature field in a FEA heat conduction analysis.
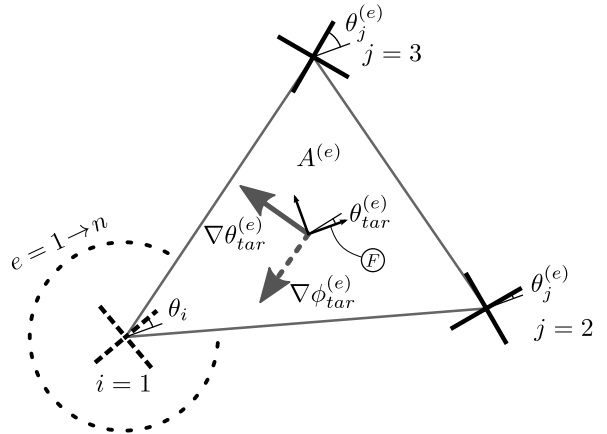
Figure 2: Optimising cross angle at node i adjacent to nodes j considering the target size gradients and orientations in the connected elements 1 to n.

2.2 Energy Functional

For the purpose of designing a cross-field solver, an energy functional is developed for minimisation. Rapid variation in cross-field directions, and hence element size variation, should be penalised. An obvious choice is to use the Dirichlet energy of the ϕ/θ-field,

$$E_{smoo} = \int \|\nabla\phi\|^2 dA = \int \|\nabla\theta\|^2 dA. \tag{6}$$

It is straightforward to prove that its value equals both the sum of the squared local geodesic curvatures of orthogonal mesh edges, and also the square of the norm of the element size gradient normalised by the local size. It is worth noting that that the Laplace equation (Eqn. (2) without source terms) is equivalent to minimising the Dirichlet energy in 2-D.

Given a target element size field for the mesh, a cross-field that accommodates those sizes will change direction according to the gradients of the prescribed size field as by Eqn. (4). Thus another penalty term to account for this is

$$E_{grad} = \int \|\nabla\theta - \nabla\theta_{tar}\|^2 dA, \tag{7}$$

where $\nabla\theta_{tar}$ is the target gradient of the local angle of the cross-field that can be computed from the target size field using Eqn. (3) and Eqn. (5). Similarly, a penalty term for the deviation of the field from prescribed directions is

$$E_{direc} = \int \frac{(\theta - \theta_{tar})^2}{A_{total}} dA. \tag{8}$$

(Note: The total area divisor is included so that the expression yields a dimensionless value)

The overall energy functional is the combination of the individual energy functionals:

$$E = E_{smoo} + w_1 E_{grad} + w_2 E_{direc}, \quad w_1, w_2 \in \mathbb{R} > 0, \tag{9}$$

where w_1 and w_2 are arbitrary weighting factors.

3 CROSS-FIELD GENERATION

3.1 Discretisation

A piece-wise linear description of a cross-field on a tri mesh is used in this work. This allows finite-element theory to be utilised to formulate the problem locally as a directly solvable linear system. On a tri element, e, an example of which can be shown in Figure 2, the angle of a cross relative to a local frame is approximated by a bilinear function, or equivalently by blends of the node angle values using linear shape functions:

$$\theta^{(e)}(x, y) = \alpha_0^{(e)} + \alpha_1^{(e)}x + \alpha_2^{(e)}y,$$
$$= \sum_{i=1}^{3} L_i^{(e)}(x, y)\theta_i^{(e)}. \tag{10}$$

A target element size field can be discretised in the same way on the tri mesh.

Considering the problem of minimising E by adjusting the cross at a node i while keeping the crosses at surrounding nodes j fixed, of elements 1 to n, the solution satisfies

the equation

$$\frac{\partial E}{\partial \theta_i} = 0$$

$$= \sum_{e=1}^{n} \frac{\partial}{\partial \theta_i} \left\| \nabla \theta^{(e)} \right\|^2 A^{(e)} + w_1 \sum_{e=1}^{n} \frac{\partial}{\partial \theta_i} \left\| \nabla \theta^{(e)} - \nabla \theta_{tar}^{(e)} \right\|^2 A^{(e)} + w_2 \sum_{e=1}^{n} \int_e \frac{(\theta^{(e)} - \theta_{tar}^{(e)})^2}{A_{total}} dA. \quad (11)$$

Using the local linear approximations of $\theta^{(e)}$ and $\theta_{tar}^{(e)}$ in Eqn (10), Eqn. (11) can be reduced and re-expressed with θ_i as a function of $L_i^{(e)}$, $L_j^{(e)}$, $A^{(e)}$, $\theta_j^{(e)}$, and $\theta_{tar}^{(e)}$. In this way a formula is constructed for calculating the optimum cross at a node.[1] Figure 2 shows the set-up for one of the n elements. In a propagation advancement step (Section 3.2) θ_i is optimised from the point of view of a single element. For a smoothing process all n elements would be taken into account.

3.1.1 Measuring angles

The local frame, F, is located at one of the nodes, call it N_f, with its x- and z-axes aligned with a cross direction and the local surface normal, $\mathbf{n}_f$. Angles θ_i and θ_j of crosses $cross_i$ and $cross_j$ are taken with respect to F, as shown in Figure 2. For an arbitrary node N_x, $cross_x$ on tangent plane E_x, is rotated onto tangent plane E_f to give $cross_x{'}$. This is effected by the rotation matrix $R(\mu\mathbf{u})$, where $\mu = \cos^{-1}(\mathbf{n}_x \cdot \mathbf{n}_f)$ and $\mathbf{u} = (\mathbf{n}_x \times \mathbf{n}_f) / |\mathbf{n}_x \times \mathbf{n}_f|$. This corresponds to a discrete parallel transport operation performed on a cross between the nodes (See [6, Section 2]). The rotation about $\mathbf{n}_f$ that moves $cross_x{'}$ onto $cross_f$ through the smallest angle (in the range $(-\pi/4, \pi/4)$) is used to describe the change in the cross-field orientation. It corresponds to the integral of the geodesic curvature of the cross-field.

3.2 Cross-field propagation

With the definition of the functional to be minimised and the discretisation of the problem, one route to generating the cross-field is to set-up an optimisation process à la Bommes [5] or Liu [11]. However computationally expensive non-linear solvers are used and an effective initialisation phase is required. Such an initialisation algorithm is proposed here. Although, the results show that the stand-alone generated cross-fields are of high quality without optimisation.

To produce boundary conforming meshes the cross-field is set as aligned with the tangent vectors of boundary edges. At a boundary corner, either the bisector direction or an offset direction of $\pi/4$ from the bisector is used. The decision is based on minimising the corresponding point source strength in Bunin's *continuum theory* and depends on the corner angle and the choices are fairly intuitive.

---

[1]The expression and its derivation are uncomplicated and are omitted here due to space limitations.
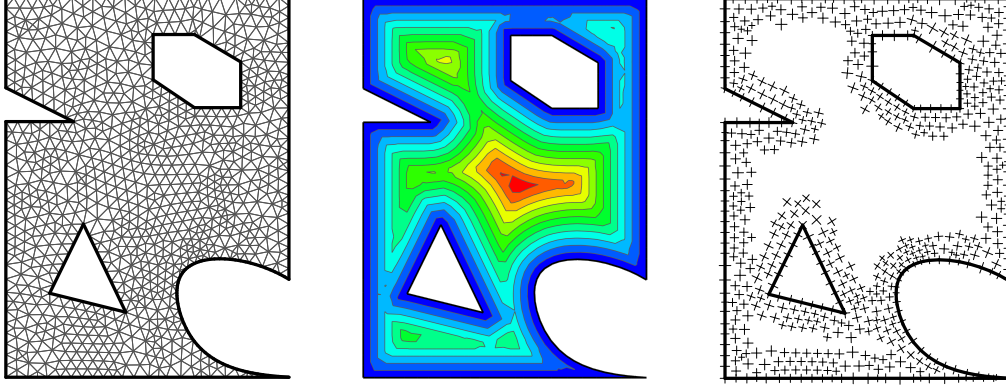
Figure 3: A 2-D test case surface with curved boundaries, sharp indents and hole features. (Left) Tri mesh on surface. (Centre) Distance field of Eikonal equation solved as part of the propagation algorithm. (Right) Propagation of cross-field from boundaries inwards.

The *fast marching alorithm*, introduced by Sethian [24], is used to propagate the cross-field from direction constraints in concordance with the distance field of the Eikonal equation. The algorithm developed in this work is similar to the finite-element based algorithm presented in [25]. A summary is given in Algorithms 1 and 2.

Algorithm 1 Cross-field propagation

▷ *Initilisation*
narrow_band_list ← {}
alive_list ← {}
for each Node, N **do**
   if N.cross ≠ None **then**
      N.d ← 0
      N.alive ← True
      alive_list.append(N)
   else
      N.alive ← False
   end if
end for
for N in alive_list **do**
   N.method*(narrow_band_list)
end for
narrow_band_list.sort()   ▷ *wrt d*
▷ *Remove first node and advance narrow band to include its neighbouring nodes*
while narrow_band_list ≠ {} **do**
   trialN ← narrow_band_list.pop(0)
   narrow_band_list.remove(0)
   trialN.alive ← True
   trialN.method*(narrow_band_list)
   narrow_band_list.sort()   ▷ *wrt d*
end while

Algorithm 2 Node method*

function METHOD*(narrow_band_list)
   ▷ *Description: For all neighbouring nodes, if not alive and not in narrow_band_list compute d and cross members and add to narrow_band_list.*
   for N1 in Node.neighbour_Ns **do**
      dN2N3_list ← {}
      for Element, E, in N1.neighbour_Es **do**
         {N2,N3} ← E.nodes.remove(N1)
         if N2.alive **and** N3.alive **then**
            d ← compute_d(N1.pos,N2.pos,N3.pos, N2.d2,N3.d3)
            dN2N3_list.append({d,N2,N3})
         end if
      end for
      if dN2N3_list={} **then**
         pass
      else
         {d',N2',N3'} ← entry in dN2N3_list with smallest d entry
         N1.d ← d'
         N1.cross ← compute_cross(N1.pos, N2'.pos, N3'.pos,N2'.cross, N3'.cross)
         narrow_band_list.append(N1)
      end if
   end for
end function

The *compute_cross* function is based on the formula suggested in Section 3.1 and the *compute_d* function is described in Appendix A.1. The process is illustrated in Figure 3.
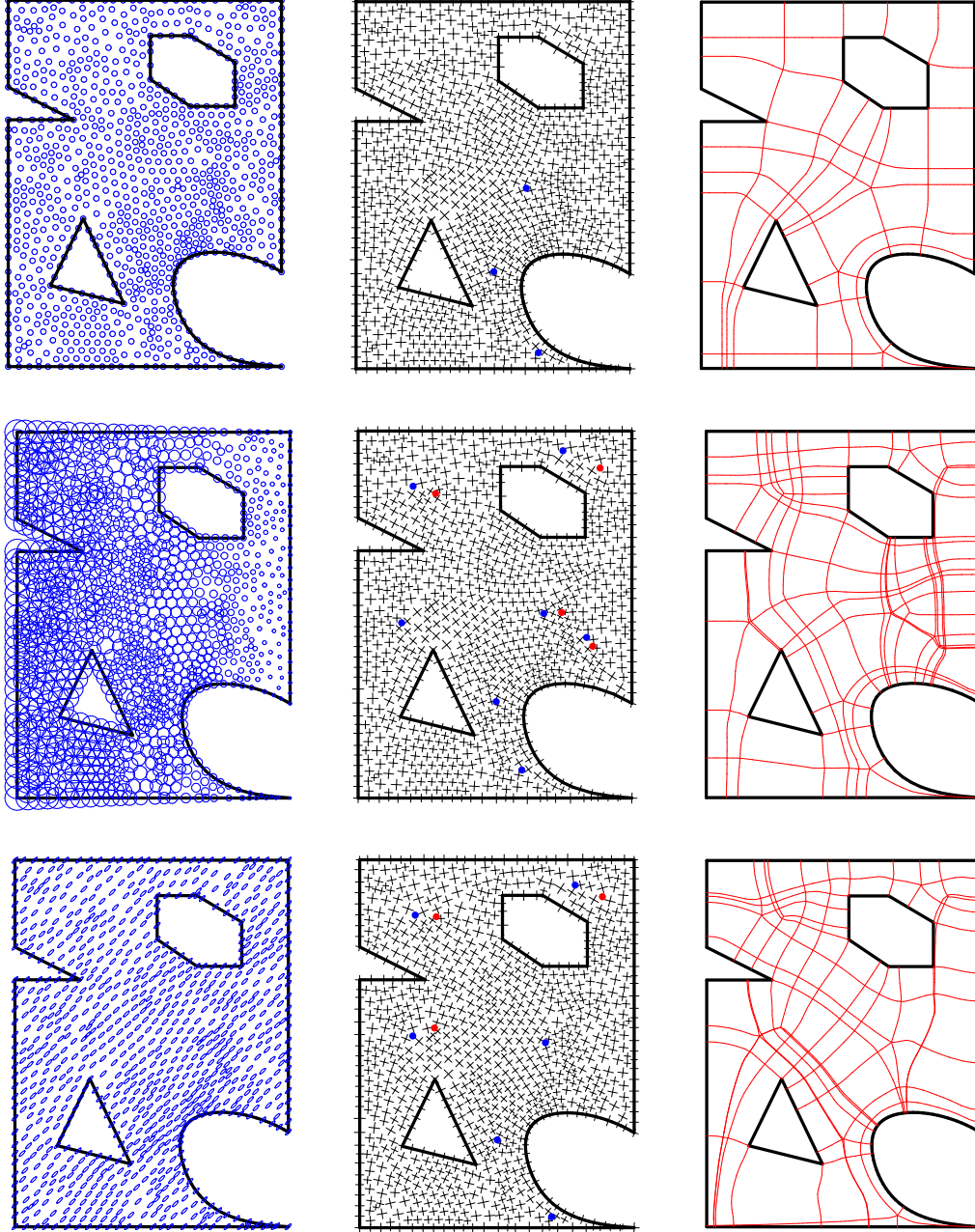
Figure 4: Examples of cross-fields and decompositions generated for three different target size fields. Blue and red dots indicate the occurrence of $k = 1$ and $k = -1$ type singularities respectively. (First row) The default unit target size field is used which results in a simple and aesthetically pleasing solution. (Second row) The target size field is isotropic and increases from right to left. The consequence is that edges tend to curve along paths running vertically which causes the occurrence of numerous extra singularities. (Third row) A constant anisotropic target element size field is used. The edges are to be aligned with the principal axes of the size tensor. Hence the E_{direc} term comes into play with a weighting factor proportional to the ratio of the principal axes lengths. Edges tend to travel diagonally across the surface which causes a different arrangement of singularities.

3.3 Cross-field singularities

Singularities occur in the cross-field in tri elements where the angle of a cross cannot be interpolated continuously between the crosses of its nodes.

From Bunin's *continuum theory*,

$$-k\frac{\pi}{2} = \sum_{i=0}^{n} \angle\big(\mathrm{PT}_{p_i \to p_{i+1}}\mathbf{c}(p_i), \mathbf{c}(p_{i+1})\big) + \iint\limits_{\text{enclosed-area}} K dA, \qquad (12)$$

where a closed-path is composed of segments between the points $p_0, p_1, ...p_n$ ($p_n=p_0$). PT is the parallel transport operator, $\mathbf{c}$ is a cross vector and $k \in \mathbb{Z}$ is the index of the enclosed singularity.

The total Gaussian curvature term can be calculated in the discretised representation of the surface from the three surface normals at the nodes using the standard angular deficit scheme (see *e.g.* [26]). Rotation angles for each edge in the tri mesh are calculated from the propagated cross-field by the method described in Section 3.1.1.

Hence, adding the rotation angles associated with each edge of a tri element with signs according to an anti-clockwise traversal and including the total curvature of the surface over the element (which is mostly negligible) gives a value equal to 0, $-\pi/2$ or $\pi/2$. These values correspond to cases of no singularity ($k = 0$), a positive singularity ($k = 1$) or a negative singularity ($k = -1$). It is certain that higher order singularities cannot occur because all edge rotation angles are in the range $(-\pi/4, \pi/4]$.

By virtue of the propagation method, singularities only occur in tri elements of inner-regions near the medial axis of the surface. For convex geometries with a high degree of symmetry, such as regular hexagons and circles, singularities occur in side-by-side tri elements. Possibly ideally they should be combined into a single higher order singularity. An additional routine would need to be incorporated for this task.

3.4 Tracing decomposition edges

With a cross assigned to every node and elements containing singularities identified, a piece-wise linear C^0 smooth cross-field can be defined. For a tri element not containing a singularity with nodes N_1, N_2 and N_3, a bilinear function describing the change in cross angle over the element with respect to $cross_1$ is determined. Referring to Eqn. (10), $\theta_1=0$, $\theta_2 = \Delta\theta_{12}$ and $\theta_3 = \Delta\theta_{13}$ where $\Delta\theta_{12}$ and $\Delta\theta_{13}$ are the rotations stored for the respective edges, as discussed in Section 3.2. $\theta(x, y)$ is taken to mean the rotation about the element normal. However, $\theta_3 - \theta_2 \neq \Delta\theta_{23}(= -\Delta\theta_{12} + \Delta\theta_{13} + \iint K dA)$ if $\iint K dA \neq 0$. This complication is ignored – a simplification that is not prone to cause problems because the total Gaussian curvature over an element is small.

Singularity elements are divided into three new elements by edges running from a new node at the centroid to the corner nodes. For a new element with nodes N_1, N_2, N_c where N_c is the new node, the singularity is placed at N_c. Thus, θ is set as a function of polar coordinates r, φ with the origin at N_c so that $\theta(r, \vartheta_2) - \theta(r, \vartheta_1) = \Delta\theta_{12}$.
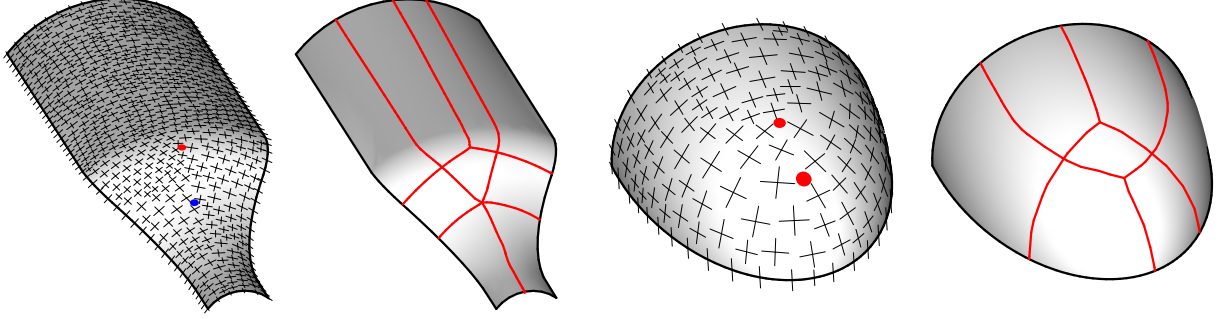
Figure 5: Cross-fields and decompositions generated on two curved surfaces with unit target size fields. In both cases the geodesic curvatures of the boundary edges are zero and the corner angles are $\pi/2$, therefore the singularities emerge as a consequence of the Gaussian curvatures of the surfaces.
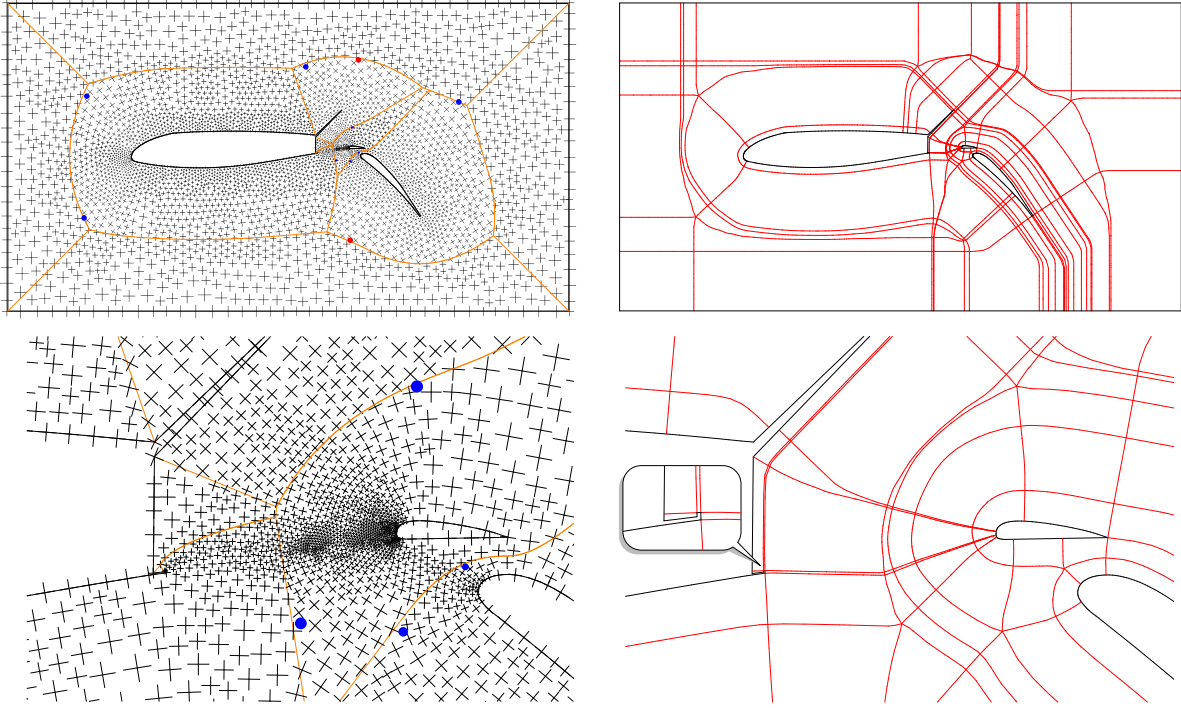
Figure 6: Multi-element aerofoil with flap and spoiler. (Left) Cross-field solution and close-up of the complex cove region. It is evident that the cross-field singularities occur on or near the medial axis represented by the orange lines. (Right) Intricate decomposition created by traced critical streamlines.

A simple numerical procedure is used for completing the task of tracing a cross-field streamline through an element given a curve tangent vector $\mathbf{d}_1$ and a position p_0 on element edge. First $\mathbf{d}_1$ is rotated onto the element plane to give $\mathbf{d}_1'$. Secondly, the reference vector is chosen from one of the four cross vectors at N_1 so that $\theta(p_0)$ best fits the angle that $\mathbf{d}_1'$ makes with the reference vector. Then Heun's method, a basic variation of the Runge-Kutta method, is used to integrate the streamline to another element edge according to

$\theta(x, y)$. The implemented method is an adaption of that described in [21, Section 3.2].

The critical streamlines radiating from singularities and boundary corners partition the surface into quadrilateral blocks, thus forming a multi-block decomposition suitable for all-quad meshing. The cross-field is evenly distorted around singularities and it is a straightforward matter to determine the starting *star geodesic* tangent vectors. The decomposition streamlines are traced until they meet the boundaries or until a pre-decided threshold distance or turn angle is exceeded.

Examples of cross-fields, singularities and multi-block decompositions are shown in Figures 4, 5 and 6.

4   DISCUSSION AND CONCLUSIONS

A novel method has been presented for generating a cross-field on a surface of arbitrary shape and genus with a provided tri mesh. Bunin's *continuum theory* is relied on as a basis for teasing out the best approach and for arguing the rationality of the approach taken. A set of specified direction constraints fix the crosses at certain nodes initially and the cross-field is propagated to the rest of the mesh by a fast marching method. The boundaries are selected automatically as the direction constraints unless alternatives are given, so that the result is boundary conforming. At each advancement step a new cross at a node is calculated by a simple formula derived by locally minimising an energy functional. The energy functional is designed to describe the composition of the cross-field smoothness and deviations from target element size changes and target directions over a region. Singularities of the propagated cross-field occur at isolated locations on the medial axis and are identified in elements of the tri mesh by a simple check.

The cross-field solver is the most important contribution of this work. The fast marching algorithm is efficient and solves the non-linear problem simply and quickly with an asymptotic complexity of $O(N \log N)$ [24]. For the example shown in Figure 6 with the tri mesh containing ∼6k nodes, the time taken for the entire process to finish was under a minute. The produced cross-fields can be tailored to suit a prescribed size field of a quad mesh. A potential application of the presented technology is as an effective cross-field initialisation method for global optimisation based solvers.

A basic streamline tracing algorithm is used to create the decomposition edges starting from singularities and boundary corners. Thus, multi-block decompositions of surfaces can be automatically constructed on which it is possible to generate all-quad meshes using widely-used algebraic mapping algorithms. Bunin's *continuum theory* deals only with the properties of a mesh with infinitesimal elements, hence it lacks guidelines for constructing a discrete mesh. For complex decompositions, such as that shown in Figure 6(right), long thin blocks are created by the simple streamline tracing algorithm. This does not cause an issue if the target element sizes are very small but a difficulty arises when the target sizes are greater than the block height. A post processing block simplification method could be used to overcome this problem by removing thin block rows with heights

much smaller than the target element sizes, with care needed to avoid violating the block topology. Depending on requirements, perhaps recently developed global parametrisation based algorithms, such as [5, 11, 9], might be a preferable way to generate the quad mesh rather than by decomposition.

Acknowledgements

REFERENCES

[1] Emil Praun, Adam Finkelstein, and Hugues Hoppe. Lapped textures. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '00, pages 465–470, New York, NY, USA, 2000. ACM Press/Addison-Wesley Publishing Co.

[2] Li-Yi Wei and Marc Levoy. Texture synthesis over arbitrary manifold surfaces. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '01, pages 355–360, New York, NY, USA, 2001. ACM.

[3] Greg Turk. Texture synthesis on surfaces. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '01, pages 347–354, New York, NY, USA, 2001. ACM.

[4] Eugene Zhang, Konstantin Mischaikow, and Greg Turk. Vector field design on surfaces. *ACM Trans. Graph.*, 25(4):1294–1326, October 2006.

[5] David Bommes, Henrik Zimmer, and Leif Kobbelt. Mixed-integer quadrangulation. *ACM Trans. Graph.*, 28(3):77:1–77:10, July 2009.

[6] Guy Bunin. A Continuum Theory for Unstructured Mesh Generation in Two Dimensions. *Comput. Aided Geom. Des.*, 25(1):14–40, January 2008.

[7] Nicolas Ray, Bruno Vallet, Laurent Alonso, and Bruno Lévy. Geometry Aware Direction Field Processing. *ACM Transactions on Graphics*, 29(1):Article 1, December 2009.

[8] Jonathan Palacios and Eugene Zhang. Rotational symmetry field design on surfaces. *ACM Trans. Graph.*, 26(3), July 2007.

[9] Felix Kälberer, Matthias Nieser, and Konrad Polthier. Quadcover - surface parameterization using branched coverings. *Computer Graphics Forum*, 26(3):375–384, 2007.

[10] Nicolas Ray, Wan Chiu Li, Bruno Lévy, Alla Sheffer, and Pierre Alliez. Periodic global parameterization. *ACM Trans. Graph.*, 25(4):1460–1485, October 2006.

[11] Yang Liu, Weiwei Xu, Jun Wang, Lifeng Zhu, Baining Guo, Falai Chen, and Guoping Wang. General planar quadrilateral mesh design using conjugate direction field. *ACM Trans. Graph.*, 30(6):140:1–140:10, December 2011.

[12] M. Nieser, U. Reitebuch, and K. Polthier. Cubecover parameterization of 3d volumes. *Computer Graphics Forum*, 30(5):1397–1406, 2011.

[13] Jin Huang, Tengfei Jiang, Yuanzhen Wang, Yiying Tong, and Hujun Bao. Automatic frame field guided hexahedral mesh generation. Technical Report MSU-CSE-12-9, Department of Computer Science, Michigan State University, East Lansing, Michigan, August 2012.

[14] Yufei Li, Yang Liu, Weiwei Xu, Wenping Wang, and Baining Guo. All-hex meshing using singularity-restricted field. *ACM Trans. Graph.*, 31(6):177:1–177:11, November 2012.

[15] Pierre Alliez, David Cohen-Steiner, Olivier Devillers, Bruno Lévy, and Mathieu Desbrun. Anisotropic polygonal remeshing. *ACM Trans. Graph.*, 22(3):485–493, July 2003.

[16] M. Marinov and L. Kobbelt. Direct anisotropic quad-dominant remeshing. In *Proceedings of the 12th Pacific Conference on Computer Graphics and Applications*, pages 207 – 216, oct. 2004.

[17] R. Bruce Simpson. Anisotropic mesh transformations and optimal error control. *Applied Numer. Math*, 14:183–198, 1992.

[18] E. F. D'Azevedo. Are bilinear quadrilaterals better than linear triangles? *SIAM J. Sci. Comput.*, 22(1):198–217, January 2000.

[19] Guy Bunin. Towards Unstructured Mesh Generation Using the Inverse Poisson Problem. *arXiv/0802.2399*, 2008.

[20] Mirela Ben-Chen, Craig Gotsman, and Guy Bunin. Conformal Flattening by Curvature Prescription and Metric Scaling. *Computer Graphics Forum*, 27(2):449–458, 2008.

[21] Nicolas Kowalski, Franck Ledoux, and Pascal Frey. A PDE Based Approach to Multidomain Partitioning and Quadrilateral Meshing. In Xiangmin Jiao and Jean-Christophe Weill, editors, *Proceedings of the 21st International Meshing Roundtable*, pages 137–154. Springer Berlin Heidelberg, 2013.

[22] Jin Huang, Yiying Tong, Hongyu Wei, and Hujun Bao. Boundary aligned smooth 3d cross-frame field. In *Proceedings of the 2011 SIGGRAPH Asia Conference*, SA '11, pages 143:1–143:8, New York, NY, USA, 2011. ACM.

[23] Yu-Kun Lai, Miao Jin, Xuexiang Xie, Ying He, Jonathan Palacios, Eugene Zhang, Shi-Min Hu, and Xianfeng Gu. Metric-driven rosy field design and remeshing. *IEEE Transactions on Visualization and Computer Graphics*, 16(1):95–108, January 2010.

[24] J. A. Sethian. *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials science*. Cambridge University Press, 2 edition, 6 1999.

[25] Renato N. Elias, Marcos A. D. Martins, and Alvaro L. G. A. Coutinho. Simple finite element-based computation of distance functions in unstructured grids. *International Journal for Numerical Methods in Engineering*, 72(9):1095–1110, 2007.

[26] Zhiqiang Xu and Guoliang Xu. Discrete schemes for gaussian curvature and their convergence. *CoRR*, abs/0804.1046, 2008.

A Appendix

A.1 Calculate distance function in tri element

The Eikonal equation states

$$\|\nabla d\| = 1, \quad d \in \mathbb{R} > 0, \tag{13}$$

where d is the distance function. In a tri element with the distance set at two nodes, N_2 and N_3, the distance at the remaining node, N_1, is calculated as follows.

The distances relative to d_2 are:

$$
\begin{aligned}
u_1 &= d_1 - d_2, \\
u_2 &= 0, \\
u_3 &= d_3 - d_1.
\end{aligned}
\tag{14}
$$

A local Cartesian coordinate frame, F, on the element plane with its origin at N_2 is used and u is approximated by a bilinear function,

$$u(x, y) = \alpha x + \beta y, \quad \alpha, \beta \in \mathbb{R}, \tag{15}$$

where F is chosen such that its x-axis is along edge-12 and the y-axis points into the element, so that $\beta > 0$. By Eqn. (13),

$$
\begin{aligned}
\|\nabla u\| &= \alpha^2 + \beta^2 = 1, \\
&\Rightarrow \beta = \sqrt{1 - \alpha^2}.
\end{aligned}
\tag{16}
$$

Substituting known values in Eqn (15) gives

$$
\begin{aligned}
u_3 &= \alpha x_3 + \sqrt{1 - \alpha^2} y_3,^0 \\
&\Rightarrow \alpha = \frac{u_3}{x_3}
\end{aligned}
\tag{17}
$$

Therefore,

$$u_1 = \frac{u_3}{x_3} x_1 + \sqrt{1 - \frac{u_3}{x_3}^2}\, y_1, \tag{18}$$

and finally,

$$d_1 = d_2 + \frac{u_3}{x_3} x_1 + \sqrt{1 - \frac{u_3}{x_3}^2}\, y_1. \tag{19}$$

A POSTERIORI ERROR ESTIMATES FOR A NEUMANN-NEUMANN DOMAIN DECOMPOSITION ALGORITHM APPLIED TO CONTACT PROBLEMS

L. GALLIMARD[*], D. CHOI[†] AND T. SASSI[†]

[*]Laboratoire Energétique, Mécanique, Electromagnétisme (LEME),
Université Paris Ouest Nanterre-La Défense
50 rue de Sèvres - 92410 Ville d'Avray, France
laurent.gallimard@u-paris10.fr

[†]Laboratoire de Mathématiques Nicolas Oresme (LMNO),
Université de Caen Basse-Normandie,
Boulevard Marechal Juin 14032 Caen Cedex, France
daniel.choi,taoufik.sassi@unicaen.fr

Key words: Error estimation, domain decomposition algorithm, contact problem, discretization error, algebraic error

Abstract. In this paper we present an error estimator for unilateral contact problem solved by a Neumann-Neumann domain decomposition algorithm. We also propose errors indicators that take into account the different approximation errors introduced by the computation: the finite element spatial discretisation and the domain decomposition algorithm.

1 INTRODUCTION

Contact problems are frequent in structural analysis. They are characterized by inequality constraints such as non-interpenetration conditions, sign condition on the normal constraints, and an active contact, an area that is a priori unknown. Several approaches exist for solving the non linear equations issued from the finite element discretization of frictionless contact problems. In this work, we consider a natural Neuman-Neumann domain decomposition algorithm, in which each iterative step consists of a Dirichlet problem for the one body, a contact problem for the other one and two Neumann problems to coordinate contact stresses. Two main approximation errors are introduced by this algorithm: a discretization error due to the finite element method (FEM) and an algebraic error due to the Neuman-Neuman domain decomposition algorithm (NNDD).

The objective of this paper is to present an a posteriori global error estimator for a frictionless contact problem, solved by a NNDD algorithm and two errors indicators which

allow to estimate the part of the error due to the spatial discretization and the part of the error due to the domain decomposition algorithm. We show how to extend the error measure in the constitutive relation developed in [4] for contact problems solved by a Neumann-Dirichlet domain decomposition algorithm and how to modify the construction of the admissible fields. The proposed errors estimators and indicators are studied on 2D-examples.

2   CONTACT PROBLEM

We consider the problem of two elastic bodies Ω^1 and Ω^2 in unilateral contact along an interface Γ_c as shown on Figure (1). We choose the orientation of the contact zone Γ_c by setting: $\mathbf{n}^c = \mathbf{n}^1$. Then, we introduce on the interface Γ_C the functions $\mathbf{w}^1$, $\mathbf{w}^2$, representing two displacement fields (one on each side of the interface), $\mathbf{t}^1$, $\mathbf{t}^2$, representing two fields of surface density forces (stresses transmitted to Ω^1 and Ω^2) and $\mathbf{t}^c$ an interior field of surface density forces. The problem of unilateral contact consists of finding $(\mathbf{u}^\alpha, \boldsymbol{\sigma}^\alpha)$ defined on Ω^α ($\alpha = 1, 2$) and $(\mathbf{w}^1, \mathbf{w}^2, \mathbf{t}^1, \mathbf{t}^2, \mathbf{t}^c)$ defined on Γ_C such that $(\mathbf{u}^\alpha, \mathbf{w}^\alpha, \mathbf{w}^c)$ satisfy the kinematic conditions (1), $(\boldsymbol{\sigma}^\alpha, \mathbf{t}^\alpha, \mathbf{t}^c)$ satisfy the equilibrium equations (2), $(\mathbf{u}^\alpha, \boldsymbol{\sigma}^\alpha)$ satisfy the elastic constitutive relation (3), $(\mathbf{w}^c, \mathbf{t}^c)$ satisfy the contact constitutive relation (4) (for the sake of simplicity we do not consider volumic forces).
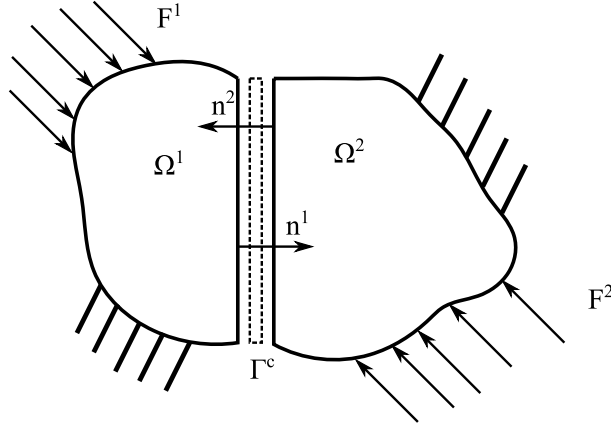
Figure 1: Notations

$$\mathbf{u} \in \mathcal{V}^\alpha, \quad \mathbf{u}^\alpha - \mathbf{w}^\alpha = 0 \text{ and } \mathbf{w}^c = \mathbf{w}^1 - \mathbf{w}^2 \text{ on } \Gamma_c \tag{1}$$

$$\forall \, \mathbf{v} \in \mathcal{V}_0^\alpha, \quad -\int_{\Omega^\alpha} \boldsymbol{\sigma}^\alpha \boldsymbol{\varepsilon}(\mathbf{v}) \, dV + \int_{\Gamma_N^\alpha} \mathbf{F}^\alpha \mathbf{v} \, dS + \int_{\Gamma_c} \mathbf{t}^\alpha \mathbf{v} \, dS = 0 \tag{2}$$

$$\mathbf{t}^c - \mathbf{t}^1 = 0 \text{ and } \mathbf{t}^c + \mathbf{t}^2 = 0 \text{ on } \Gamma_c$$

$$\boldsymbol{\sigma}^\alpha = \mathbf{K}^\alpha \boldsymbol{\varepsilon}(\mathbf{u}^\alpha) \text{ in } \Omega^\alpha, \tag{3}$$

$$\phi(-\mathbf{w}^c) + \phi^*(\mathbf{t}^c) + \mathbf{t}^c.\mathbf{w}^c = 0 \ \text{ on } \Gamma_c, \tag{4}$$

where $\mathcal{V}^\alpha = \{\mathbf{v} \in H^1(\Omega^\alpha); \mathbf{u} = \mathbf{u}_D^\alpha \text{ on } \Gamma_D^\alpha\}$, $\mathcal{V}_0^\alpha = \{\mathbf{v} \in H^1(\Omega^\alpha); \mathbf{v} = 0 \text{ on } \Gamma_D^\alpha\}$, and for any vector $\mathbf{v}$, the normal and the tangential components are defined according to $v_n = \mathbf{v}.\mathbf{n}^c$ and $v_t = \mathbf{v} - v_n \mathbf{n}^c$ and the convex potentials ϕ and ϕ^* are defined by

$$\phi(\mathbf{v}) = \begin{cases} 0 & \text{if } v_n \geq 0 \\ +\infty & \text{otherwise} \end{cases} \text{ and } \phi^*(\mathbf{g}) = \begin{cases} 0 & \text{if } g_n \leq 0 \text{ and } g_t = 0 \\ +\infty & \text{otherwise} \end{cases} \tag{5}$$

moreover for any pair (w, t) defined on Γ_c, the Legendre-Fenchel inequality leads to

$$\phi(-\mathbf{w}^c) + \phi^*(\mathbf{t}^c) + \mathbf{t}^c.\mathbf{w}^c \geq 0 \tag{6}$$

Remark: Following [2], the relation defined by equation (4) is equivalent to the Coulomb's constitutive law (7) in a frictionless case.

$$\mathbf{w}_n^c \leq 0, \quad \mathbf{t}_n^c \leq 0, \quad \mathbf{t}_n^c \, \mathbf{w}_n^c = 0 \text{ and } \mathbf{t}_t^c = 0 \ \text{ on } \Gamma_c, \tag{7}$$

3 DOMAIN DECOMPOSITION ALGORITHM

In this section we briefly recall the Neumann-Neumann domain decomposition algorithm used to solve the unilareral contact problem defined by equations (1)–(4). Given a non-negative parameter θ and an initial arbitrary normal displacement λ_1 defined on Γ_c, we define two sequences of displacements $\mathbf{u}_p^\alpha$ on each solid Ω^α, $\alpha = 1, 2$. Each iteration p of the NNDD algorithm is divided in two successive steps.

- Step 1 – Two independent elasticity problems are solved on Ω_1 and Ω_2:

  1. In Ω^1, the variational problem writes: Find $(\mathbf{u}_p^1, \boldsymbol{\sigma}_p^1)$ defined on Ω^1 and $(\mathbf{w}_p^1, \mathbf{t}_p^1)$ defined on Γ_c such that

$$\mathbf{u}_p^1 = \mathbf{u}_D^1 \text{ on } \Gamma_D^1, \quad \mathbf{u}_p^1 - \mathbf{w}_p^1 = 0 \text{ and } w_p^1\mathbf{n}^1 = \lambda_p \text{ on } \Gamma_c \tag{8}$$

$$\forall \, \mathbf{v} \in \mathcal{V}_0^1, \quad -\int_{\Omega^1} \boldsymbol{\sigma}_p^1 : \boldsymbol{\varepsilon}(\mathbf{v}) \, dV + \int_{\Gamma_N^1} \mathbf{F}^1\mathbf{v} \, dS = 0 \tag{9}$$

$$\mathbf{t}_p^1 = \boldsymbol{\sigma}_p^1\mathbf{n}^1 \text{ on } \Gamma_c$$

$$\boldsymbol{\sigma}_p^1 = \mathbf{K}^1\boldsymbol{\varepsilon}(\mathbf{u}_p^1) \text{ in } \Omega^1, \tag{10}$$

  2. In Ω^2, with the given λ_p normal displacement defined on Γ_c, we solve the following variational problem corresponding to a unilateral frictionless contact problem on Γ_c :

$$\mathbf{u}_p^2 = \mathbf{u}_D^2 \text{ on } \Gamma_D^2, \quad \mathbf{u}_p^2 - \mathbf{w}_p^2 = 0 \text{ and } \mathbf{w}_p^c = \lambda_p\mathbf{n}^1 - \mathbf{w}_p^2 \text{ on } \Gamma_c \tag{11}$$

$$\forall \mathbf{v} \in \mathcal{V}_0^2, \quad -\int_{\Omega^2} \boldsymbol{\sigma}_p^2 : \boldsymbol{\varepsilon}(\mathbf{v}) \, dV + \int_{\Gamma_N^2} \mathbf{F}^2 \mathbf{v} \, dS + \int_{\Gamma_c} \mathbf{t}_p^2 \mathbf{v} \, dS = 0 \qquad (12)$$

$$\mathbf{t}_p^c + \mathbf{t}_p^2 = 0 \text{ on } \Gamma_c$$

$$\boldsymbol{\sigma}_p^2 = \mathbf{K}^2 \boldsymbol{\varepsilon}(\mathbf{u}_p^2) \text{ in } \Omega^2, \qquad (13)$$

$$\phi(-\mathbf{w}_p^c) + \phi^*(\mathbf{t}_p^c) + \mathbf{t}_p^c . \mathbf{w}_p^c = 0 \text{ on } \Gamma_c, \qquad (14)$$

- Step 2 – With $\mathbf{t}_p^1$ and $\mathbf{t}_p^2$ obtained in step 1, we solve two independent "Neumann type" problems:

In Ω^1, we solve

$$\begin{cases} \text{Find } \mathbf{w}^1 \in \mathbf{V}^1 \text{ such that} \\ -\int_{\Omega^1} \mathbf{K}^1 \boldsymbol{\varepsilon}(\mathbf{w}^1) : \boldsymbol{\varepsilon}(\mathbf{v}) \, dV = -\int_{\Gamma_c} \frac{1}{2}(\mathbf{t}_p^1 + \mathbf{t}_p^2).(\mathbf{u}^* - \mathbf{w}^1) \quad \forall \mathbf{u}^* \in \mathbf{V}^1. \end{cases} \qquad (15)$$

In Ω^2, we solve

$$\begin{cases} \text{Find } \mathbf{w}^2 \in \mathbf{V}^2 \text{ such that} \\ -\int_{\Omega^2} \mathbf{K}^2 \boldsymbol{\varepsilon}(\mathbf{w}^2) : \boldsymbol{\varepsilon}(\mathbf{v}) \, dV = \int_{\Gamma_c} \frac{1}{2}(\mathbf{t}_p^1 + \mathbf{t}_p^2).(\mathbf{u}^* - \mathbf{w}^2) \quad \forall \mathbf{u}^* \in \mathbf{V}^2. \end{cases} \qquad (16)$$

Let ε_t be the precision of the algorithm, we have the alternative :

1. If ε_τ is small enough, the algorithm stops.

2. Else, the normal displacement λ_p is updated :

$$\lambda_{p+1} := \lambda_p + \theta(\mathbf{w}^1 - \mathbf{w}^2).\mathbf{n}$$

and we return to step 1 for iteration $p+1$.

If $\mathbf{t}_p^1 + \mathbf{t}_p^2 = 0$, it means that the equilibrium is satisfied on the contact interface, in other words the solutions $\mathbf{u}^1$ and $\mathbf{u}^2$ of step 1 constitute the unique solution of the reference problem (1)– (4).

The convergence is obtained when $|\mathbf{w}^1 - \mathbf{w}^2| \longrightarrow 0$. The proof of convergence of the NNDD algorithm (1)-(16) is given in [5] for any sufficiently small $\theta > 0$: There is a $\theta_0 > 0$ such that for any $0 < \theta \leq \theta_0$, the NNDD algorithm for unilateral frictionless contact converges.

At each step p, the approximate solution of problems (8-10) and (11-14) are computed using a classical F.E. method. The finite element spaces are denoted $\mathcal{V}_h^\alpha (\subset \mathcal{V}^\alpha)$ and the approximate solution $(\mathbf{d}_{p,h}, \mathbf{s}_{p,h})$ with $\mathbf{d}_{p,h} = (\mathbf{u}_{p,h}^1, \mathbf{w}_{p,h}^1, \mathbf{u}_{p,h}^2, \mathbf{w}_{p,h}^2, \mathbf{w}_{p,h}^c)$ and $\mathbf{s}_{p,h} = (\boldsymbol{\sigma}_{p,h}^1, \mathbf{t}_{p,h}^1, \boldsymbol{\sigma}_{p,h}^2, \mathbf{t}_{p,h}^2, \mathbf{t}_{p,h}^c)$.

4 ERROR ESTIMATION

4.1 Error in the constitutive relation

To develop an error estimation for a contact problem we use a method based on the constitutive relation error [6]. We recall here the error measure proposed in [1] for a global unilateral contact problem. Let us consider an approximate solution of problem defined by equations (1-4), denoted $(\hat{\mathbf{u}}, \hat{\mathbf{c}})$. The pair $(\hat{\mathbf{d}}, \hat{\mathbf{s}})$ is said to be an *admissible* solution if $(\hat{\mathbf{d}}, \hat{\mathbf{s}}) \in \mathcal{U}_{ad} \times \mathcal{S}_{ad}$ with

- $\mathcal{U}_{ad} = \{\hat{\mathbf{d}} = (\hat{\mathbf{u}}^1, \hat{\mathbf{u}}^2, \hat{\mathbf{w}}^1, \hat{\mathbf{w}}^2, \hat{\mathbf{w}}^c)$ such that $\hat{\mathbf{d}}$ satisfy Eq. (1) and $\phi(-\hat{\mathbf{w}}^c) = 0\}$

- $\mathcal{S}_{ad} = \{\hat{\mathbf{s}} = (\hat{\boldsymbol{\sigma}}^1, \hat{\boldsymbol{\sigma}}^2, \hat{\mathbf{t}}^1, \hat{\mathbf{t}}^2, \hat{\mathbf{t}}^c)$ such that $\hat{\mathbf{s}}$ satisfy Eq. (2) and $\phi^*(\hat{\mathbf{t}}^c) = 0\}$

The constitutive relation error on the pair $(\hat{\mathbf{u}}, \hat{\mathbf{c}})$ is defined by

$$e_{CRE}(\hat{\mathbf{d}}, \hat{\mathbf{s}}) = \left[\sum_{\alpha=1}^{2} \|\hat{\boldsymbol{\sigma}}^\alpha - \mathbf{K}^\alpha \boldsymbol{\varepsilon}(\hat{u}^\alpha)\|_{\boldsymbol{\sigma}, \Omega^\alpha}^2 + 2 \int_{\Gamma_c} \hat{\mathbf{t}}^c \hat{\mathbf{w}}^c \, dS \right]^{1/2}, \tag{17}$$

As an extension of the Prager-Synge theorem it was shown in [1] that

$$e_{CRE}(\hat{\mathbf{d}}, \hat{\mathbf{s}}) \geq \left[\sum_{\alpha=1}^{2} \|\hat{\boldsymbol{\sigma}}^\alpha - \boldsymbol{\sigma}^\alpha\|_{\boldsymbol{\sigma}, \Omega^\alpha}^2 + \|\hat{\mathbf{u}}^\alpha - \mathbf{u}^\alpha\|_{\mathbf{u}, \Omega^\alpha}^2 \right]^{1/2} \tag{18}$$

4.2 An a posteriori error estimator for a discretized Neumann-Neumann domain decomposition algorithm

When the formulation of the contact problem is obtained by a domain decomposition method the global error depends not only on the FE discretization error but also on the convergence of the iterative algorithm used (i.e. an algebraic error). Here, we develop an error measure based on the constitutive relation error for an unilateral contact problem solved by a Neumann-Neumann domain decomposition algorithm. Let us introduce new admissible spaces defined at each iteration p of the NNDD algorithm by

- $\mathcal{U}_{ad}^1(\lambda_p) = \{\hat{\mathbf{d}}^1 = (\hat{\mathbf{u}}^1, \hat{\mathbf{w}}^1)$ such that $\hat{\mathbf{d}}^1$ satisfy Eq. (8)$\}$,

- $\mathcal{S}_{ad}^1 = \{\hat{\mathbf{s}}^1 = (\hat{\boldsymbol{\sigma}}^1, \hat{\mathbf{t}}^1)$ such that $\hat{\mathbf{s}}^1$ satisfy Eq. (9)$\}$,

- $\mathcal{U}_{ad}^2(\lambda_p) = \{\hat{\mathbf{d}}^2 = (\hat{\mathbf{u}}^2, \hat{\mathbf{w}}^2, \hat{\mathbf{w}}^c)$ such that $\hat{\mathbf{d}}^2$ satisfy Eq. (11) and $\phi(-\hat{\mathbf{w}}^c) = 0\}$,

- $\mathcal{S}_{ad}^2(\lambda_p) = \{\hat{\mathbf{s}}^2 = (\hat{\boldsymbol{\sigma}}^2, \hat{\mathbf{t}}^2, \hat{\mathbf{t}}^c)$ such that $\hat{\mathbf{s}}^2$ satisfy Eq. (12) and $\phi^*(\hat{\mathbf{t}}^c) = 0\}$.

The pair $(\hat{\mathbf{d}}_p^1, \hat{\mathbf{s}}_p^1) \in \mathcal{U}_{ad}^1(\lambda_p) \times \mathcal{S}_{ad}^1$ is the solution of problem (8-10) if

$$e_{CRE}^1(\hat{\mathbf{d}}_p^1, \hat{\boldsymbol{\sigma}}_p^1) = \left[\|\hat{\boldsymbol{\sigma}}_p^1 - \mathbf{K}^2 \boldsymbol{\varepsilon}(\hat{\mathbf{u}}_p^1)\|_{\boldsymbol{\sigma}, \Omega^1}^2 \right]^{1/2} = 0 \tag{19}$$

The pair $(\hat{\mathbf{d}}_p^2, \hat{\boldsymbol{\sigma}}_p^2) \in \mathcal{U}_{ad}^2(\lambda_p) \times \mathcal{S}_{ad}^2(\lambda_p)$ is the solution of problem (11-14) if

$$e_{CRE}^2(\hat{\mathbf{d}}_p^2, \hat{\mathbf{s}}_p^2) = \left[\|\hat{\boldsymbol{\sigma}}_p^2 - \mathbf{K}^2\boldsymbol{\varepsilon}(\hat{\mathbf{u}}_p^2)\|_{\boldsymbol{\sigma},\Omega^2}^2 + 2\int_{\Gamma_c} \hat{\mathbf{t}}_p^c \, \hat{\mathbf{w}}_p^c \, dS \right]^{1/2} = 0 \qquad (20)$$

However, the pair $(\hat{\mathbf{d}}_p = (\hat{\mathbf{d}}_p^1, \hat{\mathbf{d}}_p^2), \hat{\mathbf{s}}_p = (\hat{\mathbf{s}}_p^1, \hat{\mathbf{s}}_p^2))$, is not an admissible solution for the unilateral contact problem (i.e. $\notin \mathcal{U}_{ad} \times \mathcal{S}_{ad}$) because the equilibrium equation (2) is not necessarily satisfied as $\hat{\mathbf{t}}_p^c - \hat{\mathbf{t}}_p^1 = 0$ has not been imposed. The quantity e_{CRE} defined by equation (21) is an error estimator for the problem defined by equations (8-14), when

$$e_{CRE}(\hat{\mathbf{d}}_p^1, \hat{\mathbf{s}}_p^1, \hat{\mathbf{d}}_p^2, \hat{\mathbf{s}}_p^2) = \left[\left(e_{CRE}^1(\hat{\mathbf{d}}_p^1, \hat{\mathbf{s}}_p^1)\right)^2 + \left(e_{CRE}^2(\hat{\mathbf{d}}_p^2, \hat{\mathbf{s}}_p^2)\right)^2 \right]^{\frac{1}{2}} \qquad (21)$$

We have the following property

$$e_{CRE}(\hat{\mathbf{d}}_p^1, \hat{\mathbf{s}}_p^1, \hat{\mathbf{d}}_p^2, \hat{\mathbf{s}}_p^2) = 0 \Leftrightarrow (\hat{\mathbf{d}}_p^1, \hat{\mathbf{s}}_p^1, \hat{\mathbf{d}}_p^2, \hat{\mathbf{s}}_p^2) \text{ is the exact solution of (8-14) for a fixed } \lambda_p.$$

This error measure quantify the error due to the Finite Element discretization at each step of the algorithm. In order to obtain a global error estimator for the contact problem, let us define an admissible solution for the unilateral contact problem $(\hat{\hat{\mathbf{d}}}_p = (\hat{\hat{\mathbf{d}}}_p^1, \hat{\hat{\mathbf{d}}}_p^2), \hat{\hat{\mathbf{s}}}_p = (\hat{\hat{\mathbf{s}}}_p^1, \hat{\hat{\mathbf{s}}}_p^2))$ such that

$$(\hat{\hat{\mathbf{d}}}_p^1, \hat{\hat{\mathbf{s}}}_p^1) \in \mathcal{U}_{ad}^1(\lambda_p) \times \mathcal{S}_{ad}^1, \quad (\hat{\hat{\mathbf{d}}}_p^2, \hat{\hat{\mathbf{s}}}_p^2) \in \mathcal{U}_{ad}^2(\lambda_p) \times \mathcal{S}_{ad}^2(\lambda_p) \text{ and } \hat{\hat{\mathbf{t}}}_p^c - \hat{\hat{\mathbf{t}}}_p^1 = 0$$

The global error estimator for the contact problem is defined by

$$\eta^{glo} = e_{CRE}(\hat{\hat{\mathbf{d}}}_p^1, \hat{\hat{\mathbf{s}}}_p^1, \hat{\hat{\mathbf{d}}}_p^2, \hat{\hat{\mathbf{s}}}_p^2) = \left[\left(e_{CRE}^1(\hat{\hat{\mathbf{d}}}_p^1, \hat{\hat{\mathbf{s}}}_p^1)\right)^2 + \left(e_{CRE}^2(\hat{\hat{\mathbf{d}}}_p^2, \hat{\hat{\mathbf{s}}}_p^2)\right)^2 \right]^{\frac{1}{2}} \qquad (22)$$

We have the following property

$$e_{CRE}(\hat{\hat{\mathbf{d}}}_p^1, \hat{\hat{\mathbf{s}}}_p^1, \hat{\hat{\mathbf{d}}}_p^2, \hat{\hat{\mathbf{s}}}_p^2) = 0 \Leftrightarrow (\hat{\hat{\mathbf{d}}}_p^1, \hat{\hat{\mathbf{s}}}_p^1, \hat{\hat{\mathbf{d}}}_p^2, \hat{\hat{\mathbf{s}}}_p^2) \text{ is the exact solution of the global unilateral contact problem de}$$

The admissible displacement fields are easily recovered, since the finite element fields satisfy the kinematic constraints and $\phi(\mathbf{w}_{p,h}^c) = 0$.

$$\hat{\hat{\mathbf{d}}}_p^1 = (\mathbf{u}_{p,h}^1, \mathbf{w}_{p,h}^1) \text{ and } \hat{\hat{\mathbf{d}}}_p^2 = (\mathbf{u}_{p,h}^2, \mathbf{w}_{p,h}^2, \mathbf{w}_{p,h}^c)$$

However, the stress fields and the traction forces $\mathbf{s}_{p,h}$ computed by the algorithm do not satisfy the equilibrium equations. The pair $(\hat{\mathbf{s}}_p^1, \hat{\mathbf{s}}_p^2)$ is recovered from the finite element solution and the data in 3 steps

273

- The first step, consist in recovering admissible traction fields $(\hat{\tilde{\mathbf{t}}}^1, \hat{\tilde{\mathbf{t}}}^2, \hat{\tilde{\mathbf{t}}}^c)$. We built a traction $\hat{\tilde{\mathbf{t}}}^c$ such that $\phi^*(\hat{\tilde{\mathbf{t}}}^c) = 0$ and which minimize in the least square sense $J(\hat{\tilde{\mathbf{t}}}^c)$

$$J(\hat{\tilde{\mathbf{t}}}^c) = \int_{\Gamma_c} \left(\hat{\tilde{\mathbf{t}}}^c - \frac{1}{2} \left(\mathbf{t}_{h,p}^1 - \mathbf{t}_{h,p}^2 \right) \right)^2 dS$$

- The second step, consist in recovering stress fields $\tilde{\boldsymbol{\sigma}}_{h,p}^{\alpha}$ that satify the FE-equilibrium equations on each solid Ω^{α}. Let $\tilde{\mathbf{u}}_{h,p}^{\alpha} \in \mathcal{V}_h^{\alpha}$ such that $\tilde{\boldsymbol{\sigma}}_{h,p}^{\alpha} = \mathbf{K}^{\alpha}\boldsymbol{\varepsilon}(\tilde{\mathbf{u}}_{h,p}^{\alpha})$ and

$$\forall \mathbf{v} \in \mathcal{V}_{h,0}^{\alpha}, \quad -\int_{\Omega^{\alpha}} \mathbf{K}^{\alpha}\boldsymbol{\varepsilon}(\tilde{\mathbf{u}}_{h,p}^{\alpha}) : \boldsymbol{\varepsilon}(\mathbf{v})\, dV + \int_{\Gamma_N^{\alpha}} \mathbf{F}^{\alpha}\mathbf{v}\, dS + \int_{\Gamma_c} \hat{\tilde{\mathbf{t}}}^{\alpha}\mathbf{v}\, dS = 0$$

- The recovery of equilibrated stress fields $\hat{\tilde{\boldsymbol{\sigma}}}^{\alpha}$ from $\tilde{\boldsymbol{\sigma}}_{h,p}^{\alpha}$ in each subdomain Ω^{α} is the most technical point. This step is performed with a traction-free recovery technique developed in [4].

4.3 Error indicators for the NNDD algorithm and for the FE discretization

Following the method proposed in [7, 8], we propose here two error indicators that allow us to estimate separately the part of the error due to the FE discretization from the part due to the NNDD algorithm. The discretization error is defined as the limit of the global error when the convergence criterion of the iterative algorithm tends to zero. The NNDD algorithm error is defined as the limit of the global error as the mesh size h tends to zero.

To define FE discretization error indicator η^{FE}, let us consider the reference problem defined by the step p of the NNDD algorithm: Find $\mathbf{d}_p = (\mathbf{u}_p^1, \mathbf{w}_p^1, \mathbf{u}_p^2, \mathbf{w}_p^2, \mathbf{w}_p^c)$ and $\mathbf{s}_p = (\boldsymbol{\sigma}_p^1, \mathbf{t}_p^1, \boldsymbol{\sigma}_p^2, \mathbf{t}_p^2, \mathbf{t}_p^c)$ that satisfy equations (8-14). The only approximation introduced between $(\mathbf{d}_p, \mathbf{s}_p)$ and the finite element solution $(\mathbf{d}_{p,h}, \mathbf{s}_{p,h})$ is the FE discretization. We have shown in section (4.2) that the error in the constitutive relation $e_{CRE}(\hat{\mathbf{d}}_p^1, \hat{\mathbf{s}}_p^1, \hat{\mathbf{d}}_p^2, \hat{\mathbf{s}}_p^2)$ defined in equation (21) is an error estimator for this reference problem. The quantity $e_{CRE}(\hat{\mathbf{d}}_p^1, \hat{\mathbf{s}}_p^1, \hat{\mathbf{d}}_p^2, \hat{\mathbf{s}}_p^2)$ is used to define a FE discretization error indicator for the unilateral contact problem

$$\eta^{FE} = e_{CRE}(\hat{\mathbf{d}}_p^1, \hat{\mathbf{s}}_p^1, \hat{\mathbf{d}}_p^2, \hat{\mathbf{s}}_p^2) \tag{23}$$

To define NNDD algorithm error indicator η^{NNDD}, let us consider the reference problem defined by finite element discretization of the unilateral contact problem (Eq. (1)–(4)). The only approximation introduced between the solution of this discretized problem and the finite element solution $(\mathbf{d}_{p,h}, \mathbf{s}_{p,h})$ is the approximation introduced by the NNDD algorithm. Let $(\hat{\mathbf{d}}_h, \hat{\mathbf{s}}_h)$ and admissible solution for this problem (i.e. that satisfy the discretized version of Eq. (1)–(2)). The error in the constitutive relation $e_{CRE}(\hat{\mathbf{d}}_h, \hat{\mathbf{s}}_h)$ defined by equation (17) is an error estimator for this reference problem, and is used to define NNDD algorithm error indicator for the unilateral contact problem

$$\eta^{NNDD} = e_{CRE}(\hat{\mathbf{d}}_h, \hat{\mathbf{s}}_h) \tag{24}$$

5 Numerical results

The considered example is shown on figure (2). The lower boundary of structure Ω_1 is clamped, on structure Ω_2 the applied force F^2 has a linear distribution ($F^2_{max} = 10^7$) and the applied displacement is $u^2_D = -10^{-4}$. The Young's modulus for both structures is $E = 210\,GPa$ and the Poisson's ratio is $\nu = 0.27$. The coefficient θ of the NNDD algorithm is set to 0.25.
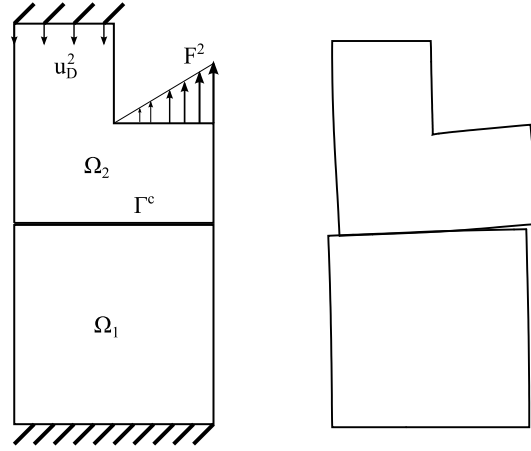
Figure 2: Unilateral contact reference model (left) - distorted structures (right)

We first study the evolution of the global error estimator η^{glo}, of the FE error indicator η^{FE}, and of the NNDD error indicator η^{NNDD} as a function of the number n_{DoF} of the degree of freedom (DoF), for a fixed number iterations of the NDDD algorithm $n_{ite} = 6$. The results are presented on figure (3). The global error η^{glo} tends to an horizontal asymptote which is the NNDD error indicator η^{NNDD}, whereas the convergence of FE error indicator η^{FE} as a function of the number of DoF is shown. The η^{glo} can be numerically related to η^{NNDD} and η^{FE} by relation (25).

$$\left(\eta^{glo}\right)^2 \approx \left(\eta^{FE}\right)^2 + \left(\eta^{NNDD}\right)^2 \tag{25}$$

To evaluate the computed global error, we compute a reference solution denoted by $\mathbf{u}_{ref}$ and we define the reference error e_{ref} and the effectivity index γ by

$$e_{ref} = \left[\sum_{\alpha=1}^{2} \|\mathbf{u}^\alpha_{ref} - \mathbf{u}^\alpha_{h,p}\|^2_{\mathbf{u},\Omega^\alpha}\right]^{1/2} \quad \text{and } \gamma = \frac{\eta^{glo}}{e_{ref}} \tag{26}$$

To obtain a reliable reference solution we choose a mesh size $h_{ref} = 1/8h$ and we set the convergence criteria of the NNDD algorithm to 10^{-8}. The results are reported on figure (4). We first study the evolution of the global error estimator η^{glo}, of the FE error
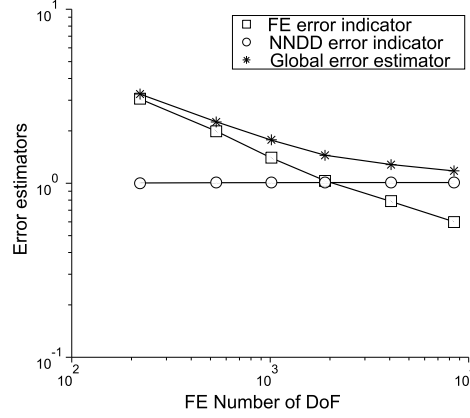
Figure 3: Computed errors as a function of the number of DoF
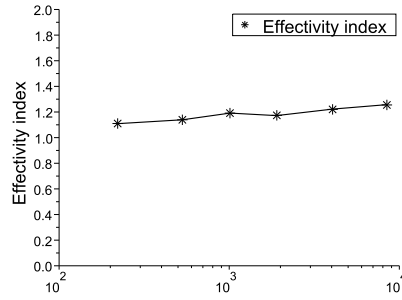
Figure 4: Effectivity index as a function of the number of DoF

indicator η^{FE}, and of the NNDD error indicator η^{NNDD} as a function of the number of n_{ite} the number of iterations of the NDDD algorithm for a fixed number of DoF $n_{DoF} = 1002$. The results are presented on figure (5). The global error η^{glo} tends to an horizontal asymptote which is the FE error indicator η^{FE}, whereas the convergence of NNDD error indicator η^{NDDD} as a function of the number of iterations is shown.

6 CONCLUSION

A global error estimator based on the constitutive relation has been introduced to verify an approximate computation of an unilateral contact problem based on a Finite Element discretization associated with a Neumann-Neumann domain decomposition algorithm. This global error is an upper bound of the exact error and the effectivity index is 1.2 on the studied examples. This error measure takes into account all the errors due to discretization, i.e. both the errors due to the spatial discretization and those due to the domain decomposition algorithm. Two error indicators are developed to estimate the
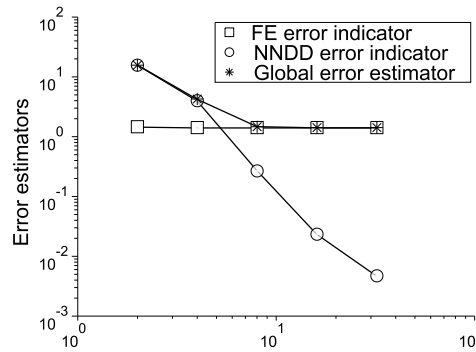
Figure 5: Computed errors as a function of the number iterations of the NNDD algorithm

contributions of each source of error. They are defined in the same way as the error, except that the reference problem is different. On the first tests, these indicators seem to behave well.

REFERENCES

[1] Coorevits, P., Hild, P., Pelle, J.: Posteriori error estimation and indicators for contact problems. Computer Methods Applied to Mechaninal Engineering (2000)

[2] De Saxcé, G.: A generalization of fenchel's inequality and its applications to the constitutive laws. Comptes rendus de l'Académie des sciences. Série II **314**(2), 125–129 (1992)

[3] Gallimard, L.: A constitutive relation error estimator based on traction-free recovery of the equilibrated stress. International Journal of Numerical Engineering (2009)

[4] Gallimard, L., Sassi, T.: A posteriori error analysis of a domain decomposition algorithm for unilateral contact problem. Computers and Structures **88**, 879–888 (2010)

[5] Hasslinger, J., Kucera, R., Sassi, T.: A domain decomposition algorithm for contact problems: Analysis and implementation. Math. Model. Nat. Phenom. **4**(1), 123–146 (2009)

[6] Ladevèze, P., Leguillon, D.: Error estimate procedure in the finite element method and application. SIAM Journal of Numerical Analysis **20**, 485–509 (1983)

[7] L. Gallimard, P. Ladevèze P., J.P. Pelle, Error estimation and time-space parameters optimization for FEM non linear computation, Computer and Structure 64 (1-4) (1997) 145-156.

[8] P. Ladevèze and N. Moes, Adaptive control for finite element analysis in plasticity, Computers and Structures 73 (1999) 45-60.

AIRFOIL OPTIMIZATION WITH TRANSITION CURVE AS OBJECTIVE FUNCTION

PEDRO V. GAMBOA[*] AND MIGUEL A.R. SILVESTRE[†]

Aerospace Sciences Department
Universidade da Beira Interior
6200-254 Covilhã, Portugal
www.ubi.pt
[*] pgamboa@ubi.pt, [†] mars@ubi.pt

Key words: Airfoil, Optimization, Low Reynolds, Aerodynamic Optimization, Transition Curve, Long Endurance UAV.

Abstract. The present work describes the design optimization of a low Reynolds number high lift airfoil where the objective function is that curve defined by the lift coefficient variation with the boundary layer transition position along the upper and lower surfaces of the airfoil. An aerodynamic shape optimization program using XFOIL as the solver, a viscous/inviscid two-dimensional panel method formulation code, and a sequential quadratic programming optimization routine, solves a minimization problem to determine the optimal airfoil geometry which minimizes the difference between its lift coefficient versus transition position curves and the specified objective curves while subject to geometric constraints and constant product of Reynolds number with the square root of lift coefficient for a given interval of lift coefficient values. The airfoil design variables are B-spline control points which define the airfoil camber line and the airfoil thickness distribution. A case study is presented for an airfoil design suitable for a long endurance UAV demonstrating the capability of the approach in producing an optimized design. Comparisons with other objective functions are also shown.

1 INTRODUCTION

In all conventional subsonic aircraft with medium/high aspect ratio wings (>6) the major single contribution to the overall aerodynamic performance of the vehicle comes from the wing airfoil and therefore its careful design is paramount. In the case of the fast growing market of unmanned aerial vehicle (UAV) applications, the need for cost reduction and the miniaturization of the sensors payload is driving the designs to smaller scale and lower airspeeds. This brings the low Reynolds ($60,000<Re<500,000$) airfoil aerodynamic problem, where the boundary layer laminar separation bubble and consequent transition influences decisively the drag coefficient [1]. Most formal approaches in the design of airfoils try to change the airfoil geometry in order to directly minimize the drag coefficient for a given flight condition or angle of attack range or to match a given pressure distribution known to be favourable for a given application. Many researchers have concentrated their efforts on optimizing the turbulent boundary layer pressure recovery strategy to maximize the value and

extension of the low pressure region in the upper surface or to delay the transition further aft thus obtaining higher lift coefficient and/or smaller drag due to extensive laminar flow at moderate/high Reynolds number [2-4]. These concepts lead to laminar separation in airfoils designed in such a way at low Reynolds numbers due to laminar bubbles that protrude massively from the airfoil contour before the transition is triggered and the flow reattaches with a high local velocity, U_e, drop and loss of momentum, leading to an unbearable drag coefficient increase taking place. The concept of surface transition ramp is a result of the later airfoil design philosophy to low Reynolds number when the presence of a strategically placed laminar boundary layer adverse pressure gradient ramp in the airfoil pressure distribution at a given angle of attack can result in a condition of near minimum drag coefficient increase due laminar separation and detached boundary layer transition. The idea is that when the separation does occur, the transition and reattachment follows shortly after. Selig [5] pioneered the application of the work previously done by Eppler [6] where, based in an inverse design by conformal mapping to reach a prescribed inviscid pressure distribution, where segments of constant velocity along the airfoil velocity distribution are prescribed for given angles of attack. Each discrete segment turns into a transition ramp when the angle of attack increases a certain amount above the value prescribed for constant velocity. So, this prescription of angles of attack along the contour, where the higher the prescribed angle of attack the closer the constant velocity segment is from the near leading edge stagnation point in the upper surface and the opposite for the airfoil's lower surface, correlates strongly with the curve of laminar separation/transition position (non-dimensional position along airfoil chord, Xtr/c) versus angle of attack/lift coefficient (the transition curve). In practice, one can realize the correlation of these transition curves with the airfoil's drag polar (see Fig. 1 showing drag polars and transition curves for the Selig's SG604x airfoil series).

One problem in the inviscid inverse design formulation in PROFOIL, as devised by Selig, to manipulate the transition curves [7] is that it does not constitute a direct control of the transition curve position for a given Reynolds number operating condition since the inviscid velocity distribution is not the actual velocity distribution because of the real flow viscosity: it would need some iteration to arrive at a desired objective transition curve. Another issue is that rather than defining the airfoil's velocity distribution with a constant velocity segment it would make more sense to define it with a constant velocity position such that the velocity distribution can be defined in a continuous way rather than in a discrete way.

One can observe in Fig. 1, and in general, that for a given lift coefficient, as long as no significant turbulent separation takes place, the further aft the separation/transition occurs, the smaller the drag coefficient. The desired objective for good airfoil performance seems to be delaying as further aft as possible the transition position for a given design Reynolds number without incurring in significant turbulent separation. So, this furthest aft transition position is imposed by the turbulent pressure recovery method. The Stratford turbulent recovery [8] is the choice that would allow the most aft transition to be implemented at a given angle of attack or lift coefficient. For the same position of transition, in the airfoil's upper and lower surfaces, the drag coefficient will be smaller if the transition curve has a shallower slope along Xtr. The later observation allows the SG6041 airfoil to have peak efficiency at a smaller lift coefficient than the others. But the consequence of using a shallow slope to increase the efficiency in a given design lift coefficient is that the turbulent pressure recovery strategy gets compromised around that design condition when pursuing a high maximum lift coefficients.
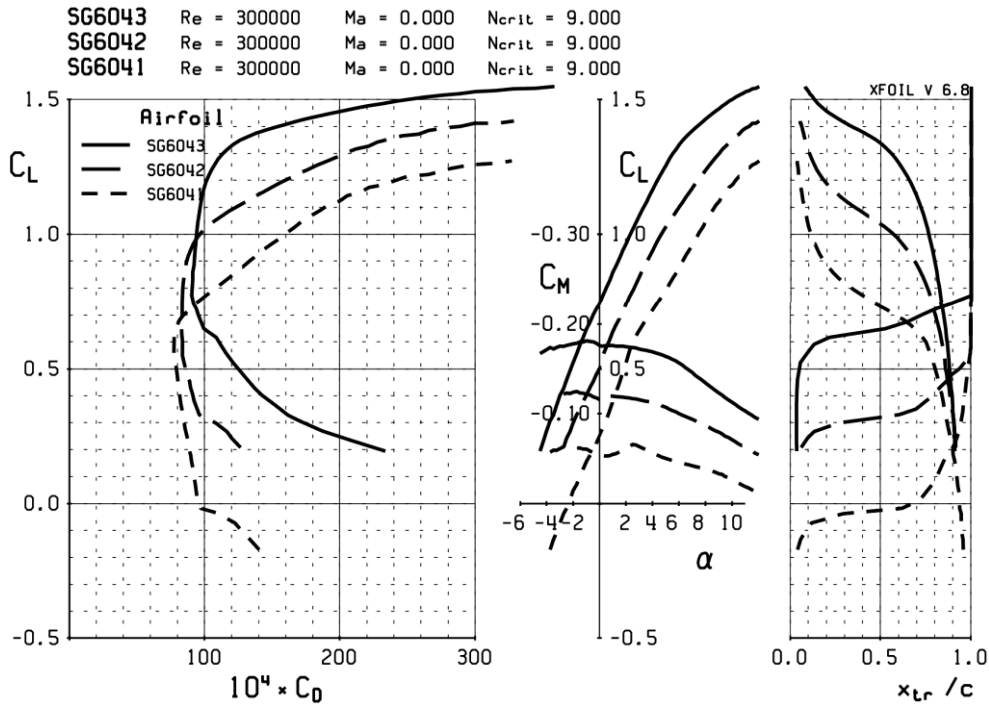
Figure 1: Performance and transition curves predictions of Selig's SG604x series airfoils using XFOIL [5]

Having this in mind, it is observed that the steepest transition curve could produce the highest lift airfoil for a given fixed wing aircraft application (with constant $Re\sqrt{C_l}$). An example application is to further improve the performance of the airfoil most used in high payload fractions design-build-fly competitions, the Selig S1223 airfoil, while using an optimization algorithm coupled to a viscous/inviscid formulation to pursue the desired transition curves.

2 AIRFOIL OPTIMIZATION

Formal numerical airfoil optimization has received increased attention from the scientific and engineering community because its performance is of utmost importance to the overall efficiency of aircraft. Both aerodynamic analysis tools and optimization algorithms have been used to optimize airfoils for specific applications, being the most common the high subsonic speed commercial transport [9-12]. The most utilized optimization algorithms applied to airfoil design range from gradient-based [9,12,13] to stochastic algorithms [10] and adjoint-methods. Some optimization work has also been produced for low *Re* applications [13,14], particularly in the design of UAV airfoils, where gradient-based algorithms are adopted even though the preferred approach still appears to be the inverse design method [15] rather than numerical aerodynamic shape optimization.

Several steps are required in order to solve an airfoil shape gradient based optimization problem: the airfoil must be mathematically defined in such a way that it is possible to change its shape; a method must be implemented to account for the deformation of the airfoil; an aerodynamic solver must be selected to obtain the necessary values to compute the objective

function and constraints; a method must be chosen to compute the gradients, and finally; an optimization algorithm must be used. In the following paragraphs, the different methods selected for each step are described.

An in-house low speed airfoil optimization code [13,16], designed for aerodynamic shape optimization of airfoils subject to operational and geometric constraints, is modified and used in this work.

2.1 Airfoil Geometry Parameterization

Two uniform cubic B-splines are used to discretize the airfoil: one for the thickness distribution and one for the camber line. The airfoil section is obtained by combining the camber line and the thickness. The coordinates of the points on the surface of the airfoil are obtained from the following expressions for all longitudinal x coordinates

$$\begin{cases} z_u(x) = z_c(x) + z_{th}(x) \\ z_l(x) = z_c(x) - z_{th}(x) \end{cases} \tag{1}$$

where (x, z_u) and (x, z_l) are points on the upper and lower surfaces, respectively and z_{th} and z_c are the abscissas of the thickness distribution and the camber line, respectively, for the given x ordinate.

In this optimization the vertical location (z-coordinate) of the control points of the uniform cubic B-splines are used as the design variables. The airfoils are represented using one B-spline with 8 control points for the thickness distribution and another B-spline with 7 control points for the camber line. From the total of 15 control points, 12 are used as design variables. In particular, the control points numbered from 1 to 7 are used in the thickness distribution and those from 10 to 14 are used in the camber line, as shown in Fig. 2(a), are used as design variables. The two control points, representing the thickness distribution, aligned at the $x = 0$, one at the fixed point $(0,0)$ and the other placed in the positive z direction, are used to force the different airfoils to have the same leading edge point. Furthermore, the z position of the moving control point (point 7) at the leading edge is also used as a design variable. This variable is used to control the sharpness of the leading edge during optimization.

In Fig. 2(b) the B-splines are used to represent the Selig S1223 airfoil [1c]. It can be observed that the two B-spline representation accurately defines the Selig S1223 airfoil shape. The leading edge and trailing edge areas are the regions that show some deviation from the original geometry due to the large curvature of the airfoil surface in the region and the small number of spline control points used. The distribution of the B-spline control points along the airfoil chord is chosen in such a way as to give a good representation of the airfoil geometry.

In most cases involving thick airfoils, a denser panelling is used near the leading and trailing edges, where the radius of curvature is smaller and/or the rate of change of the flow state variables is higher. A frequently used method for dividing the chord into panels with larger density near the edges is the full cosine method. In this method, a half-circle is divided into equally spaced angles, $\Delta\beta$, as shown in Fig. 3, and the x coordinate is obtained from

$$x = \frac{c}{2}(1 - \cos\beta) \tag{2}$$

If n chordwise panels are needed, then $\Delta\beta = \pi/n$ and the angle for the panel corner points x_i

is given by

$$\beta_i = (i-1)\Delta\beta \quad \text{for} \quad i = 1, n+1 \tag{3}$$

(a)   (b)
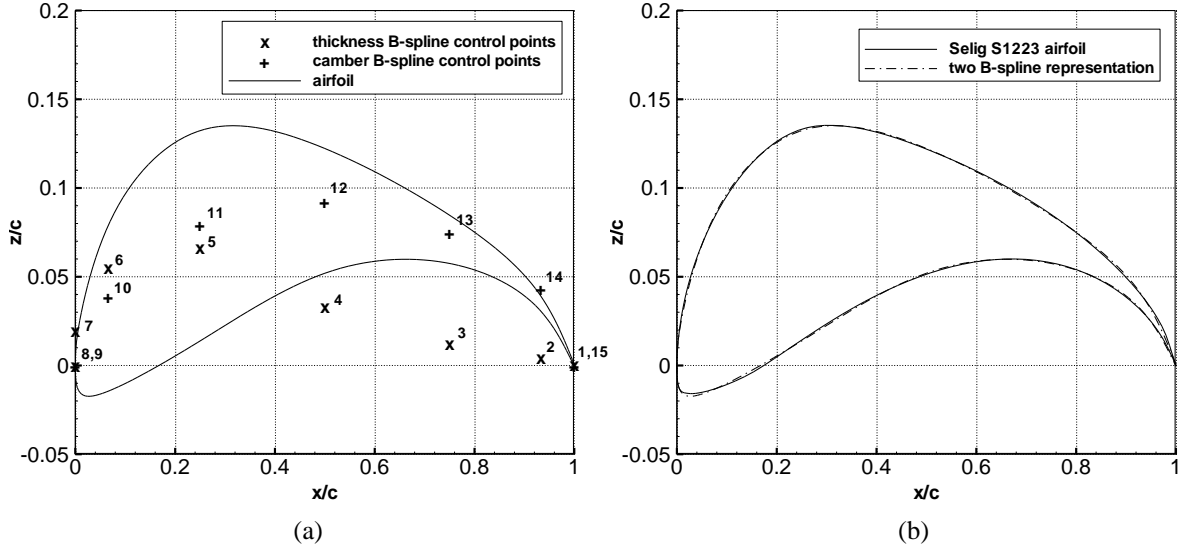
Figure 2: (a) Control points representing the B-splines used for camber line and thickness distribution of the airfoils and corresponding airfoil geometry; (b) Comparison of Selig S1223 airfoil and its B-spline representation
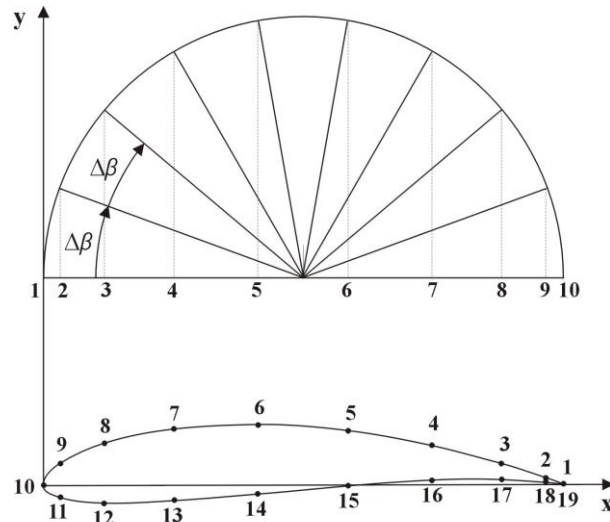
Figure 3: Airfoil surface panel distribution

Given the number of panels required for the airfoil surface, the panel distribution is obtained from Eqs. (2) and (3). Then, knowing the B-spline representations of both the thickness distribution and camber line by having their control points, Eq. (1) can be used to calculate the panel corner points for the upper and lower surfaces of the airfoil.

2.2 Aerodynamic Analysis

The 2-dimensional (2D) aerodynamic coefficients and aerodynamic properties of the airfoil as functions of angle of attack (AOA) and Reynolds number (*Re*) are obtained using the solver of the XFOIL code [17]. In XFOIL, the steady Euler equations in integral form are used to represent the inviscid flow, and a compressible lag-dissipation integral method is used to represent the boundary layers and wake. The entire viscous solution (boundary layers and wake) is strongly interacted with the incompressible potential flow via the surface transpiration model which permits proper calculation of limited separation regions. Results from XFOIL have been compared against experimental data with good agreement [16].

2.3 Optimization Approach

The general optimization problem can be stated as

$$\text{minimize:} \quad f(v) \tag{4}$$

$$\text{subject to:} \quad \begin{aligned} h(v) &= 0 \\ g(v) &\geq 0 \end{aligned} \tag{5}$$

where the design variables, v, may be flight and/or geometric parameters and the equality, $h(v)$, and inequality, $g(v)$, constraints may be lift coefficient and/or geometric parameters, for example.

The aerodynamic shape optimization is carried out with the sequential quadratic programming (SQP) constrained optimization algorithm of FFSQP3.7 [18]. The purpose of the FFSQP3.7 algorithm is the minimization of an (in general nonlinear) differentiable real function subject to (in general nonlinear) inequality and equality constraints. Numerical techniques, such as FFSQP3.7, generally assume that the design space is convex, continuous, and unimodal. Because of this, numerical techniques tend to converge quickly to a local optimum close to the initial design point. Thus, the effectiveness in finding a global optimum is highly dependent on the topology of the design space and the choice of the initial design point. Nonetheless, SQP has been shown to produce good results [19].

The gradients of the objective function and constraints are a requirement of any gradient-based optimization algorithm. In this work, the gradients are computed using forward finite-differences, which enables the problem of finding the gradients to be treated as a black box. Therefore it can be used with any fluid flow solver because it does not involve changes in the solver's code.

3.4 Aerodynamic Shape Optimization

The objective of the airfoil design is to minimize a cost function that produces a good or a set of good airfoil characteristics. In order to achieve this, a tool that searches for the best airfoil geometry is used, which may take into account geometric constraints or performance constraints imposed by the user. Figure 4 shows a flow chart that illustrates the implementation of the aerodynamic shape optimization tool. The code can be summarized as follows:

1. Create the airfoil using the B-spline approach;
2. Compute objective function, $f(v)$, and constraints, $h(v)$ and $g(v)$, of the optimization problem using the aerodynamic solver XFOIL;
3. Compute gradients of objective function and constraints using forward-differences;
4. Solve the optimization problem using the SQP method;
5. If the optimization problem has converged stop; if the optimization has not yet converged continue;
6. Use the new design variables to create new airfoil geometry and go to step 2.
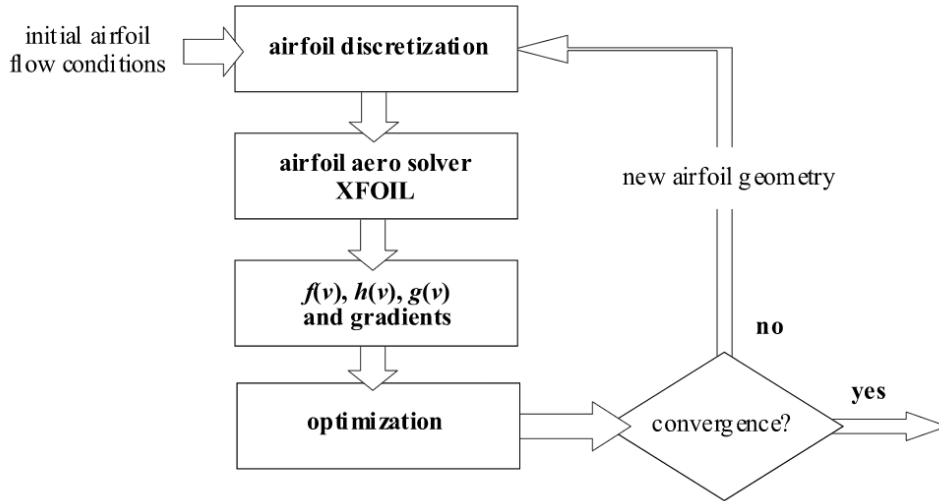
Figure 4: Flow chart of the airfoil aerodynamic shape optimization design tool

3 AIRFOIL DESIGN OPTIZATION CASE

Design-build-fly competitions have become popular within aerospace sciences students. Usually the design goal is maximum payload and/or endurance with some constraining requirements. In this scenario as in general small UAV applications high maximum lift coefficient, C_l/C_d and $C_l^{3/2}/C_d$ values are a significant part of the airfoil design goal along with large relative thickness. From the authors experience one airfoil seems to be the most widely used: Selig's S1223 [4]. An effort to improve this airfoil according to extensive laminar flow region and steep transition curve design philosophy was thus pursued by setting the desired transition curves by the transition positions in the upper surface and lower surface in 5 C_l values within the useful operation envelope of the initial S1223 airfoil for a $Re\sqrt{C_l} = 200,000$, which is a representative value for a typical design-build-fly application (see Fig. 6).

3.1 Problem Definition

Finding the airfoil geometry that gives the desired transition curves, which are essentially the transition position, Xtr, as a function of C_l, on the upper and lower surfaces of the airfoil, is implemented by minimizing the square difference between the desired curves and those produced by the current airfoil geometry.

The optimization problem statement for the study is written as

$$\text{minimize:} \quad f(v) = w \sum_{j=1}^{n} \left(Xtr_{u,j} - Xtrobj_{u,j} \right)^2 + (1-w) \sum_{j=1}^{n} \left(Xtr_{l,j} - Xtrobj_{l,j} \right)^2 \tag{6}$$

$$Re\sqrt{C_{l,j}} = 2 \times 10^5 \quad ; \quad j = 1, n \tag{7}$$

$$\text{subject to:} \quad v_k > 0 \quad ; \quad k = 1,12$$

$$\left(t/c \right)_{TE} \geq 0.01$$

where Xtr_j is the obtained transition position and $Xtrobj_j$ the objective transition position corresponding to $C_{l,j}$. The indices u and l indicate the upper and lower airfoil surfaces, respectively, and the index j denotes the jth analysis C_l point. The parameter w is a weighing factor which, in this study, is taken as 0.5 to give the same importance to the upper surface and lower surface transition curves.

The use of a constant value of $Re\sqrt{C_l}$ is representative of a set of flight conditions where the lift coefficient is adjusted as speed varies so that total lift is maintained unchanged. In this study a total of five (n=5) lift coefficient values ranging from 1.1 to 2.1 are chosen to be representative of the flight envelope required for the airfoil to be designed. From Eq. (7) it follows that the Re for these C_l values range from 138,000 to 190,700. A minimum relative thickness of 1% at the trailing edge, $(t/c)_{TE}$, is set to avoid too thin a trailing edge which is difficult to build and prone to breakage during ground handling.

The initial airfoil selected is the S1223 which is known to have good performance for the given application and is widely used in high lift radio controlled aircraft wings with low speed, heavy payload requirements. The airfoil representation using the two B-spline approach described above is shown in Fig. 5.

3.2 Results

The resulting airfoil from the optimization is shown in Fig. 5 along with the initial S1223 in the two B-spline representation. The final maximum relative thickness is 11.71% compared with the initial 12.04%, a negligible change for practical applications but the corresponding position is significantly shifted aft from 20.6% to 24.5% of the chord. This results in an important improvement for the main spar position which can be placed closer to the center of pressure and leaving more room in case a D leading edge structural configuration is used. The maximum camber position decreased from 8.72% to 8.49% and the corresponding position was displaced slightly back from the initial 48.4% to 50.0%.

The final airfoil transition curves, lift curve and drag polar are displayed in Fig. 6 with solid lines. In the same figure, the curves from the initial airfoil are drawn with dashed lines. The objective transition points versus lift coefficient that defined the objective transition curves are also shown. It is seen that the maximum lift coefficient of 2.18 from the initial airfoil S1223 is not reached. The maximum final airfoil lift coefficient is 2.12, a difference smaller than 3% although the stall seems more abrupt. The reason can be related to the difference between the objective transition point of the upper surface and the actual transition point reached in the final airfoil design. With the current algorithm a transition curve cannot be described on the upper surface for positive dXtr/dC_l values while convergence between the

final airfoil transition curve in the upper surface and the objective is weak even at slightly negative dXtr/dC_l near the maximum lift coefficient. This can be explained by the large sensitivity of the C_l curve to small perturbations in the leading edge geometry and the limited number of design parameters at the leading edge prevent the required geometric resolution from being achieved. On the other hand, the final airfoil drag polar shows a significant improvement below a lift coefficient of 1.75.
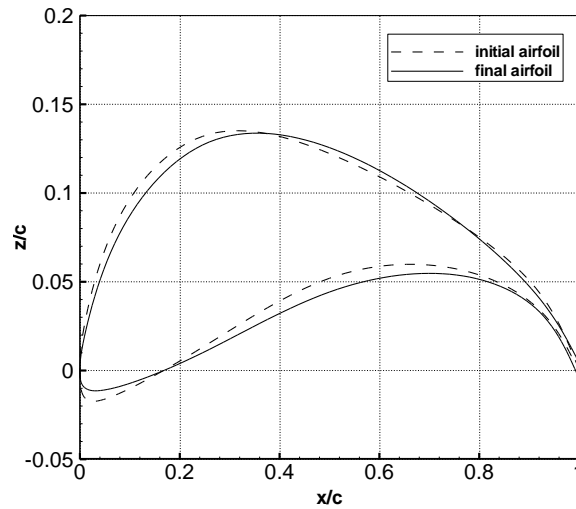
Figure 5: Airfoil optimization results for $Re\sqrt{C_l} = 2 \times 10^5$: initial airfoil (Selig S1223 B-spline representation) and final airfoil geometries
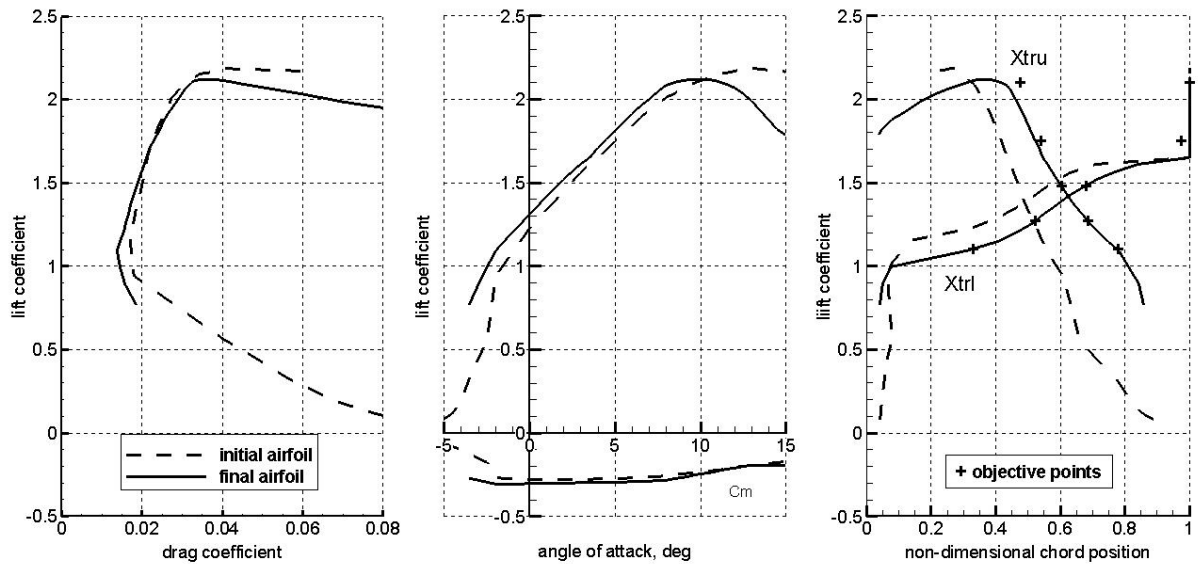
Figure 6: Airfoil optimization results for $Re\sqrt{C_l} = 2 \times 10^5$: drag polar, lift and pitching moment coefficients and transition curves

The improvements in performance can be observed in Fig. 7. The final airfoil aerodynamic efficiency, C_l/C_d, is significantly higher and it extends over a wider C_l envelope. The maximum value of $C_l^{3/2}/C_d$ is not improved but the range of high values is extended to much lower lift coefficients. This is beneficial in actual flight because it is important to have a good margin below the maximum lift coefficient to prevent an unintentional stall.
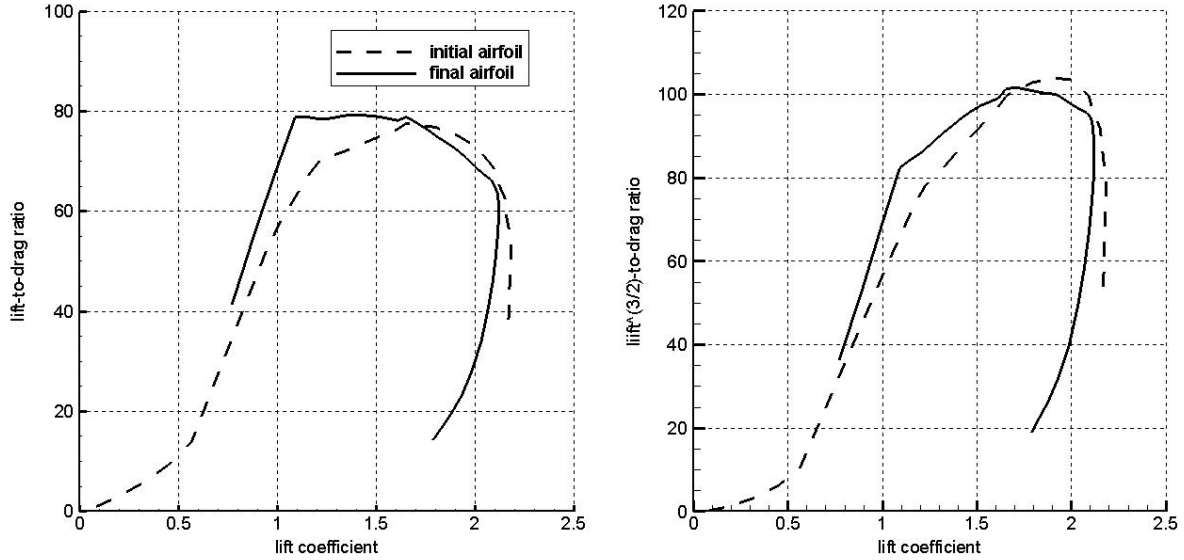
Figure 7: Airfoil optimization results for $Re\sqrt{C_l} = 2 \times 10^5$: lift-to-drag ratio and lift$^{3/2}$-to-drag ratio
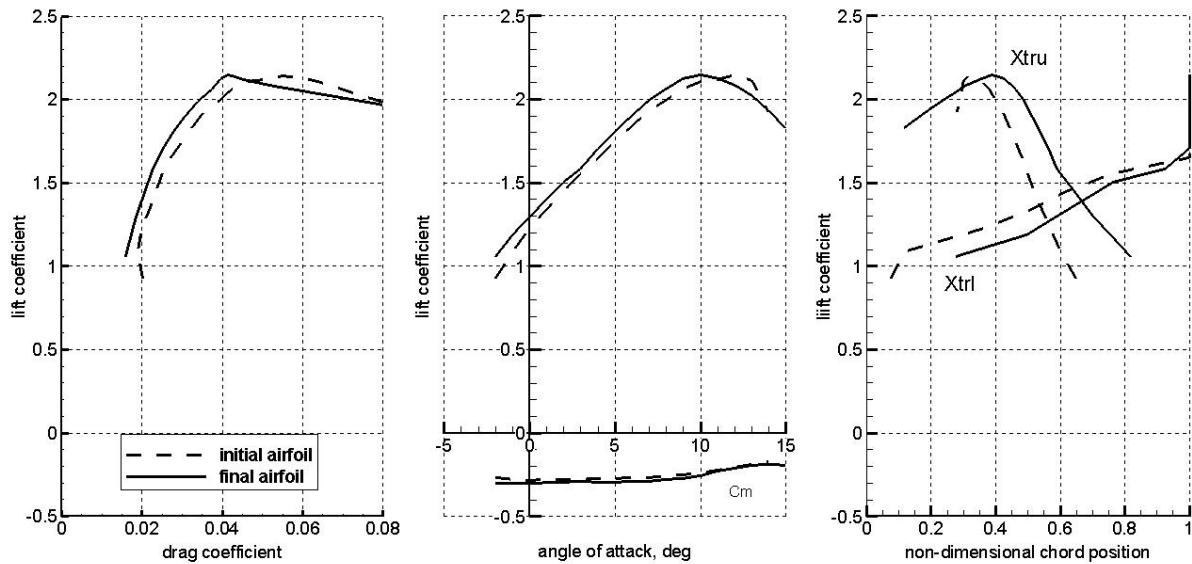
Figure 8: Airfoil results for $Re\sqrt{C_l} = 1.5 \times 10^5$: drag polar, lift and pitching moment coefficients and transition curves

In order to assess the performance of the optimized airfoil at a lower Reynolds number, aerodynamic curves were obtained with XFOIL for $Re\sqrt{C_l} = 1.5 \times 10^5$. Figures 8 and 9 show the results obtained. These clearly indicate that the new design is overall superior to the initial airfoil. One particular aspect is that the stall behavior is smooth, an important requirement for good flight handling qualities at low speed.
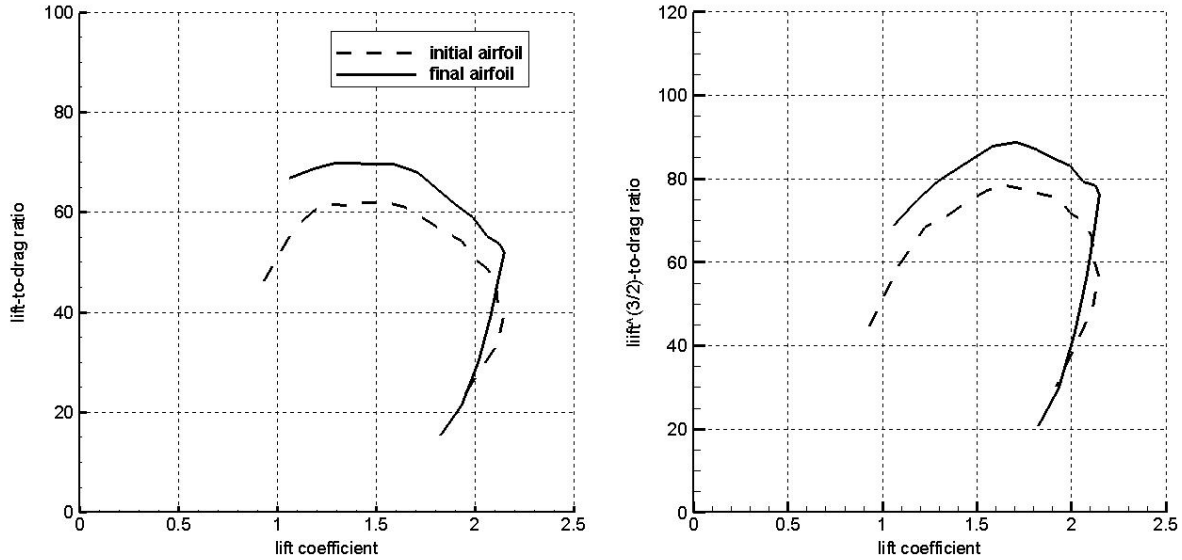
Figure 9: Airfoil results for $Re\sqrt{C_l} = 1.5 \times 10^5$: lift-to-drag ratio and lift$^{3/2}$-to-drag ratio

4 CONCLUSIONS

- Aerodynamic shape optimization using a gradient-based algorithm was performed to design a low Reynolds airfoil to match a set of transition curves. The method implemented produced a good performing airfoil but revealed some convergence difficulties near the higher lift coefficients, which were attributed to the airfoil parameterization in the leading edge region.
- The approach of using the transition curves as objective functions proved useful in producing a good airfoil design.
- This work is a preliminary investigation on this type of airfoil design approach and requires further improvements on the optimization algorithm as well as on the airfoil parameterization scheme.

REFERENCES

[1]   Drela, M. Low Reynolds-Number Airfoil Design for the M.I.T. Daedalus Prototype: A Case Study. *Journal of Aircraft*, Vol. 25, No. 8, (1988), 724–732.

[2]   Liebeck, R., and Ormsbee, A. Optimization of airfoils for maximum lift. CAI/AIAA Subsonic Aero- and Hydro-Dynamics Meeting, AIAA Paper no. 69-739, August 1969.

[3]   Smith, A.M.O. High-Lift Aerodynamics. 37th Wright Bothers Lecture. *Journal of Aircraft*, Vol. 12, No. 6, (1975) 501-530.

[4]   Selig, M.S., and Guglielmo, J.J. High-Lift Low Reynolds Number Airfoil Design. *Journal of Aircraft*, Vol. 34, No. 1, (1997), 72-79.

[5]   Selig, M. Low Reynolds Number Airfoil Design. Von Karman lecture notes, 2003.

[6]   Eppler, R. Direkte Berreehmung von Trogflugelprofilen aus der Druckverteilung. *Ingenieur-Archive*, Vol. 25, No. 1, (1957) 32-57. English translation: NASA TT F-15, 417.

[7]   Selig, M.S., Maughmer, M.D. Multi-Point Inverse Airfoil Design Method Based on Conformal Mapping. *AIAA Journal*, Vol. 30, No. 5, (1992), 1162-1170.

[8]   Stratford, B.S. The Prediction of Separation of the Turbulent Boundary Layer. *Journal of Fluid Mechanics*, Vol. 5, (1959), 1-16.

[9]   Painchaud-Ouellet, S. Tribes, C. and Trépanier, J.Y. Airfoil Shape Optimization Using a Nonuniform Rational B-Splines Parameterization Under Thickness Constraint. Vol. 44, No. 10, (2006), 2170-2178.

[10] Ray, T. and Tsai, H.M. Swarm Algorithm for Single- and Multiobjective Airfoil Design Optimization. *AIAA Journal*, Vol. 42, No. 2, (2004), 366-373.

[11] Croicu, A.M. Hussaini, M.Y. Jameson, A. and Klopfer, G. Robust Airfoil Optimization Using Maximum Expected Value and Expected Maximum Value Approaches. *AIAA Journal*, Vol. 50, No. 9, (2012), 1905-1919.

[12] Carpentieri, G., Tooren, M.J.L, and Koren, B. Arodynamic Shape Optimization by Means of Sequential Linear Programming Techniques. *European Conference on Computational Fluid Dynamics – ECCOMAS CFD 2006*, TU Delft, The Neatherlands, 2006.

[13] Gamboa, P. Vale, J. Lau, F. and Suleman, A. Optimization of a Morphing Wing Based on Coupled Aerodynamic and Structural Constraints. *AIAA Journal*, Vol. 47, No. 9, (2009), 2087-2104.

[14] Nelson, D. Numerical Optimization of Airfoils in Low Reynolds Number Flows. *Journal of Aircraft*, Vol. 46, No. 1, (2009), 331-337.

[15] Olejniczak, J. and Lyrintzis, A.S. Design of Optimized Airfoils in Subcritical Flow. *Journal of Aircraft*, Vol. 31, No. 3, (1994), 680-687.

[16] Gamboa, P. Multidisciplinary Design Optimization of Morphing Aircraft. PhD Thesis, Departamento de Ciências Aeroespaciais, Universidade da Beira Interior, (2007).

[17] Drela, M. XFOIL: An analysis and design system for low Reynolds number airfoils, in: T.J. Mueller (Ed.), Low Reynolds Number Aerodynamics, in: Lecture Notes in Engineering, Springer-Verlag, New York, (1989).

[18] Zhou, J. Tits, A.L. and Lawrence, C.T. *User's Guide for FFSQP Version 3.7*, Electrical Engineering Department and Institute for Systems Research, University of Maryland, United States of America, (1997).

[19] Secanell, M. and Suleman, A. Numerical Evaluation of Optimization Algorithms for Low Reynolds Number Aerodynamic Shape Optimization. *AIAA Journal*, Vol. 43, No.10, (2005), 2262-2267.

STOCHASTIC MODEL REDUCTION APPLIED TO INVERSE ANALYSIS

TOMASZ GARBOWSKI*

*Poznan University of Technology
Institute of Structural Engineering
ul. Piotrowo 5, 60-695, Poznan, Poland
e-mail: tomasz.garbowski@put.poznan.pl

Key words: Model approximation, Gaussian Processes, Inverse Analysis

Abstract. This article describes the use of Gaussian Processes in model reduction techniques with application to inverse problems. The work is focused on the proper construction of the model approximation, namely on training process based on limited number of learning samples. By making use of Active Learning criterion employed in training process a significant improvement in model prediction can be observed. An example of application of stochastic surrogate model for the paperboard characterization through biaxial tensile test and Digital Image Correlation measurements is also presented.

1 INTRODUCTION

The inverse analysis is a technique widely used for structure or material characterization, especially when unknown or uncertain parameters embedded in the model have to be determined. Knowing the structure responses such as displacements, accelerations, eigenmodes, etc. the backward computations can be performed to identify causes, e.g. constitutive constants, structural parameters. In the literature there are many examples of application of inverse analysis for material characterization [1, 2, 3], damage detection [4, 5], estimation of residual stresses [6, 7], to list just a few.

The inverse analysis often uses a numerical model (e.g. finite element or boundary element model, analytical model) for test simulation, which computes selected quantities to be compared to experientially measured ones. The inverse procedure, through iterative optimization algorithms, minimizes the discrepancy between experimentally measured $\mathbf{U}_{\text{EXP}}$ and numerically computed $\mathbf{U}_{\text{NUM}}$ quantities, so the minimization problem reads:

$$\arg \min_{\mathbf{x} \in \mathbb{R}^n} \|\mathbf{U}_{\text{NUM}}(\mathbf{x}) - \mathbf{U}_{\text{EXP}}\|_2^2 \tag{1}$$

where $\mathbf{x}$ is a set of sought parameters.

If the numerical model is complex and/or has to be computed many times, the iterative minimization procedure becomes very expensive, therefore, not attractive from practical point of view especially when the test has to be performed routinely 'in situ' (i.e. without a computer which can handle heavy computations). The alternative is to use a surrogate which approximates the behavior of the numerical model but is much simpler, thus less expensive. The surrogate is usually constructed as a 'black box' where for the approximation the following methods, among others, are commonly used: Radial Basis Functions (RBFs), Polynomials, Proper Orthogonal Decomposition (POD) combined with RBFs [8, 9], Artificial Neural Networks (ANNs) [10] or Gaussian Processes (GP) [11, 12].

2 MODEL REDUCTION THROUGH GAUSSIAN PROCESSES

2.1 Model approximation

In order to build a smooth and accurate analytical approximation one needs to construct a forward model to generate responses (i.e. training samples):

$$\mathbf{x}_n \Rightarrow \begin{Bmatrix} \text{Forward} \\ \text{Model} \end{Bmatrix} \Rightarrow \mathbf{u}(\mathbf{x}_n) \tag{2}$$

where $\mathbf{x}_n = [x_1, x_2, \ldots, x_M]_n^T$ is an input vector (i.e. vector of sought parameters) and $\mathbf{u}(\mathbf{x}_n) = \mathbf{u}_n = [u_1(\mathbf{x}_n), u_2(\mathbf{x}_n), \ldots, u_k(\mathbf{x}_n)]^T$ is an output vector (i.e. vector of measurable quantities) for $n = 1, \ldots, N$. Both vectors $\{\mathbf{x}_n, \mathbf{u}_n\}$ represents a single training sample and form n-th column of $(M \times N)$ parameter matrix $\mathbf{X}$ and $(K \times N)$ snapshot matrix $\mathbf{U}$:

$$\mathbf{X} = \begin{bmatrix} x_{11} & \cdots & x_{1N} \\ \vdots & \ddots & \vdots \\ x_{M1} & \cdots & x_{MN} \end{bmatrix}, \qquad \mathbf{U} = \begin{bmatrix} u_{11} & \cdots & u_{1N} \\ \vdots & \ddots & \vdots \\ u_{K1} & \cdots & u_{KN} \end{bmatrix}, \tag{3}$$

Ideally would be to find method which needs the smallest possible number of 'training samples' and in the same time is precise and robust. The approximation method based on Gaussian Processes satisfies all these requirements: it gives very good results when the number of training examples is limited. Another important feature of GP is that it gives not only the approximation of the mean value of sought solution but also its standard deviation. This feature gives a possibility of automatic and systematic improvement of the solution, because in the locations with high values of standard deviation one can expect a weak approximation and therefore, it points out where, in the parameter space, an additional experimental or numerical data is necessary to improve the solution.

The GP employed for stochastic approximation is usually formulated within Bayesian framework, thus provides additional information about the magnitude of correlation between state variables and control variables. It is very important to know the relevance of input-output correlation because based on it one can exclude from the model the parameters which do not influence the measurable quantities. The reduction saves the

experimental efforts of finding parameters which appear to be irrelevant in particular simulation. Such information obtained 'for free' during the training process is similar to the results from sensitivity analysis often performed for model check.

The stochastic model reduction techniques based on GP have, however, one significant disadvantage, namely the Gaussian Processes are usually parameterized in terms of their covariance functions. This makes it difficult to deal with multiple outputs, because ensuring that the covariance matrix is positive definite is problematic. An alternative formulation is to treat Gaussian processes as white noise sources convolved with smoothing kernels, and to parameterize the kernel instead (see [13]). Using this approach, one can extend Gaussian Processes to handle multiple, coupled outputs.

When GP are applied for model reduction (here for the forward model approximation) within the inverse problem, the number of measurable quantities (e.g. measured displacement fields, velocities or accelerations in different moments of time and in various space locations) is often very large. In such situation the probable correlations between the state variables can be computed, and consequently used to reduce the number of model outputs by the application of POD, i.e.:

$$\mathbf{U} \Rightarrow \{\mathrm{POD}\} \Rightarrow \bar{\mathbf{U}} \tag{4}$$

where

$$\bar{\mathbf{U}} = \begin{bmatrix} u_{11} & \cdots & u_{1N} \\ \vdots & \ddots & \vdots \\ u_{L1} & \cdots & u_{LN} \end{bmatrix}, \tag{5}$$

with $L \ll K$. However, the problem size, even when reduced, does not necessarily make possible an efficient application of multi-output GP. An alternative is to formulate the GP approximation on modified training samples, which instead of output consisting of truncated measurable quantities $\{\mathbf{x}_n, \bar{\mathbf{u}}_n\}$, have an output containing a particular scalars $\{\mathbf{x}_n, t_n\}$. The scalar value output can be chosen for example as a 'squared distance' from the experimental measurements to their numerically computed counterparts, namely:

$$t_n = \|\mathbf{u}_n - \mathbf{u}_{\mathrm{EXP}}\|_2^2. \tag{6}$$

By adopting a latter method the GP approximation can be constructed in two stages. First N training samples $\{\mathbf{x}_n, \mathbf{u}_n\}$ are computed through forward model (2). Later for each particular identification procedure an experimental data $\mathbf{u}_{\mathrm{EXP}}$ are used for output truncation through (6) to a single scalar (so the training data is now $\{\mathbf{X}, \mathbf{T}\}$, where $\mathbf{T} = [t_1, t_2, \ldots, t_N]$). Having a scalar output in each training sample the single-output GP approximation can be easily constructed. On this stage an additional retraining by an active learning method (see e.g. [11]) can be performed in order to improve the surrogate. Once the GP approximation is designed, any optimization algorithm can be used to find a function minimum according to formula (1). Because the surrogate in general approximates a multi-modal function (i.e. function with many local minima)

it seems reasonable to employ for function minimization some global search techniques such as genetic algorithms [14] or particle swarm methods [15]. Such algorithms require usually a big number of iterations in order to find a global minimum, however, when used on reduced model the computing time is not an issue anymore.

2.2 Linear regression model

In order to explain how to construct a model approximation by Gaussian Processes, first a Linear Regression (LR) model should be considered. LR is a linear function of model parameters $\mathbf{w}$ and nonlinear function of the input vector $\mathbf{x}$, and usually is defined as:

$$y\left(\mathbf{x}, \mathbf{w}\right) = \sum_{j=1}^{M} w_j \phi_j\left(\mathbf{x}\right),$$ (7)

which simply is a linear combination of fixed, nonlinear basis functions $\phi_j(\mathbf{x})$ of the input variables (e.g. polynomial basis functions).

If we now take N given training patterns $(\mathbf{x}_n, t_n)$, $\mathbf{x}_n$ being the input vector, t_n the response for $n = 1 \ldots N$, then the parameters $\mathbf{w}$ of the linear model can be computed by, for example, penalized least squares method:

$$\mathbf{w} = \left(\mathbf{\Phi}^T \mathbf{\Phi} + \lambda \mathbf{I}\right)^{-1} \mathbf{\Phi}^T \mathbf{t},$$ (8)

where $\mathbf{\Phi}$ is $N{\times}M$ design matrix with elements defined as $\phi_m(\mathbf{x}_n)$. The regularization parameter λ is called hyper-parameter and can be estimated using validation set or by applying Bayesian inference and maximizing evidence of dataset $p(\mathbf{t}|\lambda)$ w.r.t. λ (details are given in [12]).

2.3 Gaussian Process

Gaussian process model can be obtained by reformulation of the linear model in terms of dual representation. In this approach, linear model is trained by minimizing a regularized error, which is defined using $N{\times}N$ symmetric Gram matrix:

$$\mathbf{K} = \mathbf{\Phi}\mathbf{\Phi}^T = \phi(\mathbf{x})^T \phi(\mathbf{x}') = k(\mathbf{x}, \mathbf{x}'),$$ (9)

where $k(\mathbf{x}, \mathbf{x}')$ is a kernel function. The vector $\mathbf{k}_n = k(\mathbf{x}_n, \mathbf{x})$ represents n-th row or column of $\mathbf{K}$ matrix.

The prediction for a new input $\mathbf{x}^*$ can be computed by the formula:

$$\mathcal{GP}\left(\mathbf{x}^*|\mathbf{x}, \mathbf{t}, \lambda\right) = k(\mathbf{x}, \mathbf{x}^*)^T (\mathbf{K} + \lambda \mathbf{I})^{-1} \mathbf{t},$$ (10)

where $k(\mathbf{x}, \mathbf{x}^*)$ is a covariance between a new input $\mathbf{x}^*$ and the other inputs, $\mathbf{t} = (t_1 \ldots t_N)^T$ is a vector of training target values.

From the Bayesian point of view the dual representation of linear model leads to the Gaussian process, where the kernel function is interpreted as a covariance function of the

GP. Application of such regression model for prediction allows to compute the predictive distribution of the target variable $y(\mathbf{x}^*)$ for a new input vector $\mathbf{x}^*$. This requires evaluation of conditional distribution $p(y|\mathbf{t})$, which for the Gaussian processes is a Gaussian distribution with mean and covariance respectively given by:

$$\text{mean}(\mathbf{x}^*) = \mathbf{k}^T \mathbf{C}^{-1} \mathbf{t}, \tag{11}$$

$$\sigma^2(\mathbf{x}^*) = c - \mathbf{k}^T \mathbf{C}^{-1} \mathbf{k}, \tag{12}$$

where $\mathbf{C}$ is the $N \times N$ covariance matrix given by:

$$C(\mathbf{x}, \mathbf{x}') = k(\mathbf{x}, \mathbf{x}') + \beta^{-1} \mathbf{I}, \tag{13}$$

where β is the variance of the target distribution and $\mathbf{I}$ is an identity matrix. The covariance function $C(\mathbf{x}, \mathbf{x}')$ defines the property that vectors $\mathbf{x}$ and $\mathbf{x}'$, which are close in input space, should give rise to highly correlated outputs $y(\mathbf{x})$ and $y(\mathbf{x}')$.

2.4 Covariance function

The covariance function can be any function that will generate a non-negative definite covariance matrix for any ordered set of (input) vectors $(\mathbf{x}_1, \ldots \mathbf{x}_N)$. A stationary, non-isotropic squared exponential covariance function $k(\mathbf{x}, \mathbf{x}')$ is chosen here, and given by:

$$k(\mathbf{x}, \mathbf{x}') = \nu \exp\left(-\frac{1}{2} \sum_{i=1}^{M} w_i (x_i - x_i')^2 \right) + b, \tag{14}$$

where the term b represents a bias that controls the vertical offset of the Gaussian process, while ν controls the vertical scale of the process. The w_i parameters allow a different distance measure for each dimension. If w_i is small then the i-th input is down-weighted and have little effect on the input. These hyper-parameters play vary important role mainly because they have a direct link to model sensitivities with respect to input parameters thus provide a measure of importance of input parameters.

After defining the covariance function we can make predictions of the new input vectors but first it is necessary to learn the hyper-parameters

$$\boldsymbol{\theta} = [\nu, \omega_1, \ldots, \omega_M, b, \beta]. \tag{15}$$

In order to find those parameters one can search for the most probable set by maximizing the log likelihood function given by:

$$\ln p(\mathbf{t}|\boldsymbol{\theta}) = \frac{1}{2} \ln |\mathbf{C}| - \frac{1}{2} \mathbf{t} \mathbf{C}^{-1} \mathbf{t} - \frac{N}{2} \ln 2\pi, \tag{16}$$

using gradient-based optimization algorithms, such as a first-order batch Levenberg-Marquardt Algorithm (LMA) or Trust Region Algorithm (TRA), which provides fast convergence (see details in [16]).

2.5 Application of GP to inverse problems

Once the N training samples $\{\mathbf{X}, \mathbf{U}\}$ is generated through forward model (2) and later adjusted to the particular experimental data $\mathbf{u}_{\text{EXP}}$ by the formula (6) so the training set becomes $\{\mathbf{X}, \mathbf{T}\}$ one can learn hyper-parameters $\boldsymbol{\theta}$ of covariance matrix $C(\mathbf{x}, \mathbf{x}')$, thus construct GP approximation $\mathcal{GP}(\mathbf{X}, \mathbf{T}, \boldsymbol{\theta})$ of forward model.

From this point, by making use of equations (11) and (12), the prediction of GP output t^* for a new parameter vector $\mathbf{x}^*$ can be computed through conditional distribution

$$\mathcal{GP}(\mathbf{x}^*|\mathbf{X}, \mathbf{T}, \boldsymbol{\theta}) = \left\{ \text{mean}(t^*) = \mathbf{k}^T\mathbf{C}^{-1}\mathbf{T}, \quad \text{covariance}(t^*) = c - \mathbf{k}^T\mathbf{C}^{-1}\mathbf{k} \right\}. \qquad (17)$$

The mean value and covariance of model prediction are computed by simple matrix multiplications, however, if number of sampling points is large, the construction and inversion of covariance matrix have to be carefully designed in order to retain an efficiency.

2.6 Active learning

If one would like to improve an approximation adding new training samples, the specific training process, which automatically finds new locations of sampling points has to be employed. In order to optimize the retraining procedure of GP based surrogate model, the active learning criterion, that improve the global model fit, can be implemented. A new training points are sequentially added in the zones where the model predictions are poor and/or in the vicinity of the minimum of the approximated solution, meaning that the iterative retraining of the surrogate model is performed by adding a new training patterns, sampled in the new locations in the parameter space (selected by the algorithm itself).
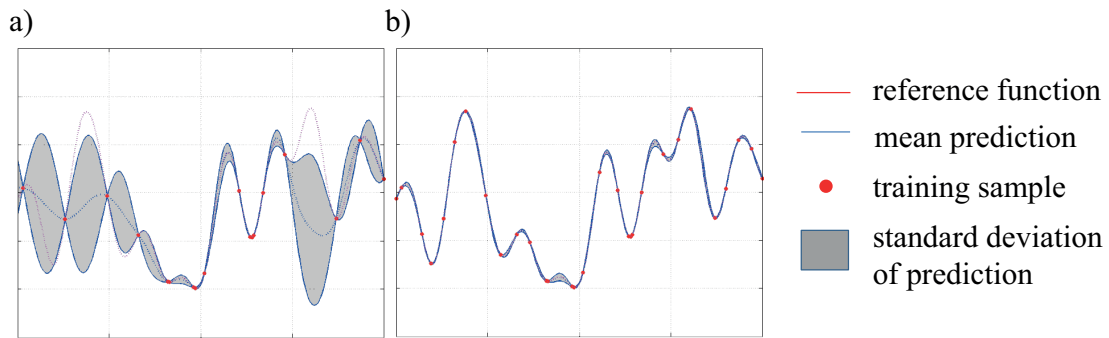
Figure 1: An example of 1D function approximation by GP and active learning algorithm: (a) 15 samples approximation, (b) 28 samples approximation

This approach is very efficient when the number of training samples is limited, so the algorithm starts building approximation with small selection of patterns (randomly or uniformly distributed) and improves the approximation by sampling the parameter space in a clever way (based on its confidence about the quality of the approximation). It stops

when the selected number of retraining samples is reached. In Fig. 1 the active training process is shown.

3   EXAMPLE

An example of a model characterization is used here to illustrate the application of above described model reduction techniques. The examples show the use of GP as numerical model's surrogate for characterization of paperboard parameters [17] through biaxial test and DIC measurements techniques combined with inverse analysis. From such test one can identify most of the in-plane parameters of paperboard (for the details see [10]).
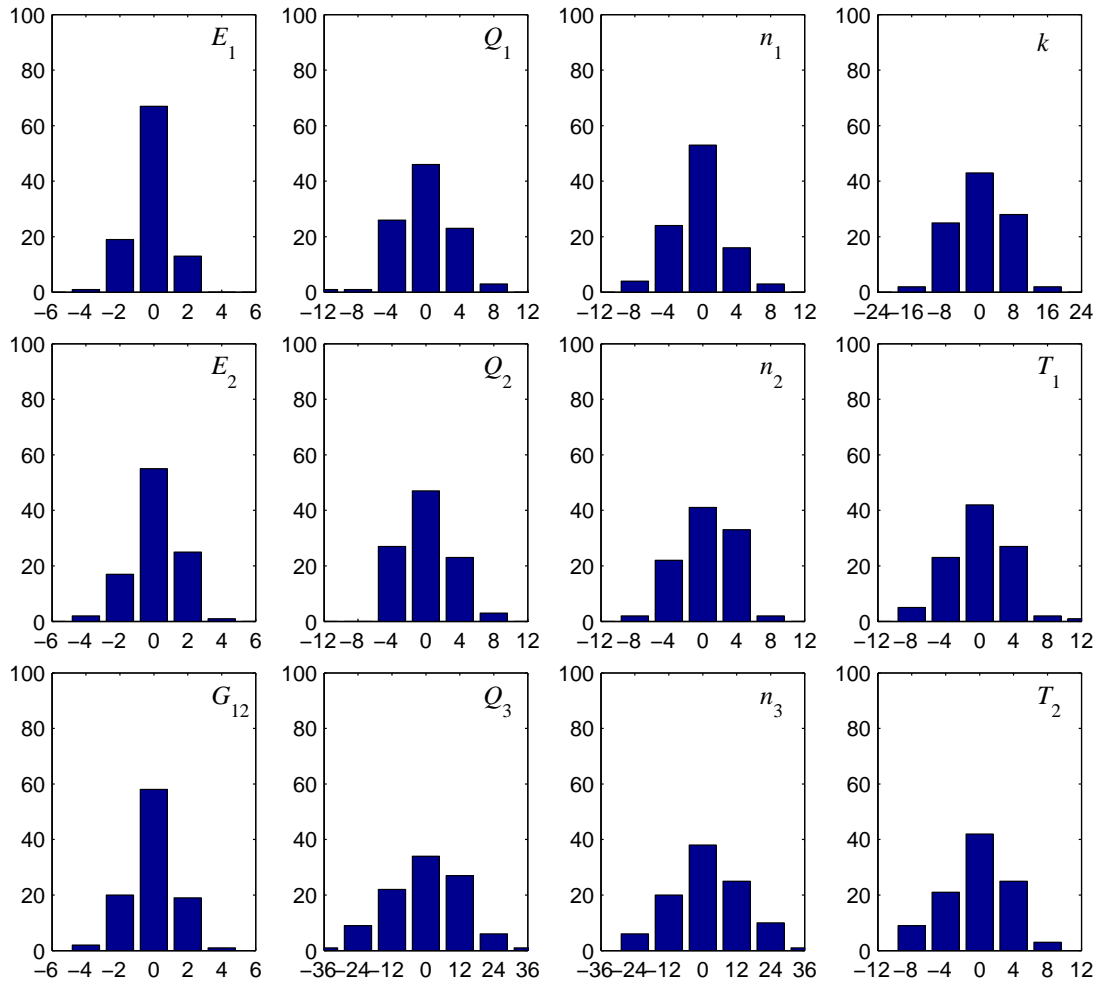
Figure 2: Performance of the GP designed for identification of 12 parameters in Xia *et al.* model with measurement noise ±0.5 µm: percentage of relative error in abscissae; in ordinates, percentages of results within each abscissae interval

Figure 2 shows the performance of GP trained and later retrained by Active Learning

algorithm for each (over one hundred) pseudo-experimental testing sample.

In the above examples the GP based approximation serves as a surrogate of complete Finite Element (FE) model, which in combination with iterative minimization routine, namely trust-region algorithm (TRA) and evolutionary-based techniques, i.e. particle swarm algorithm (PSA) gives very accurate results of model parameters identification. The results are compared to those obtained by making use of Artificial Neural Networks (details on application of ANNs are given in [10]). Both ANNs and GP approximations are constructed on 10.000 numerically generated samples additionally noised and truncated to DIC measurements precision. The sampling points are distributed in the 12-dimensional parameter space according to Optimal Latin Hypercube technique [18].

The GP model is further retrained by making use of 100 new testing data. Each testing sample is used to truncate the output matrix $\mathbf{U}$ through formula (6) to the vector $\mathbf{T}$. Active Learning criterion inserted in training algorithm automatically adds new training points (totally 1.000) in the argument space according to rules specified in previous section. In Tab. 1 the comparison of ANNs performance with GP trained by TRA and GP trained by PSA with active learning criterion is presented.

Table 1: Relative approximation error on 100 testing samples by (a) Artificial Neural Networks; (b) Gaussian Processes and Trust Region Algorithm and (c) GP combined with Active Learning and Particle Swarm Algorithm

Parameter	ANN	GP (TRA)	GP+AL (PSA)
E_1	1.29	1.49	1.11
E_2	1.55	1.61	1.25
G_{12}	1.66	1.81	1.15
Q_1	4.22	4.10	2.63
Q_2	4.05	3.92	2.89
Q_3	14.3	15.1	12.0
n_1	3.74	3.70	2.16
n_2	4.66	4.32	2.62
n_3	15.9	14.7	9.86
k	7.28	7.41	5.38
T_1	3.20	4.04	2.40
T_2	3.50	4.07	2.23

4   SUMMARY

The model reduction approach has the following important advantages w.r.t. the procedure based on evaluation of full numerical model: (a) it is much faster (the computational burden is moved to training phase) and (b) it does not require to use powerful computers for heavy and repetitive computations of numerical model, so once the model

is constructed it can be used 'in situ' on the portable computer.

The GP approximation model, which serves as a numerical model reduction, was used here in combination with inverse analysis to solve practical engineering identification problems. In the work the proper construction of the GP model was discussed, namely: (a) training process based on minimal number of training data, by making use of automatic samples selection through computed standard deviation of model prediction; (b) control parameters reduction based on input-output correlation (sensitivity-like analysis) and (c) state variables 'compaction' for single-output GP implementation. The successful application of stochastic model reduction techniques for the material model characterization problems was also presented and compared to other approximation method based on Artificial Neural Networks.

REFERENCES

[1] Bolzon, G., Maier, G. and Panico, M. Material model calibration by indentation, imprint mapping and inverse analysis. *International Journal of Solids and Structures*, (2004) **41**: 2957–2975.

[2] Ardito, R., Maier, G. and Massalongo, G. Diagnostic analysis of concrete dams based on seasonal hydrostatic loading. *Engineering Structures*, (2008) **30**: 3176–3185

[3] Maier, G., Buljak, V., Garbowski, T., Cocchetti, G. and Novati G. Mechanical characterization of materials and diagnosis of structures by inverse analyses: some innovative procedures and applications. *International Journal of Computational Methods* (2013) submitted.

[4] Knitter-Piatkowska, A., Pozorski, Z. and Garstecki, A. Application of discrete wavelet transformation in damage detection. Part I: Static and dynamic experiments. *Computers Assisted Mechanics and Engineering Sciences* (2006) **13**:21–38.

[5] Ziopaja, K., Pozorski, Z. and Garstecki, A. Damage detection using thermal experiments and wavelet transformation. *Inverse Problems in Science and Engineering* (2011) **19, 1**:127–153.

[6] Bocciarelli, M. and Maier, G. Indentation and imprint mapping method for identification of residual stresses. *Computational Materials Science*, (2007) **39**:381–392.

[7] Buljak, V. and Maier, G. Identification of residual stresses by instrumented elliptical indentation and inverse analysis. *Mechanics Research Communications*, (2012) **41**: 21–29

[8] Maier, G., Bolzon, G., Buljak, V., Garbowski, T. and Miller, B. *Chapter 24. Synergistic combinations of computational methods and experiments for structural diagnosis.* Computer Methods in Mechanics. Lectures of the CMM 2009, Eds. M. Kuczma, K. Wilmanski, Springer-Varlag Berlin Heidelberg, (2010) 453–476.

[9] Garbowski, T., Maier, G. and Novati, G. Diagnosis of concrete dams by flat-jack tests and inverse analyses based on proper orthogonal decomposition. *Journal of Mechanics of Materials and Structures*, (2011) **6, 1-4**:181–202.

[10] Garbowski, T. , Maier, G. and Novati, G. On calibration of orthotropic elastic-plastic constitutive models for paper foils by biaxial tests and inverse analyses. *Structural and Multidisciplinary Optimization*, (2012) **46, 1**:111–128.

[11] C.E. Rasmussen, C.K.I. Williams. *Gaussian Processes for Machine Learning.* MIT Pres, (2006).

[12] Bishop, C.M. *Pattern Recognition and Machine Learning.* Springer, (2007).

[13] Alvarez, M., Lawrence, N. Sparse Convolved Gaussian Processes for Multi-output Regression. *Twenty-Second Annual Conference on Neural Information Processing Systems (NIPS), Vancouver, Canada*, (2008).

[14] Burczynski, T., Kus, W., Dlugosz, A. and Orantek, P. Optimization and defect identification using distributed evolutionary algorithms *Engineering Applications of Artificial Intelligence*, (2004) **17, 4**: 337–344

[15] Kennedy, J. Particle swarm optimization. *Proceedings., IEEE International Conference on Neural Networks*, (1995) **4**: 1942–1948

[16] Nocedal, J., Wright, S. *Numerical Optimization.* Springer, (2006).

[17] Xia, Q.S., Boyce, M.C., Parks, D.M. A constitutive model for the anisotropic elastic-plastic deformation of paper and paperboard. *International Journal of Solids and Structures*, (2000) **61**:1–48.

[18] Park, J.-S. Optimal Latin-hypercube designs for computer experiments, *Journal of Statistical Planning and Inference*, (1994) **39, 1**: 95–111

HIGH-ORDER MESH GENERATION ON CAD GEOMETRIES

A. GARGALLO-PEIRÓ*, X. ROCA†, J. PERAIRE† AND J. SARRATE*

*Laboratori de Càlcul Numèric (LaCàN)
Departament de Matemàtica Aplicada III
Universitat Politècnica de Catalunya
Jordi Girona 1, E-08034 Barcelona, Spain
e-mail: abel.gargallo@upc.edu, jose.sarrate@upc.edu

†Aeronautics and Astronautics
Massachusetts Institute of Technology
Cambridge, MA 02139, USA
e-mail: xeviroca@mit.edu, jperaire@mit.edu

Key words: high-order quality; high-order mesh generation; mesh optimization; curved elements; parameterized surfaces;

Abstract. We present a technique to extend Jacobian-based distortion (quality) measures for planar triangles to high-order isoparametric elements of any interpolation degree on CAD parameterized surfaces. The resulting distortion (quality) measures are expressed in terms of the parametric coordinates of the nodes. These extended distortion (quality) measures can be used to check the quality and validity of a high-order surface mesh. We also apply them to simultaneously smooth and untangle high-order surface meshes by minimizing the extended distortion measure. The minimization is performed in terms of the parametric coordinates of the nodes. Thus, the nodes always lie on the surface. Finally, we include several examples to illustrate the application of the proposed technique.

1 Introduction

It is well known that computational methods for solving partial differential equations require domain discretizations composed by valid and high-quality elements [1, 2, 3]. If the mesh contains inverted elements, it can not be used for computational purposes. Moreover, if the mesh does not have a minimum quality, the accuracy of the finite element computation is degraded.

In order to improve the quality of a mesh the nodes can be relocated (smoothing) [4, 5, 6]. Note that in 3D applications, it is of the major importance to ensure a high-quality surface mesh. If a boundary mesh face is inverted, the corresponding mesh element

is inverted and cannot be recovered once the surface mesh is fixed. Therefore, in this work we present a technique to extend any Jacobian-based distortion (quality) for planar elements to high-order elements with the nodes on CAD surfaces. The resulting measures are expressed in terms of the parametric coordinates of the surface. We use these measures in order to develop a simultaneous smoothing and untangling technique for high-order meshes with the nodes on a parameterized surface. The resulting meshes are composed by valid and high-quality elements with the nodes on the surface. It is important to highlight that we can ensure that the optimized nodes lie on the original CAD surface and not on an approximation, since the optimization process is written in terms of the parametric coordinates of the mesh nodes.

The proposed technique relies on the framework of algebraic quality measures introduced in [2]. In order to improve the quality of a valid mesh, an optimization approach based on Jacobian-based measures is proposed in [6]. These optimization approaches can also be used to untangle inverted elements. On the one hand, references [7, 8] propose a two-step procedures that first untangle the elements and second smooth the node location. On the other hand, in Reference [9] a simultaneous smoothing and untangling technique for triangular planar meshes is proposed by means of a modification of a Jacobian-based distortion measure. It is worth to notice that this technique has been extended to quadrilateral and hexahedral meshes [10] and to non-planar triangular meshes [11]. The simulatneous smoothing-untangling is the approach selected in this work.

Several techniques have been developed to optimize meshes on surfaces, generally defined by discrete representations, see [11, 12, 13, 14, 15]. However, in our work we consider parameterized CAD geometries and our objective is to ensure that during the optimization process the nodes are always located on the surface. In [16] we already proposed to quantify the distortion (quality) of a linear surface element in terms of the coordinates on the parametric space of the CAD surface. An optimization approach based on the proposed distortion measure ensures that the nodes always lie on the surface, since the whole process is developed in the parametric space of the original surface.

Several methods have been proposed to generate high-order planar or 3D meshes, see [17, 18, 19, 20, 21]. The standard approach to generate a high-order mesh consists on an a-posteriori procedure composed by three steps: (1) generate a linear mesh; (2) increase the order of the elements and curve them to fit the boundary; and (3) optimize the node locations so that the mesh is valid and is composed by high-quality elements. The method proposed in this paper relies on the work developed for planar high-order elements presented in [18]. Specifically, we propose to extend the measures for planar high-order elements presented in [18] to high-order meshes on parametrized surfaces using the framework presented in [16], where planar measures for linear elements are extended to surfaces.

The outline of the paper is as follows. First, in Section 2, we review the definition of distortion measure for planar elements presented in [18]. Next in Section 3, we present the formulation to extend any Jacobian distortion measure for linear triangles to high-order
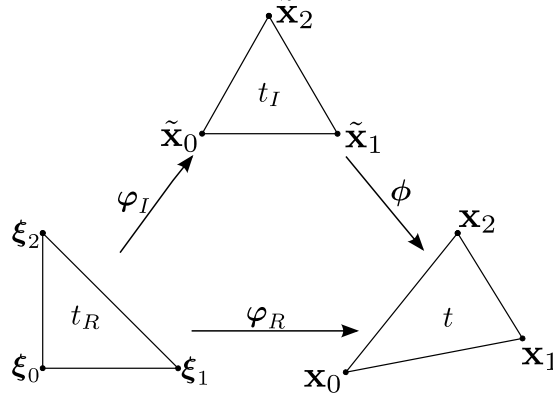
Figure 1: Mappings between the reference, the ideal and the physical elements.

elements with the nodes on parameterized surfaces. Afterwards, in Section 4, we detail the optimization procedure in terms of the parametric coordinates. We develop a non-linear least-squares problem in order to enforce the ideal configuration for the elements of the surface mesh. Finally, we present several examples to show the applications of the proposed method, Section 5.

2   Preliminaries

In this section, we first review the family of Jacobian-based distortion measures, presented in [2]. Second, we summarize the definition of distortion measure for planar high-order elements presented in [18], in which it is shown how to extend the Jacobian-based measures for linear triangles to planar high-order elements.

Let η be a Jacobian-based distortion measure for planar elements [2], with image $[1, \infty)$, taking value 1 for an ideal configuration of the element, and value ∞ when it is degenerated or tangled. Let q be the corresponding quality measure, defined as

$$q = \frac{1}{\eta}. \tag{1}$$

The image of the quality measure q is $[0, 1]$, taking value 1 for ideal configurations and 0 for degenerated ones. Our goal is to extend these measures to qualify high-order elements on parameterized surfaces.

2.1   Jacobian-based distortion measures for planar linear triangle elements

In order to determine the quality of a high-order element t on a parameterized surface, we generalize the Jacobian based quality measures for linear elements [2]. To this end, we consider a mapping ϕ from the ideal element t_I to the physical element t, see Figure 1. To determine this mapping, we consider the isoparametric mappings φ_R (from the reference element t_R to t) and φ_I (from t_R to t_I). For linear triangles, these mappings are affine.

Name	Distortion measure $\eta(\mathbf{S})$
Shape measure	$\eta(\mathbf{S}) = \dfrac{\|\mathbf{S}\|^2}{d \cdot \sigma(\mathbf{S})^{2/d}}$
Oddy et al. measure	$\eta(\mathbf{S}) = \dfrac{3}{d}\sigma^{-4/d}(\mathbf{S})\left(\|\mathbf{S}^T\mathbf{S}\|^2 - \dfrac{1}{3}\|\mathbf{S}\|^4\right)$

Table 1: Algebraic distortion measures for linear elements

The mapping between the ideal and the physical element is determined by

$$\boldsymbol{\phi} = \boldsymbol{\varphi}_R \circ \boldsymbol{\varphi}_I^{-1}. \tag{2}$$

Note that $\boldsymbol{\phi}$ is also an affine mapping, since $\boldsymbol{\varphi}_I^{-1}$ and $\boldsymbol{\varphi}_R$ are so. For linear elements it is usual to define a distortion measure in terms of the Jacobian matrix $\mathbf{S} := \mathbf{D}\boldsymbol{\phi}$. These distortion measures, herein denoted by $\eta(\mathbf{S})$, quantify a specific type of distortion of the physical element in a range scale $[1, \infty)$. Several distortion measures for linear triangles have been proposed in literature, see [2]. In Table 1 we present two distortion measures that we use to test the proposed high-order quality. Parameter d is the number of spatial dimensions, $\sigma(\mathbf{S})$ is the determinant of $\mathbf{S}$, and $\|\mathbf{S}\| = \sqrt{\mathrm{tr}(\mathbf{S}^t\mathbf{S})}$ is its Frobenius norm.

2.2 Distortion measure for planar high-order elements

Let t be a nodal high-order element of order p determined by n_p nodes with coordinates $\mathbf{x}_i \in \mathbb{R}^{d_x}$, for $i = 1, \ldots, n_p$ and being d_x the physical space dimension. Given a reference element t_R with nodes $\boldsymbol{\xi}_j \in \mathbb{R}^{d_\xi}$, being $j = 1, \ldots, n_p$ and d_ξ the reference space dimension, we consider the basis $\{N_i\}_{i=1,\ldots,n_p}$ of nodal shape functions (Lagrange interpolation) of order p. In this basis, the high-order isoparametric mapping from t_R to t can be expressed as:

$$
\begin{aligned}
\boldsymbol{\varphi}_R : \quad t_R \subset \mathbb{R}^d &\longrightarrow t \subset \mathbb{R}^d \\
\boldsymbol{\xi} &\longmapsto \mathbf{x} = \boldsymbol{\varphi}_R(\boldsymbol{\xi}; \mathbf{x}_1, \ldots, \mathbf{x}_{n_p}) = \sum_{i=1}^{n_p} \mathbf{x}_i N_i(\boldsymbol{\xi}),
\end{aligned}
\tag{3}
$$

where $\boldsymbol{\xi} = (\xi^1, \ldots, \xi^{d_\xi})^T$ and $\mathbf{x} = (x^1, \ldots, x^{d_x})^T$. Note that the shape functions $\{N_i\}_{i=1,\ldots,n_p}$ depend on the selection of $\boldsymbol{\xi}_j$, for $j = 1, \ldots, n_p$. In addition, they form a partition of the unity on t_R, and hold that $N_i(\boldsymbol{\xi}_j) = \delta_{ij}$, for $i, j = 1, \ldots, n_p$. In this paper we focus on nodal high-order triangular elements of order p, but the same approach is valid for quadrilaterals. Hence, the number of nodes n_p is $\frac{1}{2}(p+1)(p+2)$, and the space dimensions for planar meshes are $d_\xi = d_x = 2$. Therefore, the Jacobian of the isoparametric mapping (3) is a $d_x \times d_\xi = 2 \times 2$ matrix.

To define the high-order distortion measure of the physical element, we have to select first the ideal element t_I and a distribution of points. Herein, we choose a straight-sided equilateral triangle as the ideal element. In addition, we select the desired distribution of the nodes on the ideal element (*e.g.* equi-distributed or Fekete points). In general the

mappings $\boldsymbol{\varphi}_I$ and $\boldsymbol{\varphi}_R$, see Equation (3), are not affine. Hence, $\boldsymbol{\phi} = \boldsymbol{\varphi}_R \circ \boldsymbol{\varphi}_I^{-1}$ is also not affine, and the Jacobian matrix is not constant. The expression of the Jacobian is:

$$\mathbf{D}\boldsymbol{\phi}(\tilde{\mathbf{x}}; \mathbf{x}_1, \ldots, \mathbf{x}_{n_p}) = \mathbf{D}\boldsymbol{\varphi}_R(\boldsymbol{\varphi}_I^{-1}(\tilde{\mathbf{x}}); \mathbf{x}_1, \ldots, \mathbf{x}_{n_p}) \cdot \mathbf{D}\boldsymbol{\varphi}_I^{-1}(\tilde{\mathbf{x}}) \qquad (4)$$

where $\tilde{\mathbf{x}}$ is a point on the ideal element.

Similar to the linear element case, we define a distortion measure based on the Jacobian matrix of $\boldsymbol{\phi}$. However, the Jacobian of the elements is not constant. Nevertheless, the Jacobian on a point allows measuring the local deviation between the ideal and the physical element. Thus, we can obtain an elemental distortion measure by integrating the Jacobian based distortion measure on the whole ideal element.

Definition 1 *The* high-order distortion measure for a high-order planar element *with nodes* $\mathbf{x}_1, \ldots, \mathbf{x}_{n_p}$ *is*

$$\eta_\phi(\mathbf{x}_1, \ldots, \mathbf{x}_{n_p}) := \left(\frac{1}{|t_I|} \int_{t_I} \eta^2(\mathbf{D}\boldsymbol{\phi}(\tilde{\mathbf{x}}; \mathbf{x}_1, \ldots, \mathbf{x}_{n_p})) \ d\mathbf{x} \right)^{\frac{1}{2}}, \qquad (5)$$

where η is a distortion measure for linear elements based on the Jacobian matrix of the representation of the element, and $|t_I|$ is the area of the element element.

The *high-order quality measure for a high-order planar element* is $q_\phi := 1/\eta_\phi$, see [18] for an extended analysis for planar elements.

3 Distortion measure for high-order elements on parameterized surfaces

In this section, we first develop an analytical formulation to extend any Jacobian-based distortion measure for planar triangles η_ϕ, see Equation (5), to high-order elements with nodes on a parameterized surface Σ. As a result, we obtain a quality measure expressed in terms of the coordinates of the nodes in the parametric space of the surface.

3.1 Definitions

Assume that the surface Σ is parameterized by a continuously differentiable and invertible mapping

$$\begin{array}{cccc} \boldsymbol{\varphi}: & \mathcal{U} \subset \mathbb{R}^2 & \longrightarrow & \Sigma \subset \mathbb{R}^3 \\ & \mathbf{u} = (u, v) & \longmapsto & \mathbf{x} = \boldsymbol{\varphi}(\mathbf{u}), \end{array} \qquad (6)$$

where $\mathcal{U}$ is the parametric space of the surface. In this work, we use OpenCASCADE library [22] to retrieve the parameterization of the surfaces from the CAD model.

Similarly to the planar case, Equation (5), our objective is to quantify the distortion of the tangent vectors in each point of the surface elements. However, the tangent vectors on a point of the surface element live in the tangent plane, that is immersed in $\mathbb{R}^3$. Specifically, the Jacobian of the isoparametric mapping is a $d_x \times d_\xi = 3 \times 2$ not square
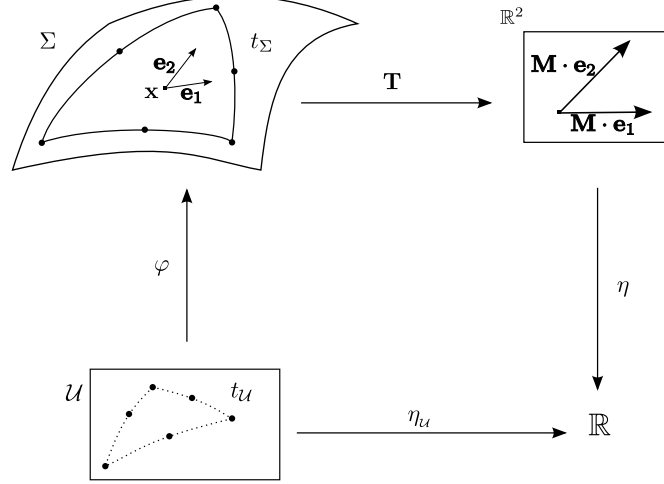
Figure 2: Diagram of mappings involved in the definition of the quality measure.

matrix. Therefore, we propose to define an embedding $\mathbf{T}$ from the tangent space on a point $\mathbf{x} = \boldsymbol{\phi}(\tilde{\mathbf{x}})$ of the surface element to $\mathbb{R}^2$, see Figure 2. Specifically, we define $\mathbf{T}$ as:

$$
\begin{aligned}
\mathbf{T}: \quad \mathbb{R}^3 \times \mathbb{R}^2 \quad &\longrightarrow \quad \mathbb{R}^2 \times \mathbb{R}^2 \\
\mathbf{D}\boldsymbol{\phi}(\tilde{\mathbf{x}}) \quad &\longrightarrow \quad \mathbf{M} \cdot \mathbf{D}\boldsymbol{\phi}(\tilde{\mathbf{x}}),
\end{aligned}
\tag{7}
$$

where $\mathbf{M}$ is a matrix composed by the two vectors corresponding to the basis derived from the Gram-Schmidt process applied to vectors $\mathbf{e}_1 := \frac{\partial \boldsymbol{\phi}}{\partial \tilde{x}^1}$ and $\mathbf{e}_2 := \frac{\partial \boldsymbol{\phi}}{\partial \tilde{x}^2}$. Hence, $\mathbf{M} = [\tilde{\mathbf{e}}_1 \quad \gamma \tilde{\mathbf{e}}_2]^T$, where $\tilde{\mathbf{e}}_i$, $i = 1, 2$ are the Gram-Schmidt orthonormal vectors, and $\gamma = \pm 1$ is determined to ensure a well oriented basis. Note that $\mathbf{T}(\mathbf{D}\boldsymbol{\phi})$ is a 2×2 matrix to which we can apply the Jacobian-based distortion measures presented in Section 2.1.

Finally, using the embedding (7) we can express the distortion and quality measures of the surface elements in terms of the parametric coordinates of the nodes, see Figure 2:

Definition 2 *The distortion measure for a high-order element on parametric coordinates with nodes $\mathbf{u}_1, \ldots, \mathbf{u}_{n_p} \in \mathcal{U}$ is*

$$
\eta_{\mathcal{U}}(\mathbf{u}_1, \ldots, \mathbf{u}_{n_p}) := \left(\frac{1}{|t_I|} \int_{t_I} \eta^2 \left(\mathbf{T} \left(\mathbf{D}\boldsymbol{\phi} \left(\tilde{\mathbf{x}}; \boldsymbol{\varphi}(\mathbf{u}_1), \ldots, \boldsymbol{\varphi}(\mathbf{u}_{n_p}) \right) \right) \right) d\mathbf{x} \right)^{\frac{1}{2}}.
\tag{8}
$$

Analogously, the quality measure for a high-order element on parametric coordinates is $q_{\mathcal{U}} := 1/\eta_{\mathcal{U}}$.

4   Application to high-order mesh optimization

In this section, we present an algorithm to optimize the distortion (quality) measure of triangular high-order meshes. It is important to point out that we want to ensure that the

nodes lie always on the surface. Therefore, the optimization approach is developed in the parametric space and the result is mapped to the surface by means of the parameterization.

The main goal of a simultaneous smoothing and untangling method is to obtain high-quality meshes composed by valid (non-inverted) elements. Note that the best possible result, can be characterized in terms of the distortion measure. That is, given a distortion measure η and a mesh $\mathcal{M}$ composed by n_N nodes and n_E elements, the node location is ideal if

$$\eta(\mathbf{T}(\mathbf{D}\phi_j(\tilde{\mathbf{x}}; \varphi(\mathbf{u}_{j_1}), \ldots, \varphi(\mathbf{u}_{j_{n_p}})))) = 1, \qquad \forall \tilde{\mathbf{x}} \in t_{I_j}, \ j = 1, \ldots, n_E, \tag{9}$$

where $e_j = (\varphi(\mathbf{u}_{j_1}), \ldots, \varphi(\mathbf{u}_{j_{n_p}}))$ is the jth element, t_{I_j} is the ideal element corresponding to e_j, and ϕ_j is the mapping between the jth ideal and physical elements. However, for a fixed mesh topology and a given surface the node location that leads to an ideal mesh distortion is not in general achievable. That is, the constrains in Equation (9) cannot be imposed strongly and therefore, we just enforce the ideal mesh distortion in the least-squares sense.

For a given mesh topology and a set of fixed nodes (nodes on the boundary of the domain), we formulate the least-squares problem in terms of the coordinates of a set of free nodes (nodes in the interior of the domain). To this end, and without loss of generality, we reorder the coordinates of the nodes, $\mathbf{u}_i$, in such a way that $i = 1, \ldots, n_F$ are the indices corresponding to the free nodes, and $i = n_F + 1, \ldots, n_N$ correspond to the fixed nodes. Thus, we can formulate the mesh optimization problem as

$$\min_{\mathbf{u}_1, \ldots, \mathbf{u}_{n_F}} f(\mathbf{u}_1, \ldots, \mathbf{u}_{n_F}; \mathbf{u}_{n_F+1}, \ldots, \mathbf{u}_{n_N}), \tag{10}$$

where f is the objective function, defined as:

$$f(\mathbf{u}_1, \ldots, \mathbf{u}_{n_F}; \mathbf{u}_{n_F+1}, \ldots, \mathbf{u}_{n_N}) := \sum_{j=1}^{n_E} \int_{t_{I_j}} (\eta(\mathbf{T}(\mathbf{D}\phi_j(\tilde{\mathbf{x}}; \varphi(\mathbf{u}_{j_1}), \ldots, \varphi(\mathbf{u}_{j_{n_p}})))) - 1)^2 \mathrm{d}\tilde{\mathbf{x}}.$$

In this work we illustrate the distortion and quality measures for high-order elements using the shape distortion measure presented Table 1. In order to untangle invalid meshes in a continuous optimization procedure, we use the modification of the Jacobian-based distortion measure presented in [9, 18].

5   Examples: mesh generation on CAD geometries

In this section we illustrate the overall process to generate a high-order mesh on a CAD geometry. Specifically, we select two different CAD geometries: a Falcon aircraft, Figure 3, and a component of a gear box, Figure 4. For each example geometry we show the complete sequence of steps of the a posteriori procedure to generate a valid high-order mesh:

Fig.	Min.Q.	Max.Q.	Mean Q.	Std.Dev.	Tang.
3(a)	0.21	1.00	0.93	0.09	0
3(c)	0.00	1.00	0.92	0.10	5
3(e)	0.25	1.00	0.97	0.06	0
4(a)	0.53	1.00	0.84	0.16	0
4(b)	0.00	1.00	0.73	0.34	130
4(c)	0.52	1.00	0.84	0.16	0

Table 2: Shape quality statistics of the meshes presented in Figure 3.

1. **A linear mesh is generated on the geometry**: Figure 3(a) shows the initial mesh generated on the Falcon aircraft, and Figure 4(a) the mesh generated on the gear box.

2. **The order of the mesh elements is increased**: We define a high-order node distribution for each element on the parametric space, and we map it to the surface. Note that for each geometry we have selected a different order. For instance, we use elements of order 3 for the Falcon aircraft, and elements of order 10 for the component of the gear box. In this step, tangled elements can be generated due to two main reasons:

   - The boundary elements can have auto-intersections due to the fact that in the parametric space the boundary edges are curved to fit the geometry, but the inner edges are maintained straight. This phenomena can be observed in Figure 4(b), .

   - If the quality of the parameterization is low, the composition of the high-order distribution on the parametric space together with the parameterization can lead to an invalid node distribution on the parametric space. This issue appears in the nose of the aircraft in Figure 3(d).

3. **The high-order mesh is optimized**: We apply the smoothing-untangling approach presented in Section 4 to the meshes. Figures 3(e) and 4(c) show the resulting meshes for each geometry.

The distortion measure selected in the presented examples is the shape distortion measure, detailed in Table 1. Table 2 details the quality statistics for each one of the presented meshes. Note that the obtained meshes are composed of high-quality elements. In all the cases we have untangled the initial inverted elements, and achieved a final high-quality configuration.

6 Concluding remarks

In this paper, we first detail a new technique to extend any distortion (quality) measure defined for planar elements to parameterized surfaces. Next, we develop an optimization
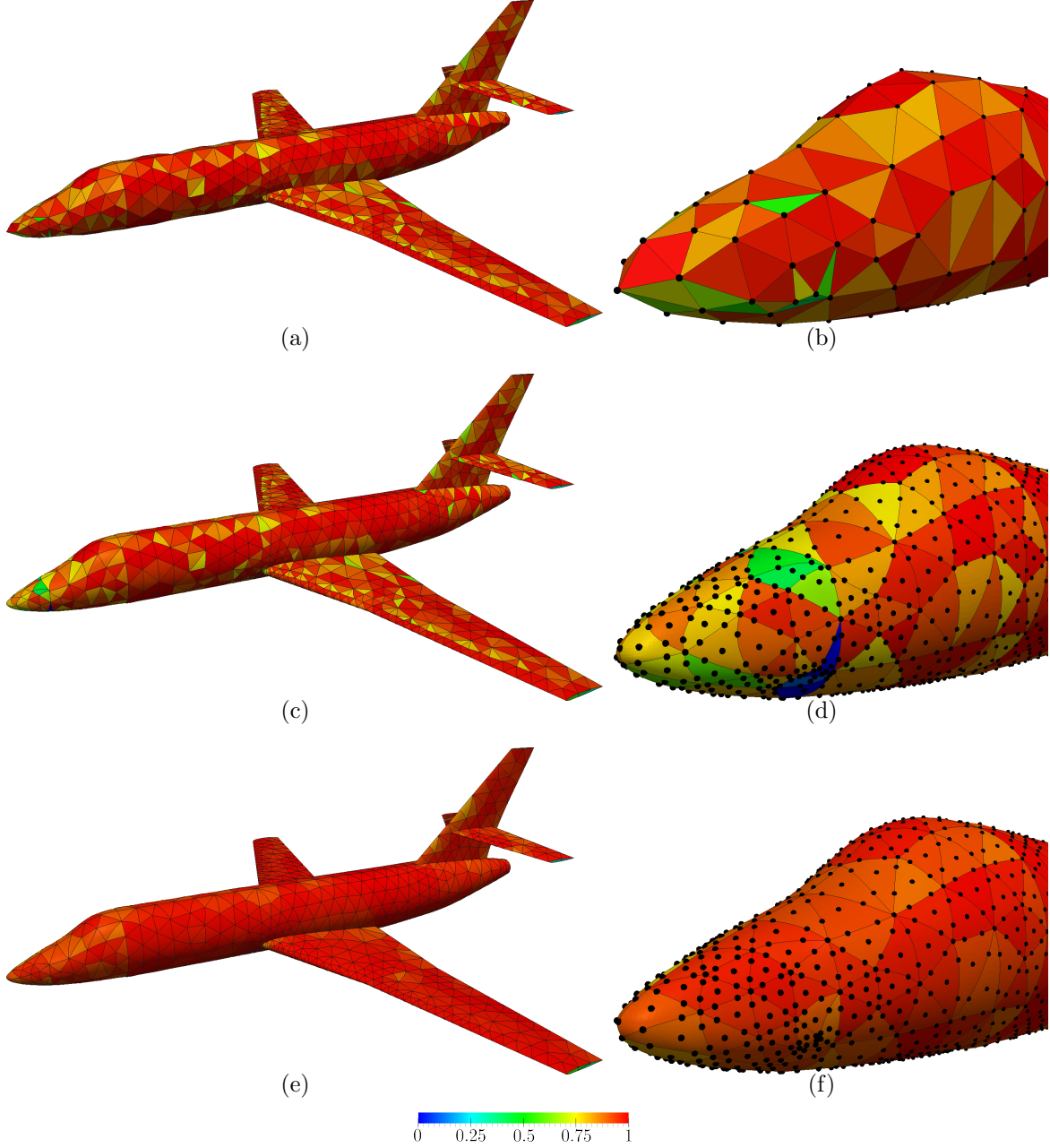
Figure 3: Order 3 mesh for a Falcon aircraft. The elements are colored according to the shape quality measure. (a,b) Initial linear mesh. (c,d) Initial order 3 mesh obtained after increase the order of the initial linear mesh. (e,f) Optimized order 3 mesh.

procedure to smooth and untangle meshes on parameterized surfaces. It is important to point out that the proposed measure expresses the quality of the elements on the surface in terms of the parametric coordinates of its nodes. Therefore, the optimization procedure

(a)

(b)

(c)
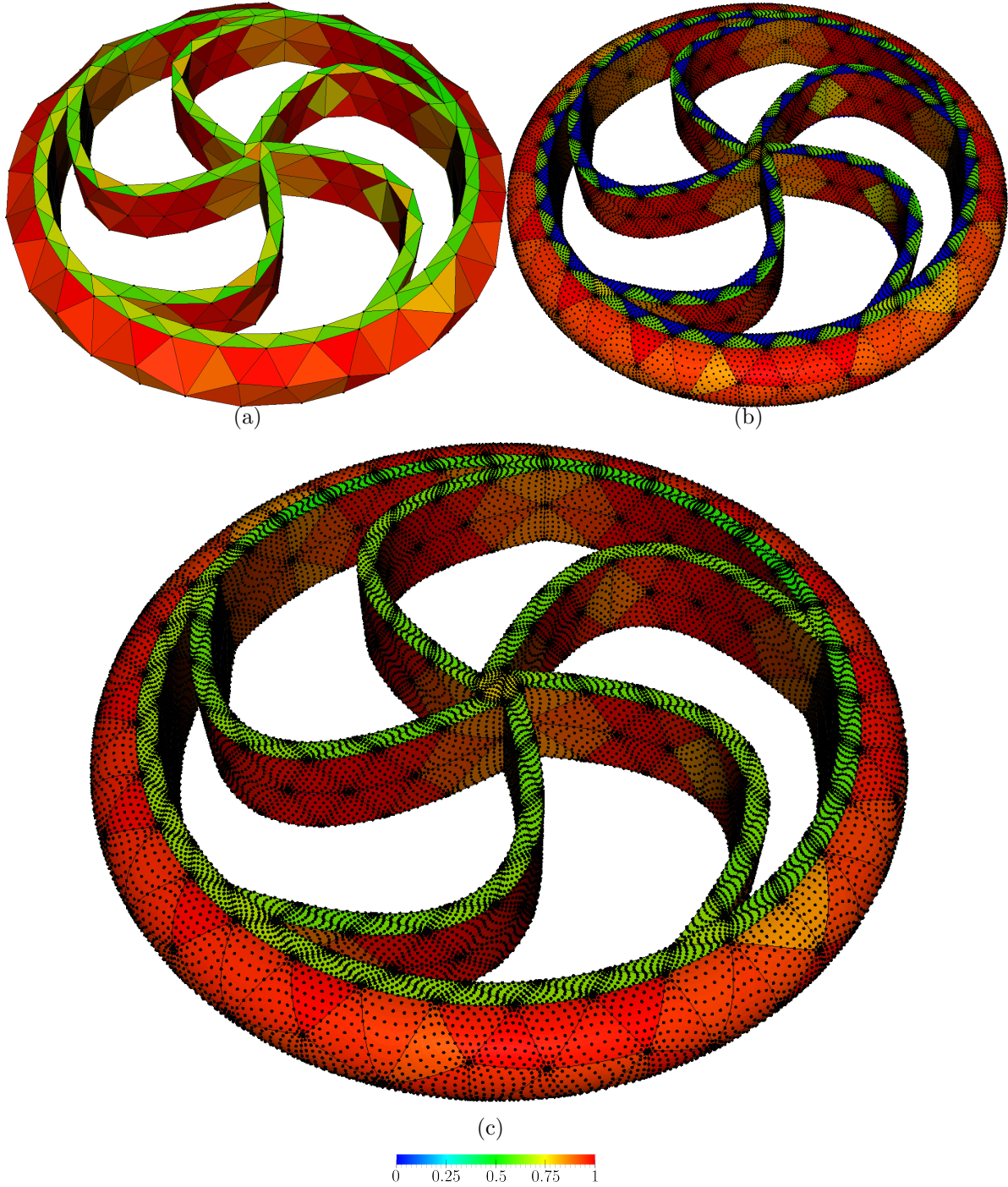
Figure 4: Order 10 mesh for a component of a gear box. The elements are colored according to the shape quality measure. (a,b) Initial linear mesh. (c,d) Initial order 10 mesh obtained after increase the order of the initial linear mesh. (e,f) Optimized order 10 mesh.

is also written in terms of the parametric coordinates. Hence, it ensures that the nodes are always placed on the surface. Finally, in the presented examples we have illustrated the mesh generation procedure with two different CAD geometries and with two different orders.

Acknowledgements

REFERENCES

[1] Field D. Qualitative measures for initial meshes. *Int. J. Numer. Methods Engrg.* 2000; **47**(4):887–906.

[2] Knupp PM. Algebraic mesh quality metrics. *SIAM J. Numer. Anal.* 2001; **23**(1):193–218.

[3] Knupp PM. Algebraic mesh quality metrics for unstructured initial meshes. *Finite Elem. Anal. Des.* 2003; **39**(3):217–241.

[4] Herrmann L. Laplacian-isoparametric grid generation scheme. *J. Engrg. Mech. Div.* 1976; **102**(5):749–756.

[5] Giuliani S. An algorithm for continuous rezoning of the hydrodynamic grid in arbitrary lagrangian-eulerian computer codes. *Nucl. Engrg. Des.* 1982; **72**(2):205–212.

[6] Knupp PM. A method for hexahedral mesh shape optimization. *Int. J. Numer. Methods Engrg.* 2003; **58**(2):319–332.

[7] Freitag LA, Plassmann P. Local optimization-based simplicial mesh untangling and improvement. *Int. J. Numer. Methods Engrg.* 2000; **49**:109–125.

[8] Freitag LA, Knupp PM. Tetrahedral mesh improvement via optimization of the element condition number. *Int. J. Numer. Methods Engrg.* 2002; **53**:1377–1391.

[9] Escobar JM, Rodríguez E, Montenegro R, Montero G, González-Yuste JM. Simultaneous untangling and smoothing of tetrahedral meshes. *Comput. Methods Appl. Mech. Engrg.* 2003; **192**(25):2775–2787.

[10] Wilson T, Sarrate J, Roca X, Montenegro R, Escobar J. Untangling and smoothing of quadrilateral and hexahedral meshes. *Proceedings of the Eight International Conference on Engineering Computational Technology*, Topping BHV (ed.), Civil-Comp Press, Stirlingshire, UK, Paper 36, 2012. doi:10.4203/ccp.100.36, doi: 10.4203/ccp.100.36.

[11] Escobar JM, Montero G, Montenegro R, Rodríguez E. An algebraic method for smoothing surface triangulations on a local parametric space. *Int. J. Numer. Methods Engrg.* 2006; **66**(4):740–760.

[12] Frey PJ, Borouchaki H. Geometric surface mesh optimization. *Comput. Visual. Sci.* 1998; **1**(3):113–121.

[13] Jiao X, Wang D, Zha H. Simple and effective variational optimization of surface and volume triangulations. *Engrg. Comput.* 2011; **27**:81–94.

[14] Garimella R, Shashkov M, Knupp PM. Triangular and quadrilateral surface mesh quality optimization using local parametrization. *Comput. Methods Appl. Mech. Engrg.* 2004; **193**(9–11):913–928.

[15] Shivanna K, Grosland N, Magnotta V. An analytical framework for quadrilateral surface mesh improvement with an underlying triangulated surface definition. *Proc. 19th International Meshing Roundtable*, Chattanooga, 2010; 85–102.

[16] Gargallo-Peiró A, Roca X, Peraire J, Sarrate J. Defining quality measures for mesh optimization on parameterized cad surfaces. *Proc. 21st International Meshing Roundtable*, Jiao X, Weill JC (eds.). Springer Berlin Heidelberg, 2013; 85–102.

[17] Persson PO, Peraire J. Curved mesh generation and mesh refinement using lagrangian solid mechanics. *Proc. 47th AIAA Aerospace Sciences Meeting and Exhibit*, 2009.

[18] Roca X, Gargallo-Peiró A, Sarrate J. Defining quality measures for high-order planar triangles and curved mesh generation. *Proc. 20th International Meshing Roundtable*. Springer Berlin Heidelberg, 2012; 365–383.

[19] Xie Z, Sevilla R, Hassan O, Morgan K. The generation of arbitrary order curved meshes for 3d finite element analysis. *Comput. Mech.* 2012; :1–14.

[20] George PL, Borouchaki H. Construction of tetrahedral meshes of degree two. *Int J Numer Meth Eng* 2012; **90**(9):1156–1182.

[21] Johnen A, Remacle JF, Geuzaine C. Geometrical validity of curvilinear finite elements. *J. Comput. Phys.* 2013; **233**(0):359 – 372.

[22] CASCADE O. Open CASCADE Technology, 3D modeling and numerical simulation. `www.opencascade.org` 2012.

VI International Conference on Adaptive Modeling and Simulation
ADMOS 2013
J. P. Moitinho de Almeida, P. Díez, C. Tiago and N. Parés (Eds)

ON THE REDUCTION OF SOLID DYNAMICS EQUATIONS AND THEIR REAL-TIME DIRECT INTEGRATION

D. GONZÁLEZ[*], E. CUETO[*] AND F. CHINESTA[†‡]

[*] Aragón Institute of Engineering Research, I3A
Universidad de Zaragoza, Spain
Email: {gonzal,ecueto}@unizar.es

[†] EADS Corporate Foundation Chair
Ecole Centrale de Nantes, France
Email: Francisco.Chinesta@ec-nantes.fr

[‡] Institut Universitaire de France

Key words: Solid dynamics, model reduction, proper generalized decomposition, real-time.

Abstract. In this work we propose a particular form of the solid dynamics equations in a multidimensional framework. This particular viewpoint of the traditional dynamics equations is then discretized and solved by PGD approximations very efficiently.

As a result, the method here proposed provides with a meta-model or surface response of the system subjected to any type of force (taking values in a given interval) and for any initial boundary conditions. Since the time interval in which the integration is performed can be virtually any, and any integration scheme can be used, the proposed method, combined with an on-line/off-line strategy, provides with a sort of "black box" that computes the response of the system for apparently very large time intervals with great accuracy and under real time requirements.

Several aspects will be studied, such as the parameterization of the space of initial conditions, efficiency issues and others, together with examples showing the potential of the proposed technique.

Special attention will be paid to the issue of choosing the appropriate time step, the number of modes, strategies for optimal balance between accuracy and speed of simulation, etc.

PARALLEL ADAPTIVE MESH ALGORITHMS FOR MULTICORE ARCHITECTURES

GERARD J. GORMAN[*], GEORGIOS ROKOS[†], PAUL H. J. KELLY[†]

[*]Department of Earth Science and Engineering
Imperial College London
London, SW7 2AZ, UK
e-mail: g.gorman@imperial.ac.uk, www.imperial.ac.uk/earthscienceandengineering

[†]Department of Computing
Imperial College London
London, SW7 2AZ, UK
e-mail: georgios.rokos09@imperial.ac.uk, p.kelly@imperial.ac.uk,
www.imperial.ac.uk/computing

Key words: parallel, adaptive mesh algorithms, multicore, MPI, OpenMP, NUMA

Abstract. Problems that require dynamic adaptation of an unstructured mesh are particularly challenging for multicore architectures. This work tackles concurrency control, memory management and locality in the context of adaptive mesh finite element simulations. We present experimental analysis of a range of implementation alternatives, and we demonstrate that good performance and parallel speedup are achievable. We study both OpenMP intra-node parallelisation and MPI internode parallel execution, with particular attention to memory hierarchy and NUMA issues.

AN ADAPTIVE MESH REFINEMENT ALGORITHM BASED ON ELEMENT SUBDIVISION WITH APPLICATION TO GEOMATERIALS

CLAIRE E. HEANEY[‡], PAUL G. BONNIER[†], RONALD B.J. BRINKGREVE[∗†] and MICHAEL A. HICKS[∗]

[‡] Cardiff University
Institute of Mechanics and Advanced Materials
Cardiff School of Engineering
Queen's Buildings, The Parade
Cardiff CF24 3AA
Wales, UK
e-mail: claire.e.heaney@gmail.com

[∗]Delft University of Technology
Geo-Engineering Section
Stevinweg 1
2628 CN Delft
The Netherlands

[†]Plaxis BV
P.O. Box 572
2600 AN Delft
The Netherlands

Key words: Adaptive Mesh Refinement, Elastoplasticity, Geomaterials

Abstract. The implementation of Adaptive Mesh Refinement (AMR) within the geotechnical software package PLAXIS 2D is described in this paper. A recovery-based algorithm is used which aims to reduce the discretisation error by refining the mesh during the solution process. The error is estimated with a Zienkiewicz-Zhu-type error estimator but based on the incremental deviatoric strain instead of stress. The deviatoric strain field is compared with an improved field calculated by superconvergent patch recovery. Once elements with large errors have been detected, mesh refinement takes place. A combination of regular subdivision and longest-edge bisection is employed. Mapping history variables from the old mesh to the new mesh is accomplished by using the recovered solutions and the shape functions. The AMR algorithm is demonstrated for a biaxial compression test.

1 INTRODUCTION

Adaptive Mesh Refinement (AMR) is a technique which refines a mesh as part of the solution procedure of a boundary value problem. The aim is to achieve the best mesh possible (within certain constraints) for each load level. AMR is applied to problems which have features developing on different scales. In geotechnics, a retaining wall is one such an example. The boundary condition will be applied along the wall which may have a height of the order of metres. Should the load be large enough, a shear band or failure surface may develop in the soil behind the wall with a width of millimetres. In order to properly model the shear band, the elements need to be sufficiently small. For most of the domain this would be a waste of computational effort, and, as the location of the shear band is not known *a priori*, AMR presents as an obvious solution by automatically generating small elements where they are most needed.

AMR works by quantifying the discretisation error in the finite element (FE) solution, and, when the global error norm exceeds a specified tolerance, certain elements are marked for refinement. Once the mesh has been refined and the solution has been mapped to the new mesh, the loading is resumed. This allows the mesh to be refined in regions where the error is high (eg. due to the large strain gradients) resulting in a more accurate description of the shear band.

It has recently been suggested that AMR is not widely used, either in industry (for structural mechanics applications) [1, 2] or in the field of geotechnics [3]. Noteworthy exceptions in geotechnics are [4, 5, 6, 3]. This paper aims to implement a straightforward AMR algorithm which progressively refines the mesh when the global error exceeds a given tolerance.

2 FORMULATION

AMR has been implemented within Plaxis's 2D displacement-based FE solver [7]. Throughout this paper 6-noded triangular elements are used within unstructured meshes. The approximation of the displacements is therefore quadratic. Load or displacement boundary conditions are applied in increments as is usual for materials that behave non-linearly. Once an increment has converged, the AMR algorithm recovers nodal fields and then calculates the error. If the error exceeds the user-defined tolerance, the refinement and mapping algorithms are called. The FE solver then applies the next increment. The four stages of the algorithm are now briefly discussed. A more detailed description can be found in [8].

2.1 Recovery

Variables defined at integration points are recovered at the nodes in order to estimate the error and to facilitate the mapping process. For this purpose, Superconvergent Patch Recovery (SPR) is used [9]. It is an efficient, local method involving the inversion of a relatively small matrix (small when compared to the size of the global stiffness matrix).

The implementation of SPR involves two stages: first, assembling the patches (groups of neighbouring elements) and second, fitting least squares surfaces to the integration point values in each patch. Node-based patches are implemented here, which are formed around each interior vertex node (often referred to as an "assembly node"). The patch is composed of all those elements which contain the assembly node. To recover the solution at nodes on or near the boundary, the standard node-based patches containing elements in contact with the boundary can be extended to include the boundary nodes (as suggested in [10]).

In each patch, a least squares fit is carried out to the data at integration points. In SPR the least squares fit is of the same degree as the displacement shape functions, which are quadratic in this case. Each patch is mapped onto the domain $[-1, 1] \times [-1, 1]$ to avoid ill-conditioning of the matrix which is inverted to obtain the least squares coefficients.

2.2 Error estimation

Zienkiewicz et. al. [11] prove that their error estimator (described in [12]) is asymptotically exact for linear elastic problems. For elastoplastic problems Boroomand et. al. [13] introduce an error estimator based on incremental energy. More recently Hicks [6] employed an error estimator based upon a measure of incremental deviatoric strain. The incremental shear strain invariant, $\Delta\gamma$, is defined as

$$\Delta\gamma = \sqrt{\Delta\varepsilon_{ij}^{\mathrm{dev}} \Delta\varepsilon_{ij}^{\mathrm{dev}}} \ , \tag{1}$$

where $\Delta\varepsilon_{ij}^{\mathrm{dev}}$ is the ij^{th} component of the incremental deviatoric strain tensor. The error in this quantity with respect to the L_2 norm is given by

$$||e||_{\Delta\gamma} = \sqrt{\int_{\Omega} (\Delta\gamma^* - \Delta\gamma)^2 \, d\Omega} \ , \tag{2}$$

where $\Delta\gamma^*$ is the recovered field based on the recovered incremental strains. This estimator should be especially suitable for geomechanical problems which exhibit large changes in strain.

The error estimator for element $\mathtt{iEl}$ is given by

$$||e||_{\Delta\gamma}^{\mathtt{iEl}} = \sqrt{\int_{\Omega_{\mathtt{iEl}}} (\Delta\gamma^* - \Delta\gamma)^2 \, d\Omega} \ , \tag{3}$$

and from this, the global error norm can then be calculated:

$$||e||_{\Delta\gamma} = \sqrt{\sum_{\mathtt{iEl}=1}^{\mathtt{nEl}} \left(||e||_{\Delta\gamma}^{\mathtt{iEl}} \right)^2} \ . \tag{4}$$

Either of the above error measures can be made relative by dividing by

$$||\Delta\gamma^*|| = \sqrt{\int_\Omega (\Delta\gamma^*)^2 \, d\Omega} \; . \tag{5}$$

Both local and global errors are used in this implementation of AMR. The global estimate is compared against a user-defined tolerance, and, if it exceeds the tolerance, mesh refinement is triggered. The local (or element) error indicates which elements should be refined. The approach followed here is to link the user-defined global tolerance with the local, element scale to provide a local tolerance. This method relies on the concept of an optimal mesh where the error is equally distributed over the elements (for example [14]). Given a user-defined global tolerance η (which is a relative measure), the condition for triggering refinement is

$$\frac{||e||_{\Delta\gamma}}{||\Delta\gamma^*||} \geqslant \eta \; . \tag{6}$$

Introducing the assumption that the local errors are equally distributed over the mesh gives

$$||e||_{\Delta\gamma}^{\text{local}} \geqslant \frac{\eta \, ||\Delta\gamma^*||}{\sqrt{\texttt{nEl}}} \; , \tag{7}$$

where $\texttt{nEl}$ is the total number of elements. So, once the global error exceeds the user-defined (global) tolerance, then all elements whose error exceeds $\frac{\eta \, ||\Delta\gamma^*||}{\sqrt{\texttt{nEl}}}$ will be marked for refinement.

2.3 Mesh Refinement

There are two ways of refining a mesh in $h-$adaptivity: by regeneration or by subdivision. Regeneration is often used and produces elegant-looking meshes directly indicative of the underlying physical mechanism. However, once the mesh has been regenerated, the solution must be mapped from the old mesh to the new mesh throughout the entire domain. In order to limit the numerical diffusion which may occur as a result of transferring the solution between meshes, and to be more efficient, subdivision is used. Here the transfer of the solution is carried out only within elements which have been refined. One way to subdivide elements is to use regular bisection. This refers to the splitting of an element into four elements by joining the midpoints of the triangle's edges [15]. Between a regularly-refined element and a non-refined element will be a non-conforming edge with hanging nodes. A straightforward solution to this is adopted here based on Rivara's longest-edge bisection method [16]. It is applied to any elements which, after being regularly refined, have non-conforming edges. This procedure is repeated until the mesh is conforming.

2.4 Mapping

New elements require values of displacement and stress, and also any state variables which are associated with the constitutive model. Displacements can be mapped using the shape functions and the nodal values of displacement. In order to map the stresses, they are first recovered at the nodes using SPR. Then they can be mapped to the new mesh just as the displacements were. Any integration-point values can be mapped using this method. After mapping, the stress field will no longer be in equilibrium with the applied force. A zero-load increment can be applied in order to resolve this, if necessary.

3 RESULTS

A drained biaxial compression test was simulated to demonstrate the performance of the AMR algorithm. The geometry and boundary conditions can be seen in Figure 1, where L, the width of the sample, is used to normalise the computed displacements. The constitutive model is linear elastic, perfectly plastic with a Mohr-Coulomb yield criterion and associated flow. The parameters are Young's Modulus $E = 10\,\text{MPa}$, Poisson's ratio $\nu = 0.2$, friction angle $\phi = 30°$ and cohesion $c = 5.5\,\text{kPa}$. Vertical displacement increments are applied to the top boundary along which horizontal displacement is prevented. The base is fully fixed.
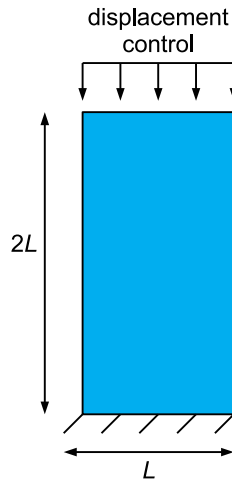
Figure 1: The biaxial test modelled in plane strain, with a fully fixed base and a displacement control through a rough and rigid end plateau.

First, the biaxial problem was simulated with the standard FE method using a fixed mesh (ie. with no adaptivity). Six-noded triangular elements were used. The global error norm was calculated based on the incremental deviatoric strain estimator defined in Section 2.2. The behaviour of the estimator can be seen in Figure 2 for a fixed mesh of 270 elements of approximately equal size. The error increases extremely rapidly during steps 4 to 7. As can be seen from Figure 4, this corresponds to the approach to the peak

of the load-displacement curve as the solution becomes non-homogeneous. The global error norm then reduces until around step 20, after which it remains constant. The global error norm controls when adaptive remeshing is triggered and is therefore crucial to the refinement process. This plot would suggest that the incremental deviatoric strain estimator is a suitable measure as it is able to identify the onset of localisation which occurs as the peak in the load-displacement curve is approached.

An AMR simulation was then carried out, starting from a mesh of 270 elements with a global error tolerance of 2.5%. Figure 3 shows the global error norm in this case, with the broken horizontal line representing the global error tolerance. The filled circles indicate a step which ends with refinement. It can be seen that refinement is initiated four times in the vicinity of the peak of the load-displacement curve, at step numbers 4, 5, 6 and 10. (At the end of the simulation, the error exceeds the tolerance; however, as over 95% of the displacements have already been applied, refinement is not allowed.)
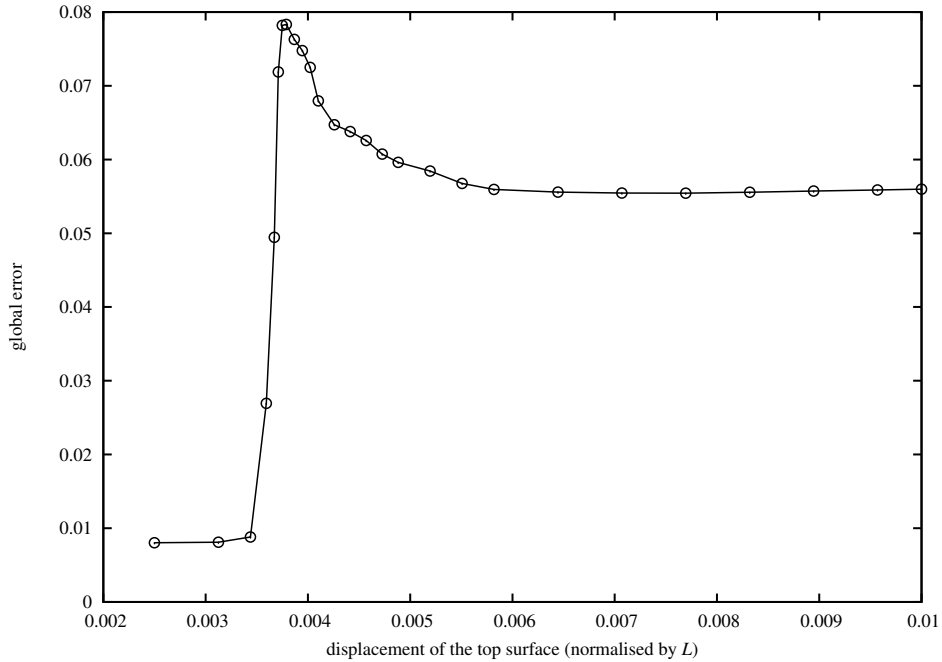
Figure 2: The global error norm using the error estimator based on the incremental deviatoric strain plotted against the normalised displacement boundary condition.

Load-displacement curves are plotted in Figure 4. Curves from two fixed-mesh simulations are shown for comparison, these using 142 and 4162 elements. The AMR results agree extremely well with the results from the finest fixed mesh, the load-displacement curves being almost coincident. The steps which trigger refinement are indicated by hollow circles. The load-displacement curve for the AMR simulation remains smooth despite the mesh refinement.
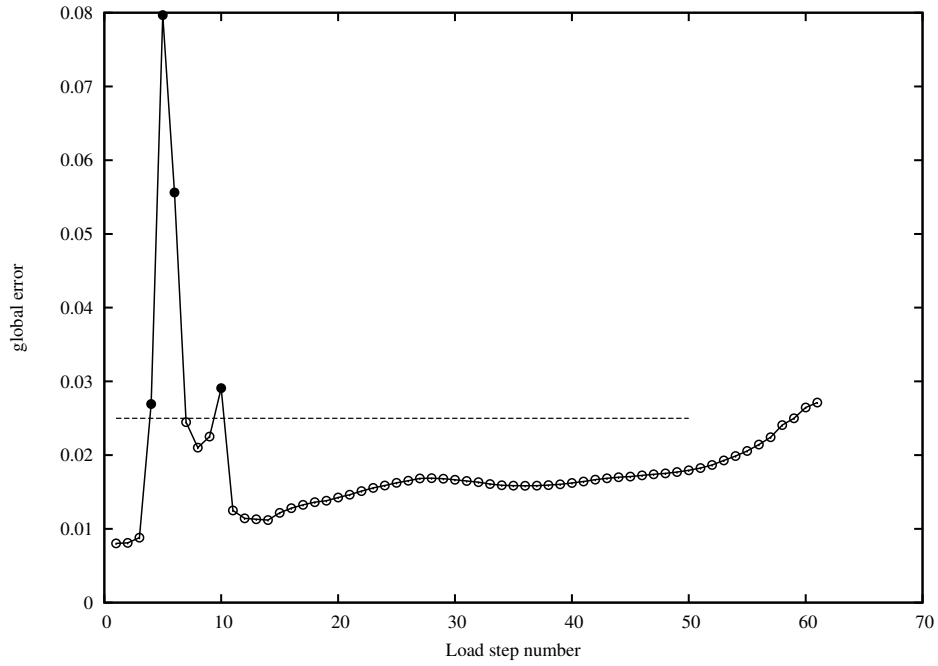
Figure 3: The global error norm plotted against the number of (load) steps during adaptive mesh refinement. Filled circles indicate when refinement is triggered.
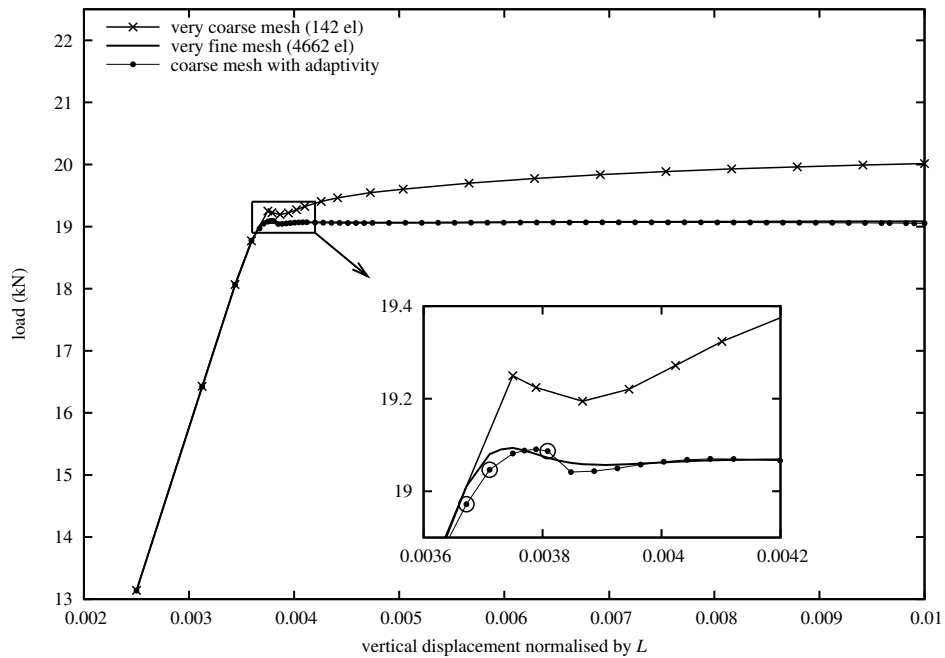
Figure 4: Load-displacement curves for the AMR simulation and for two standard FE simulations with fixed meshes.

Finally, Figure 5 shows plots of the incremental shear strain invariant ($\Delta\gamma$) at the end of the simulation. The results agree well with the analytical prediction for the inclination of the shear band (ϑ). According to Roscoe [17], this is given by $\vartheta = \pi/4 + \psi/2$ where ψ is the dilation angle (in radians).
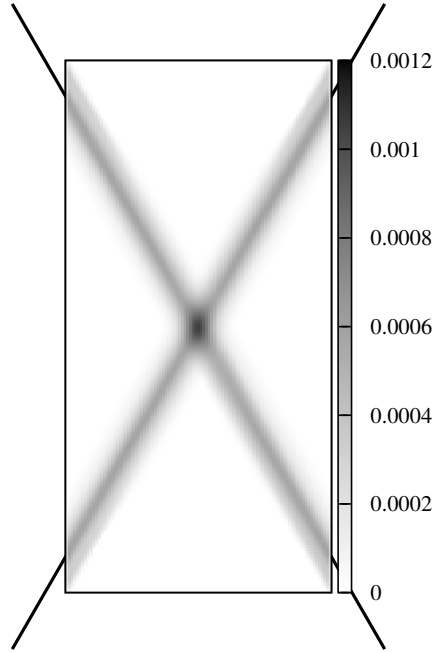
Figure 5: A contour map of the incremental deviatoric strain ($\Delta\gamma$) at the final step for the AMR algorithm. The black lines indicate the inclination of the shear band as predicted by Roscoe.

4   CONCLUSIONS

An AMR algorithm has been implemented in Plaxis 2D. Results are shown for a biaxial compression test. The implementation is recovery based and uses element subdivision rather than mesh regeneration when refinement is required. The adaptive algorithm triggers relatively few refinement steps but is nevertheless effective due to the success of the incremental error estimator in pinpointing the onset of localisation. The load-displacement curve is smooth despite the mesh refinement. This would suggest that the transfer of data from the old to the new mesh is carried out effectively. The inclination of the shear band from the numerical results agrees very well with the analytical prediction given by Roscoe.

Acknowledgments

REFERENCES

[1] Pashley, D. Adaptive Re-meshing in Structural Analysis - Part 1. *Benchmark (NAFEMS publication)* (April, 2011) 11-12.

[2] Pashley, D. Adaptive Re-meshing in Structural Analysis - Part 2. *Benchmark (NAFEMS publication)* (October, 2011) 11-13.

[3] Kardani, M. and Nazem, M. and Abbo, A.J. and Sheng, D. and Sloan, S.W. Refined h-adaptive FE procedure for large deformation geotechnical problems. *Comput. Mech.* (2012) **49**:21-33.

[4] Perić, D. and Hochard, Ch. and Dutko, M. and Owen, D.R.J. Transfer operators for evolving meshes in small strain elasto-plasticity. *Comput. Methods Appl. Mech. Engrg.* (1996) **137**:331–344.

[5] Hu, Y. and Randolph, M.F. H-adaptive FE analysis of elasto-plastic non-homogeneous soil with large deformation. *Comput. Geotech.* (1998) **23**:61–83.

[6] Hicks, M.A. Coupled computations for an elastic-perfectly plastic soil using adaptive mesh refinement. *Int. J. Numer. Anal. Meth. Geomech.* (2000) **24**:453–476.

[7] Brinkgreve, R.B.J. and Swolfs, W.M. and Engin, E. *Plaxis 2D 2011* (2011).

[8] Heaney, C.E. and Bonnier, P.G. and Brinkgreve, R.B.J. and Hicks, M.A. Adaptive Mesh Refinement with application to geomaterials. *In preparation for submission to Comput. Geotech.* (2013)

[9] Zienkiewicz, O.C. and Zhu, J.Z. The superconvergent patch recovery and *a posteriori* error estimates. *Int. J. Numer. Meth. Engng.* (1992) **33**:1331–1364.

[10] Zienkiewicz, O.C. and Zhu, J.Z. Superconvergence and the superconvergent patch recovery. *Finite Elem. Anal. Des.* (1995) **19**:11–23.

[11] Zienkiewicz, O.C. and Zhu, J.Z. The superconvergent patch recovery (SPR) and adaptive finite element refinement. *Comput. Methods Appl. Mech. Engrg.* (1992) **101**:207–224.

[12] Zienkiewicz, O.C. and Zhu, J.Z. A simple error estimator and adaptive procedure for practical engineering analysis. *Int. J. Numer. Meth. Engng.* (1987) **24**:337–357.

[13] Boroomand, B. and Zienkiewicz, O.C. Recovery procedures in error estimation and adaptivity. *Comput. Methods Appl. Mech. Engrg.* (1999) **176**:127–146.

[14] Mar, A. and Hicks, M.A. A benchmark computational study of finite element error estimation. *Int. J. Numer. Meth. Engng.* (1996) **39**:3969–3983.

[15] Bank, R.E. and Sherman, A.H. and Weisser, A. Refinement algorithms and data structures for regular local mesh refinement. *Scientific computing (IMACS Transactions).* (1983) 3–17.

[16] Rivara, M.C. Design and data structures of a fully adaptive multigrid finite element software. *ACM Transactions on Mathematical Software.* (1984) **10**:242–264.

[17] Roscoe, K.H. The influence of strains in soil mechanics. *Géotechnique* (1970) **20**:129–170.

INEXACT-HESSIAN-VECTOR PRODUCTS FOR EFFICIENT REDUCED-SPACE PDE-CONSTRAINED OPTIMIZATION

Jason E. Hicken

Department of Mechanical, Aerospace, and Nuclear Engineering
Rensselaer Polytechnic Institute
Troy, New York, 12180, United States
e-mail: hickej2@rpi.edu, www.optimaldesignlab.com

Key words: Adaptive Modeling, Simulation, PDE-constrained optimization, second-order adjoints, matrix-free, Hessian-vector products

Abstract. We investigate reduced-space Newton-Krylov (NK) algorithms for engineering parameter optimization problems constrained by partial-differential equations. We review reduced-space and full-space optimization algorithms, and we show that the efficiency of the reduced-space strategy can be improved significantly with inexact-Hessian-vector products computed using approximate second-order adjoints. Results demonstrate that the proposed reduced-space NK algorithm has excellent scaling that makes it suitable for large-scale optimization problems. Moreover, reduced-space NK combines the attractive attributes of both reduced-space quasi-Newton methods and full-space approaches — namely, modularity, robustness, and scalibilty.

1 Introduction

Partial differential equation (PDE) constrained optimization problems can be posed in the full-space or the reduced-space. In full-space formulations the PDE state variables — e.g. pressure and velocity for incompressible flows — are included as optimization variables, and the PDE becomes an explicit constraint in the optimization. In contrast, reduced-space formulations treat the state variables as implicit functions of the design variables: for a given set of design variables the PDE is solved for the states.

In practice, engineers often prefer reduced-space formulations. Reduced-space methods lend themselves to modularity, so implementation is typically easier than full-space methods. Unfortunately, conventional reduced-space optimization algorithms exhibit poor algorithmic scaling. For example, the computational cost of limited-memory quasi-Newton methods is often proportional to the number of design variables. This scaling limits the number of design variables that can be considered.

Motivated by the above observations, we consider reduced-space inexact-Newton-Krylov (INK) algorithms, which offer the potential for design-dimension-independent algorithmic scaling. One of the challenges with reduced-space INK methods for PDE-constrained optimization is the efficient computation of Hessian-vector products needed by the Krylov solver. In particular, it is widely believed that these products must computed with high accuracy to avoid convergence difficulties. This accuracy requirement can render reduced-space INK methods orders of magnitude more expensive than full-space methods [11]. In this paper, we argue that the Hessian-vector products can be computed inexactly, and numerical examples demonstrate that the resulting reduced-space INK algorithm offers an attractive alternative to its full-space counterpart.

2 PDE-constrained Optimization: Formulations and Algorithms

In this section we briefly review the generic PDE-constrained optimization problem and highlight commonly used formulations and solution strategies. For a comprehensive review of solution methods see, for example, [1].

2.1 Problem and Notation

We are interested in solving the following PDE-constrained optimization problem:

$$
\begin{aligned}
&\text{minimize} &&\mathcal{J}(\mathbf{x}, \mathbf{u}), &&\mathbf{x} \in \mathbb{R}^m,\ \mathbf{u} \in \mathbb{R}^n, \\
&\text{subject to} &&\mathcal{R}(\mathbf{x}, \mathbf{u}) = \mathbf{0}.
\end{aligned}
\tag{1}
$$

The objective functional is $\mathcal{J}$, which we will assume is C^2 continuous on its domain. The variables $\mathbf{x}$ and $\mathbf{u}$ denote the finite-dimensional control and state variables, respectively. In the context of PDE-constrained optimization, the state variables arise from the chosen discretization of the PDE; $\mathbf{u}$ may represent function values at nodes in a mesh or coefficients in a basis expansion. The control variables can be given a similar interpretation. The PDE itself, together with appropriate boundary and initial conditions, is represented by the equation $\mathcal{R}(\mathbf{x}, \mathbf{u}) = \mathbf{0}$.

A local solution of (1) must satisfy the first-order optimality conditions, which can be found by differentiating the Lagrangian. In order to define the Lagrangian of (1), we first introduce the symmetric positive definite matrix $P \in R^{n \times n}$ that defines a discrete inner product appropriate to the chosen discretization of the PDE. Then, the Lagrangian is given by

$$
\mathcal{L}(\mathbf{x}, \mathbf{u}, \boldsymbol{\psi}) \equiv \mathcal{J}(\mathbf{x}, \mathbf{u}) + \boldsymbol{\psi}^T P \mathcal{R}(\mathbf{x}, \mathbf{u}),
\tag{2}
$$

where $\boldsymbol{\psi} \in \mathbb{R}^n$ are the Lagrange multipliers, also called the adjoint or costate variables in the context of PDE-constrained optimization.

Thus, the first-order optimality conditions for a solution to (1) are [20]:

$$\mathcal{L}_{\boldsymbol{\psi}} = 0 \qquad \Rightarrow \qquad P\boldsymbol{\mathcal{R}}(\mathbf{x}, \mathbf{u}) = 0, \tag{3a}$$

$$\mathcal{L}_{\mathbf{u}} = 0 \qquad \Rightarrow \qquad \mathcal{J}_{\mathbf{u}}(\mathbf{x}, \mathbf{u}) + \boldsymbol{\psi}^T P\boldsymbol{\mathcal{R}}_{\mathbf{u}}(\mathbf{x}, \mathbf{u}) = 0, \tag{3b}$$

$$\mathcal{L}_{\mathbf{x}} = 0 \qquad \Rightarrow \qquad \mathcal{J}_{\mathbf{x}}(\mathbf{x}, \mathbf{u}) + \boldsymbol{\psi}^T P\boldsymbol{\mathcal{R}}_{\mathbf{x}}(\mathbf{x}, \mathbf{u}) = 0. \tag{3c}$$

Subscripted variables indicate differentiation with respect to that variable, e.g. $\mathcal{J}_{\mathbf{u}} \equiv \partial\mathcal{J}/\partial\mathbf{u}$. The first-order conditions (3) are also called the Karush-Kuhn-Tucker, or KKT, conditions.

2.2 Full-space Approach

The KKT conditions are a set of nonlinear algebraic equations, so a natural solution strategy is Newton's method with an appropriate globalization. As usual, the potential for rapid convergence makes Newton's method attractive. The full-space approach that we adopt here is based on the Lagrange-Newton-Krylov-Schur (LNKS) method of Biros and Ghattas [2, 3]. One difference between their method and the present scheme is the parameter continuation used for globalization; more details on this globalization can be found in [13].

The Newton update equation corresponding to (3) is

$$\begin{bmatrix} 0 & \mathcal{L}_{\boldsymbol{\psi},\mathbf{u}} & \mathcal{L}_{\boldsymbol{\psi},\mathbf{x}} \\ \mathcal{L}_{\mathbf{u},\boldsymbol{\psi}} & \mathcal{L}_{\mathbf{u},\mathbf{u}} & \mathcal{L}_{\mathbf{u},\mathbf{x}} \\ \mathcal{L}_{\mathbf{x},\boldsymbol{\psi}} & \mathcal{L}_{\mathbf{x},\mathbf{u}} & \mathcal{L}_{\mathbf{x},\mathbf{x}} \end{bmatrix} \begin{pmatrix} \Delta\boldsymbol{\psi}_k \\ \Delta\mathbf{u}_k \\ \Delta\mathbf{x}_k \end{pmatrix} = - \begin{bmatrix} \mathcal{L}_{\boldsymbol{\psi}} \\ \mathcal{L}_{\mathbf{u}} \\ \mathcal{L}_{\mathbf{x}} \end{bmatrix}, \tag{4}$$

where the subscript k denotes the current Newton iteration. Solving (4) with a direct method is usually impractical for large-scale PDE-constrained optimization problems; therefore, the approach adopted in LNKS is to use an inexact-Newton-Krylov approach. An advantage of using an inexact-Newton method [7] is that the KTT system can be solved approximately and inexpensively during the early Newton iterations

A Krylov-based approach avoids the need to form the KKT matrix explicitly, since Krylov methods use matrix-vector products. However, to be effective, Krylov-iterative solvers must be preconditioned. We adopt the preconditioner $\tilde{P}_2$ from [2], which was found to be efficient in terms of CPU time. This preconditioner approximates the full-space Hessian by dropping second-order derivatives, with the exception of $\mathcal{L}_{\mathbf{x},\mathbf{x}}$, which is replaced with a L-BFGS quasi-Newton approximation. In addition, the Jacobian $\boldsymbol{\mathcal{R}}_{\mathbf{u}}$ is replaced with a suitable preconditioner A.

2.3 Reduced-space Approaches

Full-space methods for PDE-constrained optimization are efficient [11, 2, 12], often requiring only a few multiples, typically O(10), of the PDE solution cost. Neverthe-less, full-space methods have significant disadvantages: lack of appropriate optimization

libraries; inability to leverage specialized globalization strategies, and; potentially prohibitive memory requirements. These disadvantages motivate reduced-space methods, which we review in this section.

For a valid set of control variables, the state equations will be invertible. Thus, we can invoke the implicit function theorem and define the state variables in terms of the control variables: $\mathbf{u} = \mathbf{u}(\mathbf{x})$. Consequently, the optimization problem (1) reduces to

$$\text{minimize} \quad \mathcal{J}(\mathbf{x}, \mathbf{u}(\mathbf{x})), \quad \mathbf{x} \in \mathbb{R}^m. \tag{5}$$

The first-order optimality conditions for a solution of (5) are the same as (1); after all, they solve the same problem. The difference is, in the reduced formulation, the primal and adjoint PDEs (equations (3a) and (3b)) must be solved at each optimization iteration.

In other words: in reduced-space formulations the optimization algorithm is responsible for satisfying (3c), while the user must satisfy (3a) and (3b). The advantage of this approach is that efficient software libraries are usually available to solve the primal and adjoint PDEs; typically, these libraries are parallel and have specialized globalizations tuned to their discipline. The disadvantage is the added cost of accurately solving the state and adjoint equations early in the optimization process.

We now turn to the problem of solving the first-order condition (3c) for $\mathbf{x}$. As in the full-space, we begin with Newton's method. Linearizing about the current design, $\mathbf{x}_k$, we find the Newton-update equation

$$H_k \Delta \mathbf{x}_k = -\mathbf{g}_k, \tag{6}$$

where $H_k \equiv (\partial \mathbf{g}/\partial \mathbf{x})_k$ is the reduced Hessian evaluated at $\mathbf{x}_k$, and $\mathbf{g}_k$ is the reduced gradient evaluated at $\mathbf{x}_k$. One of the challenges for reduced-space formulations is capturing the second-order information contained in H_k. This is because the reduced Hessian is the total derivative of the gradient with respect to $\mathbf{x}$, so variations in $\mathbf{u}$ and $\boldsymbol{\psi}$ must be accounted for

Quasi-Newton methods are a popular and successful class of algorithms that approximate the Hessian, thereby circumventing the need to compute the total derivative of $\mathbf{g}$. For quasi-Newton methods the Newton-update equation is replaced with

$$B_k \Delta \mathbf{x}_k = -\mathbf{g}_k, \tag{7}$$

where B_k is a quasi-Newton approximation to H_k. In this work, we consider the limited-memory BFGS quasi-Newton method [17] globalized with a strong-Wolfe-type line search algorithm [8]. For the Armijo sufficient-decrease condition we use the parameter $c_1 = 10^{-6}$, and for the curvature condition we use $c_2 = 0.999$; see [20] for the definition of these parameters.

Quasi-Newton methods are simple and effective for many problems; however, they are not necessarily suited to large-scale design spaces. For large problems, we must often resort to limited-memory quasi-Newton methods; while BFGS has a superlinear asymptotic

convergence rate [20], its limited-memory variant has only a linear rate of convergence [17]. In addition, during the early stages of optimization the quasi-Newton approximation may not capture curvature accurately, and this can lead to many subiterations in the line search or many trust-region radius updates. For these reasons, the computational cost of quasi-Newton methods typically grows with problem size; linear scaling is not unusual.

We want to retain the modularity of the reduced-space approach, but with algorithmic scaling that does not grow with problem size. The solution pursued here is to apply an inexact-Newton-Krylov strategy in the reduced-space [14, 4]. The inexact-Newton approach replaces the solution of (6) with the condition that $\Delta \mathbf{x}_k$ must satisfy

$$\|H_k \Delta \mathbf{x}_k + \mathbf{g}_k\| \leq \eta_k \|\mathbf{g}_k\|, \tag{8}$$

where η_k is the so-called forcing parameter and $\|\cdot\|$ denotes the L2 norm. To achieve superlinear convergence, we use $\eta_k = 0.1 \min\left[1, \left(\|\mathbf{g}_k\|/\|\mathbf{g}_0\|\right)^{\frac{1}{2}}\right]$ [7]. To avoid oversolving the linear problem when the iterates are near the desired nonlinear tolerance, we set $\eta_k \leftarrow \max\left(\eta_k, \tau\|\mathbf{g}_0\|/\|\mathbf{g}_k\|\right)$, where τ is the nonlinear tolerance.

Similar to the full-space approach, using a Krylov method avoids the need to form the Hessian explicitly; only Hessian-vector products are required. The computation of the Hessian-vector products plays an important role in the efficiency of reduced-space INK and is discussed in detail in the following section.

To globalize the reduced-space INK algorithm, we use the Steihaug-Toint variant of the conjugate-gradient method [24, 25] in a standard trust-region framework [6]. In the context of optimization, limited-memory BFGS has been shown to be an effective preconditioner for Krylov iterative methods [19], and this is the preconditioner adopted here.

3 Inexact Hessian-vector Products & Second-order Adjoints

The Hessian-vector products needed in reduced-space INK methods can be computed using second-order adjoints [26]. By defining a new functional, $\mathbf{g}^T\mathbf{w}$, where $\mathbf{w} \in \mathbb{R}^m$ is an arbitrary vector, and including both the state and (first-order) adjoint equations as constraints, one can show that the Hessian-vector product is given by [2, 14, 13]

$$H\mathbf{w} = \mathbf{g}_{\mathbf{x}}^T \mathbf{w} + \boldsymbol{\lambda}^T P \mathcal{R}_{\mathbf{x}}(\mathbf{x}, \mathbf{u}) + \mathbf{v}^T \mathcal{S}_{\mathbf{x}}(\mathbf{x}, \mathbf{u}, \boldsymbol{\psi}), \tag{9}$$

where $\mathcal{S}$ denotes the first-order adjoint residual

$$\mathcal{S}(\mathbf{x}, \mathbf{u}, \boldsymbol{\psi}) \equiv \mathcal{R}_{\mathbf{u}}^T P \boldsymbol{\psi} + \mathcal{J}_{\mathbf{u}}^T. \tag{3b}$$

Note that the partial derivatives with respect to $\mathbf{x}$ in (9) treat $\mathbf{u}$, $\boldsymbol{\psi}$, $\boldsymbol{\lambda}$, and $\mathbf{v}$ as constants (i.e. these are not total derivatives).

The variables $\mathbf{v} \in \mathbb{R}^n$ and $\boldsymbol{\lambda} \in \mathbb{R}^n$ are the second-order adjoint variables of the functional $\mathbf{g}^T\mathbf{w}$ corresponding to the primal and adjoint equations, respectively. The second-

order adjoints satisfy the equations (see, for example, [13])

$$P\mathcal{R}_\mathbf{u}\mathbf{v} = -\mathbf{g}_\psi^T\mathbf{w}, \tag{10}$$

$$\mathcal{R}_\mathbf{u}^T P\boldsymbol{\lambda} = -\mathbf{g}_\mathbf{u}^T\mathbf{w} - \mathcal{S}_\mathbf{u}^T\mathbf{v}. \tag{11}$$

The assembly and solution of the second-order adjoint equations (10) and (11) deserve some remarks.

- The system matrix of (10) is the Jacobian of the primal equations, and the system matrix of (11) is the transposed Jacobian. Most adjoint-based optimization algorithms are capable of solving for systems involving $\mathcal{R}_\mathbf{u}^T$, and adapting these algorithms to solve systems involving $\mathcal{R}_\mathbf{u}$ is straightforward.

- The right-hand side of the first adjoint system simplifies to

$$-\mathbf{g}_\psi^T\mathbf{w} = -\mathcal{R}_\mathbf{x}^T P\mathbf{w}.$$

  This term can be computed in the same way as the second term in the reduced gradient (3c).

- The right-hand side vector of the adjoint system for $\boldsymbol{\lambda}$ involves second derivatives. These terms amount to direction derivatives and can easily be computed using finite-difference approximations, the complex-step method, or algorithmic differentiation. See the appendix of [13] for details.

The Hessian-vector product involves the solution of the two second-order adjoint equations (10) and (11). These equations are typically solved using iterative methods, which suggests that we might reduce computational cost by sacrificing accuracy. In other words, can we compute inexact Hessian-vector products rather than exact products?

The use of inexact Hessian-vector products is entirely appropriate in the context of an inexact-Newton method: why compute accurate products when we only want an approximate solution anyway? Indeed, the analysis of Simoncini and Szyld [23] indicates that the accuracy of matrix-vector products can be relaxed provided the Krylov basis remains orthogonal, which is the case for methods like GMRES [22] and FGMRES [21]. On the other hand, inexact products can pose convergence problems when the Krylov basis is not explicitly orthogonalized, e.g. in the CG method [10, 23], although no such problems were observed in the present study.

Let $\widetilde{H_k\mathbf{w}}$ be the inexact Hessian-vector product (9) computed using the iteratively-solved second-order adjoints $\tilde{\mathbf{v}}$ and $\tilde{\boldsymbol{\lambda}}$. Then the error in the Hessian-vector product satisfies

$$\|H_k\mathbf{w} - \widetilde{H_k\mathbf{w}}\| \le \delta_\lambda\|\mathcal{R}_\mathbf{u}^{-1}\mathcal{R}_\mathbf{x}\| + \delta_v\|P^{-1}\mathcal{R}_\mathbf{u}^{-T}(\mathcal{S}_\mathbf{x} + \mathcal{S}_\mathbf{u}\mathcal{R}_\mathbf{u}^{-1}\mathcal{R}_\mathbf{x})\|,$$

where δ_λ and δ_v are bounds on the second-order adjoint residuals for $\tilde{\boldsymbol\lambda}$ and $\tilde{\mathbf{v}}$, respectively, and are defined by

$$\|P\boldsymbol{\mathcal{R}}_{\mathbf{u}}\tilde{\mathbf{v}} + \mathbf{g}_\psi^T\mathbf{w}\| \le \delta_v = \frac{1}{2}\eta_k\|\mathbf{g}_k\|, \tag{12}$$

$$\|\boldsymbol{\mathcal{R}}_{\mathbf{u}}^T P\tilde{\boldsymbol\lambda} + \mathbf{g}_{\mathbf{u}}^T\mathbf{w} + \boldsymbol{\mathcal{S}}_{\mathbf{u}}^T\tilde{\mathbf{v}}\| \le \delta_\lambda = \frac{1}{2}\eta_k\|\mathbf{g}_k\|. \tag{13}$$

Bounds for $\|\boldsymbol{\mathcal{R}}_{\mathbf{u}}^{-1}\boldsymbol{\mathcal{R}}_{\mathbf{x}}\|$ and $\|P^{-1}\boldsymbol{\mathcal{R}}_{\mathbf{u}}^{-T}(\boldsymbol{\mathcal{S}}_{\mathbf{x}} - \boldsymbol{\mathcal{S}}_{\mathbf{u}}\boldsymbol{\mathcal{R}}_{\mathbf{u}}^{-1}\boldsymbol{\mathcal{R}}_{\mathbf{x}})\|$ are also required and are more difficult to estimate. For this preliminary work, trial-and-error was used to determine that these terms are O(1) for the problem considered below; future work will investigate a priori methods of bounding these terms.

Based on the bounds δ_v and δ_λ, as well as the O(1) estimates for the remaining terms, we have

$$\|H_k\mathbf{w} - \widetilde{H_k\mathbf{w}}\| \lesssim \eta_k\|\mathbf{g}_k\|.$$

Thus, the approximate solution of the second-order adjoints, based on (12) and (13), is such that the accuracy of the inexact Hessian-vector products is comparable to the accuracy of the linear solve.

4 Results

In this section we investigate the performance of the reduced-space INK algorithm and compare its performance with that of the reduced-space quasi-Newton (QN) method and the full-space LNKS algorithm.

Our model problem for the investigations will be the inverse design of an inviscid nozzle flow. The PDE constraint for this problem is the quasi-one-dimensional Euler equations, given by

$$\frac{\partial \mathcal{F}}{\partial x} - \mathcal{G} = 0, \qquad \forall\, x \in [0, 1], \tag{14}$$

where the flux and source are

$$\boldsymbol{\mathcal{F}} = \left(\rho u A, \ (\rho u^2 + p)A, \ u(e + p)A\right)^T, \qquad \text{and} \qquad \boldsymbol{\mathcal{G}} = \left(0, \ p\tfrac{dA}{dx}, \ 0\right)^T,$$

respectively. The state variables are density (ρ), momentum per unit volume (ρu), and energy per unit volume (e). Pressure, which also appears in the Euler equations, is determined using the ideal-gas equation of state: $p = (\gamma - 1)(e - \frac{1}{2}\rho u^2)$. Finally, $A = A(x)$ denotes the spatially varying nozzle area, which, when discretized, will become our control variable. The boundary values are provided by the exact solution, which is determined using the Mach relations. The exact solution is based on a stagnation temperature of 300K and stagnation pressure of 100 kPa. The critical nozzle area is $A^* = 0.8$ and the gas constant is 287 J/(kg K). In the implementation, the equations and variables have been nondimensionalized using the density and sound speed at the inlet, $x = 0$.

The Euler equations (14) are discretized using a summation-by-parts finite-difference scheme [16]. In particular, the derivatives are approximated using a third-order accurate diagonal-norm operator, and the boundary conditions are imposed weakly using SAT penalty terms [9, 5]. To stabilize the discrete equations, we add scalar third-order accurate artificial dissipation [18].

For the reduced-space formulations, the discretized Euler equations are solved using a Newton-GMRES algorithm [15]. The GMRES [22] Krylov solver is preconditioned using an LU factorization of a first-order accurate discretization that is based on nearest-neighbours and first-order scalar dissipation. The linearized forward problem (10) is also solved using GMRES and the same preconditioner. The adjoint problems are solved using GMRES with the Jacobian-vector products and preconditioners replaced with their transposed versions.

The nozzle area $A(x)$ is discretized using cubic B-splines with open uniform knot vectors. The area is fixed at the ends of the nozzle such that $A(0) = 2$ and $A(1) = 1.5$. The control variables consist of the interior B-spline control points. To avoid confusion between the design variables and the x-coordinate, we will use $\mathbf{A}$ to denote the vector of (interior) B-spline control points. In all cases, the initial design $\mathbf{A}_0$ corresponds to the set of control points that produce the linearly varying area $A(x) = 2 - 0.5x$. The target nozzle area is a cubic function of x that passes through the fixed inlet and outlet areas and has a local minimum at $x = 0.5$ given by $A(0.5) = 1$.

In summary, the optimization problem is

$$\begin{aligned} \text{minimize} \quad & \mathcal{J}(\mathbf{A}, \mathbf{u}) = \int_0^1 \frac{1}{2}(p(\mathbf{u}) - p_{\text{targ}})^2 \, dx, \quad \mathbf{A} \in \mathbb{R}^m, \ \mathbf{u} \in \mathbb{R}^{3n}, \\ \text{subject to} \quad & \mathcal{R}(\mathbf{A}, \mathbf{u}) = \mathbf{0}, \end{aligned} \tag{15}$$

where $\mathbf{u}$ denotes the vector of state variables (ρ, ρu, and e) at each of the n nodes, and $\mathcal{R}(\mathbf{A}, \mathbf{u}) = \mathbf{0}$ denotes the discretized Euler equations. The target pressure p_{targ} is found by solving for $\mathbf{u}$ using the target nozzle area in the Euler equations.

4.1 Dynamic versus fixed tolerance for the inexact-Hessian-vector products

To demonstrate the impact of inexact Hessians on the reduced-space INK algorithm, we solve (15) with the second-order adjoint equations satisfying either 1) a fixed (relative) tolerance of 10^{-6} or 2) the dynamic tolerances (12) and (13).

Table 1 lists the computational cost for these two approaches over a range of design-variable dimensions. The cost is measured in terms of equivalent PDE solutions required to satisfy $\|\mathbf{g}(\mathbf{x}_k)\| \leq \tau \|\mathbf{g}(\mathbf{x}_0)\|$, where $\tau = 10^{-3}$. Specifically, the total number of PDE preconditioner calls (i.e. applications of $(LU)^{-1}$) used during the optimization is divided by the number of preconditioner calls to solve the PDE on the initial geometry.

The results show that, on average, the dynamic tolerance reduces the computational cost by 40.5% relative to the fixed tolerance. This illustrates the importance of using inexact Hessians in reduced-space INK algorithms.

Table 1: Number of equivalent PDE solutions required when using dynamic and static tolerances in second-order adjoint solves.

	number of design variables				
	20	30	40	50	60
fixed tolerance	102.7	103.5	107.5	113.8	108.3
dynamic tolerance	67.4	58.5	60.5	65.5	66.8
improvement (%)	34.3	43.5	43.8	42.5	38.4

4.2 Comparison of Optimization Methods

We now compare the reduced-space INK algorithm with the full-space LNKS algorithm and the reduced-space quasi-Newton algorithm. Figure 1 plots the computational cost of the three algorithms versus the design-variable dimension. Computational cost is measured using the number of equivalent PDE solves, as defined earlier. The optimizations are terminated when the relative reduced-gradient norm is below $\tau = 10^{-3}$. In the case of LNKS, the PDE and adjoint residual norms must be below 10^{-6} their initial values.

As expected, the quasi-Newton algorithm has a strong dependence on the number of design variables. In contrast, the two Newton-based algorithms have much weaker dependence on the dimension of $\mathbf{A}$. Moreover, even for this relatively small problem, there is a clear advantage to using inexact-Newton rather than quasi-Newton optimization; the computational cost is between 4 and 6 times lower using INK, and between 15 and 17 times lower using LNKS.

Comparing reduced-space INK with LNKS, we observe a factor of 3 to 4 reduction in cost using the full-space algorithm. What is not shown in the plot is the robustness issues associated with LNKS. Considerable parameter tuning was necessary to obtain convergence with LNKS, so there is trade-off between robustness and speed when moving from the reduced-space to the full-space. We argue that a factor of 3 to 4 is compensated for by the time needed to find suitable parameters in the full-space approach.

5 Summary and Discussion

We have shown that reduced-space inexact-Newton-Krylov (INK) algorithms offer an attractive compromise between reduced-space quasi-Newton and full-space Newton-type methods. For the nozzle-flow inverse design problem, reduced-space INK was 4 to 6 times faster than the quasi-Newton algorithm and was much less sensitive to the number of design variables. The LNKS full-space algorithm was found to be the most efficient scalable algorithm, but this efficiency comes at the price of robustness.

In general, engineers have focused on reduced-space quasi-Newton and full-space Newton methods. The results presented here indicate that reduced-space INK algorithms should be investigated more broadly, since they offer a scalable route to solving large-
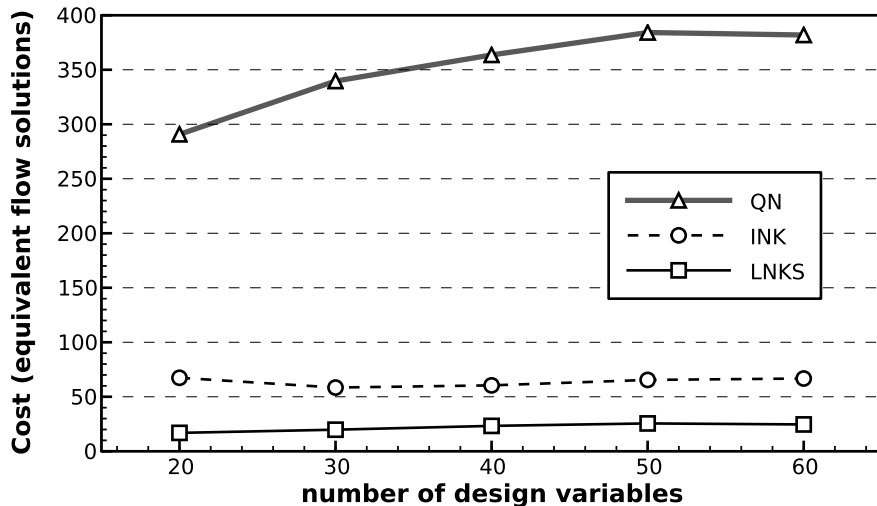
Figure 1: Cost of the nozzle-area optimization based on number of equivalent flow solutions.

scale optimization problems. Moreover, they require limited intrusion into existing PDE solvers and provide increased robustness relative to full-space algorithms.

Inexact-Hessian-vector products play an important role in the reduced-space INK algorithm presented here. The products are computed using second-order adjoints with dynamic tolerances. The use of dynamic tolerances was shown to reduce computational cost by approximately 40% relative to a fixed tolerance. We note that inexact-Hessian-vector products may pose challenges for traditional optimization algorithms that assume symmetry of the Hessian. Our current work is focused on developing optimization algorithms that are robust in the presence of inexactness in the Hessian.

REFERENCES

[1] V. Akçelik, G. Biros, O. Ghattas, J. Hill, D. Keyes, and B. van Bloemen Waanders, *Parallel Algorithms for PDE-Constrained Optimization*, in Parallel Processing for Scientific Computing, M. A. Heroux, P. Raghavan, and H. D. Simon, eds., Society for Industrial and Applied Mathematics, Jan. 2006, ch. 16, pp. 291–322.

[2] G. Biros and O. Ghattas, *Parallel Lagrange-Newton-Krylov-Schur methods for PDE-constrained optimization. part I: the Krylov-Schur solver*, SIAM Journal on Scientific Computing, 27 (2005), pp. 687–713.

[3] ——, *Parallel Lagrange-Newton-Krylov-Schur methods for PDE-constrained optimization. part II: the Lagrange-Newton solver and its application to optimal control of steady viscous flows*, SIAM Journal on Scientific Computing, 27 (2005), pp. 714–739.

[4] A. BORZÌ AND V. SCHULZ, *Computational Optimization of Systems Governed by Partial Differential Equations*, Society for Industrial and Applied Mathematics, Jan. 2011.

[5] M. H. CARPENTER, D. GOTTLIEB, AND S. ABARBANEL, *Time-stable boundary conditions for finite-difference schemes solving hyperbolic systems: methodology and application to high-order compact schemes*, Journal of Computational Physics, 111 (1994), pp. 220–236.

[6] A. R. CONN, N. I. M. GOULD, AND P. L. TOINT, *Trust Region Methods*, Society for Industrial and Applied Mathematics, Jan. 2000.

[7] R. S. DEMBO, S. C. EISENSTAT, AND T. STEIHAUG, *Inexact Newton methods*, SIAM Journal on Numerical Analysis, 19 (1982), pp. 400–408.

[8] R. FLETCHER, *Practical methods of optimization*, A Wiley-Interscience Publication, Wiley, second ed., 2000.

[9] D. FUNARO AND D. GOTTLIEB, *A new method of imposing boundary conditions in pseudospectral approximations of hyperbolic equations*, Mathematics of Computation, 51 (1988), pp. 599–613.

[10] G. H. GOLUB AND Q. YE, *Inexact Preconditioned Conjugate Gradient Method with Inner-Outer Iteration*, SIAM Journal on Scientific Computing, 21 (1999), pp. 1305–1320.

[11] E. HABER AND U. M. ASCHER, *Preconditioned all-at-once methods for large, sparse parameter estimation problems*, Inverse Problems, 17 (2001), pp. 1847–1864.

[12] S. B. HAZRA, *Direct treatment of state constraints in aerodynamic shape optimization using simultaneous pseudo-time-stepping*, AIAA Journal, 45 (2007), pp. 1988–1997.

[13] J. E. HICKEN AND J. J. ALONSO, *Comparison of Reduced- and Full-space Algorithms for PDE-constrained Optimization*, in 51st AIAA Aerospace Sciences Meeting, no. AIAA–2013–1043, Grapevine, Texas, United States, Jan. 2013.

[14] M. HINZE, R. PINNAU, M. ULBRICH, AND S. ULBRICH, *Optimization with PDE constraints*, Springer, 2009.

[15] D. E. KEYES, *Aerodynamic applications of Newton-Krylov-Schwarz solvers*, in Proceedings of the 14th International Conference on Numerical Methods in Fluid Dynamics, New York, 1995, Springer, pp. 1–20.

[16] H. O. KREISS AND G. SCHERER, *Finite element and finite difference methods for hyperbolic partial differential equations*, in Mathematical Aspects of Finite Elements in Partial Differential Equations, C. de Boor, ed., Mathematics Research Center, the University of Wisconsin, Academic Press, 1974.

[17] D. C. LIU AND J. NOCEDAL, *On the limited memory BFGS method for large scale optimization*, Mathematical Programming, 45 (1989), pp. 503–528.

[18] K. MATTSSON, M. SVÄRD, AND J. NORDSTRÖM, *Stable and accurate artificial dissipation*, Journal of Scientific Computing, 21 (2004), pp. 57–79.

[19] J. L. MORALES AND J. NOCEDAL, *Automatic Preconditioning by Limited Memory Quasi-Newton Updating*, SIAM Journal on Optimization, 10 (2000), pp. 1079–1096.

[20] J. NOCEDAL AND S. J. WRIGHT, *Numerical Optimization*, Springer–Verlag, Berlin, Germany, second ed., 2006.

[21] Y. SAAD, *A flexible inner-outer preconditioned GMRES algorithm*, SIAM Journal on Scientific and Statistical Computing, 14 (1993), pp. 461–469.

[22] Y. SAAD AND M. H. SCHULTZ, *GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM Journal on Scientific and Statistical Computing, 7 (1986), pp. 856–869.

[23] V. SIMONCINI AND D. B. SZYLD, *Theory of Inexact Krylov Subspace Methods and Applications to Scientific Computing*, SIAM Journal on Scientific Computing, 25 (2003), pp. 454–477.

[24] T. STEIHAUG, *The Conjugate Gradient Method and Trust Regions in Large Scale Optimization*, SIAM Journal on Numerical Analysis, 20 (1983), pp. 626–637.

[25] P. L. TOINT, *Towards an efficient sparsity exploiting Newton method for minimization*, 1981, pp. 57–88.

[26] Z. WANG, I. M. NAVON, F. X. DIMET, AND X. ZOU, *The second order adjoint analysis: Theory and applications*, Meteorology and Atmospheric Physics, 50 (1992), pp. 3–20.

ERROR ESTIMATION AND ADAPTIVITY FOR TURBULENT FLOW

JOHAN HOFFMAN[*]

[*]Computational Technology Laboratory
Department for High Performance Computing and Visualization
KTH Royal Institute of Technology
SE-10044 Stockholm, Sweden
e-mail: jhoffman@kth.se, www.csc.kth.se/ jhoffman

Key words: Adaptive finite element method, transient problem, turbulence

Abstract. We present recent advances on a posteriori error estimation and adaptivity for turbulent flow, including deforming domains, fluid-structure interaction and high performance computing implementation in the open source software Unicorn [1]. Fluid-structure interaction is formulated in a Unified Continuum framework [2], and turbulent fluid flow is modeled by G2 implicit large eddy simulation with residual based stabilization modeling the effect of subgrid scales, and with skin friction boundary conditions modeling turbulent boundary layers. Examples are presented, including applications to aerodynamics, aeroacoustics and biomedicine.

REFERENCES

[1] J.Hoffman, J.Jansson and M.Stckli. Unified continuum modeling of fluid-structure interaction. *Mathematical Models and Methods in Applied Sciences* (2011) **21(3)**:491-513.

[2] J.Hoffman, J.Jansson, R.Vilela de Abreu, C.Degirmenci, N.Jansson, K.Mller, M.Nazarov and J.Hiromi Sphler,. Unicorn: parallel adaptive finite element simulation of turbulent flow and fluid-structure interaction for deforming domains and complex geometry. *Computer and Fluids,* (in press).

EFFECT OF PRESTRESSED EMBOSSMENTS UNDER VARIOUS TYPES OF LOADING USING FEM ANALYSIS

J. HOLOMEK[*], M. BAJER [†] AND J. BARNAT[†]

[*†]Institute of Metal and Timber Structures
Faculty of Civil Engineering
Brno University of Technology
Veveří 331/95, 602 00 Brno, Czech Republic
e-mail: holomek.j@fce.vutbr.cz, web page: http://www.kdk.fce.vutbr.cz/

Key words: Embossment, composite slab, slip-block test, vacuum loading, numerical model, FE analysis.

Abstract. Composite slabs of trapezoidal steel sheeting and concrete are widely used for ceiling structures in all types of buildings. Prestressed embossments can serve as a meaning to ensure the composite action without need of other load bearing reinforcement. Design of the composite slab according to valid codes is governed by semi-empirical methods using bending tests to obtain unique parameters for each type of sheeting. Smaller and less expensive shear slip-block tests are considered as a meaning to obtain shear characteristics of the sheeting, which could be used for design of the sheeting. In our laboratory bending tests with different load arrangements and shear test were performed. The key role in load bearing capacity of the slab has the shear connection between steel sheeting and concrete. The FE (Finite Element) modelling of the connection must deal with a complicated geometry of the embossments and several possible failure mechanisms. Three types of numerical models are being created using Atena software. 2D and 3D models of shear test serves to describe the failure mechanism of embossments. Bending model of one rib over the whole span serves to include effects curvature due to bending. Influence of elevated temperature on shear bearing capacity is observed as well. The models are being set, calibrated and compared regarding data from the experiments performed in our laboratory.

1 INTRODUCTION

Composite slab with prestressed embossments presents a suitable solution for horizontal structures. Its load bearing capacity is mainly determined by its horizontal shear bearing capacity. The bearing capacity is determined using four-point bending tests. Research works are carried out on possibilities of determination of shear characteristics using small-scale slip-block tests, which present a less-expensive alternative to the bending tests [1]. Unfortunatelly final load bearing capacity of the sheeting is influenced by bending effects and loading arrangenent, which cannot be covered by shear tests.

Numerical models presented in this paper has its aim in both, modelling the behaviour of sheeting in small-scale shear tests and modelling of the bending effects in bending. Models are being set and calibrated regarding data from the tests performed in our laboratory. The simulation is realized using Atena software for computing because of its advanced nonlinear

concrete analysis possibilities. GiD is used for pre/post processing, because of the complicated geometry of the sheeting with embossments. 2D model is used for correct setting of shear parameters of the task and observe the effect of local bending of the sheeting. 3D model of one embossment is set to observe spacial deformation of sheeting. Finally a bending model of one rib of the slab is modelled to include effects of curvature.

The tests performed in our laboratory consists of small-scale shear test and bending tests using vacuum loading developed by prof. Melcher [2]. Finally an effect of elevated temperature on the shear bearing capacity is modelled.

All the tests and models mentioned in this paper are related to the type of sheeting Cofraplus 60 (galvanized surface, 1 mm thickness) and the thickness of the slab 110 mm.

2   SHEAR TEST

2.1 Tests performed in laboratory

A special loading device has been constructed in our laboratory to perform shear slip-block tests. The specimens are bolted to a base plate in the overlapping part of the sheeting. Two jacks are installed to apply loads. One of them is placed vertically to produce a clamping force and the second is placed horizontally to push the concrete block out of the sheeting. A roller bearing is placed between the vertical jack and concrete block to allow sufficiently horizontal movement of the block. The specimens have width of two waves of the sheeting, which allows placing the horizontal load into the centroid of the sheeting cross-section. Mutual slip between steel sheeting and concrete block is measured.

When the slip occurs the sheeting starts to deform and bends in the area of the embossments. The concrete stays almost unimpaired; there is only small abbrasion around the indentations and longitudinal edges of sheeting.

| (a) | (b) |
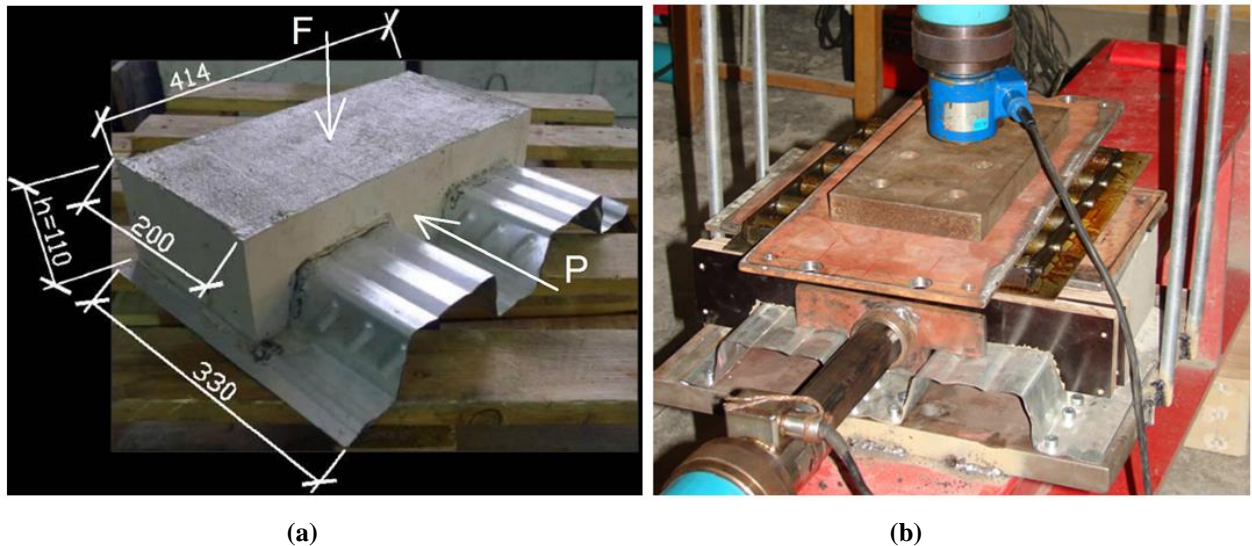
Figure 1: (a) Specimens dimensions (b) roller bearing placed on the top of the concrete block [3].

2.2 Numerical modelling

2.2.1   2D model

2D model of one embossment was created. The embossments resist to mutual slip by its stiffness in bending in combination with contact properties of the interface between steel and concrete. The modelled problem is therefore a combination of contact and bending tasks.

Length of the model is 102 mm, height of concrete is 50 mm and height of the embossment is 3 mm. The inner spacing between supports is 82 mm. Thickness of all the materials is 10 mm. Mesh is refined over the area of steel sheeting to form enough layers of elements for calculation of bending (at least 6 over the hieght). A complementary line is created between steel and concrete and is bonded with the line of the concrete to enable turning on a moving gap function. The complementary line is also an advantage because of possibility of using larger elements for concrete surface.

Figure 2: 2D model: deformed sheeting corresponding to slip 2 mm; displayed stress in longitudinal axes [kPa].
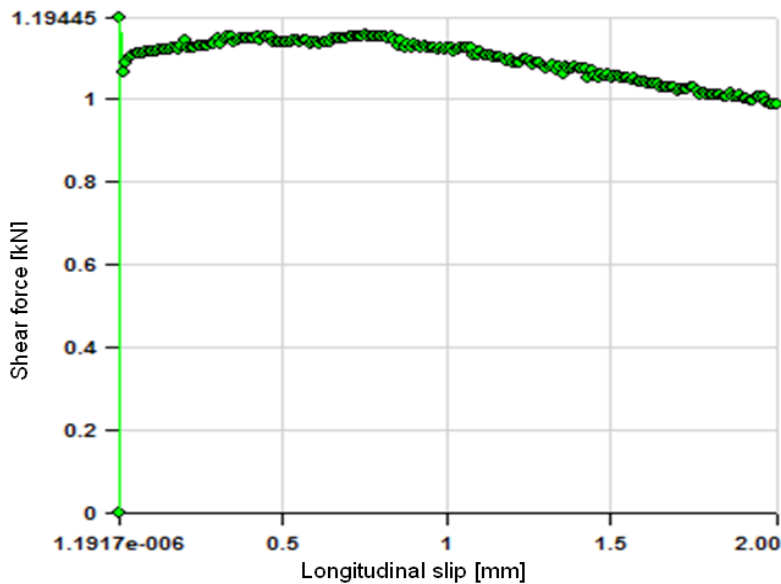
Figure 3: Load-slip dependence of 2D model.

Newton-Rhapson method used to compute in shear test models [4]. The loading is applied in two intervals. In the first interval vertical displacement 1e-7 m is assigned to the top line of the concrete block. In the second interval longitudinal displacement is assigned to left line of the concrete block. Friction was estimated to be 0.2, cohesion 1.0 Mpa and tensile strength 0.4 Mpa [5].

2.2.2   3D model of shear test

Because of the complexity of the embossment action one cannot describe the behaviour of the embossment in real test by the 2D model idealisation. Therefore 3D model of the shear test is being created in the length of one embossment of concrete and overlapping steel sheeting. The final shear bearing capacity of the embossment is influenced by distance of its end from longitudinal edge of the sheeting [6], as show the peak values of stress around the ends of the embossment in fig. 5.

(a)   **(b)**
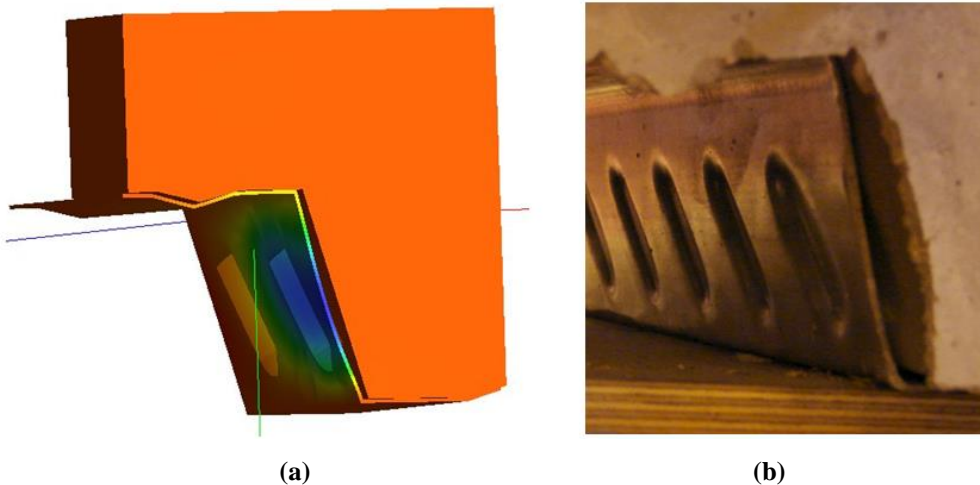
Figure 4 (a) Deformed sheeting in 3D model with displayed displacement in horizontal axes (b) deformed sheeting in real test.
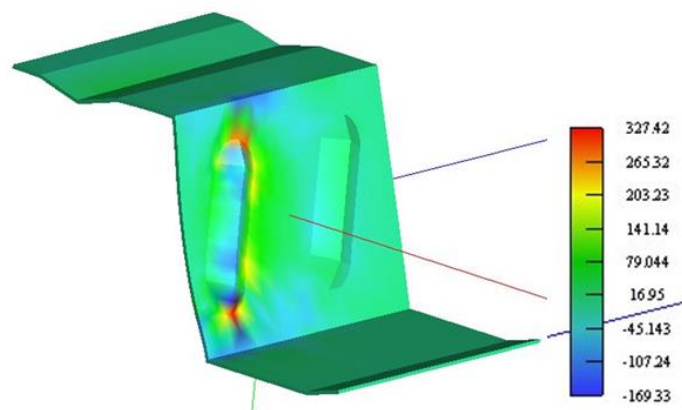
Figure 5 Deformed shape of the steel sheeting with depicted peaks stresses around the ends of embossment.

In order to obtain correct results in bending perpendicular to the plane of the sheeting there is a need to use several layers over the thickness of the sheeting. Atena enables to use Ahmad element developed by reducing of quadratic 3D bricks elements with 20 nodes. These elements include layers inside; therefore its usage significantly reduces required number of elements and time for computation [7].

2.2.3 Combination of thermal and static analysis

Design bearing capacity of the composite slab in fire according EC4 can be determined using theory of plasticity and reduced material properties without specifying the shear bearing capacity at elevated temperatures [8]. Moreover composite slabs behaviour in fire can be transformed to membrane action considering large deflections and proper supporting [9].

Thermal and static analysis was combined on 2D model to observe the sheeting behaviour at high temperatures. At first the thermal analysis was performed to obtain stationary temperature field. The field was then applied to static analysis. The same boundary conditions and material properties were used as in the normal temperature analysis. The temperature loading leads to deformation of the sheeting and subsequently reducing of shear bearing capacity (Fig. 7) in compare to normal temperature analysis (Fig. 3).
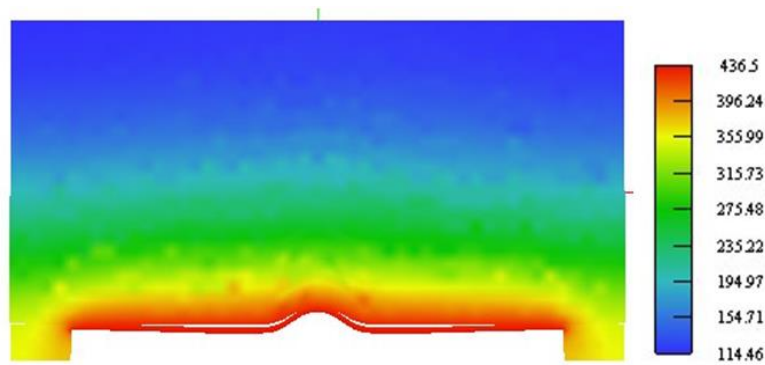
Figure 6 Deformed shape of the steel sheeting subjected to temperature load.
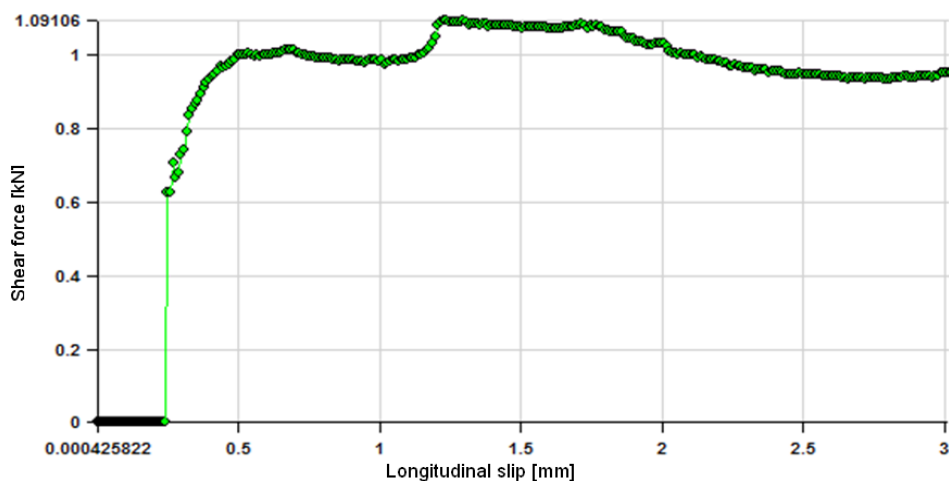
Figure 7 Load-slip dependence of 2D model subjected with temperature load.

Figure 8: Uniformly distributed cracks in concrete after finishing the test and removing of steel sheeting.

3 BENDING TESTS

3.1 Tests performed in laboratory

Ideally uniform area loading was produced using special vacuum loading device [2]. The specimens were simply supported with span 2 m and width 1.08 m. The concrete class was C20/25 and the yield strength of the steeting was 350 MPa. Slabs were loaded by almost 50 kN/m^2 using a common plastic foil. The uniformly distributed load results in uniformly distributed crack pattern over the length of the specimens. The cracks can be seen after finishing the test when the steel sheeting is removed.

3.2 Numerical modelling of bending

Model of one rib as a simply supported beam has been created. Shell elements are used for modelling the sheeting. The bending model is simplified because of the difficulties with detailed modelling of shear behaviour of embossments. The steel sheeting is plain and the effect of embossments is simulated by hardening/softening function of cohesion in Interface material.
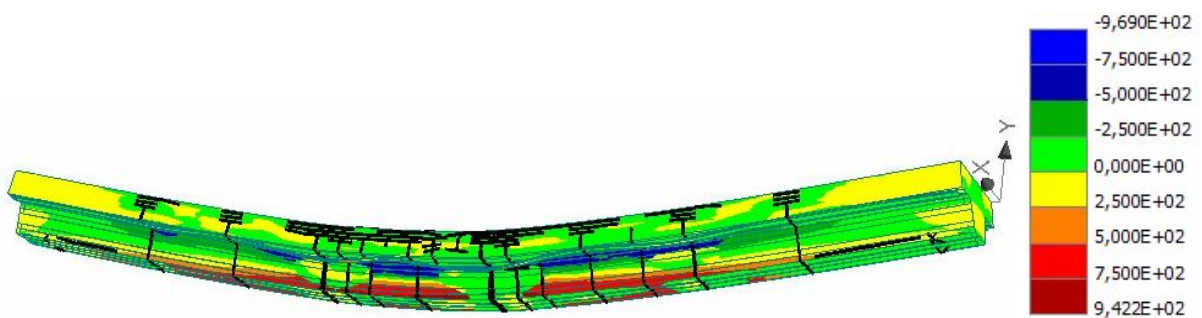
Figure 9 Stress in concrete in longitudinal axes [MPa] with crack pattern.
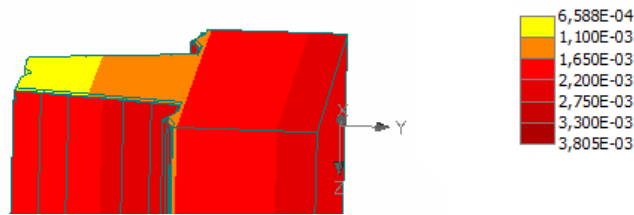
Figure 10 Mutual slip between steel and concrete at the end of the slab; longitudinal displacement [m].

Loading is realized by force applied on the top surface of concrete. Supports are realized by bricks from 3D elastic material. Arc-length method is used for computing.

The resulting bending model crack distribution and end slip development show good accordance with the real results for uniformly distributed load.

4 CONCLUSIONS

The paper presents numerical modelling of composite action of composite slabs with prestressed embossments. Models simulate the real behaviour of sheeting and concrete in slip-block test and bending test. The carried out research lead to the following conclusions:

- Numerical modelling of the embossment shear behaviour can adequately describe the real test behaviour. Both the detailed models and bending models, respectively, can present an effective tool for parametrical studies of sheeting geometry of material properties. However its correct setting is influenced by many factors and must be compared with the tests results.
- Prestressed embossments stiffen the plane parts of steel sheeting and thus present a sufficient tool for ensuring partial composite action. Distance of the ends of the embossments from sheeting longitudinal edges belongs to the most affecting factors of the final shear bearing capacity.
- The thin layer of steel sheeting can be effectively modelled using shell elements. This approach reduces the number of elements needed for analysis and enables to perform bending test simulation. The bending test modelling is realized using plain sheeting and the effect of embossments is substituted by cohesion function. Future works on the models can lead to the inclusion of embossment effect to the bending task.
- Determination of changes in stress state and shear bearing capacity at elevated temperature can be estimated using the FE analysis. Detailed model boundary conditions can be different from those in the real slab; this must be taken into account when interpreting the results.

REFERENCES

[1] M. Crisinel, F. Marimon, "A simplified method for the design of composite slabs," *Journal of Constructional Steel Research,* Vol. **60***,* p. 481 – 491 (2004).

[2] J. Melcher, Full-Scale Testing of Steel and Timber Structures: *Examples and Experience, In Structural Assessment - The Role of Large and Full Scale Testing*, Edited by K.S. Virdi et. al., London: E&FN SPON, 1997, pp. 301 – 308, ISBN 0 419 22490 4

[3] J. Holomek, R. Karásek, M. Bajer and J. Barnat, "Comparison of Methods of Testing Composite Slabs," *An international Journal of Science, Engineering and Technology,* Vol. **67**, p. 620 - 625 (2012).

[4] J. Kala, Z. Kala, "The Interaction of Local Buckling and Stability Loss of a Thin-Walled Column under Compression," *In International Conference of Numerical Analysis and Applied Mathematics (ICNAAM)*, GREECE, 2012, pp. 2074-2077, DOI: 10.1063/1.4756598, ISSN: 0094-243X, ISBN: 978-0-7354-1091-6.

[5] M. Bajer, J. Barnat, "The Glue-Concrete Interface of Bonded Anchors," *Construction and building materials*. 2012. 34(9). p. 267 - 274. ISSN 0950-0618.

[6] M. Ferrer, F. Marimon, M. Crisinel, "Designing cold-formed steel sheets for composite slabs: An experimentally validated FEM approach to slip failure mechanics," *Thin-Walled Structures*, Vol. **44**, Issue 12, p. 1261 – 1271, (2001).

[7] http://www.cervenka.cz/downloads

[8] EN 1994-1-1; 2004, Eurocode 4 – Design of composite steel and concrete structures – Part 1.2: Structural fire design

[9] C. G. Bailey "Membrane action of slab/beam composite floor systems in fire", *Engineering Structures,* Vol. **26,** p. 1691–1703, (2004)

VI International Conference on Adaptive Modeling and Simulation
ADMOS 2013
J. P. Moitinho de Almeida, P. Díez, C. Tiago and N. Parés (Eds)

IMPROVING NUMERICAL EFFICIENCY WITH MODEL REDUCTION AND HIGH-ORDER ADAPTIVE DISCONTINUOUS GALERKIN

A. HUERTA[1], D. MODESTO[1], S. ZLOTNIK[1], E. CUETO[2] AND F. CHINESTA[3]

[1]Laboratori de Càlcul Numèric (LaCàN), Dept. Applied Math.,
E.T.S. de Ingenieros de Caminos,
Universitat Politècnica de Catalunya-BarcelonaTech, Barcelona, Spain
e-mail: antonio.huerta@upc.edu

[2]Aragon Institute of Engineering Research (I3A), Universidad de Zaragoza
Maria de Luna 3, E-50018 Zaragoza, Spain
e-mail: ecueto@unizar.es

[3]GEM UMR CNRS, Ecole Centrale de Nantes
EADS Corporate Foundation International Chair
1 rue de la Noë, BP 92101, F-44321 Nantes cedex 3, France
e-mail: francisco.chinesta@ec-nantes.fr

Abstract.
Despite the impressive progress attained in the last decades by simulation-based engineering sciences, decision-making in engineering design, optimization and control, remains sub-optimal in many fields. Aerospace industry is probably one area where these limitations are more obvious.

On one hand, in a multitude of real engineering design problems a large number of scenarios must be considered and carefully analyzed. This task is very expensive both in specialized man-hours to prepare and analyze data and from a computational point of view. The space of design parameters is, in these cases, too large for an exhaustive exploration. In general, only a small sample of the parametric space is studied. Consequently, these models must be complemented with security coefficients conceived to cover the rest of the parametric space and to include unknown information, the inevitable uncertainty. Thus, in practice, designs remain suboptimal because of the computational complexity related to very rich descriptions of external actions, geometry, materials, processes, etc. In fact, even for cutting edge engineering, real practice imposes methodologies devised more than 30 years ago.

On the other hand, two contradictory goals are nowadays present in every challenging simulation based engineering problem: real-time and high fidelity. In order to speed-up engineering design or to assist decision-making strategies in engineering processes, faster simulations are required. Moreover, in many cases, there is the added restriction: such decision-making tools should run in light computing devices to increase portability, on-site evaluation, or democratize accessibility. Real-time is easier to attain with coarse models or meta-models involving few number of parameters. These requirements usually are

contradictory with high-fidelity simulations. Moreover, users are more demanding and prescribe error bounds on quantities of interest to minimize uncertainty in models and simulations.

These apparently incompatible goals can be integrated by means of a *computational vademecum*. A high-fidelity error-controlled off-line computation produces a solution of the model under consideration for all the possible design scenarios. Then, an on-line post-process, able to run on light computational devices if necessary, is used for fast decision-making purposes.

The Proper Generalized Decomposition (PGD), which relies in the assumption of separated approximations of the solution, has demonstrated its capabilities in dealing with high-dimensional problems. The multidimensional capabilities of this approach opens new possibilities to solve, for instance, problems where material or external parameters are set as additional extra-coordinates of the model. In this framework a general solution is obtained encompassing every solution for any possible value of the parameters, thus, a computational vademecum is produced. Under this rationale, parametric design, optimization of complex problems, uncertainty quantification, simulation-based control, and real-time simulation are seen as a post-process once the off-line strategy has produced the vademecum.

To illustrate the advantages of such an approach a simple shape optimization example will be shown before discussing a practical engineering problem governed by the Helmholtz equation with variable coefficients in an unbounded domain. This problem models harbor wave agitation, which is a primary engineering design challenge. Two major issues are discussed:

1. *Efficient and accurate computations*. This implies a large number of simulation challenges, which include, among others, reproducing the exact geometry to capture the small features that are influential, efficient adaptive approximations, precise high-gradients (shock-capturing) approximations, etc.

2. *Large number of external forcing conditions*. A general solution for the agitation in the harbor is obtained with the incident wavelength and its direction as extra coordinates covering an exhaustive evaluation of all possible scenarios and enabling on-line computations on tablets.
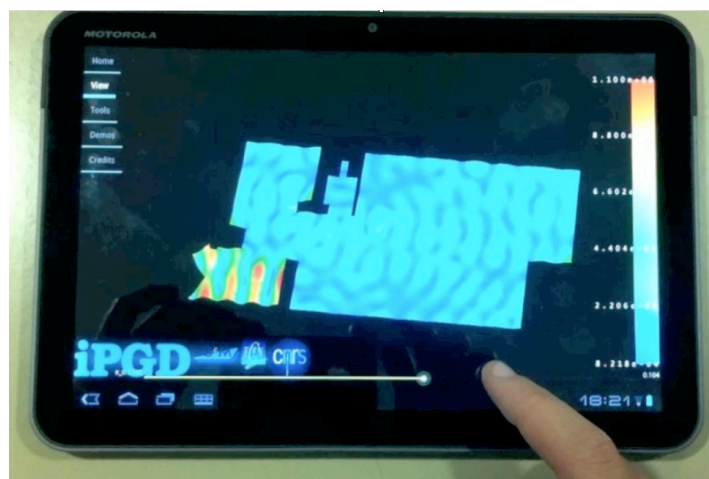
GEOMETRICALLY EXACT KIRCHHOFF-LOVE SHELL MODEL: THEORY AND MESHLESS IMPLEMENTATION

V. IVANNIKOV[*], C. TIAGO[*], J. P. MOITINHO DE ALMEIDA[*] AND P. M. PIMENTA[†]

[*] Instituto Superior Técnico, Universidade Técnica de Lisboa
Av. Rovisco Pais, 1049-001 Lisboa, Portugal
e-mail: {vladimir.ivannikov,carlos.tiago,moitinho}@civil.ist.utl.pt

[†] Polytechnic School at University of São Paulo,
P.O. Box 61548, 05424-970 São Paulo, Brazil
e-mail: ppimenta@usp.br

Abstract. A geometrically exact shell model based on the Kirchhoff-Love theory, where shear deformation is not accounted for, has been developed in the present contribution. Energetically conjugated cross sectional stresses (first Piola-Kirchhoff tensor) and strains (deformation gradient) are defined. Elastic constitutive equations are consistently derived from fully three-dimensional finite strain constitutive models. A genuine plane-stress condition is enforced by vanishing the true mid-surface normal stress. Since only the bending deformation is included in this model no special technique has to be taken into account in order to avoid shear-locking.

Since the variational basis of the formulation requires the use of C1 approximations, the generation of compatible finite elements is not trivial in the present case. In order to overcome this inconvenience, meshless approximations are used. The first-order Generalized Moving-Least Squares Approximation has been proposed. Although it increases the number of degrees-of-freedom per node, its performance and quality of results are clearly superior to the conventional Moving-Least Squares Approximation in this specific class of problems.

Since the approximation does not possess the Kronecker-delta property, the essential boundary conditions are enforced using a hybrid-displacement version of the shell formulation, by means Lagrange multipliers. The corner reactions which naturally arise from the boundary integrals are carefully treated. This issue requires introduction of extra pointwise Lagrange multipliers. Its significant influence on the accuracy of results is demonstrated in some linear examples.

Imposition of the kinematic boundary conditions along the line also requires extra discussion. The proposed theory has no explicit expression for the boundary rotation angle arising on such boundaries and, moreover, this specific quantity may lead to nonsymmetric tangent matrix. Stitching domains along the line by means of Lagrange multipliers makes it possible to apply the proposed theory not only to smooth continuous shells but also to the folded ones.

Initially curved shells are regarded as a stress-free deformed state from a chosen plane reference configuration. The mapping between both configurations allows the exact consideration of the initial configuration.

The complete linearization of the weak form is presented. For hyperelastic materials,

conservative loadings and most cases of the kinematic boundary conditions the generalized stiffness matrix is symmetric even in points far from the generalized equilibrium positions. Nonconservative loads and some specific essential boundary conditions (as was mentioned above) lead to a nonsymmetric contribution to the resultant tangent stiffness. The latter is derived for several load types.

Results of numerical examples for both linear and nonlinear cases are presented, demonstrating the robustness and efficiency of the approach.

FULLY SPACE-TIME METRIC BASED ANISOTROPIC MESH ADAPTATION FOR UNSTEADY PROBLEMS

G. JANNOUN, E. HACHEM, J. VEYSSET, J-F. ZARAGOCI, AND T. COUPEZ

MINES ParisTech, Center for Materials Forming (CEMEF),
UMR CNRS 7635, BP 207, 06904 Sophia-Antipolis, France
e-mail: ghina.el_jannoun@mines-paristech.fr, www.cemef.mines-paristech.fr/

Key words: Anisotropic Mesh Adaptation, Adaptive Time-Stepping, Paradoxical meshing technique, Time-dependent problems

Abstract. This paper presents a novel method for building unstructured meshes for time-dependent problems. We start by introducing the classical anisotropic mesh adaptation technique proposed in [1, 2]. The latter is developed based on the length distribution tensor approach and the associated a posteriori edge based error analysis. Then we extend the mesh adaptation technique to contain adaptive time advancing. A newly developed time error estimator is constructed and intends to homogenize the global error over space and time. The main purpose of this work is the development of a novel meshing algorithm, the paradoxical meshing, that provides optimal space and time meshes suitable for several simulation time subintervals. The advantage of the proposed method relies in its conceptual and computational simplicity as it only requires from the user a number of nodes and a frequency of adaptation according to which the mesh and the time-steps are automatically adapted. Numerical solutions on time-dependent problems demonstrate the accuracy and efficiency of the proposed space-time error estimator.

1 INTRODUCTION

Despite the increasing computer performances and the progress of computational fluid dynamics in modelling and simulating time dependent PDEs, numerical restrictions are still present and caused by the complexity of the numerical simulations.

Anisotropic mesh adaptation has proved to be a powerful strategy to improve the accuracy and efficiency of finite element methods. It enables the capture of multi-scaled physical or mechanical phenomena. The method, as developed in [1], allows the creation of highly stretched and highly directional elements leading to very good capture of the gradients of the solution and the internal and boundary layers. Moreover, it provides a good level of accuracy within a reasonable degree of freedom. Another extension was proposed in [2], and accounts for time-step adaptation. Based on the derived error estimator

in space and the solutions at the previous times, the proposed algorithm automatically computes an appropriate time-step for the following computations.

The above described mesh adaptation is optimal for steady problems. It is theoretically valid for transient CFD applications only when applied, together with the time adaptation method, at every solver iteration. Nevertheless this would excessively increase the computational cost and lead to an accumulation of interpolation errors. Fixing a low frequency of adaptation can help solving these issues but this might lead to a time lag between the mesh and the solution. To overcome these problems, a new fully adaptive method is proposed in this paper: the paradoxical meshing. It intends to predict the solution's evolution over a period of time and to automatically generate corresponding mesh and set of time-step sizes.

This paper is structured as follows: we start section 2 with a brief description of the classical anisotropic mesh adaptation. Section 3 is dedicated to the time adaptive technique. The good performance of these methods is evaluated on a 3D example with complex geometry. The extension of these two methods into the paradoxical meshing algorithm is described in Section 4. Finally, in section 5, we test the efficiency and accuracy of the space-time adaptive algorithm on time-dependent problems.

2   Construction of an anisotropic mesh for stationary problems

In [1], we have developed an a posteriori edge based spatial error estimator relying on the length distribution tensor approach. Working on a nodal based metric, an anisotropic mesh adaptation procedure is obtained under the constraint of a fixed number of nodes.

2.1   Edge based error estimation

We consider $u \in \mathcal{C}^2(\Omega) = \mathcal{V}$ and $\mathcal{V}_h$ a simple P^1 finite element approximation space:
$$\mathcal{V}_h = \left\{ w_h \in \mathcal{C}^0(\Omega), w_h|_K \in P^1(K), K \in \mathcal{K} \right\}$$
where $\Omega = \bigcup_{K \in \mathcal{K}} K$ and K is a simplex (segment, triangle, tetrahedron, ...).

We define $\mathbf{X} = \left\{ \mathbf{X}^i \in \mathbb{R}^d, i = 1, \cdots, N \right\}$ as the set of nodes of the mesh and we denote by U^i the nodal value of u at $\mathbf{X}^i$ and we let Π_h be the Lagrange interpolation operator from $\mathcal{V}$ to $\mathcal{V}_h$ such that:
$$\Pi_h u(\mathbf{X}^i) = u(\mathbf{X}^i) = U^i, \, \forall i = 1, \cdots, N$$
As shown in figure 1, we define the set of nodes connected to node i by

$$\Gamma(i) = \left\{ j, \exists^i K \in \mathcal{K}, \mathbf{X}^i, \mathbf{X}^j \text{ are nodes of } K \right\}$$

By introducing the following notation: $\mathbf{X}^{ij} = \mathbf{X}^j - \mathbf{X}^i$ and using the analysis carried in [1], we can set the following results:

$$\nabla u_h \cdot \mathbf{X}^{ij} = U^{ij}, \tag{1}$$
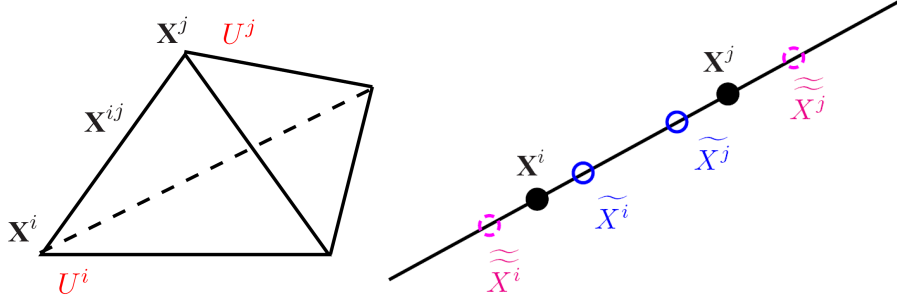
Figure 1: length X^{ij} of the edge joining nodes i and j (left). Varying the edge in its own direction (right).

$$\underbrace{||\nabla u_h \cdot \mathbf{X}^{ij}}_{U^{ij}} - \nabla u(X^i) \cdot \mathbf{X}^{ij}|| \leq \max_{Y \in [X^i, X^j]} |\mathbb{H}(u)(Y)\mathbf{X}^{ij} \cdot \mathbf{X}^{ij}|, \tag{2}$$

where $\mathbb{H}(u) = \nabla^{(2)}u$ is the associated Hessian of u. A definition of the projected second derivative of u is obtained using (1) and the interpolation operator on ∇u :

$$\nabla g_h \mathbf{X}^{\mathbf{ij}} \cdot \mathbf{X}^{\mathbf{ij}} = g^{ij} \cdot \mathbf{X}^{\mathbf{ij}} \tag{3}$$

where $\nabla g_h = \Pi_h \nabla u$, $g^i = \nabla u(\mathbf{X^i})$ and $g^{ij} = g^j - g^i$.

Using a mean value argument, we set that:

$$\exists y \in [x^i, x^j] | g^{ij} \cdot \mathbf{X}^{\mathbf{ij}} = \mathbb{H}(u)(Y)\mathbf{X}^{ij} \cdot \mathbf{X}^{ij}.$$

This projection is considered as an expression of the error along the edge:

$$e_{ij} = g^{ij} \cdot \mathbf{X}^{\mathbf{ij}} \tag{4}$$

However a gradient recovery procedure is needed as the gradient of u is not known and is not necessarily continuous at the nodes of the mesh.

2.2 Gradient Recovery

Based on an optimization analysis, the author in [1, 2] proposes a recovery gradient operator defined by:

$$G^i = (\mathbb{X}^i)^{-1} \sum_{j \in \Gamma(i)} U^{ij} \mathbf{X}^{ij} \tag{5}$$

where $\mathbb{X}^i = \frac{d}{|\Gamma(i)|} \sum_{j \in \Gamma(i)} \mathbf{X}^{ij} \otimes \mathbf{X}^{ij}$ is the length distribution tensor at node $\mathbf{X^i}$. Note that this construction preserves the second order:

$$\left|(G^i - g^i) \cdot \mathbf{X}^{\mathbf{ij}}\right| \sim \left(\mathbb{H}(u)\mathbf{X}^{ij} \cdot \mathbf{X}^{ij}\right)$$

where G^i is the recovery gradient at node i (given by (5)) and g^i being the exact value of the gradient at node i. The error is evaluated by substituting g by G in (4):

$$e_{ij} = G^{ij} \cdot \mathbf{X}^{\mathbf{ij}}$$

2.3   Metric construction from the edge distribution tensor

Taking into account this error analysis, we construct the metric for the unit mesh as follows:

$$
\mathbb{M}^i = \left(\frac{d}{|\Gamma(i)|} \sum_{j \in \Gamma(i)} \mathbf{X}^{ij} \otimes \mathbf{X}^{ij} \right)^{-1}
$$

For a complete justification of this result, the reader is referred to [1, 2].

2.4   Error behavior due to varying the edge length

We examine now how the error behaves when we change the length of the edges by stretching coefficients

$$
\mathcal{S} = \left\{ s_{ij} \in \mathbb{R}^+ , \, i = 1, \cdots, N , \, j = 1, \cdots, N , \, \Gamma(i) \cap \Gamma(j) \neq \phi \right\}
$$

Stretching factors $s \in \mathbb{R}$ are employed to link the error variations to the changes in edge lengths:

$$
\begin{cases} \widetilde{\mathbf{X}_{ij}} = s \mathbf{X}_{ij} \\ ||\widetilde{e_{ij}}|| = s^2 ||e_{ij}|| = s^2 ||G^{ij} \cdot \mathbf{X}_{ij}|| \end{cases}
\tag{6}
$$

where $\widetilde{e_{ij}}$ and $\widetilde{\mathbf{X}_{ij}}$ are the target error at edge ij and its associated edge length.

The metric associated with $\mathcal{S}$ can be redefined as:

$$
\widetilde{\mathbb{M}^i} = \frac{|\Gamma(i)|}{d} \left(\widetilde{\mathbb{X}^i} \right)^{-1} \quad \text{with} \quad \widetilde{\mathbb{X}^i} = \frac{d}{|\Gamma(i)|} \sum_{j \in \Gamma(i)} s_{ij}^2 \mathbf{X}^{ij} \otimes \mathbf{X}^{ij}
\tag{7}
$$

being is the length distribution tensor. Let n_{ij} be the number of created nodes in relation with the stretching factor s_{ij} and along the edge ij. When scaling the edges by a factor s_{ij}, the error changes quadratically so that the number of created nodes along the edge ij is given by:

$$
n_{ij} = \left(\frac{\widetilde{e_{ij}}}{e_{ij}} \right)^{-\frac{1}{2}} = s_{ij}^{-1}
$$

Recall that $\widetilde{e_{ij}}$ denotes the induced error for edge $\widetilde{X^{ij}}$. As per node i, the number of created nodes along the different edges' directions is given by the following tensor:

$$
N^i = \left(\mathbb{X}^i \right)^{-1} \left(\frac{d}{|\Gamma(i)|} \sum_{j \in \Gamma(i)} n_{ij}{}^2 \mathbf{X}^{ij} \otimes \mathbf{X}^{ij} \right)
$$

So that the total number of created nodes per node i is:

$$
n^i = \sqrt{ \det \left(\left(\mathbb{X}^i \right)^{-1} \left(\frac{d}{|\Gamma(i)|} \sum_{j \in \Gamma(i)} n_{ij}{}^2 \mathbf{X}^{ij} \otimes \mathbf{X}^{ij} \right) \right) }
$$

Assuming a uniform totally balanced error along the edge $\widetilde{e_{ij}} = e = \text{cst}$, we get a direct relation between N and e as follows:

$$n^{ij}(e) = s_{ij}^{-1}(e) = \left(\frac{e}{e_{ij}}\right)^{-\frac{1}{2}}$$

Hence for a node i we have

$$n^i(e) = \sqrt{\det\left((\mathbb{X}^i)^{-1}\left(\frac{d}{|\Gamma(i)|}\sum_{j\in\Gamma(i)} n_{ij}(e)^2 \mathbf{X}^{ij} \otimes \mathbf{X}^{ij}\right)\right)}$$

Replacing $n_{ij}(e)$ by its expression,

$$n^i(e) = e^{-\frac{d}{2}}\sqrt{\det\left((\mathbb{X}^i)^{-1}\left(\frac{d}{|\Gamma(i)|}\sum_{j\in\Gamma(i)} e_{ij}\mathbf{X}^{ij} \otimes \mathbf{X}^{ij}\right)\right)}$$

and this is equivalent to:

$$n^i(e) = e^{-\frac{d}{2}} n^i(1)$$

so that the total number of nodes in the adapted mesh is: $N = e^{-\frac{d}{2}}\sum_i n^i(1)$.

Hence, the global induced error for N nodes can be determined by:

$$e(N) = \left(\frac{N}{\sum\limits_i n^i(1)}\right)^{-\frac{2}{d}}$$

Therefore the corresponding stretching factors under the constraint of a fixed number of nodes N are given by:

$$s_{ij} = \left(\frac{e}{e(N)}\right)^{-\frac{1}{2}} = \left(\frac{\sum\limits_i n^i(1)}{N}\right)^{\frac{2}{d}} e_{ij}^{-1/2}$$

Note that the mesh does not change during time advancing but at a certain time level t^n. Hence, an optimal mesh at a time level t^n need not be an optimal one at t^{n+1} which is the case when propagating a discontinuity. This raises the question about the frequency of remeshing.

As time-dependent problems exhibit arbitrary progression with time, the duration of applicability of a mesh cannot be known apriori. When the time-step size is greater than the length of the mesh's time interval, the solution may propagate into a non pre-adapted region of the domain resulting in a mesh/solution lag. Adapting the mesh at every solver iteration guarantees that the spatial error remains bounded. Nevertheless, this approach increases significantly the computational cost and leads to the accumulation of interpolation errors polluting the solution.

3 Time adaptation procedure

The focus now can be waived to the choice of the time-step size computed at each solver iteration. The main objective is to produce a time-step that preserves the accuracy of the mesh adapted solution and avoids unnecessarily small time-step sizes. This method was first introduced and validated in [2]. In this work, we revisit the developed method with the intention of extending it to a fully space-time adaptivity.

We apply the above described analysis in 1D where the only variable is time. Denote by $\mathcal{T} = \{t^0, \cdots, t^{n-1}, t^n, t^{n+1}, \cdots\}$ and let $t^{nk} = |t^n - t^k|$, n, $k \in \mathcal{T}$ be the temporal nodes and $\Delta t^n = t^{nn+1}$ the time increments as shown in figure 2. Assume that the solution is already computed on the whole domain up to time t^n. The aim is to choose an appropriate time-step Δt^n.

Without loss of generality, the analysis will be carried on an arbitrary spatial node i. Note that at a spatial node i, we only have one time edge to be determined $(t^n t^{n+1})$.
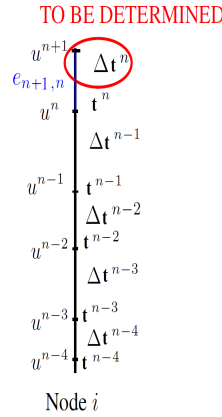
Figure 2: Temporal discretization at the spatial node i.

Define $\{\tau_{nn+1}\}$ to be the temporal edge scaling (stretching) factor such that:

$$
\begin{aligned}
\tilde{e}_{n+1,n} &= \tau_{n+1,n}^2 e_{n+1,n} \\
\left|\tilde{t}^{n+1,n}\right| &= \tau_{n+1,n} \left|t^{n+1,n}\right|
\end{aligned}
\tag{8}
$$

where $e_{n+1,n}$ is an approximation of the interpolation error from t^n to t^{n+1}, $\tilde{e}$ and $\left|\tilde{t}\right|$ are the target error at the temporal edge $t^n t^{n+1}$ and its associated edge length.

Let u_{n-1}^i, u_n^i and u_{n+1}^i be the solutions at node i and times $n - 1$, n, and $n + 1$, respectively. Using a forward difference approximation, we have that $u_{n+1} - u_n = \dot{u}_n \Delta t_n$ and $u_{n-1} - u_n = -\dot{u}_{n-1}\Delta t_{n-1}$. Then applying the recovery gradient in $1D$, we get:

$$
\dot{u}_n = \frac{u_{n,n+1}^i \Delta t_n + u_{n,n-1}^i \Delta t_{n-1}}{\Delta t_n^2 + \Delta t_{n-1}^2}
$$

and the quadratic interpolation error:

$$e_{n,n-1}^i = \dot{u}_{n,n-1}^i \Delta t_{n-1}$$

and

$$\widetilde{e}_{n,n-1}^i = {\tau_{n-1}^i}^2 \dot{u}_{n,n-1}^i \Delta t_{n-1}$$

where $\dot{u}_{n,n-1}^i = \dot{u}_n^i - \dot{u}_{n-1}^i$. Now using the equidistribution error argument, we write

$$\tilde{e}_{n,n-1}^i = e_n(N)$$

with $e_n(N)$ being the maximal error in space for a total number of nodes N. Hence the stretching factor of the time-step size is given by:

$$\tau_{n-1}^i = \left(\frac{e(N, t_n)}{e_{n,n-1}^i} \right)^{\frac{1}{2}}$$

and the optimal time-step is determined by:

$$\widetilde{\Delta t_n} = \min_i \tau_{n-1}^i \Delta t_n \tag{9}$$

Looking closely at this formula, we notice that it requires the solution at time t^{n+1} which is not yet computed. Therefore instead of computing the optimal time-step $\widetilde{\Delta t^n}$ we calculate $\widetilde{\Delta t^{n-1}}$ and we let $\Delta t^n = \widetilde{\Delta t^{n-1}}$.

3.1 Application to 3D heat transfer and turbulent flow inside an industrial furnace

In this section, we will apply the classical space and time-adaptive methods to simulate the heat transfer and fluid flows inside an industrial furnace with complex geometry. The objective of this test case is to show the applicability and the potential of the developed algorithm in simulating long time heating inside large scale furnaces.

The furnace is modelled as a hexagonal section duct of $2.7 \times 8.1 \times 5.3 m^3$ forming one heat transfer zone. All computations have been conducted by starting with a gas at rest with a constant temperature of $1463°C$. Adiabatic temperature is considered at all other boundaries for sake of simplicity. The heated air is pumped into the furnace at a velocity $14.3 m/s$ by a circular burner with $6m$ diameter and located at the left vertical wall. The air is vented out of the furnace through two outlets positioned at the bottom of the left vertical wall. The 3D computations aim at simulating an hour of heating and have been conducted in parallel on 16, $2.4Ghz$ Opteron cores.

Figure 3 (top) shows the isothermal distribution at different time-steps. When the hot fluid spreads along the volume of the furnace, it induces a turbulent motion within the geometry. This forced convection is caused by the interaction of the moving stream and the
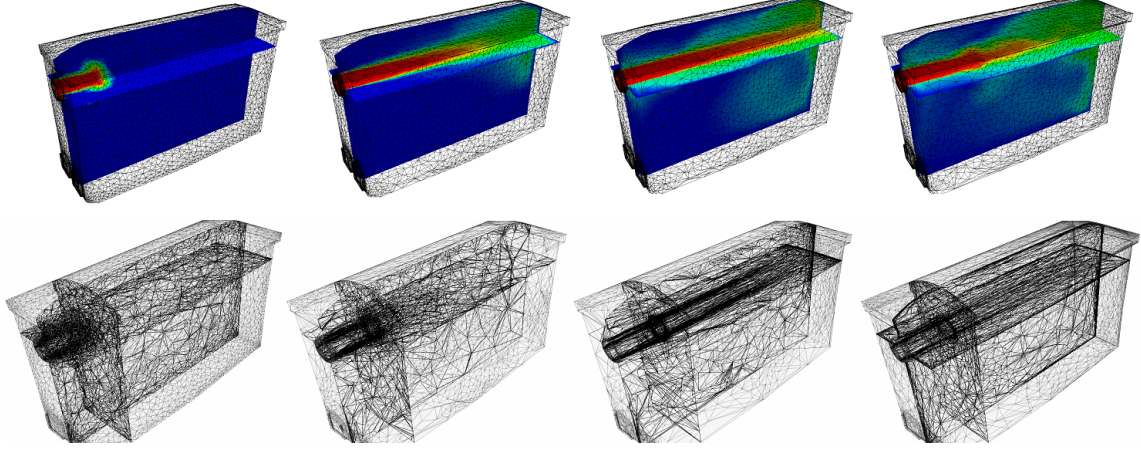
Figure 3: Isotherms (Top) and corresponding adapted meshes (bottom) at three different time-steps.

	CPU Time (s)
Non-Adaptive $\Delta t = 0.005s$	8,640,000
Adaptive Δt	172,800
Ratio	50

Table 1: CPU time for computing the solution with non-adaptive and a space/time adaptive methods

stationary fluid inside the furnace. The numerically obtained temperature distribution (fig. 3) clearly reflects the expected flow pattern. A number of small vortices inside different buffer zones can be observed. The latter are due to the turbulence dissipation and mixing of the hot and cold air. The numerical results obtained with our space and time adaptive algorithm reflect well the efficiency and potential of the methods. Figure 3 (bottom) highlights how well the mesh is adapted to sharply capture, with highly stretched elements, the gradients of the solution, the boundary layers and the emerging vortices. The algorithm builds up the mesh in a way to maximize the accuracy of the numerical solution with a fixed number of nodes (100, 000). Note that the mesh is adapted according to the velocity components and its norm as in this test case it is the motor of the induced airflow and spread of the temperature. The results describing only one hour of the heating process required 100 days of computations with a fixed time-step equal to 0.005s. Significant CPU time and computational cost were saved by applying our time adaptive procedure as it required only 2 days of calculations (see table 1). Figure 4 shows the evolution of the time-step sizes allowing at the same time a certain level of accuracy and an acceleration of the computations.
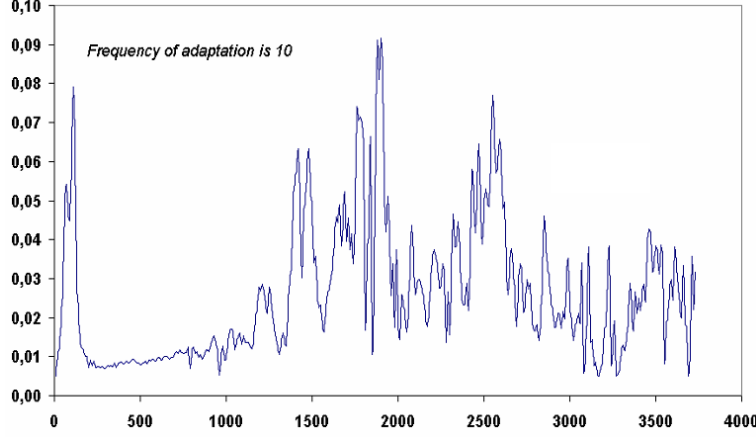
Figure 4: Time-step evolution for the simulation of an hour of heating inside an industrial furnace.

4 paradoxical meshing: Full adaptivity algorithm

When dealing with steady state problems, the classical mesh adaptation technique, presented in section 2, performs pretty well for converging the mesh-solution couple. Nevertheless, this method is no more optimal when applied to unsteady problems as the physical solution evolves in time. Together with the time-adaptive method introduced in section 3, the classical mesh adaptation can be efficiently adapted to time dependent problems.

In this paper we are interested in developing a space and time fully adaptive algorithm. The latter aims at anticipating the solution progress over a period of time and generating the optimal mesh that is adequately adapted, for a fixed number of nodes, to the evolving solution along that time interval. The analysis is carried out on a (3D+1D) mesh, i.e. the computations are performed synchronously on a 3D spatial mesh and a 1D temporal mesh. We aim at generating a mesh that holds for several solver iterations together with the corresponding optimal set of time-step sizes. Note that the user can assign a frequency of adaptation and the algorithm will accordingly adapt the meshes.

The principle consists of dividing the simulation time $[0, T]$ into n_{SI} subintervals:

$$[0, T] = [0, T^1] \cup [T^1, T^2] \cup \cdots [T^{k-1}, T^k] \cup [T^k, T^{k+1}] \cup \cdots \cup [T^{n_{\mathrm{SI}}-2}, T^{n_{\mathrm{SI}}-1}]$$

that will in turn be divided into $n_{\mathrm{freq}} - 1$ subintervals where n_{freq} is the frequency of adaptation assigned by the user. We call the adaptation method a paradoxical meshing as the resulting mesh is being adapted to n_{freq} time-steps while adapting every n_{freq} steps.

The mesh and the set of time-step sizes are computed through an iterative process along which we try to converge both meshes (the spatial and the temporal one) to the optimal configurations that give the most accurate solution for the corresponding interval of time. At every iteration, we consider each of the n_{SI} intervals at a time and divide it

into subintervals

$$[T^k, T^{k+1}] = [T^k = t_0^k, t_1^k] \cup [t_1^k, t_2^k] \cup \cdots \cup [t_l^k, t_{l+1}^k] \cup \cdots \cup [t_{n_{\text{freq}}-2}^k, t_{n_{\text{freq}}-1}^k = T^{k+1}]$$

such that $t_{l+1}^k = t_l^k + \Delta_{k,l} = T^k + l \times \Delta_{k,l}$ and $\Delta_{k,l} = \frac{\delta_k}{n_{\text{freq}}}$.
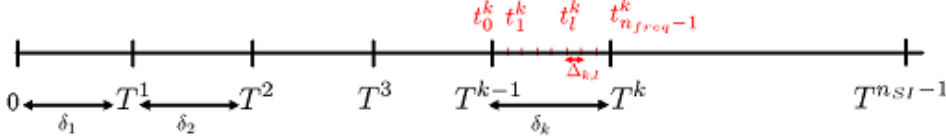
Figure 5: Temporal subintervals $[T^{k-1}, T^k]$.

The solution is predicted at each of the $\{t_l^k\}$ temporal nodes using the numerical scheme. From these solutions we construct a vector field

$$V = \{U^{k,0}, U^{k,1}, \cdots, U^{k,l}, U^{k,n_{\text{freq}}-1}\}$$

and we estimate the edge based spatial errors:

$$e_{ij} = \max_{0 \leq k \leq n_{\text{SI}}} G_{ij}^k \cdot \mathbf{X_{ij}}$$

We compute a global error $e(N, T_k)$, as in the classical approach, for equidistributing the error on the edges of the discrete domain. An optimal metric is deduced controlling the spatial error over $[T^k, T^{k+1}]$. We also compute the temporal error at the nodes t_l^k as in section 3. Once this is done we optimize the time-step sizes $\Delta t_{k,l}$ by equidistributing the error in space and time:

$$\Delta t_{k,l} = \left(\frac{e(N, T_k)}{\max_i e_i^{k,l}} \right)^{\frac{1}{2}} \times \Delta t_{k,l}$$

An optimal size $\tilde{\delta t}_k$ of the time interval $[T^k, T^{k+1}]$ is recomputed as follows:

$$\widetilde{\delta t_k} = \sum_l \Delta t_{k,l}$$

These $\widetilde{\delta t_k}$ are given to the $1D$ mesher that will generate a new discretization of the interval $[0, T]$ as well as its corresponding temporal nodes T^k. Notice that the number of subintervals n_{SI} will be automatically changed due to this remeshing. The above described algorithm is repeated iteratively until convergence of the metrics and the set of time-step sizes. Therefore, for each interval $[T^k, T^{k+1}]$ a metric is computed accounting for the solution's transient evolution. At convergence, a mesh is generated from this metric field. Computations are then resumed on the predicted optimal set of meshes with the corresponding set of time-step sizes.

The novel method that we presented herein is perceived not only as a fully adaptive technique but also as a space and time accurate way of solving time-dependent problems within reasonable computational costs.
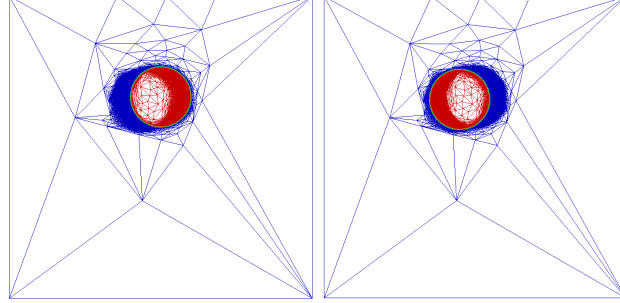
Figure 6: Paradoxical meshing algorithm applied to a rotating circle for the interval $[t^n, t^{n+1}]$.
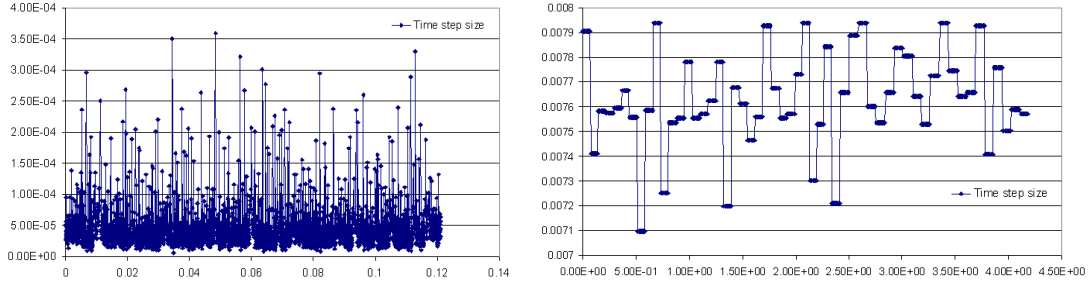
Figure 7: Time-steps generated by the classical (left) and the paradoxical (right) meshing algorithm.

5 Numerical example: Rotating Circle

In this section, we assess the performance of the newly developed fully-adaptive algorithm on a numerical test case and compare the result with the classical space and time adaptive techniques. We consider a 2D circle of radius 0.3 located at $(0.5, 0.5)$ in the computational domain $[-1.5, 1.5] \times [-1.5, 1.5]$. The simulation consists of rotating the circle in the counterclockwise direction at the rate $\theta = 1 rad/s$. The objective is to test the capability of the anisotropic paradoxical meshing technique to accurately capture the dynamically evolving interface. Figure 6 shows the adapted mesh, made up of $10,000$ nodes, for the time interval $[t^n, t^{n+1}]$ made up of 10 sub-intervals of time. We can clearly see how refined the mesh is at the location of high gradients of the solution and how accurate is the capture of the interface as it rotates from t^n(left) to time t^{n+1}(right). The elements all along the interface are isotropic yielding a well respected curvature. The time intervals' lengths $[t^n, t^{n+1}] = [t_0^n, t_1^n, \cdots, t_i^n, \cdots t_{10}^{n+1}]$ are automatically generated to guarantee the validity of the mesh for 10 consecutive time-steps. Figure 7 presents the time-step sizes for the first few iterations of the algorithm. The periodic variation of the time-steps is in good agreement with the nature of the problem, as the circle rotates at a constant rate and maintains the same behavior all over computations.

Using the classical mesh adaptation with the time adaptive technique and adapting the mesh every 10 iterations, the generated time-step sizes will be too small, as seen in figure

7(left), in order to reduce the temporal error, preventing the progress of the solution in time so that it remains in phase with the adapted mesh. Recall that the method aims at giving a better efficiency than the classical algorithm. This is exactly what we notice when comparing figures 7(left) and 7(right); the paradoxical meshing technique produces time-step sizes that are almost 200 times larger than those generated by the classical algorithm. Hence, the resulting computations will be about 200 times faster reflecting the high efficiency of the novel method. Note that the inner loop of the algorithm is repeated only two times to get the optimal meshes and time-step size for which the solution remains bounded.

6   CONCLUSIONS

In this paper, we have presented a classical anisotropic mesh adaptation that showed very good performance when applied together with the new time adaptive technique for resolving time dependent problems. An extension of these algorithms lead to a novel and very powerful method for full adaptation known as the paradoxical meshing. This method demonstrated its efficiency in generating meshes and time-step sizes that guarantee the convergence of the solution all over computations for a limited number of nodes and a fixed frequency of adaptation.

REFERENCES

[1] T. Coupez, *Metric construction by length distribution tensor and edge based error for anisotropic adaptive meshing*, Journal of Computational Physics, vol. 230, (2011), 2391-2405.

[2] T. Coupez and G. Jannoun and N. Nassif and H.C. Nguyen and H. Digonnet and E. Hachem,*Adaptive Time-step with Anisotropic Meshing for Incompressible Flows*, Submitted to Journal of Computational Physics, (2012).

[3] E. Hachem and B. Rivaux and T. Klozcko and H. Digonnet and T. Coupez, *Stabilized finite element method for incompressible flows with high Reynolds number*, Journal of Computational Physics, Vol. 229, (2010), *23*: 8643-8665.

A POSTERIORI ERROR ESTIMATION OF TARGET CONTROL PROBLEMS: WEAK FORMULATION OF INEQUALITY CONSTRAINTS

HÅKAN JOHANSSON

Department of Applied Mechanics, Chalmers University of Technology, SE-412 96 Göteborg, Sweden
hakan.johansson@chalmers.se

Key words: target control, optimal control, a posteriori error estimation

Abstract. This contribution considers the steering of movements of a mechanical system from an initial state to a target state (target control). Suitable FE-approximations for the state and control variables are discussed along with a goal-oriented a posteriori estimate of the discretization errors.

1 EXTENDED ABSTRACT

We shall in this contribution consider optimal control problems concerning the steering of motion of a mechanical system from an initial state to a target state (target or trajectory control). The motion of the system depends on forces acting as controls and is represented by a set of ordinary differential equations with load terms. By considering the equations of motion and the relevant kinematic and control limitations, a constrained optimization problem can be formulated where the control forces are sought to minimize a chosen objective function, such as the energy consumption or the deviation from desired trajectory, while reaching the defined target. As a numerical example we consider the search for an optimal brake-turning strategy for a vehicle manoeuver [4].

A discretization of finite element type in time is introduced, whereby approximations for the state (coordinates and velocities) and the control (external forces) variables are introduced. The optimality conditions are expressed in weak form, in particular, the inequality constraints are enforced weakly, whereby to what extent the inequality constraints are satisfied depend on the chosen discretization. The subject of the present work is to determine the error in the approximate solution compared to the exact solution, in particular with respect to how well the discrete solution satisfies inequality constraints and target conditions.

To this end, we employ a posteriori error estimates based on the pertinent dual problem (from linearization of the weak form) with some modification, whereby discretiza-

tion errors in both state and control variables can be estimated in terms of chosen goal quantities. The sources of errors can be traced to specific regions of the state and control time-meshes, which can be used in an adaptive mesh-refinement procedure since the control and state variables are discretized separately. Earlier work on a posteriori error estimation for optimal control problems have been based on the Heidelberg approach[1, 4], whereas the present contribution will use our previous work in error control for parameter identification problems based on a tangent form of the dual problem[2].

To illustrate the problem setting, consider the example[3] of a double pendulum representing simple mechanical system. The pendulum is to be lifted from vertical hanging to a horizontal straight position using minimal control force without violating anthropomorphic constraints (the opening angle of the middle joint must be between 0 and 135 degrees) and control constraints (the control variables are restricted by maximum and minimum values). The solution algorithm is based on a nested format with a relaxation of the constraints.
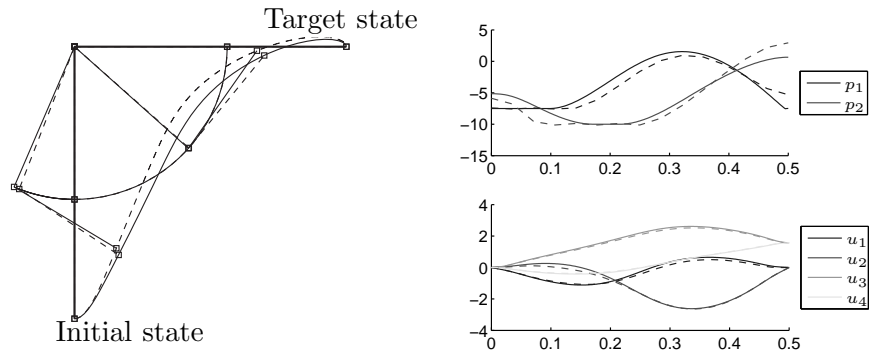
Figure 1: Left: Plot of the arm from vertical hanging to a horizontal straight position. Right: Comparison of the solved state (u_1-u_4) and control (p_1-p_2) variables using fine (dashed line) and coarse discretization (solid line).

The numerical example indicate that a discretization error in the control variable arises in order to "compensate" for discretization errors in the solution of the equations of motion in order to reach the target state.

REFERENCES

[1] R. Becker. Adaptive finite elements for optimal control problems. Habil. thesis, University of Heidelberg, (2004).

[2] H. Johansson, K. Runesson and F. Larsson. Parameter identification with sensitivity assessment and error computation. *GAMM-Mitteilungen*, 37 (2007), 430-457.

[3] M. Kaphle. Simulations of human movements through temporal discretization and optimization. Lic. thesis, Royal Institute of Technology, 2007.

[4] K. Kraft. Adaptive Finite Element Methods for Optimal Control Problems. Ph.D. thesis, Chalmers University of Technology, 2011.

ONLINE GREEDY REDUCED BASIS CONSTRUCTION USING DICTIONARIES

Sven Kaulmann[*] and Bernard Haasdonk[*]

[*]Institute for Applied Analysis and Numerical Simulation,
University of Stuttgart,
Pfaffenwaldring 57, 70569 Stuttgart, Germany
{`sven.kaulmann`|`haasdonk`}`@mathematik.uni-stuttgart.de`.

Key words: Greedy algorithms, reduced basis methods, model order reduction, dictionary

Abstract. The Reduced Basis method is a means for model order reduction for parametrized partial differential equations. In the last decades it has found broad application for problems with multi-query or real-time character. While the method has shown to be performing well for numerous different fields of applications, problems with high parameter dimension or high sensitivity with respect to the parameter still pose major challenges. In our contribution, we present a new basis generation algorithm that is particularly fit to these kinds of problems: Instead of building the reduced basis during the *offline* phase we build a large *dictionary* of basis vector candidates and compute a small parameter-adapted basis from that *dictionary* with a Greedy procedure during the *online* phase.

1 Introduction and Motivation

As numerical simulations find more and more use in real-world scenarios and industrial applications, demands concerning efficiency and reliability increase as well. Especially scenarios that call for real-time simulations or multi-query evaluations of partial differential equations (PDEs) often require means of model order reduction. Examples for such scenarios are optimal control and optimization settings.

The Reduced Basis (RB) method [6] provides model order reduction for a special class of PDEs, so-called parametrized partial differential equations, (in the weak form) given as

$$B_h(u_h(\boldsymbol{\mu}), v_h; \boldsymbol{\mu}) = L_h(v_h; \boldsymbol{\mu}) \quad \forall v_h \in X_h, \tag{1}$$

for $u_h(\boldsymbol{\mu}) \in X_h$, $\boldsymbol{\mu} \in \mathcal{P} \subset \mathbb{R}^p$ and a suitable given discrete function space X_h. Here, $B_h : X_h \times X_h \times \mathcal{P} \to \mathbb{R}$ denotes a given parametrized bilinear form and $L_h : X_h \times \mathcal{P} \to \mathbb{R}$

denotes a given parametrized linear form. We will assume B_h to be coercive and symmetric in the sequel.

Such equations arise, for example, in the context of heat diffusion. In this case, the parameter $\boldsymbol{\mu}$ could model the diffusion coefficient. The RB method now reduces the complexity of Equation 1 from $\mathcal{N} = \dim(X_h)$ to $N \in \mathbb{N}$, $N \ll \mathcal{N}$, by introducing a low-dimensional surrogate $X_N, \dim(X_N) = N$ of the high-dimensional discretization space X_h. This space X_N is the span of solutions of (1) for a given set of parameter values:

$$X_N = \langle \{u_h(\boldsymbol{\mu}_1), \ldots, u_h(\boldsymbol{\mu}_N)\} \rangle,$$
$$\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_N \in \mathcal{P}.$$

Galerkin-projection of Equation 1 then leads to a reduced dimensional equation system

$$\boldsymbol{A}_N(\boldsymbol{\mu}) \cdot \boldsymbol{u}_N(\boldsymbol{\mu}) = \boldsymbol{b}_N(\boldsymbol{\mu}), \tag{2}$$

where $\boldsymbol{u}_N(\boldsymbol{\mu}) \in \mathbb{R}^N$, $\boldsymbol{A}_N : \mathcal{P} \to \mathbb{R}^{N \times N}$, $(\boldsymbol{A}_N(\boldsymbol{\mu}))_{i,j} = B_h(\varphi_i, \varphi_j; \boldsymbol{\mu})$, $\boldsymbol{b}_N : \mathcal{P} \to \mathbb{R}^N$, $(\boldsymbol{b}_N(\boldsymbol{\mu}))_i = L_h(\varphi_i; \boldsymbol{\mu})$. Here, $\Phi = \{\varphi_1, \ldots \varphi_N\}$ denotes an orthonormal basis of the space X_N.

The main idea of the RB method is the so-called *offline-online splitting* of all computations: We introduce two phases of our computations:

Offline phase During this phase, all $\mathcal{N}$-dependent computations are performed. This phase may be very expensive as a certain amount of solutions $u_h(\boldsymbol{\mu}_i)$ needs to be computed.

Online phase During this phase, the equation at hand is solved in the reduced space X_N for a given parameter $\boldsymbol{\mu}$. This phase is ideally totally independent of $\mathcal{N}$, and therefore usually very fast.

While the computation of X_N can clearly be done during the offline phase, the assembly of the equation system (2) needs to be done for every new given parameter $\boldsymbol{\mu}$ during the online phase. As this requires evaluations of B_h at the solutions $u_h(\boldsymbol{\mu}_i)$, the online phase would hence depend on $\mathcal{N}$. We thus make the assumption of parameter separability:

Assumption 1. *Assume B_h, L_h to be parameter separable, that is*

$$B_h(u, v; \boldsymbol{\mu}) = \sum_{q=1}^{Q_B} \Theta_B^q(\boldsymbol{\mu}) B_h^q(u, v), \qquad L_h(u; \boldsymbol{\mu}) = \sum_{q=1}^{Q_L} \Theta_L^q(\boldsymbol{\mu}) L_h^q(u), \quad \forall u, v \in X_h, \tag{3}$$

for given numbers $Q_B, Q_L \in \mathbb{N}$, parameter-dependent functions $\Theta_B^q, \Theta_L^q : \mathcal{P} \to \mathbb{R}$ and parameter-independent bilinear and, respectively, linear forms $B_h^q : X_h \times X_h \to \mathbb{R}$, $L_h^q : X_h \to \mathbb{R}$.

Using Assumption 1, the assembly of the system (2) can be done in two steps: During the offline phase, after computing the reduced basis space X_N, project the parameter-independent components of B_h and L_h to X_N:

$$(\boldsymbol{A}_N^q)_{i,j} = B_h^q(\varphi_i, \varphi_j), \quad 1 \le q \le Q_B, 1 \le i,j \le N,$$
$$(\boldsymbol{b}_N^q)_i = L_h^q(\varphi_i), \quad 1 \le q \le Q_L, 1 \le i \le N.$$

During the online phase, it then only remains to sum up the precomputed components:

$$\boldsymbol{A}_N(\boldsymbol{\mu}) = \sum_{q=1}^{Q_B} \Theta_B^q(\boldsymbol{\mu}) \boldsymbol{A}_N^q, \qquad\qquad \boldsymbol{b}_N(\boldsymbol{\mu}) = \sum_{q=1}^{Q_L} \Theta_L^q(\boldsymbol{\mu}) \boldsymbol{b}_N^q.$$

1.1 Error Estimation

One crucial ingredient of the reduced basis method is *a posteriori* error estimation. Error estimation is used for basis construction during the offline phase and for certification by approximation quality control during the online phase. In our work we use a residual-based estimator shortly outlined in this paragraph. For more details we refer to [1, 6].

Definition 1.1. For B_h given from (1) we denote by $\|\cdot\|_{\boldsymbol{\mu}} : X_h \to [0, \infty)$ the parameter dependent *energy* norm

$$\|u\|_{\boldsymbol{\mu}} = \sqrt{B_h(u, u; \boldsymbol{\mu})}. \tag{4}$$

We introduce the residual and its Riesz-representative.

Definition 1.2. For a given function $u \in X_h$ let the *residual* $r_h[u] : X_h \times \mathcal{P} \to \mathbb{R}$ be given by

$$r_h[u](v; \boldsymbol{\mu}) := L_h(v; \boldsymbol{\mu}) - B_h(u, v; \boldsymbol{\mu}) \quad \forall v \in X_h.$$

Its *Riesz-representative* $r_u(\boldsymbol{\mu}) \in X_h$, given a parameter $\overline{\boldsymbol{\mu}} \in \mathcal{P}$, is defined as the solution to

$$B_h(r_u(\boldsymbol{\mu}), v; \overline{\boldsymbol{\mu}}) = r_h[u](v; \boldsymbol{\mu}) \quad \forall v \in X_h.$$

Using the Riesz-representative we can now state our error estimator.

Theorem 1.1 (Residual based a posteriori error estimate). *Given parameters $\boldsymbol{\mu}, \overline{\boldsymbol{\mu}} \in \mathcal{P}$, the energy norm of the Riesz-representative to a given reduced approximation $u_N(\boldsymbol{\mu})$ is an efficient a posteriori error estimate in the sense that*

$$\frac{1}{\gamma_{\overline{\boldsymbol{\mu}}}(\boldsymbol{\mu})} \left\| r_{u_N(\boldsymbol{\mu})}(\boldsymbol{\mu}) \right\|_{\overline{\boldsymbol{\mu}}} \quad \le \quad \left\| u_h(\boldsymbol{\mu}) - u_N(\boldsymbol{\mu}) \right\|_{\overline{\boldsymbol{\mu}}} \quad \le \quad \frac{1}{\alpha_{\overline{\boldsymbol{\mu}}}(\boldsymbol{\mu})} \left\| r_{u_N(\boldsymbol{\mu})}(\boldsymbol{\mu}) \right\|_{\overline{\boldsymbol{\mu}}}.$$

Here we used the constants $\alpha_{\overline{\boldsymbol{\mu}}}, \gamma_{\overline{\boldsymbol{\mu}}} \in \mathbb{R}$,

$$\alpha_{\overline{\boldsymbol{\mu}}} = \inf_{u \in X_h} \frac{B_h(u, u; \overline{\boldsymbol{\mu}})}{\|u\|_{\overline{\boldsymbol{\mu}}}^2}, \quad \gamma_{\overline{\boldsymbol{\mu}}} = \sup_{u,v \in X_h} \frac{B_h(u, v; \overline{\boldsymbol{\mu}})}{\|u\|_{\overline{\boldsymbol{\mu}}} \|v\|_{\overline{\boldsymbol{\mu}}}}$$

We define: $\Delta_{X_N}(\boldsymbol{\mu}) = \frac{1}{\alpha_{\overline{\boldsymbol{\mu}}}(\boldsymbol{\mu})} \left\| r_{u_N(\boldsymbol{\mu})}(\boldsymbol{\mu}) \right\|_{\overline{\boldsymbol{\mu}}}$.

Remark 1. *The constants* $\alpha_{\overline{\boldsymbol{\mu}}}, \gamma_{\overline{\boldsymbol{\mu}}}$ *can be bound by easily computable constants using the Min-Theta approach [6].*

Remark 2. *In the following we will always assume the parameter* $\overline{\boldsymbol{\mu}} \in \mathcal{P}$ *to be given and will use it without further notice.*

1.2 Efficient Evaluation of the Error Estimator

As a preparation of evaluations of the error estimator in a reduced space X_N we compute the components $r_L^q \in X_h, q \in \{1, \dots, Q_L\}$

$$B_h(r_L^q, v; \overline{\boldsymbol{\mu}}) = L_h^q(v) \quad \forall v \in X_h, \tag{5}$$

and the components $r_B^q \in X_h, q \in \{1, \dots, Q_B \cdot N\}$ with

$$B_h(r_B^{(j-1)\cdot N+i}, v; \overline{\boldsymbol{\mu}}) = B_h^j(\varphi_i, v) \quad \forall v \in X_h, \tag{6}$$

where $\Phi = \{\varphi_i | 1 \le i \le N\}$ is a basis of X_N. For the sake of simplicity of the following presentation, we collect the energy products of all Riesz-representatives in one matrix $\mathbf{G} \in \mathbb{R}^{Q_r \times Q_r}, Q_r = Q_L + Q_B N$:

$$\mathbf{G} = \begin{pmatrix} G_1 & G_2 \\ G_3 & G_4 \end{pmatrix}, \tag{7}$$

where

$$(G_1)_{i,j} = B_h(r_L^i, r_L^j, \overline{\boldsymbol{\mu}}), \quad (G_2)_{i,j} = B_h(r_L^i, r_B^j, \overline{\boldsymbol{\mu}}),$$
$$(G_3)_{i,j} = B_h(r_B^i, r_L^j, \overline{\boldsymbol{\mu}}), \quad (G_4)_{i,j} = B_h(r_B^i, r_B^j, \overline{\boldsymbol{\mu}}).$$

Finally, we define the parameter vector $\boldsymbol{\Theta}_r(\boldsymbol{\mu}, u_N) \in \mathbb{R}^{Q_r}$ for a given parameter $\boldsymbol{\mu} \in \mathcal{P}$ and a given reduced function $u_N \in X_N$:

$$(\boldsymbol{\Theta}_r(\boldsymbol{\mu}, u_N))_k = \begin{cases} \Theta_L^k(\boldsymbol{\mu}), & k \le Q_L \\ -(u_N)_i \Theta_B^j(\boldsymbol{\mu}), & \text{else} \end{cases} \tag{8}$$

where $i = ((k - Q_L) \mod N)$, $j = \frac{k - Q_L - i}{N} + 1$. The evaluation $\Delta_{X_N}(\boldsymbol{\mu})$ of the error estimator is then given as

$$\Delta_{X_N}(\boldsymbol{\mu}) = \frac{1}{\alpha_{\overline{\boldsymbol{\mu}}}(\boldsymbol{\mu})} \sqrt{\boldsymbol{\Theta}_r(\boldsymbol{\mu}, u_N) \cdot \mathbf{G} \cdot \boldsymbol{\Theta}_r(\boldsymbol{\mu}, u_N)}.$$

1.3 Summary

Using the offline-online splitting, RB methods gain impressive complexity reductions for a wide range of applications such as elliptic stationary problems [6], parabolic instationary problems [3] and hyperbolic problems [4]. The Greedy-type algorithm used for basis construction during the offline phase [6] usually yields small bases that at the same time guarantee a small error $\Delta_{X_N}(\boldsymbol{\mu})$.

In this contribution, we investigate problems with very high sensitivity with respect to the parameter $\boldsymbol{\mu}$ that yield unfeasibly large reduced bases. For these kinds of problems, we introduce a new method for model order reduction that uses a large *dictionary* of basis vector candidates to build a small, parameter-adapted basis during the online phase. Our method holds some similarity with the *locally adaptive Greedy method* introduced in [5]. As a main difference, our method does not use proximity in parameter space as an indicator for well-suited basis candidates in the basis construction but directly measures function similarity via error estimation. This will always yield ideal basis sizes.

Further ideas about Greedy methods and dictionary approaches can be found in [7].

In Section 2 we present our dictionary construction algorithm. Section 3 is dedicated to the online basis construction procedure. Finally we present some preliminary numerical results in Section 4.

2 Dictionary Construction and Offline Data Computation

During the offline phase of our new method, we construct a "dictionary" $\mathcal{D}$ with size D of basis vector candidates $\varphi_i \in X_h$:

$$\mathcal{D} = \left\{ \varphi_i \,|\, 1 \le i \le D \right\}.$$

For the experiments presented in Section 4 we use a pretty straightforward algorithm to construct the dictionary: We choose a finite subset $\mathcal{S} \subset \mathcal{P}$, referred to as the *training set* in the sequel, and compute

$$\mathcal{D} = \left\{ u_h(\boldsymbol{\mu}) \in X_h \,|\, \boldsymbol{\mu} \in \mathcal{S} \right\}. \tag{9}$$

This idea restricts the size of the training set to a certain extent as its size is directly linked to the size of the dictionary. We will comment on more elaborate methods in Section 5.

Together with the dictionary we compute the matrices $\left\{ \mathbf{A}_D^q \in \mathbb{R}^{D \times D} \,|\, 1 \le q \le Q_B \right\}$ and the vectors $\left\{ \mathbf{b}_D^q \in \mathbb{R}^D \,|\, 1 \le q \le Q_L \right\}$,

$$\begin{aligned} (\mathbf{A}_D^q)_{i.j} &= B_h^q(\varphi_i, \varphi_j), \quad \varphi_i, \varphi_j \in \mathcal{D}, \\ (\mathbf{b}_D^q)_i &= L_h^q(\varphi_i), \quad \varphi_i \in \mathcal{D} \end{aligned} \tag{10}$$

that will be needed for reduced simulation during the online phase. Furthermore, we compute the matrix $\mathbf{G} \in \mathbb{R}$ from Section 1.2 for $\Phi = \mathcal{D}$.

3 Online Basis Construction

In this section we describe the Greedy algorithm that is used to construct a space $X_N(\boldsymbol{\mu})$ from the dictionary for a given parameter $\boldsymbol{\mu} \in \mathcal{P}$. As a means to this end we define a so-called *indicator function* $\eta_\Delta : X_h \times \mathcal{P} \to [0, \infty)$ that indicates the reduction of the error from Theorem 1.1 for a given Parameter $\boldsymbol{\mu}^* \in \mathcal{P}$ in a given space X_N if X_N is enlarged with $\varphi \in X_h$:

$$\eta_\Delta(\varphi, \boldsymbol{\mu}^*; X_N) = \Delta_{X_N}(\boldsymbol{\mu}^*) - \Delta_{X_N \oplus \langle\{\varphi\}\rangle}(\boldsymbol{\mu}^*). \tag{11}$$

Using this indicator for selection of basis extension candidates from the dictionary in an iterative basis construction algorithm will yield ideal basis sizes.

Algorithm 1. *Given a parameter $\boldsymbol{\mu}^* \in \mathcal{P}$, an error tolerance $\varepsilon > 0$, a desired basis size $N \in \mathbb{N}$, $n = 0$ and $X_0 = \{0\}$ we now repeat the following steps to construct a parameter-fit reduced basis space $X_N(\boldsymbol{\mu}^*)$ from a precomputed dictionary $\mathcal{D}$:*

1. *Evaluate the error estimator $\Delta_{X_n}(\boldsymbol{\mu}^*)$. If $\Delta_{X_n}(\boldsymbol{\mu}^*) < \varepsilon$ or $n \geq N$ set $X_N(\boldsymbol{\mu}^*) = X_n(\boldsymbol{\mu}^*)$ and finish, else go on with Step 2.*

2. *Evaluate the indicator $\eta_\Delta(\varphi, \boldsymbol{\mu}^*; X_n)$ for all dictionary elements $\varphi \in \mathcal{D}$.*

3. *Find the dictionary element that maximizes the indicator function:*

$$\varphi_{\max} = \arg \max_{\psi \in \mathcal{D}} \eta_\Delta(\psi, \boldsymbol{\mu}^*; X_n).$$

4. *Set $n = n + 1$ and enrich the reduced space: $X_n = X_{n-1} \oplus \langle\{\varphi_{\max}\}\rangle$.*

Clearly, in a naive implementation, Step 2, which includes reduced simulation in the space $X_n \oplus \langle\{\varphi\}\rangle$ and evaluation of the error estimator for **all** dictionary elements $\varphi \in \mathcal{D}$, will be too costly to be applicable, especially for large dictionaries $\mathcal{D}$ ($\mathcal{O}(DN^4)$). We will therefore now point out how Algorithm 1 can be performed with a complexity of $\mathcal{O}(|\mathcal{D}| \cdot N^3)$ which will be favorable over the standard Greedy RB approach where the complexity is also cubic in the basis size but bases are usually a lot larger than with our method.

3.1 Simultaneous Reduced Simulation and Indicator Evaluation

As a first step for evaluation of the indicator η_Δ (11) we need to compute all reduced solutions $u_{n,\varphi} \in X_n \oplus \langle\{\varphi\}\rangle$ for all dictionary elements $\varphi \in \mathcal{D}$ and a given space X_n.

Proposition 3.1. *The solution $\mathbf{u}_{n,\varphi}(\boldsymbol{\mu})$ of Equation (2) in the space $X_n \oplus \langle\{\varphi\}\rangle$ for a given parameter $\boldsymbol{\mu} \in \mathcal{P}$ is given by*

$$\mathbf{u}_{n,\varphi}(\boldsymbol{\mu}) = \begin{pmatrix} \mathbf{u}_n \\ 0 \end{pmatrix} + \frac{\sigma(\varphi, \boldsymbol{\mu}) - \boldsymbol{\beta}(\varphi, \boldsymbol{\mu})\mathbf{u}_n}{\gamma(\varphi, \boldsymbol{\mu}) - \boldsymbol{\beta}(\varphi, \boldsymbol{\mu})\mathbf{A}_n^{-1}\boldsymbol{\alpha}(\varphi, \boldsymbol{\mu})} \cdot \begin{pmatrix} -\mathbf{A}_n^{-1}\boldsymbol{\alpha}(\varphi, \boldsymbol{\mu}) \\ 1 \end{pmatrix},$$

with suitably chosen $\mathbf{A}_n \in \mathbb{R}^{n \times n}$, $\mathbf{u}_n \in \mathbb{R}^n$ *and functions* $\boldsymbol{\alpha} : \mathcal{D} \times \mathcal{P} \to \mathbb{R}^{n \times 1}$, $\boldsymbol{\beta} : \mathcal{D} \times \mathcal{P} \to \mathbb{R}^{1 \times n}$, $\sigma, \gamma : \mathcal{D} \times \mathcal{P} \to \mathbb{R}$.

Proof. We define the matrix $\mathbf{A}_n \in \mathbb{R}^{n \times n}$ and the vectors $\mathbf{u}_n, \mathbf{b}_n \in \mathbb{R}^n$ for the space X_n as in (2). Let

$$
\begin{aligned}
(\boldsymbol{\alpha}(\varphi))_i &= (\boldsymbol{\alpha}(\varphi, \boldsymbol{\mu}))_i = B_h(\varphi_i, \varphi; \boldsymbol{\mu}), \quad 1 \le i \le n, \\
(\boldsymbol{\beta}(\varphi))_i &= (\boldsymbol{\beta}(\varphi, \boldsymbol{\mu}))_i = B_h(\varphi, \varphi_i; \boldsymbol{\mu}), \quad 1 \le i \le n, \\
\gamma(\varphi) &= \gamma(\varphi, \boldsymbol{\mu}) = B_h(\varphi, \varphi; \boldsymbol{\mu}), \\
\sigma(\varphi) &= \sigma(\varphi, \boldsymbol{\mu}) = L_h(\varphi; \boldsymbol{\mu}).
\end{aligned}
$$

for a given basis $\{\varphi_i | 1 \le i \le n\} \subset X_n$ of X_n. The projection of Equation (1) onto the space $X_n \oplus \langle \{\varphi\} \rangle$ for a given function $\varphi \in \mathcal{D}$ is then given by

$$
\mathbf{A}_{n,\varphi} \cdot \mathbf{u}_{n,\varphi} = \mathbf{b}_{n,\varphi}, \tag{12}
$$

where

$$
\mathbf{A}_{n,\varphi} = \begin{pmatrix} \mathbf{A}_n & \boldsymbol{\alpha}(\varphi) \\ \boldsymbol{\beta}(\varphi) & \gamma(\varphi) \end{pmatrix} \in \mathbb{R}^{(n+1) \times (n+1)},
$$

$$
\mathbf{b}_{n,\varphi} = \begin{pmatrix} \mathbf{b_n} \\ \sigma(\varphi) \end{pmatrix} \in \mathbb{R}^{n+1}.
$$

Multiplication of Equation (12) with the invertible block diagonal matrix $\mathrm{diag}(\mathbf{A}_n^{-1}, 1)$ then yields:

$$
\begin{aligned}
\mathbf{A}_{n,\varphi} \cdot \mathbf{u}_{n,\varphi} &= \mathbf{b}_{n,\varphi}, \\
\Leftrightarrow \begin{pmatrix} \mathbf{A}_n^{-1} & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \mathbf{A}_n & \boldsymbol{\alpha}(\varphi) \\ \boldsymbol{\beta}(\varphi) & \gamma(\varphi) \end{pmatrix} \cdot \mathbf{u}_{n,\varphi} &= \begin{pmatrix} \mathbf{A}_n^{-1} & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \mathbf{b_n} \\ \sigma(\varphi) \end{pmatrix} \\
\Leftrightarrow \begin{pmatrix} \mathbf{Id}_n & \mathbf{A}_n^{-1}\boldsymbol{\alpha}(\varphi) \\ \boldsymbol{\beta}(\varphi) & \gamma(\varphi) \end{pmatrix} \cdot \mathbf{u}_{n,\varphi} &= \begin{pmatrix} \mathbf{u}_n \\ \sigma(\varphi) \end{pmatrix} \\
\Leftrightarrow \begin{pmatrix} \mathbf{Id}_n & \mathbf{A}_n^{-1}\boldsymbol{\alpha}(\varphi) \\ 0 & \gamma(\varphi) - \boldsymbol{\beta}(\varphi)\mathbf{A}_n^{-1}\boldsymbol{\alpha}(\varphi) \end{pmatrix} \cdot \mathbf{u}_{n,\varphi} &= \begin{pmatrix} \mathbf{u}_n \\ \sigma(\varphi) - \boldsymbol{\beta}(\varphi)\mathbf{u}_n \end{pmatrix},
\end{aligned}
$$

where $\mathbf{Id}_n \in \mathbb{R}^{n \times n}$ denotes the n by n identity matrix.

Using back substitution we find the solution $\mathbf{u}_{n,\varphi}$:

$$
\begin{aligned}
(\mathbf{u}_{n,\varphi})_{n+1} &= \frac{\sigma(\varphi) - \boldsymbol{\beta}(\varphi)\mathbf{u}_n}{\gamma(\varphi) - \boldsymbol{\beta}(\varphi)\mathbf{A}_n^{-1}\boldsymbol{\alpha}(\varphi)}, \\
(\mathbf{u}_{n,\varphi})_k &= (\mathbf{u}_n)_k - (\mathbf{A}_n^{-1}\boldsymbol{\alpha}(\varphi))_k (\mathbf{u}_{n,\varphi})_{n+1}, \quad k \in \{1, \dots, n\}.
\end{aligned}
$$

Which can be rewritten in the form

$$
\mathbf{u}_{n,\varphi} = \begin{pmatrix} \mathbf{u}_n \\ 0 \end{pmatrix} + \frac{\sigma(\varphi) - \boldsymbol{\beta}(\varphi)\mathbf{u}_n}{\gamma(\varphi) - \boldsymbol{\beta}(\varphi)\mathbf{A}_n^{-1}\boldsymbol{\alpha}(\varphi)} \cdot \begin{pmatrix} -\mathbf{A}_n^{-1}\boldsymbol{\alpha}(\varphi) \\ 1 \end{pmatrix}. \tag{13}
$$

$\square$

Using Proposition 3.1, only one matrix-vector multiplication and two vector-vector multiplications are needed for the computation of one reduced solution in Step (2) in Algorithm 1. Only once per loop iteration in Algorithm 1, the matrix $\mathbf{A}_n$ needs to be inverted. The quantities $\boldsymbol{\alpha}, \boldsymbol{\beta}, \gamma, \sigma$ and $\mathbf{A}_n$ can be extracted from $\mathbf{A}_D(\boldsymbol{\mu}), \mathbf{b}_D(\boldsymbol{\mu})$ where

$$\mathbf{A}_D(\boldsymbol{\mu}) = \sum_{q=1}^{Q_B} \Theta_B^q(\boldsymbol{\mu}) \mathbf{A}_D^q \in \mathbb{R}^{D \times D}, \quad \mathbf{b}_D^q(\boldsymbol{\mu}) = \sum_{q=1}^{Q_L} \Theta_L^q(\boldsymbol{\mu}) \mathbf{b}_D^q \in \mathbb{R}^D. \tag{14}$$

Here $\{\mathbf{A}_D^q | 1 \leq q \leq Q_B\}$, $\{\mathbf{b}_D^q | 1 \leq q \leq Q_L\}$ are the precomputed quantities from Section 2.

In the sequel, we will outline how to efficiently evaluate the indicator function η_Δ for all possible extensions in Step (2) of Algorithm 1. When evaluating the indicator $\eta_\Delta(\varphi, \boldsymbol{\mu}^*; X_n)$ for all dictionary elements φ we need to evaluate $\Delta_{X_n \oplus \langle \{\varphi\} \rangle}(\boldsymbol{\mu}^*)$ for all $\varphi \in \mathcal{D}$. The next proposition proofs that these values can be computed simultaneously for the whole dictionary.

Proposition 3.2. *For suitable choice of matrices* $\mathbf{g}_1 \in \mathbb{R}^{(n+1) \times D}$, $\mathbf{g}_2 \in \mathbb{R}^{1 \times D}$ *the vector* $\Delta \in \mathbb{R}^D$ *with*

$$\Delta = \begin{pmatrix} 1, & \cdots, & 1 \end{pmatrix} \cdot \left(\begin{pmatrix} 1 & \cdots & 1 \\ -(\mathbf{u}_{n,\varphi_1})_1 & \cdots & -(\mathbf{u}_{n,\varphi_D})_1 \\ \vdots & \ddots & \vdots \\ -(\mathbf{u}_{n,\varphi_1})_{n+1} & \cdots & -(\mathbf{u}_{n,\varphi_D})_{n+1} \end{pmatrix} \circ \begin{pmatrix} \mathbf{g}_1 \\ \mathbf{g}_2 \end{pmatrix} \right)$$

contains the squared error estimators for all possible basis extensions:

$$\Delta = \begin{pmatrix} \Delta_{X_n \oplus \langle \{\varphi_1\} \rangle}(\boldsymbol{\mu}^*), & \cdots, & \Delta_{X_n \oplus \langle \{\varphi_D\} \rangle}(\boldsymbol{\mu}^*) \end{pmatrix}. \tag{15}$$

Here we used the Hadamard *product* $\mathbf{M} \circ \mathbf{N} \in \mathbb{R}^{m \times n}$ *of two matrices* $\mathbf{M}, \mathbf{N} \in \mathbb{R}^{m \times n}$, $(\mathbf{M} \circ \mathbf{N})_{i,j} = \mathbf{M}_{i,j} \cdot \mathbf{N}_{i,j}$.

Proof. Given a parameter $\boldsymbol{\mu}^* \in \mathcal{P}$ we define

$$\mathbf{S}(\boldsymbol{\mu}^*) = \begin{pmatrix} \Theta_L^1(\boldsymbol{\mu}^*) & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \Theta_L^{Q_L}(\boldsymbol{\mu}^*) & 0 & \cdots & 0 \\ 0 & & & \\ \vdots & & \mathbf{C}(\boldsymbol{\mu}^*) & \\ 0 & & & \end{pmatrix} \in \mathbb{R}^{(Q_L + Q_B \cdot D) \times (D+1)},$$

where the coefficient matrix $\mathbf{C}(\boldsymbol{\mu}^*) \in \mathbb{R}^{(Q_B \cdot D) \times D}$ is given as

$$\mathbf{C}(\boldsymbol{\mu}^*) = \begin{pmatrix} \boldsymbol{\Theta}_B(\boldsymbol{\mu}^*) & & 0 \\ & \ddots & \\ 0 & & \boldsymbol{\Theta}_B(\boldsymbol{\mu}^*) \end{pmatrix},$$

with $\boldsymbol{\Theta}_B(\boldsymbol{\mu}^*) \in \mathbb{R}^{Q_B}$, $(\boldsymbol{\Theta}_B(\boldsymbol{\mu}^*))_k = \Theta_B^k(\boldsymbol{\mu}^*)$. Using $\mathbf{S}(\boldsymbol{\mu}^*)$ we define the matrix

$$\overline{\mathbf{G}} = \overline{\mathbf{G}}(\boldsymbol{\mu}^*) = \mathbf{S}(\boldsymbol{\mu}^*)^\intercal \cdot \mathbf{G} \cdot \mathbf{S}(\boldsymbol{\mu}^*) \in \mathbb{R}^{(D+1)\times(D+1)}, \tag{16}$$

with $\mathbf{G}$ as computed in Section 2. For the exposition of the rest of the simultaneous indicator evaluation we need some additional notation:

- Given a set of indices $I = [i_1, \ldots, i_m] \subset \mathbb{N}$ we define $I \Vdash l := [i_1 + l, \ldots, i_m + l]$ for $l \in \mathbb{N}$.

- Given a set of indices $I = [i_1, \ldots, i_m] \subset \mathbb{N}$ we define the set of indices $[I, l] \subset \mathbb{N}$: $[I, l] := [i_1, \ldots, i_m, l]$ for $l \in \mathbb{N}$.

- Given a matrix $\mathbf{M}$ and two sets of indices $I = [i_1, \ldots, i_{|I|}] \subset \mathbb{N}$, $J = [j_1, \ldots, j_{|J|}] \subset \mathbb{N}$ we define the matrix $\mathbf{M}_{I,J} \in \mathbb{R}^{|I|\times|J|}$

$$(\mathbf{M}_{I,J})_{k,l} := \mathbf{M}_{i_k, j_l}.$$

Assume a basis $\Phi \subset \mathcal{D}$ of the space X_n to be given. Let $I_\Phi \subset \mathbb{N}$ be an index set for Φ and $I_\mathcal{D} \subset \mathbb{N}$ be an index set for $\mathcal{D}$. Additionally we use the vectors $\mathbf{u}_{n,\varphi} = \mathbf{u}_{n,\varphi}(\boldsymbol{\mu}^*)$ from Section 3.1.

Using the above notation we can define

$$\mathbf{g}_1 = \overline{\mathbf{G}}_{[1, I_\Phi + 1], [1, I_\Phi + 1]} \cdot \begin{pmatrix} 1 & \cdots & 1 \\ -(\mathbf{u}_{n,\varphi_1})_1 & \cdots & -(\mathbf{u}_{n,\varphi_D})_1 \\ \vdots & \ddots & \vdots \\ -(\mathbf{u}_{n,\varphi_1})_n & \cdots & -(\mathbf{u}_{n,\varphi_D})_n \end{pmatrix}$$

$$+ \overline{\mathbf{G}}_{[1, I_\Phi + 1], I_\mathcal{D} + 1} \cdot \begin{pmatrix} -(\mathbf{u}_{n,\varphi_1})_{n+1} & & 0 \\ & \ddots & \\ 0 & & -(\mathbf{u}_{n,\varphi_1})_{n+1} \end{pmatrix},$$

$$\mathbf{g}_2 = \begin{pmatrix} 1 & \cdots & 1 \end{pmatrix} \cdot \left(\overline{\mathbf{G}}^\intercal_{I_\mathcal{D}+1, [1, I_\Phi + 1]} \circ \begin{pmatrix} 1 & \cdots & 1 \\ -(\mathbf{u}_{n,\varphi_1})_1 & \cdots & -(\mathbf{u}_{n,\varphi_D})_1 \\ \vdots & \ddots & \vdots \\ -(\mathbf{u}_{n,\varphi_1})_n & \cdots & -(\mathbf{u}_{n,\varphi_D})_n \end{pmatrix} \right)$$

$$+ \begin{pmatrix} \overline{\mathbf{G}}_{2,2}, & \cdots, & \overline{\mathbf{G}}_{D+1,D+1} \end{pmatrix} \circ \begin{pmatrix} -(\mathbf{u}_{n,\varphi_1})_{n+1}, & \cdots, & -(\mathbf{u}_{n,\varphi_D})_{n+1} \end{pmatrix}.$$

Using this definition of $\mathbf{g}_1$ and $\mathbf{g}_2$, one can show by performing all remaining multiplications that the vector Δ as defined in the proposition indeed represents the desired error estimators. $\qquad \square$
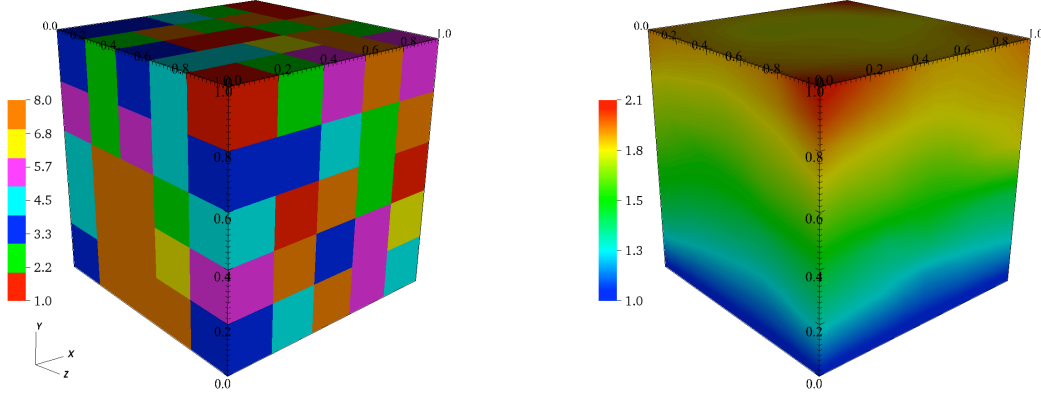
Figure 1: Heat diffusion coefficient $\lambda(\boldsymbol{\mu}) : \Omega \to \mathbb{R}$ (left), solution $u_h(\boldsymbol{\mu})$ for $\boldsymbol{\mu} = (1, 2, 3, 4, 5, 6, 7, 8)$ (right).

4   Experiments

In this section we present some preliminary numerical results for the method introduced in this paper. All tests were performed using our C++ library DUNErb, based on the *Distributed and Unified Numerics Environment* (DUNE). Both packages can be found online[1].

For our tests, we solve the heat equation on the unit cube $\Omega = [0, 1]^3$. The problem statement is as follows: Find $u(\boldsymbol{\mu}) \in H_0^1(\Omega)$ such that

$$
\begin{aligned}
-\nabla \cdot (\lambda(\mathbf{x})\nabla u(\mathbf{x})) &= 1 && \text{in } \Omega, \\
u(\mathbf{x}) &= 0 && \text{on } \Gamma_D = [0, 1] \times 0 \times [0, 1], \\
\lambda(\mathbf{x})\nabla u(\mathbf{x}) \cdot \mathbf{n} &= 0 && \text{on } \partial\Omega \setminus \Gamma_D,
\end{aligned}
\tag{17}
$$

where $\boldsymbol{\mu} \in \mathcal{P} = (0, 10]^8$. The heat diffusion coefficient $\lambda(\boldsymbol{\mu}) : \Omega \to \mathbb{R}$ has the form

$$
\lambda(\boldsymbol{\mu})(\mathbf{x}) = \sum_{i=1}^{8} (\boldsymbol{\mu})_i \cdot \chi_i(\mathbf{x}),
$$

where, as usual, $(\boldsymbol{\mu})_i \in \mathbb{R}$ denotes the i-th component of the vector $\boldsymbol{\mu}$ and the functions $\chi_i : \Omega \to \{0, 1\}$ denote the characteristic functions for the eight subdomains of Ω sketched in the left plot in Figure 1. Furthermore, the right plot shows a typical solution for a given parameter.

4.1   Offline Phase

We discretize Ω using 1000 cubes, X_h is a linear discontinuous galerkin space with 4000 degrees of freedom. The training set $\mathcal{S} \subset \mathcal{P}$ is given by a lognormal distribution

---

[1]`http://users.dune-project.org/`
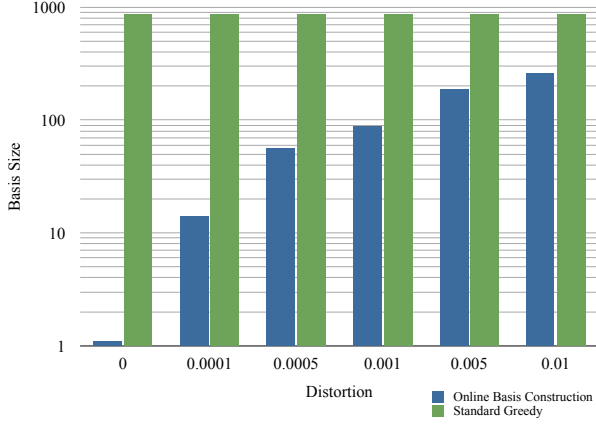
Figure 2: Mean basis size N during the online phase for the standard Greedy method and our online basis construction algorithm for different values of ρ
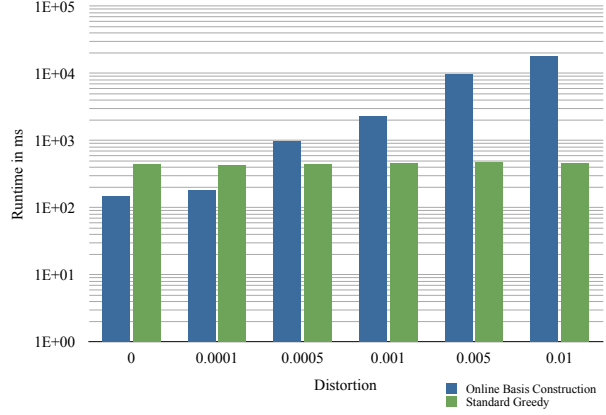
Figure 3: Mean total runtime during the online phase for the standard Greedy method and our online basis construction algorithm for different values of ρ

centered at 2 in each component. We generate a traditional [6] Greedy-basis Φ_G with a training set $\mathcal{S}_G$ with $|\mathcal{S}_G| = 1000$ and a tolerance of 10^{-5} and a dictionary $\mathcal{D}$ using the approach described in Section 2 using a training set $\mathcal{S}_\mathcal{D} \supset \mathcal{S}_G$ with $|\mathcal{S}_\mathcal{D}| = 2000$. While the generation of the traditional Greedy-basis takes more than 9 hours and produces about 600 megabytes of data, the generation of the dictionary takes only one hour but produces about 1.1 gigabytes of data.

4.2 Online Phase

Using both the basis Φ_G and the dictionary $\mathcal{D}$ we run online simulations on the test set $\mathcal{T} = \mathcal{S}_G + \rho\mathcal{R}$ where $\mathcal{R}$ contains random numbers in $(0, 1]^8$ and $\rho \in \mathbb{R}$ denotes a distortion scale. For our algorithm, we use the same error tolerance as for the standard Greedy algorithm: $\varepsilon = 10^{-5}$.

Figure 2 shows the resulting basis sizes N for the standard Greedy method (which is fixed by the basis construction during the offline phase, here: $N = 871$) and the basis size N resulting from Algorithm 1 for different values of ρ. We see that, especially for small disturbances of the training parameters, our online basis generation algorithm yields substantially smaller bases.

For small disturbances ρ this pays out in terms of runtime: Figure 3 shows mean online runtimes for the two algorithms and different values of ρ. This runtime includes reduced simulation, error estimation and, for our algorithm, the time needed for basis construction. Beginning with distortions in the range of $5 \cdot 10^{-5}$ our algorithm is slower than the standard approach as we then need lots of basis enrichment iterations in Algorithm 1. Still, it pays out to use our algorithm even in these cases as we fulfill the error bound in all cases while the standard Greedy method violates the error tolerance for the cases $\rho = 5 \cdot 10^{-3}$ (error: $\max_{\boldsymbol{\mu}\in\mathcal{T}} \Delta_{X_N}(\boldsymbol{\mu}) = 1.64 \cdot 10^{-5}$) and $\rho = 1 \cdot 10^{-2}$ (error: $\max_{\boldsymbol{\mu}\in\mathcal{T}} \Delta_{X_N}(\boldsymbol{\mu}) = 3.28 \cdot 10^{-5}$).

5 Outlook

In our future work we will investigate two different dictionary construction algorithms: the "Offline–Greedy"-algorithm and the "Randomized Offline–Greedy"-algorithm. The "Offline–Greedy"-algorithm will use Algorithm 1 with a small maximum basis size N during the offline phase to iteratively find the parameter $\boldsymbol{\mu}$ worst approximated in the current dictionary and enrich the dictionary with $u_h(\boldsymbol{\mu})$. This algorithm will hopefully build up a dictionary that combines good approximation quality with small online basis sizes, even for large distortions ρ.

REFERENCES

[1] Felix Albrecht, Bernard Haasdonk, Sven Kaulmann, and Mario Ohlberger. The Localized Reduced Basis Multiscale Method. In Angela Handlovičová, Zuzana Minarechová, and Daniel Ševčovič, editors, *Algoritmy 2012*, pages 393–403. Slovak University of Technology in Bratislava, Publishing House of STU, April 2012.

[2] Jeanine Bernlöhr. Online Reduzierte Basis Generierung für parameterabhängige elliptische partielle Differentialgleichungen. *Diploma Thesis, University of Stuttgart*, June 2012.

[3] Bernard Haasdonk and Mario Ohlberger. Reduced basis method for finite volume approximations of parametrized linear evolution equations. *M2AN. Mathematical Modelling and Numerical Analysis*, 42(2):277–302, 2008.

[4] Bernard Haasdonk and Mario Ohlberger. Reduced basis method for explicit finite volume approximations of nonlinear conservation laws. In *Hyperbolic problems: theory, numerics and applications*, pages 605–614. Amer. Math. Soc., Providence, RI, 2009.

[5] Yvon Maday and Benjamin Stamm. Locally adaptive greedy approximations for anisotropic parameter reduced basis spaces. *arXiv.org*, math.NA, April 2012.

[6] Anthony T. Patera and Gianluigi Rozza. Reduced Basis Approximation and a Posteriori Error Estimation for Parametrized Partial Differential Equations. *Version 1.0, Copyright MIT 2006, to appear in (tentative rubric) MIT Pappalardo Graduate Monographs in Mechanical Engineering*.

[7] Vladimir Temlyakov. *Greedy approximation*, volume 20 of *Cambridge Monographs on Applied and Computational Mathematics*. Cambridge University Press, Cambridge, 2011.

QUANTIFYING THE COMPUTABILITY OF THE LORENZ SYSTEM

Benjamin Kehlet[*][†] **and Anders Logg**[†][*]

[*]Department of Informatics, University of Oslo
P.O.Box 1080 Blindern, 0316 OSLO,
Norway

[†]Center for Biomedical Computing at Simula Research Laboratory,
P.O.Box 134, 1325 Lysaker,
Norway

Key words: Lorenz, chaos, high precision, high order, finite element, time stepping

Abstract. It is well known that the computation of accurate trajectories of the Lorenz system is a difficult problem. Computed solutions are very sensitive to the discretization error determined by the time step size and polynomial order of the method, as well as round-off errors.

In this work, we show how round-off errors limit the computability of the Lorenz system and quantify exactly the length of intervals over which solutions can be computed, expressed in terms of the floating point precision. Using adjoint-based *a posteriori* error analysis techniques, we estimate the stability of computations with respect to initial data, discretization, and round-off errors, respectively.

The analysis is verified by computing an accurate solution on the time interval $[0, 1000]$ using a very high order (order 200) finite element method and very high floating point precision (400 digits).

1 Introduction

In a classic paper from 1963 [13], Edward Lorenz studied the computability of a simple system of three ordinary differential equations,

$$\begin{cases} \dot{x} = \sigma(y - x), \\ \dot{y} = rx - y - xz, \\ \dot{z} = xy - bz, \end{cases} \tag{1}$$

where $\sigma = 10$, $b = 8/3$, and $r = 28$. Lorenz computed numerical solutions of the system (1) and found the solutions to be very sensitive to changes in initial data. The

equations had been devised by Lorenz as a simple model of atmospheric flow, based on a truncated Fourier expansion of the partial differential equations governing Rayleigh–Bénard convection [14, 12, 15]. In his paper, Lorenz computed solutions on the interval $[0, 60]$. As we shall see below, the Lorenz system is not computable on the equipment that was available to Lorenz in 1963 beyond time $T \approx 25$.

It is known that, given enough resources, the Lorenz system is computable over arbitrarily long time intervals. However, one may easily (and falsely) come to the conclusion that the Lorenz system is not computable, or computable only over very short time intervals, either by numerical experiments or by a simplistic analysis. Indeed, a standard *a priori* error estimate indicates that the growth rate of the error is

$$\|e(T)\| \leq Ce^{LT}\epsilon, \tag{2}$$

where $\|e(T)\|$ denotes some norm of the error at the final time T, L is the Lipschitz constant of (1), and ϵ is the size of the residual or local truncation error in a numerical solution of (1). The Lipschitz constant is of size $L \approx 33$ which indicates that solutions are not computable beyond $T \approx 1.1$, even if the residual is close to machine precision ($\epsilon_{\mathrm{mach}} \sim 10^{-16}$ on most computers).[1] However, the estimate (2) is overly pessimistic; it is well known that solutions of the Lorenz system may be computed on short time intervals. In fact, one may easily compute accurate solutions over time intervals of length $T = 25$ with any standard ODE solver.

In [3], it was demonstrated that the Lorenz system is indeed computable on intervals of moderate length ($T = 30$) on a standard desktop computer. The computability of the Lorenz system was linked to the growth of a *stability factor* in an *a posteriori* estimate of the error at the final time. It was shown that the growth rate of the stability factor is non-constant. On average the growth is exponential but with a rate much smaller than indicated by (2).

In [10], the computability of the Lorenz system was further extended to $T = 48$ using high order ($\|e(T)\| \sim \Delta t^{30}$) finite element methods. As we shall see below, this is the "theoretical limit" for computations with 16 digit precision. Solutions over longer time intervals have been computed based on shadowing (the existence of a nearby exact solution), see [2], but for unknown initial data. Other related work on high-precision numerical methods applied to the Lorenz system include [17] and [5]. For an overview of some recent results obtained with high-precision numerical methods, we also refer to [1].

In this paper, we study and quantify the computability of the Lorenz system. In particular we answer the following fundamental question: *How far is the Lorenz system computable for a given machine precision?*

As we shall see, obtaining a sequence of converging approximations for the solution of the Lorenz system is non-trivial. In particular, such a sequence of solutions cannot be

---

[1]The value of the Lipschitz constant was computed as the maximum l^2-norm of the Jacobian $J = \partial f / \partial u$ of the right-hand side f of the Lorenz system over the interval $[0, 1000]$.

obtained by simply decreasing the size of the time step; see for example [16]. This has led to misconceptions regarding the computability of the Lorenz system; see for example [18]. To obtain a sequence of converging solutions, one must also control the effect of round-off errors. This was also noted by Lorenz [11] in a response to [18].

In this manuscript, we define computability as the length T of the maximum time interval $[0, T]$ on which a solution is computable to within a given precision $\epsilon > 0$ using a given machine precision $0 < \epsilon_{\text{mach}} < \epsilon$; that is, the maximum T such that $\inf_U \|u - U\|_{L^\infty(0,T;l^\infty)} \leq \epsilon$, where the infimum is taken over all numerical approximations U of the exact solution u computed with some time-stepping method and machine precision ϵ_{mach} (as made more precise in Section 3). If the computability $T_\epsilon = T_\epsilon(\epsilon_{\text{mach}})$ does not depend strongly on ϵ, we write $T = T(\epsilon_{\text{mach}})$. As we shall see, this is the case for the Lorenz system as a result of exponential growth of errors as function of the final time T. The definition of computability $T(\epsilon_{\text{mach}})$ is closely related to the definition of a critical predictable time T_c in [8] and the definition of a decoupling time $\hat{T}$ in [16].

2 Numerical method and implementation

We consider the numerical solution of general initial value problems for systems of ordinary differential equations,

$$\begin{aligned} \dot{u}(t) &= f(u(t), t), \quad t \in (0, T], \\ u(0) &= u_0. \end{aligned} \tag{3}$$

The right-hand side $f : \mathbb{R}^N \times [0, T] \to \mathbb{R}^N$ is assumed to be Lipschitz continuous in u and continuous in t. Our objective is to analyze the error in an approximate solution $U : [0, T] \to \mathbb{R}^N$, for example a numerical solution of the Lorenz system.

The continuous and discontinuous Galerkin methods cG(q) and dG(q) are formulated by requiring that the residual $R = \dot{U} - f(U, \cdot)$ be orthogonal to a suitable space of test functions. By making a piecewise polynomial Ansatz, the solution may be computed on a sequence of intervals partitioning the computational domain $[0, T]$ by solving a system of equations for the degrees of freedom on each consecutive interval. For a particular choice of numerical quadrature and degree q, the cG(q) and dG(q) methods both reduce to standard implicit Runge–Kutta methods.

In the case of the cG(q) method, the numerical solution U is a continuous piecewise polynomial of degree q that on each interval $[t_{n-1}, t_n]$ satisfies $\int_{t_{n-1}}^{t_n} R \, v \, dt = 0$ for all $v \in \mathcal{P}^{q-1}([t_{n-1}, t_n])$.

The results were obtained using the finite element package DOLFIN [9] version 0.9.2 together with the multi-precision library GMP [4]. For a detailed discussion on the implementation, we refer to [7]. The source code as well as scripts to reproduce all results presented in this manuscript are available on request.

3 Error analysis

The error analysis is based on the solution of an auxiliary *dual problem*. The dual (adjoint) problem takes the form of an initial value problem for a system of linear ordinary differential equations,

$$
\begin{aligned}
-\dot{z}(t) &= \bar{A}^{\top}(t)z(t), \quad t \in [0, T), \\
z(T) &= z_T.
\end{aligned}
\tag{4}
$$

Here, $\bar{A}(t) = \int_0^1 \frac{\partial f}{\partial u}(sU(t) + (1-s)u(t), t)\, \mathrm{d}s$ denotes the Jacobian matrix of the right-hand side f averaged over the approximate solution U and the exact solution u.

The Lorenz system is quadratic in the primal variable u. Hence, the average in $\bar{A}$ corresponds to evaluating the Jacobian matrix at the midpoint between the two vectors $U(t)$ and $u(t)$. It follows that the dual problem of the Lorenz system is

$$
\begin{cases}
-\dot{\xi} = -\sigma\xi + (r - \bar{z})\eta + \bar{y}\zeta, \\
-\dot{\eta} = \sigma\xi - \eta + \bar{x}\zeta, \\
-\dot{\zeta} = -\bar{x}\eta - b\zeta,
\end{cases}
\tag{5}
$$

where $z = (\xi, \eta, \zeta)$ denotes the dual solution and $(\bar{x}, \bar{y}, \bar{z}) = (U + u)/2$.

In [6], we prove the following *a posteriori* error estimate:

Theorem 3.1 (Error estimate). *Let* $u : [0, T] \to \mathbb{R}^N$ *be the exact solution of* (3) (*assuming it exists*), *let* $z : [0, T] \to \mathbb{R}^N$ *be the solution of* (4), *and let* $U : [0, T] \to \mathbb{R}^N$ *be any piecewise smooth approximation of* u *on a partition* $0 = t_0 < t_1 < \cdots < t_M = T$ *of* $[0, T]$, *that is,* $U|_{(t_{m-1}, t_m]} \in \mathcal{C}^\infty((t_{m-1}, t_m])$ *for* $m = 1, 2, \ldots, M$ *(U is left-continuous).*

Then, for any $p \geq 0$, *the following error estimate holds:*

$$
\langle z_T, U(T) - u(T) \rangle = \mathbf{E}_D + \mathbf{E}_G + \mathbf{E}_C,
$$

where

$$
\begin{aligned}
|\mathbf{E}_D| &\leq S_D \, \|U(0) - u(0)\|, \\
|\mathbf{E}_G| &\leq S_G \, C_p \max_{[0,T]} \left\{ \Delta t^{p+1}(\|[U]\|/\Delta t + \|R\|) \right\}, \\
|\mathbf{E}_C| &\leq S_C \, C_p' \max_{[0,T]} \|\Delta t^{-1}\bar{R}\|,
\end{aligned}
$$

where C_p *and* C_p' *are constants depending only on* p. *The stability factors* S_D, S_G, *and* S_C *are defined by*

$$
S_D = \|z(0)\|, \quad S_G = \int_0^T \|z^{(p+1)}\|\, \mathrm{d}t, \quad S_C = \int_0^T \|\pi z\|\, \mathrm{d}t.
$$

Furthermore the following bound for the computational error is proved in [6]:

Theorem 3.2. *Assume that the round-off error is a random variable of size $\pm\epsilon_{mach}$ with equal probabilities. Then the root-mean squared expected computational error $\mathbf{E}_C$ of Theorem 3.1 is bounded by*

$$(E[\mathbf{E}_C^2])^{1/2} \leq S_{C_2} \sqrt{C'_p} \frac{\epsilon_{mach}}{\min_{[0,T]} \sqrt{\Delta t}},$$

where $S_{C_2} = \left(\int_0^T \|\pi z\|^2 \, dt \right)^{1/2}$ and C'_p is a constant depending only on p.

We note that the computational error (accumulated round-off error) is inversely proportional to (the square root of) the time step; that is, a *smaller* time step yields a *larger* accumulated round-off error.
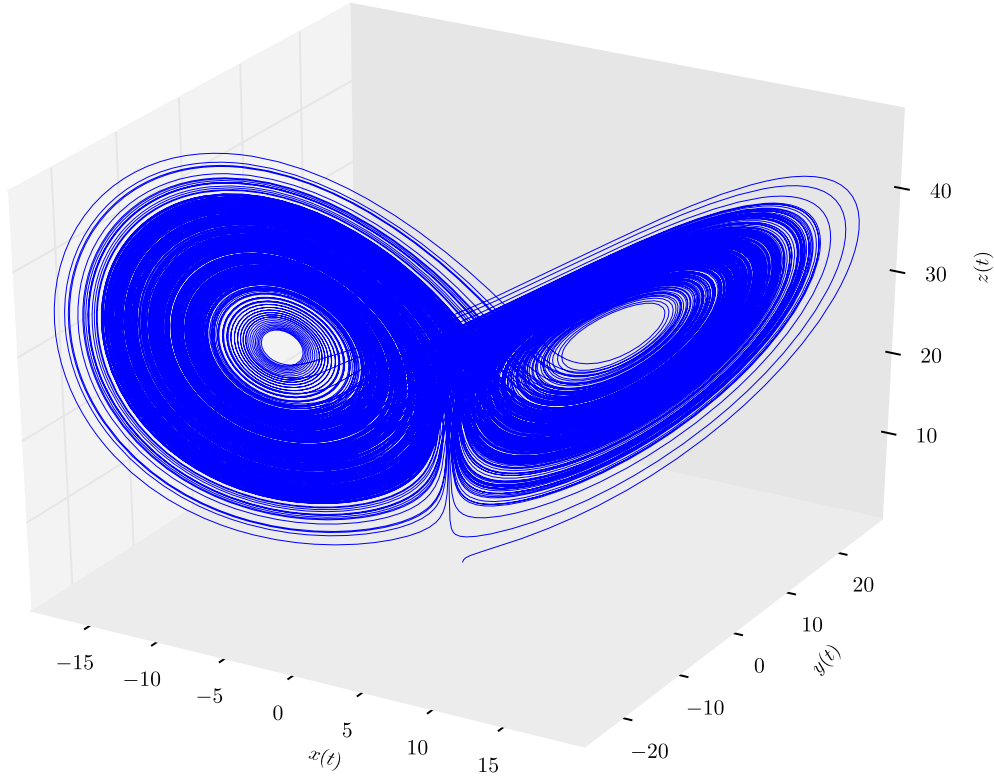
Figure 1: Phase portrait of the solution of the Lorenz system on the time interval $[0, 1000]$ for $u(0) = (1, 0, 0)$.

4 Numerical results

In this section, we present numerical results in support of Theorem 3.1 and Theorem 3.2.

4.1 Solution of the Lorenz system

The phase portrait of the solution of the Lorenz system is plotted in Figure 1. The solution was computed with cG(100), which is a method of order $2q = 200$, a time step of size $\Delta t = 0.0037$, 420-digit precision arithmetic[2], and a tolerance for the discrete residual of size $\epsilon_{\mathrm{mach}} \approx 2.26 \cdot 10^{-424}$. The solution trajectory revolves around one of the two unstable fixed points $P_{\pm} = (\pm 6\sqrt{2}, \pm 6\sqrt{2}, 27)$ for a while and then, seemingly at random, jumps to the other fixed point. Phase portraits ("attractors") resembling the phase portrait of Figure 1 are commonly displayed in most books on dynamical systems and chaos theory. However, in one way the phase portrait of Figure 1 is significantly different. It is the phase portrait of a well-defined dynamical system, namely the Lorenz system (1) with initial condition $(1, 0, 0)$, not the result of an unspecified discrete map which includes both the effect of a particular time-stepping scheme and the unknown effect of round-off errors.

To verify the computed solution, we perform a simple experiment where we compute the solution with methods of increasing order. The time step is fixed ($\Delta t = 0.0037$) and so is the arithmetic precision (420 digits). By Theorem 3.1, we expect the discretization error $\mathbf{E}_G$ to decrease exponentially with increasing order while the computational error $\mathbf{E}_C$ remains bounded. The error should therefore decrease, until $\mathbf{E}_G < \mathbf{E}_C$. Since no analytic solution or other reference solution is available, we compare the cG(10) solution with the cG(20) solution and conclude that when the two solutions no longer agree to within some tolerance (here 10^{-16}), the cG(10) solution is no longer accurate. The same experiment is repeated for cG(20/30), cG(30/40), ..., cG(90/100), cG(99/100). The solutions are displayed in Figure 2. The results indicate that the cG(99) solution is accurate on the time interval $[0, 1025]$. Alone, this does not prove that the cG(99) is accurate at time $T = 1025$. However, together with the error estimate of Theorem 3.1 and the numerically computed values of the stability factors presented below, there is strong evidence that the solution is accurate over $[0, 1025]$.

We emphasize that similar results may be obtained with other numerical methods and other software. In particular, Theorem 3.1 shows that the solution is computable with any solver that (i) discretizes the equations with high order and (ii) solves the discrete equations with high precision. The authors are aware of two such solvers: the DOLFIN solver used in this work and Taylor [5]. The full reference solution is available on request.

---

[2]The requested precision from GMP was 420 digits. The actual precision is somewhat higher depending on the number of significant bits chosen by GMP.
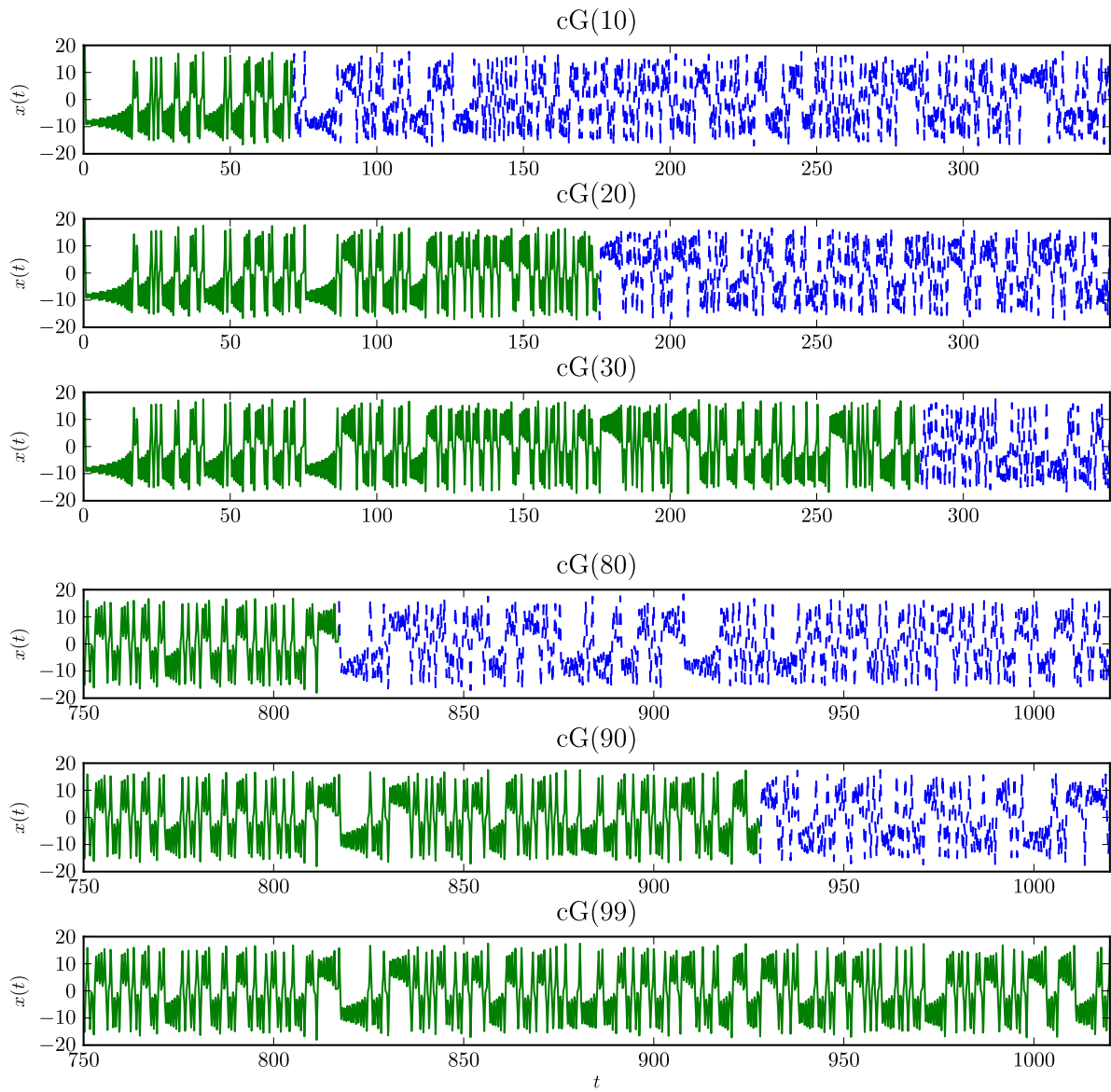
Figure 2: Computed numerical solutions (x-component) for the Lorenz system with methods of increasing order, starting at cG(10) (a method of order 20) and increasing up to cG(99) (a method of order 198).

S_D	S_G	S_C
$0.510 \cdot 10^{388}$	$28.9 \cdot 10^{388}$	$2.08 \cdot 10^{388}$

Table 1: Size of the stability factors S_D, S_G (for cG(1)), and S_C at $T = 1000$.
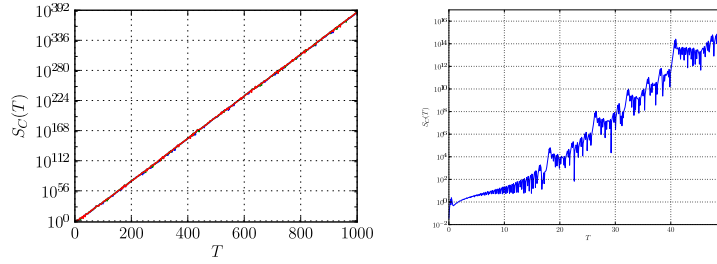
Figure 3: Growth of the stability factors S_C on the time interval $[0, 1000]$ (left) and $[0, 50]$ right.

4.2 Dual solution and stability factors

The dual solution grows exponentially backward in time. The size of the dual solution at time $t = 0$ is $S_D = \|z(0)\| \approx 0.510 \cdot 10^{388}$. By Theorem (3.1), it follows that perturbations in initial data for the Lorenz system are amplified by a factor 10^{388} at time $T = 1000$. The amplification of round-off errors may be estimated similarly by integrating the norm of the dual solution over the time interval. One finds that $S_C = \int_0^T \|\pi z\| \, dt \approx 2.08 \cdot 10^{388}$, which is the amplification of errors caused by finite precision arithmetic. The stability factor for discretization errors depends on the numerical method and in the case of the cG(1) method, one finds that $S_G = \int_0^T \|\dot{z}\| \, dt \approx 28.9 \cdot 10^{388}$. This is summarized in Table 1.

By repeatedly solving the dual problem on time intervals of increasing size, it is possible to examine the growth of the stability factors as function of the end time T. The result is displayed in Figure 3. Note that each data point (T, S) in Figure 3 corresponds to a solution of the dual problem on the interval $[0, T]$.

By Figure 3, it is evident that the stability factors grow exponentially with the end time T. On $[0, 1000]$, the growth of the stability factor(s) may be approximated by

$$S(T) \sim 10^{0.388T} \sim 10^{0.4T}. \tag{6}$$

The rate of growth is very stable and it is therefore reasonable to extrapolate beyond time $T = 1000$ to predict the computability of the Lorenz system on $[0, \infty)$. We return to this question below in Section 5.

A growth rate of $10^{0.388T}$ is far below the growth rate e^{33T} indicated by the simple analytic *a priori* error estimate (2). A close inspection of the growth of the stability factor S_C (Figure 3) explains the discrepancy between the two estimates. The growth rate of the
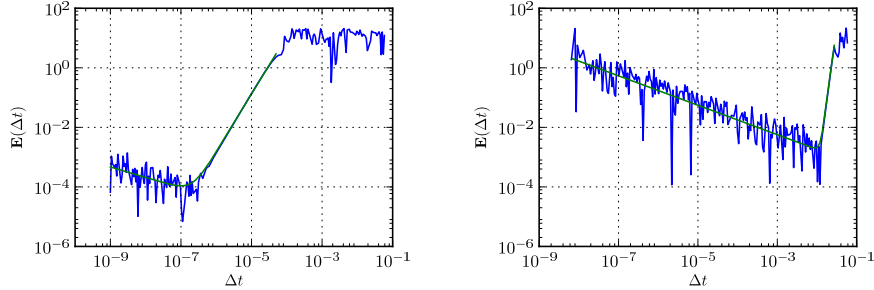
Figure 4: Error at time $T = 30$ for the cG(1) solution (left) and at time $T = 40$ for the cG(5) solution (right) of the Lorenz system. The slopes of the green lines are $-0.35 \approx -1/2$ and $1.95 \approx 2$ for the cG(1) method. For the cG(5) method, the slopes are $-0.49 \approx -1/2$ and $10.00 \approx 10$.

stability factor is not constant; it is not even monotonically increasing. While it sometimes grows very rapidly, the average growth rate is much smaller. The analytic *a priori* estimate must account for the worst case growth rate and will therefore overestimate the rate of error accumulation by a large margin.

4.3 Error propagation

We conclude this section by examining how the error depends on the size of the time step Δt. In Section 3, we found that the discretization error $\mathbf{E}_G$ scales like Δt^{2q} for the cG(q) method. On the other hand, we expect the computational error $\mathbf{E}_C$ to scale like $\Delta t^{-1/2}$. Since initial data is represented with very high precision, we have $\mathbf{E} \approx \mathbf{E}_G + \mathbf{E}_C \sim \Delta t^{2q} + \Delta t^{-1/2}$. We thus expect the error to decrease when the time step is decreased, at least initially. However, at the point where $\mathbf{E}_G = \mathbf{E}_C$, the computational error will start to dominate and we expect to see the error *increase* with decreasing time step. This is confirmed by the results presented in Figure 4, which also confirm the convergence rates $\mathbf{E}_G \sim \Delta t^{2q}$ and $\mathbf{E}_C \sim \Delta t^{-1/2}$. We also note that the error remains bounded for large values of Δt; the numerical solution stays close to the attractor but in the wrong place.

5 Computability of the Lorenz system

5.1 A model for the computability of the Lorenz system

Based on the analysis of Section 3 and the numerical results of Section 4, we develop a model for the computability of the Lorenz system. We consider the cG(q) method and make the following Ansatz for the error at the final time T as function of the time step Δt, the polynomial degree q, and the precision ϵ_{mach},

$$\mathbf{E} = \left[C_1^{[q]} \| U(0) - u(0) \| + C_2^{[q]} \Delta t^\alpha + C_3^{[q]} \Delta t^\beta \epsilon_{\mathrm{mach}} \right] \cdot 10^{0.388T}.$$

q	2	3	4	5
α	4.04	5.46	8.15	10.00
β	-0.47	-0.50	-0.50	-0.49

Table 2: Values of the constants α and β as function of q at time $T = 40$.

q	2	3	4	5
$C_2^{[q]}$	0.000356	0.000135	0.000032	0.000007
$C_3^{[q]}$	0.0031	0.0036	0.0042	0.0048

Table 3: Values of the constants $C_2^{[q]}$ and $C_3^{[q]}$ as function of q.

To determine the constants α, β, $C_1^{[q]}$, $C_2^{[q]}$, and $C_3^{[q]}$, we repeat the experiment of Figure 4 for $q = 2, 3, 4, 5$ on the interval $[0, 40]$ using the cG(100) solution as a reference. The constants α and β may be determined by a least-squares fitting of a linear polynomial to the regime where the error is dominated by the discretization error $\mathbf{E}_G$ or the computational error $\mathbf{E}_C$, respectively. The results are given in Table 2. As expected, we find that $\alpha \approx 2q$. Furthermore, we find that $\beta \approx -1/2$ in agreement with Theorem 3.2.

Next, we fix the constants $\alpha = 2q$ and $\beta = -1/2$ and determine the constants $C_1^{[q]}$, $C_2^{[q]}$, and $C_3^{[q]}$ as function of q. In Section 3, we found that $S_D(T) = \|z(0)\| \approx 0.510 \cdot 10^{0.388T}$; hence $C_1^{[q]} \approx 0.5$. By fitting curves of the form $C_2^{[q]} \Delta t^{2q} \cdot 10^{0.388T}$ and $C_3^{[q]} \Delta t^{-1/2} \cdot 10^{0.388T}$ to the two regimes where either $\mathbf{E}_G$ or $\mathbf{E}_C$ dominates, we find values for the constants $C_2^{[q]}$ and $C_3^{[q]}$. We expect $C_2^{[q]}$ to decrease with increasing q (it is essentially an interpolation constant) and $C_3^{[q]}$ to grow at a moderate rate (by a close inspection of the proof of Theorem 3.2). The results are listed in Table 3. Based on these results, we find that

$$C_2^{[q]} < 0.001,$$
$$C_3^{[q]} \approx 0.002 + 0.0005q.$$

We thus arrive at the following model for the propagation of errors:

$$\mathbf{E} \approx \left[0.5 \|U(0) - u(0)\| + 0.001\Delta t^{2q} + (0.002 + 0.0005q)\Delta t^{-1/2}\epsilon_{\mathrm{mach}}) \right] \cdot 10^{0.388T}. \quad (7)$$

5.2 Optimal time step

Based on the model (7), we determine an estimate of the optimal time step size by setting $\mathbf{E}_G = \mathbf{E}_C$. We find that

$$\Delta t = ((2 + 0.5q)\epsilon_{\mathrm{mach}})^{\frac{1}{2q+1/2}} \approx \epsilon_{\mathrm{mach}}^{\frac{1}{2q+1/2}} \quad (8)$$

for large values of q. Inserting the values $\epsilon_{\text{mach}} = 10^{-420}$ and $q = 100$ used in this work, we find $\Delta t \approx 0.008$ which is reasonably close to the value of $\Delta t = 0.0037$ which was used to compute the solution.

5.3   Computability as function of machine precision

To answer the question posed in the introduction — *How far is the solution computable for a given machine precision?* — we insert the approximate optimal time step Δt given by (8) into (7). Neglecting data errors, that is, assuming $U(0) = u(0)$, we find that

$$\mathbf{E} \approx 2 \cdot 0.001 \Delta t^{2q} \cdot 10^{0.388T} \approx 0.002 \epsilon_{\text{mach}}^{\frac{2q}{2q+1/2}} \cdot 10^{0.388T} \approx 0.002 \epsilon_{\text{mach}} \cdot 10^{0.4T}$$

for large values of q. Let $n_{\text{mach}} = -\log_{10} \epsilon_{\text{mach}}$ be the number of significant digits. It follows that $\mathbf{E} \approx 0.002 \cdot 10^{0.4T - n_{\text{mach}}}$. We conclude that the computability T_ϵ, that is, the time T_ϵ at which the solution is no longer accurate to within a precision ϵ is

$$T_\epsilon(\epsilon_{\text{mach}}) = \frac{n_{\text{mach}} + \log_{10}(\epsilon/0.002)}{0.4}.$$

Since T_ϵ does not depend strongly on ϵ (for $\epsilon_{\text{mach}} \ll \epsilon$), we find that the computability of the Lorenz system is given by

$$T(\epsilon_{\text{mach}}) = n_{\text{mach}}/0.4 = 2.5\, n_{\text{mach}}.$$

With six significant digits available to Lorenz in 1963, the computability was limited to $T \approx 2.5 \cdot 6 = 15$. With 16 significant digits, the computability is limited to $T \approx 2.5 \cdot 16 = 40$. Finally, with 420 significant digits, as was used in this work, the computability is limited to

$$T \sim 2.5 \cdot 420 = 1050 > 1000.$$

A more precise estimate is possible by considering the actual size of the stability factor at any given time T. Noting that $S_C(T) \approx 2 \cdot 10^{388}$ at $T = 1000$, we may obtain the estimate

$$\mathbf{E} \approx 0.001 \epsilon_{\text{mach}} S_C(T).$$

With $\epsilon_{\text{mach}} = 10^{-16}$, it follows from Figure 3 that $\mathbf{E} = 0.001$ at $T \approx 50$. Furthermore, for $\epsilon_{\text{mach}} = 10^{-6}$ we find that the computability is limited to $T \approx 25$.

6   Conclusions

We have investigated the computability of the Lorenz system and come to the conclusion that the size of the time interval on which the solution is computable scales linearly with the number of digits, $T \sim 2.5\, n_{\text{mach}}$. Thus, with 420 digits of precision, as used in this work, the computability is limited to $2.5 \cdot 420 \approx 1000$. Furthermore, if a precision of 840 digits is used, one may compute the solution on the time interval $[0, 2000]$ and if a precision of 4200 digits is used, one may compute the solution on the time interval $[0, 10000]$.

Acknowledgements

REFERENCES

[1] D.H. Bailey, R. Barrio, and J.M. Borwein. High-precision computation: Mathematical physics and dynamics. *Applied Mathematics and Computation*, 218(20):10106–10121, 2012.

[2] B. A Coomes, H. Kocak, and K. J Palmer. Rigorous computational shadowing of orbits of ordinary differential equations. *Numerische Mathematik*, 69(4):401–421, 1995.

[3] D. Estep and C. Johnson. The pointwise computability of the Lorenz system. *Math. Models. Meth. Appl. Sci.*, 8:1277–1305, 1998.

[4] T. Granlund et al. GMP (GNU multiple precision arithmetic library). *http://gmplib.org/*, 1996–2010.

[5] A. Jorba and M. Zou. A software package for the numerical integration of ODEs by means of high-order Taylor methods. *Experimental Mathematics*, 14(1):99–117, 2005.

[6] B. Kehlet and A. Logg. A posteriori error analysis of round-off errors in the numerical discretization of ordinary differential equations. *In prepration*, 2013.

[7] Benjamin Kehlet. *Analysis and implementation of high-precision finite element methods for ordinary differential equations with application to the Lorenz system*. MSc thesis, Department of Informatics, University of Oslo, 2010.

[8] S. Liao. On the reliability of computed chaotic solutions of non-linear differential equations. *Tellus A*, 61(4):550–564, 2009.

[9] A. Logg, K.-A. Mardal, and G. N. Wells, editors. *Automated Solution of Differential Equations by the Finite Element Method*, volume 84 of *Lecture Notes in Computational Science and Engineering*. Springer, 2012.

[10] Anders Logg. Multi-Adaptive Galerkin methods for ODEs II: Implementation and applications. *SIAM J. Sci. Comput.*, 25(4):1119–1141, 2003.

[11] E. N Lorenz. Reply to comment by L.-S. Yao and D. Hughes. *Tellus A*, 60(4):806–807, 2008.

[12] Edward N. Lorenz. Maximum simplification of the dynamic equations. *Tellus*, 12:243–254, 1960.

[13] Edward N. Lorenz. Deterministic nonperiodic flow. *J. Atmosph. Sci.*, 20:130–141, 1963.

[14] Lord Rayleigh. On convective currents in a horizontal layer of fluid when the higher temperature is on the under side. *Phil. Mag.*, 32:529–546, 1916.

[15] B. Saltzman. Finite amplitude free convection as an initial value problem. *J. Atmosph. Sci.*, 19:329–341, 1962.

[16] J. Teixeira, C.A. Reynolds, and K. Judd. Time step sensitivity of nonlinear atmospheric models: Numerical convergence, truncation error growth, and ensemble design. *Journal of the Atmospheric Sciences*, 64(1):175–189, 2007.

[17] D. Viswanath. The fractal property of the Lorenz attractor. *Physica D: Nonlinear Phenomena*, 190(1-2):115–128, 2004.

[18] L. S Yao and D. Hughes. Comment on "Computational periodicity as observed in a simple system" by Edward N. Lorenz (2006a). *Tellus A*, 60(4):803–805, 2008.

DAMAGE DETECTION THROUGH WAVELET TRANSFORM AND INVERSE ANALYSIS

ANNA KNITTER-PIATKOWSKA* AND TOMASZ GARBOWSKI*

*Poznan University of Technology
Institute of Structural Engineering
Piotrowo 5, 60-695 Poznan, Poland
e-mails: anna.knitter-piatkowska@put.poznan.pl, tomasz.garbowski@put.poznan.pl

Key words: Damage Detection, Wavelet Transform, Inverse Analysis

Abstract. Detection, localization and estimation of details of concentrated defects hidden in structural elements as an important part of structural health monitoring is considered here. In this work the effectiveness of discrete wavelet transform combined with inverse analysis is also discussed. The efficiency of the method is studied particularly when applied to eigenmodes of a cantilever steel beam expressed in amplitudes of vertical displacements, velocities, accelerations or strains. The structural response signal measured in discrete points is transformed using wavelet decomposition which clearly improve identifiability of damaged structure. Authors use a parametrized finite element model which mimic the real structure and by changing control parameters embedded in the numerical model minimize the discrepancy between the wavelet representation of both 'real' and numerically computed measurable quantities. For the discrepancy function minimization (within least-square framework) the deterministic, iterative Trust Region Algorithm is used. Also another technique which is applied to minimize the objective function within a frame of global minimization techniques i.e. Genetic Algorithm is tested and checked here.

1 INTRODUCTION

The problem of damage detection belongs to a wide class of identification problems, where unknown parameters of a structure are determined from experimental tests. It is connected with structural health monitoring and safety assessment. The damage can have different forms such as cracks, voids, inclusions or delamination, often found in composites. Localized damage is extremely dangerous because it can initiate progressive failure of a whole structure.

Among a large number of non-destructive testing X-rays, vibration, acoustic emission, heat transfer, magnetic field , eddy current or ultrasonic methods (see e.g. [1], [2], [3], [4],

[5], [6], [7]) also wavelet transform combined with inverse analysis can be used. Because even small and local damage leads to stiffness reduction, increase of damping and decrease of a natural frequency of the structure, damage detection methods based on analysis of structural dynamic response can be easily applied to identify the presence of damage [8]. It happens that the experiments limited to measurement of eigenfrequences are insufficient, since the global dynamic response is rather insensitive to damage localized on a small area, therefore the localization and severity of defect is not easy to identify. On the other hand methods based on local inspection or heat generation are capable to find damage position, form and/or magnitude but have small range of applicability.

For structural health monitoring different types of response, namely eigenfrequencies/eigenmodes, displacements, velocities and accelerations can be monitored. For this purpose modern scanning laser vibrometer for non-contact measurement are often applied. Vibrometer is capable of gathering vibration data in all three-dimensional coordinate system and have an extended range of ultra high vibration frequencies up to 600 MHz *www.ects.pl.*

The most fundamental challenge is the fact that damage is typically local phenomenon and may not significantly influence the global response of structures. Therefore the method which enables to extract the desired detailed information from a numerous data representing the global response of a damaged structure called Wavelet Transformation (WT) is proposed. Signal decomposition using WT allows to detect and localize the damage because wavelets demonstrate strong disturbance in a place where some defect is present. There are many wavelet functions e.g. Haar wavelet, Symlet, Coiflet, Meyer, Mexican hat or Morlet and new ones are constantly developed. It follows from the experience (see [9]) that in the class of considered problems the most effective appeared Daubechies wavelet of 4th order with two vanishing moments [10]. Estimation of the magnitude of the damage can be done by making use of e.g. Lipschitz exponent [11]. However, data processing of the structural response signal using CWT or DWT has appeared to be rather ineffective in identification of the type or shape of a defect. Therefore, some alternative method which provides a more precise damage identification is needed. In the literature a few attempts can be found, e.g. a combination of WT and artificial neural networks [12] or with inverse analysis [13].

The inverse analysis provides an important tool if one would like to characterize a bigger number of damage parameters in the locally deteriorated elements of the structures. Such technique uses, besides the experimentally obtained data (here the wavelet representations of the experimental measurements), also their numerical counterparts obtained from the computer test simulation. In the inverse analysis a variety of different minimization techniques can be employed for the discrepancy minimization. The discrepancy between experimental and numerical measurable quantities, called the objective function, is usually minimized in the frame of least square approach [14]. In the literature many authors solve an inverse problems by making use of iterative minimization gradient based algorithms (e.g. [15]), based on soft computing methods (e.g. [16]), etc. The in-

verse analysis using any minimization algorithm, searches for a set of embedded (usually unknown or uncertain) structural or constitutive parameters (not easily accessible from the experiment) by making use of indirect measurement. Such approach was successfully used by many researchers in various fields (e.g. [17]). However, the application of WT in its discrete form together with inverse analysis for structural diagnosis is still an open and unsolved problem.

2 BASIS OF WAVELET THEORY

Wavelets are functions that satisfying certain mathematical requirements can be used to represent data or other functions. Nevertheless this concept is not new. In the early 1800's Joseph Fourier, French mathematician, discovered that using superposition of sines and cosines he could represent other functions. Fourier transform is a perfect tool for analyzing the stationary signals representing them in frequency domain. Wavelets have advantages over it in situations when the signal contains discontinuities, spikes or sharp edges. In wavelet transform the data are cut into different frequency components and then each component is analyzed with resolution matched to its scale. It reduces the effects of the Heisenberg uncertainty principle [18], which in this case means the inability of precise signal analysis in time domain and frequency domain at the same time.

Fourier transform is a basic tool for harmonic analysis and signal processing. It decomposes a function/signal into sinusoids of different frequencies. The transformation is reversible and lossless and the function can be reconstructed from its transform. Fourier transform is defined over the space $\mathbf{L}^2(\mathbb{R})$ of square-integrable functions.

Fourier transform represents a signal through a linear combination of basis function and is defined as:

$$F(\omega) = \int_{-\infty}^{\infty} f(t) \cdot e^{-i\omega t} dt = \left\langle f(t),\, e^{i\omega t} \right\rangle, \qquad f \in \mathbf{L}^2(\mathbb{R}), \tag{1}$$

where i is the imaginary unit ($i^2 = -1$), ω - circular frequency $[rad/s]$ and t is time. The inner product of (1) can be written in the form:

$$\left\langle f,\, g \right\rangle = \int_{t \in R} f(t) \cdot \bar{g}(t)\, dt, \tag{2}$$

where $\bar{g}(t)$ is the complex conjugate of $g(t)$ function.

The Fourier coefficient $F(\omega)$ is obtained by multiplying function f and sinusoidal wave $e^{i\omega t}$. As the $e^{i\omega t}$ covers the entire real axis, the value of $F(\omega)$ depends on the values of $f(t)$ for all $t \in (\mathbb{R})$. It is therefore difficult to analyze any local properties of f on the basis of $F(\omega)$. Such analysis requires the decomposition of the signal using the set of functions well localized in time and frequency. Wavelet transformation is well suited for this purpose.

It is considered that $\psi(t) \in \mathbf{L}^2(\mathbb{R})$ is a wavelet (mother) function if it satisfies admissibility condition:

$$\int_0^\infty \frac{|\Psi(\omega)|^2}{\omega} d\omega < \infty, \tag{3}$$

where $\Psi(\omega)$ is Fourier transform of function $\psi(t)$.

Average value of wavelet function is equal to zero, it means that the wavelet integral over real axis disappears:

$$\int_{-\infty}^{\infty} \psi(t)\, dt = 0. \tag{4}$$

In wavelet transform there is only one wavelet (mother) function. For signal decomposition copies of wavelet, which are called wavelet family, are used. They are obtained by scaling and translating ψ according to formula:

$$\psi_{a,b} = \frac{1}{\sqrt{|a|}} \psi\left(\frac{t-b}{a}\right), \tag{5}$$

where the variable t denotes time or space coordinate, a is the scale parameter and b indicates the wavelet translation in time/space domain; $a, b \in (\mathbb{R})$; $a \neq 0$. The scale factor $|a|^{-1/2}$ is a normalization coefficient which ensures constant wavelet energy regardless of the scale. This means that $\|\psi_{a,b}\| = \|\psi\| = 1$ [19].

Continuous wavelet transform of given function $f(t)$ is obtained by integration the product of the signal function and the wavelet functions [20]:

$$W f(a,b) = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{\infty} f(t) \cdot \bar{\psi}\left(\frac{t-b}{a}\right) dt = \langle f(t), \psi_{a,b} \rangle, \qquad f \in \mathbf{L}^2(\mathbb{R}). \tag{6}$$

The inner product of (6) can be written in the form:

$$\langle f(t), \psi_{a,b} \rangle = \int_{t \in R} f(t) \cdot \bar{\psi}_{a,b}\, dt, \tag{7}$$

where $\bar{\psi}_{a,b}$ is the complex conjugate of $\psi(t)$ wavelet.

On the basis of formulas (1), (2), (6) and (7) it can be concluded that the wavelet transform is a transformation similar to the Fourier transform. Both of them are based on the use of the product of a signal $f(t)$ and the remaining part, called the kernel of the transform. The main difference is that the kernel in Fourier transform are sinusoidal functions (periodic, representing one frequency) and in wavelet transform the kernel is wavelet function which satisfies conditions (3) and (4). Next dissimilarity is that wavelet functions are localized in space. Fourier sinusoidal functions are not.

An important role in applications plays a dyadic wavelet transformation. Substituting $a = 1/2^j$ and $b = k/2^j$, k, $j \in \mathcal{C}$ in the (5) a wavelet family is obtained:

$$\psi_{j,k}(t) = 2^{j/2}\psi\left(2^j t - k\right), \tag{8}$$

where $j = 0, ..., J - 1$ is scale parameter, $k = 0, ..., 2^j - 1$ translation parameter and J is the maximum level of transformation.

Discrete wavelet transform is defined as:

$$W\,\psi_{j,k} = 2^{j/2}\int_{-\infty}^{\infty} f(t) \cdot \bar{\psi}\left(2^j t - k\right) dt = \langle\, f(t),\, \psi_{j,k}\rangle \tag{9}$$

and wavelet coefficients are given by:

$$d_{j,k} = \langle\, f(t),\, \psi_{j,k}\rangle. \tag{10}$$

A linear combination of wavelet functions $\psi_{j,k}$ and wavelet coefficients $d_{j,k}$ allows to represent a discrete signal (the number of data is equal to 2^J) in the form:

$$f(t) = \sum_{j=0}^{J-1} d_{j,k}\,\psi_{j,k}(t) \tag{11}$$

Multi-level signal representation is possible thanks to multi-resolution analysis (MRA) (see [21]), closely connected with wavelet transform. For multi-resolution signal analysis a scaling wavelet $\varphi(t)$ (father) is required:

$$\varphi_{j,k}(t) = 2^{j/2}\varphi\left(2^j t - k\right). \tag{12}$$

Discrete signal $f(t)$ can be approximate using wavelet $\psi(t)$ and scaling $\varphi(t)$ functions according to:

$$f(t) = \sum_{k=-\infty}^{\infty} a_{j,k} \cdot \varphi_{j,k}(t) + \sum_{k=-\infty}^{\infty}\sum_{j=0}^{\infty} d_{j,k} \cdot \psi_{j,k}(t), \tag{13}$$

where $a_{j,k}$ are scaling function coefficients derived from the formula:

$$a_{j,k} = \langle\, f(t),\, \varphi_{j,k}\rangle. \tag{14}$$

A wavelet function has a band-like spectrum, so the coefficients $d_{j,k}$ have high frequencies information (details). Coefficients $a_{j,k}$ have low-pass information with a constant component which is called signal approximation.

Multi-resolution analysis of discrete signal can be expressed in Mallat's algorithm:

$$f_J = S_J + D_J + ... + D_n + ... + D_1, \qquad n = J - j \tag{15}$$

where S_J is a smooth signal representation, D_n and are S_n are details and rough parts of a signal, j is the level of decomposition and J level of MRA. The idea of multiresolution analysis using Mallat pyramid is presented in Fig.1.
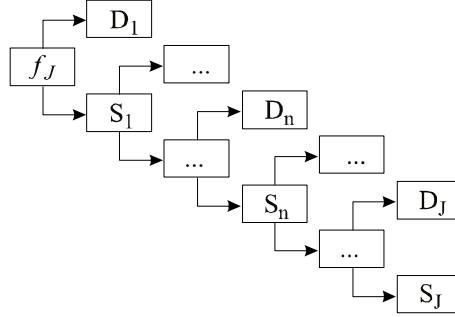
Figure 1: Multiresolution analysis of discrete signal

3 PROBLEM FORMULATION

Studies on the identification of defects reported in the literature are most often related to the beams, frames or plates (see e.g. [22]). They are in fact structural elements, which are most common in engineering practice. However, the specific type of the structure does not make any difference, provided that the response signal for any action (not necessarily defined) can be received. The main task of this study is to detect localization of damage in the structure, if such damage exists. Moreover the position, type, shape and severity of defect should be found. A cantilever beam made of steel, with Young's modulus $E = 200$ GPa and mass density $\rho = 7850$kg/m^3 is considered. The length of the beam is 0.96 m and rectangular cross section has dimensions 4×8 cm. Damage in beam is modeled as local stiffness reduction, obtained by reducing the height of cross-section or the value of Young's modulus. Authors use a parametrized Finite Element (FE) model which mimic the real structure subjected to dynamic mechanical excitation. All control parameters gathered in the vector $\mathbf{x}$ are embedded in the numerical model; by changing them one can minimize the discrepancy between the wavelet representation of both 'real' and numerically computed measurable quantities. Here the 'real' experiment is substituted with a numerical one, called here pseudo-experiment, in which all parameters (i.e. damage localization, type or shape of damage, number of monitored points, etc.) are known (Fig. 2a). By different initialization of the vector $\mathbf{x}$ in the numerical model (which is different from the pseudo-experimental one) (Fig. 2b), and by comparing the converged values of the sought parameters to those parameters used for pseudo-experimental data generation, one can check the robustness of the proposed method.

The effectiveness of the method was studied when applied to eigenmodes expressed in amplitudes of vertical displacements, velocities, accelerations or strains. The structural response of this kind is a discrete signal measured in points uniformly distributed along the span of a beam and transformed using WT. The response of undamaged structure in such case is unnecessary.
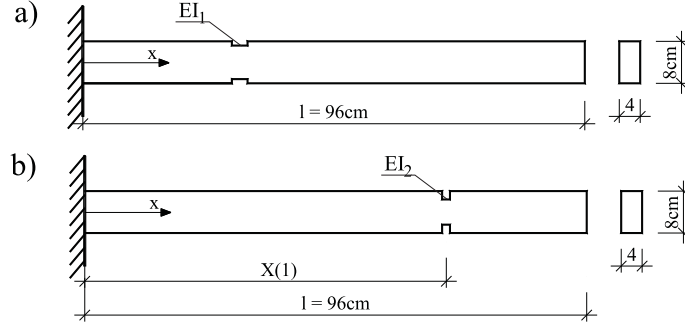
Figure 2: Steel cantilever beam: a) real damaged structure (experiment), b) computer model with variable defect parameters.

4 PROBLEM SOLUTION AND RESULTS

Among a large group of optimization algorithms in frame of nonlinear least square problems, the Gauss-Newton (GN) or Levenberg-Marquardt (LM) (see e.g. [23, 24] for more details) are the most efficient and often implemented into the inverse procedure. Here, however, another powerful algorithm is programmed and used for objective function minimization, namely trust region algorithm (TRA). TRA uses a simple idea, similar to the one in LM algorithm, where the new step is performed in the direction which combines a Gauss-Newton and steepest descent direction. LM algorithm computes a new direction using a following formula:

$$\Delta \mathbf{x} = - \left(\mathbf{H_x} + \lambda \mathbf{I} \right)^{-1} \mathbf{g_x}, \tag{16}$$

where: λ is an internal parameter, $\mathbf{g_x} = \nabla \omega \left(\mathbf{x} \right)$ is a gradient of objective function ω with respect to parameters $\mathbf{x}$:

$$\mathbf{g_x} = \frac{\partial \omega}{\partial \mathbf{x}} \tag{17}$$

and Hessian $\mathbf{H_x} = \nabla^2 \omega \left(\mathbf{x} \right)$ is a second partial derivative of ω with respect to parameters $\mathbf{x}$:

$$\mathbf{H_x} = \frac{\partial^2 \omega}{\partial \mathbf{x}^2}. \tag{18}$$

In the nonlinear least square approach, the gradient and Hessian can be computed based on Jacobian:

$$\mathbf{J} \left(\mathbf{x} \right) = \frac{\partial \mathbf{R}}{\partial \mathbf{x}}, \tag{19}$$

so the gradient and Hessian are defined, respectively:

$$\mathbf{g} \left(\mathbf{x} \right) = \mathbf{J}^T \mathbf{R}, \qquad \mathbf{H} \left(\mathbf{x} \right) \simeq \mathbf{J}^T \mathbf{J}. \tag{20}$$

Such approximation of the Hessian, which can be computed 'for free' once the Jacobian is available, represents a distinctive feature of least squares problems. This approximation is, however valid if residuals are small, meaning we are close to the solution, therefore

some techniques may be required in order to precondition Hessian to be semi-positive defined (see e.g. [25]).

One of the main issues of the trust region approach, that to a large extent determines the success and the performance of this algorithm, is the decision strategy of how large the trusted region should be. Allowing it to be too large can make the algorithm facing the same problem as the classical Newton direction line search, when the minimizer of model function is quite far from the minimizer of the actual objective function. On the other hand using too small region the algorithm misses an opportunity to take a substantial step that could move it much closer to the solution.

Each k-th step in the trust region algorithm is obtained by solving the sub-problem defined by

$$\min_{\mathbf{d}_k} m_k\left(\mathbf{d}_k\right) = f\left(\mathbf{x}_k\right) + \mathbf{d}_k^T \nabla f\left(\mathbf{x}_k\right) + \frac{1}{2}\mathbf{d}_k^T \nabla^2 f\left(\mathbf{x}_k\right)\mathbf{d}_k, \qquad \|\mathbf{d}_k\| \leq \Delta_k, \qquad (21)$$

where Δ_k is the trust region radius. By writing the unknown direction as a linear combination of Newton and steepest descend direction, the sub-problem will obtain the following form:

$$\begin{aligned}\min m_k\left(\mathbf{x}_k\right) = \quad & f\left(\mathbf{x}_k\right) + \left[\alpha_1\mathbf{d}_k^{SD} + \alpha_2\mathbf{d}_k^N\right]^T \nabla f\left(\mathbf{x}_k\right) + \\ + \quad & \frac{1}{2}\left[\alpha_1\mathbf{d}_k^{SD} + \alpha_2\mathbf{d}_k^N\right]^T \nabla^2 f\left(\mathbf{x}_k\right)\left[\alpha_1\mathbf{d}_k^{SD} + \alpha_2\mathbf{d}_k^N\right]\end{aligned} \qquad (22)$$

under the constrains:

$$\left\|\alpha_1\mathbf{d}_k^{SD} + \alpha_2\mathbf{d}_k^N\right\| \leq \Delta_k. \qquad (23)$$

The problem now becomes two dimensional and it is solved for the unknown coefficients α_1 and α_2. In order to find both alphas in (22) the set of nonlinear equations has to be solved using for example a Newton-Raphson techniques. Herein this approach is implemented into inverse procedure for the discrepancy minimization.

TRA, however, is a 'local' algorithm and if objective function is non-convex it may stuck in the local minimum. Therefore a regularization method or multi-start techniques can be beneficial. Here, very good results are obtained when the procedure is divided into two steps: first decomposition of the output structural response signal, e.g. strains, using DWT (see Fig. 3) for preliminary estimate of damage location and second the application of TRA on the limited search field (i.e. to the number of elements, where the wavelet disturbances is clearly evidenced). The advantage of this approach is relatively small number of iterations where damage details/sought parameters are properly specified (see Fig. 4).

Another possible technique which can be applied to minimize the non-convex functions are methods belonging to the global minimization search family. Here, Genetic Algorithm (GA) is programmed and employed for the damage detection problem. Some more details on GA and other evolutionary-based algorithms can be found in many textbooks and articles (e.g. [26], [27]). Unfortunately, the first approach to the damage detection problem with GA was unsuccessful (i.e. none of the sought parameters were found) when as
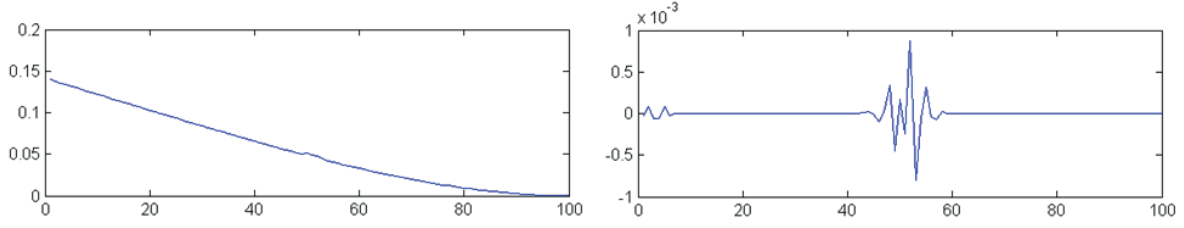
Figure 3: Structural response signal: a) expressed in strains , b) decomposed using DWT (detail 1).
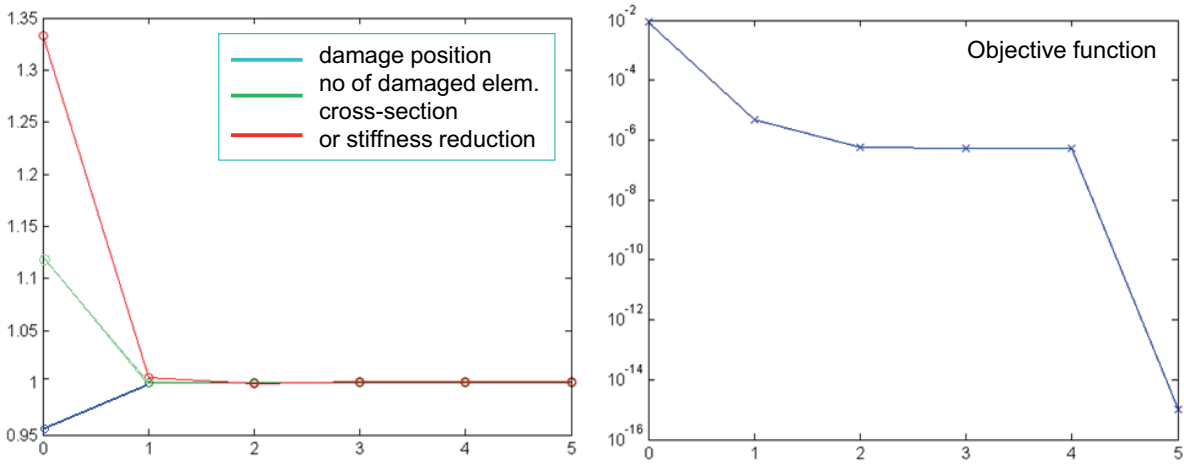
Figure 4: Trust region algorithm: a) identified damage details, b) objective function.

the output signal the direct structural dynamic response (e.g. expressed in strains) was used (see Fig. 5). Only the solution provided by application of GA on the output signal represented by wavelet coefficients appeared to be successful (see Fig. 6). All defect details, such as location, intensity, shape or number, were clearly identified within relatively small number of iterations.

5   CONCLUSIONS

The contribution of this work is a novel approach to Structural Health Monitoring (SHM) based on damage detection through wavelet transformation, numerical FE modeling and mathematical programming. The inverse analysis employed here uses two distinct minimization algorithms in order to select the most suitable technique of DWT application to SHM. The effectiveness of the method is studied by the way of an example of cantilever steel beam subjected to mechanical excitation. The eigenvibrations are considered. The examples proved that application either TRA or GA is very efficient in determining the details of damage such as location, severity, shape or number of defective elements. However, the prerequisite is that as the output the structural response signal (e.g. strains) represented in wavelet coefficients is taken into consideration. In the case of
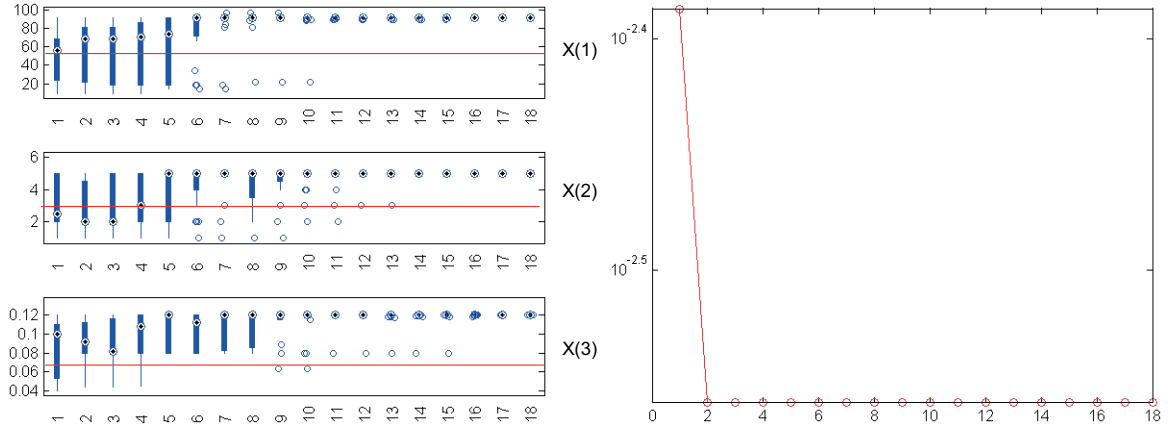
Figure 5: Genetic Algorithm: a) unidentified damage details, b) fitness function.
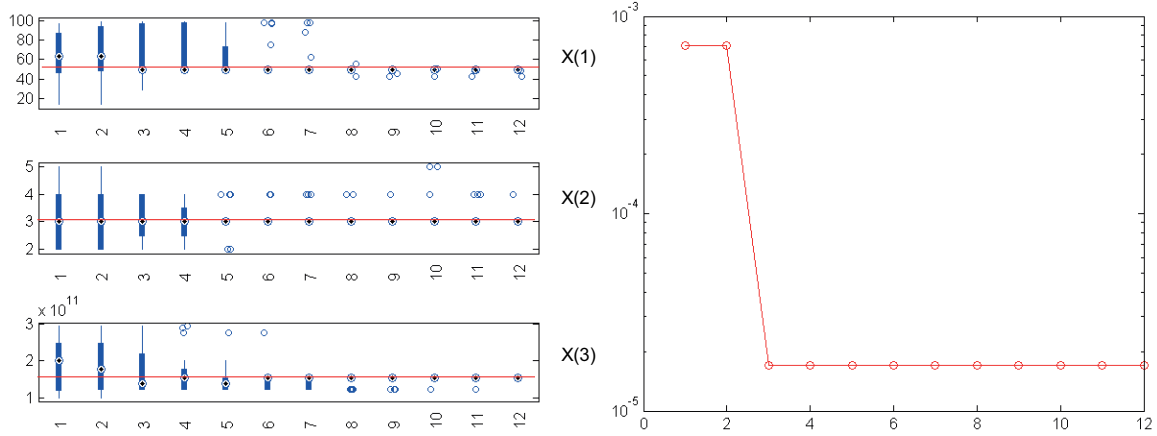
Figure 6: Genetic Algorithm: a) identified damage details, b) fitness function.

TRA application it allows to limit the search field to the number of elements, where the evident wavelet disturbances is visible, therefore the procedure is performed in a small number of iterations. In GA, when as the output signal the direct structural dynamic response is used, damage detection failed. Only when the output signal is expressed in wavelet coefficients, as mentioned, damage details are properly specified.

This preliminary work serves as a check of the usefulness of the proposed technique, and will be validated, in future, by a real experiment on structural elements.

REFERENCES

[1] Shinoda, K., et al. Application of digital radiography to aerospace-craft. *Key Engineering Materials* (2004) **270-273**:1361–1365.

[2] Salawu, O. S. Detection of structural damage through changes in frequency: a review.

Engineering Structures (1997) **19, 9**:718–723.

[3] Rogers, L.M. Crack detection using acoustic emission methods - fundamentals and applications. *Key Engineering Materials* (2005) **293-294**:33–48.

[4] Ziopaja, K., Pozorski, Z. and Garstecki, A. Damage detection using thermal experiments and wavelet transformation. *Inverse Problems in Science and Engineering* (2011) **19, 1**:127–153.

[5] Xiaochun, S., Xinjun, W. and Yihua, K. The method and apparatus of magnetic nondestructive testing for boiler waterwall tube. *Key Engineering Materials* (2004) **270-273**:642–646.

[6] Gros, X.E. An eddy current approach to the detection of damage caused by low-energy impacts on carbon fiber reinforced materials. *Materials and Design* (1995) **16(3)**:167–173.

[7] Zhang, G., Hu H. and Ta, D. Ultrasonic detection of the metallurgical defects in the steel and its evaluation by neural networks based on the wavelet transform noise suppression. *Key Engineering Materials* (2004) **270-273**:160–167.

[8] Krawczuk, M., Ostachowicz, W. Modeling and vibration analysis of a cantilever composite beam with a transverse open crack. *Journal of Sound and Vibration* (1995) **183(1)**:69–89.

[9] Knitter-Piatkowska, A., Pozorski, Z. and Garstecki, A. Application of discrete wavelet transformation in damage detection. Part I: Static and dynamic experiments. *Computers Assisted Mechanics and Engineering Sciences* (2006) **13**:21–38.

[10] Daubechies, I. *Ten lectures on wavelets*. Society for Industrial and Applied Mathematics, Philadelphia, (1992).

[11] Hong, J-C., et al. Damage detection using the Lipschitz exponent estimated by the wavelet transform: applications to vibration modes of a beam. *International Journal of Solids and Structures* (2002) **39**:1803–1816.

[12] Rucka, M., Wilde, K. Neuro-wavelet damage detection technique in beam, plate and shell structures with experimental validation. *Journal Of Theoretical and Applied Mechanics* (2010) **48, 3**:579–604.

[13] Amjady, N., Keynia, F. Short-term load forecasting of power systems by combination of wavelet transform and neuro-evolutionary algorithm. *Energy* (2009) **34, 1**:46–57.

[14] Kariya, T., Kurata, H. *Generalized Least Squares*. John Wiley and Sons, Chichester, UK, (2004).

[15] Garbowski, T., Maier, G. and Novati, G. Diagnosis of concrete dams by flat-jack tests and inverse analyses based on proper orthogonal decomposition. *Journal of Mechanics of Materials and Structures*, (2011) **6, 1-4**:181–202.

[16] Garbowski, T. , Maier, G. and Novati, G. On calibration of orthotropic elastic-plastic constitutive models for paper foils by biaxial tests and inverse analyses. *Structural and Multidisciplinary Optimization*, (2012) **46, 1**:111–128.

[17] Maier, G., Buljak, V., Garbowski, T., Cocchetti, G. and Novati G. Mechanical characterization of materials and diagnosis of structures by inverse analyses: some innovative procedures and applications. *International Journal of Computational Methods* (2013) submitted.

[18] Masanao, O. Universally valid reformulation of the Heisenberg uncertainty principle on noise and disturbance in measurement. *Physical Review A* (2003) **67(4)**:id. 042105.

[19] Strang, G. and Nguyen, T. *Wavelets and filter banks*. Wellesley-Cambridge Press, Wellesley, (1996).

[20] Chui, C.K. *An introduction to wavelets.* Academic Press, San Diego, (1992).

[21] Mallat, S.G. A theory for multiresolution signal decomposition: The wavelet representation. *IEE Trans. on Pattern Anal. and Machine Int.* (1989) **11(7)**:674–693. San Diego, (1998).

[22] Dems, K. and Mroz, Z. Identification of damage in beam and plate structures using parameter dependent frequency changes. *Engineering Computations* (2001) **18, 1/2**:96–120.

[23] Wang, Y. GaussNewton method. *WIREs Comp. Stat.* (2012) **4**:415–420.

[24] Ranganathan, A. The Levenberg-Marquardt algorithm. *Tutorial on LM Algorithm* (2004)

[25] Kelley, C.T. *Iterative methods for optimization.* Society for Industrial Mathematics, Vol. 18, (1987).

[26] Eiben, A.E., Smith,J.E. *Introduction to Evolutionary Computing.* Springer, (2008).

[27] Akbari, R., Ziarati, K. A multilevel evolutionary algorithm for optimizing numerical functions. *International Journal of Industrial Engineering Computations* (2010) **2**:419–430.

VI International Conference on Adaptive Modeling and Simulation
ADMOS 2013
J. P. Moitinho de Almeida, P. Díez, C. Tiago and N. Parés (Eds)

A VARIATIONAL MULTISCALE APPROACH FOR ERROR ESTIMATION IN ADAPTIVE ISOGEOMETRIC ANALYSIS

Mukesh Kumar[†] [*], Trond Kvamsdal[†], Kjetil Andre Johannessen[†], Arne Morten Kvarving[‡], Knut Morten Okstad[‡]

[*]Department of Mathematical Sciences, Norwegian University of Science and Technology,
N-7491 Trondheim, Norway
e-mail: Mukesh.Kumar@math.ntnu.no, Trond.Kvamsdal@math.ntnu.no,
Kjetil.Andre.Johannessen@math.ntnu.no

[†]Department of Applied Mathematics, SINTEF ICT, N-7465 Trondheim, Norway
e-mail: Arne.Morten.Kvarving@sintef.no, Knut.Morten.Okstad@sintef.no

Key words: Isogeometric analysis, NURBS, A posteriori error estimation, Fluid dynamics, Variational multiscale method, Adapted methods.

Abstract. In this work, we present an explicit a-posteriori error estimator in isogeometric analysis for fluid dynamics problem like advection-diffusion equations. The technique is based on the theortical framework of the variational multiscale (VMS) method [1] and recently derived explicit formula of the fine-scale Green's function [2]. This technique is adequate for the methods with a local error distribution, such as stabilized methods, where the element local problem captures most of the error and the proposed error intrinsic parameter is an approximation to the solution of the dual problem. The proposed technique can be implemented straightforwardly in existing codes and is computationally efficient. We consider different test examples to show the robustness and effectivitness of this technique as a posteriori error estimator in isogeometric analysis.

REFERENCES

[1] T. J. R. Hughes, Multiscale phenomena: Green's functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods, *Computer Methods in Applied Mechanics and Engineering*, 127(1995), 387-401.

[2] T. J. R. Hughes, G. Sangalli, Variational multiscale analysis: the fine-scale Green's function, projection, optimization, localization, and stabilized methods, *SIAM Journal on Numerical Analysis*, 45(2)(2007), 539-557.

A posteriori error estimation in adaptive isogeometric analysis

Trond Kvamsdal*[†], Mukesh Kumar[†]

Kjetil A. Johannessen[†], Arne Morten Kvarving[‡], and Knut M. Okstad[‡]

[†]Department of Mathematical Sciences,
Norwegian University of Science and Technology
Alfred Getz vei 1, NO 7491 Trondheim
Trond.Kvamsdal@math.ntnu.no, Mukesh.Kumar@math.ntnu.no, Kjetil.Andre.Johannessen@math.ntnu.no

[‡]Department of Applied Mathematics, SINTEF ICT
Alfred Getz vei 1, NO 7465 Trondheim
Arne.Morten.Kvarving@sintef.no, Knut.Morten.Okstad@sintef.no,

Key words: Isogeometric analysis, LR B-splines, posteriori error estimators, adaptive methods.

ABSTRACT

Reliability and efficiency are two major challenges in simulation based engineering. These two challenges may be addressed by error estimation combined with adaptive refinements. A lot of research has been performed on error estimation and adaptive mesh refinement. However, adaptive methods are not yet an industrial tool, partly because the need for a link to traditional CAD-system makes this difficult in industrial practice. Here, the use of an isogeometric analysis framework introduced by Professor Thomas J. R. Hughes (UT at Austin) and coworkers [1] may facilitate more widespread adoption of this technology in industry, as adaptive mesh refinement does not require any further communication with the CAD system.

A posteriori error estimation in numerical approximation of partial differential equations aims at:

- give an upper bound on the error of numerical solution, if possible give a guaranteed upper bound;
- estimate the error locally and assure that this represents a lower bound for the actual error, up to a multiplicative constant (i.e. efficiency);
- assure that the ratio of the estimated error and actual error goes to one, i.e., asymptotic exactness.

Three main techniques of a posteriori estimates in the finite element method have evolved during the last decades; *(i)* Explicit residual-based estimators *(ii)* Implicit residual based estimators and *(iii)* Recovery based estimators, see Ainsworth and Oden [2] . The purpose of this project is to extend these posteriori techniques in adaptive isogeometric analysis framework for elliptic problems. We also discussed the above three properties for our developed posteriori error estimators. The adaptive refinement is achieved using local refinement strategies developed in Johannessen *et al.* [3]. The developed a posteriori based adaptive refinement methodology will be tested on some classical benchmark elliptic problems.

References

[1] T. J. R. Hughes, J. A. Cottrell and Y. Bazilevs: Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, *Computer Methods in Applied Mechanics and Engineering*, 194(2005), 4135-4195.

[2] M. Ainsworth and J. T. Oden: A Posteriori Error Estimation in Finite Element Analysis, John Wiley & Sons, 2000.

[3] K. A. Johannessen, T. Kvamsdal and T. Dokken: Isogeometric analysis using LR B-splines, In review. *Computer Methods in Applied Mechanics and Engineering*.

VERIFICATION AND VALIDATION FOR THE LARGE EDDY SIMULATION OF INCOMPRESSIBLE TURBULENT FLOWS WITH FENICS

AURÉLIEN LARCHER, JOHAN HOFFMAN

Computational Technology Laboratory
High Performance Computing and Visualization
KTH Royal Institute of Technology, Sweden

Abstract. We describe a framework for verification and validation in the frame of the development of turbulence models for the Large Eddy Simulation of incompressible flows, by means of residual-based subgrid stabilisation.

The main components consist of a posteriori error estimation of the numerical error, uncertainty quantification of data and modeling errors, and systematic verification of the software implementation in FEniCS (www.fenicsproject.org) by manufactured solutions.

We introduce the different components, and present the combined framework in a number of examples .

ADAPTIVE REDUCED ORDER MULTISCALE FINITE ELEMENT METHODS BASED ON COMPONENT MODE SYNTHESIS

MATS G. LARSON*, HÅKAN JAKOBSSON*, AND TOR TROENG*

*Department of Mathematics and Mathematical Statistics
Umeå University
90187 Umeå, Sweden
e-mail: mats.larson@math.umu.se

Key words: Multiscale method, model reduction, a posterior error estimate

Abstract. We present a reduced order finite element method based on the variational multiscale method together with a component mode synthesis representation for the fine scale part of the solution. We present an a posteriori error estimate in the energy norm for the discrete error in the approximation which measures the error associated with model reduction in the fine scale.

1 INTRODUCTION

In this contribution we briefly describe a recent multiscale finite element method, introduced in [6], which builds on using a reduced order model for the fine scale in a variational multiscale method, see [2] and the later developments [5].

Model reduction methods are commonly used to decrease the computational cost associated with simulations involving repeated use of large scale finite element models of for instance a complicated structure. The objective of model reduction methods is to find a low dimensional subspace of the finite element function space that still captures the essential behavior of the solution sufficiently well. A classical model reduction method is component mode synthesis (CMS), see [3].

In CMS the computational domain is split into subdomains and a reduced basis associated with the subdomain is constructed by solving localized constrained eigenvalue problems associated with the subdomains together with modes that represent the displacements of the interface between the subdomains, as in the Craig-Bampton method [1].

Here we construct a multiscale finite element method where the coarse scale is represented by piecewise linear continuous elements on a coarse mesh and the fine scale is defined by a CMS related approach on a refined mesh, using the coarse mesh elements

as subdomains in the CMS method. The coupling modes are computed for each pair of neighboring elements and couple the response in the subdomains. Thus the fine scale is finally represented as a direct sum of functions with support in each element and functions associated with each edge supported in the two elements neighboring the edge. Adaptive reduction is accomplished by choosing a basis in each such subspace consisting of a truncated sequence of eigenmodes. The eigenmodes are numerically computed and capture fine scale effects.

We note that in the original CMS method the interface problem is global, which is a serious limitation since the reduced mass matrix is dense. In the multiscale method we present here we instead get a mass matrix with a block structure that is similar to finite element methods based on higher order polynomials. Furthermore, the size of all eigenvalue problems solved in the fine scale computations can be controlled by refining the coarse scale mesh.

We derive an a posteriori error estimate for the multiscale finite element method that can be used to automatically tune the number of subscale modes in an adaptive algorithm. For further details we refer to [6] and the previous work on a posteriori error estimates for component mode synthesis [4].

2   LINEAR ELASTICITY

Strong form: The equations of linear elasticity take the form: find displacements $\boldsymbol{u}$ such that

$$-\nabla \cdot \boldsymbol{\sigma}(\boldsymbol{u}) + \tau \boldsymbol{u} = \boldsymbol{f}, \qquad\qquad \boldsymbol{x} \in \Omega, \tag{1a}$$

$$\boldsymbol{\sigma}(\boldsymbol{u}) = 2\mu\boldsymbol{\varepsilon}(\boldsymbol{u}) + \lambda(\nabla \cdot \boldsymbol{u})\boldsymbol{I}, \quad \boldsymbol{x} \in \Omega, \tag{1b}$$

$$\boldsymbol{u} = \boldsymbol{0}, \qquad\qquad\qquad \boldsymbol{x} \in \Gamma_D, \tag{1c}$$

$$\boldsymbol{n} \cdot \boldsymbol{\sigma}(\boldsymbol{u}) = \boldsymbol{g}_N, \qquad\qquad \boldsymbol{x} \in \Gamma_N, \tag{1d}$$

where $\tau \geq 0$ is a real parameter, $\boldsymbol{f}$ is a body force, $\boldsymbol{g}_N$ is a traction force, $\boldsymbol{\varepsilon}(\boldsymbol{u}) = \frac{1}{2}(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^T)$ is the linear strain tensor, $\boldsymbol{\sigma}$ the stress tensor, $\boldsymbol{I}$ is the $d \times d$ identity matrix, and λ and μ are the Lamé parameters given by $\lambda = E\nu[(1 + \nu)(1 - 2\nu)]^{-1}$ and $\mu = E[2(1 + \nu)]^{-1}$, where E and ν is Young's modulus and Poisson's ratio respectively. The coefficients can have multiscale behavior, i.e. exhibit variation on a very fine scale or on multiple scales.

Weak form: The corresponding variational form of (1) reads: find $\boldsymbol{u} \in V = \{\boldsymbol{v} \in [H^1(\Omega)]^d : \boldsymbol{v}|_{\Gamma_D} = \boldsymbol{0}\}$ such that

$$A(\boldsymbol{u}, \boldsymbol{v}) = b(\boldsymbol{v}), \quad \forall \boldsymbol{v} \in V, \tag{2}$$

where $A(\cdot, \cdot)$ is the bilinear form

$$A(\boldsymbol{v}, \boldsymbol{w}) = a(\boldsymbol{v}, \boldsymbol{w}) + \tau(\boldsymbol{v}, \boldsymbol{w}) \tag{3}$$
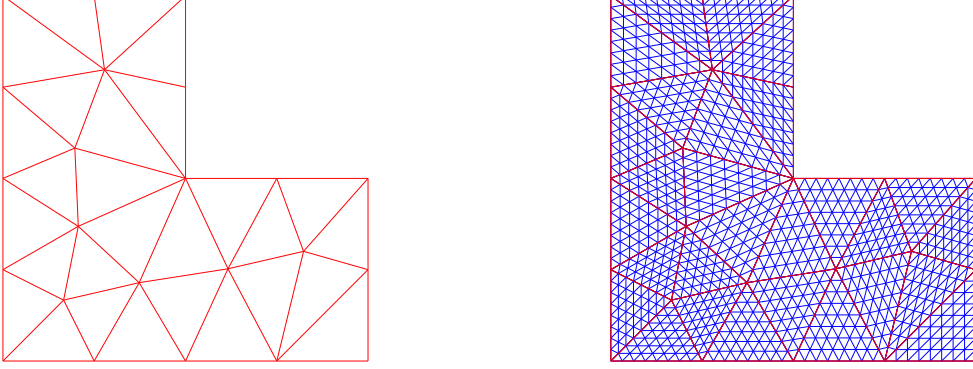
Figure 1: Left: coarse mesh $\mathcal{T}^H$ on an $L-$shaped domain. Left: fine mesh $\mathcal{T}^h$ obtained by a sequence of uniform refinements.

with

$$a(\boldsymbol{v}, \boldsymbol{v}) = 2(\mu\boldsymbol{\varepsilon}(\boldsymbol{v}) : \boldsymbol{\varepsilon}(\boldsymbol{w})) + (\kappa\nabla \cdot \boldsymbol{v}, \nabla \cdot \boldsymbol{w}), \tag{4}$$

and $b(\cdot)$ is the linear form

$$b(\boldsymbol{v}) = (\boldsymbol{f}, \boldsymbol{v}) + (\boldsymbol{g}_N, \boldsymbol{v})_{\Gamma_N}. \tag{5}$$

3 MULTISCALE METHOD

The mesh and finite element spaces: Let $\mathcal{T}^H$ be a coarse mesh on Ω consisting of shape regular triangles ($d = 2$) or tetrahedra ($d = 3$) and let $\mathcal{T}^h$ be a fine mesh obtained by a sequence of uniform refinements of $\mathcal{T}^H$. See Figure 3. Let $V^H \subset V^h$ be the corresponding spaces of continuous piecewise linear functions.

We then have the following splitting

$$V^h = V^H \oplus \left(\bigoplus_{E \in \mathcal{E}^H} V_E^h\right) \oplus \left(\bigoplus_{T \in \mathcal{T}^H} V_T^h\right) \tag{6}$$

Here $V_T^h \subset V^h$ is the space of functions with support in element $T \in \mathcal{T}^H$, $\mathcal{E}^H$ is the set of edges in the coarse mesh $\mathcal{T}^H$, and if the edge E is shared by elements T_1 and T_2 in $\mathcal{T}^H$ then the edge space V_E^h is defined by

$$V_E^h = \{v \in V^h \ : \ \text{supp}(v) \subset T_1 \cup T_2, a(v, w) = 0 \ \forall w \in V_{T_1}^h \oplus V_{T_2}^h\} \tag{7}$$

Note that this means that the functions in V_E^h are uniquely determined, through harmonic extension, by the restriction to the edge.

Multiscale finite element space: To construct a multiscale basis in this finite element space we use Fourier expansions in terms of eigenmodes determined by the following eigenvalue problems. Reduction, is then obtained by truncating the Fourier expansion.

- Basis in V_T^h: Let $(\boldsymbol{Z}_i, \Lambda_i) \in V_T^h \times \mathbb{R}^+$, for $i = 1, 2, \ldots, \dim(V_T^h)$, be the eigenpairs defined by

$$a(\boldsymbol{Z}, \boldsymbol{v}) = \Lambda(\boldsymbol{Z}, \boldsymbol{v}), \quad \forall \boldsymbol{v} \in V_T^h \tag{8}$$

Using modal truncation we obtain a reduced subspace $V_T^{h,m_T} \subset V_T^h$, defined by

$$V_T^{h,m_T} = \mathrm{span}\{\boldsymbol{Z}_i\}_{i=1}^{m_T}, \tag{9}$$

where $m_T \ll \dim(V_T^h)$.

- Basis in V_E^h: Let $(\boldsymbol{Z}_i, \Lambda_i) \in V_E^h \times \mathbb{R}^+$, for $i = 1, 2, \ldots, \dim(V_E^h)$, be the eigenpairs defined by

$$a(\boldsymbol{Z}, \boldsymbol{v}) = \Lambda(\boldsymbol{Z}, \boldsymbol{v}), \quad \forall \boldsymbol{v} \in V_E^h \tag{10}$$

Using modal truncation we obtain a reduced subspace $V_E^{h,m_E} \subset V_E^h$, defined by

$$V_E^{h,m_E} = \mathrm{span}\{\boldsymbol{Z}_i\}_{i=1}^{m_E}, \tag{11}$$

where $m_E \ll \dim(V_E^h)$.

Finally, we arrive at the following reduced order multiscale finite element space

$$V^{h,\boldsymbol{m}} = V^H \oplus \left(\bigoplus_{E \in \mathcal{E}^H} V_E^{h,m_E} \right) \oplus \left(\bigoplus_{T \in \mathcal{T}^H} V_T^{h,m_T} \right) \tag{12}$$

where $\boldsymbol{m} = (\cup_{E \in \mathcal{E}^H} m_E) \cup (\cup_{T \in \mathcal{T}^H} m_T)$ is the multi index containing the indices m_E and m_T for all edges $E \in \mathcal{E}^H$ and elements $T \in \mathcal{T}^H$.

Multiscale finite element method: The multiscale method is then directly obtained by using this reduced order space in the standard variational formulation: find $\boldsymbol{U^m} \in V^{h,\boldsymbol{m}}$ such that

$$A(\boldsymbol{U^m}, \boldsymbol{v}) = b(\boldsymbol{v}), \quad \forall \boldsymbol{v} \in V^{h,\boldsymbol{m}}, \tag{13}$$

Note that this is a coupled system involving both the coarse piecewise linear functions and the edge and element spaces spanned by the eigenmodes define above. Fine scale effects are captured in computations of the eigenfunctions on the fine mesh.

4   A POSTERIORI ERROR ESTIMATE

A posteriori error estimates: Let $\|\!|\!| \cdot \|\!|\!|$ denote the energy norm, $\|\!|\!|v\|\!|\!|^2 = A(v,v)$ and let $\boldsymbol{U}^h$ denote the standard finite element solution in V^h. Then we have the following a posteriori error estimate

$$\|\!|\!|\boldsymbol{U}^h - \boldsymbol{U^m}\|\!|\!|^2 \leq \sum_{E \in \mathcal{E}^H} \eta_E^2 + \sum_{E \in \mathcal{T}^H} \eta_T^2. \tag{14}$$

Here we introduced the following subspace indicators

$$\eta_E^2 = \frac{\|\boldsymbol{R}_E(\boldsymbol{U^m})\|^2}{\Lambda_{E,m_E+1}}, \quad E \in \mathcal{E}^H, \tag{15}$$

$$\eta_T^2 = \frac{\|\boldsymbol{R}_T(\boldsymbol{U^m})\|^2}{\Lambda_{T,m_T+1}}, \quad T \in \mathcal{T}^H, \tag{16}$$

where the subspace residual $\boldsymbol{R}_I : V_I^h \to V_I^h$, is defined by

$$(\boldsymbol{R}_I(\boldsymbol{w}), \boldsymbol{v}) = b(\boldsymbol{v}) - A(\boldsymbol{w}, \boldsymbol{v}), \quad \forall \boldsymbol{v} \in V_I^h, \quad I \in \mathcal{E}^H \cup \mathcal{T}^H \tag{17}$$

The indicators measure the error contribution due to reduction in the corresponding subspaces V_E^{h,m_E}, $E \in \mathcal{E}$, and V_T^{h,m_T}, $T \in \mathcal{T}$.

Adaptive algorithm: Based on the a posteriori error estimate (14) we may construct an adaptive solution procedure as follows:

1. Start with a guess of the subspace dimensions in V_E^{h,m_E} and V_T^{h,m_T}.

2. Solve the problem (13) and compute the subspace indicators (15) and (16).

3. If an indicator is large according to some refinement criterion, increase the number of modes in that subspace.

4. If $\sum_{E \in \mathcal{E}} \eta_E^2 + \sum_{T \in \mathcal{T}} \eta_T^2 <$ TOL, where TOL is a predetermined tolerance, stop. Otherwise, go to 2.

5   NUMERICAL EXAMPLE

We finally consider linear elasticity with $\tau = 0$, Young's modulus $E = 1$, and Poisson's ratio $\nu = 0.3$ on the $L-$shaped domain clamped, and free on the reminder of the boundary, at one side and exposed to a gravity force F_g acting on the whole domain, see Figure 5 (left). As is well known the solution is singular in the corner. We use an unstructured triangulation to construct the coarse mesh and a sequence of uniform refinements to construct the fine scale mesh, see 3. In figure 5 (right) we compare the adaptive
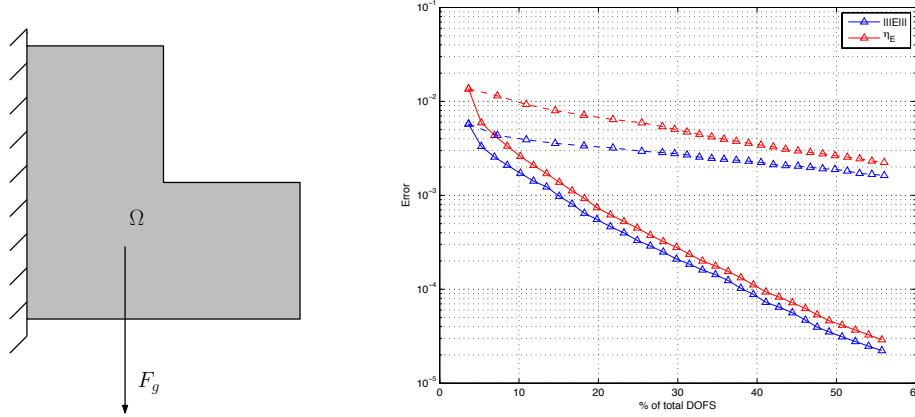
Figure 2: Left: set up of the problem in the numerical example. Right: decay of the error for the reduced displacement $\boldsymbol{U}_r$ using a uniform refinement strategy (dashed line) compared to an adaptive strategy (solid line).

strategy described above with a uniform strategy. We plot the estimated energy norm error compared to the actual energy norm error. We note that the error estimate is sharp and that the adaptive strategy outperforms the uniform strategy. We also note that the adaptive method actually produces an exponentially convergent sequence of approximate solutions.

REFERENCES

[1] Craig, R.R. and Bampton, M.C.C., Coupling of substructures for dynamic analysis. *AIAA Journal.* (1968) **6**:1313–1321

[2] Hughes, T.J.R., Feijóo, G.R., Mazzei, L., and Quincy, J.B. The variational multi-scale methoda paradigm for computational mechanics. *Computer methods in applied mechanics and engineering.* (1998) **166.1**:1313-1321.

[3] Hurty, W.C. Dynamic analysis of structural systems using component modes, *AIAA Journal.* (1965) **4**:678–685

[4] Jakobsson, H., Bengzon, F., and Larson, M.G. Adaptive component mode synthesis in Linear Elasticity. *Int. J. Numer. Meth. Engng.* (2011) **200.41–44**:829–844.

[5] Larson, M.G. and Målqvist, A. Adaptive variational multiscale methods based on a posteriori error estimation: energy norm estimates for elliptic problems. *Computer methods in applied mechanics and engineering.* (2007) **196.21**:2313–2324

[6] Troeng, T., Jakobsson, H., and Larson, M.G. A reduced order multiscale finite element method based on component mode synthesis. (2013) Technical report, Umeå University.

GOAL-ORIENTED ERROR CONTROL FOR THE QUASI CONTINUUM METHOD

FREDRIK LARSSON[*], ARASH MEMARNAHAVANDI[*] AND KENNETH RUNESSON[*]

[*]Chalmers University of Technology
Department of Applied Mechanics
SE41320 Göteborg, Sweden
e-mail: fredrik.larsson@chalmers.se, www.chalmers.se/am

Key words: Goal-oriented adaptivity, atomistic-to-continuum, Quasi-Continuum

Abstract. In order to consider virtual materials (not physically manufactured) or in order to predict complex behavior on one length scale where the physics is better understood on a lower scale, homogenization (or coarse-graining) can be a useful tool. Considering atomistic systems, homogenization can be used to derive continuum properties whereby the need for empirical continuum models i avoided. When there is a large separation of scales, i.e. when the length scales of the sought continuum solution by far exceeds the atomistic length scale, computational homogenization can be adopted, see e.g. [1]. Using this approach, the continuum stress-strain response can be obtained implicitly by considering a representative volume element (RVE), also called representative lattice unit in the case of a homogenization of a discrete lattice. However, when considering atomistic systems, it is well known that defects play an important role. When investigating the influence from single (or a few) defects, very large RVE's need be considered.

In the case of scale-mixing, i.e. when the continuum scale and the atomistic scale needs be resolved concurrently in the spatial domain, one popular method is the Quasi-Continuum (QC) method, cf. [2, 3]. It allows for coarse graining of atomistic response in terms of interpolation on a "finite-element-type" mesh. The QC method is an approximation of the atomistic problem, rather than a homogenization technique.

In this contribution we establish the RVE for carrying out atomistic-to-continuum homogenization of a molecular statics problem. In particular, we are interested in computing the representative response for different imperfections in a lattice. To this end, we wish to consider relatively large lattices on the atomistic scale. In order to facilitate such an analysis, we proceed along the lines of, e.g., [4] and device a goal-oriented adaptive QC procedure for solving the atomistic problem on the RVE. Within the goal-oriented framework it becomes natural to consider the macro scale (continuum) stress as the goal-quantity that is solved for.

The QC method is introduced in two steps. First, we consider the restriction of atom displacement in terms of the representative atoms as a model reduction, i.e., we describe the positions of all atoms in terms of discrete weights and the placements of certain so-called representative atoms. Based on this approximation, while accounting for the exact summation of all the bond-energies, we are able to compute goal-oriented error estimators in a straight-forward fashion based on an adjoint (dual) problem pertaining to the chosen output of interest. This computable error estimator pertains to a discretization error in the finite element method. The second step in the QC method is that of quadrature. For large QC elements, i.e. for a large amount of atoms whose placements are governed by the same representative atoms, the bond energy and its derivatives are typically computed using an appropriate discrete quadrature. We show how this approximation generates a quadrature error (in addition to the discretization error) in the framework for error estimation presented above. The combined error is estimated approximately based on the same dual problem in conjunction with a hierarchical strategy for approximating the residual.

As a model problem, we consider a mono-layer of graphene. The homogenization of the macro-scale membrane forces, including initial relaxation, is considered for defective graphene lattices. The 0 Kelvin condition is considered by omitting lattice vibration and the Carbon-Carbon energy bonds are modeled via the Tersoff-Brenner potential, cf. [5], which involves next-nearest neighbor couplings. In particular, we study the accuracy and robustness of the proposed error estimator and the pertinent adaptive algorithm.

REFERENCES

[1] Larsson, R. and Samadikhah, K. Atomistic continuum modeling of graphene membranes. *Computational Material Science* (2011) **50**:1744–1753.

[2] Miller, R. and Tadmor, E. The quasicontinuum method: Overview,applications ,and current directions. *Journal of Computer-Aided Design* (2002) **9**:203–239.

[3] Knap, J. and Ortiz, M. An analysis of the the quasicontinuum method. *Journal of the Mechanics and Physics of Solids* (2001) **49**:1899–1923.

[4] Prudhomme, S., Baumann, P. and Oden, J.T. Error control for molecular statics problems. *International Journal for Multiscale Computational Engineering* (2006) **4**:647–662.

[5] Brenner, D. Empirical potential for hydrocarbons for use in simulating chemical vapor deposition of diamond films. *Physical Review B* (1990) **42**:9458–9471.

AN IMPRINTING ALGORITHM TO INSERT GEOMETRIC DETAILS INTO HEXAHEDRAL MESHES

Nicolas Le Goff, Franck Ledoux, Jean-Christophe Weill

*CEA, DAM, DIF, F-91297 Arpajon, France
e-mail: nicolas.le-goff@cea.fr, franck.ledoux@cea.fr, jean-christophe.weill@cea.fr

Key words: Parametric studies in Numerical Simulation, Hexahedral Meshes, Adaptive model, Sheet operations, Mesh Imprint

Abstract. In numerous computational engineering applications, hexahedral meshes may be preferred over tetrahedral meshes. However, automatic hexahedral meshing remains an unsolved issue and thus generating a hexahedral mesh is known as a time-consuming stage that requires a lot of user interactions in the simulation process. A possible way for designing and optimizing a CAD model or a geometric shape requires parametric studies where the shape is enriched by inserting geometric details into it. Then we must "adapt" the initial mesh and not generate it anew for each new detail taken into account. In order to perform such studies with hexahedral meshes, we provide an imprinting method allowing us to automatically add geometric details into an existing mesh. This addition is done using geometric projections, sheets (layers of hexahedral elements) insertions and combinatorial algorithms while preserving the hexahedral mesh structure as best as possible.

1 INTRODUCTION

The definition of a real mechanical piece using only numerical modeling and simulation has been increasingly used for several years. A lot of research efforts have been put into the quality control of the numerical solutions and into the design of sophisticated, complex and coupled modeling, which leads to increasingly time-consuming computations. Most of these simulations rely on the finite element method (FEM) or the finite volume method (FVM). Both of them require that the geometric model be discretized by a mesh. In most cases, they are purely tetrahedral or hexahedral, that is to say exclusively composed of tetrahedral elements or hexahedral elements. In this work, we focus on the generation of hexahedral meshes, and more precisely on the adaptation of an existing hexahedral mesh to fit new geometric features that are inserted into a CAD model during an adaptive simulation process.

The classical process for designing and optimizing a geometric shape requires parametric studies where the shape is modified and/or enriched by adding geometric details. Considering a first shape with an associated mesh, we want to "adapt" the initial mesh and not to regenerate it from scratch for each new part taken into account (see Fig. 1). In order to perform such studies with hexahedral meshes, we provide an imprinting method that allows us to automatically add geometric details into a hexahedral mesh. This addition is done using geometric projections, sheets (layers of hexahedral elements) insertions and combinatorial algorithms, while preserving the hexahedral mesh structure as best as possible. Some authors have studied the insertion of complex geometric models into an existing grid or octree structure in order to get the initial mesh [8, 12, 4, 9, 5, 6, 13]. In our work, we focus on CAD models where sharp features are numerous and must be preserved; corners and ridges are typically difficult to capture in an existing mesh. The main contributions of our work are:

- Contrary to existing algorithms [8, 12, 4, 9, 5, 6, 13], our method can be applied onto any unstructured hexahedral meshes, it is not restricted to grids or octrees;

- While these algorithms only use a grid or octree to discretize the inner volume of one or several geometrical domains, we discretize both the inner and outer volumes;

- Both the initial geometric domain and the geometric details to be inserted have several corners and ridges.

The remainder of this paper is organized as follows: Section 2 gives an overview of our algorithm while introducing necessary terminology. Section 3 discusses the detailed algorithm for properly capturing the new geometric entities into the mesh. Section 4 explains how to improve the mesh quality in the vicinity of the inserted details and to improve the robustness of our algorithm. Section 5 draws conclusions and outlines future works.

2 MAIN STEPS OF THE IMPRINTING ALGORITHM

2.1 Background notions

A traditional representation [3] of a hexahedral mesh is to consider a 4-tuple (H, F, E, N) where H is a non-empty set of hexahedra, F is the non-empty set of all quadrilaterals adjacent to one or more hexahedra in H, E is the non-empty set of all edges adjacent to one or more hexahedra in H and N is the non-empty set of all nodes adjacent to one or more hexahedra in H. Hexahedra are 3-dimensional cells, or 3-cells, quadrilaterals are 2-cells, edges are 1-cells and nodes are 0-cells. In this work, the geometric domain Ω that we want to discretize is a 3-dimensional geometric object represented by its boundary. It is thus a BRep object described as a 3-tuple (S,C,V) [2] where S is a non-empty set of geometric surfaces enclosing a 3-dimensional space and such that $\forall (s_1, s_2) \in S^2$, $s_1 \cap s_2 = \emptyset$, C is the non-empty set of curves adjacent to one or more surfaces in S and V is the non-empty set of vertices adjacent to one or more surfaces in S.
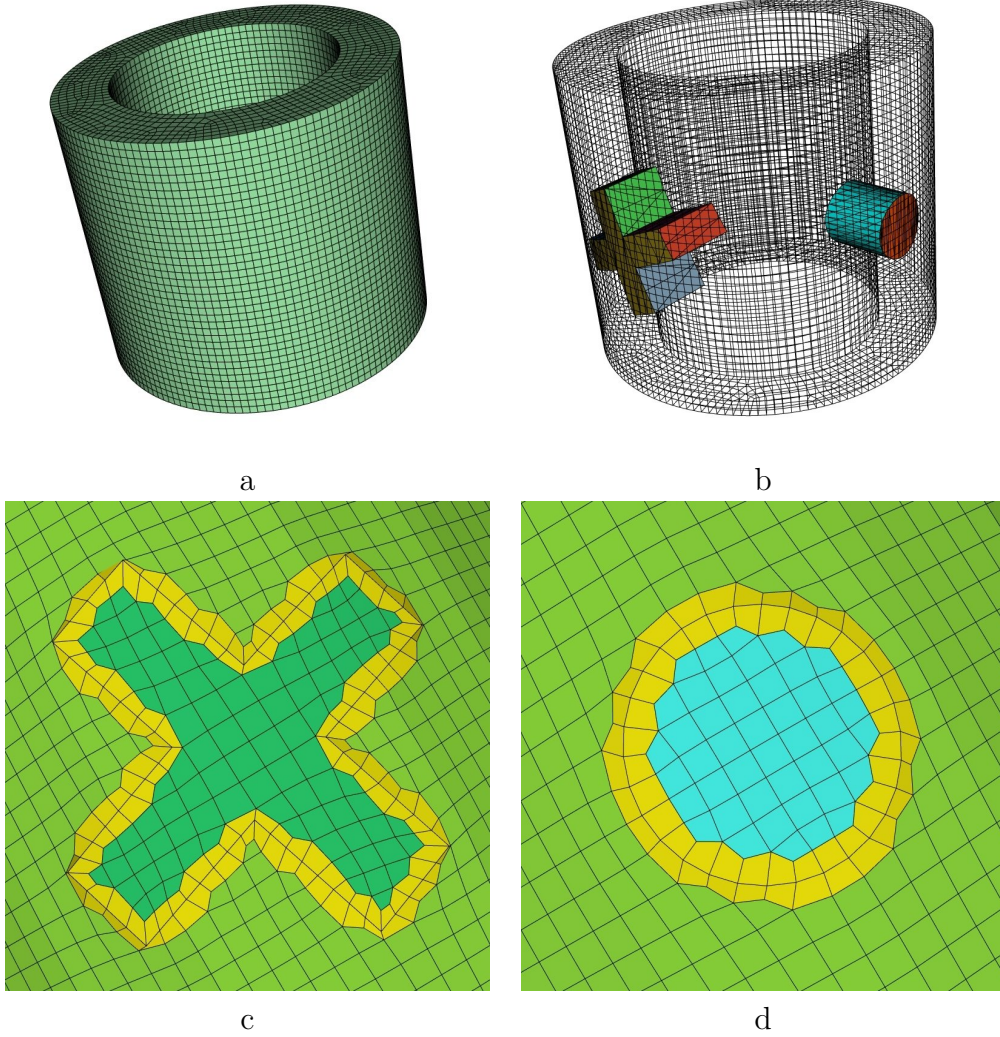
Figure 1: A hexahedral mesh is modified in order to add some geometrical details that can be relevant for the numerical study or to get a more geometric-sharp model. In (a), the first mesh was obtained using a sweeping algorithm [1]. In (b), two shapes, a cylinder and a cross shapes are added. In (c) and (d), close-up of the imprint on the side of the original mesh resulting from the insertion of respectively the cross and the cylindre shapes.

Let $M = (H, F, E, N)$ be a hexahedral mesh discretizing[1] the BRep object $G = (S, C, V)$. In order to initialize boundary conditions for FEM and FVM methods, it is mandatory to associate[2] each i-cell to a j-dimensional geometric entity with $j \geq i$. To get a valid association, some constraints must be satisfied:

- A mesh surface $s_M \subseteq F$, i.e. a set of pairwise adjacent faces of F forming a 2-

---

[1] The notion of discretization is not detailed in this paper. See for instance [3].

[2] This association is similar to the classification notion introduced by REMACLE and SHEPHARD in [7].

manifold, must be associated to each geometric surface $s \in S$. It means that all the faces in s_M, all the edges and nodes adjacent to a face of s_M are geometrically on surface s within a tolerance, and s_M discretizes surface s (i.e. every point $x \in s$ is contained in exactly one face, $f \in s_M$, and s_M wholly fills s.

- A mesh line $l_M \subseteq E$, i.e. a set of pairwise adjacent edges of E forming a 1-manifold, must be associated to each geometric curve $c \in C$. It means that all the edges in l_M and all the nodes adjacent to an edge of l_M are geometrically on curve c within a tolerance, and l_M discretizes curve c (i.e. every point $x \in c$ is contained in exactly one edge, $e \in l_M$, and l_M wholly fills c.

Implicitly, it means that if two geometric surfaces s_1 and s_2 of a BRep object share a curve c then the edges of the mesh line associated to c are also associated to s_1 and s_2 and such a line of edges separates the two sets of faces associated to s_1 and s_2.

2.2 Overview of the algorithm

Starting from a hexahedral mesh $M = (H, F, E, N)$ that discretizes a BRep geometric object $G = (S, C, V)$, the aim of our algorithm is to adapt M in order to discretize both G and G_2, where $G_2 = (S_2, C_2, V_2)$ is a new geometric object fully enclosed into G. The global process of our method is the following one:

1. Cells of H are split into two sets: those inside G_2, denoted H_2, and those outside; cells that are intersected by the geometric object will either be classified as inside or outside depending on a few criteria(see Section 3.1). Some refinement patterns can be applied to ensure the right topology of H_2 (see Section 4.1);

2. Each vertex of V_2 is captured by a node located on the boundary of H_2 (see Section 3.2);

3. Each curve of C_2 is captured by a line composed of edges of E located on the boundary of H_2 of which the endnodes capture the endpoints of c (see Section 3.3);

4. Each surface s of S_2 is captured by a mesh surface composed of faces of F located on the boundary of H_2 and delimited by mesh lines capturing the bounding curves of s (see Section 3.4);

5. Layers of hexahedra are inserted along the boundary of H_2 in order to improve the quality of elements (see Section 4.2).

3 CAPTURING GEOMETRIC ENTITIES INTO A HEXAHEDRAL MESH

3.1 Extraction of inner cells

The first step consists in choosing which hexahedra of the original mesh M will be considered as being part of the inserted geometric entity G_2. Thanks to the fact that G_2

is fully enclosed into G, hexahedra are divided into two categories: inside or outside the geometric detail, and the set of faces of M delimiting the two areas will be considered as the discrete boundary of G_2 in M.

- We first identify the hexahedra intersected by G_2 and mark the remaining cells as either inside or outside. Let H_2 be the set composed of intersected and inner hexahedra;

- The intersected hexahedra of H_2 will then be classified as inside or outside depending on whether more or less than half of their volume is located inside the geometric detail; those classified as outside are removed from H_2. This is done by, for each hexahedron h of H_2, taking a set of points S_h located inside the cell and determining if most of them reside within G_2 or not. For each point P in S_h the projected point P_{S_2} on the surfaces S_2 is computed, then the sign of the scalar product between $\mathbf{P_{S_2}P}$ and the outward normal to the surface at P_{S_2} determines whether the point is inside or outside (negative is inside, positive is outside).
  Currently we take an arbitrary number of 27 points located inside each intersected hexahedron using trilinear interpolation; Gauss points or some other quadrature rule could be used.

At the end of this step, the hexahedra of M are separated into two sets: those inside G_2 and those outside. In the following steps, our algorithm is restricted to selecting boundary nodes, edges and faces among the discrete boundary of G_2 in M.

3.2 Vertices' classification

A boundary node $n \in N$ will be associated to each vertex $v \in V_2$ considering a distance criterion, meaning the nearest node of N will be chosen for each vertex v of V_2. A node cannot be associated to more than one vertex, and in case of conflict, for example if two vertices both have the same nearest node, vertices' classification is done on a first-come, first-served principle.

3.3 Curves' classification

Curves' classification is done in two steps.

First for each vertex $v \in V_2$ we associate an edge to every curve adjacent to said vertex (see Fig. 2-a-b). Let C_v be the list of curves adjacent to v, ordered around v in a direct order. Let n be the node associated to v and E_n be the set of boundary edges of M adjacent to n. We are looking for the list of ordered edges $L_n \subseteq E_n$, ordered around n in a direct order, that best matches C_v. We define such a list as the list of ordered edges that maximizes the cost function:

$$f(L_n) = \sum_{i=1}^{|C_v|} \mathbf{C_v}[i].\mathbf{L_n}[i]$$

where $\mathbf{C_v}[i]$ and $\mathbf{L_n}[i]$ are the vectors of respectively the ith curve/edge of C_v/L_n pointing outward from v/n. This phase is not mandatory but it allows us to select a better solution near the vertices, which is typically where a good selection will improve robustness by avoiding crossing between lines of selected edges; this non-crossing property is mandatory for the algorithm used during the surfaces' classification phase.

The second step builds the remainder of the lines for each curve $c \in C_2$ (see Fig. 2-c-d), by starting from one of the curve's endpoints and building a contiguous line of boundary edges to reach the other endpoint. A set of selectable edges is computed as the boundary edges part of every hexahedron intersected by c, and a shortest path algorithm is used where each edge is weighted by its Hausdorff distance to the curve [10]. This way, we extract a suitable line of edges. Let us note that we do not start and end at the endpoints of c, but rather we start from the first edges associated to c at its endpoints during the previous step.

In case of curves that do not have endpoints (circles for example in the cylindrical shape inserted, see Fig. 2-f) we arbitrarily put a few points on the curve and associate them to boundary nodes , then build lines of edges that connect all those nodes using the same method as described above, i.e. a weighted shortest path algorithm applied on a restricted set of edges.

3.4 Surfaces' classification

Surfaces' classification is fairly straightforward once the edges' lines have been determined. Sets of faces are delimited by the lines, and for each set of faces s_M delimited by a set of lines L_M the corresponding surface $s \in S_2$ is the surface delimited by the curves the lines in L_M are associated to (see Fig. 2-e). This is sufficient to characterize all the surfaces of S_2 but in two cases:

- When there are no curves, for example if the geometric detail is a sphere, there is only one surface s in G_2 which is then associated to s_M;

- When there are only two surfaces, hence delimited by the same set of curves we have to choose an order of traversal for the curves and lines of edges and discriminate between the two surfaces by determining which surface is on the left or on the right.
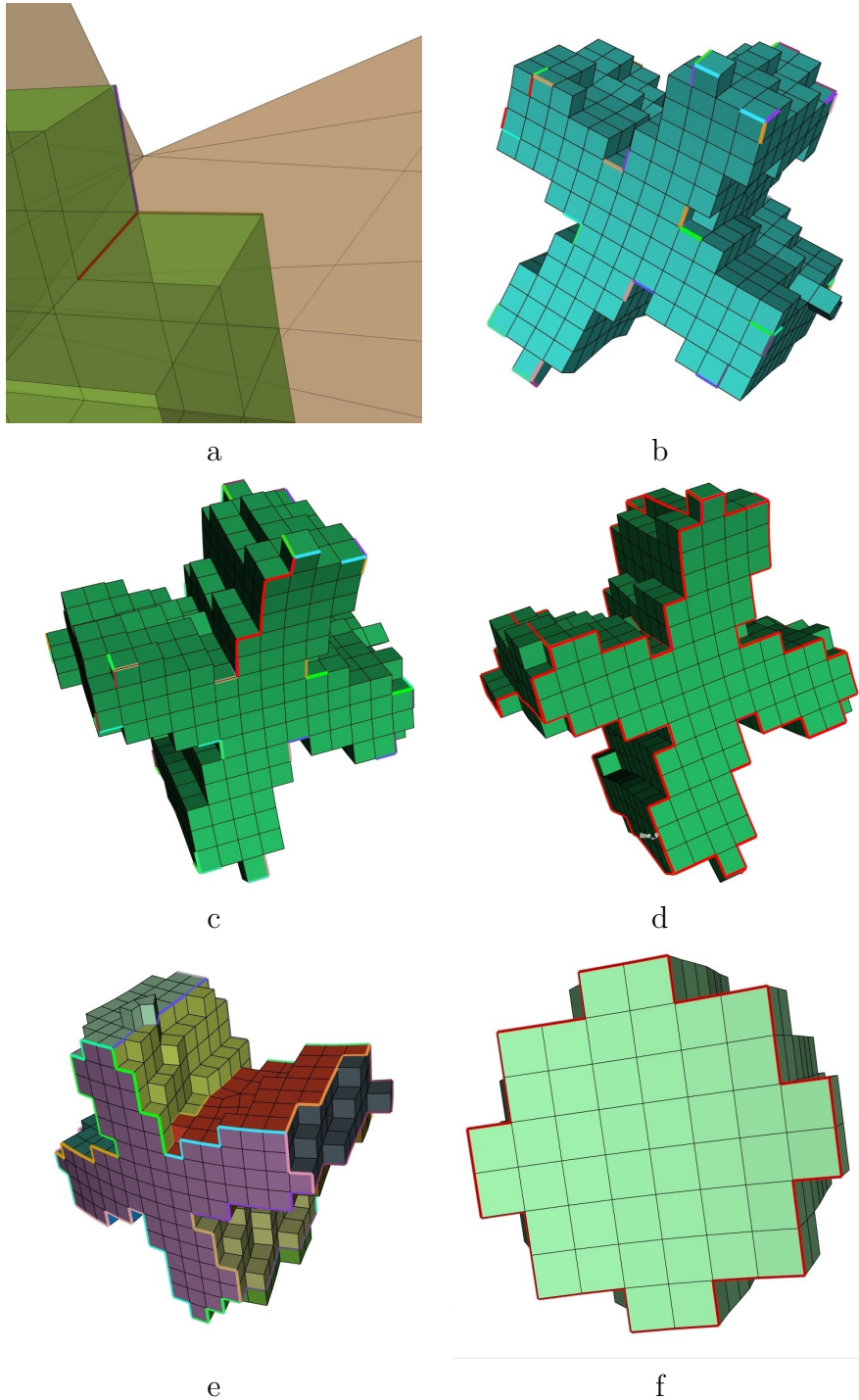
Figure 2: Curves' classification, from a best combination around each vertex to building a line of edges. In (a), close-up of the combination of edges that best matches the 3 curves of the cross shape at this vertex. In (b), edges at every vertex have been associated to curves. In (c), a line of edges (in red) has been associated to a curve. In (d), every curve in the cross shape has been associated to a line of edges. In (e), surfaces are classified to sets of faces. In (f), a curve of the cynlindrical shape is associated to a line of edges despite having no endpoints.

4 ROBUSTNESS AND QUALITY IMPROVEMENT

At this point in the paper the main contributions of our work have been outlined; the geometric detail G_2 has been inserted into the initial mesh and its surfaces, curves and vertices have been associated to mesh entities, but in order to be more robust and obtain a resulting mesh of better quality our algorithm needs to apply the following steps:

4.1 Refinement

The quality and robustness of the geometric detail classification strongly depends on the initial mesh. We use a 3-refinement strategy similar to the refinement used in [14] in order to get a valid result at the end of the first step of our algorithm. Indeed, such a refinement ensure that the topology of the set of inner hexahedra of H associated to G_2 will be the same as the topology of $G2$. In Fig. 3 the mesh is refined in the thin areas using a criterion based on whether at least two non-neighbor surfaces intersect a hexahedron. That allows the algorithm to better capture the thin top and bottom parts, and to disjoin the two parts on the right side of the model. Otherwise depending on the position and the size of the hexahedra near the thin space on the right of the model the space would not be captured, meaning the selected hexahedra in this area would form one block instead of two, and the corresponding surfaces and curves would not be classified.

a	b
c	d
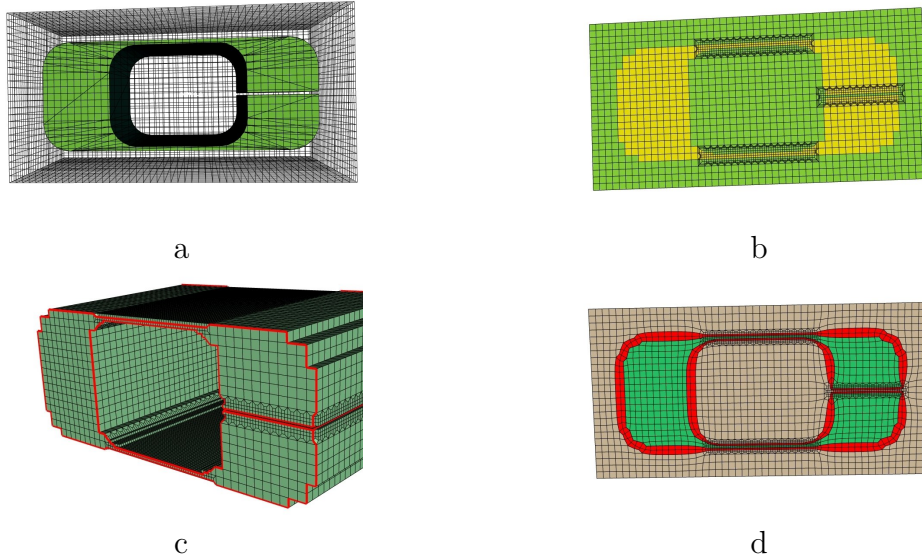
Figure 3: A hexahedral mesh is refined in order to facilitate the geometric detail insertion. In (a), the geometric detail is represented inside the original mesh, which is a regular cartesian grid. In (b), three areas are refined, around the thin parts of the geometric detail; the inside hexahedra are represented in yellow, the ouside ones in green. In (c), after curves' classification. In (d), after sheets insertions, sheets represented in red.

4.2   Sheet insertion

After having classified the vertices, curves and surfaces of the geometric detail G_2, sheets can be inserted in order to offer good quality elements near the surface of the geometric detail, and to provide boundary-aligned elements in case the numerical simulation favours such a feature in a mesh (see Fig. 4). Depending on the requirements on the resulting mesh, fundamental sheets and chords [2] can be inserted.

a   b
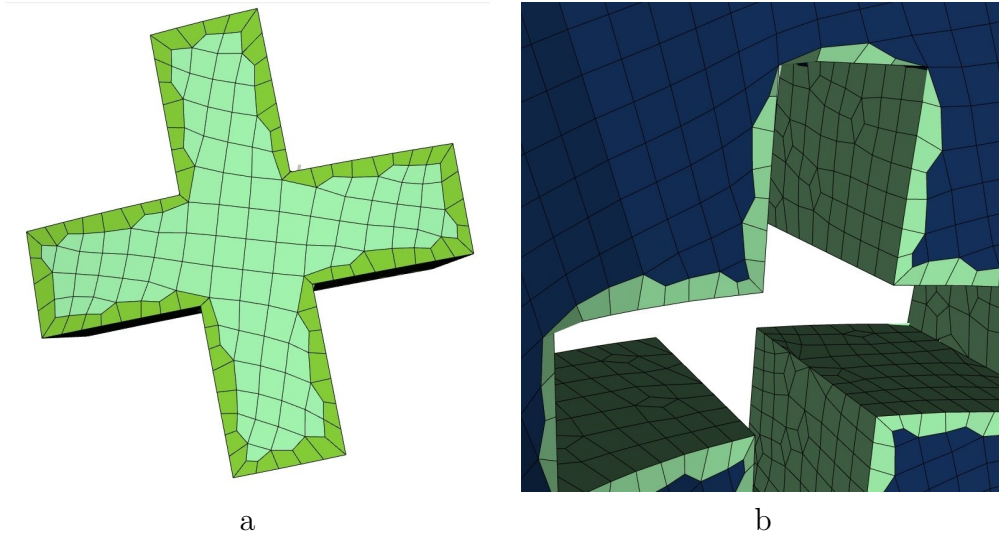
Figure 4: Sheet insertions after classification of the cross shape inside the original mesh. In (a), one sheet was inserted around the cross shape, inside the geometric detail. In (b), a sheet was inserted around the shape but this time located on the side of the outer volume.

4.3   Mesh smoothing

A laplacian smoothing constrained by the geometric classification was applied to the examples shown in this paper. But to get better quality, it seems mandatory to apply more evolved algorithms. Indeed, as we insert sharp geometric objects, non convex areas with sharp ridges appear. In such areas,algorithms merging untangling technics and geometric smoothing should be used [11]. In order to select a suitable method, we need to further study the impact of the geometric constraint on the smoothing method.

5   CONCLUSIONS

In this work we introduced a method to insert a geometric detail into an existing mesh. The approach consists in selecting an initial good set of hexahedra, so as to simplify the curves' and surfaces' classification steps that could prove overly difficult otherwise. This is a strictly *a priori* selection, and slight changes could be applied to the selection, i.e. adding or removing a select few hexahedra in order to improve quality or robustness.

A lot of work remains to be done concerning robustness; for example we have at the moment ignored the possibility that during vertices' classification a vertex v could have more adjacent curves than there are adjacent boundary edges to the nearest node, not to mention any boundary nodes, hence an impossiblity to classify curves. Such an issue could be resolved by refining the mesh around good nodes candidates, thus adding adjacent edges to those nodes. Same wise the hexahedra selection must form a 3-manifold, as that is an essential property for the surfaces' classification step; the geometric criteria that we currently use, i.e. keeping hexahedra which are at least half located inside G_2 is not sufficient and needs to be supplemented with topological criteria. Concerning quality, sheets insertion needs to be further developped in order to adress and correct badly shaped cells that can have several edges or even faces classified on the same curve or surface. This could be done considering [2].

REFERENCES

[1] P. Knupp J. Shepherd, S.A. Mitchell and D.R. White. Methods for multisweep automation. In proceedings of the 9th International Meshing Roundtable, pages 77–87, 2000.

[2] F. Ledoux, N. Le Goff, S. J. Owen, M. L. Staten, and J.-Ch. Weill. A constraint-based system to ensure the preservation of sharp geometric features in hexahedral meshes. In proceedings of the 21st International Meshing Roundtable, pages 315–332. Sandia National Laboratories, September 2012.

[3] F. Ledoux and J. Shepherd. Topological and geometrical properties of hexahedral meshes. Engineering with Computers, 26(4):419–432, 2010.

[4] L. Maréchal. Advances in octree-based all-hexahedral mesh generation: Handling sharp features. In Brett W. Clark, editor, proceedings of the 18th International Meshing Roundtable, pages 65–84. Springer, 2009.

[5] S. J. Owen and J. F. Shepherd. Embedding features in a cartesian grid. In proceedings of the 18th International Meshing Roundtable, pages 116–138. Sandia National Laboratories, October 2009.

[6] J. Qian and Y.e Zhang. Sharp feature preservation in octree-based hexahedral mesh generation for cad assembly models. In proceedings of the 19th International Meshing Roundtable, pages 243–262. Sandia National Laboratories, October 2010.

[7] J.-F. Remacle and M.S. Shephard. An algorithm oriented mesh database. International Journal for Numerical Methods in Engineering, 58(2), 2003.

[8] R. Schneiders. An algorithm for the generation of hexahedral element meshes based on a octree technique. In proceedings of the 6th International Meshing Roundtable, pages 183–194, 1997.

[9] J. F. Shepherd. Conforming hexahedral mesh generation via geometric capture methods. In Brett W. Clark, editor, proceedings of the 18th International Meshing Roundtable, pages 85–102. Springer, 2009.

[10] R. Shonkwiler. Computing the hausdorff set distance in linear time for any lp point distance. Information Processing Letters, 38:201–207, 1991.

[11] T.J. Wilson, J. Sarrate, X. Roca, R. Montenegro, and J.M Escobar. Untangling and smoothing of quadrilateral and hexahedral meshes. In B.H.V Topping, editor, Proceedings of the Eighth International Conference on Engineering Computational Technology. Civil-Comp Press, Stirlingshire, UK, Paper 36, 2012.

[12] K. H. Lee Y. Su and A. Senthil Kumar. Automatic hexahedral mesh generation for multi-domain composite models using a hybrid projective grid-based method. Computer-Aided Design, 36:203–215, 2004.

[13] T. J. R. Hughes Y. Zhang and C. L. Bajaj. An automatic 3d mesh generation method for domains with multiple materials. Computer Methods in Applied Mechanics and Engineering, 199(5-8):405–415, January 2010.

[14] Y. Zhang and C. Bajaj. Adaptive and quality quadrilateral/hexahedral meshing from volumetric data. In proceedings of the 13th International Meshing Roundtable, pages 365–376. Sandia National Laboratories, September 2004.

IMPROVEMENT IN PERFORMANCE PARAMETERS BY SHAPE OPTIMIZATION OF A CONICAL FLOW AROUND DIFFUSER

M. LENARCIC[∗], S. ERNE[∗] AND C. BAUER[∗]

[∗]Institute for Energy Systems and Thermodynamics (IET)
Vienna University of Technology
Getreidemarkt 9/E302, 1060 Vienna, Austria
e-mail: markus.lenarcic@tuwien.ac.at, www.iet.tuwien.ac.at

Keywords: Optimization, Conical Flow Around Diffuser, 3D Flow, Performance Parameter, Turbulence Model

Abstract. *This work aims to develop a fully automated shape design optimization of a 3D conical flow around diffuser with fixed main dimensions surrounded by a turbulent incompressible flow. The optimization process is based on OpenFOAM-1.6-ext in combination with a meta-model assisted evolutionary algorithm (MAEA). To improve the global diffuser performance, typical cost functions are considered describing the corresponding operation of the diffuser. The inner and outer contours of the 3D conical flow around diffuser walls are each independently parameterized with a smooth Bézier-Spline of 4th-order. To provide an additional degree of freedom for the optimization, the inner and outer diffuser wall are either splitted into two continuous segments. Enabling a discontinuous transition in flow direction, the resulting effect on the flow behavior and performance parameters as well are investigated. The obtained results are additionally compared with simulation results of a commercial code.*

1 INTRODUCTION

Diffusers are mounted downstream of turbine impellers to convert the remaining kinetic energy into pressure by decelerating the flow. The flow rate of the medium passing through the turbine is significantly affected by the diffusers geometry. In consequence, measures like geometrical restrictions must be taken to avoid adverse flow.

For this reason, a fully automated shape design optimization of a 3D conical diffuser with fixed main dimensions surrounded by turbulent incompressible free flow has been developed for improving performance parameters such as pressure recovery coefficient and diffuser efficiency. By using a fully 3D mesh, any three dimensional effects like intermittently appearing flow separation can be captured. A metamodel assisted evolutionary algorithm (MAEA) is embedded in the standalone optimization tool EASY [1], which interacts with the mesh motion- and CFD-solver included in the CFD-package OpenFOAM-1.6-ext. Regarding the optimization, reasonable cost functions characterizing the proper operation of the diffuser are considered, such as total pressure loss Δp_{tot} and pressure recovery coefficient C_p. For mesh motion, diffuser wall shapes are parametrized with a smooth Bézier-Spline of 4th-order, see Fig. 2 (top). For providing the possibility of establishing a so-called shock-diffuser at the diffuser outlet during optimization, an additional degree of freedom is implemented according to Fig. 2 (bottom) in order to investigate the influence of a discontinuity transition at the inner and outer diffuser wall. To preserve appropriate inlet profiles, a fully 3D physical model of the hydrokinetic turbine with implemented original diffuser are steady-stately calculated and 1D-profiles for velocity- and turbulence-quantities of the swirling flow exiting the impeller are assumed as inlet boundary condition. Strong convex curvature occurring in the diffuser necessitates the application of a modified two equation $k - \omega$ shear stress model with streamline-curvature correction (SST-CC k-ω) [6] to capture the highly turbulent flow.

In order to evaluate the optimization process, the pressure recovery coefficient obtained by optimized geometry is compared with results from both original- and conical diffuser geometries and with commercial code. For comparison purposes and for getting better insight into the diffuser performance under realistic flow conditions, a free surface flow simulation is done. Additionally, some basic aspects of swirling flow through conical diffuser enhancing vortex formations are briefly mentioned.

1.1 Numerical Details and Boundary Conditions

The flow simulations consider an incompressible isothermal fluid flow using the Reynolds-averaged Navier-Stokes (RANS) method. For the steady-state pressure-velocity coupling the SIMPLE-Algorithm was used, for the transient one the PISO-Algorithm. For all convective terms a 2nd-order interpolation scheme as well as a 1st-order implicit Euler scheme for time discretization was applied. To enhance stability of the SIMPLE velocity-pressure coupling, moderate under-relaxation factors were defined. The mesh was generated in accordance to the y^+-criterion ($y^+ \geq 30$) required from turbulence models using a standard wall function. During mesh movement, a constant cell height of cells near the moving walls was ensured.

2 METHODS AND NUMERICAL SETUP

The set of boundary conditions was defined as usual for ducted single phase flow. To avoid numerical oscillations at the domain outlet a fixed mean static pressure boundary condition was prescribed. For the turbulence quantities and static pressure zero gradient boundary conditions were chosen at the domain inlet. A 1D-profile including $U_1(r)$, $k_1(r)$ and $\omega_1(r)$ was given at
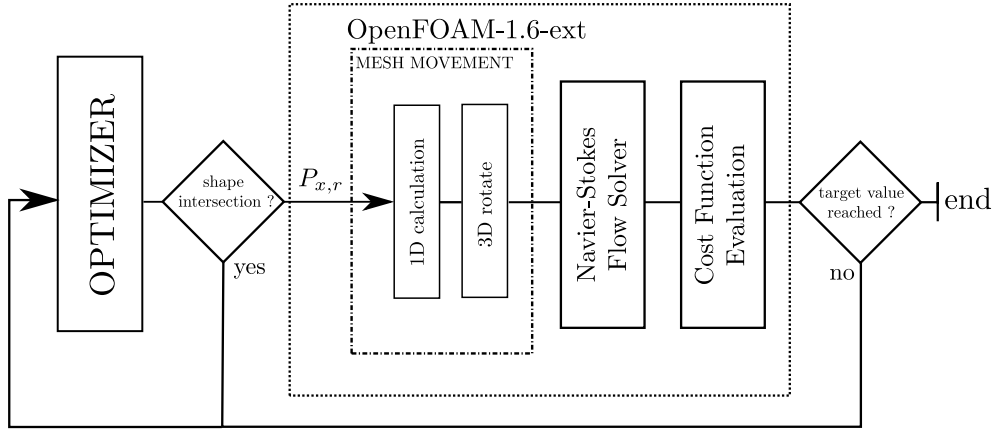
Figure 1: Work flow of the optimization process.

the diffuser inlet, assuming an axisymmetric inflow. At the ambient inlet, a permanent bulk velocity U_{1AMB} (free flow velocity) was predefined, see Fig. 2. To prescribe the turbulent quantities at the ambient inlet, the turbulent kinetic energy k and the dissipation rate ω were set with $k = 1.5(U_{1AMB}I)^2$ and $\omega = C_\mu^{0.75}k^{1.5}/l_t$. Based on previous studies, the turbulent intensity $I = 0.03$ and an uniform turbulent mixing length $l_t = 0.01m$ were expected at the domain inlet. The distance between the solid diffuser wall and the free slip boundary defined at the farfield was chosen sufficiently big, see Fig. 4 . For demonstrating the effect of the diffuser geometry contour, a comparison between optimized- (OPT$_D$/OPT$_C$ with or without either one discontinuous transition along the inner and outer diffuser walls), original- (ORIG) and conical (CON) wall shapes with same grid refinement was simulated.

2.1 Mesh Movement and Parameterization

For the parameterized boundary mesh movement, a mesh motion solver based on Laplacian smoothing was applied. A variable quadratic diffusivity was prescribed during deformation to keep distortions in the boundary region as low as possible. The vertices of the inner and outer wall shapes of the 3D conical flow around diffuser were each independently parameterized with a smooth Bézier-Spline of 4th-order and rotated by $360°$ yielding a 3D circular contour. Such a curve of degree N is defined by the following parameterization

$$\mathbf{X}(t) = \sum_{i=0}^{N} \binom{N}{i}(1-t)^{N-i}t^i\mathbf{P}_i, \text{ with } t \in [0,1] \tag{1}$$

and is therefore ideally suited for varying the order of the Bézier-Spline. To ensure clarity about the effect of a discontinuous transition along the inner and outer diffuser walls, their contours were parametrized both with a polynomial of 3rd-order and 1st-order, enabling a discontinuity, as can be seen in Fig. 2 (bottom). On the other hand, the contour is defined by a Bézier-Spline of 4th-order causing smooth contour shapes, see Fig. 2 (top). Due to restrictions on the optimizer's directives, the inlet- and outlet coordinates of the diffuser wall shapes were assumed to be constant, that means P_0, $P_N = const$. Additionally, with considering a discontinuity along the diffuser walls, P_{N-1} was also assumed to be constant ($P_{N-1} = const$), see Fig. 2 (bottom). All integrated values describing the diffuser characteristic were calculated at the diffuser inlet R_1 and -outlet R_2.

As a compromise between accuracy and computational costs the diffuser walls are only parameterized with a smooth Bézier-Spline of max. 4th-order , which have already provided
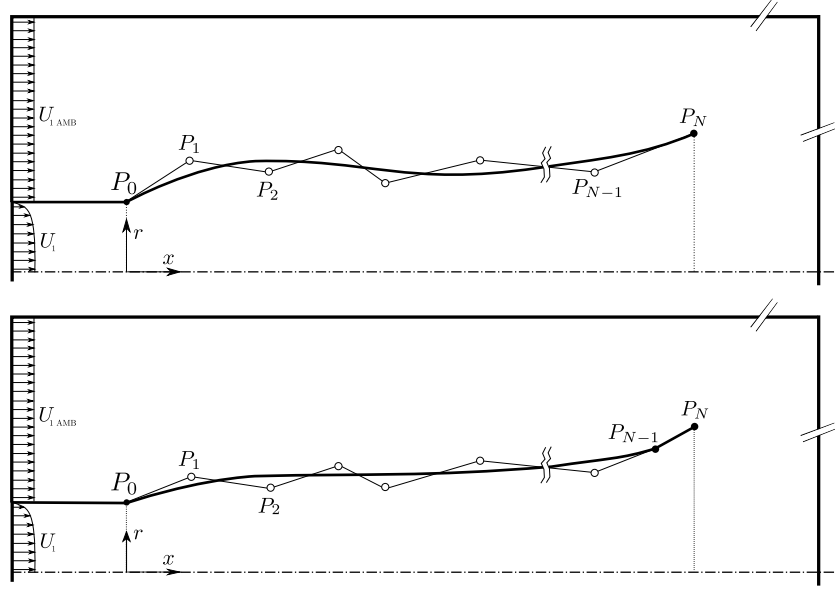
Figure 2: Interior diffuser boundary with a Bézier-Spline Parametrization of N-order (top) or with an additional Bézier-Spline Parametrization of 1st-order for considering a discontinuity (bottom); (dashed) evaluation plane.

appropriate results for a quasi 2D wedge type diffuser [9].

2.2 Mesh Independence Study

For the given problem a mesh independence study with the conical diffuser was performed to determine the required number of computational cells. Calculations of Δp_{tot} at three stages of grid refinement ($h = 1, 2, 4$) were introduced and opposed to an estimated value $f_{h=0}$ from a Richardson Extrapolation [8] with higher order based on the lower order values f_{h_i}, see Tab. 1 and Fig. 3. The estimation $f_{h=0}$ becomes

$$f_{h=0} \approx f_{h=1} + \frac{f_{h=1} - f_{h=2}}{r^p - 1} \qquad (2)$$

whereas r is the grid refinement and p the formal order of accuracy of the algorithm. The order of convergence is defined as $p = ln(f_{h=4} - f_{h=2})/(f_{h=2} - f_{h=1}))/ln(2)$ and h prescribes the grid spacing. So, one can estimate the value of a quantity f as the grid spacing goes to zero ($h \rightarrow 0$). The Grid Convergence Index (CGI), based upon a Richardson error estimator, represents a

$Grid$	h_i	$r = h_{i+1}/h_i$	$f_{h_i}/f_{h=4}$
-	-	-	m^2/s^2
A^{fine}	1	2	0.4546
B	2	2	0.4747
C	4	–	1

Table 1: Three different grid refinement stages with the integration variable Δp_{tot} as quantity of f_{h_i}, Normalized Grid Spacing h_i.

$|\epsilon_{21}|$	$|\epsilon_{42}|$	$\epsilon_{21}/\epsilon_{42}$	p	GCI_{21}	GCI_{42}
10^{-1}	10^{-1}	–	–	$\%$	$\%$
0.42	5.25	0.08	4.70	0.22	2.62

Table 2: Grid Convergence Index for three integration variables at different mesh refinements.

measurement of the difference between the computed data and the asymptotic numerical value.

The CGI on the fine grid reads

$$CGI_{fine} = \frac{F_s|\epsilon_{21}|}{r^p - 1},$$ (3)

while the CGI of a coarser one is defined as $CGI_{coarse} = F_s|\epsilon_{42}|r^p/(r^p-1)$, wherein the relative error is $\epsilon_{i+1,i} = (f_{h_{i+1}} - f_{h_i})/f_{h_i}$. The incorporated safety factor of $F_s = 1.25$ is suggested in several works. As can be seen from Tab. 2, the convergence ratio $R = \epsilon_{21}/\epsilon_{42} < 1$ shows a monotonic convergence behavior.
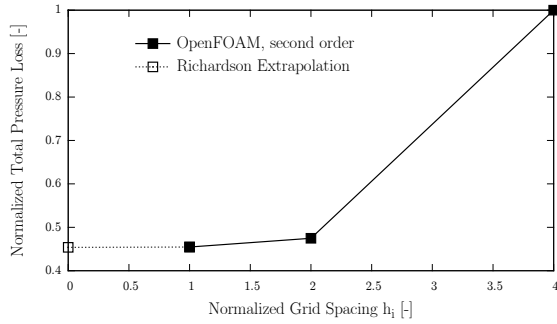
Figure 3: Calculated Δp_{tot} at different refinement stages against the asymptotic approximation obtained from the Richardson Extrapolation.
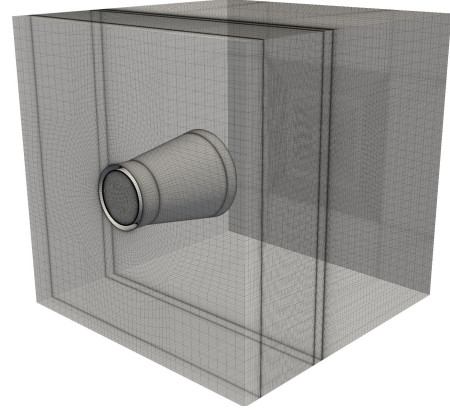
Figure 4: Computational area of the conical diffuser.

Due to high computational costs during the time-consuming optimization process, the medium size grid ($h = 2$) with about 2×10^6 nodes was chosen for all calculations. Figure 4 illustrates a typical computational grid used in the present work.

2.3 Flow Characteristics

In order to predict the diffuser efficiency, the well known pressure recovery coefficient C_p as an averaged integrated value was used during optimization. Based on theoretical considerations, a simple estimation of C_p can be derived from geometrical conditions only. After applying the Bernoulli-Equation along a streamline from in- to outlet of the diffuser, the resulting ideal pressure recovery coefficient is given by

$$C_{pi} = 1 - \frac{1}{AR^2},$$ (4)

which only depends of the area ratio $AR = D_2^2/D_1^2$ of the diffuser geometry.

The diffuser performance is characterized by the flow averaged pressure recovery coefficient using the normalized total kinetic energy $0.5|U_1|^2$ at the diffuser inlet

$$C_p = \frac{\displaystyle\int_{\psi_2} p_2 \, d\psi_2 - \int_{\psi_1} p_1 \, d\psi_1}{0.5 \displaystyle\int_{\psi_1} |U_1|^2 \, d\psi_1}$$ (5)

where ψ is the face flux at the calculation sections. Based on flowrate weighted values, the inlet swirl is defined by the swirl number

$$S = \frac{\displaystyle\int_{\psi_1} U_{ax} U_\theta r \, d\psi_1}{R_1 \displaystyle\int_{\psi_1} U_{ax}^2 \, d\psi_1},\tag{6}$$

whereas the characteristic radius is the diffuser inlet radius R_1. The quantity represents the ratio between angular and stream-wise momentum of the diffuser flow at the domain inlet. To ensure simulation accuracy, all cost functions and performance quantities used in the optimization process are averaged.

Due to swirl inflow, centrifugal forces gain a radial pressure distribution in the diffuser, which leads to a low pressure level at the diffuser centerline. At a certain level of inlet swirl, defined by the swirl number S, the arising of an axisymmetric streamwise vortex may be a cause of worsening the diffuser performance. With increasing swirl, the pressure level at the diffuser centerline decreases due to low velocity. So, the resulting dead water region reduces the effective diffuser cross section similar to an annular discharge channel. To quantify the influence of this cross-sectional area reduction the variable $D_{2,vc}$ is introduced characterizing the magnitude of the vortex core:

$$AR_{vc} = \frac{D_2^2 - D_{2,vc}^2}{D_1^2}\tag{7}$$

Hence the corrected ideal pressure recovery coefficient is given now by $C_{pi,vc} = 1 - 1/AR_{vc}^2$. Here, the outer diameter $D_{2,vc}$ was taken from a simple post simulation process, which extracts the vortex core region inside the diffuser. Information about the bounding streamline between main flow and vortex flow was obtained by applying a simple filter function of $|u| \to 0$, which was found to be sufficiently accurate for the present study. Interpolation between those calculated discrete points was done by a modest polynomial interpolation $D_{2,vc}(x) = \sum a_i x^i$ $(i = 0 \ldots 3)$.

Since the diffuser geometry strongly influences the occurrence of undesired flow phenomena inside the diffuser, the basic geometry specifications of the present diffuser are characterized by parameters like the non-dimensional length L/D_1 and the non-dimensional area ratio AR for conical diffusers, as defined in Eq. 8 and 9.

$$\frac{L}{D_1} \approx 1.7\tag{8}$$

$$AR = \left(1 + 2\frac{L}{D_1} tan\theta\right)^2 \approx 2.9\tag{9}$$

2.4 Optimization Strategy

Optimization of hydraulic flow devices often involves time and is associated with computational cost, especially when optimizing complex 3D geometries. One efficient possibility is applying an evolutionary algorithm (EA) with an inexact pre evaluation procedure for optimization. By introducing low-cost surrogate evaluation models, also called "metamodels",

a significantly decreasing number of calls can be reached. Thus, this method aims to reduce computational resources caused by the evaluation code, here CFD. Further details on the subject can be found in [2]. The optimization software EASY used in this study, is based on Metamodel-Assisted-Evolutionary-Algorithms, widely denominated as MAEA, which are presented thoroughly in [3]. Another time-saving technique for global optimization is called the hierarchical distributed MAEA (DMAEA) yielding a drastic reduced number of evaluations by introducing separately handled subpopulation.

To find the optimal hydraulic layout of a 3D conical diffuser a (μ, λ) MAEA is coupled with the open source CFD software package OpenFOAM-1.6-ext solving in parallel. This technique subdivides the main population (offspring, parent and elite) into three different sub populations, which are interacting among themselves over specific evolution operators. As discussed above, metamodels are highly beneficial towards performance issues and need a database to store the individuals. The implemented MAEA utilizes locally trained metamodels, for which neighbor entries in the database are selected. λ radial basis function networks [4] are applied to pre evaluate the population members before they are evaluated by problem specific tools and stored in the database.

3 TURBULENCE MODEL VALIDATION

Outflow of an axial turbine is often characterized by a strong circumferential flow component, which can alter the flow regime in the diffuser radically. Appearing angular momentum leads to streamline curvature as well as a radial pressure gradient, which results in a low pressure field at the diffuser centerline. Turbulent axisymmetric flow with swirl appears in several industrial flow devices and needs to be addressed in a more specific way.

Thus, a turbulence model with modified production terms of the turbulent kinetic energy following [6] was implemented into OpenFOAM-1.6-ext, which considers the turbulence anisotropy of swirling flow. The correction factor f_{corr} was defined in Eq. 10 and multiplied in the transport equations for both k and ω. In simple terms, it should be noted that r^* and $\widetilde{r}$ represent a function of the fluid normal stress $\mathbf{S}$ and shear stress tensor $\mathbf{\Omega}$. Flow with vanishing streamline curvature therefore yields $f_{\text{corr}} = 1$.

$$f_{\text{corr}} = (1 + c_{r1}) \frac{2r^*}{1 + r^*} \left(1 - c_{r3} \tan^{-1}(c_{r2} \widetilde{r}) - c_{r1} \right) \tag{10}$$

The turbulence models were briefly tested using a three-dimensional case [5] at a moderate swirl level of $S = 0.388$ (see Eq. 6). As can be seen in Fig. 11, the original SST k-ω model underestimates the angular velocity W in the wall near region, whereas the center line velocity is predicted well. The corrected model shows a slight improvement of the angular velocity although an overestimation at the end of the divergent part is observed, leading to a minor backflow in the center region.

4 RESULTS

4.1 IDEAL PRESSURE RECOVERY COEFFICIENT C_{pi}

To point out the influence of strong dead water flow at the diffuser center line on the effective cross-sectional area, the optimized geometry is taken as an representative example. The curves of different ideal pressure recovery coefficients $C_{pi}(x)$ at swirl inflow ($S = 0.07$, calculated with Eq. 6), are shown in Fig. 6. The development of C_{pi} using the common formulation Eq. 4 shows the best pressure recovery and ends up at a value over 0.9. Based on the consideration of
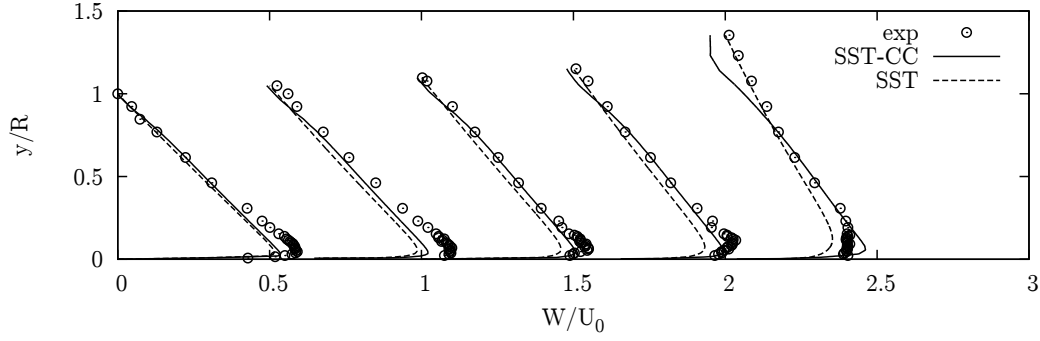
Figure 5: Comparison of circumferential velocity W using standard SST k-ω and SST-CC k-ω.

a corrected area ratio AR_{vc} (Eq. 7), the obtained $C_{pi,vc}$ clearly shows a weakening trend towards diffuser performance of about 6.2%, readily seen in Fig. 6. Figure 7 illustrates the shape of the optimized wall shape and vortex boundary streamline respectively. Both curves bear a certain geometrical resemblance to one another.
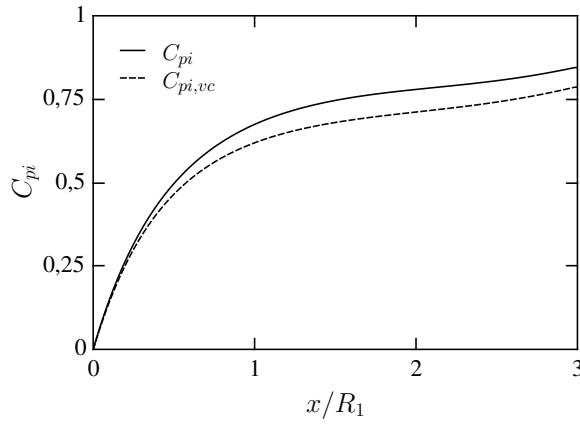
Figure 6: Development of the pressure recovery coefficient in the conical diffuser.
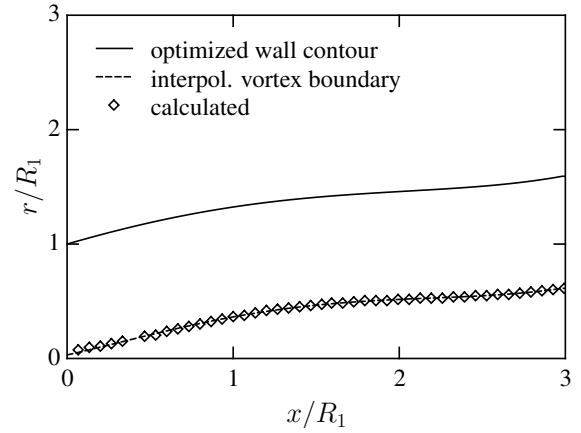
Figure 7: Wall contour of the diffuser wall and the interpolated vortex core bounding streamline.

4.2 OPTIMIZATION

Table 3 shows steady-stately and transiently calculated pressure recovery coefficients C_p for the different diffuser geometries ORIG, CON, OPT$_C$ and OPT$_D$. Due to steady-state optimization process using C_p as cost-function, a significant increase of C_p compared to the original- and conical diffuser is gained, yielding a concave-flat-convex curved wall shape, as can be seen in Fig. 8. Comparable results have been already obtained in [9, 10], for additional qualitatively descriptions of the results see [10, 11, 13].

In order to take time depending factors into account, transient investigations with a maximum Courant Number of $Co_{max} = 1$ were performed using a monitoring period of 15 seconds from reaching a converged state. Comparing steady-state- and transient results, significant differences in C_p occur with the optimized geometries OPT$_C$ and OPT$_D$. While OPT$_C$ achieves a higher C_p during steady-state optimization process, the value drops more than for OPT$_D$ if transient effects are regarded. This can be attributed to a higher sensitivity of the steady OPT$_C$-contour with respect to time-dependent influences. Compared to the original diffuser ORIG, the improvements of C_p according to transient simulations correspond to a respective percentage

Geo	C_{p_S}	C_{p_T}	$\Delta C_{p_T}*$	$\Delta C_{p_{S-T}}**$
-	-	-	%	%
$ORIG$	0.459	0.438	0.00	4.60
CON	0.534	0.506	15.5	5.20
OPT_C	0.680	0.580	32.4	14.7
OPT_D	0.643	0.592	35.2	7.90

Figure 8: Optimized diffuser wall shapes OPT_C and OPT_D using C_p as cost-function.

Table 3: Comparison of the steady-stately calculated pressure recovery coefficient C_{p_S} and the transiently calculated one C_{p_T} for the different geometries ORIG, CON, OPT_C and OPT_D; difference of C_{p_T} related on ORIG geometry (*), difference between steady-state and transient results (**).

of approximately 32% for OPT_C and 35% for OPT_D - and related to the conical diffuser CON of approximately 15% for OPT_C and 17% for OPT_D. Figure 9 and Fig. 10 show the influence of different diffuser wall shapes on the established flow pattern. A strongly pronounced hub vortex formation displaces the remaining fluid towards the diffuser walls leading to an energy enrichment of the boundary layer, which in turn prevents flow separation in those regions.
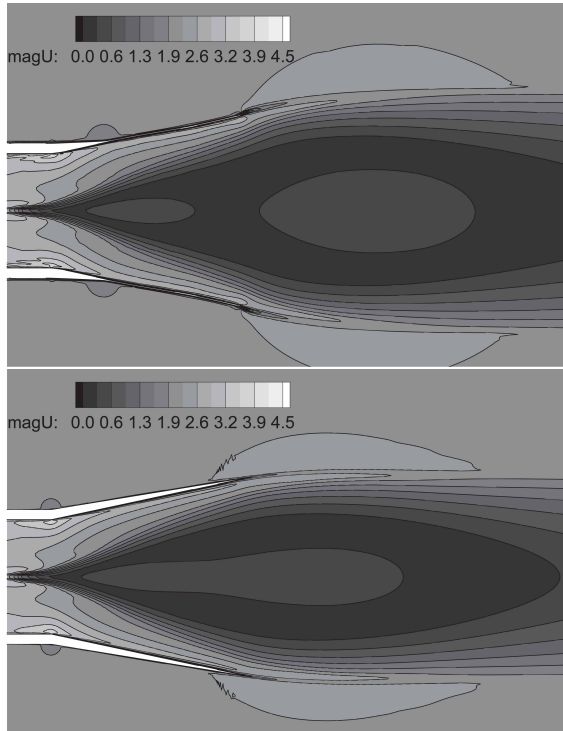
Figure 9: Comparison of the transiently calculated velocity U for the diffuser geometries ORIG and CON.
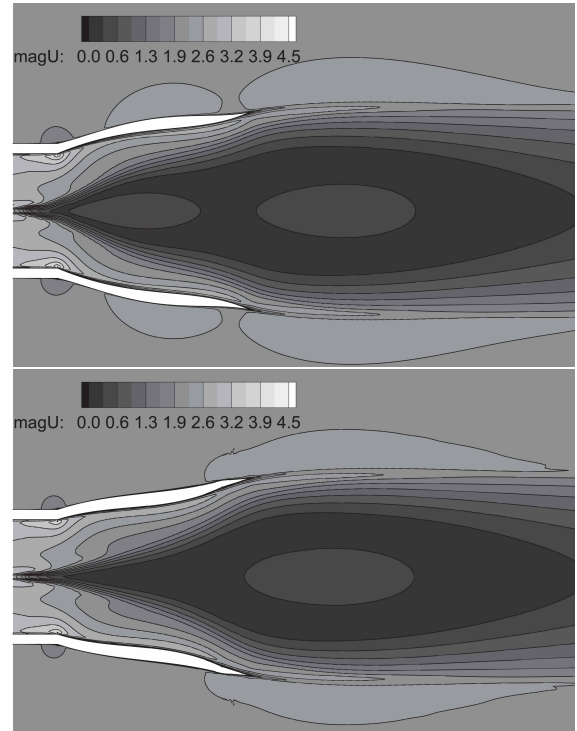
Figure 10: Comparison of the transiently calculated velocity U for the optimized diffuser geometries OPT_C and OPT_D.

Since hub vortex structures are less developed for OPT_D, a higher diffuser efficiency is reached than for the other optimized geometry OPT_C. Additionally, the discontinuity at the

OPT$_D$-contour stabilizes the flow and time sequence of C_p and Δp$_{tot}$ are about half the size of amplitudes for OPT$_C$. Despite transient effects, an almost steady flow field is induced with a standard deviation for C_p and Δp$_{tot}$ of $SD_{C_p} < 3 \times 10^{-3}$ and $SD_{\Delta p_{tot}} < 5 \times 10^{-3}$ for all investigated geometries.

Compared to transient results from commercial code, differences of C_p and Δp$_{tot}$ are less than 1.5%.

4.3 FREE SURFACE FLOW

Hydrokinetic turbines driven by river current often have mounted a diffuser or augmented channel unit downstream improving the flow characteristics of the turbine. To study the diffuser characteristics under more realistic conditions, free surface flow simulations in a gravitational field with appropriate inflow conditions were performed. A constant inflow water level and a unique flow velocity for both fluid and air was prescribed at the ambient inlet boundary. The side walls were defined again as symmetric planes.
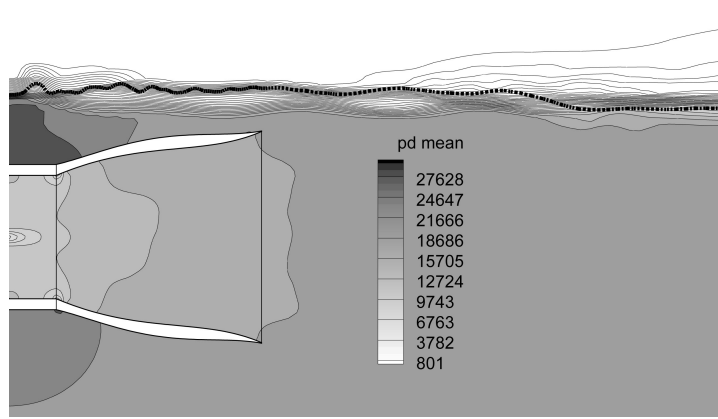


Figure 11: Contour plot of static pressure distribution p_d at y-z plane for the optimized geometry OPT$_C$. (dashed line) instantaneous contour of free surface; p_d in N/m^2.

The diffuser inflow conditions remain the same as those used in the single phase flow. Numerical details and discretization methods the used flow solver is based on, can be found in [7]. For the present study a coarsened block-structured mesh was used, however, with the same boundary layer resolution. The maximum Courant Number was $Co_{max} < 0.25$ during the transient simulation of 60 seconds, after reaching a quasi steady-state flow solution. Compared with single phase studies using the same computational settings, the free surface flow simulation tends to stronger fluctuations in the diffuser characteristics. As the simulation additionally takes account of the hydrostatic pressure distribution and free surface stress forces, the variation over time of C_p is increased significantly.

Figure 11 illustrates the static pressure distribution including the low pressure region in the outer air flow. Even in the two phase simulation an improvement of C_p due to wall shape optimization comes apparent. Compared to the conical diffuser CON, the OPT$_C$ shows an improvement corresponding to a respective percentage increase of approximately 10%. A comparison between the time-averaged pressure recovery coefficient of the single phase- and two phase simulation shows that they are broadly in the same range, as can be seen in in Tab. 4.

Figure 12 draws streamlines of both CON and OPT$_C$ diffuser, where the simulation using the optimized wall shape shows less single vortices in the backflow region. Also, the streamwise growth of the vortex core inside the optimized diffuser is significantly weaker than in the

Geometry	$C_{p_{1Ph}}$	$C_{p_{2Ph}}$	$\Delta C_{p_{1Ph-2Ph}}$*
-	-	-	%
CON	0.506	0.492	2.8
OPT_C	0.580	0.548	5.5

Table 4: Comparison of transiently calculated pressure recovery coefficient C_p for the geometries CON and OPT_C; single phase- (1Ph) and two phase simulation (2Ph); difference of C_p related on results from 1Ph calculation (*).

original diffuser observing a more stable vortex structure. Both plots present a reduction of the diffuser area due to vortex formations at the centerline and a strongly disturbed flow after the diffuser.
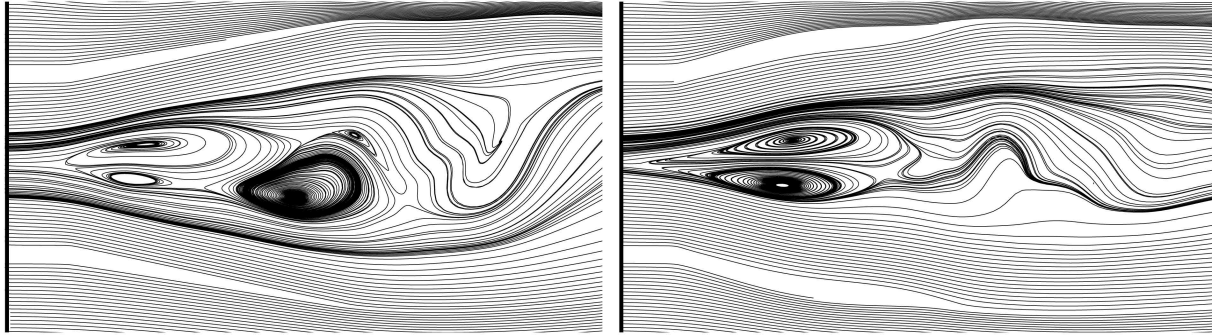
Figure 12: Time averaged velocity streamlines projected on y-z plane; left: conical contour (CON), right: optimized contour (OPT_C).

5 CONCLUSION

In this work an efficient coupling of OpenFOAM-1.6-ext with the optimization tool EASY using a generalized metamodel assisted evolutionary algorithm (MAEA) is outlined. During fully automated shape design optimization of a 3D conical flow around diffuser, the inner and outer diffuser walls are each independently parameterized with a smooth Bézier-Spline of max. 4th-order. A modified version of the SST k-ω model is applied to capture streamline curvature effects inside the diffuser flow. The influence of a discontinuity transition at the inner and outer diffuser wall is carried out as well. The resulting optimized geometries show a comparable diffuser efficiency, leading to a moderate increase of C_p compared to the original- and conical geometry. Compared to the original diffuser ORIG, the improvements of C_p according to transient simulations correspond to a respective percentage of approximately 32% for OPT_C and 35% for OPT_D - and related to the conical diffuser CON of approximately 15% for OPT_C and 17% for OPT_D. The qualitative progress of the optimized curve shapes are quite similar to those of previous diffuser investigations by [12, 9, 10]. Transient single phase simulations cause an almost steady flow field with negligible standard deviations of C_p.

A comparison between the time-averaged pressure recovery coefficient C_p calculated by transient single phase- and two phase simulations for conical and optimized diffuser geometries shows appropriate agreements with each other. However, the free surface flow simulation has shown significantly stronger pressure fluctuations appearing more realistic, but is found to be too time consuming for an evaluation code application.

Further studies are aimed at developing a wall shape optimization of an oval 3D flow around

diffuser using Bézier-Surfaces for parameterization. Additionally, a simplified wall blowing concept is applied on the turbine hub in order to reduce vortex formations.

REFERENCES

[1] K.C. Giannakoglou: The EASY (Evolutionary Algorithms SYstem) software, http://velos0.ltt.mech.ntua.gr/EASY(2008).

[2] K.C. Giannakoglou: Design of optimal aerodynamic shapes using stochastic optimization methods and computational intelligence. *Progress in Aerospace Sciences*, 38(11) (2002), 43–76.

[3] I.C. Kampolis and K.C. Giannakoglou: Distributed Evolutionary Algorithms with Hierarchical Evaluation. *Engineering Optimization*, 44(11) (2009), 1037–1049.

[4] S. Haykin: Neural Networks: A Comprehensive Foundation. *Pearson Education*, - (2001).

[5] P. D. Clausen, S.G. Koh, and D.H. Wood *Measurements of a swirling turbulent boundary layer developing in a conical diffuser*. Fluid Mechanics and Heat Transfer, 6 (1993), 39-48.

[6] P. E. Smirnov and F. R. Menter *Sensitization of the SST Turbulence Model to Rotation and Curvature by Applying the Spalart-Shur Correction Term*. Journal of Turbomachinery, 131 (2009), 041010-1 - 041010-8.

[7] H. Rusche: Computational fluid dynamics of dispersed two-phase flows at high phase fractions. PHD Thesis, *Imperial College of Science, Technology and Medicine, UK* 2002.

[8] L.F. Richardson: The deferred approach to the limit. Part I. Single lattice. Part II. Interpenetrating lattices. Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character, 226 (1927), 299361

[9] S. Erne, M. Lenarcic, C. Bauer and S.A. Kyriacou: Shape optimization of a flow around circular diffuser in a turbulent incompressible flow, ECCOMAS, 2012

[10] M. Lenarcic, S. Erne and C. Bauer: Efficiency improvement by performing shape optimization of a circular flow around diffuser in a turbulent incompressible flow, 17th International Conference on Hydropower Plants, 2012

[11] Y. Ohya and T. Karasudani: A Shrouded Wind Turbine Generating High Output Power with Wind-lens Technology, Energies, 2010, ISBN 0-933283-00-8

[12] B. Schmandt and H. Herwig: Diffuser and Nozzle Design Optimization by Entropy Generation Minimization. *Entropy*, 13 (2011), 1380–1402.

[13] D. Japikse: TURBOMACHINERY DIFFUSER DESIGN TECHNOLOGY, Concepts ETI Inc., Norwich, 1984,

A SIMPLE RECOVERY BASED ERROR ESTIMATOR FOR THE GFEM INCLUDING BLENDING ELEMENTS

RAFAEL MARQUES LINS[*] AND SERGIO PERSIVAL BARONCINI PROENÇA[*]

[*] Department of Structural Engineering (SET)
São Carlos School of Engineering (EESC)
University of São Paulo (USP)
Avenue Trabalhador Sãocarlense, 400, 13566-590, São Carlos-SP, Brazil
e-mails: {mlins, persival}@sc.usp.br

Key words: Error Estimator, Singular Value Decomposition, Blending Elements, Generalized Finite Element Method.

Abstract. The generalized finite element method (GFEM) allows setting good conditions of local approximations by means of enrichments functions with special features. One major advantage of this method is the mesh independence for crack simulations. Moreover, the strong gradients typical of linear elastic fracture mechanics problems can be faced by customized enrichments. In spite of the good performance presented by the method on several numerical simulations it is very important to assess error estimates from the obtained results. In this context, a new a posteriori error estimator for the generalized finite element method is hereby considered aiming mainly to guarantee the more accurate and reliable stress distribution. A secondary aim is to employ the error estimates as indicator in hp-adaptive strategies. The proposed estimator is then based on the superconvergent patch recovery (SPR) technique, a widely used technique for evaluating recovered stress fields from the conventional finite element solutions. The GFEM-SPR procedure explores the clouds and partition of unity concepts to obtain recovered stress fields from interpolation polynomials. Such functions are identified using the singular value decomposition (SVD) strategy over superconvergent point values defined in each cloud in coincidence with the quadrature integration points. A particular issue that appears when enrichment is imposed over a localized region of the solid domain is related to the so called blending elements. Such elements blend nodes with and without enrichment, however presenting lack of partition of unity property and, hence, also losing the reprodutibility feature. Thus, the accuracy and convergence ratio of the GFEM can be affected. In this paper a modification in the standard GFEM proposed in literature to properly account for blending elements is adopted and the error estimator is also improved. Some benchmarks problems discretized by two-dimensional triangular and quadrilateral element meshes are presented in order to assess the efficiency and computational performance of the procedure hereby proposed. The energy norms of the recovered solutions, as well the effectivity index of the estimator are presented by comparing numerical and analytic solutions when available.

FROM SEGMENTED MEDICAL IMAGES TO SURFACE AND VOLUME MESHES, USING EXISTING TOOLS AND ALGORITHMS

CLAUDIO LOBOS*, RODRIGO ROJAS-MORALEDA[†]

*,[†] Departamento de Informática
Universidad Técnica Federico Santa María
Av. España 1680, 2390123, Valparaíso, Chile
e-mail: clobos@inf.utfsm.cl*
rodrigo.rojas@postgrado.usm.cl[†]

Key words: Surface and Volume Meshing, Segmented Images, Finite Element Method.

Abstract. In a medical context, one of the most used techniques to produce an initial mesh (starting from segmented medical images) is the Marching Cubes (MC) introduced by Lorensen and Cline in [1]. Unfortunately, *MC* presents several issues in the meshing context. These problems can be summarized in three types: topological (presence of holes), of quality (sharp triangles) and accuracy in the representation of the target domain (the staircase effect). Even though there are several solutions to overcome topological and quality issues, the staircase effect remains as a challenging problem.

On the other hand, the Computational Geometry Algorithms Library (CGAL) [2], has implemented the *Poisson Surface Reconstruction* algorithm introduced in [3], which is capable of producing accurate and high quality triangulations based on a point set and its normal directions.

This paper shows how surface meshes can be produced using both, *MC* and CGAL. Moreover, starting from the generated quality surface mesh, this work also shows how volume meshes can be produced. Therefore, a complete workflow, starting from segmented medical images to surface and volume meshes, is introduced in this work. In particular, tetrahedral and mixed–element meshing techniques are presented to produce a simulation with the Finite Element Method.

1 INTRODUCTION

In the context of biomechanical modelling of human organs, the most common way to produce a mesh suitable for biomedical applications is to perform the following steps:

- Acquire volumetric medical images of the patient's organ using magnetic resonance imaging (MRI), ultrasound (US), computed tomography scanner (CT) or other imaging techniques.

- Achieve image segmentation in order to produce a cloud of points or an initial surface mesh defining the geometry of the modelled domain.

- Add internal nodes to produce the 3D elements that will conform the final volumetric mesh.

One of the most used methods for extracting surfaces from volumetric information arising from sources such as MRI data and CAT scans is *implicit modelling*, leading to an abundance of approaches. One of the most widespread approach is *Marching Cubes* (MC) [1], which belongs to the family of techniques that uses a structured grid for implicit modelling [4].

One apparent drawback of *MC* is the uniform sampling density which does not take the local surface curvature into account, producing problems like: *(i)* presence of holes or surface crossing (topology), *(ii)* sharp and flat triangles (quality) and *(iii)* the *MC* characteristic staircase effect. Even though many approaches have been proposed to control the mesh complexity, e.g., by adaptive octree descent with sophisticated refinements, the quality and the staircase effect remain as challenging problems.

The *Poisson Surface Reconstruction Method* (PSR) tackles these issues by estimating the surface as the iso-contour to an indicator function defined from the normal vectors to the *MC* boundary. The Computational Geometry Algorithms Library (CGAL) implements a variant of this algorithm which solves for a piecewise linear function [2].

This paper shows how to obtain high accuracy surfaces, that approximate a volume outer boundary from *MC* algorithms using a *PSR* method implemented in *CGAL*. Moreover, this paper shows how to obtain volume meshes starting from the acquired surface mesh.

2 MATERIALS AND METHODS

The overall proposed workflow is shown in Figure 1, where three main components are introduced: input data, surface meshing and volume meshing. The surface

meshing is based on the *MC* method, the rendered mesh is improved in order to use CGAL to apply a *PSR* method. The output of CGAL is finally used to construct a high quality volume mesh.
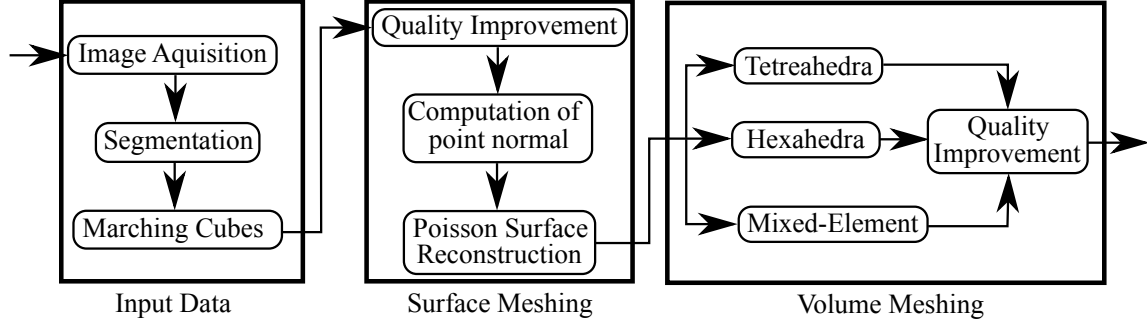
Figure 1: Main Workflow with three steps: Input Data, Surface Meshing and Volume Meshing.

2.1 Image acquisition and segmentation

Volumetric medical images from imaging acquisition technology, such as CT, MRI and PET, are represented by a stack of 2D image slices in 3D space, where the tissue type surrounding a Voxel (volumetric pixel) determines its value. The variations in tissue type give rise to varying intensity which is quantized as a scalar value known as grey level, or more often as vector or tensor values when Voxels contain multiple scalar values, e.g. in ultrasound. The segmentation problem is essentially a classification problem. A label representing the region to which an image Voxel belongs is assigned to each Voxel. The assignment is however, subject to some constraints, such as piecewise continuity and smoothness. Depending on the imaging technique, the segmentation could be difficult and a research area in itself [5].

2.2 Surface Meshing

The first step is to use *MC* to render a surface from segmented contours. The principle behind the *MC* algorithm is to subdivide space into a series of small cubes. By the values at its eight vertices -below or above an isovalue- the intersections at the cube edges are found. These intersections are used to generate triangles representing the isosurface by a lookup table of surface configurations of cubes. However, the *MC* shows important drawbacks for some surface configurations: the topology of the isosurface is not unique and the resulting surface of two adjacent neighbor cubes can produce: holes, poor quality triangles explained below and low accuracy in the representation known as the staircase effect.

With respect to holes in the triangle mesh, there are new versions of *MC* [4, 6], that address and solve this problem. Alternatively a conventional solution is the use of the *ear clipping* technique [7], adding new triangles to close the hole.

One of the most important quality metrics for surface meshes is the *minimal angle*. There are two types of poor quality triangles: sharp and flat. The first refers to a triangle that has one angle close to cero and two close to 90 degrees. A conventional solution is the edge collapsing technique, which will "join" the two nodes opposite to the "close to cero" angle. The second one has two angles close to cero and one close to 180 degrees; there is no proper or general solution for this case.

These repairing techniques allow to get a surface mesh without sharp triangles nor holes; the next section is dedicated to tackle the problem of low accuracy representation or staircase effect.

2.3 Quality meshes from implicit functions

Starting with a surface mesh free of holes and sharp triangles, problems like *staircase effect* and remaining *poor quality triangles* can be overcome by the *Poisson Surface Reconstruction* method (PSR) [2] which is available in the library CGAL. The *PSR* formulation considers [3] all points at once, without resorting to heuristic spatial partitioning or blending and so, is highly resilient to data noise. This approach allows a hierarchy of locally supported basis functions, and the solution reduces to a well conditioned sparse linear system.

CGAL takes as input a set of normal vectors (3D oriented set of points). The normal of each node is computed as the average of incident triangle normals. Given an oriented set of points. CGAL algorithm builds a 3D Delaunay triangulation from these points and refines it by *Delaunay refinement* so as to remove all badly shaped (non isotropic) triangles and to tessellate a loose bounding box of the input oriented points [2]. The *PSR* defines an indicator function $\tilde{\chi}$ as 1 at *points inside* the model, and 0 at *points outside*. A new quality surface is obtained by extracting an isosurface.

The principle behind this method is the relationship between oriented points (normal vectors) and the indicator function. The gradient of the indicator function is a vector field that is zero almost everywhere (since the indicator function is constant almost everywhere), except at points near the surface, where it is equal to the inward surface normal. Thus, the oriented point set can be considered as a sample points from the gradient of the indicator function shown in Figure 2, [3].

The indicator function thus reduces to inverting the gradient operator, i.e. finding the scalar function $\tilde{\chi}$ whose gradient best approximates a vector field $\vec{V}$ defined by the samples, i.e.

$$Min|\nabla \tilde{\chi} - \vec{V}|.$$
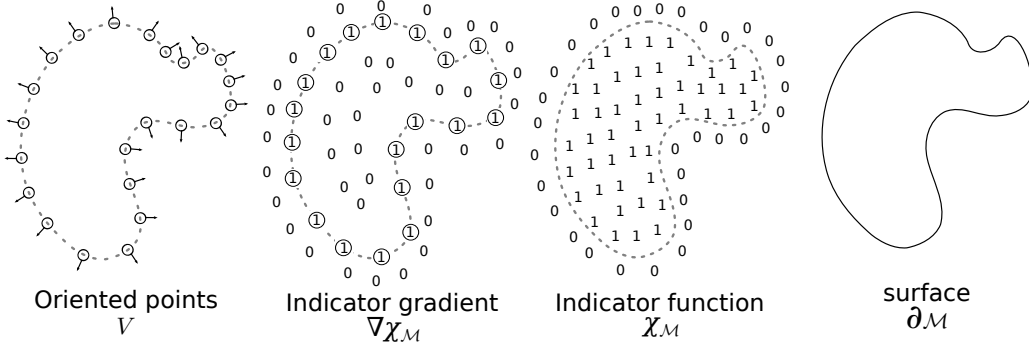
Figure 2: Four stages to reconstruct a surface: obtain a set of sample oriented points, each oriented point can be considered as a sample point of the gradient of some indicator function, the isosurface induced by the indicator function defines the reconstructed model.

Applying the divergence operator, this variational problem transforms into a standard Poisson problem: *Compute the scalar function $\tilde{\chi}$ whose Laplacian (divergence of gradient) equals the divergence of the vector field $\vec{V}$,*

$$\Delta\tilde{\chi} \equiv \nabla \cdot \nabla\tilde{\chi} = \nabla \cdot \tilde{V}.$$

In order to implement and solve this approach in a discrete domain, some problems must be considered [3].

2.3.1 Define the gradient field

Because the indicator function is a piecewise constant function, explicit computation of its gradient field would result in a vector field with unbounded values at the surface boundary. To avoid this, the indicator function is convolved with a smoothing filter and considers the gradient field of the smoothed function [3].

$$\nabla(\chi_M * \tilde{F})(q_0) = \int_{\partial M} \tilde{F}(q_0)\bar{N}_{\partial M}(p)dp$$

2.3.2 Approximating the gradient field

Using the point set S to partition ∂M into distinct patches $\wp_s \subset \partial M$, we can approximate the integral over a patch $\wp_s$ by the value at point sample $s.p$, scaled by the area of the patch:

$$\nabla(\chi_M * \tilde{F})(q) \approx \sum_{s \in S} |\wp_s|\tilde{F}_{s.p}(q)s.\vec{N} \equiv \vec{V}(q)$$

2.3.3 Solving the Poisson problem

Having formed a vector field $\vec{V}$, we want to solve the function $\tilde{\chi}$ so that $\nabla\tilde{\chi} = \vec{V}$. However, $\vec{V}$ is generally not integrable (there is not guarantee that is curl-free), so an exact solution does not generally exist. To find the best least-squares approximate solution, we apply the divergence operator to form the *Poisson equation*.

$$\Delta\tilde{\chi} = \nabla \cdot \vec{V}$$

2.4 Volume Meshing

Following the workflow shown in Figure 1, there are three type of volume meshing analyzed in this work and they vary in terms of element type employed: tetrahedra, hexahedra or mixed–elements, meaning a mix of tetrahedra, hexahedra, pyramids and prisms (wedges). In order to compare the different results generated by these meshing techniques, it is first necessary to explain how quality of volume elements is measured.

2.4.1 Quality of volume meshes

The Jacobian Ratio (JR) is one of the most used quality criterion in hexahedra meshes [8, 9, 10, 11, 12] and in order to understand how it works, it is necessary to understand some concepts first.

The FEM establish a bijective mapping function F between the modeled domain referential (x_1, x_2, x_3), in which the elements of the mesh are defined, and a referential parent system (ξ_1, ξ_2, ξ_3) for each type of element. The Jacobian matrix J of mapping F considered at parent frame point ξ is then defined as:

$$J(\xi) := \frac{\partial F}{\partial \xi}(\xi)$$

If for some reason, F is no longer bijective, a Finite Element Analysis (FEA) cannot be carried out over the mesh. In order to detect the bijection property of F, two values must be computed for each node. The first is the determinant of J, $|J(\xi)|$. The second is the Jacobian Ratio JR defined as:

$$JR_i = \frac{|J(\xi)|_i}{|J(\xi)|_{max}}$$

where $|J(\xi)|_{max}$ is the maximum value of $|J|$ among all element nodes. Note that JR is normalized to present values between $(-\infty, 1]$.

Element category	invalid	very bad	questionable	good
JR value	$(-\infty, 0.001]$	$(0.001, 0.0\overline{3}]$	$(0.0\overline{3}, 0.2]$	$(0.2, 1]$

Table 1: Quality categories for Jacobian Ratio

It is now possible to say that F is not longer bijective when $JR \leq 0$ for at least one of its nodes. Moreover, the quality of an element e can be defined as $JR_e := |J|_{min}/|J|_{max}$ and therefore, if any element presents a $JR_e \leq 0$, then not only is the element invalid, but the entire mesh is considered as not suitable for FEA.

The JR is not only used to detect invalid elements, but also poor quality elements. A perfect element is the one with a $JR = 1$. An element is said to present "questionable" quality when $JR \in (0, 0.2)$ after the Verdict library of mesh quality metrics[1] and some authors [9, 11]. On the other hand, the documentation of one of the most important commercial Finite Element Solver (FES), ANSYS[2], states that an element with a JR value[3] less than 0.001 presents so poor quality that it is considered as invalid. Moreover, an element with a JR value $\in (0.001, 0.0\overline{3}]$ is suitable for FEA although the solution from the FES will be inaccurate. Regarding both quality boundary definitions, Table 1 shows the different categories for element quality.

The JR criterion is widely used in hexahedra meshes and it can be easily extended to prisms and pyramids by the computation of the $|J|$ for those element types. Unfortunately it is not a good quality measure for tetrahedra as the value of $|J|$ is the same for each tetrahedron node no matter their position, in other words, the value of JR for a tetrahedron is always 1. Moreover, the JR does not allow the detection of sliver (or flat) tetrahedra, which is one of the most common problems for this type of element and it adds imprecision to the solution found by a FES.

On the other hand, there is a quality criterion used by several authors [13, 14, 15] that considers important geometric aspects of the tetrahedron, like its volume V and its edge lengths l_i:

$$A_\gamma := \frac{\left(\frac{1}{6}\sum_{i=1}^{6} l_i^2\right)^{3/2}}{8.47867 \times V}$$

As it compromises the volume, it helps to detect sliver tetrahedra. Moreover, if V is the signed volume, A_γ can detect element inversion. Finally, the use of the edge

---

[1]The verdict mesh verification library (2007) Sandia National Laboratories, `http://cubit.sandia.gov/verdict.html`

[2]http://www.ansys.com

[3]Note that ANSYS defines the Jacobian Ratio as JR^{-1} regarding the notation used here.

lengths will avoid to consider extremely sharp tetrahedra as "good" elements.

In order to consider a similar scale to JR, the final quality of a tetrahedron is defined as $Q_{tetra} = A_\gamma^{-1}$. With this definition Q_{tetra} is negative for any inverted element, bad when it's close to 0 and perfect when its value is 1.

The JR might also fail in detecting inverted pyramids. For this reason a signed and normalized version of the aspect ratio (AR) presented in [16] is used. In this case, a negative AR shows the presence of inverted elements.

2.4.2   Tetrahedral meshing using TetGen

To generate tetrahedral meshes, one of the most used softwares is TetGen (`http://tetgen.org`), developed by Hang Si. This program has implemented several meshing algorithms that allow to produce a tetrahedral mesh. Some of its capabilities are: to preserve the input surface mesh and constraint the mesh generation process in order to create quality elements, e.g., constraining the minimal dihedral angle. It is important to note that TetGen, as well as CGAL, is capable of improving the quality of a surface mesh. However, TetGen will use all the nodes of the input mesh and will add more (Steiner points) in order to build a constrained or conforming Delaunay triangulation. In contrast to TetGen, CGAL may or may not use all input nodes in the process of achieving a quality triangulation. As it will be show in the result section, CGAL tends to produce meshes with less triangles and nodes than the quality mesh generated by TetGen (in the context of surface meshes generated with the MC).

2.4.3   Hexahedral meshing

One of the most common techniques to produce a hexahedral mesh is the Octree [17]. The Octree allows regions of different refinement level to coexist in the same mesh. By the use of several templates containing only hexahedra, transitions are performed between coarse and refined regions [9, 10, 18].

The Cubit Tool Suite developed at Sandia National Laboratories (`http://cubit.sandia.gov/`) implements most of the state of the art algorithms in hexahedral meshing. Unfortunately, even for academic purposes, it is not easy to obtain a license (as in the case of TetGen).

2.4.4   Mixed–Element mesheing

Once a hexahedral mesh is produced, there are two options to achieve surface representation: (1) remove all hexahedra intersecting the boundary (meaning an

element with inside and outside nodes) and then fill the gap adding new hexahedra in a proper manner [9] or (2) project the outside nodes onto the surface.

A recent work [19] has proposed to replace boundary hexahedra by mixed–elements in order to improve the overall mesh quality. Once this replacement is done, project outside nodes onto the surface. This technique tends to produce less quality issues than using only hexahedral; although, none of these techniques can ensure mesh validity for a general case.

3   RESULTS AND DISCUSSIONS

The surface mesh generated for Breast simulation is shown in Figure 3. As mentioned in section 2.4.2, TetGen improves the quality of the input mesh; however, the final results have more triangles than the output generated by CGAL and may present flat triangles. Table 2 summarizes statistics for surface meshing in the context of the workflow shown in Figure 1. Finally, the CGAL mesh was contrained to present triangles with min angle above 30 degrees.
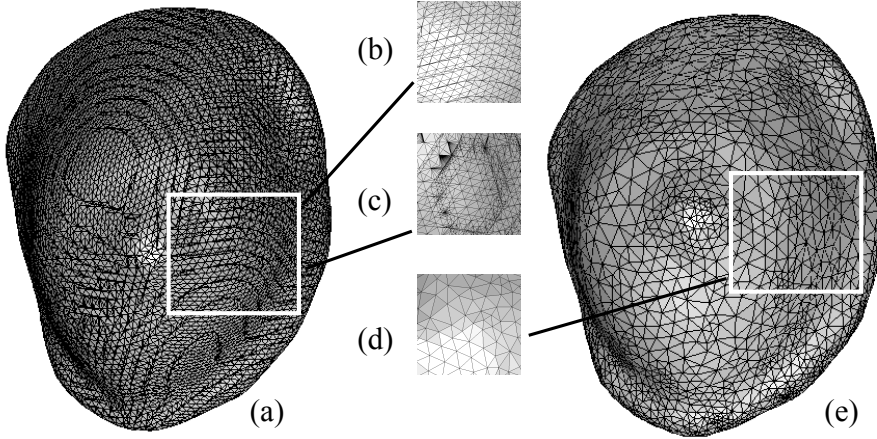
Figure 3: Breast surface meshing: (a) input data with the *MC*, (b) zoom to previous mesh, (c) zoom to the same region in a TetGen optimized mesh, (d) zoom to CGAL mesh and (e) the output mesh by CGAL

With respect to Volume meshing, the output of TetGen and the mentioned mixed–element technique are compared. Unfortunately, the Cubit output could not be compared as it was to difficult to obtain the software. Table 3 summarizes statistics for volume meshing. Finally, Figure 4 shows the results for volume meshing, where the overall mesh generated by TetGen is omitted because it is equal (in terms of surface) to the mesh generated by CGAL.

	MC	MC improved	TetGen	CGAL
nodes	12189	10104	50975	4058
triangles	24374	20204	101946	8112
min angle	0.008	0.0108	0	30
max angle	179.2	179.9	180	119.9

Table 2: Statistics for surface meshing
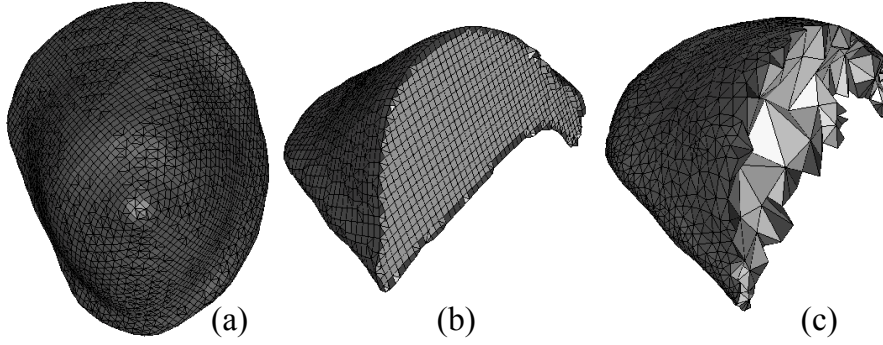
Figure 4: Breast volume meshing: (a) output by mixed–elements, (b) cut to see internal elements and (c) same cut to TetGen mesh (using the surface generated by CGAL).

4 CONCLUSIONS

In this paper, we have proposed and evaluated a workflow based of sophisticated existing tools to improve poor quality volumetric models, making the process simple enough for non–experts to use in a wide range of applications.

The major advantage of the proposed workflow is that, the ability to remove progressively poor quality elements allows to generate high quality surface meshes and subsequently an accurate volume mesh of the actual geometry, from a point–cloud acquired with conventional volumetric imaging methods.

Performance metrics over the polygonalization quality show that our workflow can successfully generate volumes free of *bad geometry problems*, with good quality triangles in many complicated surface meshes.

	Elements	Nodes	Aspect–ratio	Jacobian–ratio
TetGen	15680	4801	0.0623	–
Mixed–Elements	35748	27029	0.204	0.1894

Table 3: Statistics for volume meshing

ACKNOWLEDGES

REFERENCES

[1] Lorensen, W. and Cline, H. *Marching Cubes: A high resolution 3D surface construction algorithm.* Computer Graphics (1987) **21**(4): 163–169.

[2] Alliez, P., Saboret, L. and Guennebaud, G. *Surface Reconstruction from Point Sets.* In CGAL User and Reference Manual. CGAL Editorial Board, 4.1 edition, 2012 (http://www.cgal.org).

[3] Kazhdan, M., Bolitho, M. and Hoppe, H. *Poisson Surface Reconstruction.* In Symp. on Geometry Processing (2006): 61–70.

[4] Schaefer, S. and Warren, J. *Dual Marching Cubes: Primal Contouring of Dual Grids.* In: Proceedings of Pacific Graphics (2004): 70–76.

[5] Hu, Grossberg, Mageras *Survey of Recent Volumetric Medical Image Segmentation Techniques.* In book: Biomedical Engineering, InTech (2009): 321–346.

[6] Raman, S. and Wenger, R. *Quality Isosurface Mesh Generation Using an Extended Marching Cubes Lookup Table.* Computer Graphics Forum (2008) **27**(3): 791–798.

[7] ElGindy, H., Everett, H. and Toussaint, G. *Slicing an ear using prune-and-search.* Pattern Recognition Letters (1993) **14**(9): 719–722.

[8] Knupp, P. *Achieving finite element mesh quality via optimization of the jacobian matrix norm and associated quantities. part ii – a framework for volume mesh optimization and the condition number of the jacobian matrix.* International Journal for Numerical Methods in Engineering (2000) **48**: 1165–1185.

[9] Ito, Y., Shih, A. and Soni, B. *Octree–based reasonable-quality hexahedral mesh generation using a new set of refinement templates.* International Journal for Numerical Methods in Engineering (2009) **77**(13): 1809–1833.

[10] Zhang, H. and Zhao, G. *Adaptive hexahedral mesh generation based on local domain curvature and thickness using a modified grid–based method.* Finite Element Analysis and Desing (2007) **43**(9): 691–704.

[11] Shepherd, J. and Johnson C. *Hexahedral mesh generation for biomedical models in scirun.* Engineering with Computers (2009) **25**: 97–114.

[12] Kwok, W. and Chen Z. *A simple and effective mesh quality metric for hexahedral and wedge elements.* In: Proceedings of the 9th International Meshing Roundtable (2000): 325–333.

[13] Ferrant, M., Warfield, S., Nabavi, A., Jolesz, F. and Kikinis, R. *Registration of 3d intraoperative mr images of the brain using a finite element biomechanical model.* In: Proceedings of the Third International Conference on Medical Image Computing and Computer–Assisted Intervention (2000): 19–28.

[14] Berzins, M. *Mesh quality: A function of geometry, error estimates or both?* Engineering with Computers (1999) **15**(3): 236–247.

[15] Parthasarathy, V., Graichen, C. and Hathaway, A. *A comparison of tetrahedron quality measures.* Finite Elements in Analysis and Design (1993) **15**: 255–261.

[16] Lobos, C., Bucki, M., Hitschfeld, N. and Payan, Y. *Mixed-element mesh for an intra–operative modelling of the brain tumor extraction.* In: Proceedings of the 16th International Meshing Roundtable (2007): 387–404.

[17] Shephard, M. and Georges, M. *Automatic three-dimensional mesh generation by the finite octree technique.* International Journal for Numerical Methods in Engineering (1991) **32**:709–749.

[18] Schneiders, R. *Refining quadrilateral and hexahedral element meshes.* In: Proceedings of the Fifth International Conference on Numerical Grid Generation in Computational Field Simulations (1996): 679–688.

[19] Lobos, C. *A Set of Mixed–Elements Patterns for Domain Boundary Approximation in Hexahedral Meshes.* In: Proceedings of the 20th Medicine Meets Virtual Reality (2013): 268–272

ADAPTIVE LIMIT ANALYSIS USING DEVIATORIC FIELDS

ANDREI V. LYAMIN[*], KRISTIAN KRABBENHOFT[*] AND SCOTT W. SLOAN[*]

[*] Centre of Excellence for Geotechnical Science and Engineering (CGSE)
The University of Newcastle
e-mail: andrei.lyamin@newcastle.edu.au,   www.cgse.edu.au

Key words: Limit Analysis, Finite Elements, Error Estimation, Adaptive Remeshing

Abstract. Accurate estimates of limit loads for difficult stability problems in geotechnical engineering can rarely be obtained from a single finite element limit analysis without using an excessive number of elements. Therefore, efficient adaptive strategies which maximize the solution accuracy using minimum number of elements in the mesh are of great interest. This study explores the possibility of using the internal dissipation calculated from deviatoric stresses and strain rates as suitable control field for purely frictional materials. The performance observed for considered set of problematic for other adaptive schemes geotechnical examples is very promising. Moreover, the proposed approach works very well also for cohesive and cohesive frictional materials, suggesting its use as general engine for adaptive mesh refinement.

1 INTRODUCTION

For complex, practical stability problems in geotechnical engineering, accurate estimates of the collapse load or factor of safety can rarely be obtained from a single analysis and a trial and error process is usually required. The key to obtaining accurate solutions lies in accurately capturing the areas of plasticity within the problem domain, as their pattern and intensity govern the solution. The development of an efficient mesh adaptivity strategy, which is able to pinpoint the fine detail of a structure's collapse mechanism, is thus of the highest priority in modern limit and shakedown analysis.

A critical aspect of any adaptive meshing process is the estimation of the discretisation error present in a given finite element solution. Since *a priori* error estimates play only an indicative role (Borges *et al.*[1]), useful error estimates must employ *a posteriori* techniques to predict the overall discretisation error in one or more solution norms (or control variables). Generally speaking, two major approaches have been practiced so far. The first is the Hessian based error estimation, where the spatial distribution of the error in solution is obtained on the basis of information gathered from the matrix of second derivatives of some control variable (Zienkiewicz *et al.*[2], Almeida *et al.*[3], Lyamin *et al.*[4]). And the second is a so-called *gap* adaptivity scheme, which is based on the fact that for limit analysis applications the global error in the solution can be readily obtained as the sum of elemental differences between upper and lower bound estimates (Ciria *et al.*[5], Muñoz *et al.* [6]).

The major advantage of Hessian based error estimation when combined with optimal-mesh-adaptive scheme is that it usually provides the element size distribution which converges (keeping the number of elements in the mesh constant) very quickly to a steady, smoothly graded mesh pattern, which can be either isotropic or anisotropic. It is very general and based on the fact that, at some point $\mathbf{x}$ in the vicinity of a point $\mathbf{x}_0$, the difference between the variable of interest $\mathbf{u}$ and its discrete approximation $\mathbf{u}_h$ can be estimated using the following expression

$$\left\| \mathbf{u} - \mathbf{u}_h \right\| \simeq C \left\| (\mathbf{x} - \mathbf{x}_0)^T \mathbf{H}_R(\mathbf{u}_h(\mathbf{x}_0))(\mathbf{x} - \mathbf{x}_0) \right\| \tag{1}$$

where C is a positive constant and $\mathbf{H}_R(\mathbf{u}_h(\mathbf{x}_0))$ denotes a recovered Hessian matrix. An anisotropic error estimator for element e of a partition $\mathcal{T}_h$ of the domain Ω can then be introduced as

$$\eta_e = n\Omega_e^{1/2} \left| \lambda_n(\mathbf{x}_0) \right| h_n^2 \geq \left\{ \int_{\Omega_e} \left((\mathbf{x} - \mathbf{x}_0)^T \mathbf{H}_R(\mathbf{u}_h(\mathbf{x}_0))(\mathbf{x} - \mathbf{x}_0) \right)^2 \mathrm{d}\Omega \right\}^{1/2}; \qquad \eta = \sum_e \eta_e \tag{2}$$

where n is the problem dimensionality, h_n is the minimum dimension of element e, and λ_n is the largest eigenvalue of the element Hessian matrix. It is assumed also that the estimated error yields the same value in any direction, i.e. $\left| \lambda_1 \right| h_1^2 = \left| \lambda_2 \right| h_2^2 = \ldots = \left| \lambda_n \right| h_n^2$.

The choice of a suitable control variable is not obvious for plasticity problems. Several approaches have been practiced so far including those based on power dissipation or its gap (Ciria et al.[5], Muñoz et al. [6]), plastic multipliers (Lyamin et al.[4]) and strain rate (Christiansen & Pedersen[7]) fields employed as control variables. All these schemes work quite well for cohesive or cohesive-frictional materials, but for purely frictional soils their performance stalls as e.g. plastic multipliers have substantially high values for all zero stress points on the surface of soil domain, therefore cannot indicate reliably plastic areas. Similar conclusion can be made about performance of schemes based on power dissipation or strain rates. This study explores the possibility of using the internal dissipation calculated from deviatoric stresses and strain rates (called also "shear power" in the rest)

$$u = \int_{\Omega_e} \mathbf{s} : \dot{\boldsymbol{\varepsilon}}' \mathrm{d}\Omega, \quad s_{ij} = \sigma_{ij} - \tfrac{1}{3} I_1^\sigma \delta_{ij}, \quad \dot{\varepsilon}_{ij}' = \dot{\varepsilon}_{ij} - \tfrac{1}{3} I_1^{\dot{\varepsilon}} \delta_{ij} \tag{3}$$

as suitable control field for purely frictional materials. In above σ_{ij}, s_{ij} and $\dot{\varepsilon}_{ij}$, $\dot{\varepsilon}_{ij}'$ are the Cartesian and deviatoric stresses and strain rates, respectively, and I_1^σ, $I_1^{\dot{\varepsilon}}$ are the first invariants for stresses and strain rates.

The performance observed for considered set of problematic for other adaptive schemes geotechnical examples is very promising. Moreover, the proposed approach works very well also for cohesive and cohesive frictional materials, suggesting its employment as general engine for adaptive mesh refinement.

2 THE OPTIMAL MESH ADAPTIVE SCHEME

Usually mesh refinement proceeds with gradual adjustment of the element size aiming to distribute local error uniformly over the problem domain. The other alternative is to obtain the

element size distribution which minimizes the global error given by equation (2). This approach is known as optimal-mesh-adaptive technique and is described in detail e.g. by Almeida *et al.*[3] In brief, the optimal-mesh-adaptive procedure can be cast as constrained optimization problem, which for two-dimensional case becomes

$$\text{minimise} \quad \mathcal{F}(h_{2T}) = \{\eta(h_{2T})\}^p = \sum_{T \in \mathcal{T}_k} \Omega_T \left| 2\lambda_{2T} \right|^p h_{2T}^{2p}$$

$$\text{subject to} \quad N_e = (4/\sqrt{3}) \sum_{T \in \mathcal{T}_k} \Omega_T / (s_T h_{2T}^2) \quad \text{to find } h_{2T},\, T \in \mathcal{T}_k \tag{4}$$

where h_{2T} and s_T are the new size and the stretching of element T, $\mathcal{T}_k$ is the finite element discretization at the adaptation step k and N_e is the desired number of elements at the step $k+1$. For $p = 2$ and the case of equilateral elements (no stretching) the solution to problem (4) is given by

$$h_{2T} = \sqrt{\left(4\Omega_e / \left(\sqrt{3} N_e \eta_e\right)\right)\eta} \tag{5}$$

The advancing front algorithm (Peraire *et al.*[8]) has been employed for generating the mesh. As the meshing time is only a small fraction of the total CPU time in adaptive limit analysis, this algorithm was chosen in order to give full control of the mesh quality, including the shape of the elements and the rate of change of the element size throughout the mesh from one iteration to the next. Both refinement and coarsening of the mesh have been allowed.

3   LIMIT ANALYSIS

The lower (LB) and upper (UB) bound limit analysis formulations used in this investigation stem from the methods originally developed by Sloan[9][10], but have evolved significantly over the past two decades to incorporate the major improvements described in Lyamin and Sloan[11][12] and Krabbenhoft et al.[13][14] . Key features of the methods include the use of linear finite elements to model the stress/velocity fields, and collapsed solid elements at all inter-element boundaries to simulate stress/velocity discontinuities. The solutions from the lower bound formulation yield statically admissible stress fields, while those from the upper bound formulation furnish kinematically admissible velocity fields. This ensures that the solutions preserve the important bounding properties of the limit theorems.

Both formulations result in convex mathematical programs, which (considering the dual form of upper bound problem) can be cast in the following form:

$$\text{maximize} \quad \lambda$$

$$\text{subject to} \quad \mathbf{A\sigma} = \mathbf{p}_0 + \lambda\mathbf{p} \tag{6}$$

$$f_i(\mathbf{\sigma}) \leq 0, \quad i = \{1,\ldots,N\}$$

where λ is a load multiplier, $\mathbf{\sigma}$ is a vector of stress variables, $\mathbf{A}$ is a matrix of equality constraint coefficients, $\mathbf{p}_0$ and $\mathbf{p}$ are vectors of prescribed and optimizable forces, respectively, f_i is the yield function for stress set i and N is the number of stress nodes. The

solutions to problem (6) can be found efficiently by using general Interior-Point methods (IPM) or specialised conic optimization solvers (SOCP).

4   NUMERICAL EXAMPLES

Two representative examples from the soil mechanics are considered in this section to illustrate the efficiency of proposed adaptive approach. First example, so-called N_γ problem, is about estimating the bearing capacity of rigid footing resting on cohesionless soil (sand). Second example is known as "passive earth pressure" case. Here the maximum lateral pressure which can be exerted to the soil cut, before it collapses upwards, needs to be found. Both examples are treated as two-dimensional problems and considered under plain strain conditions.

Adaptive refinement proceeds by specifying the initial and target number of elements in the mesh, and the number of adaptive iterations. If this target number of elements is reached before the maximum number of iterations has exceeded, no additional elements are injected. However, some improvement can still be achieved by redistributing the element sizes in the remaining iterations if a better pattern of the control variable can be found. In examples considered the thresholds on mesh refinement and coarsening factors between 2 iterates were set to 0.25 and 1.5, respectively.

4.1 Rigid rough strip footing on cohesionless soil (N_γ problem)

For a rigid strip footing resting on a ponderable purely frictional soil with no surcharge the bearing capacity is usually estimated by using reduced Terzaghi[15] equation of the form

$$q = 0.5\gamma B N_\gamma \qquad (7)$$

where γ is soil unit weight, B is the width of the footing and N_γ is the bearing capacity factor, which depends on soil friction angle, ϕ. There is no exact solution available for N_γ and over the years several empirical expressions were suggested and used in practice (Brinch Hansen[16], Caquot & Kerisel[17]). Recently very accurate estimates done by numerical limit analysis were reported (Hjiaj et al.[18]) and eventually quasi-exact values of N_γ were obtained by the method of characteristics (Martin[19]). Therefore, besides the standard for limit analysis UB-LB gap error estimation, this allows direct check of the accuracy of adaptively obtained
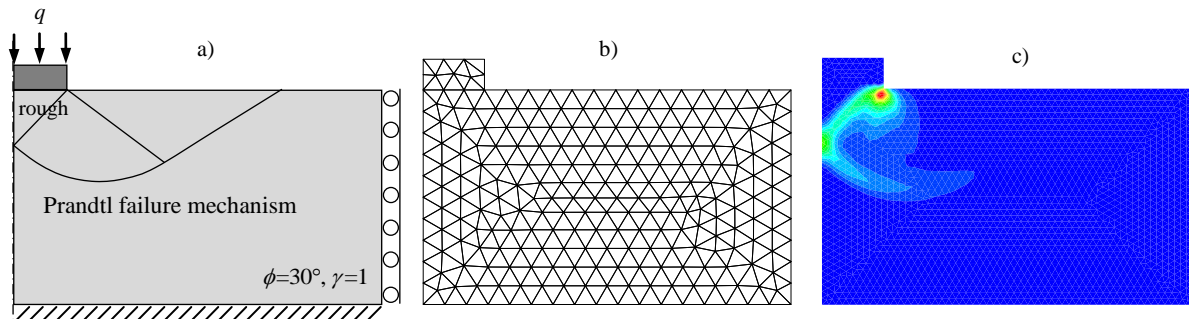

Figure 1. Geometry (a), initial mesh (b) and shear power dissipation plot (c) for strip footing.

solutions for this problem.

The problem description (including Prandtl[20] failure mechanism) together with the initial mesh used for analysis and corresponding shear power dissipation is given in Figure 1. Next, in Figure 2 the distributions of several traditionally used for adaptive limit analysis control variables are plotted. Due to the absence of cohesion in soil mass it is evident that power dissipation (all zeros) is not an alternative to govern refinement procedure in this case. And

a) power dissipation (UB)   b) rate of work by gravity (UB)   c) strain rate (LB)   d) plastic multiplier (LB)
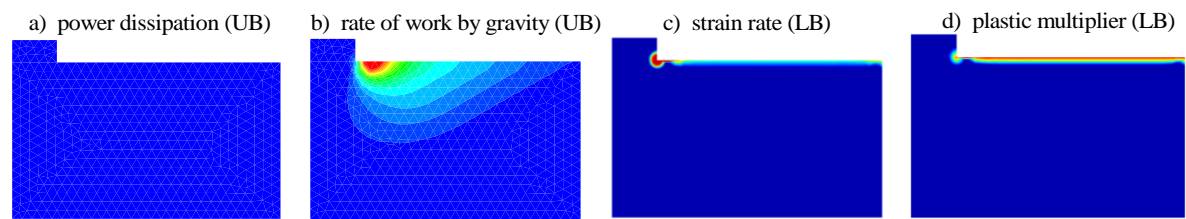
Figure 2. Distributions of commonly used control variables in the case of N_γ problem.

even if power loss due to soil unit weight is taken into account (Figure 2b) the resultant distribution does not resemble the actual collapse mechanism (slip line) to be considered as a good choice. Neither it will work when UB-LB gap of elemental power dissipation would be used. Similar comments are applied to another pair of control variables, strain rate and plastic multiplier fields. It is clear that all zero-stress points (soil surface boundary, LB case) are at plastic state, therefore will have some non-zero plastic multipliers as shown in (Figure 2d). This "noise" prevents plastic multipliers to be employed as adaptivity guide either. On the other hand, the dissipation computed using deviatoric terms of stresses and strains (shear power) has very distinctive distribution resembling classical Prandtl[20] collapse mechanism for strip footing. And, as can be judged from results presented in Figure 3c, it works efficiently for both lower and upper bound discretizations. The final mesh and corresponding shear power dissipation are illustrated in Figure 3a,b.
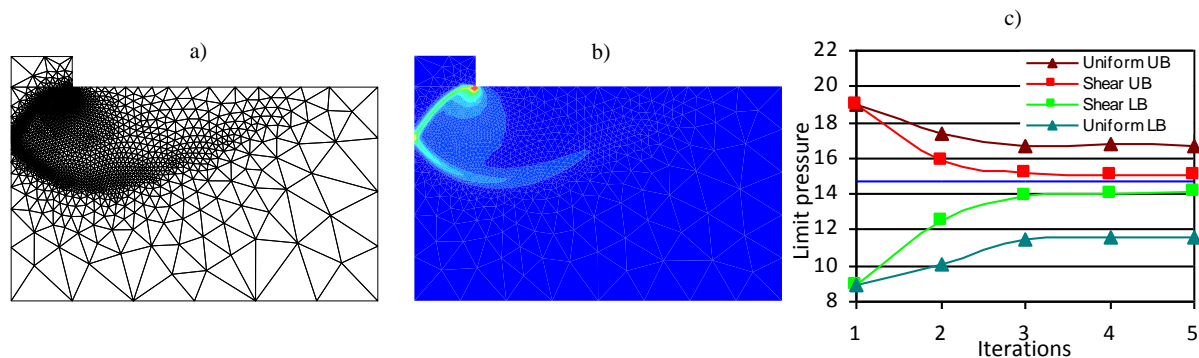
Figure 3. Final mesh (a), shear power dissipation (b) and convergence diagram for strip footing.

4.2 Passive earth pressure

This is another classical problem in soil mechanics, where the lateral pressure, p, is applied to the soil mass to cause its collapse, as shown in Figure 4. There are several theories for this problem (the most famous are due to Coulomb[21] and Rankine[22]) with different analytical solutions accounting for various soil slope angles, soil/wall interface conditions, mode of
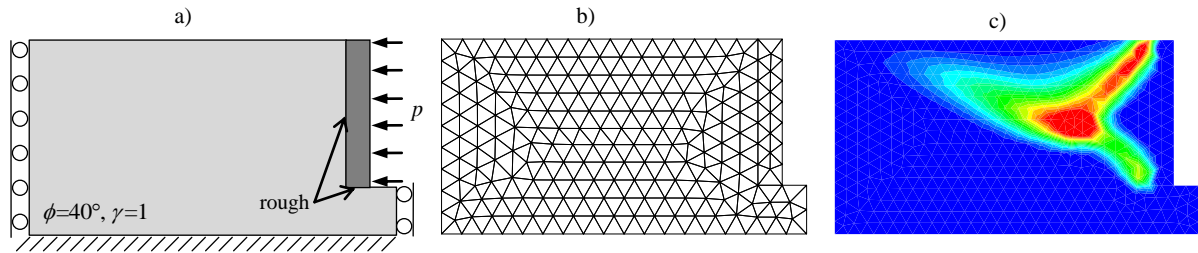
Figure 4. Geometry (a), initial mesh (b) and shear power dissipation plot (c) for passive earth pressure.

failure (no rotation or rotation allowed), etc. But our main focus here is not actually to compare results obtained to existing solutions, rather demonstrate that proposed mesh
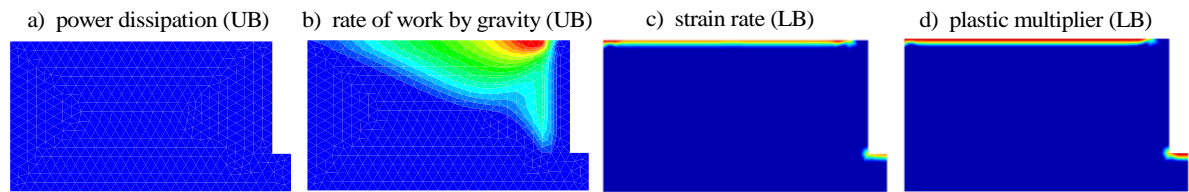
Figure 5. Distributions of commonly used control variables in the case of passive earth pressure problem.

refinement approach performs reliably when applied to sands. For this purpose, in the same way as for N_γ case, the distributions of most popular control variables traditionally used within the limit analysis adaptive schemes are given in Figure 5. It appears that the same comments as those given in previous section are applicable here as well - none of the distributions in Figure 5 seems to be suitable to assist with effective mesh refinement. On the other hand, using proposed adaptive scheme based on shear power dissipation results in robust refinement procedure as presented in Figure 6.
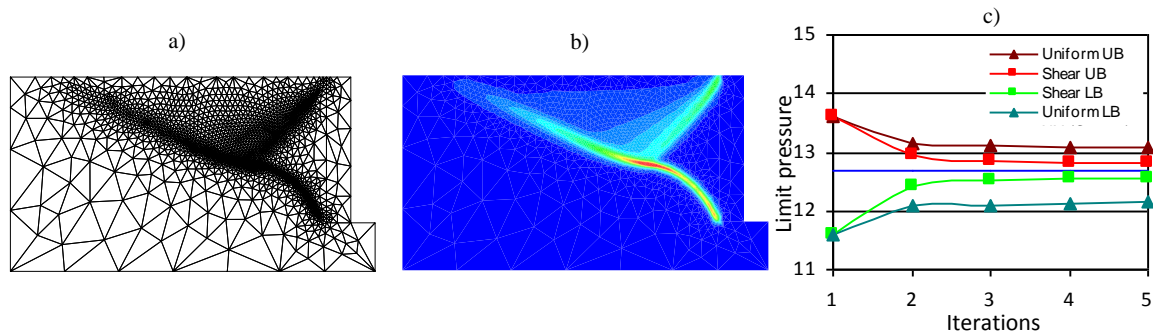
Figure 6. Final mesh (a), shear power dissipation (b) and convergence plot (c) for passive earth pressure problem.

5 CONCLUSIONS

Based on deviatoric stress and strain fields elemental power dissipation was employed to control mesh refinement process in limit analysis computations for purely frictional materials. Both lower and upper bound counterparts of limit analysis were tested. The obtained results show that the proposed approach works reliably for demanding applications, where traditionally used control variables fail to perform.

REFERENCES

[1] Borges, L.A., Zouain, N., Costa, C. and Feijóo, R. An adaptive approach to limit analysis, *International Journal of Solids and Structures,* (2001) **38**: 1707-1720.

[2] Zienkiewicz, O.C., Huang, M. and Pastor, M. Localization problems in plasticity using finite elements with adaptive remeshing, *International Journal for Numerical and Analytical Methods in Geomechanics*, (1995) **19**: 127-148.

[3] Almeida, R.C., Feijóo, R., Gleão, A.C., Padra, C. and Silva, R.S. Adaptive finite element computational fluid dynamics using an anisotropic error estimator, *Computer Methods in Applied Mechanics and Engineering,* (2000) **182**: 379-400.

[4] Lyamin AV, Sloan SW, Krabbenhoft K and Hjiaj M Lower bound limit analysis with adaptive re-meshing, *International Journal for Numerical Methods in Engineering* (2005) **63**(14): 1961-1974.

[5] Ciria H, Peraire J and Bonet J. Mesh adaptive computation of upper and lower bounds in limit analysis. *Int. J. Numer. Meth. Engng.* (2008) **75**: 899–944.

[6] Muñoz JJ, Bonet J, Huerta A, Peraire J. Upper and lower bounds in limit analysis: Adaptive meshing strategies and discontinuous loading. *Int. J. Numer. Meth. Engng.* (2009) **77**: 471-501.

[7] Christiansen, E. and Pedersen, O.S. Adaptive mesh refinement in limit analysis, *International Journal for Numerical Methods in Engineering,* (2001) **50**: 1331-1346.

[8] Peraire, J., Vahdati, M., Morgan, K. and Zienkiewicz, O.C. Adaptive remeshing for compressible flow computations, *Journal of Computational Physics,* (1987) **72**: 449-466,

[9] Sloan, S.W. Lower bound limit analysis using finite elements and linear programming. *International Journal for Numerical and Analytical Methods in Geomechanics*, (1988) **12**(1): 61–77.

[10] Sloan, S.W. Upper bound limit analysis using finite elements and linear programming. *International Journal for Numerical and Analytical Methods in Geomechanics*, (1989) **13**(3): 263–282.

[11] Lyamin, A.V., and Sloan, S.W. Lower bound limit analysis using nonlinear programming. *International Journal for Numerical Methods in Engineering*, (2002) **55**(5): 573–611.

[12] Lyamin, A.V., and Sloan, S.W. Upper bound limit analysis using linear finite elements and nonlinear programming. *International Journal for Numerical and Analytical Methods in Geomechanics*, (2002) **26**(2): 181–216.

[13] Krabbenhoft, K., Lyamin, A.V., Hjiaj, M., and Sloan, S.W. A new discontinuous upper

bound limit analysis formulation. *International Journal for Numerical Methods in Engineering*, (2005) **63**(7): 1069–1088.

[14] Krabbenhoft, K., Lyamin, A.V., and Sloan, S.W. Formulation and solution of some plasticity problems as conic programs. *International Journal of Solids and Structures*, (2007) **44**(5): 1533–1549.

[15] Terzaghi, K. Theoretical Soil Mechanics. John Wiley & Sons, NewYork (1943).

[16] Brinch Hansen, J. A Revised and Extended Formula for Bearing Capacity. The Danish Geotechnical Institute, (1970) Bulletin No. 28.

[17] Caquot, A. and Kerisel, J. Sur le terme de surface dans le calcul des fondations en milieu pulverent. *Proc. 3rd ICSMFE*, Zurich, (1953) **1**: 336-337.

[18] Hjiaj, M., Lyamin, A.V., Sloan, S.W. Numerical limit analysis solutions for the bearing capacity factor N_γ. *International Journal of Solids and Structures,* (2005) **42**(5-6) :1681-1704.

[19] Martin, C.M. Exact bearing capacity calculations using the method of characteristics. *Proc. 11th Int. Conf. of IACMAG*, Turin, (2005) Vol. 4, pp 441-450.

[20] Prandtl, L. Über die Härte plastischer Körper. Göttingen Nachr, *Math. Phys*. K1, **12**, 74-85, (1920).

[21] Coulomb C.A. Essai sur une application des regles des maximis et minimis a quelques problemes de statique relatifs a l'architecture. *Memoires de l'Academie Royale pres Divers Savants*, (1776) **7**.

[22] Rankine, W. On the stability of loose earth. *Philosophical Transactions of the Royal Society of London*, (1857) **147**.

FE ADAPTIVE ANALYSIS OF MULTI-REGIONS MODELS

LUIZ F. MARTHA[*]**, JOAQUIM B. CAVALCANTE-NETO**[†]**, WILLIAM M. LIRA**[‡]**, ANTONIO O. MIRANDA**[††]**, AND RAFAEL A. SOUSA**[*]

[*] Departament of Civil Engineering
Pontifical Catholic University of Rio de Janeiro (PUC-Rio)
Rio de Janeiro - RJ, Brazil
e-mail: lfm@tecgraf.puc-rio.br, rfldesousa@hotmail.com

[†] Department of Computing
Federal University of Ceara
Fortaleza - CE - Brazil
email: joaquim@lia.ufc.br

[‡] Laboratory of Scientific Computing and Visualization (LCCV)
Technology Center, Federal University of Alagoas
Maceio - AL - Brazil
email: william@lccv.ufal.br

[††] Department of Civil Engineering
University of Brasilia (UnB)
Brasilia - DF - Brazil
email: acmiranda@unb.br

Key words: 3D Adaptive Analysis, Finite Element Method, Mesh Generation, Multi-regions.

Abstract. This work presents a methodology for adaptive generation of 3D finite element meshes using geometric modeling with multi-regions and parametric surfaces, considering a geometric model described by curves, surfaces, and volumes. The adaptive strategy adopted in this methodology is based on independent refinements of these entities. From an initial model, new sizes of elements obtained from numerical error analysis and from geometric restrictions are stored in a global background structure, a recursive spatial composition represented by an octree. Based on this background structure, the model curves are initially refined using a binary partition algorithm. The discretization of curves is then used as input for the refinement of adjacent surfaces. The surface discretization also employs the background octree-based refinement, which is coupled to an advancing front technique for the generation of an unstructured triangulation. Surface meshes are finally used as input for the refinement of adjacent volumetric domains. In all stages of the adaptive strategy, the refinement of curves, surface meshes, and solid meshes is based on estimated numerical errors associated with the mesh of the previous step in the adaptive process. In addition, curve and surface refinement takes into account curvature information. An example is presented in order to validate the methodology proposed in this work.

1 INTRODUCTION

In numerical simulations using the Finite Element Method (FEM), two important aspects to be considered are the automatic generation of the model's finite element mesh and the definition of the level of refinement associated with this mesh. For the first aspect, there are a variety of algorithms with different techniques to generate planar, surface, and volumetric meshes. The second aspect, i.e., the level of refinement, is usually defined manually by a specialist based on his/her own experience. However, this refinement should consider the fact that the density of the generated elements varies according to the gradient of the obtained solution, which is initially unknown. In this context, this work presents a methodology to automate the refinement process of three-dimensional meshes based on adaptive technique procedures. In this work, the proposed methodology is applied to stress analysis of solid structures using a displacement-based finite element formulation. However, since the methodology essentially treats geometric modeling and mesh generation aspects of the problem, it could be used in other types of 3D finite element simulation.

Adaptive procedures try to automatically refine and coarsen a mesh, relocate its nodes, or adjust its cells to improve response accuracy. Usually, the computation begins with a trial solution obtained from a coarse mesh. The discretization error of this solution is estimated. If it fails to satisfy a prescribed accuracy metric, adjustments are made to achieve the desired solution with minimal effort. Common procedures are [1,2]: local/global refinement and/or coarsening of a mesh (h-refinement), relocating or moving a mesh (r-refinement), and locally varying the polynomial degree of elements (p-refinement).

Some strategies have been proposed to efficiently automate the 3D mesh refinement process. These strategies can be divided in two approaches: local and global refinement. In local refinement, the process uses an initial mesh and locally, using a set of elements, refines or coarsens elements in the mesh. Most works in the literature are based on this approach, as described ahead. Kallinderis and Vijayant [3] and Muthukrishnan [4] present an adaptive grid scheme based on the division/deletion of tetrahedral cells. Golias and Tsiboukis [5] and Golias and Dutton [6] employ a set of topological Delaunay transformations of tetrahedral elements and a technique for node reposition. Lee and Lo [7, 8] approach mesh refinement by inserting additional nodes at the midpoint of the longest or quasi-longest line segment of the mesh that bisects the original edges to generate new elements. In Merrouche [9], the mesh adaptation is achieved by a 3D bisection method. De Cougny and Shephard [10] present an adaptive scheme based on subdivision patterns (for refinement), edge collapsing (for coarsening), and mesh optimization (following refinement and coarsening). Lee et al. [9] only increase the order and density of 3D finite element meshes. Lee and Xu [11] generate a surface mesh for the mid-surface of the thin-walled structure, controlling element size, and convert the surface mesh to a 3D solid mesh by extrusion. More recently, Zhang et al. [12, 13] generates tetrahedral and hexahedral meshing in multi-material domains using grid-based method that employs an octree structure, refining meshes also locally.

In global refinement, on the other hand, at each refinement step, the entire mesh is deleted and another is generated based on new sizes of elements obtained from a discretization error estimation analysis. This process is used by Kettil et al. [14] only in regions with complicated parts. Hughes et al. [15] refine the structured meshes with NURBS surfaces. Our work presents a methodology that employs the global h-refinement approach.

A previous paper [16] proposed a two-dimensional self-adaptive strategy that was able to perform simulations involving automatic generation of meshes and adaptive methods. Other works have considered the same problem through different approaches, such as the study by Mark Shephard's team [17], from the Rensselaer Polytechnic Institute. Cavalcante-Neto [18] proposed a technique for the generation of volumetric meshes of tetrahedral elements for arbitrary region domains. Combining this technique with the implementation of 3D error estimators, the authors defined a prototype of an environment for adaptive generation in three dimensions. However, in this previous work, the complete process was not performed automatically, i.e., the mesh was generated independently from the error estimation and had to be manually combined. Moreover, despite treating multi-regions, it was not very efficient, and parametric surfaces, which are used in several types of simulations, were not considered.

This paper aims to present a methodology for adaptive generation of three-dimensional finite element meshes, using geometric modeling with multi-regions and parametric surfaces. Basically, the whole mesh adaptive process involves three steps: (1) analysis of a finite element (FE) model with discretization error estimation; (2) construction of a background structure to store new FE sizes that take into account the estimated discretization error and curve and surface curvature; and (3) hierarchical refinement of a FE model that is represented geometrically by curves, surfaces, and volume regions. This process may be repeated until a desired maximum allowed error metric is achieved. The methodology described herein covers only the last two steps, since discretization error estimation can be computed through different processes [1]. In this work, discretization error estimation is based on a standard technique used in the literature [19], in which the error is evaluated through the difference between stress field computed using conventional FE procedures and stress field obtained by means of more accurate recovery procedures (e.g. ZZ, SPR, or REP) [19-22].

The paper is organized as follows. Next section explains the proposed adaptive refinement strategy. Section 3 describes all the steps required to generate the background data structure that is used to define FE sizes in the adaptive process. The following section presents the hierarchical refinement of curves, surfaces, and volumes. An example of adaptive refinement is presented in Section 5. Finally, in Section 6, there is a conclusion.

2 ADAPTIVE REFINEMENT STRATEGY

The three-dimensional geometric model has a topological description of the vertices, curves, surfaces, and regions, as well as an associated geometric description, which consists of the coordinates of the vertices and the mathematical representation of the curves and surfaces. The geometric model can contain many regions. In this environment, the attributes of the simulation, such as the properties of the materials, loads and restrictions, are associated with the geometric entities. In this framework, the entities of finite element mesh (nodes and elements) automatically receive the attributes of the geometric entities that are related to. Using this approach, it is possible to create new meshes without losing the attributes.

Figure 1 illustrates the automatic adaptive strategy of the proposed refinement process. The input data are the initial volumetric mesh of problem in question and the geometric entities (curves and surfaces), and as well as their associated attributes. Initially, this mesh is numerically analyzed, the information required to initiate the adaptive procedure. Such information basically consists of numerical discretization errors associated with each

volumetric element of the mesh. From these errors, the need for adaptive refinement is verified. If the results converge, the adaptive process is concluded with a final discretization.
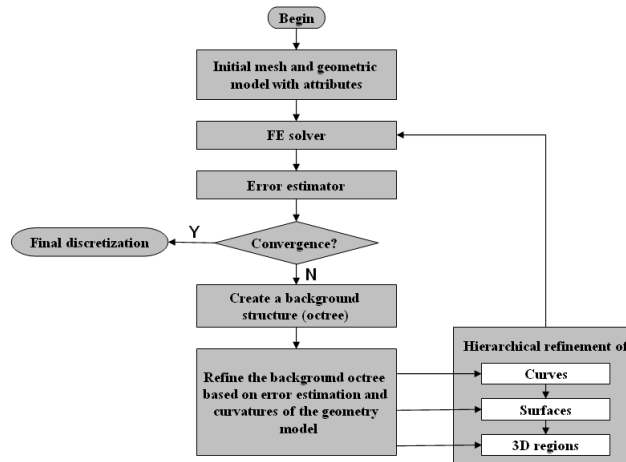

Figure 1: The proposed adaptive refinement process.

If convergence is not reached, the sizes of the new elements are computed based on the estimated discretization error. All the resized data are stored in an auxiliary background structure. Although many background structures are published in the literature, as reviewed by Quadros [23], the present work uses a background octree structure, which has the advantage of not only allowing fast search procedures down to internal leaves but also representing the desired size of the elements defined by the size of the internal leaves. For these reasons, an octree is used to support the discretization of curves, surface meshes, and the volumetric mesh.

In addition to discretization error estimation, curve and surface discretization is also required, especially when the curves and surfaces present high curvatures. In such locations, the meshes should be locally refined. Therefore, new element size data, based on the geometric information of the curves and surfaces, are computed and stored in the background octree. After this procedure, the background octree is internally finalized to provide a better transition between regions with elements of highly varying sizes.

Using the size information from the background octree, the next step consists of a three level hierarchical approach to create a new volumetric mesh. First, the curves are refined based on the size of the elements stored in the octree structure. This refinement subdivides the curves into segments with sizes consistent with those of the discretization error analysis and geometric criteria. After refining the curves, the meshes associated with each of the model's surfaces are discretized using an advancing front scheme in parametric space. This meshing scheme starts by subdividing curves on the boundary of each surface. Geometric curvature information is considered in surface refinement because the background octree takes this information into account. The last stage of the adaptive refinement process is related to the discretization of the domains of the model's regions. Such discretization uses a 3D advancing front technique that starts from the triangulated meshes associated with the boundary surfaces of each 3D region, also considering the sizes of the elements provided by the background octree. As can be seen, this adaptive meshing methodology supports multi-regions in a

consistent manner, considering curve and surface curvature information in addition to the estimated discretization error. Finally, a new discretization error analysis is performed to assess the quality of the results. If convergence is not obtained, the whole adaptive process is repeated as described above. The next sections summarize the proposed refinement strategy.

3 THE BACKGROUND OCTREE

An octree is a tree data structure based on a cell with eight children. Each cell of an octree represents a cube in the physical space. Each child represents one octant of its parent. On the leaves of the tree are the computational cells of the grid. In this work, the background octree has two main objectives. The first is to develop local guidelines used to define the discretization of curves and surfaces. The second is to define the sizes of tetrahedral elements to be generated during the advancing front procedure. The octree generation includes four steps. Figure 2 depicts the external appearance of the background octree of a hypothetical model (Figure 2-a). In the first step (Figure 2-b), the octree is initialized based on the input mesh data, which are the new element sizes obtained in the discretization error analysis. The second step (Figure 2-c) refines the octree based on the geometric curvatures of curves. In the third step (Figure 2-d), the octree is refined based on the geometric curvatures of surfaces. Finally, in the last step (Figure 2-e), the octree is refined in order to obtain a better transition between the sizes of the elements generated in the advancing front surface refinement.

3.1 Background octree refinement based on error analysis

Initially, a bounding cube is created based on the maximum range of the three Cartesian coordinates of the input model. This cube is the octree's root cell. In the first step of the octree refinement (Figure 2-b), each discretization error result of an element is used to determine the local depth of the subdivision. A characteristic size of each element is calculated for the estimated error and the octree cell containing the element's central point is determined. If the size of the cell edge is larger than the calculated characteristic size, then this cell is subdivided into eight smaller cells. This process is repeated recursively and finishes when the size of the cell is smaller than the given size. This process is repeated for every element of the current FE mesh. The characteristic size is calculated considering an equilateral tetrahedron with the same volume of each element.

The background octree works as a density function to guide the adaptive process. It could be replaced by other functions. It has the advantage of also allowing fast search procedures down to internal leaves. It could also have a different orientation to better adapt to models that are not parallel to Cartesian coordinates. However, the bounding cube parallel to the Cartesian coordinates is easier and faster to implement and usually gives very good results.

3.2 Octree refinement based on curve curvature

In some cases, when only the discretization error is considered in the adaptive process the new generated mesh (in the following step of the process) does not respect the actual geometry of the model's curves. This behavior occurs when parts of a curve, for example, present high curvatures in a region where the discretization error is low. In these situations, it is necessary to refine the background octree based on the curvatures of the curves to preserve the original geometric characteristics of the model.

460

The methodology used to refine the curves of the model based on their curvatures is a one-dimensional version of the procedure applied to discretize the background octree. The refinement of each curve employs a recursive spatial numbering technique similar to a binary tree data structure [24].
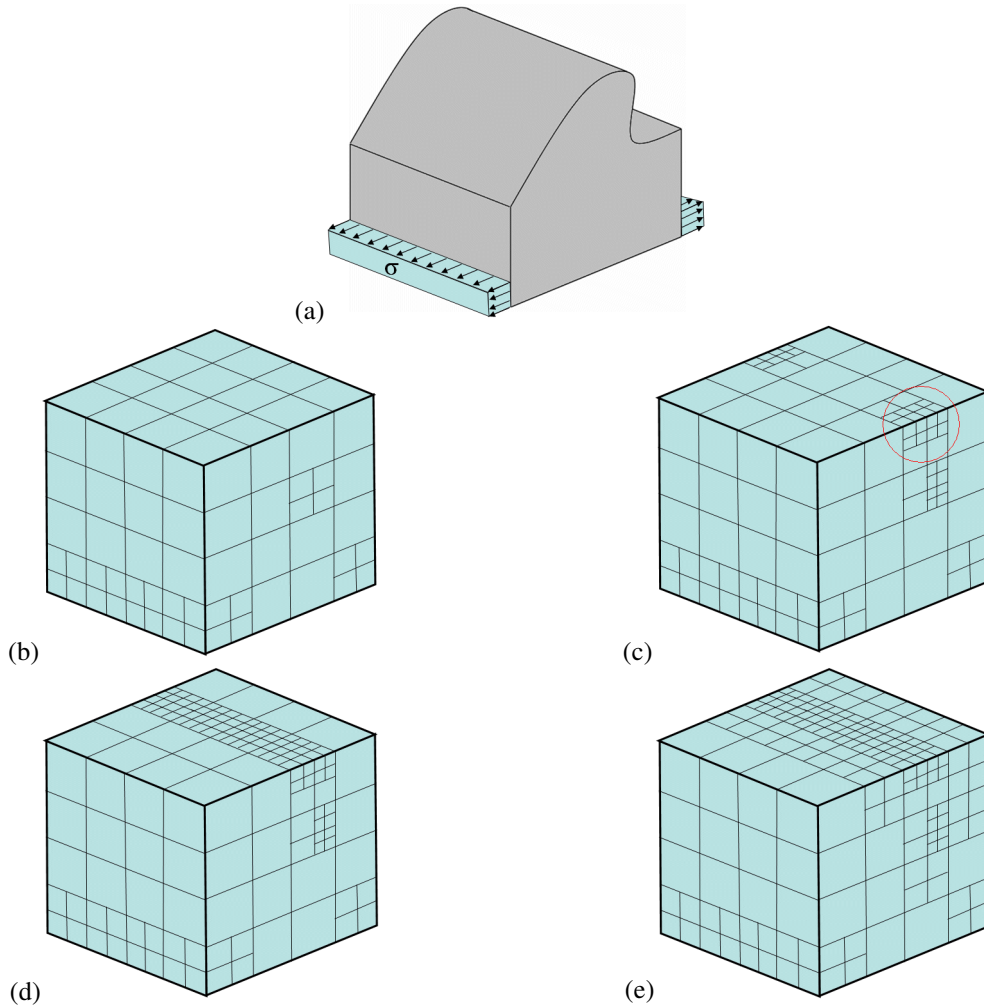
Figure 2: A hypothetical model to explain the steps of octree construction: (a) Model geometry and loading; (b) External appearance of background octree based on discretization error analysis; (c) Refinement after considering curve curvatures; (d) Refinement after considering surface curvatures; (e) Refinement after considering maximum cell size at boundary cells and maximum difference of one level between adjacent cells.

The main purpose is to generate a discretization on a curve according to its curvatures. The curvatures are calculated for specific curve segments. At first, the whole length of the curve is considered as the segment to be tested. If the curvature of the segment is lower than the maximum allowed curvature, the process is interrupted. Otherwise, the segment is recursively subdivided in two segments, and each one is tested in the same way, until the maximum curvature criterion is satisfied. At the end of this process (Figure 2-c), all the curve segment sizes and their middle points are transported to the background octree, using the same procedure explained in Section 3.1.

3.3 Octree refinement based on surface curvature

After the background octree is refined considering curve curvatures, the octree refinement is increased based on surface curvatures. This step (Figure 2-d) captures high curvatures of surfaces, computes the required element sizes and their locations, and passes this information to the background octree. The reason to perform this step is the same as the previous stage: to represent the original geometry of the model.

As in Section 3.2, the methodology applied to refine the surfaces of the model based on their curvatures is a two-dimensional version of the procedure used to discretize the background octree: a background quadtree, which is created similarly to the one presented by Miranda and Martha [25]. This way of computing the curvatures has shown to be efficient and robust, and this is the main reason for its adoption. The background quadtree generation follows some steps:

- Quadtree initialization based on given boundary edges;
- Refinement to force maximum cell size;
- Refinement to provide minimum size disparity for adjacent cells;
- Refinement to force minimum curvature difference between adjacent cells: this stage is explained ahead.

As described in detail by Miranda and Martha [25], the first step has some modifications in relation to the original 3D algorithm [26]. The second and third steps have not changed. The fourth step was added to take high surface curvatures into account.

The fourth step of the quadtree generation refines this auxiliary structure to force a minimum curvature difference between adjacent cells. Initially, the algorithm stores in each cell gradient vectors of the quadtree evaluated at the center of the cell. Then, it computes a vector normal to the surface of each cell. Finally, the algorithm obtains the cosine of the angle θ between the normal vectors of the two adjacent cells and compares it to a minimum value, $\cos\theta_{min}$. This kind of comparison is similar to comparing the angle between the normal vectors and the maximum angle. If $\cos\theta$ is smaller than $\cos\theta_{min}$, then a new cell size, H_{new}, is obtained from the current size, H_{old}, as $H_{new} = (H_{old}/\cos\theta_{min})\cdot\cos\theta$. This new size is used to locally refine the adjacent cells of the quadtree. This process is repeated recursively for every cell. The new element sizes stored in the auxiliary surface quadtree are transferred to the global background octree. At the end of this step, the background octree is refined considering the geometric curvatures of all of the model's surfaces.

3.4 Octree final refinement

The previous step can leave large octree cells in the interior of a 3D region. In the first step of this final stage, the octree is refined to guarantee that no cell in the interior is bigger than the largest cell on the boundary. This will avoid excessively large elements in the domain interior. The octree is subsequently processed to force a single difference level between neighboring cells (Figure 2-e). This leads to a natural transition between regions with different degrees of refinement. This refinement is performed by traversing the octree and examining the difference in tree depth between adjacent cells. If the difference is larger than one level, the adequate cells are refined until the criterion is satisfied.

4   HIERARCHICAL REFINEMENT

After the construction of the background octree, considering the new element sizes based on the discretization error and on curvatures of the geometric model, the final step is to re-generate the mesh of the whole model. As mentioned previously, it is assumed that the three-dimensional geometric model has a topological description of the vertices, curves, surfaces, and regions, as well as an associated geometric description, which consists of the coordinates of the vertices and the mathematical representation of the curves and surfaces. The geometric model can contain many regions. Mesh re-generation employs a hierarchical refinement of (1) curves, (2) surfaces, (3) and regions.

The methodology used to refine the model's curves is similar to the one mentioned in Section 3.2. At the beginning of the process, a curve is defined by its mathematical geometric description and by two nodes (initial and final points). Then, the curve length and middle node are obtained. From the middle node, one can determine the cell in the background octree where this node is located. A comparison is then made to verify whether the segment size is smaller than that of the corresponding cell. If the criterion is satisfied, the curve refinement process ends considering the nodes generated so far. Otherwise, the new node is inserted on the curve, this curve is subdivided in two partitions, and each one is tested in the same way, until the criterion is met.

Surface mesh generation is based on the algorithm presented by Miranda and Martha [25]. This algorithm is applied to the generation of triangular meshes on each surface with arbitrary geometry, using its parametric description. The parametric description is used because it is common and efficient, since the surface mesh is generated using two-dimensional triangulation techniques. However, additional length and angle corrections are needed to consider metric distortions between parametric and 3D Cartesian spaces. With this procedure, generated triangles present good shape in 3D space.

3D mesh generation in each closed region of the model is based on a technique presented by Cavalcante-Neto [26] and is used to obtain tetrahedral elements in arbitrary domains. Similarly to the procedure applied to generate surface meshes, this one is based on an advancing front technique coupled to a recursive spatial decomposition technique (octree). Originally, the algorithm employed an independent background octree in each 3D region to control the distribution of the node points generated in the interior. In the adaptive methodology proposed here, the global background octree is used for this purpose.

The algorithm was designed to meet four specific requirements: to avoid producing elements with poor aspect ratios; to generate meshes conforming to existing triangular meshes at the boundary of a domain; and to generate meshes exhibiting good transitions between regions of different element sizes. The input to the algorithm is a triangular surface mesh, which describes the domain to be meshed. This mesh is obtained from the surface meshes on the boundary of a 3D region to be meshed. The algorithm steps are as follows:

- A two-pass advancing front procedure is applied to generate elements. In the first pass, elements are generated based on geometric criteria, producing well-shaped elements. The background octree presented in Section 3 is used to control the sizes of the elements and the position of the interior nodes. The octree determines an ideal position for an optimal node to form a new element. This ideal position defines a search region where an optimal node for the new element may be located. This region is a sector of a

sphere whose center is the ideal position and whose radius is proportional to the octree cell size. If one or more existing nodes are inside this region, they are ranked based on a solid angle criterion, in order to get the best node for the new element. However, if no existing node is found, a new node is inserted at the ideal position and an element is generated using this node. In the second pass, elements are generated based only on the criterion that they have valid topology. Here, any existing node that forms a valid new element can be used, regardless of whether it is close to the ideal position or not. However, the same quality criterion is used and the node that forms the best solid angle is chosen for the generation of the new element.

- If the advancing front procedure cannot progress, a back-tracking strategy [27] is employed to delete some elements, and the procedure is restarted. It consists basically of back tracking a few steps in the mesh generation and deleting faces that hinder the front from converging. This creates better regions where valid elements can be then generated. It is possible that the process of finding better regions may fail, for instance, if faces to be removed are part of the original boundary. When this occurs, other elements are deleted instead and the procedure is restarted. If a mesh still cannot be generated for this region, the algorithm fails and terminates. In principle, it is possible to create a boundary input mesh that forces the failure of the volume mesh generation. Such failure, however, has not yet been observed in "non-contrived" input, i.e., in any realistic input boundary meshes in many examples tested so far.

- Once a valid mesh is created, the quality of the element shapes is improved by using the standard Laplacian smoothing technique and locally deleting poorly shaped elements and those adjacent to them. The boundary contraction is then restarted.

After the generation of volumetric elements in all regions of the model as exposed above, a new error analysis is performed to assess the quality of the results. If convergence is not obtained, the whole adaptive process is repeated as described in the previous sections. The next section provides some examples of the proposed adaptive refinement process.

5 EXAMPLE

This section presents an example of adaptive 3D finite element mesh that was generated using the proposed adaptive methodology. It is important to emphasize that this paper does not aim to compute the performance of mesh generators (surface and volume) or assess the quality of the elements generated, since these tasks were covered in previous works [25, 26].

The adaptive strategy proposed in this paper results from the application of unstructured mesh generation techniques in surfaces and regions, combined with numerical errors associated with discretization. Numerical error estimators are implemented based on procedures developed for two-dimensional models [28] extended to three dimensions. These error estimators are supported by error estimation techniques widely adopted in the literature, called Superconvergent Path Recovery technique (SPR) [20,21] and Recovery by Equilibrium in Patches (REP) [22]. The numerical error estimators were implemented in a finite element numerical analysis program [29]. It is worth stressing that these error estimation techniques can be easily and directly replaced by any other technique that is more recent or efficient, as this is supported by the object-oriented organization of the analysis program. In the present example, SPR is employed for error estimation.

The presented example is a model of a bike suspension rocker [30], shown in Figure 3, that is composed by four cylinders that are connected by a central body. Uniformly distributed forces, of unitary intensity in the y direction are applied to the internal faces of the two top cylinders. Displacement restrictions are applied to the internal face of the lower cylinder.
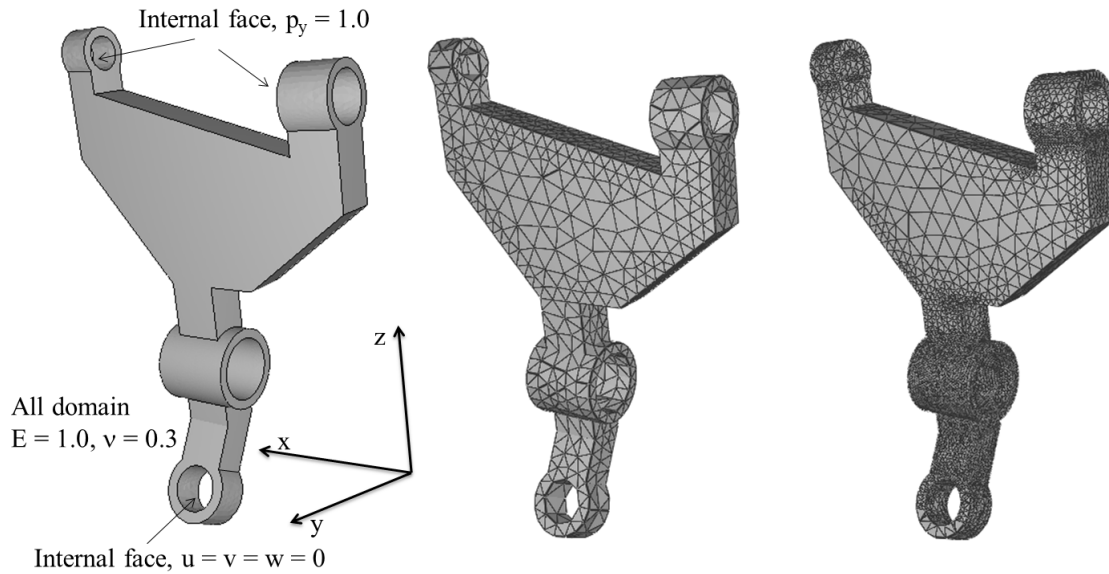

Figure 3: Example: bike suspension rocker model and finite element mesh refinement.

The adaptive refinement of this example (see Figure 3) was carried out until the target relative error (3%) was reached. The mesh is refined in the intermediate cylinder and only part of the lower cylinder, where there are concentrations of stress. The number of linear tetrahedral elements in the initial mesh is 6223 and the number of nodes is 10303. In the final mesh, the number of elements is 186238 and the number of nodes is 273633. This example demonstrates the importance of considering the curvatures of the supporting surface in the adaptive refinement, which is another characteristic of the present methodology.

6   CONCLUSIONS

This paper described a methodology for adaptive generation of three-dimensional finite element meshes, using geometric modeling with multi-regions and parametric surfaces. The mesh adaptive process involves three steps: (1) FE analysis with error estimation; (2) construction of a structure to store the new sizes of the FE; and (3) refinement of the FE model. The approach adopted is the global refinement of the whole model in each adaptive refinement, using a background octree structure. After the construction of the octree, the new model is geometrically re-discretized employing a hierarchical curve, surface and volume refinement. Some important characteristics of the proposed methodology are:

- The ability to refine and coarsen in regions of high and low response gradients.
- The use of only one background octree for all regions of the model, allowing a smooth transition between regions and elements.
- The hierarchical refinement of curves, surfaces, and volumes.

- The consideration of curvatures of curves and surfaces in the adaptive refinement.

A recent article [31] published by the authors compares results of convergence rates of the proposed methodology with results presented in the literature. It was demonstrated that the current methodology converges faster to a lower relative error, because the 3D mesh generator used has more freedom to create new elements based on desired element sizes. This characteristic generates a desired mesh with the application of only a few steps of the adaptive refinement. In contrast, using a local refinement strategy, (local) element manipulations restrict the shape quality of new elements.

Obviously, the current 3D mesh generation takes more time to create new elements, because the whole FE model must be created at each step. In this work, most of meshes were generated in less than one minute of clock time. However, in models that require a large number of elements (one million or more), time consumption can increases exponentially. While in many problems this is not an issue, it can be a limitation of the current approach that can be solved in two manners: (1) decomposing the domains into sub-domains and applying the mesh generator to each sub-domain; or (2) using a parallel 3D mesh generator. The latter option is our future work and is currently under development.

REFERENCES

[1] Mackerle, J. Error estimates and adaptive finite element methods: A bibliography (1990-2000). *Eng. Comput.* (2001) **18**:802-914.

[2] McRae, D.S. R-refinement grid adaptation algorithms and issues. *Comput. Meth. Appl. Mech. Eng.* (2000) **189**:1161-1182.

[3] Kallinderis, Y. and Vijayant, P. Adaptive Refinement-Coarsening Scheme for Three-Dimensional Unstructured Meshes. *AIAA Journal* (1993) **43**(8):1440-1447.

[4] Muthukrishnan, N., Shiakolas, P.S., Nambiar, R.V., and L, L.K. Simple algorithm for adaptive refinement of three dimensional finite element tetrahedral meshes. *AIAA Journal* (1995) **33**(5):928-932.

[5] Golias, N.A. and Tsiboukis, T.D. An approach to refining three-dimensional tetrahedral meshes based on delaunay transformations. *Int. J. Num. Meth. Engng.* (1994) **37**:793-801.

[6] Golias, N.A. and Dutton, R.W. Delaunay triangulation and 3D adaptive mesh generation. *Finite Elem. Anal. Des.* (1997) **25**:331-341.

[7] Lee, C.K. and Lo, S.H. Automatic adaptive refinement finite element procedure for 3D stress analysis. *Finite Elem. Anal. Des.* (1997) **25**:135-166.

[8] Lee, C.K. and Lo, S.H. A full 3D finite element analysis using adaptive refinement and pcg solver with back interpolation. *Comput. Meth. Appl. Mech. Eng.* (1999) **170**:39-64.

[9] Merrouche, A., Selman, A., and Knopf-Lenoir, C. 3D adaptive mesh refinement. *Commun. Numer. Methods Eng.* (1998) **14**(5):397-407.

[10] De Cougny, H.L. and Shephard, M.S. Parallel refinement and coarsening of tetrahedral meshes. *Int. J. Num. Meth. Engng.* (1999) **46**:1101-1125.

[11] Lee, C.K. and Xu, Q.X. A new automatic adaptive 3D solid mesh generation scheme for thin-walled structures. *Int. J. Num. Meth. Engng.* (2005) **62**:1519-1558.

[12] Qian, J. and Zhang, Y. Automatic unstructured all-hexahedral mesh generation from b-reps for non-manifold cad assemblies. *Eng. Comput.* (2012) **28**:345-359.

[13] Zhang, Y., Hughes, T.J., and Bajaj, C.L. An automatic 3D mesh generation method for

domains with multiple materials. *Comput. Meth. Appl. Mech. Eng.* (2010) **199**:405-415.

[14] Kettil, P., Ekevid, T., and Wiberg, N.E. Towards fully mesh adaptive FE-simulations in 3D using multi-grid solver. *Computers and Structures* (2003) **81**:735-746.

[15] Hughes, T.J., Cottrell, J.A., and Bazilevs, Y. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Comput. Meth. Appl. Mech. Eng.* (2005) **194**:4135-4195.

[16] Cavalcante-Neto, J.B., Martha, L.F., Menezes, I.F.M., and Paulino, G.H. A methodology for self-adaptive finite method analysis using an object oriented approach. *Fourth World Congress on Computational Mechanics* (1998).

[17] Baehmann, P.L. and Shephard, M.S. Adaptive multiple-level h-refinement in automated finite element analysis. Engineering with Computers (1989) **5**:235-247.

[18] Cavalcante-Neto, J.B. Mesh generation and error estimative for finite element 3D models with cracks. *PhD dissertation, Pontifical University Catholic of Rio de Janeiro*.

[19] Zienkiewicz, O.C. and Taylor, R.L. *The finite element method*. McGraw Hill (1991).

[20] Zienkiewicz, O.C. and Zhu, J.Z. The superconvergent patch recovery and a posterior error estimates. Part 1: The recovery technique. *Int. J. Num. Meth. Engng.* (1992) **33**:1331-1364.

[21] Zienkiewicz, O.C. and Zhu, J.Z. The superconvergent patch recovery and a posterior error estimates. Part 2: Error estimates and adaptivity. *Int. J. Num. Meth. Engng.* (1992) **33**:1365-1382.

[22] Boroomand, B. and Zienkiewicz, O.C. Recovery by equilibrium in patches (REP). *Int. J. Num. Meth. Engng.* (1997) **40**:137-164.

[23] Quadros, W.R., Vyas, V., Brewer, M., Owen, S.J., and K, S. A computational framework for automating generation of sizing function in assembly meshing via disconnected skeletons. *Eng. Comput.* (2010) **26**(3):231-247.

[24] Knuth, D. *The art of computer programming.* Addison-Wesley (1997).

[25] Miranda, A.C.O. and Martha, L.F. Mesh generation on high-curvature surfaces based on a background quadtree structure. *11th International Meshing Roundtable* (2002) 333-341.

[26] Cavalcante-Neto, J.B., Wawrzynek, P.A., Carvalho, M.T.M., Martha, L.F., and Ingraffea, A.R. An algorithm for three-dimensional mesh generation for arbitrary regions with cracks. *Eng. Comput.* (2001) **17**(1):75-91.

[27] Cavalcante-Neto, J.B., Martha, L.F., Wawrzynek, P.A., and Ingraffea, A.R. A backtracking procedure for optimization of simplex meshes. *Commun. Numer. Methods Eng.* (2005) **21**(12):711-722.

[28] Paulino, G.H., Menezes, I.F.M., Cavalcante Neto, J.B., and Martha, L.F. A methodology for adaptive finite element analysis: Towards an integrated computational environment. *Comput. Mech.* (1999) **23**:361-388.

[29] Martha, L.F. and Parente Jr., E. An object-oriented framework for finite element programming. *Fifth World Congress on Computational Mechanics* (2002).

[30] Cuillière, J., Francois, V., and Drouet, J. Automatic 3D mesh generation of multiple domains for topology optimization methods. *21th International Meshing Roundtable* (2012) 243-249.

[31] Miranda, A.C.O., Lira, W.W.M., Cavalcante-Neto, J.B., Sousa, R.A., and Martha, L.F., A 3D adaptive mesh generation approach using geometric modeling with multi-regions and parametric surfaces. *J. Comput. Inf. Sci. Eng.* (2013), accepted for publication.

ON THE KINEMATIC STABILITY OF HYBRID EQUILIBRIUM TETRAHEDRAL MODELS

EDWARD A. W. MAUNDER*, JOSÉ P. MOITINHO DE ALMEIDA[†]

AND ORLANDO J.B.A. PEREIRA[†]

* College of Engineering, Mathematics and Physical Sciences, University of Exeter
Harrison Building, North Park Road, Exeter EX4 4QF, UK
e-mail: e.a.w.maunder@exeter.ac.uk

[†] Department of Civil Engineering, Architecture and Georesources
Instituto Superior Técnico, Technical University of Lisbon,
Av. Rovisco Pais, 1049-001 Lisboa, Portugal
email: moitinho@civil.ist.utl.pt; orlando@civil.ist.utl.pt

Key words: Equilibrium models, tetrahedral elements, kinematic stability.

Summary. *This paper is concerned with establishing the nature of the kinematic instabilities that arise in tetrahedral hybrid equilibrium models when the elements are formulated with polynomial approximation functions of a general degree. The instabilities are due to the spurious kinematic (or zero energy) modes, and these modes are first derived for a single element. The paper continues by identifying those spurious modes that can be propagated from one element to another via an interface. It is shown that at least three such modes exist for all degrees.*

1 INTRODUCTION

Hybrid equilibrium elements have been used to generate dual analyses for error estimation of conforming models[1]. Dual analyses may involve reanalysis of a complete mesh, or may involve local analyses of star patches[2]. In any event it becomes important to know whether spurious kinematic modes associated with hybrid equilibrium models may exist, and if so, whether they will affect the dual analyses. These questions have been studied for plate elements[3-5]. In this paper, we investigate the form taken by spurious kinematic modes for a single tetrahedral element of general polynomial degree, and consider the propagation of these modes between a pair of adjacent elements of the same degree. The results of this investigation should help to determine the general kinematic stability of patches of tetrahedral elements[6,7], thereby setting the basis for robust implementations of these approaches. The definition of spurious modes associated with an edge and an interface of an element exploit an orthogonal basis of polynomials for a triangular face[8] that are expressed in terms of area coordinates. These enable the spurious modes to be generated in a hierarchical fashion which takes advantage of cyclic symmetry.

2 GENERAL FEATURES OF SPURIOUS KINEMATIC MODES OF TETRAHEDRA

Spurious kinematic modes refer to boundary displacements that have the nature of pseudo-mechanisms and cause no internal stress. They do zero work with admissible boundary tractions, which are those that equilibrate with internal stress fields. Displacements of a face of a tetrahedron are described by complete polynomials of degree d, and this implies that the dimensions of the spaces of displacements and rigid body modes for an element are defined in Equation (1). Internal stress fields are described by polynomials of the same degree, and complete within the constraints set by equilibrating with zero body forces. In this case the dimensions of the stress and hyperstatic stress spaces are given by[9] Equation (2).

$$n_v = 4 \times 3 \times 0.5(d+1)(d+2); \ n_{rbm} = 6 \tag{1}$$

$$n_s = 0.5(d+1)(d+2)(d+6); \ n_{hyp} = 0.2(d-3)(d-2)(d+2) \text{ for } d > 2 \tag{2}$$

and then the number of independent spurious kinematic modes is given by Equation (3).

$$n_{skm} = (n_v - n_{rbm}) - (n_s - n_{hyp}) = 6(d+1) \text{ for } d > 2. \tag{3}$$

When $d \leq 2$, the element is isostatic and then:

$$n_{skm} = 0.5(d+1)(d+2)(6-d) - n_{rbm}. \tag{4}$$

Tractions applied to the boundary are considered as belonging to a space dual to that of displacements. Admissible tractions are those that equilibrate with an internal stress field, and the necessary and sufficient conditions for admissibility correspond to the need for complementary shear stresses along an edge of a tetrahedral element. With reference to Figure 1, the complementary shear stress condition along edge 3-4 has the form in Equation (5).

$$\sin\varphi \cdot \tau_{1n} - \cos\varphi \cdot \sigma_1 + \sin\varphi \cdot \tau_{2n} + \cos\varphi \cdot \sigma_2 = 0, \tag{5}$$

where φ is the dihedral angle between faces adjacent to the edge. For traction fields of degree $d \geq 2$, $(d+1)$ independent conditions associated with each edge lead to the homogeneous admissibility conditions on generalised element tractions represented by vector $\boldsymbol{g}$, i.e. $\boldsymbol{A^T g = 0}$ where the dimensions of $\boldsymbol{A}$ are $n_v \times n_{skm}$. The spurious kinematic modes are then defined in terms of the dual basis for displacements by the columns of $\boldsymbol{A}$.

3 SPURIOUS KINEMATIC MODES FOR A TETRAHEDRAL ELEMENT OF GENERAL DEGREE

A convenient basis for polynomial displacement or traction functions over a triangular face is derived from the functions in the *Digital Library of Mathematical Functions* [8]. These $n = 0.5(d+1)(d+2)$ functions have the properties of orthogonality and the benefit of a hierarchical structure. When expressed in terms of area coordinates L_i they can be organised in a vector $\boldsymbol{h}$ to give Legendre polynomials along a particular edge, which leads to a very simple form of $\boldsymbol{A}$ when it is restricted to the two faces adjacent to that edge. This form, with dimensions $(4n \times (d+1))$, is defined by $\bar{A}$ in Equation (6), e.g. for the edge where $L_1 = 0$.

$$\bar{A} = \left[\begin{array}{c} \sin\varphi \cdot \Delta_1^{-1}HJ \\ \hline -\cos\varphi \cdot \Delta_1^{-1}HJ \\ \hline \sin\varphi \cdot \Delta_2^{-1}H \\ \hline \cos\varphi \cdot \Delta_2^{-1}H \end{array} \right] \tag{6}$$

where Δ_1 and Δ_2 are the areas of faces 1 and 2 in Figure 1, $J = I_{(d+1)}$ with even numbered diagonal coefficients = -1, and the $n \times (d+1)$ matrix H and the n dimensional vector h are defined in Equation (7).
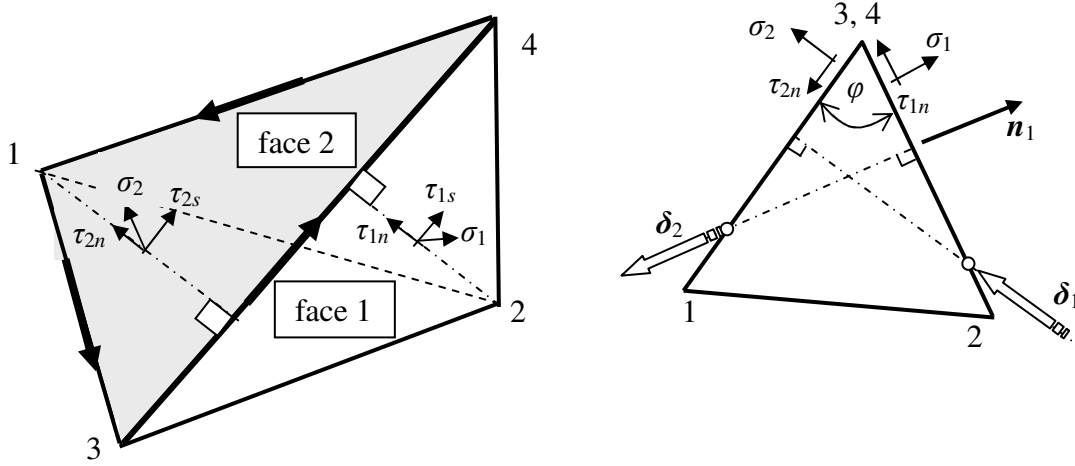

Figure 1: Traction components on a tetrahedron relative to edge connecting vertices 3 and 4. The right hand view is projected from vertex 3 to vertex 4.

$$H = \left[\begin{array}{ccccc} 2 & 0 & 0 & \cdots & 0 \\ \hline 4 & 0 & 0 & \cdots & 0 \\ \hline 0 & 12 & 0 & \cdots & 0 \\ \hline 6 & 0 & 0 & \cdots & 0 \\ \hline 0 & 18 & 0 & \cdots & 0 \\ \hline 0 & 0 & 30 & 0 & \cdots \\ \hline 8 & 0 & 0 & 0 & \cdots \\ \hline 0 & 24 & 0 & 0 & \cdots \\ \hline 0 & 0 & 40 & 0 & \cdots \\ \hline 0 & 0 & 0 & 56 & \cdots \\ \hline \ddots & \ddots & \ddots & \ddots & \ddots \end{array} \right] \quad \text{when } h = \left\{ \begin{array}{c} 1 \\ \hline 1 - 3L_1 \\ \hline -L_2 + L_3 \\ \hline 1 + 2L_1\left(-4 + 5L_1\right) \\ \hline \left(-1 + 5L_1\right)\left(L_2 - L_3\right) \\ \hline L_2^2 - 4L_2L_3 + L_3^2 \\ \hline 1 - 5L_1\left(3 + L_1\left(-9 + 7L_1\right)\right) \\ \hline \left(1 + 3L_1\left(-4 + 7L_1\right)\right)\left(-L_2 + L_3\right) \\ \hline \left(1 - 7L_1\right)\left(L_2^2 - 4L_2L_3 + L_3^2\right) \\ \hline -\left(L_2 - L_3\right)\left(L_2^2 - 8L_2L_3 + L_3^2\right) \\ \hline \vdots \end{array} \right\} \tag{7}$$

Similar bases of h for other edges of the face are obtained using cyclic symmetry. Then the mth column of H defines a signature function k_{2e}^m for face 2 corresponding to edge e as a combination of the basis functions in h. The total displacement vector of a point in face 2 due to the spurious kinematic modes associated with its three edges, oriented as in Figure 1, is

given by Equation (8), where $\boldsymbol{n}_e$ is the unit outward normal vector to the other face adjacent to edge e, and a_e^m is the amplitude of the mth spurious mode associated with edge e.

$$\boldsymbol{\delta}_2 = \left(-\frac{1}{\Delta_2} \sum_m \sum_e k_{2e}^m \cdot a_e^m \right) .\boldsymbol{n}_e \tag{8}$$

4 PROPAGATION OF "MALIGNANT" SPURIOUS MODES BETWEEN TETRAHEDRAL ELEMENTS.

Propagation of spurious kinematic modes can occur between a pair of elements A and B when they result in compatible displacements at the interface. The displacements are resolved into in-plane and normal components as indicated in Figure 2 at a point P. For each signature function k_{je}^m displacements are evaluated at a set of n grid points with a common set of rigid body constraints. This leads to the displacement Equation (9) for element A.

$$\boldsymbol{u} = \boldsymbol{E}^A \cdot \boldsymbol{a}^A \quad \text{and} \quad w = \boldsymbol{C}^A \cdot \boldsymbol{a}^A \tag{9}$$

where $\boldsymbol{E}^A$ and $\boldsymbol{C}^A$ contain displacement components corresponding to spurious modes of unit amplitude, and have dimensions $(2n \times 3(d+1))$ and $(n \times 3(d+1))$ respectively. The amplitudes of the spurious modes are collected in the vector $\boldsymbol{a}^A$. The matrices can be expressed as in Equation (10), where the diagonal matrices are defined in terms of the Kronecker products in Equation (11) and φ_e^A denotes the dihedral angle at edge e of element A.

$$\boldsymbol{E}^A = \left[\boldsymbol{E}_1 \mid \boldsymbol{E}_2 \mid \boldsymbol{E}_3 \right] \boldsymbol{D}_s^A = \boldsymbol{E} \cdot \boldsymbol{D}_s^A \text{ , and } \quad \boldsymbol{C}^A = \left[\boldsymbol{C}_1 \mid \boldsymbol{C}_2 \mid \boldsymbol{C}_3 \right] \boldsymbol{D}_c^A = \boldsymbol{C} \cdot \boldsymbol{D}_c^A \tag{10}$$
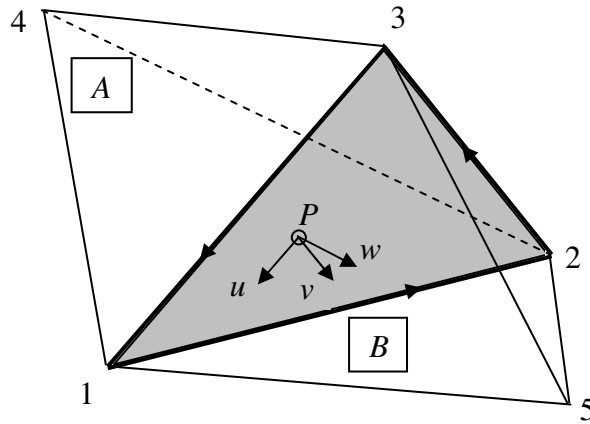
Figure 2: Interface between elements A and B.

Matrices $\boldsymbol{E}$ and $\boldsymbol{C}$ are partitioned in Equation (10) to match the coefficients from edges 1 to 3. Since these matrices are only dependent on the signature functions, which are expressed in terms of area coordinates, they are independent of the shape of the interface or the dihedral angles.

$$D_s^A = \frac{1}{2\Delta} \begin{bmatrix} \sin\varphi_1^A & 0 & 0 \\ 0 & \sin\varphi_2^A & 0 \\ 0 & 0 & \sin\varphi_3^A \end{bmatrix} \otimes I_{d+1} \quad \text{and} \quad D_c^A = \frac{1}{2\Delta} \begin{bmatrix} \cos\varphi_1^A & 0 & 0 \\ 0 & \cos\varphi_2^A & 0 \\ 0 & 0 & \cos\varphi_3^A \end{bmatrix} \otimes I_{d+1}. \quad (11)$$

Compatibility conditions take the form in Equation (12), where the vector $\lfloor a^A \mid a^B \rfloor^T$ contains the amplitudes of the $6(d+1)$ spurious modes associated with the edges of the interface belonging to elements A and B. The diagonal matrices D_s^B and D_c^B for element B are similar to those for element A, but involve the dihedral angles φ_e^B.

$$\left[\begin{array}{c|c} ED_s^A & -ED_s^B \\ \hline CD_c^A & CD_c^B \end{array} \right] \left\{ \begin{array}{c} a^A \\ \hline a^B \end{array} \right\} = \left\{ \begin{array}{c} 0 \\ \hline 0 \end{array} \right\} \quad (12)$$

It is found that generally ($d \geq 2$) E has full column rank and so any spurious mode involves in-plane deformation of the interface, and compatibility requires $D_s^A \cdot a^A = D_s^B \cdot a^B$. Eliminating a^A from the second set of Equation (12), leads to Equation (13).

$$C \left[D_{\text{cot}}^A + D_{\text{cot}}^B \right] \left[D_s^B \right] a^B = 0 \quad (13)$$

where the suffix "cot" implies that $\cot\varphi_e$ replaces $\cos\varphi_e$ or $\sin\varphi_e$ in the diagonal matrices. The consequence of Equation (13) is that compatibility can be satisfied:

- when C is column rank deficient, and/or
- the geometrical configuration is degenerate in the sense that $\cot\varphi_e^A + \cot\varphi_e^B = 0$ for one or more dihedral angles, i.e. faces in adjacent elements are coplanar.

Numerical trials involving singular value decomposition reveal that, when $d > 3$, C has column rank $3d$ and hence 3 spurious kinematic modes can be propagated via the interface in the non-degenerate case. These modes are linearly related to independent solutions of the homogeneous equations $Ca = 0$, and such solutions are given by the $3 \times 3(d+1)$ matrix in Equation (14).

$$a^T = \left[\begin{array}{cccccc|cccccc|cccccc} 0 & 1 & 0 & 1 & \cdots & -1 & 0 & -1 & 0 & \cdots & 1 & 0 & 1 & 0 & \cdots \\ 1 & 0 & 1 & 0 & \cdots & 0 & 1 & 0 & 1 & \cdots & -1 & 0 & -1 & 0 & \cdots \\ -1 & 0 & -1 & 0 & \cdots & 1 & 0 & 1 & 0 & \cdots & 0 & 1 & 0 & 1 & \cdots \end{array} \right]. \quad (14)$$

The corresponding spurious mode amplitudes for element A can then be defined in Equation (15).

$$a^A = \frac{1}{\Delta^2} \left[D_s^A \right]^{-1} \left[D_{\text{cot}}^A + D_{\text{cot}}^B \right]^{-1} a. \quad (15)$$

This solution can also be expressed, using the Kronecker matrix product in Equation (16).

$$
\mathbf{a}^A = \begin{bmatrix} \dfrac{\sin\varphi_1^B}{\sin\left(\varphi_1^A + \varphi_1^B\right)} & 0 & 0 \\[3ex] 0 & \dfrac{\sin\varphi_2^B}{\sin\left(\varphi_2^A + \varphi_2^B\right)} & 0 \\[3ex] 0 & 0 & \dfrac{\sin\varphi_3^B}{\sin\left(\varphi_3^A + \varphi_3^B\right)} \end{bmatrix} \otimes \mathbf{I}_{d+1} \cdot \mathbf{a} \tag{16}
$$

It is observed from the form of $\mathbf{a}^T$ that the three modes can be characterised by a single one, which generates two more independent ones by using cyclic symmetry. When the pair of elements are symmetrical about the interface, $\mathbf{D}_{\text{cot}}^A = \mathbf{D}_{\text{cot}}^B$ and in this case $\mathbf{CD}_c^A \cdot \mathbf{a}^A = \mathbf{0}$ and consequently the interface remains plane. The characteristic mode for a pair of regular tetrahedra of degree 4 is illustrated by the in-plane displacements shown in Figure 3.

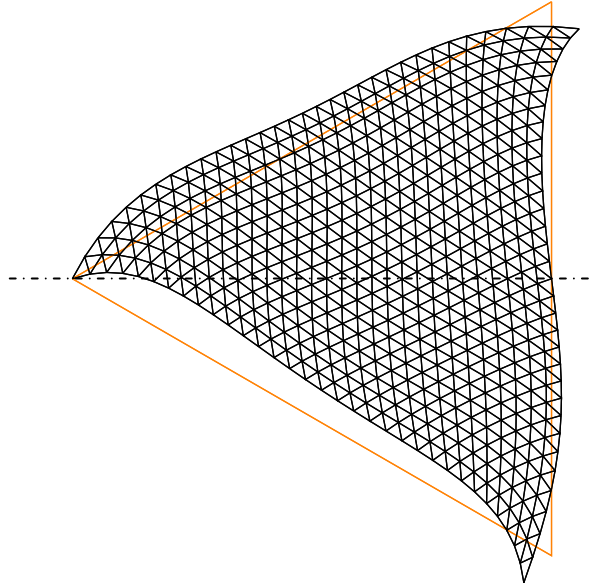Figure 3: Characteristic spurious kinematic mode that can be propagated between a pair of regular tetrahedral elements of degree 4.

The number of malignant modes for non-degenerate cases increases for $d < 4$, and the complete set of numbers is presented in Table 1.

Table 1: Number of malignant modes for a general degree.

d	0	1	2	3	≥ 4
number of malignant modes	3	6	6	5	3

It should be noted that when:

$d = 1$, rank(E) = 3, and the 6 spurious modes associated with the interface of one of the elements can freely exist in a state of constant strain coupled with a rigid body displacement. Thus only 3 of the combined modes can be propagated to involve deformations.

$d = 0$, the 3 spurious modes associated with the interface of one of the elements can freely exist as rigid body translations. Thus all the modes can be freely propagated as rigid body modes.

5. CLOSURE

A pair of tetrahedral hybrid equilibrium elements always has the potential for at least three spurious kinematic modes to be propagated from one element to the other. This feature of 3D tetrahedral models is more complicated than the case with 2D models with triangular elements, where such propagation is normally blocked for degrees greater than two.
Thus establishing the existence of spurious kinematic modes in a pair of tetrahedral elements is just the first stage in understanding when and how these modes can propagate in a more general mesh. Whilst experience has shown[6] that assemblies of four tetrahedra into a single macro-element is free of spurious kinematic modes, it is intended to pursue further research to address the stability of more general configurations.

REFERENCES

[1] Almeida JPM, Pereira OJBA. Upper bounds of the error in local quantities using equilibrated and compatible finite element solutions for linear elastic problems. *Comp. Meth. Appl. Mech. Engng.* (2006),**195**:279-296.
[2] Almeida JPM, Maunder EAW. Recovery of equilibrium on star patches using a partition of unity technique. *Int. J. Num. Meth. Engng.* (2009) **79**:1493-1516.
[3] Maunder EAW, Almeida JPM. A triangular hybrid equilibrium plate element of general degree. . *Int. J. Num. Meth. Engng.* (2005) **63**:315-350.
[4] Maunder EAW, Almeida JPM. The stability of stars of triangular equilibrium plate elements. . *Int. J. Num. Meth. Engng.* (2009) **77**:922-968.
[5]Maunder EAW. On the stability of hybrid equilibrium and Trefftz finite element models for plate bending problems. *Comp. Ass. Mech. Engng. Sci.* (2008) **15**, 279-288.
[6] Almeida JPM, Pereira OJBA, A set of hybrid equilibrium finite element models for the analysis of three-dimensional solids. *Int. J. Num. Meth. Engng*. (1996) **39,** 2789-2802.
[7] Pereira OJBA, Hybrid equilibrium hexahedral elements and super-elements. *Commun. Numer. Meth. Engng*. (2008) **24**, 157-165.
[8] National Institute of Standards and Technology *Digital Library of Mathematical Functions*, http://dlmf.nist.gov/18.37.E7, Release 1.0.5 of 2012-10-01.
[9] Kempeneers M, Debongnie J-F, Beckers P, Pure equilibrium tetrahedral finite elements for global error estimation by dual analysis. *Int. J. Num. Meth. Engng*. (2009) **81,** 513-536.

MATHEMATICAL MODELING OF SPALLATION OF CONDENSED MATTER UNDER THE INFLUENCE OF CONCENTRATED ENERGY FLOW

VLADIMIR I. MAZHUKIN, ALEXANDER V. MAZHUKIN, MIKHAIL M. DEMIN

Keldysh Institute of Applied Mathematics of RAS
Miusskaya sq., 4. Moscow, 125047
e-mail: vim@modhef.ru

Key words: Modeling, Adaptive grid, Dynamic adaptation, Spallation, Laser irradiation.

Abstract. The process of mechanical spallation of a solid aluminum target under the influence of an ultrashort laser pulse is considered. Modeling was carried out using the method of dynamic adaptation with explicit front tracking. The pulse duration was 100 fs, fluence 0.27 J/cm^2. Modeling allowed to determine the thickness of the spalled layers, investigate the regimes in which spallation occurs in melt.

1 INTRODUCTION

The study of dynamic fragmentation in shock loaded metals and evaluation of geometric and kinematic properties of the resulting fragments is a topical issue for both fundamental and applied science. Among the dynamic processes of fragmentation, spall fracture of solid materials has been one of the most widely studied phenomena in a few decades [1,2]. The spallation is defined as a break of the media due to the stress that exceeds the strength of the substance. The main mechanism of rupture is the propagation of a compression pulse in a solid sample, that is reflected from the free surface and after the interaction with the incident unloading wave creates a tensile stress, which can lead to the damage in the material, ranging from small voids and cracks to complete destruction and release of spalled material.

In recent years, there has been increasing interest in such phenomena in the liquid phase, which develop after partial or complete melting, for example, of metals which are exposed at the same time to compression or tension. Rapid heating of the metal target by ultra-short laser pulses (USLP) with femto-and picosecond duration may result in formation of stressed states in the subsurface region. The unloading of these states may result in the ablation of the irradiated melted layer. The process of dynamic fragmentation of liquid leads to the formation of a cloud of liquid droplets ejected into the space at a high speed. Understanding of the physical processes accompanying the process of fast ablation is necessary to determine the optimum regimes of treatment of materials with USLP. The main method of theoretical research is molecular - dynamics simulation [3-5] that uses atomistic models. Continuous models are used much less frequently. Their use is limited by high computational complexity,

which is in the first place associated with the description of the initiation and propagation of the phase fronts.

This paper deals with the theoretical modeling of the mechanisms of fragmentation of liquid, calculation of the process of the ejection of molten droplets and determination of the amount of obtained fragments. The modeling of fragmentation is based on the use of the continuum hydrodynamic non-equilibrium model [6,7].

2 STATEMENT OF THE PROBLEM

Laser radiation propagates from the right to the left and, striking the surface of the metal target, is partly absorbed and partly reflected. The absorbed energy is used for heating, phase changes, generation of shock waves in the solid phase and dynamic fragmentation of the irradiated target. The main features of the high-power laser action on metals are associated with the high speed and volume type of the energy release of the laser pulse in the electronic component. Ultrafast heating of the metal targets by high-power laser pulses causes a strong deviation from the state of local thermodynamic equilibrium (LTE) and requires the appropriate adjustments in the mathematical model. Along with the thermodynamic equilibrium, we should consider the kinetic non-equilibrium of the high-speed phase transitions and strongly overheated metastable states caused by powerful cross-flow of material through the phase boundary. The model should also take into account the powerful dynamic effects associated with the rapid propagation of the phase fronts. The inclusion of these processes requires the explicit description of the kinetics of the phase transitions and the formulation of the conservation laws at the phase fronts, which are hydrodynamic discontinuities. The mathematical description of these processes is accomplished within the framework of the two-temperature and spatially one-dimensional multi-front non-equilibrium hydrodynamic Stefan problem, written for two phases - solid and liquid. When large negative pressure values result in fragmentation of the target, the following mathematical statement is written for both the bulk target and for every spalled part.

2.1 System of equations.

$$
\left(
\begin{aligned}
& \frac{\partial \rho}{\partial t} + \frac{\partial(\rho u)}{\partial x} = 0 \\
& \frac{\partial(\rho u)}{\partial t} + \frac{\partial(\rho u^2)}{\partial x} + \frac{\partial P}{\partial x} = 0 \\
& \frac{\partial(\rho_e \varepsilon_e)}{\partial t} + \frac{\partial(\rho_e u \varepsilon_e)}{\partial x} = -\left(P_e \frac{\partial u}{\partial x} + \frac{\partial W_e}{\partial x} + g(T_e)(T_e - T_{ph}) + \frac{\partial G}{\partial x} \right) \\
& \frac{\partial(\rho \varepsilon_{ph})}{\partial t} + \frac{\partial(\rho_{ph} u \varepsilon_{ph})}{\partial x} = -\left(P_{ph} \frac{\partial u}{\partial x} + \frac{\partial W_{ph}}{\partial x} - g(T_e)(T_e - T_{ph}) \right) \\
& \frac{\partial G}{\partial x} + \alpha(T_e)G = 0 \\
& P = P(\rho, T), \quad \varepsilon_e = C_e(T_e)T_e, \quad \varepsilon_{ph} = C_{ph}(T_{ph})T_{ph}
\end{aligned}
\right)_k
, t > 0,
\begin{aligned}
& 0 < x < \Gamma_{sl} \cup \\
& \Gamma_{sl} < x < \Gamma_{l\upsilon}
\end{aligned}
\quad (1)
$$

where $W_e = -\lambda_e(T_e,T_{ph})\dfrac{\partial T_e}{\partial x}, W_{ph} = -\lambda_{ph}(T_{ph})\dfrac{\partial T_{ph}}{\partial x}, \quad P(\rho,T) = P(\rho_e,T_e) + P(\rho_{ph},T_{ph}).$

Here: $\rho, u, \varepsilon, T, P$ - are the density, gas-dynamic velocity, internal energy, temperature and pressure of the media respectively, $\alpha(T_e)$ - coefficient of volume absorption, G - density of the laser radiation, $C_{e,ph}$ - heat capacity, $\lambda_{e,ph}$ - heat conductivity, $g(T_e)$ - electron-phonon coupling factor. Indexes s, l, υ correspond to solid, liquid and vapor phases respectively, e, ph - correspond to electron and phonon gas. The expressions for the electron heat capacity, heat conductivity and electron-phonon coupling factor are described in detail in the book [7].

2.2 Initial and boundary conditions

$t = 0: \quad u(0,x) = 0, \quad p = 0, \quad \rho = \rho_0, \quad T_e = T_{ph} = T_0 = 293\,\text{K}.$

On the left stationary border, we write the condition of a zero mass and heat flux:

$x = \Gamma_s: \quad \rho_s u_s = 0, \quad W_T = 0$

On the moving interphase melting boundary $x = \Gamma_{sl}(t)$, we use the boundary conditions according to the non-equilibrium model of melting that is a non-equilibrium kinetic analogue to the equilibrium Stephan problem formulated for a state sufficiently far from the local thermodynamic equilibrium. The system of equations at the boundary consists of the conservation laws of mass, momentum and energy, written for a stationary (laboratory) coordinate system:

$$x = \Gamma_{sl}(t): \qquad \rho_s(u_s - \upsilon_{sl}) = \rho_l(u_l - \upsilon_{sl})$$

$$P_{ph,s} + \rho_s(u_s - \upsilon_{sl})^2 = P_{ph,l} + \rho_s(u_s - \upsilon_{sl})^2 \qquad (2)$$

$$\left(\lambda_{ph}\frac{\partial T_{ph}}{\partial x}\right)_s - \left(\lambda_{ph}\frac{\partial T_{ph}}{\partial x}\right)_l = \rho_s L_m^{ne}\upsilon_{sl}$$

These conservation laws are supplemented with the pressure dependence of the melting temperature and kinetic formula for the surface overheating dependence of the melting front velocity $\Delta T_{sl} = T_{sl} - T_m(P_s)$:

$$T_m = T_m(P_s) = T_{m,0} + kP_{ph,s}$$

$$\upsilon_{sl}(\Delta T_{sl}) = \frac{af}{\lambda}(3k_B T_{sl}/m)^{1/2}\left(1 - exp\left(\frac{L_m^{ne}\mu}{R}\frac{\Delta T_{sl}}{T_m T_{sl}}\right)\right)$$

Additional account of the hydrodynamic effects is made by non-equilibrium heat of melting L_m^{ne}:

$$L_m^{ne} = L_m^{eq}(T_m(P_s)) + \Delta C_{ps} \Delta T_{sl} + \frac{\rho_s + \rho_l}{\rho_s - \rho_l} \frac{(u_s - u_l)^2}{2},$$

where $\Delta C_{ps} = (C_{ps} - C_{pl})$.

At the phase boundary, the electronic component is assumed to be continuous relative to the electron density N_e and temperature T_e:

$$\left(\lambda_e \frac{\partial T_e}{\partial x} \right)_s = \left(\lambda_e \frac{\partial T_e}{\partial x} \right)_l, \quad T_{e,s} = T_{e,l}.$$

At the moving evaporation front, $x = \Gamma_{kv}(t)$, the model of surface evaporation in the approximation of the Knudsen layer is used as a boundary condition. The model consists of three conservation laws, and three additional parameters at the outer side of the Knudsen layer (T_v, ρ_v, u_v). In general, two of these three parameters are determined using the specific approximating relationships [8], and the third is found from the gas-dynamic equations. The expressions for the evaporation front are given in detail in the book [7].

The boundary conditions for the electron component take the form:

$$-\lambda \frac{\partial T_e}{\partial x} = \sigma T_e^4, \qquad G(t) = (1 - R_k(T_e))G_0 \exp(-(t/\tau_L)^2)$$

where σ - is the Stefan- Boltzmann constant.

At the shock wave in solid, $x = \Gamma_{sh,s}(t)$, we write Rankine-Hugoniot conditions [9] .

3. COMPUTATIONAL ALGORITHM AND METHOD OF DYNAMIC ADAPTATION

The main computational feature of the considered problem is the existence of discontinuous solutions, moving interfacial and contact boundaries. For the numerical solution of the mathematical model (1)-(3), we use the finite-difference method of dynamic adaptation [10]. The method is based on the procedure of transition to an arbitrary nonstationary coordinate system. This transition is made through an automatic conversion of coordinates using the sought solution, which allows formulating the problem of construction and adaptation of the grid at the differential level, i.e. in the resulting mathematical model, one part of differential equations describes the physical processes and the other - the behavior of grid. The method of dynamic adaptation allows to automate the problem of creation of the new domains and perform calculations with explicit tracking of any number of discontinuities, interfacial and contact boundaries. The method of dynamic adaptation and the used finite difference schemes are described in detail in the earlier papers [10-12]. The total number of nodes of the computational grid was set to 200 in the initial target. Additionally, each of the spalled fragments contained approximately 30 nodes in the average.

4. ALGORITHM OF COMPUTATION AND CONSTRUCTION OF COMPUTATIONAL GRIDS

The compression wave, generated due to heating and thermal expansion of the target material, is followed by the rarefaction wave moving from the surface into the interior of the target. If the pressure in the rarefaction wave reaches a sufficiently large negative value, a mechanical spallation may occur, i.e. a formation of voids in the target material. As a result of the spallation, pieces of matter are splitted from the target and move in the direction opposite to the direction of the laser pulse. The spallation is introduced as follows: at the point where the criterion of spallation is met, i.e. at a certain value of the pressure, two new moving boundaries are introduced with saturated vapor between them and boundary conditions describing the kinetics of the surface evaporation and condensation. In the regimes when the temperature in the point of mechanical spallation is not high enough ($T < T_b$), it is possible to replace the saturated vapor with vacuum. That is made for simplicity and without significant loss in accuracy of calculations. Then at these two new boundaries, we use the boundary conditions for the contact discontinuity of condensed matter - vacuum. In addition to the criterion of spallation, we also introduce an additional condition: the minimum size of the spalled material should not be less than several atomic layers, namely 1 nm. A detailed computational grid is automatically generated in the new domain. The nodes of the grid are concentrated on both boundaries according to the law of geometric progression. The minimum spatial step in the spalled domain was set to be 0.01 nm. If the minimum step in moving boundaries was set larger, for example, 0.1 nm, it was not enough to maintain the accuracy in the calculation of the boundary conditions: the amplitude of the periodic oscillations of the gas-dynamic velocity was slowly gradually growing upon reflection from the moving boundaries, while with the sufficient accuracy the amplitude should either not change or decrease due to numerical viscosity. These fluctuations are shown in Fig.7.

When multiple spallations occur, the speed of different spalled parts may be different, and one part may catch up with another. In this case, an algorithm of collapsing of the voids and uniting of two fragments into one was implemented at a certain small distance between the individual phases. This distance was chosen to be 0.4 nm.

The expressions for the spallation criterion were taken from works of Grady [13] and Povarnitsyn [14]. The equation of state used in paper is a modification of EOS from the work of Lomonosov [15].

4.1 Volume melting

The paper of Mei and Lu [16] provides a review of different estimates for the maximum overheating of the solid phase after which volume (homogenous) melting starts. These estimates give the maximum value in the range of 1.1 - 1.38 of the melting temperature. Based on this, we have chosen the following algorithm of consideration of volume melting. When the temperature reaches $1.4 T_m$ in some point, $T_m = T_m(p_s)$, then the whole region of solid from the first point where the temperature exceeds T_m is declared to be liquid and the surface melting boundary is moved to that point. The starting temperature of this region is recalculated via the latent heat with the account of the energy conservation law.
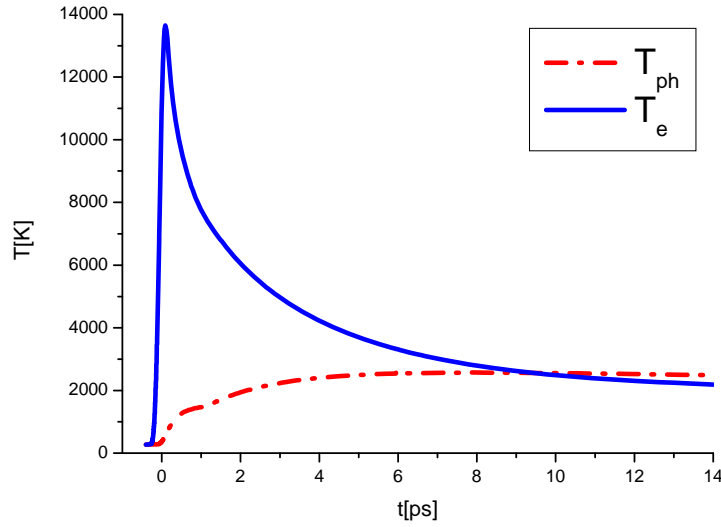
Figure 1. Time dependence of electron and phonon temperature at the target surface.

4. ANALYSIS OF THE RESULTS

Action of USLP on the surface of Al target is considered. The laser pulse with the wavelength $\lambda_L = 0.8 \ \mu m$ has Gaussian time profile, $G = G_0 \exp(-(t/\tau)^2)$, with $\tau = 0.1$ ps and maximum intensity $G_0 = 1.52 \times 10^{12}$ W/cm^2, corresponding to the fluence $F = 0.27 \, \text{J/cm}^2$. The absorbed fluence is approximately $F_{abs} \approx 0.06 \text{J/cm}^2$. The computation starts at $t = -4\tau$. Temperature dependence of the surface reflectivity $R(T_e)$ and the volume absorption coefficient $\alpha(T_e)$ were determined through the longitudinal dielectric permeability, which is determined by solving the kinetic equation [17] and with 5% error are approximated by the following analytical expressions:

$$R(T_e) = 0.7845465 - 0.0048568 T_e, \quad \alpha(T_e) = 100\exp(13.65497 - 0.026894 T_e + 1.66510^{-4} T_e^{\ 2}).$$

In these expressions, T_e is measured in electron-volts.

The laser fluence was chosen to be slightly above the spallation threshold, which for our computations was determined to be $F_{th} \approx 0.055 \text{ J/cm}^2$.

The evolution of the processes in the target is convenient to represent in the form of two consecutive stages: thermodynamically non-equilibrium with $T_e \gg T_{ph}$, Fig. 1,2 and hydro-dynamical with $T_e \sim T_{ph}$, Fig. 4–9. Melting of the target starts from the irradiated surface with a large delay relative to the laser pulse, $t \sim +0.2\text{ps}$. Significant overheating of the melting surface and high spatial temperature gradients provide high speed of the melting front propagation, $\upsilon_{sl} \sim 2.3\text{km/s}$, Fig.3. In this paper, the maximum value of overheating of the

solid phase was limited by the value of $1.4 T_m$. The decomposition of the overheated metastable state occurs in the form of a volume melting of the overheated region.

The later second stage of the processes $t \geq 0.2$ ps is associated with the evolution of hydrodynamic processes. High speed of the melting front is one of the causes of the appearance of high pressure region in the solid phase, $p_s \sim \rho_s \upsilon_{sl}^2 \approx 20$kbar, which leads to the formation of the strong compression in front of the melting boundary. Another cause of the compression wave is thermal expansion. Propagation of the compression wave has a loading effect on the substance and corresponds to a positive pressure half-wave. The compression stage is followed by an unloading stage, which corresponds to the negative pressure half-wave.
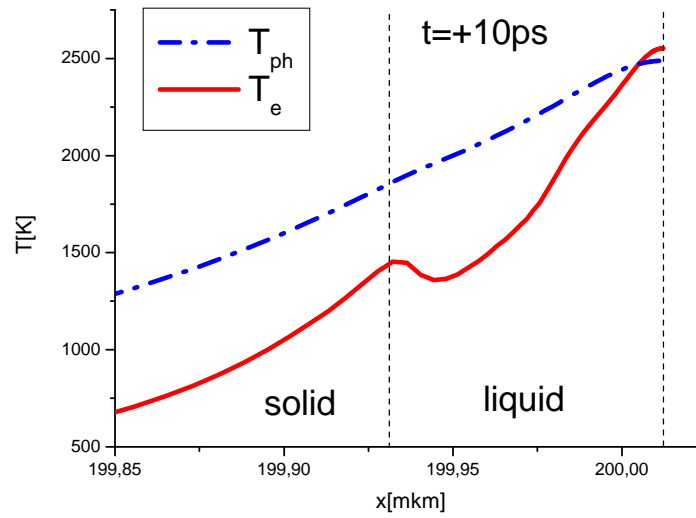

Figure 2. Spatial profile of electron and phonon temperature at $t = +10$ps.

Fig. 4 shows the pressure profiles before the first spallation and right after that. Fig. 5 and 6 show the density profiles and electron and phonon temperatures at $t = +0.4$ ns. There are 9 spalled parts seen at the figures with sizes from 2 to 19 nm. There were 20 spallation events in this computation, some of the spalled parts joined and formed larger fragments. The total size of the spalled material, converting via the density of solid, is 78 nm which is in good agreement with the values reported from MD simulation [18]. The temperature of the spalled parts is in the range from 1000K to 2100K. Repeated passage of the pressure waves, reflected from the left and right free boundaries, is observed in each of the spalled parts. These waves are most noticeable in the parts with the maximum sizes, in this modeling these are the most left and most right part with the sizes 45 nm and 35 nm correspondingly, where the pressure amplitude reaches the value of 1-2GPa. Fig. 7 shows the time dependence of hydro-dynamic velocity (black curve) at the right free boundary of the target (bulk liquid or solid), at the left boundary of the most left spalled part (red curve), the closest to the bulk target, and at the right boundary of the most right fragment, the farthest from the bulk target. The frequency of the velocity oscillations is different in different parts due to the different sizes of the parts.

Also, there are several velocity peaks during one period. That is caused by the fact that these fragments were created as a result of collision of several smaller parts which already had own oscillations with own frequency. The average movement velocity is about 200m/s for the left part and 500 m/s for the right part. It should be noted that in this modeling, almost entire liquid is spalled after the passage of the rarefaction wave, and the remaining 6 nm of liquid are already crystallized at the moment +0.4ns. Figure 7 shows the amount of liquid in bulk and in the spalled parts. It is seen that the whole liquid is spalled except the mentioned 6 nm. Figure 3 shows the velocity of the melting front. The maximum value of velocity reaches the value of 2300 m/s. At the moment +45ps, melting ceases completely and crystallization starts. The maximum crystallization velocity is 58 m/s.
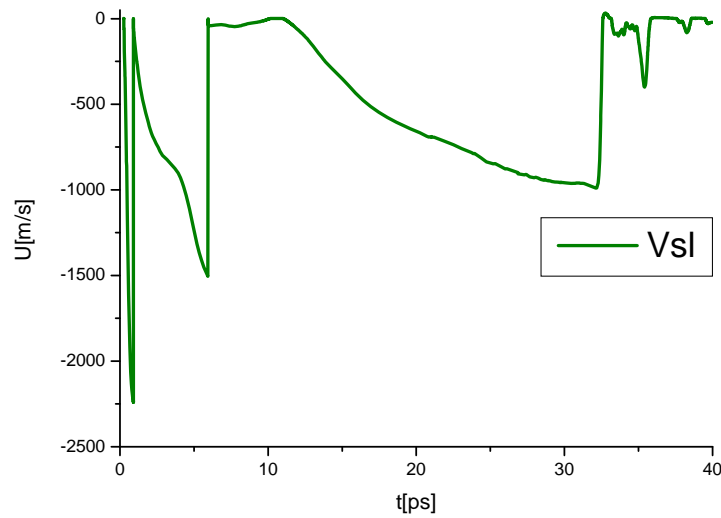
Figure 3. Melting front velocity.

5. CONCLUSION

Algorithm of solution of the problem of laser ablation in a multiply-connected domain area using the dynamic adaptation method was developed. This algorithm was used to investigate the mechanism of fragmentation of a massive metal target by ultrashort laser pulses. A good agreement with molecular dynamic simulation for the spallation threshold and the amount of the fragmented material is obtained.

ACKNOWLEDGMENT

Figure 4. Pressure profiles before and after first spallation.
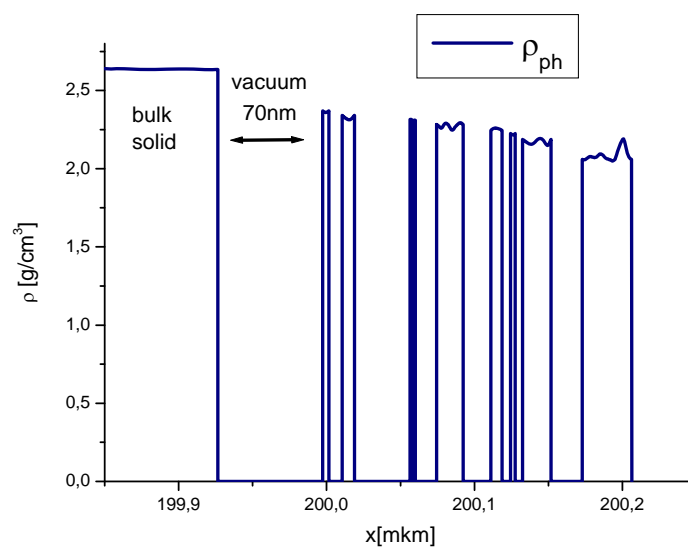
Figure 5. Density profile at t= +0.4ns
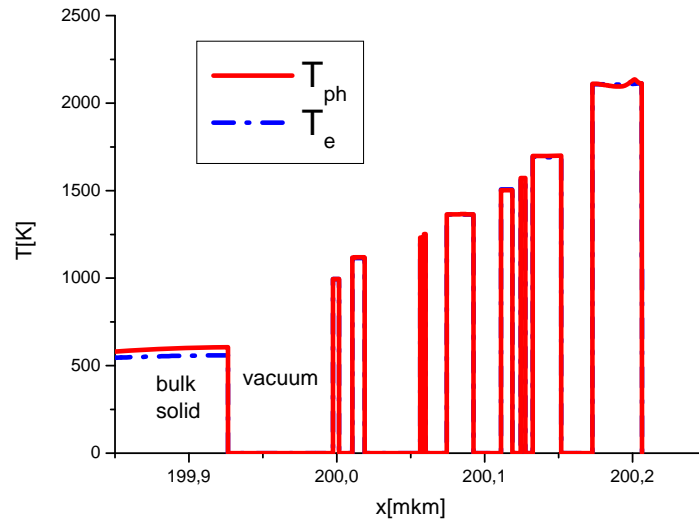
Figure 6. Electron and phonon temperature at t= +0.4ns
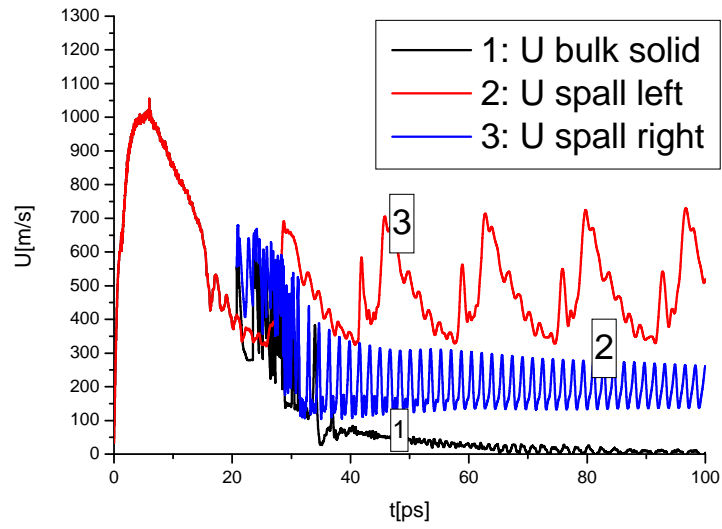
Figure 7. Hydro-dynamic velocity at the very right boundary of bulk phase, right boundary of the right spalled part and left boundary of the left spalled part.
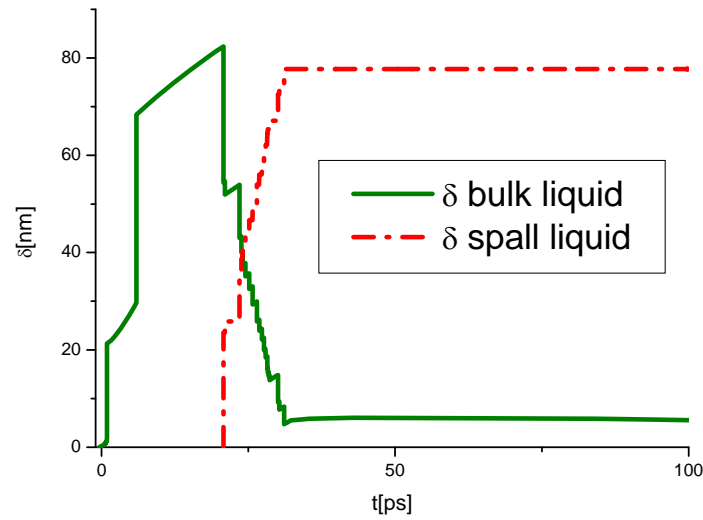
Figure 8. Amount of material in bulk liquid and spalled liquid.
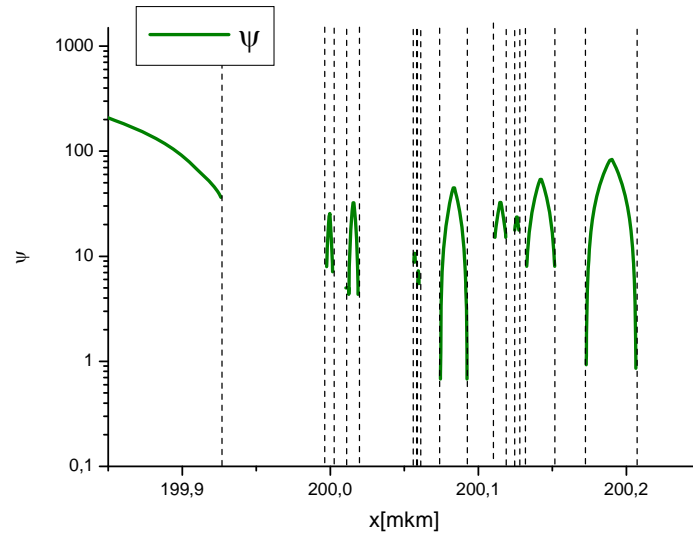
Figure 9. Non-dimensional grid step.

REFERENCES

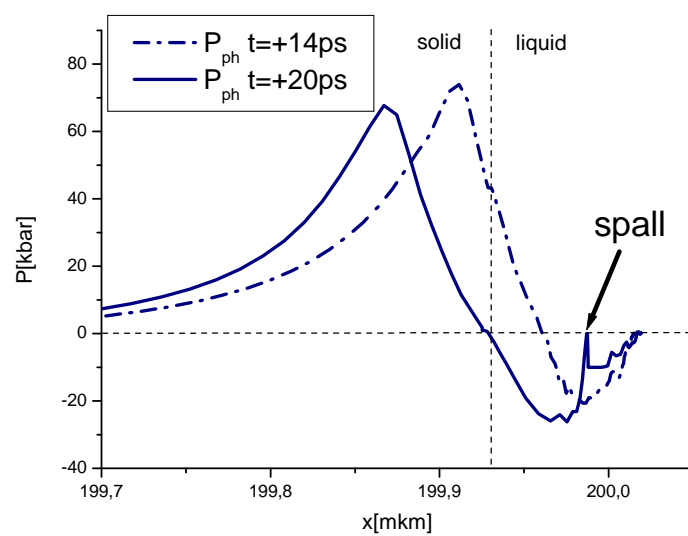[1] Davison, L., Grady, D. E. and Shahinpoor, M. *High Pressure Shock Compression of Solids II: Dynamic Fracture and Fragmentation.* Springer, New York, 1996.

[2] Antoun, T., Seaman, L., Curran, D. R., Kanel, G. I., Razorenov, S. V. and Utkin, A. V. *Spall Fracture.* Springer, Berlin, 2002.

[3] Ivanov, D. S. and Zhigilei, L. V. Combined atomistic-continuum modeling of short-pulse laser melting and disintegration of metal films. *Phys. Rev. B* (2003) **68**:064114

[4] Perez, D. and Lewis, L. J. Molecular-dynamics study of ablation of solids under femtosecond laser pulses. *Phys. Rev. B* (2003) **67**: 184102

[5] Stegailov, V. V., Starikov, S. V. and Norman, G. E. Atomistic simulation of laser ablation of gold: The effect of electronic pressure. *AIP Conf. Proc.* (2012) **1426**: 905;

[6] Mazhukin, V.I., Mazhukin, A.V., Demin, M.M. and Shapranov, A.V. - Mathematical modeling of short and ultrashort laser action on metals. *Surface Modification Technologies (SMT 24)*. Eds. T.S.Sudarshan, Eckhard Beyer, Lutz-Michael Berger, (2011), **24**: 201-208. (ISBN 978-81-910571-2-6).

[7] Mazhukin, V.I. Kinetics and Dynamics of Phase Transformations in Metals under action of ultra-short high-power laser pulses. Ch. 8, pp.219 - 276. In "*Laser Pulses – Theory, Technology, and Applications*", Ed. by I. Peshko. 2012, P. 544, InTech, Croatia.

[8] Mazhukin, V.I. and Samokhin, A.A. Boundary conditions for gas-dynamical modeling of evaporation processes. *Mathematica Montisnigri*, 2012, vol. XXIV, pp.8 - 17.

[9] Landau L. D., Lifshic E. M. Teoreticheskaja fizika, t. V. Statisticheskaja fizika. Chast' 1. (1976) M.: Nauka. 616 p.

[10] Mazhukin, A.V., Mazhukin, V.I. and Breslavskii, P.V. Dynamic adaptation in gas dynamics problems with non-linear heat conductivity. *Proceedings of the V International Conference on Adaptive Modeling and Simulation (ADMOS 2011)*, Eds. D.Aurby, P.Diez, N.Pares. Paris, France. (2011) 105 - 116.

[11] Mazhukin, V.I., Mazhukin, A.V. and Shapranov, A.V. Dynamic adaptation in differential equations of the parabolic type in partial derivatives. *Encyclopedia of low-temperature plasma*, Series B, Vol. VII –1, Mathematical modeling in low-temperature plasma, Part 1, pp. 190 – 216, Moskva, Yanus-K (2008).

[12] Mazhukin, V.I., Mazhukin, A.V. and Shapranov, A.V. Dynamic adaptation in differential equations of the hyperbolic type in partial derivatives. *Encyclopedia of low-temperature plasma*, Series B, Vol. VII –1, Mathematical modeling in low-temperature plasma, Part 1, pp. 217 - 247, Moskva, Yanus-K (2008).

[13] Povarnitsyn, M. E., Itina, T. E., Sentis, M., Khishchenko, K. V. and Levashov P. R. Material decomposition mechanisms in femtosecond laser interactions with metals (2007) Phys. Rev. B. **75**:235414.

[14] Grady, D. E. The spall strength of condensed matter. *J. Mech. Phys. Solids* (1988) **36**: 353-384.

[15] Lomonosov, I.V. Multi-phase equation of state for aluminum. *Laser and Particle Beams* (2007) **25**:567–584.

[16] Mei, Q.S. and Lu, K. Melting and superheating of crystalline solids: From bulk to nanocrystals. *Progress in Materials Science* (2007) **52**:1175–1262.

[17] Mazhukin V.I., Mazhukin A.V., Koroleva O.N. Optical properties of electron Fermi-gas of metals at arbitrary temperature and frequency. *Laser Physics*, (2009) **19**:1179-1186.

[18] Zhakhovskii, V, Inogamov, N and Nishihara, K. Laser ablation and spallation of crystalline aluminum simulated by Molecular Dynamics. *Journal of Physics: Conference Series* (2008) **112**:042080

--

CONSTRUCTION OF DYNAMICALLY ADAPTING COMPUTATIONAL GRIDS IN SYSTEMS OF DIFFERENTIAL EQUATIONS DESCRIBING THE NON-EQUILIBRIUM PROCESSES

ALEXANDER V. MAZHUKIN[*], OLGA N. KOROLEVA[†]

[*] Keldysh Institute of Applied Mathematics
125047 Miusskaya sq. 4 Moscow, Russia
e-mail: vim@modhef.ru

[†] Moscow University for the Humanities
111395 Yunosti st. 5 Moscow, Russia
email: koroleva.on@mail.ru

Key words: Modeling, Adaptive grid, Dynamic adaptation, Systems of differential equations, Laser irradiation.

Abstract. The processes occurring in solid targets (metals, semiconductors) initiated by pulsed flows of condensed energy is described by two-temperature model (TTM). The simplest TTM model for metals consists of two, and for semiconductors of three non-linear parabolic equations. Ultrafast impact (pico-femtosecond pulse duration) predetermines the appearance of large solution gradients that require in numerical solution application of computational grids with dynamic adaptation. Transition to an arbitrary non-stationary system of coordinates, the velocity of which is unknown and depends on the desired solution is the basis of the construction of a dynamically adaptive grids. Velocity of the system of coordinates for the numerical discretization is used as a function that control the motion of grid nodes . Agreed change of movement of grid nodes with the solution is achieved by constructing of transformation function derived from the principle of quasi-stationarity.

Simulation of some specific regimes of pulsed heating and melting of semiconductor silicon (Si), using a numerical grid with the controlled distribution of nodes was carried out.

1   INTRODUCTION

Pulsed laser radiation is a widely used tool for precision machining of materials, including semiconductors. Among semiconductor materials, silicon was most widespread in the instrument-making and is one of the most promising materials for thin-film nanotechnology.

To optimize existing and develop new technologies of laser surface treatment of semiconductors it is necessary to perform a detailed study of the dynamics of processes occurring in the irradiation zone and leading to surface modification, including an analysis of the processes of heating, melting and evaporation.

In this paper, we use the methods of mathematical modeling to investigate the action of laser pulse (picosecond) with a wave length of $\lambda_L = 0.53\mu m$ and the photon energy exceeding the band gap of silicon target $\hbar\omega > E_g$. The main feature of these regimes of laser action is

highly non-equilibrium of heating and melting processes, which results in a large difference between the carriers temperature and the lattice temperature. The numerical solution of system of partial differential equations of parabolic type was carried out by means of the dynamic adaptation method. The use of arbitrary nonstationary system of coordinates allows to construct grids with a constant number of nodes in regions with moving boundaries and allows to concentrate grid nodes in regions of large gradients of solutions.

Laser radiation with intensity $G(t)$, Gaussian distribution and wavelength λ_L extending from left to right (Fig. 1) falls on silicon target surface, where the part of the radiation is reflected and some is absorbed. The released energy of laser pulse causes heating, melting (moving boundary Γ_{sl} - melting front) and evaporation (moving boundary Γ_{lv} - evaporation front).

2 MODEL

The mathematical model consists of transport equations of the laser radiation, which takes into account the temperature dependence of the reflectivity of the surface, the carrier balance equation that takes into account generation (photo-ionization) and recombination of charged particles (Auger recombination, and photo-recombination), the balance equations of energy carriers and the lattice, taking into account the absorption of laser energy, the exchange of energy between the electron and phonon subsystems, heat and mass transfer [1-2].
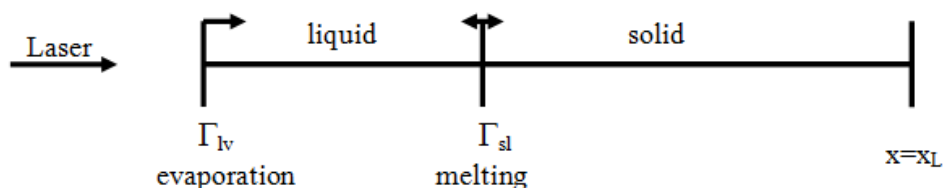
Figure 1. Scheme of laser irradiation.

The basis of first-order phase transitions is the mechanism of heterogeneous melting and evaporation. The process of melting - crystallization is described in approximation of classical variant of Stefan problem and the process of evaporation is described in approximation of Knudsen layer (single-phase version of Stefan problem).

Semiconductor has the properties of metal after melting temperature is reached. And therefore it is necessary to write equations for the solid and liquid regions.

Equations for solid region:

$$\frac{\partial N}{\partial t} = -\frac{\partial J}{\partial x} + I_{en} - R_{en},$$ (1)

$$\frac{\partial \varepsilon_e}{\partial t} = -\frac{\partial W_e}{\partial x} - \frac{\partial G}{\partial x} - g(T_e)_{lat}(T_e - T_{lat}),$$ (2)

$$\frac{\partial \varepsilon_{lat}}{\partial t} = -\frac{\partial W_{lat}}{\partial x} + g(T_e)_{lat}(T_e - T_{lat}),$$ (3)

$$\frac{\partial G}{\partial x} = -(\alpha + \beta_1)G - \beta_2 G^2,$$ (4)

$$\Gamma_{sl} < x < x_L$$

Equation (1) is equation for a concentration. Equations (2), (3) are energy equations for electron component and lattice. Equation (4) is equation of laser energy transfer.

Equations for liquid region:

$$\frac{\partial \varepsilon_e}{\partial t} = \frac{\partial}{\partial x}\left(\lambda_e(T_e, T_l)\frac{\partial T_e}{\partial x}\right) - g(T_e)(T_e - T_l) - \frac{\partial G}{\partial x}$$ (5)

$$\frac{\partial \varepsilon_l}{\partial t} = \frac{\partial}{\partial x}\left(\lambda_l(T_l)\frac{\partial T_l}{\partial x}\right) + g(T_e)(T_e - T_l)$$ (6)

$$\frac{\partial G}{\partial x} = -\alpha G, \qquad G(t) = G_0 \exp\left(-\left(\frac{t}{\tau}\right)^2\right)$$ (7)

$$\Gamma_{l\upsilon} < x < \Gamma_{sl}$$

Where N – carrier concentration, J – particle current density, ε_e и ε_{lat} – internal energy of electron gas and lattice, W_e и W_{lat} – heat flow of electron gas and lattice, x_L – right end of the sample, λ_e и λ_l – heat conductivity coefficient of electron gas and liquid metal, $I_{en} = \frac{\beta_1 G}{\hbar\omega} + \frac{\beta_2 G^2}{2\hbar\omega} + k_i(T_e)N$ and $R_{en} = \gamma N^3$ – electron-hole pairs generation and recombination velocities, γ - Auger recombination coefficient and ω - laser irradiation frequency, α – free-carrier absorption coefficient, β_1 and β_2 - coefficients of one and two photon absorption, k_i - collision ionization coefficient, $g(T_e)$, $g(T_e)_{lat}$ – electron-lattice energy exchange factor for metal and semiconductor, $g(T_e)_{lat} = C_e/\tau_E$, τ_E - energy relaxation time, C_e – heat capacity. $g(T_e)$ and other thermophysical properties of metals reported in [4].

Equations (5), (6) are energy equations for electron component and lattice one. Equation (7) is equation of laser energy transfer.

Boundary conditions:

$x = x_L$: $\qquad J = 0, \ W_e = W_{lat} = 0$

$x = \Gamma_{sl}(t)$: for electron component: $\qquad (W_e)_s = (W_e)_l, \ (T_e)_s = (T_e)_l,$

for lattice: $\qquad T_s = T_l = T_m, \ \left(\lambda_{lat}\frac{\partial T_{lat}}{\partial x}\right)_s - \left(\lambda_{lat}\frac{\partial T_{lat}}{\partial x}\right)_l = \rho_s L_m \upsilon_{sl},$

$x = \Gamma_{lv}(t)$: for electron component: $\quad J = 0, \ W_e = 0, \ G_l(t) = AG(t)$

for lattice:

$$\lambda \frac{\partial T_l}{\partial x} = \rho_l L_v \upsilon_{l\upsilon}, \qquad \begin{aligned} \rho_l \upsilon_{l\upsilon} &= \rho_\upsilon (\upsilon_{l\upsilon} - u) \\ P_l + \rho_l \upsilon_{l\upsilon}^2 &= P_\upsilon + \rho_\upsilon (\upsilon_{l\upsilon} - u)^2 \end{aligned}$$

$$T_\upsilon = 0.633 T_l, \quad \rho_\upsilon = 0.326 \rho_{sat}, \quad u = (\gamma R T_\upsilon)^{1/2},$$

$$\rho_{sat} = \frac{P_{sat}}{RT_l}, \quad P_{sat} = p_b \exp\left[\frac{L_v}{RT_b}\left(1 - \frac{T_b}{T_l}\right)\right]$$

Indexes *lat, l, v, s, sat, b* mean values affiliation to semiconductor lattice, metal liquid phase, vapor, solid phase, saturated vapor and boiling under normal conditions.

3 METHOD

The method of dynamic adaptation is based on a transition to an arbitrary non-stationary coordinate system. The usage of an arbitrary non-stationary coordinate system allows to formulate the problem of the grid generation and adaptation at the differential level, i.e. in the resulting mathematical model, one part of the differential equations describes the physical processes and the other part – the behavior of the nodes of the grid [5]. The transition to an arbitrary non-stationary coordinate system is performed using an automated coordinate transformation via the sought solution.

According to the papers [5] – [7], we will perform a transition from the physical space $\Omega_{x,t}$ with Euler variables (x,t) to some computational space with an arbitrary non-stationary coordinate system $\Omega_{q,\tau}$ with variables (q,τ). This transformation can be performed using a substitution of variables of a common form $x = f(q,\tau), \quad t = \tau$, with a single-valued non-degenerate reverse transformation $q = \varphi(x,t), \quad \tau = t$.

During the transition from one coordinate system to another, the partial derivatives of the dependent variables are connected via the following expressions:

$$\frac{\partial}{\partial t} = \frac{\partial}{\partial \tau} + \frac{Q}{\psi}\frac{\partial}{\partial q}; \quad \frac{\partial}{\partial x} = \frac{1}{\psi}\frac{\partial}{\partial q}; \quad \frac{\partial^2}{\partial x^2} = \frac{1}{\psi}\frac{\partial}{\partial q}\frac{1}{\psi}\frac{\partial}{\partial q} \tag{8}$$

where $\psi = \partial x / \partial q$ - is the Jacobian of the reverse transformation.

Using a replacement of variables of the common form and expressions (8), we can write the differential model (1)–(4) and (5)–(7) in the variables (q,τ):

$$\frac{\partial(\psi N)}{\partial \tau} = -\frac{\partial(Q_s N)}{\partial q} - \frac{\partial J}{\partial q} - \psi (R_{en} - I_{en}) \tag{9}$$

$$\frac{\partial(\psi \varepsilon_{\ell at})}{\partial \tau} = -\frac{\partial(Q_s \varepsilon_{\ell at})}{\partial q} + \psi q_e(T_e)(T_e - T_{\ell at}) - \frac{\partial W_{\ell at}}{\partial q} \tag{10}$$

$$\frac{\partial(\psi \varepsilon_e)}{\partial \tau} = -\frac{\partial(Q_s \varepsilon_e)}{\partial q} - \frac{\partial G}{\partial q} - \psi q_e(T_e)(T_e - T_{eat}) - \frac{\partial W_e}{\partial q} \tag{11}$$

$$\frac{\partial G}{\partial q} = -\psi(\alpha + \beta_1)G - \psi \beta_2 G^2 \tag{12}$$

$$\frac{\partial \psi}{\partial \tau} = -\frac{\partial Q_s}{\partial q} \tag{13}$$

$$\Gamma_{sl} < x < x_L, \qquad \tau > 0$$

$$\frac{\partial(\psi\varepsilon_e)}{\partial\tau} = -\frac{\partial(Q_\ell\varepsilon_e)}{\partial q} - \frac{\partial W_e}{\partial q} - \frac{\partial G}{\partial q} - \psi g(T_e)(T_e - T_\ell) \tag{14}$$

$$\frac{\partial(\psi\varepsilon_\ell)}{\partial\tau} = -\frac{\partial(Q_\ell\varepsilon_\ell)}{\partial q} - \frac{\partial W_\ell}{\partial q} + \psi g(T_e)(T_e - T_\ell) \tag{15}$$

$$\frac{\partial G}{\partial q} = -\psi\alpha G \tag{16}$$

$$\frac{\partial\psi}{\partial\tau} = -\frac{\partial Q_\ell}{\partial q} \tag{17}$$

$$\Gamma_{\ell\upsilon} < x < \Gamma_{s\ell}, \qquad \tau > 0$$

where (13), (17) – are the equations of the reverse transformation with the transformation functions Q_s, Q_ℓ. The functions Q_s, Q_ℓ characterize the speed of the non-stationary coordinate system in the solid and liquid phases accordingly and are not predefined and should be determined.

Thus, during the transition to an arbitrary non-stationary coordinate system, the initial differential models are transformed to the extended differential systems with additional equations (13) and (17). Their type, properties and form of the boundary conditions depend on the particular form of the functions Q [7]. At this stage of discussion, the functions Q_s, Q_ℓ are not defined yet. After their determination, the equations (13) and (17) are used for construction of the grids that adapt to the gradients of solution and to the moving domain boundaries. Their differential analogues describe the dynamics of the grid nodes and the functions Q_s, Q_ℓ perform the controlled motion of the grid nodes in an agreement with the dynamics of the sought solution. The agreement is achieved by introduction of a functional dependency of the function Q on the sought solution. But since the solution is not known beforehand, a problem arises with the determination of the optimum transformation function Q that will provide a complete matching of the adaptation mechanism with the solution. If there are no complete matching, fitting coefficients are inserted into the controlling function. By the adjustment of the fitting coefficients, it is possible to make the degree of the mismatching lower. At the same time, the fact of the presence of the fitting coefficients in an adaptation method is an evidence of its imperfection.

Such matching can be obtained using the quasi-stationary principle [1], [8], which states that it is necessary to switch to a such coordinate system, where the time derivatives will be small or satisfy the relation: $\partial N/\partial\tau = \partial\varepsilon_e/\partial\tau = \partial\varepsilon_{\ell at}/\partial\tau = \partial\varepsilon_\ell/\partial\tau = 0$. Then the transformation functions will take the form:

$$Q_s = -\frac{\dfrac{\partial W_e}{\partial q} + \dfrac{\partial W_{\ell at}}{\partial q} + \dfrac{\partial G}{\partial q} + \dfrac{\partial J}{\partial q} + \psi(R_{en} - I_{en})}{\left|\dfrac{\partial N}{\partial q}\right| + \left|\dfrac{\partial\varepsilon_e}{\partial q}\right| + \left|\dfrac{\partial\varepsilon_{\ell at}}{\partial q}\right|} \qquad\qquad Q_\ell = -\frac{\dfrac{\partial W_e}{\partial q} + \dfrac{\partial W_\ell}{\partial q} + \dfrac{\partial G}{\partial q}}{\left|\dfrac{\partial\varepsilon_e}{\partial q}\right| + \left|\dfrac{\partial\varepsilon_\ell}{\partial q}\right|}$$

The usage of adapting grids allowed to perform all computations of grids with total number of nodes less than 100.

5   RESULTS

Regimes of irradiation with a Gaussian intensity distribution in the pulse $G(t) = G_0 exp\left(-(t/\tau)^2\right)$ were considered. Pulse duration $\tau_L = 10$ ps. The maximum value of the intensity varied from $G_0 = 3 \times 10^9$ to 5×10^{10} W/cm^2. Figures 2 and 3 show the time profiles of temperature and radiation at intensity $G_0 = 3 \times 10^9$ W/cm^2. It can be seen that at such intensity the gap between the electron temperature and the temperature of the lattice is clearly seen, but the melting does not occur yet.

Figure 2. Time dependences of incident G (black dotted curve) and absorbed part AG (red solid curve) of laser radiation, $G_0 = 3 \times 10^9$ W/cm^2.

Increasing the intensity by one order of magnitude leads to the melting of silicon, which starts at the back front of the laser pulse. Typical time profiles of the laser radiation, surface temperatures, melting front velocity and (non)-equilibrium carrier concentrations on the surface for 3×10^{10} W/cm^2 energy pulse are shown in Fig. 4-7. Since the melting of lattice starts at the back front of the pulse, the maximum velocity υ_{sl} reaches relatively low value of ~ 27 m/s.

Figure 3. Time dependences of electron T_e (black solid curve) and lattice (red dotted curve) temperatures.
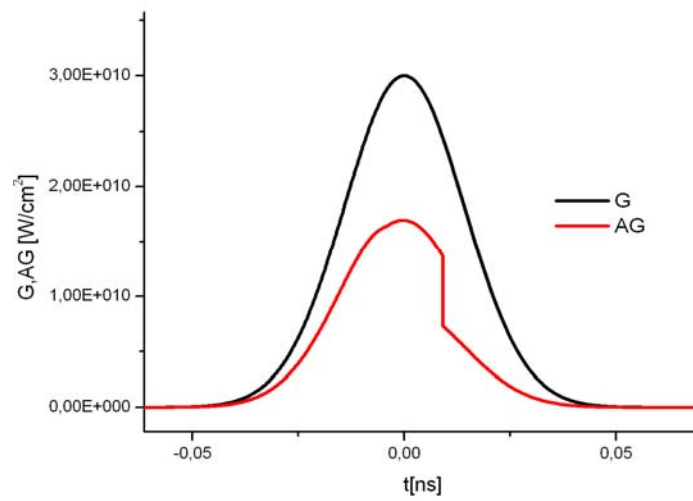
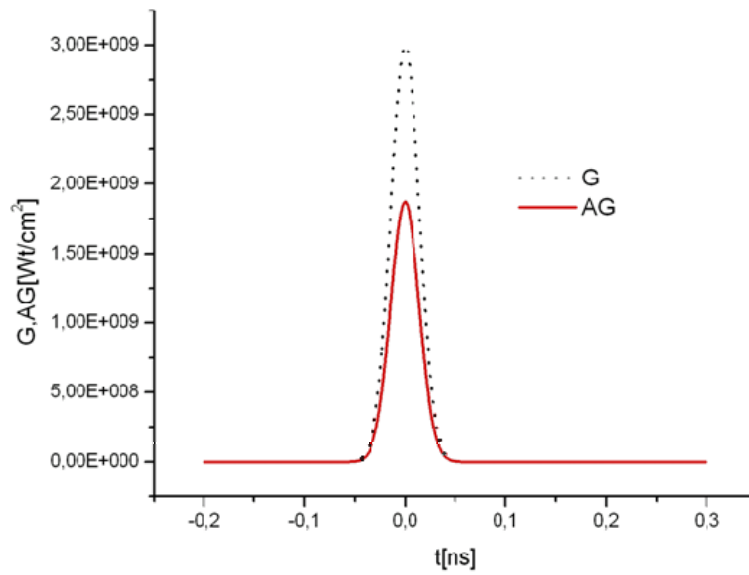**Figure 4**. Time dependences of incident G (black curve) and absorbed part AG (red curve) of laser radiation intensity, $G_0 = 3 \times 10^{10}$ W/cm^2.
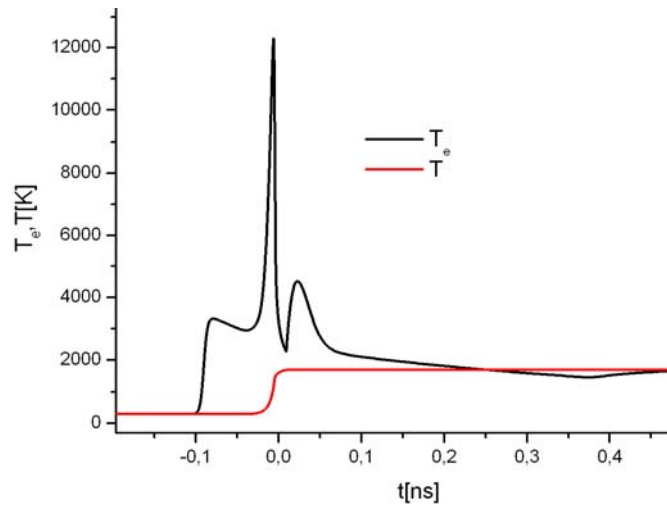
Figure 5. Time dependences of electron T_e (black curve) and lattice (red curve) temperatures.
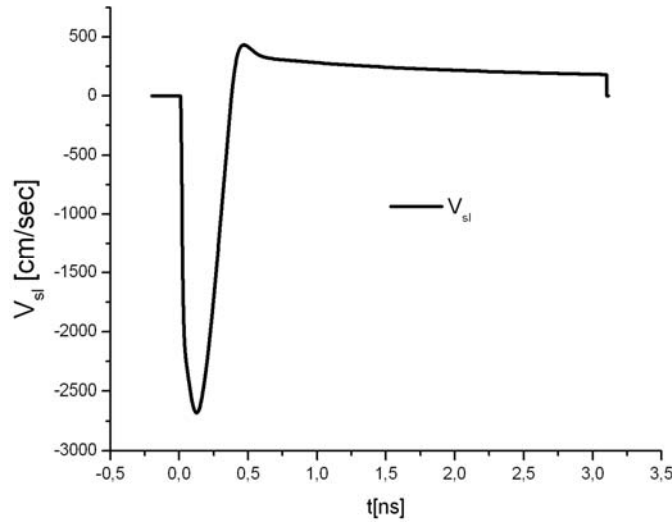
Figure 6. Time dependence of the melting velocity υ_{sl}.

As seen from the time dependences of temperatures (Fig.5 and Fig.9) throughout pulse duration there is a noticeable gap between the phonon and electron temperatures, which reaches 12000K at the peak of the pulse. By the end of the pulse the phonon and electron temperatures become equal.

Maximum melting front velocity reaches 27 m/s for 3×10^{10} W/cm^2 and 225 m/s for 5×10^{10} W/cm^2.

The presence of two peaks in the electron temperature in all regimes of irradiation should be noted, that indicates a change of the mechanism of absorption (photoprocesses replaced by inverse bremsstrahlung).

Figure 7. Time dependences of nonequilibrium N (black curve) and equilibrium N_0 (red curve) carrier concentrations on the surface.

A small change in the intensity of the radiation to 5×10^{10} W/cm^2 (Fig. 8-11) leads to the beginning of melting near the maximum of intensity that provides high value of melting velocity υ_{sl} =225 m/s and the gap between temperatures.
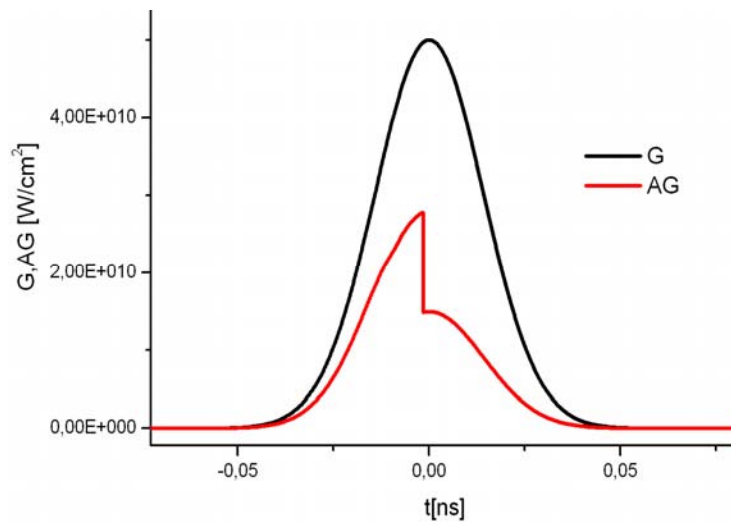
Figure 8. Time dependences of incident G (black curve) and absorbed part AG (red curve) of laser radiation intensity, $G_0 = 5 \times 10^{10}$ W/cm^2.
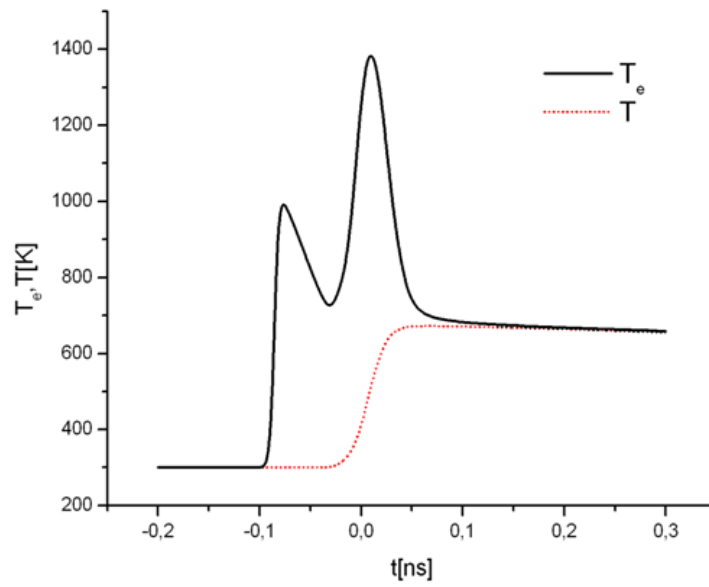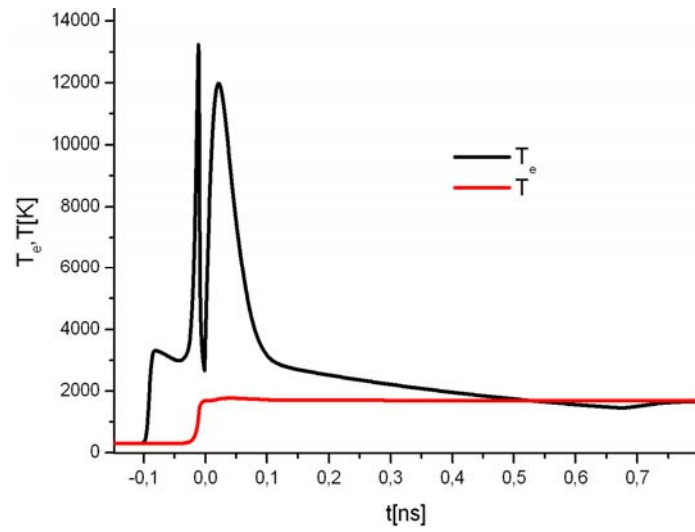
Figure 9. Time dependences of electron T_e (black curve) and lattice (red curve) temperatures.
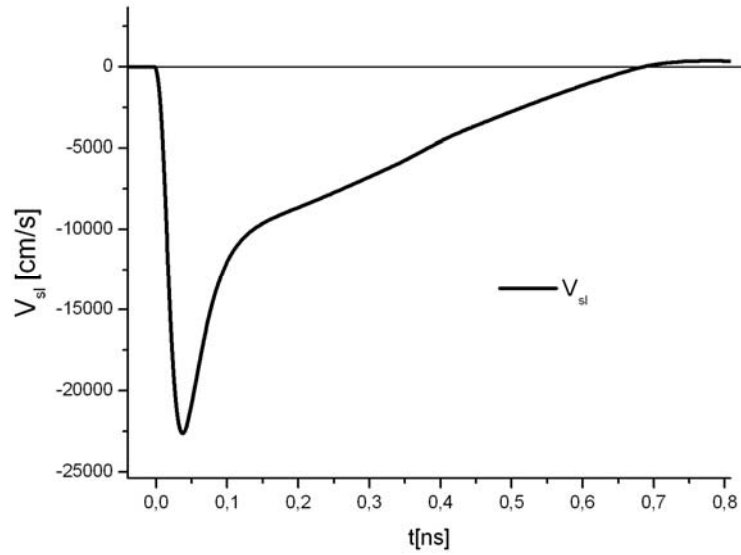
Figure 10. Time dependence of the melting velocity υ_{sl}.
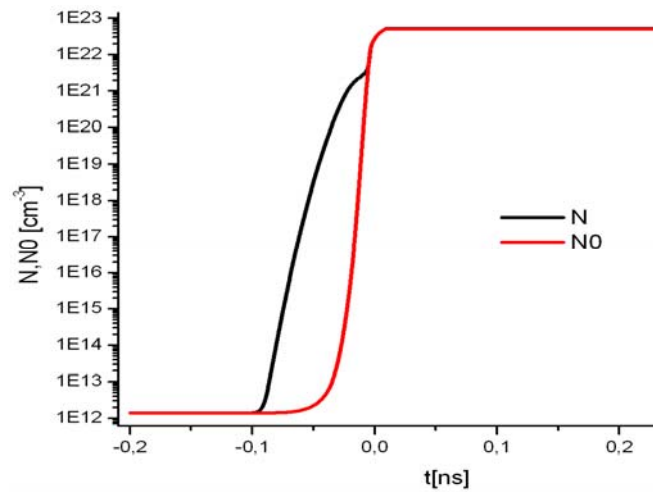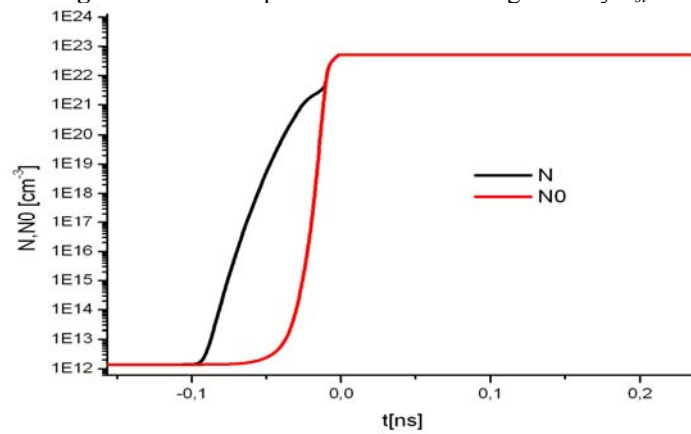

Figure 11. Time dependences of nonequilibrium N (black curve) and equilibrium N_0 (red curve) carrier concentrations on the surface.

4 CONCLUSION

Application of dynamic adaptation method to the numerical solution of the problems of non-equilibrium heating and phase transformations in semiconductor materials was considered. Computational features of these problems are the presence of two moving interphase boundaries and the presence of regions of rapid change of the solution components with different scales in space and time. The use of arbitrary time-dependent system of coordinates allows us to construct computational grids with a constant number of nodes in regions with moving boundaries and concentrate grid points in regions of large gradients of solutions.

Two functions of coordinate transformation by which we make controlled node distribution for nonlinear systems of differential equations of parabolic type were defined.

Numerical solution of two typical regimes of laser irradiation on the crystalline silicon was obtained.

REFERENCES

[1] K.Sokolowski-Tinten, D. von der Linde (2000), "Generation of dense electron hole plasmas in silicon", *Phys. Rev. B*, 61, pp. 2643-2650.

[2] P.Allenspacher, B.Huttner, W.Riede (2003), "Ultrashort pulse damage of Si and Ge semiconductors", *SPIE*, 4932, 2003, pp. 358-365.

[3] V.I.Mazhukin, A.V.Mazhukin (2007), "Dynamic adaptation in parabolic equations", *Computational Mathematics and Mathematical Physics*, vol. 47, 11, pp. 1913-1936.

[4] V.I.Mazhukin, "Kinetics and Dynamics of Phase Transformations in Metals Under Action of Ultra-Short High-Power Laser Pulses", *Laser Pulses – Theory, Technology, and Applications* Ed. by I. Peshko, Chapter 8, 2012, pp-219-276.

[5] V.I.Mazhukin, M.M.Demin, A.V.Shapranov, I.Smurov. The method of construction dynamically adapting grids for problems of unstable laminar combustion. *Numerical Heat Transfer, Part B: Fundamentals,* 2003, vol.44, № 4, pp. 387 - 415.

[6] V.I.Mazhukin, I. Smurov, C. Dupuy, D. Jeandel. Simulation of Laser Induced Melting and Evaporation Processes in Superconducting. *J. Numerical Heat Transfer Part A*, 1994, v. 26, pp. 587-600.

[7] A.V.Mazhukin. Dynamic adaptation in convection — diffusion equations. *Computational methods in applied mathematics*, vol.8(2008), No.2, pp.171–186.

[8] V.I.Mazhukin, A.A.Samarskii, O.Kostelianos, A.V.Shapranov. Dynamic adaptation method for nonstationary problems with large gradients. *Mathematical modeling*, 1993, vol. 5, N 4, pp. 32-56.

VERY HIGH ORDER MESH ADAPTATION

Estelle Mbinky[*]**, Frédéric Alauzet**[*] **AND Adrien Loseille**[*]

[*]INRIA Paris Rocquencourt
e-mail: {Estelle.Mbinky,Frederic.Alauzet,Adrien.Loseille}@inria.fr

Key words: Anisotropic mesh adaptation

Abstract. This paper addresses the construction of anisotropic metrics from higher-order interpolation error in 2 dimensions [2, 3] for mesh adaptation. Our approach is based on homogeneous polynomials that model a local interpolation error. Optimal orientation and ratios are found by using the Sylvester decomposition [4]. Then we apply a global calculus of variation to get the optimal metric field minimizing the $\mathbf{L}^p$ norm of the interpolation error. We illustrate this approach on 3D numerical examples.

REFERENCES

[1] A. Loseille, F. Alauzet : Continuous mesh framework, Part I: well-posed continuous interpolation error and Part II: validations and applications, SIAM in Numerical Analysis, Vol. 49, Issue 1, 2011.

[2] W. Cao : An interpolation error estimate on anisotropic meshes in R^n and optimal metrics for mesh refinement, SIAM in Numerical Analysis, Vol. 45 (2007), no. 6, 2368-2391.

[3] J.-M.Mirebeau : Optimal meshes for finite elements of arbitrary order, Springer Science+Business Media,LLC 2010, 18 February 2010.

[4] P. Comon, B. Mourrain : Decomposition of quantics in sums of powers of linear forms, February 1999.

TIME ADAPTIVITY AND ANISOTROPIC MESH ADAPTATION FOR CFD APPLICATIONS

STEFANO MICHELETTI* AND SIMONA PEROTTO*

*MOX - Modeling and Scientific Computing
Dipartimento di Matematica "F. Brioschi", Politecnico di Milano
Piazza Leonardo da Vinci 32, I-20133, Milano, Italy
e-mail: {stefano.micheletti,simona.perotto}@polimi.it, mox.polimi.it

Key words: Space-time Adaptation, Anisotropic Meshes

Abstract. In this communication we present some recent results about time adaptivity with applications to fluid-dynamics. In particular, we are interested in phenomena characterized by temporal multiscale as well as strong spatial heterogeneities, e.g., heat flow problems, shallow water flows, hydrogeology, particle diffusion phenomena, etc. The proposed adaptation procedure relies on a theoretical tool, i.e., an a posteriori error estimator, driving the automatic choice of both the spatial and temporal meshes. The key point is to identify, in the error estimator, separate space and time contributions, as discussed, e.g., in [1, 2, 3]. Thus, on the one hand, we devise a sound criterion to update the time step, able to follow the evolution of the problem under investigation. On the other hand, we exploit an anisotropic adapted triangular grid. It is in fact well known that, by better orienting the mesh elements according to the main features of the solution, it is possible to maximize the solution accuracy for a fixed number of elements, rather than reduce the number of degrees of freedom for a fixed solution accuracy (see, e.g., [4, 5]). Application to purely diffusive problems was first provided in [6, 7]. Here we extend our approach to nonlinear problems, such as the shallow water system considered in [8, 9].

REFERENCES

[1] Verfürth, R. A posteriori error estimate for finite element discretizations of the heat equation. *Calcolo* (2003) **40**:195–212.

[2] Cascón, J.M., Ferragut, L. and Asensio, M.I. Space-time adaptive algorithm for the mixed parabolic problem. *Numer. Math.* (2006) **103**:367–392.

[3] Schmich, M. and Vexler, B. Adaptivity with dynamic meshes for space-time finite element discretizations of parabolic equations. *SIAM J. Control Optim.* (2008) **30(1)**:369–393.

[4] Micheletti, S. and Perotto, S. Output functional control for nonlinear equations driven by anisotropic mesh adaption: The Navier-Stokes equations. *SIAM J. Sci. Comput.* (2008) **30(6)**:2817–2854.

[5] Farrell, P.E., Micheletti, S. and Perotto, S. An anisotropic Zienkiewicz-Zhu type error estimator for 3D applications. *Int. J. Numer. Meth. Engng* (2011) **85(6)**:671–692.

[6] Micheletti, S. and Perotto, S. Anisotropic mesh adaption for time-dependent problems. *Int. J. Numer. Meth. Fluids* (2008) **58**:1009–1015.

[7] Micheletti, S. and Perotto, S. Space-time adaptation for purely diffusive problems in an anisotropic framework. *Int. J Numer. Anal. Model.* (2010) **7(1)**:125–155.

[8] Porta, G.M., Perotto, S. and Ballio, F. Anisotropic mesh adaptation driven by a recovery-based error estimator for shallow water flow modeling. *Int. J. Numer. Meth. Fluids* (2012) **70(3)**:269–299.

[9] Porta, G.M., Perotto, S. and Ballio, F. A space-time adaptation scheme for unsteady shallow water problems. *Math. Comput. Simulation* (2012) **82(12)**:2929–2950.

GOAL-ORIENTED ERROR ESTIMATION FOR NONLINEAR PARABOLIC EQUATIONS BASED ON THE RECONSTRUCTION OF EQUILIBRATED FLUXES

Igor Mozolevski[*], **Serge Prudhomme**[†]

[*]Federal University of Santa Catarina (UFSC)
Campus Universitário, Trindade, 88040-900 Florianópolis, SC, Brazil
e-mail: igor.e.mozolevski@gmail.com

[†]École Polytechnique de Montréal
C.P. 6079, Succ. Centre-ville, Montréal, Québec, Canada H3C 3A7
e-mail: serge.prudhomme@polymtl.ca

Key words: Adjoint problem, Discontinuous Galerkin Method, Asymptotically exact estimator

Abstract. We derive a goal-oriented estimate of the error in finite element approximations of nonlinear parabolic equations based on a representation that involves an equilibrated flux reconstruction in space variable of the solution to the primal and to the (linear) dual backward-in-time problem. The error estimate can be applied to any arbitrary finite element discretization of the primal problem that admits a flux reconstruction in space variable satisfying some local space-time conservation and approximation properties [1]. We assume that an implicit-in-time Euler-type scheme is employed for the primal problem. The adjoint problem is then approximated using the same type of discretization scheme in time and a discontinuous Galerkin (dG) finite element method in space on the same mesh as the one used for the primal problem. Owing to the local conservation property of the dG, reconstructed equilibrated fluxes associated with the dual problem [2] can be straightforwardly obtained for the calculation of the error estimates. In fact, the dG method naturally produces fully computable elementwise contributions to the error, which are accurate even on the original mesh since the support of the basis functions coincides with the elements, see [3]. In this talk, we prove, and confirm with numerical experiments, that the proposed error estimator is asymptotically exact.

REFERENCES

[1] V. Dolejší, A. Ern, and M. Vohralík. A framework for robust a posteriori error control in unsteady nonlinear advection-diffusion problems. Technical report, Sept. 2012. HAL: hal-00652979, version 2.

[2] A. Ern, S. Nicaise, and M. Vohralík. An accurate $\mathbf{H}$(div) flux reconstruction for discontinuous Galerkin approximations of elliptic problems. *C. R. Math. Acad. Sci. Paris*, 345(12):709–712, 2007.

[3] I. Mozolevski and S. Prudhomme. Goal oriented estimates for elliptic equations based on equilibrated fluxes reconstruction. In preparation.

APPLICATION OF A FULLY EQUILIBRATED SUPERCONVERGENT PATCH RECOVERY SCHEME FOR ERROR BOUNDING

E. Nadal[*], **J.P. Moitinho de Almeida**[†], **J.J. Ródenas**[*], **F.J. Fuenmayor**[*], **O.A. González-Estrada**[††]

[*]Centro de Investigación en Tecnología de Vehículos (CITV)
Universitat Politècnica de València, C/ Vera, s/n, E-46022-Valencia, Spain
e-mail: {jjrodena, ffuenmay}@mcm.upv.es ennaso@upvnet.upv.es

[†]Department of Civil Engineering and Architecture
Instituto Superior Técnico, Technical University of Lisbon, Lisbon, Portugal
e-mail: moitinho@civil.ist.utl.pt

[††]Institute of Mechanics & Advanced Materials,
Cardiff University, School of Engineering, Queen's Building, The Parade, Cardiff CF24 3AA
Wales, UK.
e-mail: estradaoag@cardiff.ac.uk

Key words: Error bounding, Error estimator, Recovery technique, Statically admisible

Abstract. In this communication we present preliminary results assessing the application of a novel recovery technique, based on the well-known Superconvergent Patch Recovery (SPR) technique, to linear elasticity problems solved within the framework of the Finite Element Method (FEM). This recovery procedure provides statically admissible stress fields which are used to obtain upper bounds of the error in the energy norm. Traditionally, most error bounding approaches are residual-based, however their accuracy is not always high. Therefore practitioners and engineers generally prefer to use recovery-based techniques, because of their high accuracy and easy implementation.

Exploiting the fact that the strain energy of the difference between a kinematically admissible stress field (typically a raw FE solution) and a statically admissible stress field (for example our recovered solution) directly provides an upper bound of the error in the energy norm [1], previous recovery techniques, such as the SPR-C [2], tried to obtain upper bounds [3], relying on some correction terms, which depend on the exact solution, to account for their lack of equilibrium. In this work we compare the performance of both techniques.

1 Introduction

Numerical methods to solve Boundary Value Problems (BVP) such as the 2D linear elasticity problems have experimented a huge increase in their use by practitioners. More specifically, the displacement-based Finite Element Method (FEM) is widely used today in industries such as aerospace, civil engineering, automotive, etc. FEM is a powerful method for a vast type of engineering problems, however it is only able to provide an approximated solution. Therefore, some *error* level has to be accounted for to define the safety factors during the design process of mechanical parts.

During a Finite Element (FE) analysis there are several sources of error like geometrical simplifications of the actual component to make it suitable for the analysis, geometrical errors due to the FE discretization of the domain, the discretization error due to the FE approximation to the solution, etc. In this work we are going to focus only on the discretization error.

During the last part of the 20th century, scientist have developed techniques to obtain an estimation or a bound of the error in energy norm to quantify the quality of the FE results. The first approaches yielding upper bounds in the error estimation were based on explicit residuals [4], however their applicability is limited, since they are constant dependent. Later, a new procedure based on implicit residuals appeared which, under certain circumstances, is also able to provide upper bounds [5, 6, 7, 8, 9, 10, 11, 12]. Different approaches, related to the concept of dual analysis, working with a compatible and with an equilibrated solution, were also used to directly obtain upper error bounds. Some of them solving two global problems in parallel [13] or post-processing the FE solution [6, 14, 15]. The main characteristic of these error bounding techniques is that the error is evaluated by comparing the two solutions, one compatible and the other equilibrated, which are complementary in nature, and whose errors are orthogonal.

Other techniques, which traditionally were unable to obtain bounds for the error in energy norm, use the so-called Zienkiewicz and Zhu (ZZ) error estimator [16]. In this case the FE solution (compatible) is compared with an improved solution, not necessary equilibrated, obtained with a recovery procedure such as, e.g., the Superconvergent Patch Recovery (SPR) [17, 18]. Today, the ZZ error estimator is widely used due to its simplicity (only uses standard FE results) and high accuracy. However, the main drawback is that despite of the fact that the ZZ error estimator in combination with the SPR technique is asymptotically exact, it is unable to guarantee an upper bound of the error in energy norm.

Some works to improve the original SPR technique have been carried out. Ródenas and *et al.* proposed to add some constraints to impose local equilibrium to the recovered solution [2] obtaining a quasi-equilibrated recovered solution, using the so-called SPR-C technique. Díez *et al.*[3] presented a methodology to obtain computable upper bounds of the error in energy norm considering the quasi-equilibrated recovered field. Those ideas were also applied by Ródenas and co-workers [19, 20, 21, 22] in the eXtended Finite

Element Method (XFEM) framework [23, 24]. In all those methods the upper bound property was not strongly guaranteed by using directly the recovered solution. Thus, they needed the evaluation of some correction terms to compute the lack of equilibrium [3] for which only an estimation was available.

In this work, we present a procedure which directly recovers a fully equilibrated recovered solution from the superconvergent stresses. Then, directly comparing the FE solution with the recovered one, using a version of the the ZZ error estimator, we obtain guaranteed upper bounds of the error in the energy norm, yielding sharp estimations as shown in the section devoted to numerical tests.

2 Problem Statement

Let us consider the displacement field $\mathbf{u}$ taking values in $\Omega \subset \mathbb{R}^2$ as the solution of the 2D linear elasticity problem given by

$$-\nabla \cdot \boldsymbol{\sigma}\left(\mathbf{u}\right) = \mathbf{b} \qquad \text{in } \Omega \tag{1}$$

$$\boldsymbol{\sigma}\left(\mathbf{u}\right) \cdot \mathbf{n} = \mathbf{t} \qquad \text{on } \Gamma_N \tag{2}$$

$$\mathbf{u} = \mathbf{0} \qquad \text{on } \Gamma_D \tag{3}$$

where Γ_N and Γ_D are the parts of the boundary where the Neumann and Dirichlet conditions are applied, such that $\partial\Omega = \bar{\Gamma}_N \cup \bar{\Gamma}_D$ and $\Gamma_N \cap \Gamma_D = \emptyset$. $\mathbf{b}$ are the body loads and $\mathbf{t}$ are the tractions imposed along Γ_N. We consider a homogeneous Dirichlet boundary condition in (3) for simplicity.

The problem can be rewritten in its variational form as:

$$\text{Find } \mathbf{u} \in V : \forall \mathbf{v} \in V \quad a(\mathbf{u}, \mathbf{v}) = l(\mathbf{v}) \tag{4}$$

where $V = \left\{\mathbf{v} \mid \mathbf{v} \in H^1(\Omega), \mathbf{v}|_{\Gamma_D}(\mathbf{x}) = \mathbf{0}\right\}$ is the standard test space for the elasticity problem. The symmetric and bilinear form $a : V \times V \to \mathbb{R}$ and the continuous linear form $l : V \to \mathbb{R}$ are defined in vectorial form as:

$$a(\mathbf{u}, \mathbf{v}) := \int_\Omega \boldsymbol{\sigma}^T(\mathbf{u})\boldsymbol{\varepsilon}(\mathbf{v})d\Omega = \int_\Omega \boldsymbol{\sigma}(\mathbf{u})^T \mathbf{D}^{-1}\boldsymbol{\sigma}(\mathbf{v})d\Omega \tag{5}$$

$$l(\mathbf{v}) := \int_\Omega \mathbf{b}^T\mathbf{v}d\Omega + \int_{\Gamma_N} \mathbf{t}^T\mathbf{v}d\Gamma, \tag{6}$$

where $\boldsymbol{\sigma}$ represents the stresses, $\boldsymbol{\varepsilon}$ are the strains and $\mathbf{D}$ is the elasticity matrix of the constitutive relation $\boldsymbol{\sigma} = \mathbf{D}\boldsymbol{\varepsilon}$.

3 Fully equilibrated recovery procedure

Traditionally, recovery-type error estimators were unable to provide error bounds in energy norm. Díez *et al.*[3] made a first attempt by adding some correction terms to the ZZ error estimator in order to ensure the upper bound property. In this section we are

going to show a scheme for a new recovery procedure that, directly using the ZZ error estimator, yields upper error bounds in energy norm.

When the recovered stress field $\boldsymbol{\sigma}^*$ is statically admissible in the ZZ error estimator, it yields an upper error bound. Thus, in this section we aim to the evaluation of an statically admissible stress field. To do that, $\boldsymbol{\sigma}^*$ has to fulfil the internal equilibrium equation (IEE), the boundary equilibrium equation (BE) and equilibrium of tractions along the internal element edges (IB). Note that the normal stress tangent to the boundary is no necessarily continuous along the element edges.

The technique presented here, called SPR-FE (Fully Equilibrated), is based in the SPR technique developed by Zienkiewicz and Zhu [17]. In the SPR-FE, as in SPR, we create patches of elements with the elements connected by the vertex nodes, so-called patch assembly nodes (AN), see Figure 1a. There are two main differences between the traditional SPR and the SPR-FE: a) in the SPR each recovered stress component is represented by a single polynomial on each patch, while for the SPR-FE a polynomial surface is fitted for each stress component on each element of the patch. In Figure 1a we fit, by minimizing (7), a different polynomial surface for each stress component at elements I, II, III, IV. b) the second difference is that the SPR technique builds up the global recovered field in an element by adding the contributions of each patch using a partition of unity. However, in the SPR-FE the global recovered field is obtained by directly adding the contributions of all patches $\boldsymbol{\sigma}^* = \sum_i^{AN} \boldsymbol{\sigma}_i^k$ connected to one element k since the partition of unity is implicit in the functional (7). Note that when we apply the constraints for internal and boundary equilibrium the problem loads will be also affected by the partition of unity N_i^k.

For the statically admissibility condition, we add the constraints that are necessary to enforce the required continuity and equilibrium in the recovered solution using a point collocation approach, the number of points will depend on the degree of the recovered field. This is obtained by adding continuity of tractions along the internal edges (red edges). We enforce the recovered tractions to zero along the external edges (blue edges) and finally we enforce the equilibrium equation at each element, separately. The recovery process will be described below in more detail.

3.1 Recovery procedure

We minimize the following functional on each of the k elements of the patch with assembly node i:

$$\Psi = \int_{\Omega_k} \left(\boldsymbol{\sigma}_k^* - N_i^k \boldsymbol{\sigma}^h \right)^2 \, d\Omega \qquad k = I, II, III, IV \tag{7}$$

where N_i^k is the linear shape function of the node i, in element k. $\boldsymbol{\sigma}^h$ is the FE stress field and $\boldsymbol{\sigma}_k^* = \mathbf{P}_k \mathbf{a}_k$ is the recovered stress field for the element k, where $\mathbf{a}_k = \{\mathbf{a}_k^{xx}, \mathbf{a}_k^{yy}, \mathbf{a}_k^{xy}\}^T$ are the coefficients for each stress component and $\mathbf{P}_k$ is the matrix for the polynomial expansion $\mathbf{p}_k = \{x^m y^n \ : \ m, n \leq q\}_k$, where q is the polynomial degree

(a) Patch of elements. Assembly node i represented in orange.

(b) Patch in contact with a boundary (green line) of the domain.
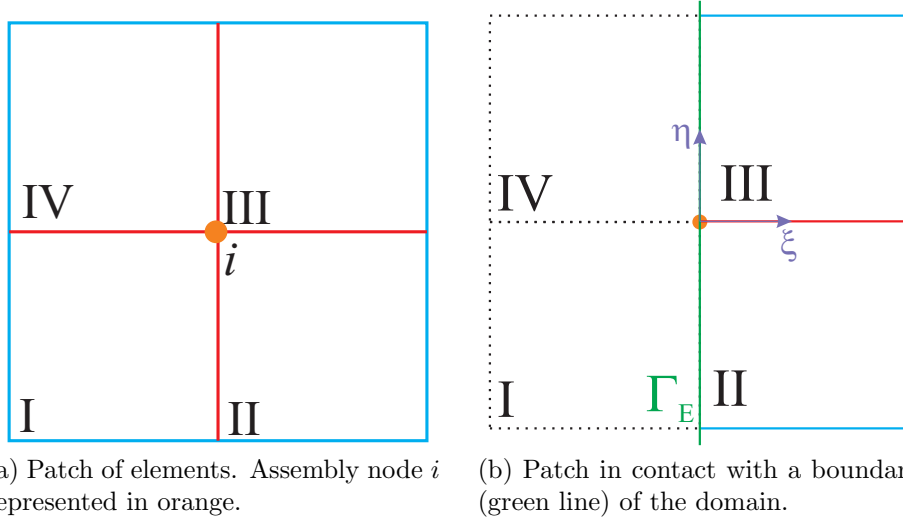
Figure 1: Internal patch formed by 4 elements (left) and patch in contact with the boundary formed by 2 elements (right).

$$\mathbf{P}_k = \begin{bmatrix} \mathbf{p}_k & 0 & 0 \\ 0 & \mathbf{p}_k & 0 \\ 0 & 0 & \mathbf{p}_k \end{bmatrix} \tag{8}$$

For each element k, integrating numerically after the minimization of (7) we obtain the following expression:

$$\sum_{pg} \mathbf{P}_k^T \mathbf{P}_k |\mathbf{J}| \omega \mathbf{a}_k = \sum_{pg} \mathbf{P}_k^T N_i^k \boldsymbol{\sigma}^h |\mathbf{J}| \omega \tag{9}$$

where $|\mathbf{J}|$ is the Jacobian of the coordinates transformation, ω is the weight of each integration point and pg is the number of integration points. This expression yields a linear system of equations for each element of the patch $\mathbf{M}_k \mathbf{a}_k = \mathbf{g}_k$. Due to the constraints we have imposed, we need some interaction between the different recovered stress fields. Thus, we assemble all four systems together and we obtain the following linear system for the patch:

$$\begin{bmatrix} \mathbf{M}_I & 0 & 0 & 0 \\ 0 & \mathbf{M}_{II} & 0 & 0 \\ 0 & 0 & \mathbf{M}_{III} & 0 \\ 0 & 0 & 0 & \mathbf{M}_{IV} \end{bmatrix} \begin{Bmatrix} \mathbf{a}_I \\ \mathbf{a}_{II} \\ \mathbf{a}_{III} \\ \mathbf{a}_{IV} \end{Bmatrix} = \begin{Bmatrix} \mathbf{g}_I \\ \mathbf{g}_{II} \\ \mathbf{g}_{III} \\ \mathbf{g}_{IV} \end{Bmatrix} \quad \Rightarrow \quad \mathbf{Ma} = \mathbf{g} \tag{10}$$

3.2 Internal equilibrium constraint

In contrast with the SPR-C presented in [2], where the internal equilibrium equation was $\nabla \cdot \boldsymbol{\sigma}_k^* + \mathbf{b} = 0$, in the SPR-FE we have to take into account the partition of unity

introduced in the functional (7). Therefore, it will affect to the body forces $\mathbf{b}$ and also bring up a new term, first introduced in [15]: *the fictitious body forces*, $\nabla N_i^k \boldsymbol{\sigma}^h$. Their role is to ensure that the forces applied to each patch satisfy global equilibrium for the isolated patch. If their were not considered, then Equation (10), would generally have no solution. Nevertheless, when we sum up the contributions of the four patches of a single element k these terms will sum to zero, cancelling their effect at a global level.

Then, the internal equilibrium equation to impose in this case is $\nabla \cdot \boldsymbol{\sigma}_k^* + N_i^k \mathbf{b} = \nabla N_i^k \boldsymbol{\sigma}^h$ at each element k. These constraints are independently enforced in all elements. This generates the internal equilibrium matrix for each element $\mathbf{C}_k^{IEE}$ and the independent term $\mathbf{h}_k^{IEE}$.

3.3 External patch edge constraint

The next step is to add the constraints along the external boundaries of the patch, that is, the constraints along the blue edges in Figure 1a. These constraints will ensure tractions continuity when we sum up the contributions from the patches related to an element. Since the partition of unity function is zero at the external edges of the patch, the equation to be imposed there is $\boldsymbol{\sigma}_k^* \cdot \mathbf{n} = \mathbf{0}$, where $\mathbf{n}$ is the outward normal vector along the patch boundary. This generates for each element the matrix $\mathbf{C}_k^{BE}$ and the independent term $\mathbf{h}_k^{BE} = 0$.

3.4 Internal patch edge constraint

Finally, it is also necessary to add the constraints along the internal boundaries of the patch (red edges), *i.e.* the interfaces between elements. These are also used to ensure tractions continuity along the element interface. The equation to be imposed is $\boldsymbol{\sigma}_k^* \cdot \mathbf{n}_k + \boldsymbol{\sigma}_l^* \cdot \mathbf{n}_l = \mathbf{0}$, where $k \neq l$, $\mathbf{n}_k$ and $\mathbf{n}_l$ are the outward normal vectors of each elements in the common edge ($\mathbf{n}_k = -\mathbf{n}_l$), generating $\mathbf{C}_k^{IB}$ and $\mathbf{C}_l^{IB}$ respectively, the RHS is again null $\mathbf{h}_k^{IB} = 0$.

A particular situation occurs when an internal edge coincides with a boundary where the tractions are prescribed. In Figure 1b we illustrate such a boundary (green line), which is internal to the patch. This is a typical situation when the assembly nodes (orange point) are over the boundary. In this case, the equations to be imposed have to take into account the Neumann boundary condition then, $\boldsymbol{\sigma}_k^* \cdot \mathbf{n}_k = N_i^k \mathbf{t}$. When the boundary condition is non-homogeneous the corresponding term in the RHS is generally not null, $\mathbf{h}_k^{IB} \neq 0$. Note that the opposite element sharing the edge, *i.e.* element l in the general case, does not exist for this type of patch.

3.5 System resolution considerations

Adding all constraints to (10) we obtain the following linear system to solve at each patch:

$$
\begin{bmatrix}
\mathbf{M} & (\mathbf{C}^{IEE})^T & (\mathbf{C}^{EB})^T & (\mathbf{C}^{IB})^T \\
\mathbf{C}^{IEE} & 0 & 0 & 0 \\
\mathbf{C}^{EB} & 0 & 0 & 0 \\
\mathbf{C}^{IB} & 0 & 0 & 0
\end{bmatrix}
\begin{Bmatrix}
\mathbf{a} \\
\boldsymbol{\lambda}^{IEE} \\
\boldsymbol{\lambda}^{EB} \\
\boldsymbol{\lambda}^{IB}
\end{Bmatrix}
=
\begin{Bmatrix}
\mathbf{g} \\
\mathbf{h}^{IEE} \\
0 \\
\mathbf{h}_k^{IB}
\end{Bmatrix}
\tag{11}
$$

and it could be rewritten as $\mathbf{M}_C \mathbf{a}_C = \mathbf{g}_C$, where C indicates that the constraints are included.

The basis $\mathbf{p}$ for the stress field has to be able to represent all constraints to guarantee the statical admissibility property. Thus, we need to analyse the minimum degree required for the stress field to guarantee the equations system (11) is solvable. We consider a bi-quadratic representation of the displacement field. Linear or bilinear FE solutions cannot be directly applied to this recovery procedure since they do not guarantee rotational equilibrium of the patch [25]. Then, the FE stress field $\boldsymbol{\sigma}^h$ has quadratic terms, the partition of unity used in (7) is bilinear and its divergence has linear terms. Analyzing the constraints we need at least a 4th order polynomial interpolation because of the fictitious body forces. In Table 1 we show the total size of the system to be solve at each node. We have to pay attention to the "effective" number of "free coefficients" (last column), the difference between the number of "Coefficients" $\mathbf{a}$ and the number of independent constraints. For degree 4 there are more constraints (192+80+40 ¿ 300) than coefficients, therefore some constraints have to be linearly dependent. In fact there are always linearly dependent constraints, as indicated by the difference between the dimension of the system and its rank, and thus we obtain 16 effective free coefficients for degree 4. We use the Singular Value Decomposition (SVD) technique to solve that system, identifying the dependent equations and their consistency.

Degree	Coefficients	IEE ctr	EB ctr	IB ctr	System size	Rank	Eff free coef
4	300	192	80	40	612	584	16
5	432	280	96	48	856	828	40
6	588	384	112	56	1140	1108	68

Table 1: Number of coefficients and constraints

4   Numerical results

In this section we verify numerically that the upper bound property is satisfied by the error estimation when the SPR-FE is used in the ZZ error estimator in (12). We also define the global effectivity index $\theta = \|\mathbf{e}_{es}\|/\|\mathbf{e}\|$ as the ratio between the estimated error $\|\mathbf{e}_{es}\|$ and the exact error, where $\boldsymbol{\sigma}$ is the exact stress field:

$$
\|\mathbf{e}_{es}\|^2 = \int_\Omega \left(\boldsymbol{\sigma}^* - \boldsymbol{\sigma}^h\right)^T \mathbf{D}^{-1} \left(\boldsymbol{\sigma}^* - \boldsymbol{\sigma}^h\right) \mathrm{d}\Omega
\tag{12}
$$

$$\|\mathbf{e}\|^2 = \int_\Omega \left(\boldsymbol{\sigma} - \boldsymbol{\sigma}^h\right)^T \mathbf{D}^{-1} \left(\boldsymbol{\sigma} - \boldsymbol{\sigma}^h\right) \mathrm{d}\Omega \tag{13}$$

We compare the results obtained with the SPR-FE with those obtained with the SPR-C [2]. In all problems, plane strain and bi-quadratic elements will be considered for all analyses.

4.1 Problem 1. 2×2 square

This problem has an analytical cubic solution in displacements with body forces. The problem model, material properties and exact solution are represented in Figure 2.
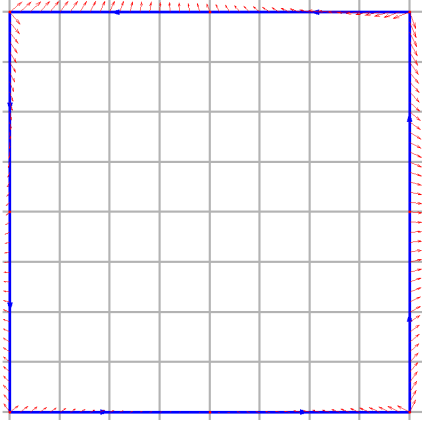
$$u_x = x + x^2 - 2xy + x^3 - 3xy^2 + x^2 y$$
$$u_y = -y - 2xy + y^2 - 3x^2 y + y^3 - xy^2$$
$$\sigma_{xx} = \frac{E}{1+\nu}(1 + 2x - 2y + 3x^2 - 3y^2 + 2xy)$$
$$\sigma_{yy} = \frac{-E}{1+\nu}(1 + 2x - 2y + 3x^2 - 3y^2 + 2xy)$$
$$\sigma_{xy} = \frac{E}{1+\nu}\left(-x - y + \frac{x^2}{2} - \frac{y^2}{2} - 6xy\right)$$
$$b_x = \frac{-E}{1+\nu}(1+y) \qquad b_x = \frac{-E}{1+\nu}(1-x)$$
$$E = 1000, \; \nu = 0.3$$

Figure 2: Problem 1. Model, material and analytical solution.

Figure 3 shows the results obtained with the SPR-FE recovered stress field with 4th (blue line) and 5th (red line) order polynomial interpolation. Black lines correspond to the results obtained with the SPR-C technique. In terms of effectivity of the error estimator, we observe that in all cases θ is above 1 (satisfies the upper bound property) and very close to 1 (very accurate error estimation). Theoretically, for the SPR-FE the upper bound is guaranteed but not for the SPR-C, however the results are quite similar. Regarding the computational cost, we observe a considerable difference between the techniques. This is due to the complexity of the technique used to solve system (11).

4.2 Problem 2. 2×2 square in cylinder under internal pressure

The analytical solution of this problem corresponds to a pipe under internal pressure. However, we have extract form the problem an squared area (green area) for the analysis and we have run the FE analysis applying the corresponding Neumann boundary conditions and constraining the rigid body motions. The problem model, material properties and exact solution is represented in Figure 4, in polar coordinates.
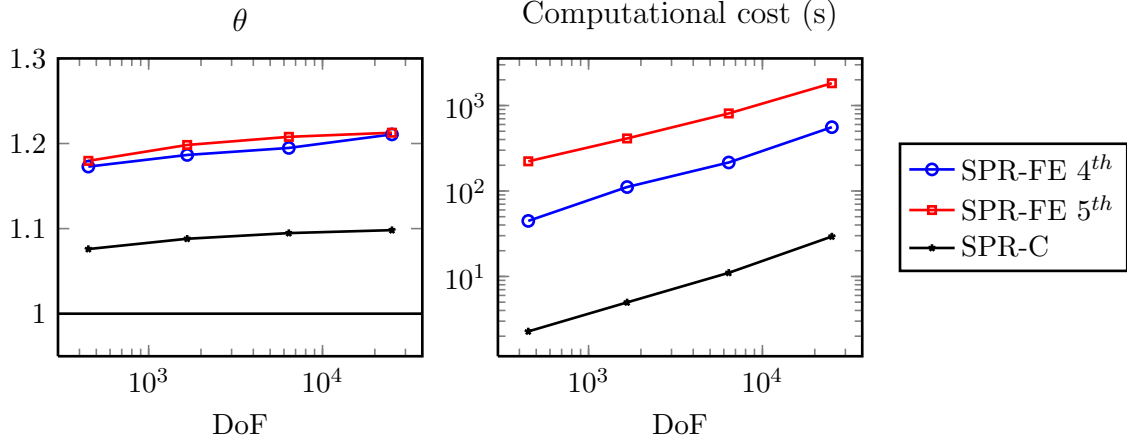
Figure 3: Problem 1. Q8. Global effectivity index θ and overall computational cost for SPR-FE and SPR-C techniques.
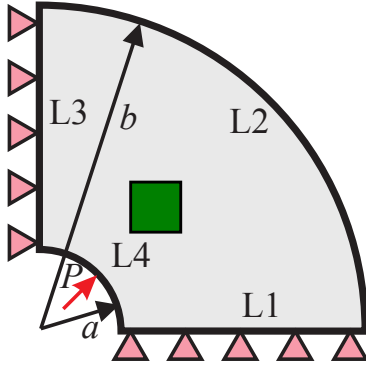
$$u_r(r) = \frac{P(1+\nu)}{E(c^2-1)} \left(r(1-2\nu) + \frac{b^2}{r} \right)$$

$$\sigma_r(r) = \frac{P}{c^2-1} \left(1 - \frac{b^2}{r^2} \right)$$

$$\sigma_\theta(r) = \frac{P}{c^2-1} \left(1 + \frac{b^2}{r^2} \right)$$

$$a = 5 \quad b = 20 \quad P = 1$$

$$E = 1000 \quad \nu = 0.3 \quad c = \frac{b}{a}$$

Figure 4: Problem 2: Model, material and analytical solution.

For this problem the upper bound property is not strictly guaranteed because the tractions over the boundary of the domain (green area) cannot be represented by the polynomial basis used for the recovery. However, in Figure 5 we observe again an upper bound for the two techniques and the results are very accurate for all of them.

5   Conclusions

In this work we have presented a novel technique that is able to provide an upper error bound in energy norm. This technique is based on the ZZ error estimator and it uses an elaborated recovery procedure. Results showed that the computational cost to obtain that recovered field is quite high in comparison with standard recovery procedures. The main advantage of the SPR-FE is that, in contrast with other recovery procedures
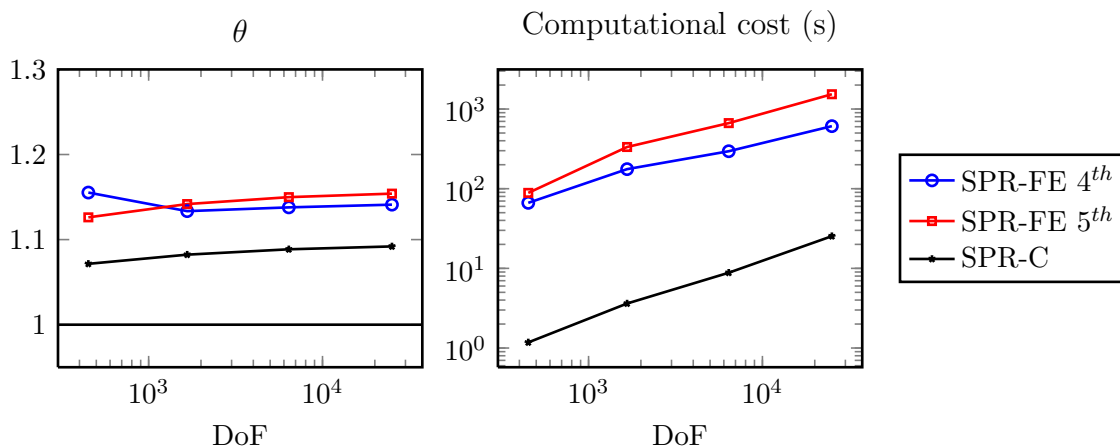
Figure 5: Problem 2. Q8. Global effectivity index θ and computational cost for SPR-FE and SPR-C techniques.

such as the SPR-C, it is able to obtain guaranteed error bounds without any correction terms. Nevertheless, the SPR-C is obtaining for these examples numerical upper bounds and error estimates close to one. We are currently working to improve the computational cost associated to the SPR-FE to make it competitive with traditional error bounding techniques.

ACKNOWLEDGEMENTS

REFERENCES

[1] Práger W, Synge JL. Approximation in elasticity based on the concept of function space. *Quart. Appl. Math.* 1947; **5**:241–269.

[2] Ródenas JJ, Tur M, Fuenmayor FJ, Vercher A. Improvement of the superconvergent patch recovery technique by the use of constraint equations: the SPR-C technique. *International Journal for Numerical Methods in Engineering* 2007; **70**(6):705–727.

[3] Díez P, Ródenas JJ, Zienkiewicz OC. Equilibrated patch recovery error estimates: simple and accurate upper bounds of the error. *International Journal for Numerical Methods in Engineering* 2007; **69**(10):2075–2098.

[4] Babuška I. Error-Bounds for Finite Element Method. *Numerische Mathematik* 1970; **16**:322–333.

[5] Babuška I, Rheinboldt WC. A-posteriori error estimates for the finite element method. *International Journal for Numerical Methods in Engineering* 1978; **12**(10):1597–1615.

[6] Ladevèze P, Leguillon D. Error estimate procedure in the finite element method and applications. *SIAM Journal on Numerical Analysis* 1983; **20**(3):485–509.

[7] Ladevèze P, Maunder EAW. A general method for recovering equilibrating element tractions. *Computer Methods in Applied Mechanics and Engineering* 1996; **137**(2):111–151.

[8] Ladevèze P, Rougeot P. New advances on a posteriori error on constitutive relation in fe analysis. *Computer Methods in Applied Mechanics and Engineering* 1997; **150**(1-4):239–249.

[9] Ladevèze P, Rougeot P, Blanchard P, Moreau JP. Local error estimators for finite element linear analysis. *Computer Methods in Applied Mechanics and Engineering* 1999; **176**(1-4):231–246.

[10] Pled F, Chamoin L, Ladevèze P. An enhanced method with local energy minimization for the robust a posteriori construction of equilibrated stress fields in finite element analyses. *Computational Mechanics* Sep 2011; .

[11] Díez P, Parés N, Huerta A, Diez P, Pares N. Recovering lower bounds of the error by postprocessing implicit residual a posteriori error estimates. *International Journal for Numerical Methods in Engineering* Mar 2003; **56**(10):1465–1488.

[12] Díez P, Parés N, Huerta A. Accurate upper and lower error bounds by solving flux-free local problems in stars. *Revue européenne des éléments finis* 2004; **13**(5-6-7):497.

[13] Pereira OJBA, de Almeida JPM, Maunder EAW. Adaptive methods for hybrid equilibrium finite element models. *Computer Methods in Applied Mechanics and Engineering* 1999; **176**(1-4):19–39.

[14] Almeida OJB, Moitinho JP. A posteriori error estimation for equilibrium finite elements in elastostatic problems. *Computer Assisted Mechanics and Engineering Sciences* 2001; **8**(2-3):439–453.

[15] Moitinho JP, Maunder EAW. Recovery of equilibrium on star patches using a partition of unity technique. *International journal for ...* 2009; **79**:1493–1516.

[16] Zienkiewicz OC, Zhu JZ. A simple error estimator and adaptive procedure for practical engineering analysis. *International Journal for Numerical Methods in Engineering* 1987; **24**(2):337–357.

[17] Zienkiewicz OC, Zhu JZ. The superconvergent patch recovery and a posteriori error estimates. Part 1: The recovery technique. *International Journal for Numerical Methods in Engineering* 1992; **33**(7):1331–1364.

[18] Zienkiewicz OC, Zhu JZ. The superconvergent patch recovery and a posteriori error estimates. Part 2: Error estimates and adaptivity. *International Journal for Numerical Methods in Engineering* 1992; **33**(7):1365–1382.

[19] Ródenas JJ, Giner E, Fuenmayor FJ, González-Estrada OA. Accurate recovery-type error estimation for linear elastic fracture mechanics in FEM and X-FEM based on a singular+smooth field splitting. *International Conference on Adaptive Modeling and Simulation. ADMOS 2007*, International Center for Numerical Methods in Engineering (CIMNE), 2007; 202–205.

[20] Ródenas JJ, González-Estrada OA, Tarancón JE, Fuenmayor FJ. A recovery-type error estimator for the extended finite element method based on singular+smooth stress field splitting. *International Journal for Numerical Methods in Engineering* 2008; **76**(4):545–571.

[21] Ródenas JJ, González-Estrada OA, Díez P, Fuenmayor FJ. Upper bounds of the error in the extended finite element method by using an equilibrated-stress patch recovery technique. *International Conference on Adaptive Modeling and Simulation. ADMOS 2007*, International Center for Numerical Methods in Engineering (CIMNE), 2007; 210–213.

[22] Ródenas JJ, González-Estrada OA, Díez P, Fuenmayor FJ. Accurate recovery-based upper error bounds for the extended finite element framework. *Computer Methods in Applied Mechanics and Engineering* 2010; **199**(37-40):2607–2621.

[23] Moës N, Dolbow J, Belytschko T. A finite element method for crack growth without remeshing. *International Journal for Numerical Methods in Engineering* 1999; **46**(1):131–150.

[24] Sukumar N, Prévost JH. Modeling quasi-static crack growth with the extended finite element method. {Part I}: Computer implementation. *International Journal of Solids and Structures* 2003; **40**(26):7513–7537.

[25] Maunder EAW, Moitinho JP. Recovery of equilibrium on star patches from conforming finite elements with a linear basis. *International Journal for . . .* 2012; **89**:1497–1526.

CONFORMAL HEXAEDRAL MESHES AND ADAPTIVE MESH REFINEMENT

GÉRALD NICOLAS[*], THIERRY FOUQUET[*]

[*] EDF R&D
1 avenue du Général de Gaulle
92140, Clamart, France
e-mail: gerald.nicolas@edf.fr
e-mail: thierry.fouquet@edf.fr

Key words: h-refinement; adaptive meshing; hexahedron; conformal mesh; finite element; structural mechanics

Abstract. During a numeric simulation based on the finite element method, the h-refinement of the mesh consists in splitting the elements where an error indicator is higher than a threshold. One major point is that the final mesh must be conformal. When the mesh is defined only by triangles or tetrahedra, the junction between two zones with a different level of refinement has been solved for many years. When the mesh is made of hexahedra, this junction cannot be made of hexahedra. A proposal is made in this paper to connect the zones with some specific elements. Two applications are presented here and show the efficiency of the method.

1. INTRODUCTION

In a numerical simulation using the finite element method, the mesh has to be fine enough to guarantee the accuracy of the solution. To achieve this goal, mesh adaptation offers an effective compromise, combining a fine mesh with a low computational cost. When using the h-refinement method, some meshes are divided but difficulties occur at the interface between two zones with different levels of refinement, if a conformal mesh is required. That problem is solved either by specific finite elements in the junction [1] or by a specific splitting of these meshes [2].

If the initial mesh is made of triangles or tetrahedra, the splitting of the meshes at the interface produces new triangles or tetrahedra. Since the early 90's, this method has been implemented in HOMARD, our software for mesh refinement ([3], available in [4]). But in some numeric simulations, the initial mesh is made of hexahedra because they are more efficient than the tetrahedra. In that case, the transition is not as simple as it is with the tetrahedra: the conformal connection cannot be made with others hexahedra. To solve this problem, we developed a new method. First, the error indicator from the computed solution is used to produce a non-conformal mesh [5]. Then, every hexahedron that is located at the interface between the zones of different levels of refinement is examined: using tetrahedra and pyramids makes possible a conformal connection.

The first part of this paper, chapter 2, presents a basic application on a 3D structure that shows the advantages of the h-refinement when the mesh is made of hexahedra. The transition zone will be described and the convergence of the computation is easily reached. Secondly, chapter 3, the central part of the method is detailed. Last, chapter 4, an industrial application of the method is presented.

2. FIRST EXAMPLE OF APPLICATION

To give an idea of the effectiveness of the h-refinement with a mesh made of hexahedra, we present an example in structural mechanics. Lo et al suggested this test case in [1]. The model represents a 3D cross with one fixed face (see. Figure 1). A uniform pressure is applied on one face. The objective is to get an accurate value of the field of displacement along the line opposite to the fixed face. The simulations are done with *Code_Aster* [6], the open source finite element software for mechanical analysis. The mesh adaptations for this test case have been driven using the goal-oriented estimation of the error [7]. The goal of the mesh adaptation is to increase the accuracy on the displacement.
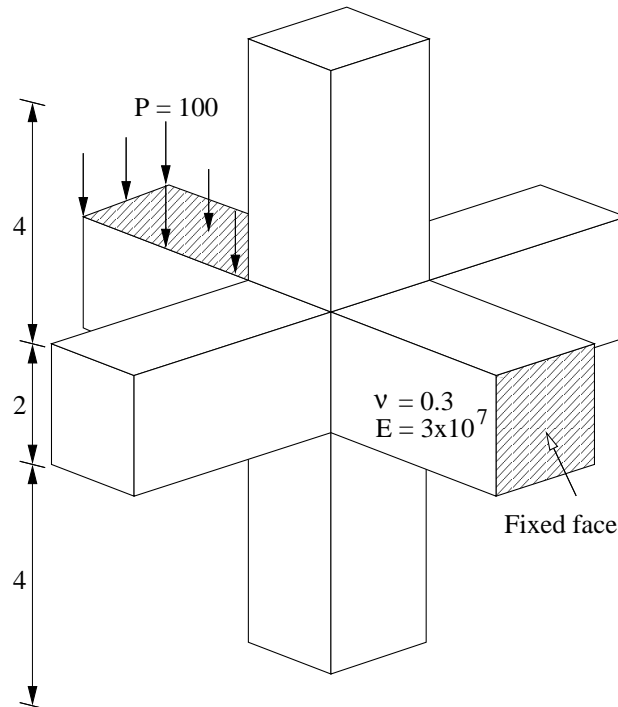
Figure 1: Description of the 3D cross

The initial mesh is composed of second order hexahedra. The threshold is established as 10% of the largest error in the initial mesh. The adaptation stops when the global error indicator on the structure is 20% of its value on the initial mesh. The evolution of the refined meshes along the iterations of the adaptation ensures the diminution of the global error of the problem. When the convergence is reached, the aspect of the meshes all along the iterations is similar to those obtained by Lo. When we look at the meshes (see. Figure 2), we see that the refinement occurs near the fixed face and near the edges between two branches of the cross,

as expected. The transition zones between two different levels of refinement are visible in the figure each time a volume is not a hexahedron anymore.
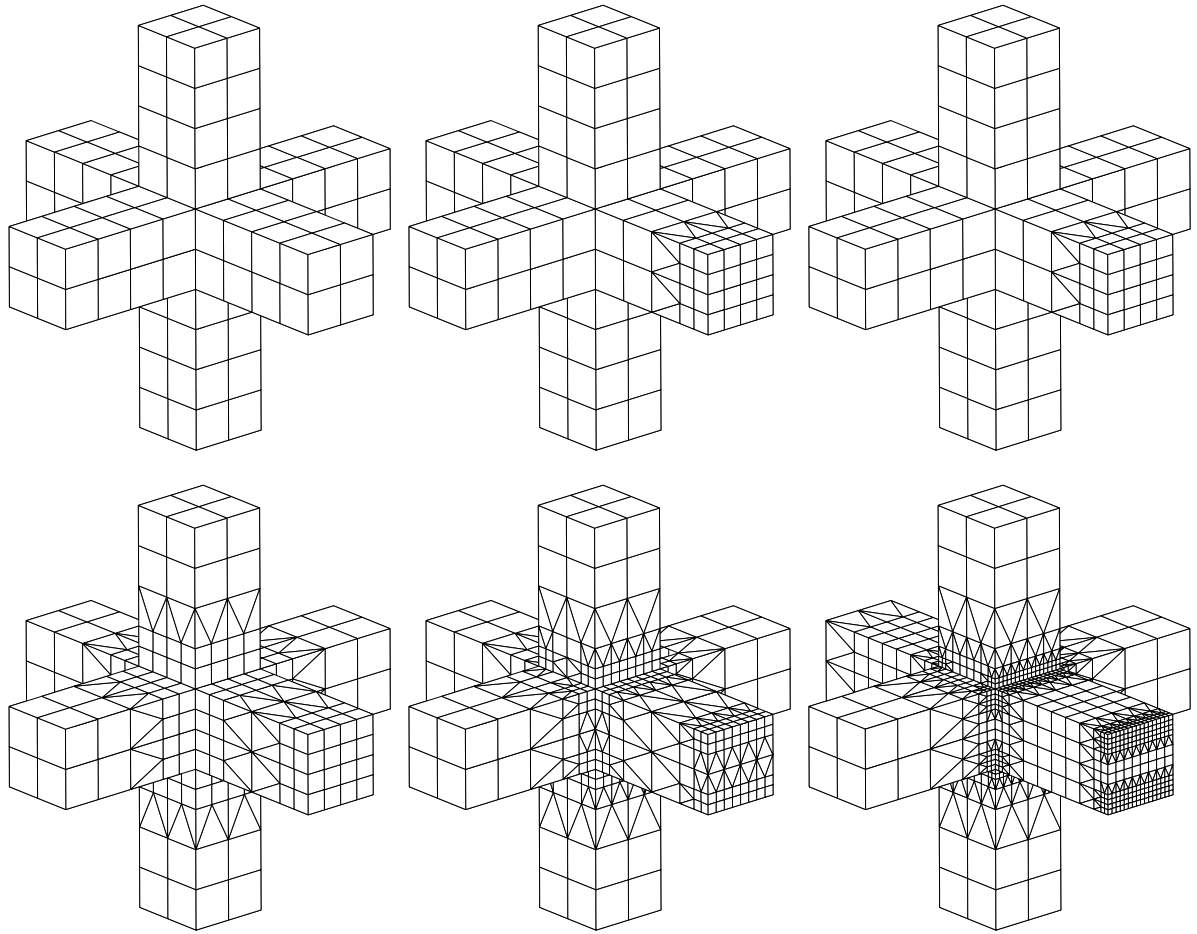
Figure 2: Initial and adapted meshes of the 3D cross

For some different strategies (see. Figure 3), we present the value of the total error plotted against the number of nodes, in a log-log-plot. First, a uniform refinement is tested (circle). As expected, a fast convergence cannot be reached with a uniform refinement. In the other cases, the threshold is defined either by an absolute value (abs plot), or by the percentage of the "worst" meshes (pcm plot). We can see that the influence of the strategy on the speed of convergence is rather small. Whatever the choice, the convergence is reached with more or less the same number of degrees of freedom. This conclusion is coherent with our experience: the most important is to catch the elements where the error is high and to be sure that they are refined.
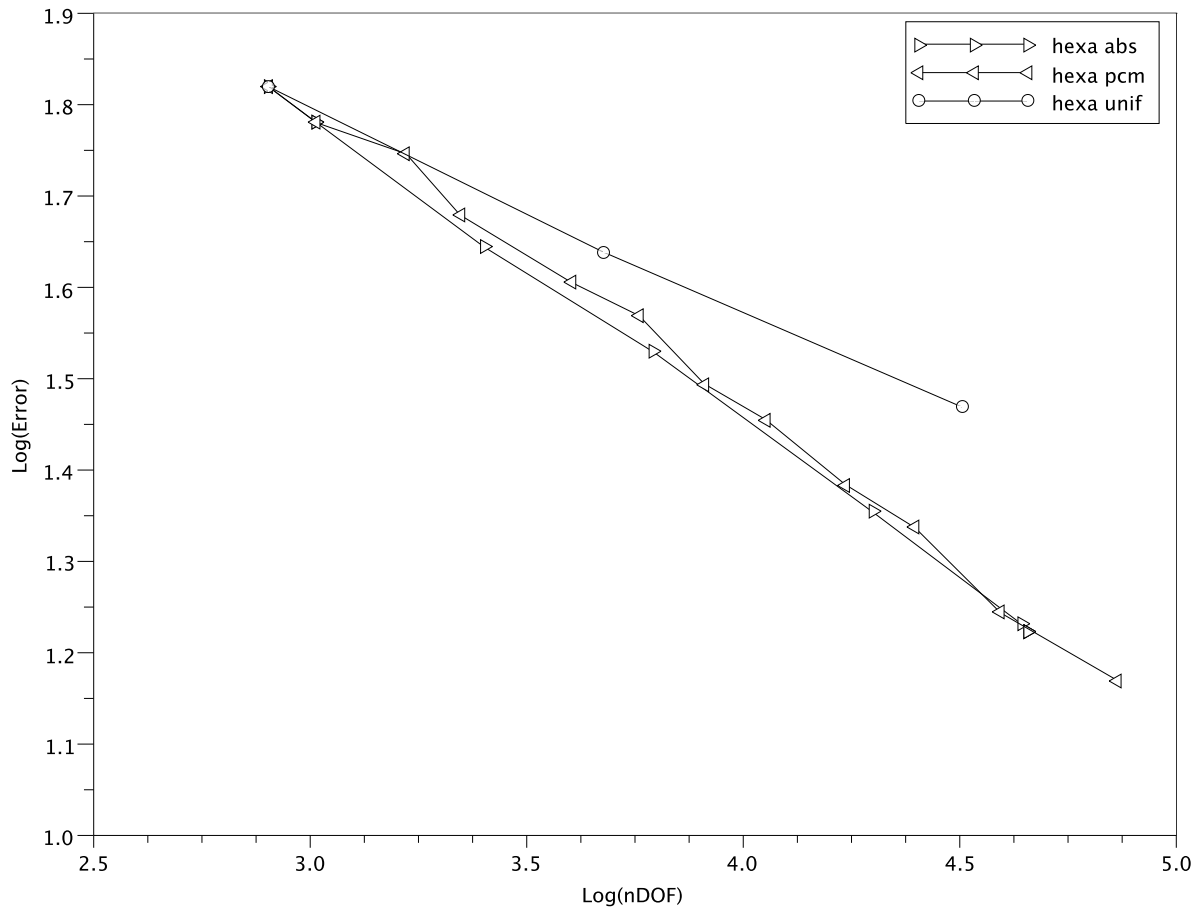
Figure 3: Convergence

3. ALGORITHM

The algorithm is described with details in [3]. The guidelines are presented here; a special focus is made on the technique that is employed to produce a conformal mesh when hexahedra are present.

3.1. Basic ideas

At the beginning of the process, all the elements of the initial mesh belong to level #0. The computation of the physical problem produces an error indicator over every element. Giving a threshold identifies some elements: those where the error indicator is higher than this threshold are split. That phase creates some elements that belong to level #1. Then, we have to solve the junction between the two different levels of refinement by introducing some special divisions of the elements to produce a conformal mesh. A new computation is made over this new mesh and the same adaptation can be processed one more time, until convergence is reached.

In our method, we decided to deal with 3D meshes and 2D meshes as well. When the mesh is composed of 3D elements, the faces of every element are defined in the data structure. If a

3D element has to be split because of the value of the error indicator, the decision of refinement is transferred to its faces. Then the resolution of the conflicts between the levels of refinement is made uniquely by considering the 2D faces. This is the central part of the algorithm and it is the same whatever the composition of the initial mesh. Last, examining the final decisions of its faces makes the refinement of every 3D element.

We illustrate this technique for a 2D mesh made of quadrangles. It is based on a three-stage procedure:

    A. Mark every quadrangle that requires refinement according to the error indicator
    B. Mark every quadrangle that has at least two refined neighbours
    C. Stop the propagation using transition elements

These stages are illustrated in Figure 4. Suppose that the grey quadrangles are the elements over which the error is higher than the threshold (A1). The first action of the phase A is to refine these elements: all their edges are equally split in two by placing a node at their midsections, so that four internal quadrangles are created (A2). The central quadrangle with two refined neighbours is then refined (B1). In our example, that action modifies the status of its neighbour on the right: it now has two split neighbours so it also has to be refined (B2). At this point, there are no more quadrangles with 2 refined neighbours or more. In the last step, all the quadrangles with one refined neighbour are split using a special technique (C).

A1. Error > threshold   A2. Refined meshes

B1/B2 Rules of 2 neighbours   C. The transition meshes
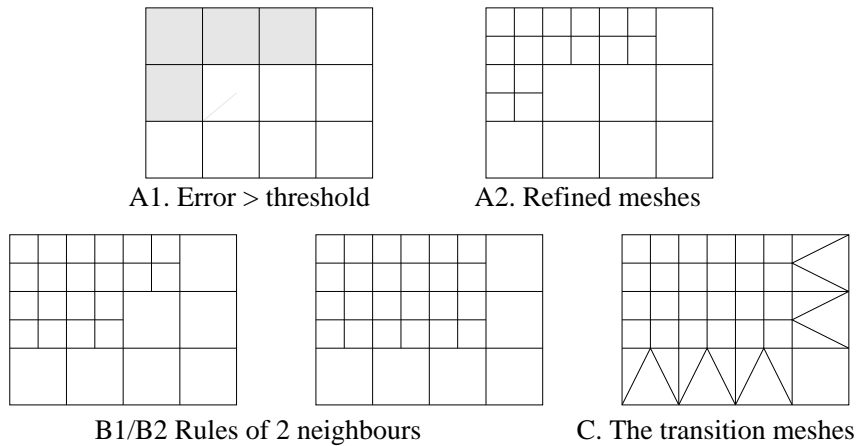
Figure 4: Algorithm for a 2D mesh

3.2.   The transition elements

A special point must be made regarding the transition elements. They are defined to ensure a conformal transition between two different levels of refinement. When 3D elements are present in the mesh, at the end of the phase B of the algorithm, these 3D elements can be sorted by the status of their faces:

- No face is split: the 3D element is kept as is
- All the faces are regularly split: the 3D element is regularly refined
- Other situations: the 3D element is located into a transition zone between two different levels of refinement. Each situation must be analyzed (see. 3.2.2)

3.2.1. Refinement of a quadrangle

At the end of the phase B of our algorithm (Mark every quadrangle that has at least two refined neighbours) every quadrangle is in one of these 3 situations: either no edge is split, or a unique edge is split, or all edges are split. If all the edges are split, placing a node at its midsection equally splits each edge. Adding a node at the centre of the quadrangle produces the 4 quadrangles of the regularly refinement

When a unique edge is split, some transition elements are needed. We decided that no additional node on any edge would be introduced to create them. Consequently, the pending node is connected to the opposite vertex, to produce three triangles.
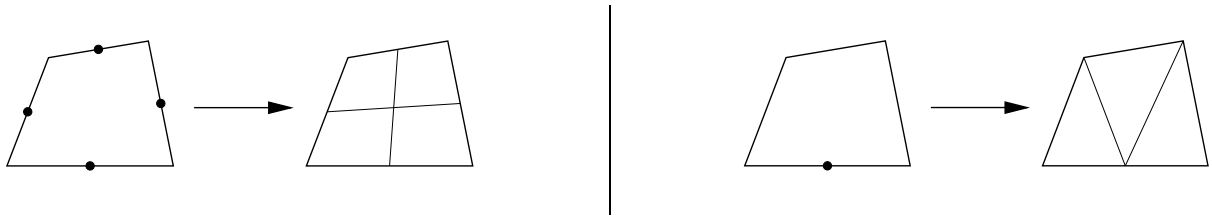
Figure 5: Refinement of a quadrangle: regular and transition

3.2.2. Transition elements for the hexahedra

The situation is much more complex when a hexahedron is located at the interface between two zones with a different level of refinement. The analysis of this case constitutes the central point of our work. When the rules are applied, one of six cases can happen at the end of the phase B:

- No edge is split: the hexahedron is kept as is
- All the edges are split: the hexahedron is regularly refined.
- A unique edge is split. The two quadrangular faces that share this edge are refined by transition while the four others faces are kept intact. Two internal edges are created from the middle of the split edge to the two vertices on the opposite edge. This internal division produces four pyramids whose bases are the four non-split quadrangular faces of the hexahedron.
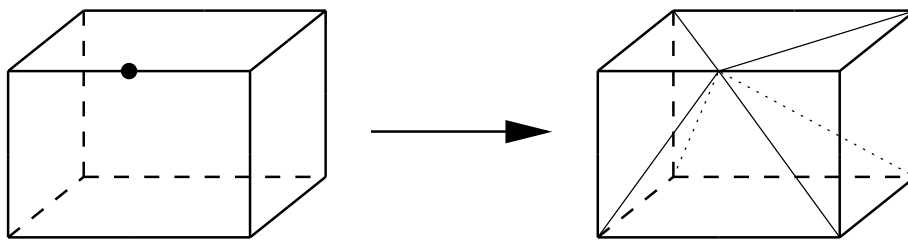
Figure 6: One edge is split

- Two edges on two different faces are split. The two faces of the hexahedron that do not have a split edge are kept intact, while the other four are split into triangles. An additional node is created at the centroid of the hexahedron. Ten internal edges are created with that centroid as a vertex: two are created to connect it to the centre of the two split edges while the other eight are created with the eight vertices of the hexahedron. Two pyramids are then built with bases on the two intact quadrangular faces of the hexahedron, and twelve tetrahedra are created on the remaining triangles.
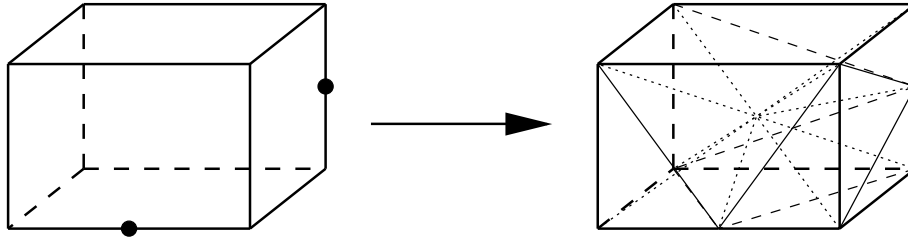
Figure 7: Two edges are split

- Three edges on three different faces are split. Two of these edges cannot belong to the same face, so the six faces of the hexahedron are split in triangles. An additional node is created at the centroid of the hexahedron. Eleven internal edges are created with that centroid as a vertex: three in connection with the centre of the three split edges, while the other eight with the eight vertices of the hexahedron. Eighteen tetrahedra are then created with the eighteen triangles
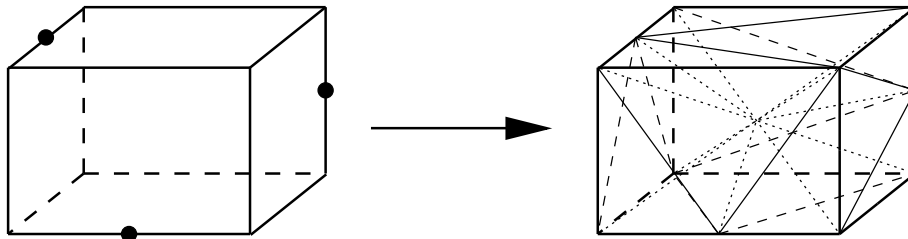
Figure 8: Three edges are split

- A face and its four edges are split. The opposite face of the hexahedron is kept intact, while the other four are split into triangles. Four internal edges are created from the centre of the regularly split face to the vertices of the opposite quadrangular face. A pyramid is then built on the intact quadrangular face of the hexahedron. Four more pyramids are also built on the four quadrangular faces produced by the refinement of the refined face. Last, four tetrahedra are created on the four remaining triangles that are located in the centre of the lateral faces.
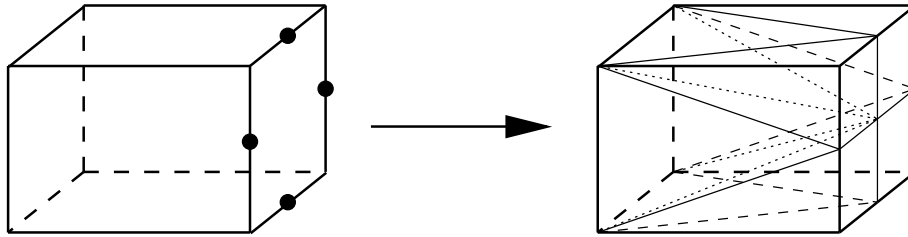
Figure 9: Four edges are split

3.2.3. Two comments about the pyramids

Usually, the hexahedra or the tetrahedra are preferred in a numerical simulation rather than the pyramids because of their properties. However, the pyramidal element offers an effective solution to maintain the compatibility between zones [8]. For the h-refinement, thanks to these pyramidal elements, there is no need to define interfacial relations. The algorithm is the same over the whole mesh, whatever the type of elements.

In a further iteration of the adaptive process, if nothing special is done, the pyramids could be split. There should be a risk in the quality of the mesh. To avoid that phenomenon, we give a temporary status to the transition elements. At the beginning of a new iteration, these transition elements are removed. The algorithm is applied over the plain elements. At the end, new transition elements are added to make a conformal mesh. Doing that, the transition elements are never split and the quality of the mesh is saved.

4. NUMERICAL APPLICATION

We illustrate this algorithm with the simulation of an industrial installation: the mechanical analysis of an arch dam, during the filling up. Two major parts are described: the foundation and the arch. The foundation represents the rocky part around the river valley and gives the stability to the structure. The arch made of concrete is modelled and the variation of the thickness is taken into account.

The initial mesh is mainly made of hexahedra, with a few prisms and tetrahedra in the conformal connection between the arch and the foundation (see Table 1 and Figure 11). The simulation is divided into two parts. Firstly, a calculation is done without any water: the objective is to get an initial state of the stresses in the calculation, considering its weight and the reaction of the foundation. Secondly, the level of the water rises in the upstream side, up to the top of the arch (see Figure 10). The pressure of the water on the upstream face of the arch modifies the field of the stresses and of the displacement in the arch.
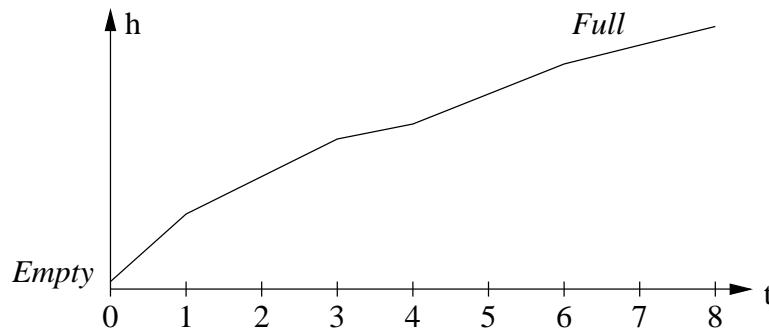
Figure 10 - Filling up of the dam
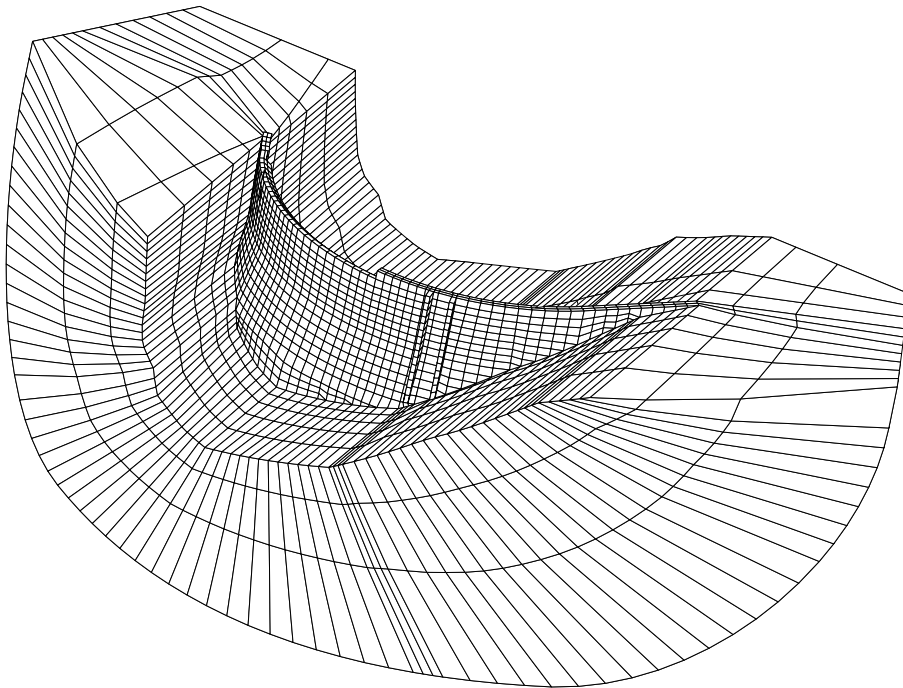
Figure 11 - Initial mesh of the dam, upstream side

The strategy of the adaptation is based on a succession of adaptation. The first calculation is made until time #1, when the first level of water is reached. At this point, the calculation is stopped and the distribution of the stresses and the displacement is examined. On every single element, the variation of the displacement between the element and its neighbours is computed and stored. The elements where this value is higher than the mean over the domain plus four times the square deviation are selected ([9], [10]). These selected elements are split at the beginning of the algorithm and the propagation is done until a conformal mesh is obtained. At this point, this new mesh is used for a second calculation from the very beginning of the building of the dam to the time #2 of the filling up. We operate the same

mechanism until time #8 (see Figure 12). Doing this, the mesh is adapted for every position of the upper level of the water.
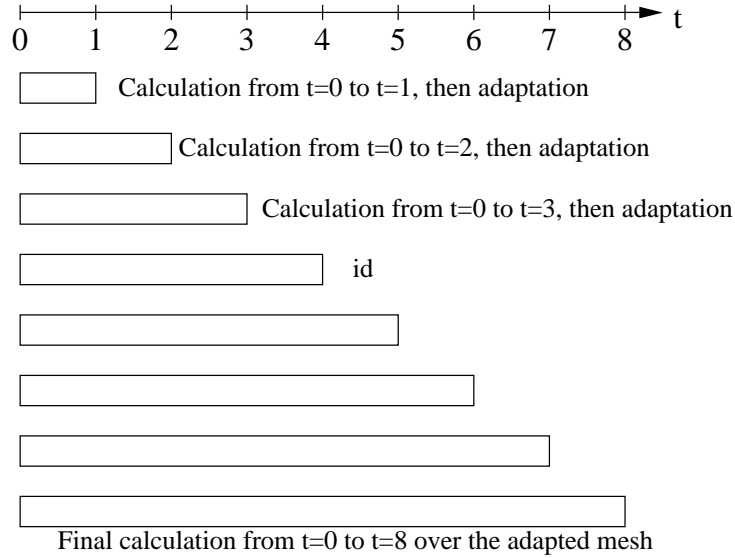
Figure 12 - The scheme of the adaptation

At the end of the process, the mesh is refined where it is necessary (see Figure 13), mainly at the centre and the bottom of the arch. The results of the simulation are similar to the reference. The number of degrees of freedom in the adapted mesh is much lower than in the reference mesh (see Table 1). This reduction allows using less memory and is very effective for large problems.

Mesh	Nodes	Hexahedra	Prisms	Tetrahedra	Pyramids
0	21 982	4 019	108	4	0
1	29 158	4 996	114	564	898
2	32 758	5 587	243	681	1 074
3	36 158	6 218	254	779	1 217
4	39 154	6 787	285	841	1 298
5	39 693	6 809	285	905	1 422
6	40 191	6 836	285	961	1 524
7	40 297	6 838	285	985	1 545
Reference	153 007	32 152	864	32	0

Table 1 - Number of elements during the adaptation
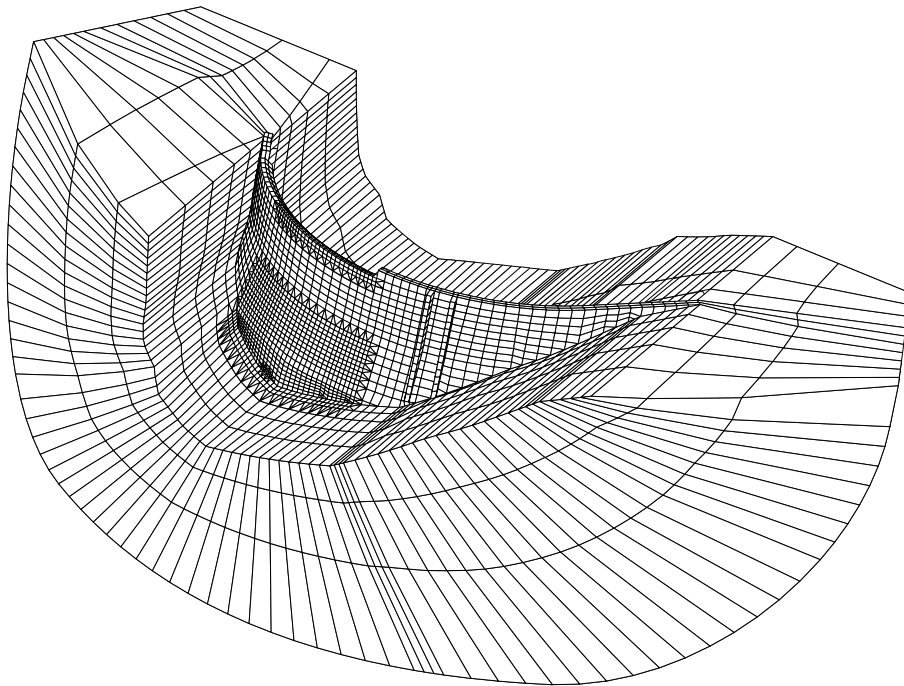
Figure 13 - Final mesh of the dam, upstream side

5. CONCLUSIONS

In this paper, we presented an h-refinement method for the adaptation of a conformal hexahedral mesh. The algorithm that propagates the decisions of splitting an element is governed by a simple rule on the faces: every quadrangular face that has at least two refined edges is to be refined. The conformal junction between two zones with a different level of refinement is achieved by a combination of tetrahedra and pyramids.

Some numerical experiments show that this technique is effective to increase the quality of the results. Thanks to this process, the confidence into the conclusion of a computation is improved.

REFERENCES

[1] S.H. Lo, D. Wu, K.Y. Sze, Adaptive meshing and analysis using transitional quadrilateral and hexahedral elements, *Finite Elements in Analysis and Design*, Vol. 46, pp. 2-16, 2010, doi:10.1016/j.finel.2009.06.010

[2] R. E. Bank, *PLTMG, A software package for solving elliptic partial differential equations, User's guide 6.0*, SIAM, Philadelphia, 1990.

[3] G. Nicolas and T. Fouquet, Adaptive Mesh Refinement for Conformal Hexahedral Meshes, *Finite Elements in Analysis and Design*, Vol. 67, pp. 1-12, 2013, doi:10.1016/j.finel.2012.11.008

[4] Salome web site: www.salome-platform.org

[5] G. Nicolas and T. Fouquet, Hexahedral mesh adaptation for finite volume methods, *ADMOS 07 Conference*, Goteborg 2007.

[6] *Code_Aster* web site: `www.code-aster.org`

[7] S. Prudhomme, J. T. Oden, On goal-oriented error estimation for elliptic problems: applications to the control of pointwise errors, *Comput. Methods Appl. Mech. Engrg.*, 176 (1999) 313-331.

[8] S. J. Owen, S. Saigal, Formation of pyramid elements for hexahedra to tetrahedral transitions, *Comput. Methods Appl. Mech. Engrg.*, 190 (2001) 4505-4518

[9] A. Chatziionnou, Y. Lu, A parallel adaptative finite element method for simulation of photon migration with the radiative-transfer-based model, *Numerical Methods in Engineering*, 25, Issue 6, pp. 751-770, 2009

[10] E. Shirani, M. Ameri, Combination of adaptative-grid-redistribustion and embedding methods, *Numerical Methods in Engineering*, 20, Issue 3, pp. 229-240, 2004

VI International Conference on Adaptive Modeling and Simulation
ADMOS 2013
J. P. Moitinho de Almeida, P. Díez, C. Tiago and N. Parés (Eds)

GOAL ORIENTED MESH ADAPTATION WITHOUT FINE GRID INTERPOLATION FOR FINITE-VOLUME CFD

J. PETER (ONERA), S. BOURASSEAU (SNECMA), M. NGUYEN-DINH (AIRBUS-F)

Abstract. In aeronautical CFD, engineers require accurate predictions of the forces and moments but they are less concerned with flow-field accuracy. Hence, the so-called "goal oriented" mesh adaptation strategies have been introduced to get satisfactory values of functional outputs at an acceptable cost, using local node displacement and insertion of new points rather than mesh refinement guided by uniform accuracy[2, 3, 4, 5, 6]. Most often, such methods involve the adjoint vector of the function of interest. Our purpose is to present goal oriented criteria of mesh quality and local mesh adaptation strategies in the framework of finite-volume schemes and a discrete adjoint vector method [1]. They are based on the total derivative of the goal with respect to (w.r.t.) mesh nodes. More precisely, a projection of the goal derivative, removing all components corresponding to geometrical changes in the solid walls or the support of the output [7]. The methods are assessed in the case of 2D and 3D Euler flow computations, with structured and unstructured meshes.

REFERENCES

[1] J. Peter, R.P. Dwight. Numerical sensitivity analysis: a survey of approaches. Computers and Fluids Vol.39 (3). pp. 373-391. 2010

[2] D.A. Venditti, D.L. Darmofal. Adjoint error estimation and grid adaptation for functional outputs: Application to quasi-one-dimensional flow. Journal of Computational Physics Vol. 164 pp. 40–69. 2000.

[3] D.A. Venditti, D.L. Darmofal. Grid adaptation for functional outputs: Application to two-dimensional inviscid flows. Journal of Computational Physics Vol. 176 pp. 40–69. 2002.

[4] D.A. Venditti, D.L. Darmofal. Anisotropic grid adaptation for functional outputs: Application to two-dimensional viscous flows. Journal of Computational Physics Vol. 187 pp. 22–46. 2003.

[5] R.P. Dwight. Goal-oriented mesh adaptation using a dissipation based error indicator International Journal for Numerical Methods in Fluids Vol.56(8) pp. 1193-2000. 2007.

[6] R.P. Dwight. Heuristic a posteriori estimation of error due to dissipation in finite volumes schemes and application to mesh adaptation. Journal of Computational Physics Vol. 227 pp. 2845–2863. 2008.

[7] J. Peter, M. Nguyen-Dinh, P. Trontin. Goal oriented mesh adaptation using total derivative of aero- dynamic functions with respect to mesh coordinates – With application to Euler flows. Computers and Fluids Vol. 72 pp. 194-214. 2012.

NEW BOUNDING TECHNIQUES FOR GOAL-ORIENTED ERROR ESTIMATION IN FE SIMULATIONS

F. PLED[*], L. CHAMOIN[†] AND P. LADEVEZE[†]

[*,†]LMT-Cachan (ENS-Cachan/CNRS/Paris 6 University)
61 Avenue du Président Wilson, 94235 Cachan Cedex, France
e-mail: {pled,chamoin,ladeveze}@lmt.ens-cachan.fr, www.lmt.ens-cachan.fr

Key words: Verification, Finite Element Method, Goal-oriented error estimation, Constitutive relation error, Saint-Venant's principle, Non-intrusive Techniques, Guaranteed Error Bounds

Abstract. In this work, we propose new bounding techniques that enable to derive accurate and strict error bounds on outputs of interest computed from numerical approximation methods such as the finite element method. These techniques are based on Saint-Venant's principle and exploit specific homotheticity properties in order to improve the quality of the bounds computed from the classical bounding technique. The capabilities of the proposed approaches are illustrated through two-dimensional numerical experiments carried out on a linear elasticity problem.

1 INTRODUCTION

In the context of finite element (FE) model verification, research and engineering activities focus on the development of robust goal-oriented error estimation methods designed to achieve strict and high-quality error bounds associated to specific quantities of interest. A general method [1] consists in using extraction techniques as well as robust global error estimation methods, and involves the global solution of an auxiliary problem, also known as dual or adjoint problem. The derivation of accurate local error bounds entails a fine resolution of this auxiliary problem. Nevertheless, the classical bounding technique may provide low-quality error bounds on specific quantities of interest, particularly when the global estimated errors related to both reference (primal) and adjoint (dual) problems are mainly concentrated in disjoint regions. The main source of overestimation presumably stems form the Cauchy-Schwarz inequality, especially when the zone of interest is located far from the predominant contributions of the global estimate associated to reference problem. This observation has spurred the development of new bounding techniques able to circumvent, or at least alleviate, this serious drawback by optimizing the sharpness and practical relevance of the classical computed bounds. In this work, we propose and ana-

lyze new improved bounding techniques based on non-classical and innovative tools, such as homotheticity properties [2]. These techniques are carefully tailored for the derivation of inequalities between appropriate quantities over two homothetic domains contained in the whole structure. Such relations are based on Saint-Venant's principle and seem to be limited to solely linear problems. The classical and enhanced techniques can be combined with an intrusive approach (local refinement techniques) or a non-intrusive one (handbook techniques [3]) to get a reliable solution of the adjoint problem.

The paper is organized as follows. Section 2 presents both reference and adjoint problems and defines the discretization error. Section 3 recalls basics on goal-oriented error estimation using extraction (or adjoint-based) techniques and the concept of constitutive relation error through the construction of admissible solutions. Section 4 describes the main features of the improved bounding techniques, while Section 5 provides some numerical experiments conducted on a linear elasticity problem with comparative results between conventional and alternative bounding techniques.

2 REFERENCE AND ADJOINT PROBLEMS

2.1 Reference problem and discretization error

Let us consider a mechanical structure occupying an open bounded domain $\Omega \subset \mathbb{R}^d$ (d being the space dimension), with Lipschitz boundary $\partial\Omega$. The prescribed loading acting on Ω consists of: a displacement field $\underline{U}_d$ on part $\partial_u\Omega \subset \partial\Omega$ ($\partial_u\Omega \neq \varnothing$); a traction force density $\underline{F}_d$ on the complementary part $\partial_f\Omega$ of $\partial\Omega$ such that $\overline{\partial_u\Omega \cup \partial_f\Omega} = \partial\Omega$, $\partial_u\Omega \cap \partial_f\Omega = \varnothing$; a body force field $\underline{f}_d$ within Ω. Structure Ω is assumed to be made of an isotropic, homogeneous material with linear and elastic behavior characterized by Hooke's tensor $\boldsymbol{K}$. Assuming a quasi-static loading, an isothermal case and a small perturbations state, the reference problem consists of finding a displacement/stress pair $(\underline{u}, \sigma)$ in the space domain Ω, which verifies:

- the kinematic conditions:
$$\underline{u} \in \boldsymbol{\mathcal{U}}; \quad \underline{u} = \underline{U}_d \quad \text{on } \partial_u\Omega; \quad \varepsilon(\underline{u}) = \frac{1}{2}\big(\mathbb{V}\underline{u} + \mathbb{V}^T\underline{u}\big) \quad \text{in } \Omega; \tag{1a}$$

- the weak form of equilibrium equations:
$$\sigma \in \boldsymbol{\mathcal{S}}; \quad \forall \underline{u}^* \in \boldsymbol{\mathcal{U}}_0, \quad \int_\Omega \mathrm{Tr}\big[\sigma\,\varepsilon(\underline{u}^*)\big]\,\mathrm{d}\Omega = \int_\Omega \underline{f}_d \cdot \underline{u}^*\,\mathrm{d}\Omega + \int_{\partial_f\Omega} \underline{F}_d \cdot \underline{u}^*\,\mathrm{d}S; \tag{1b}$$

- the constitutive relation:
$$\sigma = \boldsymbol{K}\,\varepsilon\big(\underline{u}\big) \quad \text{in } \Omega, \tag{1c}$$

where $\varepsilon(\underline{u})$ represents the classical linearized strain tensor corresponding to the symmetric part of the gradient of displacement field $\underline{u}$. Affine spaces $\boldsymbol{\mathcal{U}} = \big\{\underline{u} \in [\mathcal{H}^1(\Omega)]^d\big\}$ and $\boldsymbol{\mathcal{S}} = \big\{\sigma \in \mathcal{M}_s(d) \cap [\mathcal{L}^2(\Omega)]^{d^2}\big\}$ guarantee the existence of finite-energy solutions, $\mathcal{M}_s(d)$

representing the space of symmetric square matrices of order d. Lastly, $\boldsymbol{\mathcal{U}}_0 \subset \boldsymbol{\mathcal{U}}$ denotes the vectorial space associated to $\boldsymbol{\mathcal{U}}$.

In practical applications, the exact solution of the reference problem, hereafter denoted $(\underline{u}_{ex}, \sigma_{ex})$, remains usually out of reach and only an approximate solution, referred to as $(\underline{u}_h, \sigma_h)$, can be obtained through numerical approximation methods (such as the FE method (FEM) associated with a space mesh Ω_h mapping Ω). Such a numerical approximation is searched in a discretized space $\boldsymbol{\mathcal{U}}_h \times \boldsymbol{\mathcal{S}}_h \subset \boldsymbol{\mathcal{U}} \times \boldsymbol{\mathcal{S}}$. A displacement-type FEM leads to a displacement field $\underline{u}_h$ verifying kinematic constraints (1a) and a stress field σ_h computed *a posteriori* from constitutive relation (1c).

The resulting discretization error, denoted $\underline{e}_h = \underline{u}_{ex} - \underline{u}_h$, can be assessed in terms of:

- a global measure defined with respect to the classical energy norm $\|\bullet\|_{u,\Omega} = \left(\int_\Omega \mathrm{Tr} \left[\boldsymbol{K} \, \varepsilon(\bullet) \, \varepsilon(\bullet) \right] \mathrm{d}\Omega \right)^{1/2}$, providing a global discretization error $e_\Omega = \|\underline{e}_h\|_{u,\Omega}$;

- a local measure defined with respect to a specific output of interest $I(\underline{u})$ of the problem, providing a local error $e_I = I(\underline{u}_{ex}) - I(\underline{u}_h)$. Under the assumption of a linear quantity of interest with respect to displacement $\underline{u}$, it merely reads: $e_I = I(\underline{e}_h)$.

2.2 Adjoint problem

The quantity of interest, hereafter denoted I, is a goal-oriented output, such as the mean value of a stress component over a local region or the displacement value at a specific point, for instance. These meaningful quantities of practical interest to engineers are usually defined by means of extraction techniques, i.e. by expressing the local quantity I being considered in the global form involving global extraction operators, also called extractors. In this work, for the sake of simplicity, the quantity of interest is represented as a linear functional $\mathcal{L}$ of displacement field $\underline{u}$ on a finite support under the following global form:

$$I = \mathcal{L}(\underline{u}) = \int_\Omega \left(\mathrm{Tr} \left[\tilde{\sigma}_\Sigma \, \varepsilon(\underline{u}) \right] + \tilde{\underline{f}}_\Sigma \cdot \underline{u} \right) \mathrm{d}\Omega, \tag{2}$$

where so-called extractors $\tilde{\sigma}_\Sigma$ and $\tilde{\underline{f}}_\Sigma$, known analytically, can be mechanically viewed as a prestress field and a body force field, respectively. In the following, let $I_{ex} = \mathcal{L}(\underline{u}_{ex})$ and $I_h = \mathcal{L}(\underline{u}_h)$ be the unknown exact value of the quantity of interest I being studied and its approximate value obtained through the FEM, respectively.

Once the quantity of interest has been put into such a global form, the classical approach then consists of introducing an auxiliary problem, also called adjoint problem, which is similar to the reference problem, except that the external mechanical loading $(\underline{F}_d, \underline{f}_d)$ is replaced by the extractors on the one hand, and the non-homogeneous Dirichlet boundary conditions are changed to homogeneous kinematic constraints on the other hand. The adjoint problem consists of finding a displacement/stress pair $(\tilde{\underline{u}}, \tilde{\sigma})$, in the space domain Ω, which verifies:

- the kinematic conditions:

$$\tilde{\underline{u}} \in \boldsymbol{\mathcal{U}}_0; \tag{3a}$$

- the weak form of equilibrium equations:

$$\tilde{\sigma} \in \boldsymbol{\mathcal{S}}; \quad \forall \underline{u}^* \in \boldsymbol{\mathcal{U}}_0, \quad \int_\Omega \mathrm{Tr}\left[\tilde{\sigma}\varepsilon(\underline{u}^*)\right] \mathrm{d}\Omega = \mathcal{L}(\underline{u}^*) = \int_\Omega \left(\mathrm{Tr}\left[\tilde{\sigma}_\Sigma\, \varepsilon(\underline{u}^*)\right] + \underline{\tilde{f}}_\Sigma \cdot \underline{u}^*\right) \mathrm{d}\Omega; \tag{3b}$$

- the constitutive relation:

$$\tilde{\sigma} = \boldsymbol{K}\,\varepsilon\left(\tilde{\underline{u}}\right) \quad \text{in } \Omega. \tag{3c}$$

For similar reasons to the reference problem, the exact solution $(\tilde{\underline{u}}_{ex}, \tilde{\sigma}_{ex})$ of the adjoint problem remains out of reach in most practical applications, and one can only obtain an approximate solution, denoted $(\tilde{\underline{u}}_h, \tilde{\sigma}_h)$. This last solution lies in a discretized FE space associated with a space mesh $\tilde{\Omega}_h$, mapping the physical domain Ω.

3 BASICS ON GOAL-ORIENTED ERROR ESTIMATION BASED ON CONSTITUTIVE RELATION ERROR

We review here the classical procedure based on the concept of constitutive relation error (CRE) to obtain strict local error bounds on functional outputs.

3.1 Constitutive relation error

Starting from an admissible solution $(\hat{\underline{u}}_h, \hat{\sigma}_h)$ provided by one of the existing techniques [4], one can measure the global residual on constitutive relation (1c), called the CRE measure and denoted $e_{\mathrm{cre},\Omega} \equiv e_{\mathrm{cre},\Omega}(\hat{\underline{u}}_h, \hat{\sigma}_h) = \|\hat{\sigma}_h - \boldsymbol{K}\,\varepsilon(\hat{\underline{u}}_h)\|_{\sigma,\Omega}$, with $\|\bullet\|_{\sigma,\Omega} = \left(\int_\Omega \mathrm{Tr}\left[\bullet\,\boldsymbol{K}^{-1}\,\bullet\right]\mathrm{d}\Omega\right)^{1/2}$. Computing the CRE measure $e_{\mathrm{cre},\Omega}$ provides a guaranteed upper bound of the global discretization error $\|\underline{e}_h\|_{u,\Omega}$, as the well-known Prager-Synge hypercircle theorem leads to the following bounding inequality:

$$\|\underline{e}_h\|_{u,\Omega}^2 = \|\underline{u}_{ex} - \hat{\underline{u}}_h\|_{u,\Omega}^2 \leqslant \|\underline{u}_{ex} - \hat{\underline{u}}_h\|_{u,\Omega}^2 + \|\sigma_{ex} - \hat{\sigma}_h\|_{\sigma,\Omega}^2 = e_{\mathrm{cre},\Omega}^2, \tag{4}$$

which conveys the guaranteed nature of the CRE measure $e_{\mathrm{cre},\Omega}$ with respect to the global discretization error.

Introducing the average admissible field $\hat{\sigma}_h^m = \frac{1}{2}\left(\hat{\sigma}_h + \boldsymbol{K}\,\varepsilon(\hat{\underline{u}}_h)\right)$, one can directly deduce another fundamental relation, called the Prager-Synge's equality:

$$\|\sigma_{ex} - \hat{\sigma}_h^m\|_{\sigma,\Omega} = \frac{1}{2} e_{\mathrm{cre},\Omega}. \tag{5}$$

Equations (4) and (5) are key relations to derive guaranteed error bounds in both global and local robust error estimation methods.

In the same manner as for the reference problem, an admissible solution of the adjoint problem, hereafter referred to as $(\hat{\tilde{\underline{u}}}_h, \hat{\tilde{\sigma}}_h)$, can be derived from one of the existing equilibration techniques. Then, the associated CRE measure $\tilde{e}_{\text{cre},\Omega} \equiv e_{\text{cre},\Omega}(\hat{\tilde{\underline{u}}}_h, \hat{\tilde{\sigma}}_h)$ of the adjoint problem can be computed leading to a global estimate of the discretization error $\tilde{\underline{e}}_h = \tilde{\underline{u}}_{ex} - \tilde{\underline{u}}_h$ of the adjoint problem.

Now, let us focus on the main principles of the classical bounding technique involved in goal-oriented error estimation methods based on extraction techniques and the concept of CRE.

3.2 Basic identity and classical bounding technique

The expression of the quantity of interest I reformulated in the global form (2) as well as properties of both admissible solutions $(\hat{\underline{u}}_h, \hat{\sigma}_h)$ and $(\hat{\tilde{\underline{u}}}_h, \hat{\tilde{\sigma}}_h)$ lead to the following basic identity:

$$I_{ex} - I_h - I_{hh} = \left\langle \sigma_{ex} - \hat{\sigma}_h^m, \hat{\tilde{\sigma}}_h - \boldsymbol{K}\,\varepsilon(\hat{\tilde{\underline{u}}}_h) \right\rangle_{\sigma,\Omega}, \tag{6}$$

where $\langle \bullet, \circ \rangle_{\sigma,\Omega} = \int_\Omega \text{Tr}\left[\bullet \boldsymbol{K}^{-1} \circ\right] d\Omega$ is an energetic inner product defined on the stress field space $\boldsymbol{\mathcal{S}}$. I_{hh} can be viewed as a computable correction term involving known quantities of both reference and adjoint problems:

$$I_{hh} = \left\langle \hat{\tilde{\sigma}}_h^m, \hat{\sigma}_h - \boldsymbol{K}\,\varepsilon(\hat{\underline{u}}_h) \right\rangle_{\sigma,\Omega} + \mathcal{L}(\hat{\underline{u}}_h - \underline{u}_h), \tag{7}$$

where $\hat{\tilde{\sigma}}_h^m = \dfrac{1}{2}(\hat{\tilde{\sigma}}_h + \boldsymbol{K}\,\varepsilon(\hat{\tilde{\underline{u}}}_h))$. $I_h + I_{hh}$ can be interpreted as a new approximate solution of the exact value I_{ex} of the quantity of interest.

The fundamental equality (6), which does not call for any orthogonality property of the FE solutions and allows to build the finite-dimensional spaces associated to reference and adjoint problems independently, is the keystone of the classical bounding technique as well as the improved ones described in section 4.

Subsequently, the classical bounding procedure merely consists of applying the Cauchy-Schwarz inequality to (6) with respect to inner product $\langle \bullet, \circ \rangle_{\sigma,\Omega}$ and then using Prager-Synge's equality (5). This yields:

$$|I_{ex} - I_h - I_{hh}| \leqslant \frac{1}{2}\,e_{\text{cre},\Omega}\,\tilde{e}_{\text{cre},\Omega}. \tag{8}$$

Eventually, the derivation of strict lower and upper bounds $(\xi_{\text{inf}}, \xi_{\text{sup}})$ of I_{ex} (or, equivalently, of the local error $I_{ex} - I_h$) can be achieved straightforwardly, just having a global error estimation procedure at hand:

$$\xi_{\text{inf}} \leqslant I_{ex} \leqslant \xi_{\text{sup}}, \tag{9}$$

with

$$\begin{aligned} \xi_{\text{inf}} &= I_h + I_{hh} - \frac{1}{2}\,e_{\text{cre},\Omega}\,\tilde{e}_{\text{cre},\Omega}; \\ \xi_{\text{sup}} &= I_h + I_{hh} + \frac{1}{2}\,e_{\text{cre},\Omega}\,\tilde{e}_{\text{cre},\Omega}. \end{aligned} \tag{10}$$

Besides, owing to the independent natures of spatial discretizations associated to reference and adjoint problems, a convenient way to achieve accurate and sharp bounds of I_{ex} is to perform a local space refinement of the adjoint mesh $\tilde{\Omega}_h$ alone around the zone of interest ω in order to properly solve the adjoint problem while keeping a reasonable computational cost. In most common situations, the discretization error related to the adjoint problem is concentrated in the vicinity of the zone of interest, whereas that related to the reference problem may be scattered around zones which present some singularities or other error sources. However, when the error related to the reference problem is mostly located outside and far from the zone of interest, the classical bounding technique may yield large and low-quality local error bounds and thus makes useless bounding result (9). This is the point that we are revisiting here.

The proposed bounding techniques we present in the following section are intended to get around this serious drawback proper to the classical technique in order to sharpen the local error bounds.

4 IMPROVED BOUNDING TECHNIQUES

4.1 Homotheticity transfromation

Let us consider a reference subdomain, denoted ω_1 and included in Ω, defined by a point $\underline{O}$ and a geometric shape. The set of homothetic domains ω_λ associated to ω_1 is defined as:

$$\omega_\lambda = \mathcal{H}_{[\underline{O},\lambda]}(\omega_1) \tag{11}$$

where $\mathcal{H}_{[\underline{O};\lambda]}$ stands for the homothetic transformation operator centered in point $\underline{O}$, called homothetic center, and parameterized by a nonzero positive scalar $\lambda \in \,]0\,,\lambda_{\max}]$, also called magnification ratio, scale factor or similitude ratio, such that $\omega_\lambda \subset \Omega$ (see Figure 1). The geometric shape defining the set of homothetic domains ω_λ can be chosen arbitrarily. Nevertheless, these physical domains are supposed to be basic in practice, such as a circle or a rectangle in 2D, and a sphere or a rectangular cuboid (also called rectangular parallelepiped or right rectangular prism) in 3D, for instance.

For a given pair $(\omega_\lambda, \omega_{\bar{\lambda}})$ of homothetic domains included in Ω, represented in Figure 1 and parameterized by $(\lambda, \bar{\lambda})$, such that $\omega_\lambda \subset \omega_{\bar{\lambda}} \subset \Omega$, i.e. $\lambda \in \,]0\,,\bar{\lambda}]$, the position $\underline{v}_\lambda$ of a point $\underline{M}_\lambda$ along boundary $\partial\omega_\lambda$ can be defined from the position $\underline{v}_{\bar{\lambda}}$ of the corresponding point $\underline{M}_{\bar{\lambda}}$ along boundary $\partial\omega_{\bar{\lambda}}$ by the following relation:

$$\underline{v}_\lambda = \begin{cases} \dfrac{\lambda}{\bar{\lambda}}\,\underline{v}_{\bar{\lambda}}(\bar{s}) & \text{parameterized by } (\lambda, \bar{s}) \text{ in 2D;} \\[2ex] \dfrac{\lambda}{\bar{\lambda}}\,\underline{v}_{\bar{\lambda}}(\bar{s}_1, \bar{s}_2) & \text{parameterized by } (\lambda, \bar{s}_1, \bar{s}_2) \text{ in 3D,} \end{cases} \tag{12}$$

where $\bar{s}$ (resp. $\bar{s}_1$ and $\bar{s}_2$) represent the curvilinear abscissa along boundary $\partial\omega_{\bar{\lambda}}$ in 2D (resp. 3D).

Such a parameterization leads to various homotheticity properties [2] that are at the root of fundamental inequalities such as the one introduced in Section 4.3.
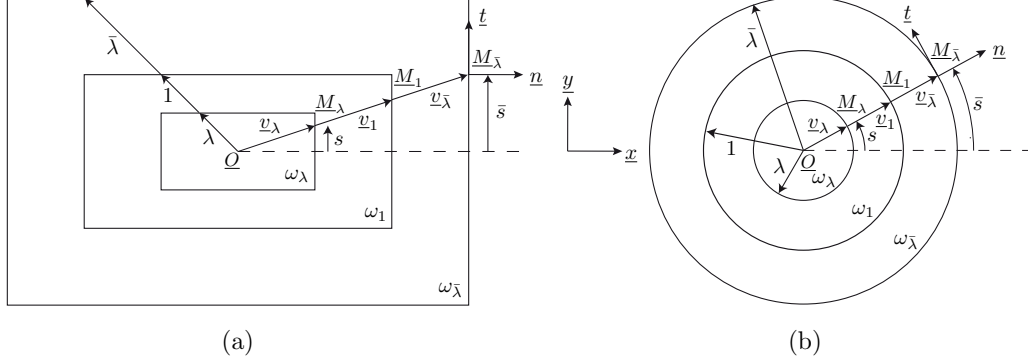
Figure 1: Examples of rectangular (a) and circular (b) homothetic domains in 2D.

4.2 Principle

First, let us recall the expression of quantity q involved in basic identity (6) for building local error bounds :

$$q = \left\langle \sigma_{ex} - \hat{\sigma}_h^m, \hat{\tilde{\sigma}}_h - \boldsymbol{K}\,\varepsilon(\hat{\tilde{\underline{u}}}_h) \right\rangle_{\sigma,\Omega}, \tag{13}$$

where $\hat{\sigma}_h^m$ and $\hat{\tilde{\sigma}}_h - \boldsymbol{K}\,\varepsilon(\hat{\tilde{\underline{u}}}_h)$ are given quantities coming from reference and adjoint problems, respectively, and σ_{ex} is the unknown exact stress solution of the reference problem.

By considering a subdomain ω_λ of domain Ω and its complementary part $\Omega \setminus \omega_\lambda$, the approach consists in splitting quantity q into two distinct contributions q_{ω_λ} and $q_{\Omega \setminus \omega_\lambda}$:

$$q = q_{\omega_\lambda} + q_{\Omega \setminus \omega_\lambda},$$

where

$$q_{\omega_\lambda} = \left\langle \sigma_{ex} - \hat{\sigma}_h^m, \hat{\tilde{\sigma}}_h - \boldsymbol{K}\,\varepsilon(\hat{\tilde{\underline{u}}}_h) \right\rangle_{\sigma,\omega_\lambda}; \tag{14a}$$

$$q_{\Omega \setminus \omega_\lambda} = \left\langle \sigma_{ex} - \hat{\sigma}_h^m, \hat{\tilde{\sigma}}_h - \boldsymbol{K}\,\varepsilon(\hat{\tilde{\underline{u}}}_h) \right\rangle_{\sigma,\Omega \setminus \omega_\lambda}. \tag{14b}$$

When quantity $\hat{\tilde{\sigma}}_h - \boldsymbol{K}\,\varepsilon(\hat{\tilde{\underline{u}}}_h)$ involved in the CRE measure of the adjoint problem is mostly concentrated over part ω_λ, i.e. by choosing a subdomain ω_λ surrounding the zone of interest ω, part $q_{\Omega \setminus \omega_\lambda}$ can be accurately bounded by simply using the Cauchy-Schwarz inequality with respect to inner product $\langle \bullet, \circ \rangle_{u,\Omega \setminus \omega_\lambda}$ and the Prager-Synge's equality (5):

$$\left| q_{\Omega \setminus \omega_\lambda} \right| \leqslant \left\| \sigma_{ex} - \hat{\sigma}_h^m \right\|_{\sigma,\Omega \setminus \omega_\lambda} \tilde{e}_{\mathrm{cre},\Omega \setminus \omega_\lambda} \leqslant \frac{1}{2}\,e_{\mathrm{cre},\Omega}\,\tilde{e}_{\mathrm{cre},\Omega \setminus \omega_\lambda}. \tag{15}$$

Given that the discretization error associated to the adjoint problem is mainly located around the zone of interest $\omega \subset \omega_\lambda$, $\tilde{e}_{\mathrm{cre},\Omega \setminus \omega_\lambda}$ is a relatively small computable term. It follows that the main contribution to the local error comes from part q_{ω_λ}. Consequently, quantity q_{ω_λ} has to be bounded with a particular care in order to derive accurate local error bounds while preserving the guaranteed nature of the error estimate.

4.3 Fundamental inequality

Let us consider the space $\boldsymbol{\mathcal{V}}$ of functions satisfying homogeneous equilibrium conditions:

$$\boldsymbol{\mathcal{V}} = \{\underline{v} \in \boldsymbol{\mathcal{U}} \text{ such that } \underline{\mathrm{div}}(\boldsymbol{K}\,\varepsilon(\underline{v})) = \underline{0} \text{ in } \Omega\}, \tag{16}$$

and let us introduce the Steklov constant, or Steklov eigenvalue, h defined in [5] as:

$$h = \max_{\underline{v} \in \boldsymbol{\mathcal{V}}_{|\omega_1}} \boldsymbol{\mathcal{S}}_1(\underline{v}), \tag{17}$$

with

$$\boldsymbol{\mathcal{S}}_1(\underline{v}) = \frac{\left\| \boldsymbol{K}\,(\underline{v} \otimes \underline{n})_{\mathrm{sym}} \right\|_{\sigma,\partial\omega_1}^2}{\|\underline{v}\|_{u,\omega_1}^2}, \tag{18}$$

where $(\bullet)_{sym}$ represents the symmetric part of tensor of order 2 (or matrix) $\bullet$. Then, for any homothetic domain $\omega_\lambda \subset \Omega$ parametrized by $\lambda > 0$, one can derive a relation involving the product of constant h and parameter λ:

$$h\lambda = \max_{\underline{v} \in \boldsymbol{\mathcal{V}}_{|\omega_\lambda}} \boldsymbol{\mathcal{S}}_\lambda(\underline{v}), \tag{19}$$

with

$$\boldsymbol{\mathcal{S}}_\lambda(\underline{v}) = \frac{\left\| \boldsymbol{K}\,(\underline{v} \otimes \underline{n})_{\mathrm{sym}} \right\|_{\sigma,\partial\omega_\lambda}^2}{\|\underline{v}\|_{u,\omega_\lambda}^2}. \tag{20}$$

Let $(\omega_\lambda, \omega_{\bar{\lambda}})$ be a pair of homothetic domains such that $\lambda \in \,]0\,,\bar{\lambda}]$, i.e. $\omega_\lambda \subset \omega_{\bar{\lambda}}$. The following key inequality holds [2]:

$$\|\sigma_{ex} - \hat{\sigma}_h\|_{\sigma,\omega_\lambda}^2 \leqslant \left(\frac{\lambda}{\bar{\lambda}} \right)^{1/h} \|\sigma_{ex} - \hat{\sigma}_h\|_{\sigma,\omega_{\bar{\lambda}}}^2 + \gamma_{\lambda,\bar{\lambda}}, \tag{21}$$

where

$$\gamma_{\lambda,\bar{\lambda}} \equiv \gamma_{\lambda,\bar{\lambda}}(\hat{\underline{u}}_h, \hat{\sigma}_h) = \int_{\lambda'=\lambda}^{\bar{\lambda}} \left[\left(\frac{\lambda'}{\lambda} \right)^{-1/h} \frac{1}{h\lambda'} e_{\mathrm{cre},\omega_{\lambda'}}^2 \right] \mathrm{d}\lambda'. \tag{22}$$

Let us note that, using Prager-Synge's equality (5), unknown term $\|\sigma_{ex} - \hat{\sigma}_h\|_{\sigma,\omega_{\bar{\lambda}}}$ involved in the right-hand side term of fundamental inequality (21) is readily bounded as:

$$\|\sigma_{ex} - \hat{\sigma}_h\|_{\sigma,\omega_{\bar{\lambda}}}^2 \leqslant \left(\|\sigma_{ex} - \hat{\sigma}_h^m\|_{\sigma,\omega_{\bar{\lambda}}} + \|\hat{\sigma}_h^m - \hat{\sigma}_h\|_{\sigma,\omega_{\bar{\lambda}}} \right)^2 \leqslant \frac{1}{4}\left(e_{\mathrm{cre},\Omega} + e_{\mathrm{cre},\omega_{\bar{\lambda}}} \right)^2 \tag{23}$$

It follows that fundamental result (21) can be rewritten in terms of perfectly known quantities as:

$$\|\sigma_{ex} - \hat{\sigma}_h\|_{\sigma,\omega_\lambda}^2 \leqslant \left(\frac{\lambda}{\bar{\lambda}} \right)^{1/h} \frac{1}{4}\left(e_{\mathrm{cre},\Omega} + e_{\mathrm{cre},\omega_{\bar{\lambda}}} \right)^2 + \gamma_{\lambda,\bar{\lambda}}. \tag{24}$$

This last inequality is the key point to derive sharp bounds for part q_{ω_λ}.

4.4 Final bounding result

Applying the Cauchy-Schwarz inequality with respect to scalar product $\langle \bullet, \circ \rangle_{\sigma, \omega_\lambda}$ and then using the key inequality (24) introduced in the previous section leads to the following bounding result:

$$|q_{\omega_\lambda} - I_{hhh,\lambda}| \leqslant \left[\left(\frac{\lambda}{\bar{\lambda}} \right)^{1/h} \frac{1}{4} \left(e_{\text{cre},\Omega} + e_{\text{cre},\omega_{\bar{\lambda}}} \right)^2 + \gamma_{\lambda,\bar{\lambda}} \right]^{1/2} \tilde{e}_{\text{cre},\omega_\lambda}. \tag{25}$$

Collecting both inequalities (25) and (15) for parts q_{ω_λ} and $q_{\Omega \backslash \omega_\lambda}$, respectively, one obtains:

$$|I_{ex} - I_h - I_{hh} - I_{hhh,\lambda}| \leqslant \tilde{e}_{\text{cre},\omega_\lambda} \, \delta_{\lambda,\bar{\lambda}} + \frac{1}{2} \, e_{\text{cre},\Omega} \, \tilde{e}_{\text{cre},\Omega \backslash \omega_\lambda}, \tag{26}$$

where

$$\delta_{\lambda,\bar{\lambda}} \equiv \delta_{\lambda,\bar{\lambda}}(\hat{\underline{u}}_h, \hat{\sigma}_h) = \left[\left(\frac{\lambda}{\bar{\lambda}} \right)^{1/h} \frac{1}{4} \left(e_{\text{cre},\Omega} + e_{\text{cre},\omega_{\bar{\lambda}}} \right)^2 + \gamma_{\lambda,\bar{\lambda}} \right]^{1/2} \tag{27}$$

and

$$I_{hhh,\lambda} = \frac{1}{2} \left\langle \hat{\sigma}_h - \boldsymbol{K} \, \varepsilon(\hat{\underline{u}}_h), \hat{\bar{\sigma}}_h - \boldsymbol{K} \, \varepsilon(\hat{\bar{\underline{u}}}_h) \right\rangle_{\sigma, \omega_\lambda} \tag{28}$$

are fully calculable from the computed approximate solutions of both reference and adjoint problems.

Thus, this improved technique provides the following guaranteed lower and upper bounds $(\chi_{\text{inf}}, \chi_{\text{sup}})$ of I_{ex}:

$$\chi_{\text{inf}} \leqslant I_{ex} \leqslant \chi_{\text{sup}}, \tag{29}$$

with

$$\chi_{\text{inf}} = I_h + I_{hh} + I_{hhh,\lambda} - \left| \tilde{e}_{\text{cre},\omega_\lambda} \, \delta_{\lambda,\bar{\lambda}} + \frac{1}{2} \, e_{\text{cre},\Omega} \, \tilde{e}_{\text{cre},\Omega \backslash \omega_\lambda} \right|; \tag{30a}$$

$$\chi_{\text{sup}} = I_h + I_{hh} + I_{hhh,\lambda} + \left| \tilde{e}_{\text{cre},\omega_\lambda} \, \delta_{\lambda,\bar{\lambda}} + \frac{1}{2} \, e_{\text{cre},\Omega} \, \tilde{e}_{\text{cre},\Omega \backslash \omega_\lambda} \right|. \tag{30b}$$

These bounds depend on both parameters λ and $\bar{\lambda}$ associated to subdomains ω_λ and $\omega_{\bar{\lambda}}$, respectively. In order to get a practical minimizer, one seeks to reduce ratio $\lambda/\bar{\lambda}$ as much as possible by choosing: the smallest parameter λ such that domain ω_λ surrounds the zone of interest ω; the largest parameter $\bar{\lambda}$ such that domain $\omega_{\bar{\lambda}}$ remains a homothetic mapping of ω_λ (preserving its geometric shape) contained in Ω, and leading to sharp error bounds.

A second improved technique has been introduced and relies on similar homotheticity arguments, but differs from the first one presented in this paper in the way of bounding part q_{ω_λ}, which involves another fundamental inequality. The interested reader can refer to [2] for more information. This alternative bounding technique leads to guaranteed

lower and upper bounds $(\zeta_{\inf}, \zeta_{\sup})$ of I_{ex}. Those bounds involve only one parameter $\bar{\lambda}$, which can be chosen such that subdomain $\omega_{\bar{\lambda}}$ recovers the zone where the solution of adjoint problem has stiff gradients.

5 NUMERICAL RESULTS

All numerical experiments have been performed assuming that the material remains isotropic, homogeneous, linear and elastic with Young's modulus $E = 1$ and Poisson's ratio $\nu = 0.3$. Furthermore, the two-dimensional examples are assumed to satisfy the plane-stress approximation. The balance technique used to derive a statically admissible stress field is the element equilibration technique (EET) combined with a p-refinement technique consisting of a $p + k$ discretization, p being the FE interpolation degree and k an additional degree equal to 3 (see [4]).

Performances of the proposed bounding techniques are illustrated through the two-dimensional cracked structure of Figure 2, which presents two round cavities. A homogeneous Dirichlet boundary condition is imposed to the bigger circular hole, whereas a unit internal constant pressure p_0 is applied to the smaller one. Furthermore, the top-left edge is subjected to a unit normal traction force density $\underline{t} = +\underline{n}$. Besides, a single edge crack emanates from the bottom of the smaller cavity. The two lips of this crack as well as the remaining sides are traction-free boundaries. The FE mesh Ω_h consists of 7 751 linear triangular elements and 4 122 nodes (i.e. 8 244 d.o.f.), see Figure 2. The reference mesh $\Omega_{\bar{h}}$ used to compute an "overkill" solution and to define a "quasi-exact" value, denoted I_{ex} for convenience, of the quantity of interest is built up by dividing each element into 256 elements; thereby, it is made of 1 984 256 linear triangular elements and 996 080 nodes (i.e. 1 992 160 d.o.f.).
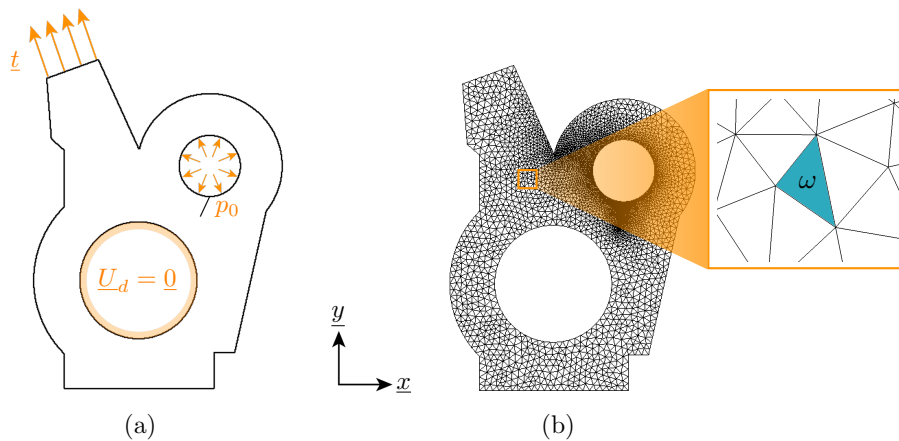

Figure 2: Cracked structure model problem (a) and associated finite element mesh (b).

The quantity of interest being considered in this work is a linear function of displacement field $\underline{u}$ associated to reference problem. It is the average value of the component

σ_{xx} of the stress field σ in a local zone $\omega \subset \Omega$:

$$I = \langle \sigma_{xx} \rangle_\omega = \frac{1}{|\omega|} \int_\omega \sigma_{xx} \, d\Omega, \tag{31}$$

where extraction domain ω corresponds to an element of FE mesh Ω_h illustrated in Figure 2 and $|\omega|$ represents its measure. The loading of the the adjoint problem involves an extractor corresponding to a uniform prestress field $\tilde{\sigma}_\Sigma = \boldsymbol{K} \, \tilde{\varepsilon}_\Sigma$ over element ω, where $\tilde{\varepsilon}_\Sigma = \frac{1}{|\omega|} \underline{x} \otimes \underline{x}$.

The main contributions to the error estimate $e_{\mathrm{cre},\Omega}$ associated to reference problem are by a majority located near the crack tip, while that to the error estimate $\tilde{e}_{\mathrm{cre},\Omega}$ associated to adjoint problem are concentrated around the zone of interest ω. Therefore, the error estimates for both reference and adjoint problems are localized in disjoint regions.

The homothetic center $\underline{O}$ coincides with the center of the circle $\mathcal{C}_\omega$ circumscribed by element ω and the values of parameters $\underline{\lambda}$ and $\bar{\lambda}$ involved in the first improved technique are set to $2\,r_{\mathcal{C}_\omega}$ and $14\,r_{\mathcal{C}_\omega}$, respectively, where $r_{\mathcal{C}_\omega}$ corresponds to the radius of $\mathcal{C}_\omega$. The value of parameter $\bar{\lambda}_{\mathrm{opt}}$ involved in the second improved technique is set to $9\,r_{\mathcal{C}_\omega}$, which enables to achieve the sharpest bounds for quantity of interest I.

The results obtained for classical bounding technique as well as first and second improved variants are presented in terms of the normalized bounds $(\bar{\xi}_{\mathrm{inf}}, \bar{\xi}_{\mathrm{sup}})$, $(\bar{\chi}_{\mathrm{inf}}, \bar{\chi}_{\mathrm{sup}})$, $(\bar{\zeta}_{\mathrm{inf}}, \bar{\zeta}_{\mathrm{sup}})$, respectively, with respect to I_{ex}. Figure 3 shows the evolutions of the normalized lower and upper bounds of $\tilde{I}_{ex}$ for quantity of interest I as functions of the number of elements $\tilde{N}_e$ contained in the FE mesh $\Omega_{\tilde{h}}$ associated to adjoint problem for the classical bounding technique as well as the two improved ones. The adjoint mesh $\Omega_{\tilde{h}}$ has been locally refined near the zone of interest ω, since the loading and the contributions to the global error estimate of the adjoint problem are highly localized in this region. One can see a slight improvement in the bounds obtained with the first improved technique compared to the classical one. As regards the second improved technique, a very clear improvement is observed allowing to achieve sharp local error bounds without refining too much the adjoint problem, thus keeping an affordable computing time.

6 CONCLUSION AND PROSPECTS

In this paper, we introduced new approaches related to the general framework of robust goal-oriented error estimation dealing with extraction techniques. These techniques are based on mathematical tools which are not classical in model verification. Various linear quantities of interest (such as the local average of a stress component, the pointwise value of a displacement component or a stress intensity factor) are considered in [2] to illustrate the effectivity of the proposed techniques. Those numerical experiments clearly demonstrate the efficiency of these methods to produce strict and relevant bounds on the errors in linear local quantities of interest compared to the classical bounding technique, especially when the discretization error related to the reference problem is not concentrated in
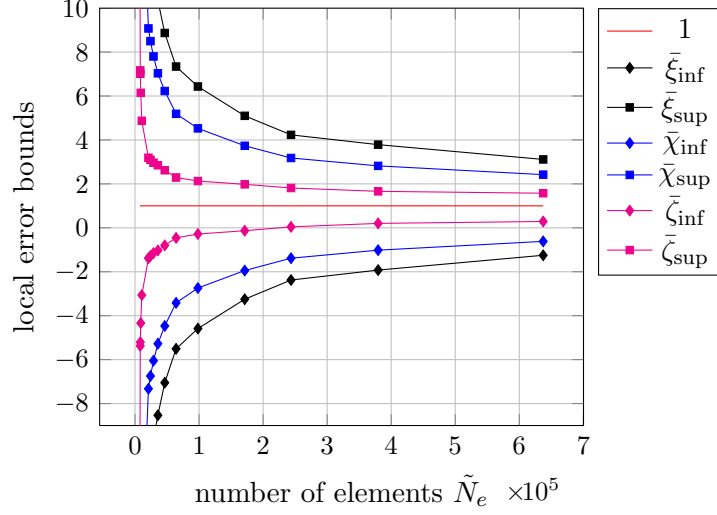
Figure 3: Evolutions of the lower and upper normalized bounds of I_{ex} for local quantity I_1, obtained using the classical bounding technique as well as first and second improvements, with respect to the number of elements $\tilde{N}_e$ associated to the discretization of the adjoint problem.

the local zone of interest. Nevertheless, the second proposed technique seems to achieve sharper local error estimates than the first one. Finally, such powerful methods may open up opportunities and help widen the field of robust goal-oriented error estimation methods. Both techniques could be easily extended to other quantities of interest but are restricted to linear problems, i.e. cases where Saint-Venant's principle is well established.

REFERENCES

[1] R. Becker and R. Rannacher. A feed-back approach to error control in finite element methods: Basic analysis and examples. *Journal of Numerical Mathematics*, 4:237–264, 1996.

[2] P. Ladevèze, F. Pled, and L. Chamoin. New bounding techniques for goal-oriented error estimation applied to linear problems. *International Journal for Numerical Methods in Engineering*, 93(13):1345–1380, 2013.

[3] L. Chamoin and P. Ladevèze. A non-intrusive method for the calculation of strict and efficient bounds of calculated outputs of interest in linear viscoelasticity problems. *Computer Methods in Applied Mechanics and Engineering*, 197(9-12):994–1014, 2008.

[4] P. Ladevèze and J. P. Pelle. *Mastering Calculations in Linear and Nonlinear Mechanics.* Springer, New York, 2004.

[5] M. W. Steklov. Sur les problèmes fondamentaux de la physique mathématique. *Annales Scientifiques de l'École Normale Supérieure*, 19:455–490, 1902.

IDENTIFICATION OF ADMITTANCE COEFFICIENTS FROM IN-SITU MEASUREMENTS IN ACOUSTIC CAVITIES

A. PROGNEAUX[†*], PH. BOUILLARD[†], A. DERAEMAEKER[†], E.FOLTÊTE[‡] AND M. OUISSE[‡]

[†]Université Libre de Bruxelles, Building, Architecture and Town Planning department
Avenue F.D. Roosevelt, 50 CP 194/02, B-1050 Bruxelles, Belgium
[*] e-mail: aprognea@ulb.ac.be

[‡]FEMTO-ST Institute, Department of Applied Mechanics
Rue de l'Epitaphe 24, F-25000 Besançon, France

Key words: Admittance coefficient, acoustical numerical model, updating technique, in-situ measurements

Abstract. This paper proposes a method of identification of the admittance coefficient, from in-situ measurements, by applying the CRE-based updating technique to the acoustical problem. Local estimators are developed to localize defective sensors. The process is illustrated on a 1D case.

1 INTRODUCTION

In recent decades, sound intensity and quality are taking an increasingly important place in the design process of products like cars or aircrafts, and different types of absorbing materials have therefore been developed and used in such products to achieve this purpose. To predict the influence of absorbing materials on the sound propagation inside cavities, industries generally use numerical tools, in which the acoustical properties of absorbing materials are described by the admittance coefficient. However, the conditions in which these parameters are measured can differ significantly from the ones in which the materials are really used. In this paper, the parameters required to describe admittance coefficients are identified, from in-situ measurements, by using the updating technique based on the CRE [1]. The main advantages of this method are that the updated parameters keep a physical meaning, that it allows taking into account the measurement error and that it allows locally evaluating the modeling and measurement errors [2]. The CRE-based updating method is therefore applied to the acoustical problem and the process is applied on simple 1D test case.

2 ACOUSTICAL FORMULATION

2.1 Acoustical problem

Let us consider an acoustical domain Ω with boundary $\partial\Omega$. The pressure field is the solution of the Helmholtz equation (1) with associated Dirichlet (2) and generalized (3) boundary conditions.

$$\Delta p + k^2 p = 0 \text{ in } \Omega \tag{1}$$

$$p = p \text{ on } \partial_D\Omega \tag{2}$$

$$v_n = \frac{j}{\omega\rho}\frac{\partial p}{\partial n} = \lambda A_n p + (1-\lambda)\bar{v}_n \text{ on } \partial_G\Omega \tag{3}$$

where $k = \frac{\omega}{c}$ is the wave number, ω is the angular frequency, ρ is the density of the fluid, A_n is the admittance coefficient, describing the absorbing properties of the materials, $\bar{v}_n$ is the normal component of the prescribed velocity and λ is a parameter allowing to define the nature of the boundary ($\lambda = 0$ for a vibrating border, $\lambda = 1$ for an absorbing border and $0 < \lambda < 1$ for a border at the same time vibrating and absorbing)

2.2 Construction of the error

The principle of CRE-based updating technique is to split the set of mathematical equation into a set of reliable equations and a set of less-reliable equations on which the CRE is constructed. In the acoustical problem, the less-reliable equation is the generalized boundary condition and the CRE is expressed by

$$\xi_\omega^2 = \frac{\omega^2\rho^2}{D_\omega^2}\int_{\partial_G\Omega}(v_n - \lambda A_n p - (1-\lambda)\bar{v}_n)^*(v_n - \lambda A_n p - (1-\lambda)\bar{v}_n)d\Gamma \tag{4}$$

where D_ω^2 is a normalization factor. To take the errors of the measurements into account, the modified CRE is defined by

$$e_\omega^2 = \xi_\omega^2 + \frac{r}{1-r}\eta_\omega^2 \tag{5}$$

where

$$\eta_\omega^2 = \frac{|\pi p - \tilde{p}|^2}{|\tilde{p}|^2} \tag{6}$$

where $\tilde{p}$ are the measured pressures $|.|^2$ denotes an energy norm. In Eq. (5), $\frac{r}{1-r}$ is a weighting factor translating the confidence on the measurements. If the measurements are assumed to be accurate, this factor will tend to the infinity ($r \to 1$).

2.3 Local indicators

Error (5) can be rewritten as follows

$$e_\omega^2 = \sum_{i=1}^{N_{\text{Bound}}} \xi_{\omega,i}^2 + \sum_{j=1}^{N_{\text{Sens}}} \frac{r}{1-r} \eta_{\omega,j}^2 \tag{7}$$

where N_{Bound} and N_{Sens} are respectively the number of boundaries and of sensors and $\xi_{\omega,i}^2$ and $\eta_{\omega,j}^2$ are respectively the local estimators of the CRE and of the error in measurements.

2.4 Updating on a frequency range

In order to further regularize the problem, the updating process is generally performed on a frequency range $[\omega_{\min}, \omega_{\max}]$. The expression of the modified CRE (7) becomes

$$e_T^2 = \frac{1}{N_{\text{Freq}}} \sum_{f=1}^{N_{\text{Freq}}} e_{\omega_f}^2 = \frac{1}{N_{\text{Freq}}} \sum_{f=1}^{N_{\text{Freq}}} \left(\sum_i^{N_{\text{bound}}} \xi_{\omega_f,i}^2 + \sum_j^{N_{\text{Sens}}} \eta_{\omega_f,j}^2 \right) = \xi_T^2 + \frac{r}{1-r} \eta_T^2 \tag{8}$$

where N_{Freq} is the number of frequencies in the frequency range.

3 Implementation of the two-stages updating technique

The first step consists in the localization of the defective sensor, by looking at the distribution of η_{iT}^2 of each sensor i on the global error in measurements η_T^2. If the sources of error are identified, it is possible to correct the measurement. Otherwise, measurements are removed from the set of measurements. The second step is the two-stages updating process, consisting in

- the localization of the most erroneous parameters, by looking at the distribution of ξ_{ET}^2 of each boundaries E on the global CRE ξ_T^2. All the boundaries such as

$$\xi_{ET}^2 \geq \delta \max_E \xi_{ET}^2 \tag{9}$$

with $\delta = 0.8$, for example, are considered as the worst modeled and the parameters used to describe the admittance coefficients and/or the normal component of the prescribed velocity are considered as the most erroneous.

- the correction of the parameters identified as the most erroneous.

At each iteration of the two-stages process, the global modified CRE e_T^2 is calculated and compared to the required quality level e_{T0}^2. If this level is reached, the process ends. Otherwise, a new iteration is performed.

4 APPLICATION

4.1 Reference problem

Let us consider a 1D acoustical domain of 1m length. The domain is meshed with 40 elements, and the frequency range is from 100Hz to 1000Hz. The domain is excited by a loudspeaker covered by felt at $x = 0m$ and that at $x = L = 1m$, the border is covered by foam. The boundary conditions are therefore defined by

- at $x = 0$: $\lambda_0 = 0.5$ - $\bar{v}_{n,0}(\omega) = f_{\mathrm{HP}}(F/m, \zeta, \phi, \omega_0, \omega)$ - $A_{n,0}(\omega) = f_{\mathrm{DBM}}(\sigma, d, \omega)$

- at $x = L$: $\lambda_L = 1$ - $A_{n,L}(\omega) = f_{\mathrm{DB}}(\sigma, d, \omega)$

where f_{HP} represents the velocity of a membrane of a loudspeaker (F is applied force to the loudspeaker, m is the mass of the membrane of the loudspeaker, ζ is the damping ratio, ϕ is a phase and ω_0 is the eigen-frequency of the membrane), f_{DB} and f_{DBM} respectively represent the Delany-Bazley model (for the felt) and the Delany-Bazley-Miki model (for the foam) used to describe the admittance coefficient of the absorbing materials (σ is the resistivity and d is the stiffness of the material). The exact values of the coefficients used in these models are given in Table 1. The second column of this table gives the initial value of the parameters.

Let us consider that all the nodes, excepted those at the boundaries, are considered as a sensor location. The measured pressures are given by the numerical solution of the 1D acoustical problem, using the exact value of the parameters. In order to verify the step of localization of defective sensors, 3 sensors (at $x = 0.25m$, $x = 0.5m$ and $x = 0.75m$) are considered as defective with an error of 50%.

4.2 Localization of defective sensors

The first step of the process is to ensure that the pressures are correctly measured. Figure 1 gives the distribution of the local estimators η_{iT}^2. It is clearly shown that the 3 defective sensors have a bigger contribution to the global error in measurement η_T^2 than the other ones. It is therefore possible to localize defective sensors.

4.3 Updating process

Table 1 gives the results of the updating process, considering with (third column) and without (fourth column) defective sensors, allowing to conclude that the correction of the erroneous measurements allows to improve the results of the updating process.

5 Conclusions

A method to identify acoustical properties of absorbing material, from in-situ measurements, based on the CRE updating technique, is developed. In addition, local estimators are used to localize and correct the erroneous measurements. The technique is applied on
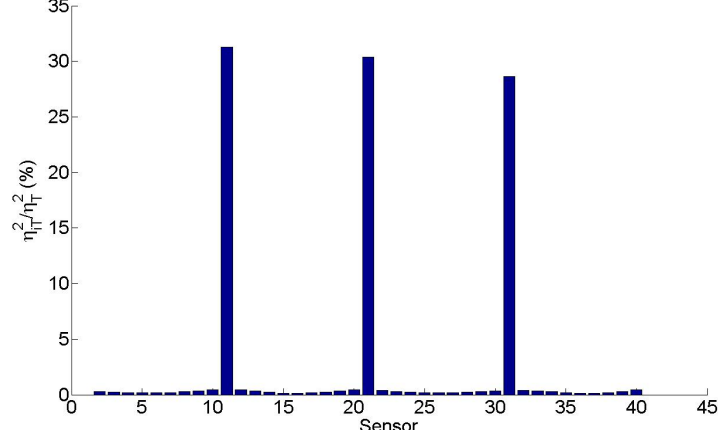
Figure 1: η_{iT}^2/η_T^2 (in percent) for $r \to 1$

Parameters		Exact value	Initial value	Final Value	
				without	with
$\bar{v}_{n,1}$	F/m (m/s^2)	0.20	0.15	0.1922	0.20
	ϕ (rad)	$20\pi/180$	$30\pi/180$	$31.1535\pi/180$	$20.0006\pi/180$
	ζ	0.45	0.60	0.4480	0.45
	ω_0 (Hz)	$2\pi200$	$2\pi150$	$2\pi199.5813$	$2\pi200.0005$
$A_{n,1}$	σ (Ns/m^4)	$2\ 10^4$	$1.5\ 10^4$	$1.5003\ 10^4$	$2.0200\ 10^4$
	d (m)	0.02	0.015	0.0203	0.0200
$A_{n,2}$	σ (Ns/m^4)	$1.5\ 10^3$	$2\ 10^3$	$1.6003\ 10^3$	$1.4963\ 10^3$
	d (m)	0.015	0.02	0.0150	0.0150

Table 1: Values of the parameters of the problem

a 1D simple test case, in order to illustrate the localization of defective sensors and the influence of the correction of the erroneous measurements on the results of the process.

REFERENCES

[1] P. Ladevèze. A modeling error estimator for dynamic model updating, in *New Advances in Adaptive Computational Methods in Mechanics*, pp. 135-151, (1998).

[2] A. Deraemaeker, P. Ladevèze, T. Romeuf. Model validation in the presence of uncertain experimental data. *Engng. Comput* (2004) **21**:808-833.

VI International Conference on Adaptive Modeling and Simulation
ADMOS 2013
J. P. Moitinho de Almeida, P. Díez, C. Tiago and N. Parés (Eds)

ANALYSIS OF LINEARIZATION ERROR FOR GOAL-ORIENTED ADAPTIVITY OF NONLINEAR PROBLEMS

Serge Prudhomme[∗]**, Kris van der Zee**[†]**, and Luca Dedè**[‡]

[∗]École Polytechnique de Montréal
C.P. 6079, Succ. Centre-ville, Montréal, Québec, Canada H3C 3A7
e-mail: serge.prudhomme@polymtl.ca

[†]Eindhoven University of Technology
P.O. Box 513, 5600 MB Eindhoven, The Netherlands
e-mail: k.g.v.d.zee@tue.nl

[‡]École Polytechnique Fédérale de Lausanne
Avenue Picard, Station 8, CH-1015 Lausanne, Switzerland
e-mail: luca.dede@epfl.ch

Key words: Nonlinear primal problem, Linearization error, Refinement indicators

Abstract. We propose in this talk to address the issue and effect of linearization in the quality of the error estimates in quantities of interest for strongly nonlinear problems (see e.g. [1, 2]). It is well known that the error representation in this case can be decomposed into two contributions: 1) one contribution in the form of the product of the residual by the solution of a linearized adjoint problem that describes the discretization error and 2) the other contribution that combines all higher-order terms with respect to the error in the primal solution that describes the linearization error. In most works on goal-oriented error estimation, the linearization error contribution is usually neglected with respect to the discretization error. However, when the nonlinear effects are significant, one cannot assume any longer that the latter is dominant over the former. In fact, it becomes obvious in those cases that linearization errors should be also controlled. We will present here the construction of refinement indicators that combine both sources of errors in order to simultaneously control those in a balanced manner.

REFERENCES

[1] V. Dolejší, A. Ern, and M. Vohralík. A framework for robust a posteriori error control in unsteady nonlinear advection-diffusion problems. Technical report, Sept. 2012. HAL: hal-00652979.

[2] K.G. van der Zee, J.T. Oden, S. Prudhomme, and A.J. Hawkins. Goal-oriented error estimation for Cahn-Hilliard models of binary phase transition. *Numerical Methods for Partial Differential Equations*, 27(1):160–196, 2011.

ACCURATE MODELLING OF STRAIN DISCONTINUITIES IN BEAMS USING AN XFEM APPROACH

S. RAMAN[*], B.C.N.MERCATORIS[†] AND A.DERAEMAEKER[*]

[*]Department of Building, Architecture and Town Planning (BATir)
Université Libre de Bruxelles
Avenue FD Roosevelt 50, CP 194/2, 1050 Brussels, Belgium
e-mail: sraman@ulb.ac.be,aderaema@ulb.ac.be, Website:batir.ulb.ac.be

[†]Department of Environmental Sciences and Technologies
Faculty of Gembloux Agro-Bio Tech
University of Liège
Passage des Dèportès 2, 5030 Gembloux, Belgium
e-mail: benoit.mercatoris@ulg.ac.be

Key words: XFEM,Timoshenko beams, Euler-Bernoulli beams, Assumed natural strain method, Shear locking

Abstract. The aim of this paper is to study the possibility of using extended finite element methods to model piezoelectric transducers attached to beam structures without the need for a conforming mesh. The main focus of this study is to propose enrichment functions to represent accurately the strain discontinuities in Euler-Bernoulli and Timoshenko beams. Further, we evaluate the performance of the enrichment functions on simple static cases with a special emphasis on the shear locking in the Timoshenko beam.

1 INTRODUCTION

Thin piezoelectric transducers are widely used in applications such as active vibration control, wave generation in materials and structural health monitoring. The finite element modelling of piezoelectric transducers is well established; an overview of the existing models can be found in [1]. Current practice for the modelling of structures equipped with flat piezoelectric transducers requires the development of specific beam or plate elements which are usually not available in commercial codes. The most important criteria when using the finite element method to model piezoelectric transducers attached to host structures is that the mesh must exactly match the boundary between the piezoelectric transducers and the host structure. This requirement of conforming meshes leads to extensive remeshing of the structure when optimal transducers configurations are investigated.

The need for conforming meshes arises due to the following reasons: the occurrence of

a strain jump across the interface between the piezoelectric transducer and the host structure due to the additionnal stiffness of the piezoelectric transducer and the distributed efforts acting on the edges of the patch when used as an actuator, the continuity of the displacement field across the interface, and the presence of an electric field only in the piezoelectric material. To overcome meshing difficulties and capture local phenomenon, the extended finite element method (XFEM) for weak discontinuities was proposed for two-dimensional problems [2]. In this paper, we will make use of XFEM to develop enriched Euler-Bernoulli and Timoshenko beam elements that can capture jumps in strains across the interface between two materials using a non-conforming mesh. We identify the location of the interface using an implicit level-set method. This paper is organized as follows: Section 2 gives a brief overview about the behaviour of piezoelectric transducers under actuation and their impact on the host structures. In Section 3, we develop the enriched Euler-Bernoulli beam element with special emphasis on finding the right enrichment function. In section 4, we develop the enriched Timoshenko beam finite elements. In section 5, the shear locking problem of the enriched Timoshenko beams are explained and an assumed natural strain method to avoid shear locking are proposed and tested for an enriched Timoshenko beam element.

2 PIEZOELECTRIC ACTUATORS ATTACHED TO HOST STRUCTURES

Piezoelectric transducers operate in two modes: sensors and actuators. They are used as sensors when generation of a surface charge happens as a result of mechanically straining the piezoelectric material. For instance, this effect is usually used in force and acceleration sensors. They also function as actuators when the geometry of the piezoelectric material changes due to an applied electric field. Actuating the piezoelectric transducer produces equivalent forces on the host structures as described in [3]. A cantilever beam with attached piezoelectric transducers is considered in this paper as shown in Figure 1.

Actuating the piezoelectric transducers produces bending moments and point-forces in the host structure at the boundaries of the piezoelectric actuator as shown in Figure 2. These stresses result in jumps in the membrane deformation and curvature of the beam. In this study, we are interested in capturing these jumps using a non-conforming mesh. Specifically we will be tackling the jump in the curvature of beams. This case is similar to two-material beams where deformation jumps occur at the material interfaces. For the sake of simplicity, a two-material beam considered here in order to develop the enriched beam finite elements. These enriched finite elements could then be adapted to a multi-layer coupled electro-mechanical beam model.

3 ENRICHED EULER-BERNOULLI BEAM FINITE ELEMENT

Assuming Euler-Bernoulli theory, the transverse deflection w of the beam is governed by the fourth order differential equation given by

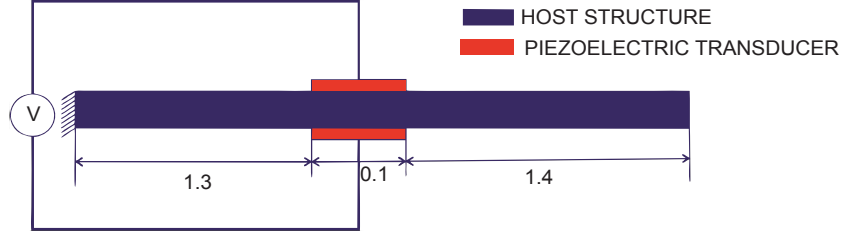$$\frac{d^2}{dx^2}(EI\frac{d^2w}{dx^2}) = q(x) \tag{1}$$

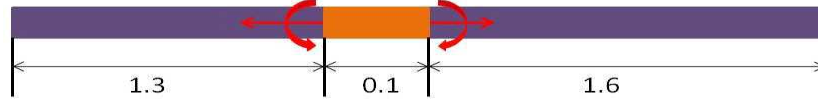**Figure 1**: Cantilever beam with piezoelectric transducers

Figure 2: Equivalent loads on a cantilever beam with piezoelectric transducers

where E is the Young's Modulus of the beam, I is the area moment of inertia about the transverse axis of the beam and q is the distributed transverse load. The weak form of this equation is given for an element by

$$\int_{x_e}^{x_{e+1}} \left(EI\frac{d^2v}{dx^2}\frac{d^2w}{dx^2} - vq \right)dx + \left[v\frac{d}{dx}\left(EI\frac{d^2w}{dx^2} \right) - \frac{dv}{dx}EI\frac{d^2w}{dx^2} \right]_{x_e}^{x_{e+1}} = 0 \qquad (2)$$

The essential boundary conditions involve the specification of the deflection w and the slope $\frac{dw}{dx}$. The natural boundary conditions involve the specification of the bending moment $EI\frac{d^2w}{dx^2}$ and the shear force $\frac{d}{dx}(EI\frac{d^2w}{dx^2})$ at the boundaries. The curvature of the beam is given by $\frac{d^2w}{dx^2}$ and is considered as the generalized strain measure of the beam. Hermite shape functions are used to approximate the deflection at any point of the beam using finite elements, which reads

$$u^{FEM} = \Sigma_{i=1}^2(H_iw_i + R_i\theta_i) \qquad (3)$$

where u^{FEM} represents the deflection of the beam, H_i and R_i are the Hermite cubic shape functions and w_i and θ_i are the nodal deflections and nodal rotations. Considering the problem defined in Figure 2, classical finite element modelling requires conforming meshes in order to capture properly the material interfaces. An extended finite element method (XFEM) allows the use of non-conforming meshes. In a XFEM approach, the displacement field is enriched using the partition of unity technique [4] and is given by

$$u^{XFEM} = \Sigma_{i=1}^2(H_iw_i + R_i\theta_i) + \Sigma_{j=1}^n N_j\psi_ja_j \qquad (4)$$

where N_j are the partition of unity shape functions, n is the total number of functions forming the partition of unity, ψ_j are the enrichment functions and a_j are the additional degrees of freedom related to the enrichment.

3.1 Enriched Nodes

The location of an interface is found using the implicit level-set method. The level-set is a measure of the signed distance between a node and the considered discontinuity. The elements where the discontinuity is present are found using level-sets as described in [5]. The kinematics of all the nodes belonging to this element needs to be enriched.
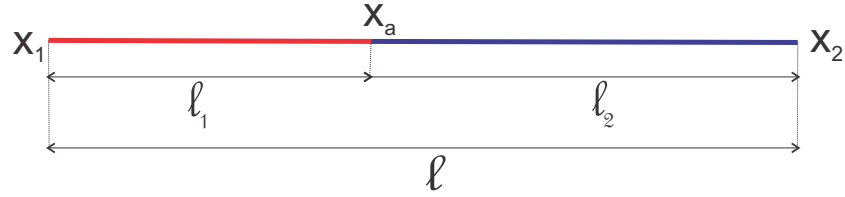

Figure 3: Enriched beam element
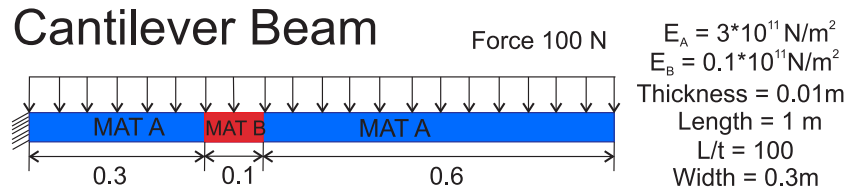
Figure 4: Two-material Cantilever beam with uniform load
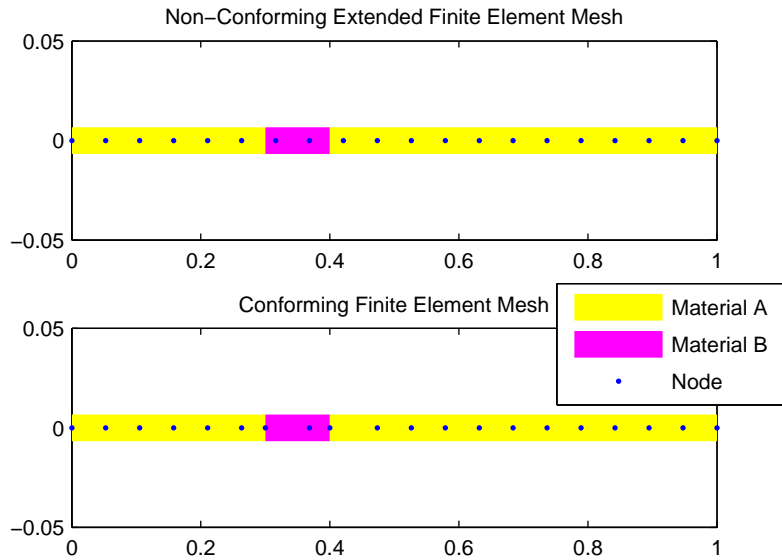
Figure 5: Non-conforming and conforming meshes for the considered problem
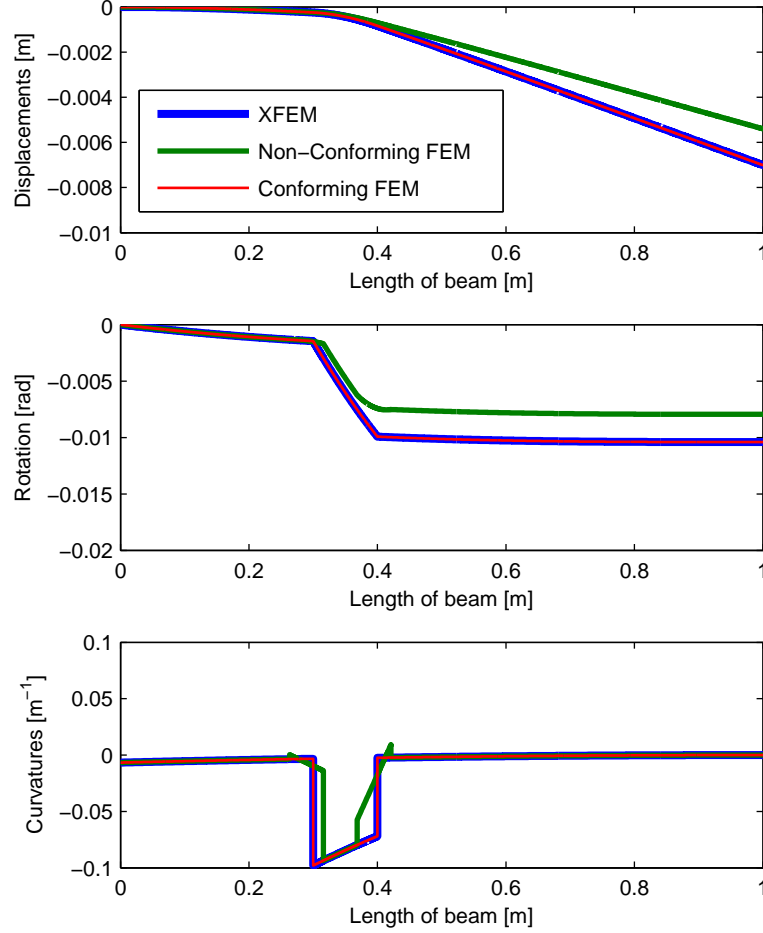
Figure 6: Comparison of Displacements, Rotations and Curvature of two-material Euler-Bernoulli beam

3.2   Enrichment functions for Euler-Bernoulli beams

In the field of XFEM, two-material problems are classified as weak discontinuity problems because the discontinuity occurs in the derivatives of the primary variables which remain continuous. The modelling of weak discontinuities using XFEM is explained in [2] and [5] for two-dimensional problems. In this case, the primary variable is discretized using linear shape functions and the enrichment functions are therefore not suitable for the case of Euler-Bernoulli beams which require cubic shape functions. Since two independent jumps, a curvature jump and a transverse shear jump, are needed to incorporate properly a discontinuity in an Euler-Bernoulli beam, two enrichment functions have to be defined. Their properties are listed as follows

- The enrichment functions have to be cubic and piece-wise continuous

- They have to vanish at the boundary of the element to avoid problems with blending elements as documented in [6]

- The first derivative of enrichment functions have to be continuous at the interface

- The first derivative of the enrichment functions should also get to zero at the boundary of the element

- The second and third derivatives of the enrichment functions should be discontinuous at the location of the interface

Considering an element depicted in Figure 3 whose extremities are located at X_1 and X_2 with a material discontinuity located at X_a, the enrichment functions are derived by means of the conditions set forth above. From the admissible space, the following set of enrichment functions are found

$$\psi_1 = \begin{cases} 3s_1^2 - 2s_1^3 & \text{if} \quad x < X_a \\ 1 - 3s_2^2 + 2s_2^3 & \text{if} \quad x > X_a \end{cases} \tag{5}$$

$$\psi_2 = \begin{cases} \ell_1 s_1^2 (s_1 - 1) & \text{if} \quad x < X_a \\ \ell_2 s_2 (s_2 - 1)^2 & \text{if} \quad x > X_a \end{cases}$$

where

$$s_1 = \frac{x - X_1}{\ell_1} \quad \text{and} \quad s_2 = \frac{x - X_a}{\ell_2} \tag{6}$$

3.3 Partition of unity for Euler-Bernoulli beams

Partition of unity is formed by a set of shape functions which add up to one. In case of Euler-Bernoulli beams, the H_i functions sum up to one and form a partition of unity. With the partition of unity established, we can now write the discretized XFEM expression for the beam element shown in Figure 3 as

$$u^{XFEM} = H_1 w_1 + + R_1 \theta_1 + H_2 w_2 + R_2 \theta_2 + H_1 \psi_1 a_1 + H_2 \psi_1 a_2 + H_1 \psi_2 a_3 + H_2 \psi_2 a_4 \tag{7}$$

with degrees of freedom w_1, θ_1, a_1 and a_3 at node 1 and the remaining at node 2.

3.4 Implementation

Considering the beam problem defined in Figure 4 and the conforming and non-conforming meshes depicted in Figure 5, the XFEM solution and the FEM solution on the same non-conforming mesh are plotted in Figure 6. These solutions are compared with the conforming mesh solution also given in Figure 6. It can be observed that the jump in curvature is properly captured by the XFEM using a non-conforming mesh. For the FEM solution using a non-conforming mesh,the change in material properties is carefully accounted for during numerical integration by applying the material property of the material where the integration point is located.

4   ENRICHED TIMOSHENKO BEAM FINITE ELEMENT

In this second part, the proposed approach is extended in order to incorporate a weak discontinuity within a Timoshenko beam theory. The strain energy considering both bending and shear contributions is given as

$$U = \frac{1}{2} \int_V \sigma_x \epsilon_x dV + \frac{1}{2} \int_V \tau_{xy} \gamma_{xy} dV \tag{8}$$

where the normal stresses are given by the Hooke's law

$$\sigma_x = E\epsilon_x \tag{9}$$

and the transverse shear stress is given by

$$\tau_{xy} = kG\gamma_{xy} \tag{10}$$

where k is the shear correction factor that varies according to the cross-section of the beam and G is the shear modulus given by

$$G = \frac{E}{2(1+\upsilon)} \tag{11}$$

where υ is the Poisson's ratio for the beam material. Considering a Timoshenko beam theory, the strain energy becomes

$$U = \frac{1}{2} \int_0^l EI(\frac{\partial \theta}{\partial x})^2 dx + \frac{1}{2} \int_0^l kAG(-\frac{\partial w}{\partial x} + \theta)^2 dx \tag{12}$$

In contrast with the Euler-Bernoulli beam element, independent linear interpolations are used here for the rotation and the deflection. For the case of a classical finite element, this reads

$$w^{FEM} = N_1 w_1 + N_2 w_2 \tag{13}$$
$$\theta^{FEM} = N_1 \theta_1 + N_2 \theta_2$$

where $N_1 = 1 - x/l$ and $N_2 = x/l$ where l is the length of the element. As with an Euler-Bernoulli beam, the Timoshenko beams can be enriched by introducing the partition of unity based enrichment functions as follows

$$w^{XFEM} = N_1 w_1 + N_2 w_2 + N_1 \psi_1 a_1 + N_2 \psi_1 a_2 \tag{14}$$
$$\theta^{XFEM} = N_1 \theta_1 + N_2 \theta_2 + N_1 \psi_2 a_3 + N_2 \psi_2 a_4$$

Since linear interpolations are used, the enrichment functions ψ_1 and ψ_2 can be ramp functions as described in [5]. The partition of unity is formed by the shape functions N_1 and N_2. As described in the previous section, the location of the discontinuity is found using the level-set and the elements containing a discontinuity are enriched.
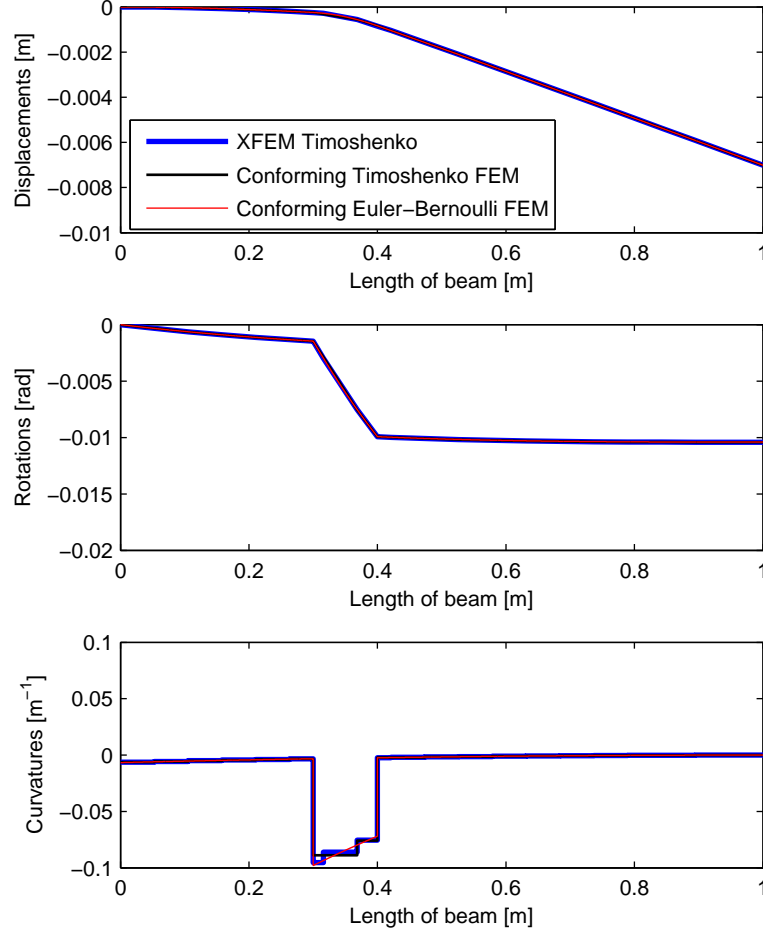
Figure 7: Comparison of the novel XFEM Timoshenko beam theory with respect of the clasical conforming beam theories for the two-material beam problem. The deflection, the rotation and the curvature are compared.

4.1   Locking in Timoshenko beams

Without using any specific treatment, the element beam described in the previous section suffers from shear locking. This occurs due to inconsistent interpolation for w and θ. In order to avoid the occurence of shear locking, many techniques were proposed such as the assumed natural strain method, the reduced integration method and the consistent interpolated element method. A detailed description of these methods can be found in [7]. In this paper, an assumed natural strain method is used to avoid any shear locking in the enriched Timoshenko beam element.
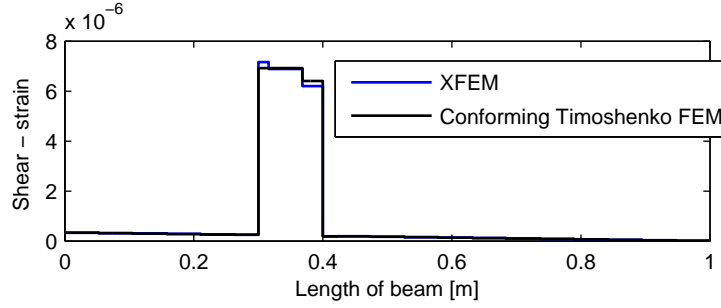
Figure 8: Comparison of shear-strains of two-material Timoshenko beam

5   SHEAR LOCKING TREATMENT IN AN ENRICHED TIMOSHENKO BEAM ELEMENT

The strain energy in Equation (12) is made of two parts namely the bending energy and the shear energy. The stiffness matrix can therefore be split into two parts: the bending part and the shear part which reads

$$K = \int_{x_1}^{x_2} [B^b]^T EI[B^b]dx + \int_{x_1}^{x_2} [B^s]^T kGA[B^s]dx \tag{15}$$

where B^b and B^s are the operators linking respectively the curvature and the transverse shear to the degrees of freedom of the beam. Using Equation (14), these operators are given for an XFEM beam element as

$$B^b = \begin{bmatrix} 0 & \dfrac{dN_1}{dx} & 0 & \dfrac{dN_2}{dx} & 0 & 0 & \dfrac{d(N_1\psi)}{dx} & \dfrac{d(N_2\psi)}{dx} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{16}$$

$$B^s = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\dfrac{dN_1}{dx} & N_1 & -\dfrac{dN_2}{dx} & N_2 & -\dfrac{d(N_1\psi)}{dx} & -\dfrac{d(N_2\psi)}{dx} & N_1\psi & N_2\psi \end{bmatrix}$$

The B^s matrix in the above equation will lead to a shear locking problem due to the presence of shape functions and their derivatives together. Based on the definition of the transverse shear strain $\chi = \theta - \frac{dw}{dx}$, an assumed natural strain method is used to treat properly the shear locking. The transverse shear strain is assumed piece-wise constant on each side of the interface. This is motivated by the fact that a conforming mesh with regular beam element would lead to a piece-wise constant transverse shear field with a discontinuity at the material interface. A classical collocation method is used to determine the assumed strain as follows

$$\int_{x_1}^{x_a} \overleftrightarrow{\chi_1} - \chi = 0$$
$$\int_{x_a}^{x_2} \overleftrightarrow{\chi_2} - \chi = 0 \tag{17}$$
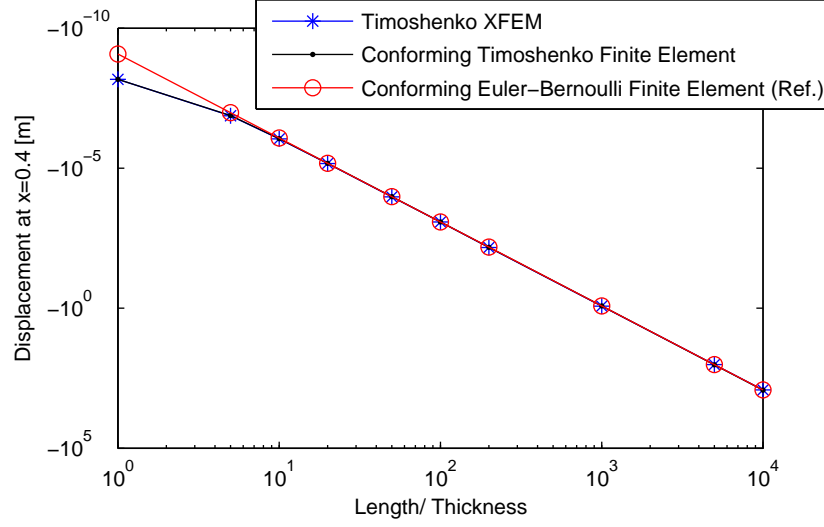
Figure 9: Behaviour of Elements as t→0

Expanding the above equations with the assumption that χ_1 and χ_2 are constant leads to

$$\overleftrightarrow{\chi_1} = \frac{1}{\ell_1} \int_{x_1}^{x_a} N_1\theta_1 + N_2\theta_2 + N_1\psi b_1 + N_2\psi b_2 \tag{18}$$

$$-(\frac{dN_1}{dx}w_1 + \frac{dN_2}{dx}w_2 + \frac{d(N_1\psi)}{dx}a_1 + \frac{d(N_2\psi)}{dx}a_2)$$

$$\overleftrightarrow{\chi_2} = \frac{1}{\ell_2} \int_{x_a}^{x_2} N_1\theta_1 + N_2\theta_2 + N_1\psi b_1 + N_2\psi b_2 \tag{19}$$

$$-(\frac{dN_1}{dx}w_1 + \frac{dN_2}{dx}w_2 + \frac{d(N_1\psi)}{dx}a_1 + \frac{d(N_2\psi)}{dx}a_2)$$

Using Equations (19) and (2), the B^s operator can be split into two contributions related to each part of enriched beam element as follows

$$B_1^s = \frac{1}{\ell_1} \int_{x_1}^{x_a} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\dfrac{dN_1}{dx} & N_1 & -\dfrac{dN_2}{dx} & N_2 & -\dfrac{d(N_1\psi)}{dx} & -\dfrac{d(N_2\psi)}{dx} & N_1\psi & N_2\psi \end{bmatrix} \tag{20}$$

and

$$B_2^s = \frac{1}{\ell_2} \int_{x_a}^{x_2} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\dfrac{dN_1}{dx} & N_1 & \dfrac{dN_2}{dx} & N_2 & -\dfrac{d(N_1\psi)}{dx} & -\dfrac{d(N_2\psi)}{dx} & N_1\psi & N_2\psi \end{bmatrix} \tag{21}$$

5.1 Implementation

The considered problem is depicted in Figure 4. The Poisson's ratio is assumed to be 0.3 for both the materials. The deflection, the rotation and the curvature are compared in Figure 7 for the proposed XFEM Timoshenko formulation using the non-conforming mesh shown in Figure 5, with respect to a Timoshenko formulation and an Euler-Bernoulli formulation both using a refined conforming mesh. Since the beam is very thin, it is shown that the results of both Euler-Bernoulli and Timoshenko formulations are coherent. Also from Figure 8, it can be observed that the jumps in the transverse shear strain field is also captured accurately using XFEM. As shown in Figure 9, the solution of the enriched Timoshenko formulation and the solutions of conforming Timoshenko and Euler-Bernoulli approches are in good agreement when the beam thickness tends to zero. This effectively proves that any shear locking does not occur when using the extended Timoshenko beam finite element. The difference between the Euler-Bernoulli approach and the Timoshenko approach can also be observed in Figure 9, where for smaller values of length over thickness ratio, the solutions of the two approaches are different because the shear effects which are more prevalent at these ratios are only considered when modelling using the Timoshenko theory. Also the contribution from strain-energy to the total energy of the beam reduces as the length to thickness ratio increases. The enriched element behaves no different in this regard.

6 CONCLUSION AND PERSPECTIVES

In this study, we have detailed the enrichment functions for Euler-Bernoulli beam. Simple static case was tested with the newly proposed enrichment function and found to be satisfactory. A shear-locking free enriched Timoshenko beam finite element was also developed and tested for the same static case. An assumed natural strain method was used in order to avoid shear locking in Timoshenko beams. The current work will form the basis for the development of plate elements using an enriched finite element approach.

Acknowledgments

REFERENCES

[1] A.Benjeddou. Advances in piezoelectric finite element modelling of adaptive structural elements: A survey. *Computers & Structures.* (2000) **76(1-3)**:347-363.

[2] N.Sukumar, D.Chopp, N.Mes, and T.Belytschko. Modelling holes and inclusions by level sets in the extended finite-element method. *Computer Methods in Applied Mechanics and Engineering.* (2001) **190(46-47)**:6183-6200.

[3] A.Deraemaeker, G.Tondreau, F.Bourgeois.Equivalent loads for two-dimensional distributed anisotropic piezoelectric transducers with arbitrary shapes attached to thin plate structures. *The Journal of the Acoustical Society of America.* (2011)**129(2)**:681.

[4] J.Melenk and I.Babuka. The partition of unity finite element method: basic theory and applications. *Computer Methods in Applied Mechanics and Engineering*(1996) **139.1** : 289-314.

[5] N.Moës ,M.Cloirec ,P.Cartraud and J-F.Remacle.A computational approach to handle complex microstructure geometries. *Computer Methods in Applied Mechanics and Engineering.* (2003)**192(28)**:3163-3177.

[6] T-P.Fries. A corrected XFEM approximation without problems in blending elements. *International Journal for Numerical Methods in Engineering* (2007) **75(5)**:503-532

[7] JN.Reddy. On locking-free shear deformable beam finite elements. *Computer Methods in Applied Mechanics and Engineering* (2007) **149(1)**:113-132.

ADAPTIVE TIMESTEP CONTROL FOR THE GENERALISED-α METHOD

JOACHIM RANG[*]

[*]Institute of Scientific Computing
TU Braunschweig
Hans-Sommer Str. 65, 38106 Braunschweig, Germany
e-mail: j.rang@tu-bs, web page: http://www.wire.tu-bs.de

Key words: Adaptivity, generalised-α methods

Abstract. In this article an adaptive timestep control for the generalised-α methods is introduced. If the methods for first and second order ODEs are formulated as onestep schemes a second solution can be computed with the backward Euler method, which costs no additional computing time. If the generalised-α methods are formulated as multistep methods an adaptive timestep control is only introduced for first order ODEs. In this case the method is formulated with variable coefficients and a second solution is computed with the Leapfrog method. Numerical examples show in the case of the onestep versions the advantages of the adaptive algorithms.

1 INTRODUCTION

In this article we consider the generalised-α methods, which are introduced for first order ODEs in [JWH00] and for second order ODEs in [CH93]. The generalised-α methods are usually of second order and allow the damping of high frequencies, which can be controlled by certain parameters. An analysis for first order problems can be found in [DP03]. In the case of second order ODEs many papers can be found, which analyse the generalised-α method, for example [EBB02]. It is well known that the generalised-α method for first order problems can be formulated as onestep and multistep methods. In the case of second order methods this statement is only true if the ODE is linear in the first derivative (see [EBB02]). For both classes of multistep methods second order can be achieved if a further order condition is satisfied. Together with stability conditions (see [EBB02]) a robust and effective class of methods is obtained. If these parameter sets are used for onestep methods theoretically only first order can be reached. But the error constant is very small so that the observed numerical order of convergence is two. Moreover in our experience the onestep versions achieve better results than the multistep versions.

For solving ODEs or DAEs a good time integration method needs an error estimator to increase efficiency. This error estimator suggests a new timestep size to reach a given accuracy. If the timestep size is too small a lot of unnecessary computational work has to be done. Otherwise, if the timestep size is too large, the results become less accurate. In [HNW93] two approaches for time-adaptive one-step methods are presented. The first one is called *Richardson extrapolation* and can be applied to every one-step method. In this case the calculations for computing an approximation of the solution at the next timestep are repeated with the timestep size $\tau/2$ and compared with the first result, i.e. the computational work increases by a factor of 3. Thus, the question of efficiency arises.

A more effective control of timesteps can be achieved with the so-called embedding technique, which can be used for many Runge–Kutta and Rosenbrock–Wanner methods [HW96, SW92]. In this case a second solution can be computed with almost the same coefficients and without solving a further linear or non-linear solution, i.e. there are almost no further computational costs. Applications can be found in [HHR12, JR10, Lan01, Ran04].

In this article different approaches are considered. In the case of the onestep formulation of the generalised-α method the backward Euler method can be used to compute a second solution without any further computation. As the numerical examples will show this is an effective way of computing adaptive timestep sizes.

In the case of the multistep formulation we need methods, which have variable coefficients. In the case of second order ODEs we get a formulation, which involves potentials of the mass and the damping matrix. Therefore we only develop for the onestep version a formulation with variable coefficients. In this case a second solution can be computed with the Leapfrog method. This approach can be found in [GS00] for a backward difference formular.

This paper is structured as follows. First we introduce the generalised-α methods for first and second order ODEs. A short analysis about convergency and stability is given. Then adaptive algorithms are explained and numerical examples illustrate the adavantages of the new adaptive methods.

2   THE GENERALISED-α METHOD FOR 1ST ORDER ODES

In the following we consider the ODE

$$\dot{\mathbf{u}} = \mathbf{f}(t, \mathbf{u}), \quad \mathbf{u}(0) = \mathbf{u}_0. \tag{1}$$

To determine the numerical solution of (1) we use the generalised-α method, which is given by the formulas (see [JWH00, DP03])

$$\dot{\mathbf{u}}_{n+\alpha_m} = \mathbf{f}(t_{n+\alpha_f}, \mathbf{u}_{n+\alpha_f}), \tag{2}$$

$$\mathbf{u}_{n+1} = \mathbf{u}_n + \tau\dot{\mathbf{u}}_n + \tau\gamma(\dot{\mathbf{u}}_{n+1} - \dot{\mathbf{u}}_n), \tag{3}$$

$$\dot{\mathbf{u}}_{n+\alpha_m} = \dot{\mathbf{u}}_n + \alpha_m(\dot{\mathbf{u}}_{n+1} - \dot{\mathbf{u}}_n), \tag{4}$$

$$\mathbf{u}_{n+\alpha_f} = \mathbf{u}_n + \alpha_f(\mathbf{u}_{n+1} - \mathbf{u}_n). \tag{5}$$

It is well known that the generalised-α method can be formulated as a onestep and a twostep method.

2.1 The formulation as onestep method and its analysis

First we manipulate the formulas (2)–(5) to obtain a non-linear system consisting of two decoupled equations. For simplification we define $\mathbf{f}_{n+\alpha_f} := \mathbf{f}(t_{n+\alpha_f}, \mathbf{u}_{n+\alpha_f})$. A simple calculation gives us

$$\mathbf{u}_{n+1} = \mathbf{u}_n + \tau \left(1 - \frac{\gamma}{\alpha_m} \right) \dot{\mathbf{u}}_n + \frac{\tau\gamma}{\alpha_m} \mathbf{f}_{n+\alpha_f}, \tag{6}$$

$$\dot{\mathbf{u}}_{n+1} = \frac{1}{\tau\gamma} \left(\mathbf{u}_{n+1} - \mathbf{u}_n - \tau(1-\gamma)\dot{\mathbf{u}}_n \right), \tag{7}$$

if $\alpha_m \neq 0$. We call the scheme (6)–(7) the *onestep generalised-α method*. The starting value $\dot{\mathbf{u}}_0$ can be computed from the ODE (1). Next we want to determine the order of consistency. Therefore the numerical solution $\mathbf{u}_{n+1}$ can be expanded in a Taylor series as follows

$$\mathbf{u}_{n+1} = \mathbf{u}_n + \tau\dot{\mathbf{u}}_n + \frac{\tau^2\gamma\alpha_f}{\alpha_m} \ddot{\mathbf{u}}_n + \mathcal{O}(\tau^3).$$

For consistency of order 2 we get the condition $\dfrac{\gamma\alpha_f}{\alpha_m} = \dfrac{1}{2}$. Since $\mathbf{u}_{n+1}$ depends on $\dot{\mathbf{u}}_n$ we use equation (7) for expanding $\dot{\mathbf{u}}_{n+1}$ in a Taylor series and get

$$\dot{\mathbf{u}}_{n+1} = \dot{\mathbf{u}}_n + \frac{\tau\alpha_f}{\alpha_m} \ddot{\mathbf{u}}_n + \mathcal{O}(\tau^2),$$

i. e. $\dot{\mathbf{u}}_{n+1}$ is of order 1 if $\dfrac{\alpha_f}{\alpha_m} = 1$. Summarising our results we have consistency of order 2 if $\alpha_m = \alpha_f$ and $\gamma = 1/2$. It can be easily shown that the generalised-α method is zero-stable if $\alpha_m > 1/2$. In other words our method is convergent if $\alpha_m > 1/2$.

2.2 Formulation as multistep method and its analysis

The generalised-α method can be formulated as a twostep method as follows

$$\mathbf{u}_{n+1} = \frac{2\alpha_m - 1}{\alpha_m} \mathbf{u}_n - \frac{\alpha_m - 1}{\alpha_m} \mathbf{u}_{n-1} + \frac{\tau(1-\gamma)}{\alpha_m} \mathbf{f}_{n-1+\alpha_f} + \frac{\tau\gamma}{\alpha_m} \mathbf{f}_{n+\alpha_f}. \tag{8}$$

For $\alpha_m = 3/2$, $\alpha_f = 1$ and $\gamma = 1$ we obtain the backward difference formula (BDF) from Gear (see [HW96]). Next we expand $\mathbf{u}_{n+1}$ in a Taylor expansion and compare it with the exact solution. Then we have

$$\mathbf{u}_{n+1} = \mathbf{u}_n + \tau\dot{\mathbf{u}}_n + \frac{\tau^2}{2} \frac{2\alpha_f - \alpha_m + 2\gamma - 1}{\alpha_m} \ddot{\mathbf{u}}_n + \mathcal{O}(\tau^3).$$

Comparing the Taylor expansions for $\mathbf{u}(t_{n+1})$ and $\mathbf{u}_{n+1}$ leads to the condition for second order consistency

$$\gamma = \frac{1}{2} - \alpha_f + \alpha_m, \tag{9}$$

which is already known from [JWH00, DP03]). The generalised-α method in form (8) is convergent of order 2 if $\alpha_m > 1/2$ and condition (9) holds. For stability the setting

$$\alpha_f = \gamma = \frac{1}{1+\rho_\infty}, \qquad \alpha_m = \frac{3-\rho_\infty}{2(1+\rho_\infty)}. \tag{10}$$

is used (see [JWH00, DP03]). Note that condition (9) is automatically satisfied. For $\rho_\infty = 0$ we get the BDF-2 method.

3 THE GENERALISED-α METHOD FOR SECOND ORDER ODES

3.1 The formulation as onestep method

In the following we consider the second order ODE

$$\ddot{\mathbf{u}} = \mathbf{f}(t, \mathbf{u}, \dot{\mathbf{u}}), \quad \mathbf{u}(0) = \mathbf{u}_0, \dot{\mathbf{u}}(0) = \dot{\mathbf{u}}_0. \tag{11}$$

The generalised-α method can be written as

$$\mathbf{u}_{n+\alpha_f} = \alpha_f \mathbf{u}_{n+1} + (1-\alpha_f)\mathbf{u}_n, \tag{12}$$

$$\dot{\mathbf{u}}_{n+\alpha_f} = \alpha_f \dot{\mathbf{u}}_{n+1} + (1-\alpha_f)\dot{\mathbf{u}}_n, \tag{13}$$

$$\ddot{\mathbf{u}}_{n+\alpha_m} = \alpha_m \ddot{\mathbf{u}}_{n+1} + (1-\alpha_m)\ddot{\mathbf{u}}_n, \tag{14}$$

$$\mathbf{u}_{n+1} = \mathbf{u}_n + \tau\dot{\mathbf{u}}_n + \tau^2 \left[\left(\frac{1}{2} - \beta\right)\ddot{\mathbf{u}}_n + \beta\ddot{\mathbf{u}}_{n+1} \right], \tag{15}$$

$$\dot{\mathbf{u}}_{n+1} = \dot{\mathbf{u}}_n + \tau\left[(1-\gamma)\ddot{\mathbf{u}}_n + \gamma\ddot{\mathbf{u}}_{n+1}\right], \tag{16}$$

$$\ddot{\mathbf{u}}_{n+\alpha_m} = \mathbf{f}(t_{n+\alpha_f}, \alpha_f \mathbf{u}_{n+1} + (1-\alpha_f)\mathbf{u}_n, \alpha_f \dot{\mathbf{u}}_{n+1} + (1-\alpha_f)\dot{\mathbf{u}}_n), \tag{17}$$

where $t_{n+\alpha_f} = t_n + \tau\alpha_f$. To abbreviate we write

$$\mathbf{f}_{n+\alpha_f} := \mathbf{f}(t_{n+\alpha_f}, \alpha_f \mathbf{u}_{n+1} + (1-\alpha_f)\mathbf{u}_n, \alpha_f \dot{\mathbf{u}}_{n+1} + (1-\alpha_f)\dot{\mathbf{u}}_n).$$

These equations can be simplyfied to

$$\mathbf{u}_{n+1} = \mathbf{u}_n + \tau\dot{\mathbf{u}}_n + \tau^2 \left[\left(\frac{1}{2} - \frac{\beta}{\alpha_m}\right)\ddot{\mathbf{u}}_n + \frac{\beta}{\alpha_m}\mathbf{f}_{n+\alpha_f} \right], \tag{18}$$

$$\dot{\mathbf{u}}_{n+1} = \dot{\mathbf{u}}_n + \tau \left[\left(1 - \frac{\gamma}{\alpha_m}\right)\ddot{\mathbf{u}}_n + \frac{\gamma}{\alpha_m}\mathbf{f}_{n+\alpha_f} \right]. \tag{19}$$

$$\ddot{\mathbf{u}}_{n+1} = \frac{1}{\alpha_m}\left[\ddot{\mathbf{u}}_{n+\alpha_m} - (1-\alpha_m)\ddot{\mathbf{u}}_n\right] = \frac{1}{\alpha_m}\left[\mathbf{f}_{n+\alpha_f} - (1-\alpha_m)\ddot{\mathbf{u}}_n\right]. \tag{20}$$

Next we expand these three expression into Taylor expansions and get

$$\mathbf{u}_{n+1} = \mathbf{u}_n + \tau\dot{\mathbf{u}}_n + \frac{1}{2}\tau^2\ddot{\mathbf{u}}_n + \mathcal{O}(\tau^3),$$

$$\dot{\mathbf{u}}_{n+1} = \dot{\mathbf{u}}_n + \tau\ddot{\mathbf{u}}_n + \gamma\frac{\alpha_f}{\alpha_m}\tau^2\ddot{\mathbf{u}}_n + \mathcal{O}(\tau^3),$$

$$\ddot{\mathbf{u}}_{n+1} = \ddot{\mathbf{u}}_n + \tau\frac{\alpha_f}{\alpha_m}\dddot{\mathbf{u}}_n + \mathcal{O}(\tau^2).$$

It follows that the method is of order 2 if $\alpha_f/\alpha_m = 1$ and $\gamma\alpha_f/\alpha_m = 1/2$. This is the same result as in the previous section.

3.2 Formulation as multistep method

As in the previous section the generalised-α method can be written as a multistep method if the ODE (11) is linear in $\dot{\mathbf{u}}$. Therefore we consider the problem as in [EBB02]

$$M\ddot{\mathbf{u}} + C\dot{\mathbf{u}} + \mathbf{S}(\mathbf{u}) = \mathbf{F}(t), \mathbf{u}(0) = \mathbf{u}_0, \quad \dot{\mathbf{u}}_0 = \mathbf{v}_0. \tag{21}$$

Then equation (17) reads as

$$M\ddot{\mathbf{u}}_{n+\alpha_m} = \mathbf{F}(t_{n+\alpha_f}) - S(\alpha_f\mathbf{u}_{n+1} + (1 - \alpha_f)\mathbf{u}_n) - C(\alpha_f\dot{\mathbf{u}}_{n+1} + (1 - \alpha_f)\dot{\mathbf{u}}_n). \tag{22}$$

The generalised-α method can be formulated as a multistep method with the help of (15), (16), and (22). These formulas are evaluated at time t_n, t_{n+1}, and t_{n+2} (see for example [EBB02]). Then we get

$$\sum_{j=0}^{3}[M\alpha_j + \tau C\gamma_j]\mathbf{u}_{n+j} + \tau^2\sum_{j=0}^{2}\delta_j[\mathbf{S}_{n+j+\alpha_f} - \mathbf{F}(t_{n+j+\alpha_f})] = 0, \tag{23}$$

where

$$\alpha_0 = 1 - \alpha_m, \quad \alpha_1 = 3\alpha_m - 2, \quad \alpha_2 = 1 - 3\alpha_m, \quad \alpha_3 = \alpha_m,$$

$$\gamma_0 = (1 - \alpha_f)(\gamma - 1), \quad \gamma_1 = 1 - 2\alpha_f - 2\gamma + 3\gamma\alpha_f, \gamma_2 = \alpha_f + \gamma - 3\gamma\alpha_f, \quad \gamma_3 = \alpha_f\gamma,$$

$$\delta_0 = \frac{1}{2} + \beta - \gamma, \quad \delta_1 = \frac{1}{2} - 2\beta + \gamma, \quad \delta_2 = \beta$$

and

$$\mathbf{F}_{n+j-\alpha_f} = \mathbf{F}(\alpha_f t_{n+j+1} + (1 - \alpha_f)t_{n+j}) = \mathbf{F}(t_{n+j} + \alpha_f\tau)$$

$$\mathbf{S}_{n+j+\alpha_f} = \alpha_f\mathbf{S}(\mathbf{u}_{n+j+1}) + (1 - \alpha_f)\mathbf{S}(\mathbf{u}_{n+j}).$$

The method has consistency order 2 if $\gamma = \frac{1}{2} + \alpha_m - \alpha_f$. The method is zero-stable and convergent if $\alpha_m \geq 1/2$, $\alpha_f \leq 1/2$ and $\gamma \leq 1/2$ (see [EBB02]). For stability often the setting

$$\beta = \frac{(1 + \alpha_m - \alpha_f)^2}{4}, \alpha_f = \frac{1}{1 + \rho_\infty}, \alpha_m = \frac{2 - \rho_\infty}{1 + \rho_\infty}$$

is used (see [CH93]).

4 ADAPTIVITY

4.1 Adaptivity for the onestep version

If the generalised-α methods are formulated as onestep methods the so-called PI-controller from Gustafsson et. al. [GLS88] can be used. To suggest a new timestep size we need solutions of order p and $p - 1$. The approximation of the generalised-α method can be used as a second order approximation since the error constant is very small and the methods behave in our numerical experiments as a second order method. As the second solution with order 1 we use the backward Euler method. The next timestep size τ_{n+1} is proposed to be

$$\tau_{n+1} = \rho \frac{\tau_n^2}{\tau_{n-1}} \left(\frac{TOL \cdot r_n}{r_{n+1}^2} \right)^{1/2}, \tag{24}$$

where $\rho \in (0, 1]$ is a safety factor, $TOL > 0$ is a given tolerance, and $r_{n+1} := \|\mathbf{u}_{n+1} - \hat{\mathbf{u}}_{n+1}\|$. In [HNW93, HW96, SW92] different error measures can be found, which use a combination of relative and absolute errors. For further details about the numerical error and the implementation of automatic steplength control we refer to [HW96, Lan01]. The algorithm reads as follows:

- Compute the numerical solution $(\mathbf{u}_{n+1}, \dot{\mathbf{u}}_{n+1})^\top$ with the help of the generalised-α method (6), (7).

- Compute the second solution with the backward Euler method and use $\dot{\mathbf{u}}_{n+1}$ as approximation for $\mathbf{f}(t_{n+1}, \mathbf{u}_{n+1})$, i. e. $\hat{\mathbf{u}}_{n+1} = \mathbf{u}_n + \tau_n \dot{\mathbf{u}}_{n+1}$.

- Compute the numerical error with r_{n+1} and approximate the new timestep length τ_{n+1} with (24).

- If the numerical error is smaller than the given tolerance the timestep is accepted, otherwise it is rejected and has to be recomputed with the new timestep length τ_{n+1}.

In case of second order ODEs we use equations (12)–(17) to compute the numerical approximation $\mathbf{u}_{n+1}$. As in the case of the first order ODEs the backward Euler can be used for computing the first order solution.

The chemical reaction E5 This chemical reaction problem is called E5 and can be found in the collection by Enright, Hull, and Lindberg [EHL75]. The equations are given by

$$\begin{aligned}
\dot{u}_1 &= -Au_1 - Bu_1u_3, \\
\dot{u}_2 &= Au_1 - MCu_2u_3, \\
\dot{u}_3 &= Au_1 - Bu_1u_3 - MCu_2u_3 + Cu_4, \\
\dot{u}_4 &= Bu_1u_3 - Cu_4
\end{aligned}$$

with the initial conditions $u_1(0) = 1,76 \times 10^{-3}$ and $u_i(0) = 0$, $i \in \{2, 3, 4\}$. Moreover we set as in [HW96] $A = 7,89 \times 10^{-10}$, $B = 1,1 \times 10^7$, $C = 1,13 \times 10^3$, and $M = 10^6$. The equations should be solved in the time interval $[0, 10^{13}]$. Note that the variables u_2, u_3, and u_4 satisfy the equation $u_2 - u_3 - u_4 = 0$. The parameter ρ is chosen to be 0, 1/4, 1/2, 3/4, and 9/10, resp. For ρ tending to 1 the algorithm becomes instable. We compare the generalised-α methods with other implicit and linear-implicit second order solvers like ROS2 (see [VSBH99]), ROS2S (see [HHR12]), and the method of Ellsiepen (see [EH01]). It can be observed from Figure 1 that the generalised-α methods with the new stepsize controller are more effective than the other second order methods.

Figure 1: Comparison of generalised-α methods for first order ODEs: CPU time versus error

Kepler's problem   Consider the second order ODE

$$\ddot{y}_i = -\frac{y_i}{(y_1^2 + y_2^2)^{3/2}}, \quad i = 1, 2.$$

The initial conditions are given by $\mathbf{u}_0 = \left(0, \sqrt{\dfrac{1+e}{1-e}}, 1-e, 0\right)^{\top}$, where $e \in [0, 1)$ is a given parameter. In our numerical example we choose $e = 1/2$. We solve the problem in the interval $[0, 20000]$ with the generalised-α methods for second order ODEs and use the new adaptive timestep control. The parameter ρ is chosen to be 0, 1/4, 1/2, 3/4, and 9/10, resp. We compare the generalised-α methods with other implicit and linear-implicit second order solvers like ROS2 (see [VSBH99]), ROS2S (see [HHR12]), and the method of

Ellsiepen (see [EH01]). It can be observed from Figure 2 that the generalised-α methods with the new stepsize controller are more effective than the other second order methods. In this case we get better results for a larger ρ.
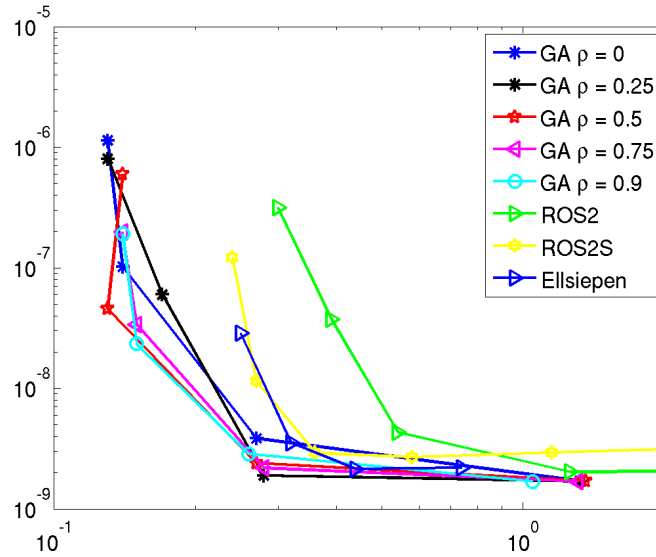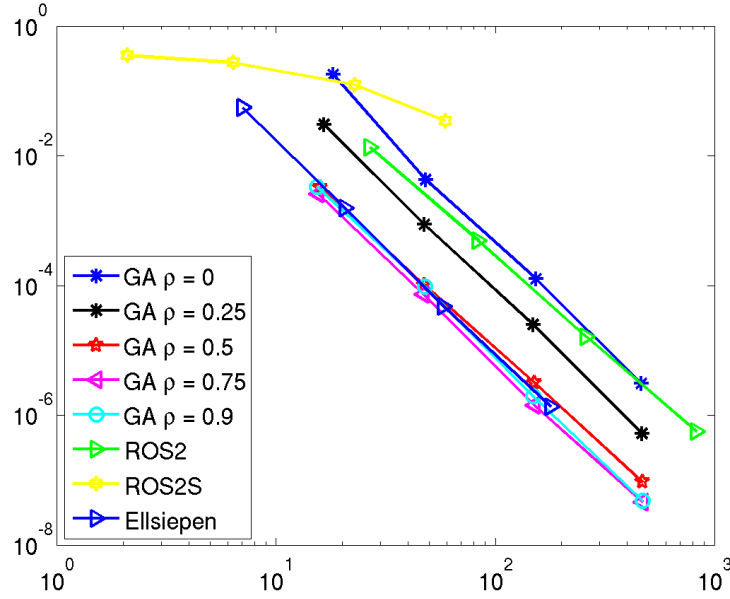

Figure 2: Comparison of generalised-α methods for second order ODEs: CPU time versus error

4.2   Adaptivity for the multistep version

In this section we derive first a multistep formula for the generalised-α method, which has variable coefficients. Note that this idea works only for the generalised-α method for first order problems. In the case of the generalised-α method for second order problems matrix potentials must be computed.

Let us start with the generalised-α method for first order problems. We want to formulate this method as a twostep method with variable coefficients. We consider equations (6) for t_n and t_{n+1} and (7) for t_n. A simple calculation leads to

$$\mathbf{u}_{n+1} = \mathbf{u}_n + \frac{\tau_{n+1}}{\tau_n}\frac{\alpha_m - 1}{\alpha_m}(\mathbf{u}_n - \mathbf{u}_{n-1}) - \tau_{n+1}\frac{\gamma - 1}{\alpha_m}\mathbf{f}_{n-1+\alpha_f} + \tau_{n+1}\frac{\gamma}{\alpha_m}\mathbf{f}_{n+\alpha_f}.$$

The BDF-2 method with variable timesteps is a special case with the setting $\gamma = \alpha_f = 1$ and $\alpha_m = (2\tau_{n-1} + \tau_n)(\tau_{n-1} + \tau_n)$.

Next we want to derive the condition for order 2. Therefore we compute a Taylor expansion of $\mathbf{u}_{n+1}$ and obtain

$$\mathbf{u}_{n+1} = \mathbf{u}_n + \tau_{n+1}\dot{\mathbf{u}}_n + \frac{\tau_{n+1}}{\alpha_m}\left[-\frac{\tau_n}{2}(\alpha_m - 1) - \tau_n(\gamma - 1)(\alpha_f - 1) + \tau_{n+1}\gamma\right]\ddot{\mathbf{u}}_n + \mathcal{O}(\tau_{n+1}^3)$$

It follows

$$t_n(1 - \alpha_m) - 2\tau_n(\gamma - 1)(\alpha_f - 1) + 2\gamma\alpha_f\tau_{n+1} = \tau_{n+1}\alpha_m. \tag{25}$$

Next we consider the problem $\dot{u} = 0$. Applying the adaptive generalised-α method we obtain

$$u_{n+1} - \left(1 + \omega_n\frac{\alpha_m - 1}{\alpha_m}\right)u_n + \omega_n\frac{\alpha_m - 1}{\alpha_m}u_{n-1} = 0.$$

It follows

$$\xi^2 - \left(1 + \omega_n\frac{\alpha_m - 1}{\alpha_m}\right)\xi + \omega_n\frac{\alpha_m - 1}{\alpha_m} = 0,$$

which has the solutions $\xi_1 = 1$ and $\xi_2 = \omega_n(\alpha_m - 1)/\alpha_m$. It follows

$$\frac{\omega_n}{\omega_n + 1} \leq \alpha_m \leq \frac{\omega_n}{|\omega_n - 1|}.$$

For the A-stability of the method we consider the problem $\dot{u} = \lambda u$, $\lambda < 0$. Using the adapative generalised-α method we get

$$u_{n+1} = \left(1 + \omega_n\frac{\alpha_m - 1}{\alpha_m}\right)u_n - \omega_n\frac{\alpha_m - 1}{\alpha_m}u_{n-1} + \frac{\tau_{n+1}(1 - \gamma)}{\alpha_m}\lambda((1 - \alpha_f)u_{n-1} + \alpha_f u_n))$$
$$+ \frac{\tau_n\gamma}{\alpha_m}\lambda((1 - \alpha_f)u_n + \alpha_f u_{n+1}).$$

We are interested in the case $\lambda \to -\infty$ and get

$$0 = \tau_n\gamma\alpha_f\xi^2 + (\tau_{n+1}(1 - \gamma)\alpha_f + \tau_n\gamma(1 - \alpha_f))\xi + \tau_{n+1}(1 - \gamma)(1 - \alpha_f).$$

The solutions of this equation are given by

$$\xi_1 = \frac{\alpha_f - 1}{\alpha_f}, \quad \xi_2 = \tau_{n+1}\frac{\gamma - 1}{\tau_n\gamma}.$$

As in the case of constant coefficients we solve $\xi_1 = -\rho_\infty$ and $\xi_1 = -\rho_\infty$ together with (25) and get

$$\gamma = \frac{\tau_{n+1}}{\tau_{n+1} + \rho_\infty\tau_n}, \alpha_f = \frac{1}{\rho_\infty + 1}, \alpha_m = \frac{2\tau_{n+1}\tau_n - \tau_n\rho_\infty}{\tau_n + \rho_\infty\tau_n + \tau_{n+1} + \tau_{n+1}\rho_\infty}.$$

For adaptivity we want to use a so-called *predictor-corrector scheme* (see for example [GS00]). The predictor is a scheme, which needs no solution of a linear or nonlinear system, for example an explicit method. The corrector is the desired method, in our case the generalised-α-method.

Let us assume that the predictor and the corrector are of order p, i. e. it holds

$$\mathbf{u}_{n+1}^p - \mathbf{u}(t_{n+1}) = C_p \frac{\tau_{n+1}^{p+1}}{(p+1)!} \mathbf{f}^{(p+1)}(t_n), \tag{26}$$

$$\mathbf{u}_{n+1}^c - \mathbf{u}(t_{n+1}) = C_c \frac{\tau_{n+1}^{p+1}}{(p+1)!} \mathbf{f}^{(p+1)}(t_n), \tag{27}$$

where $\mathbf{u}_{n+1}^p$ is the approximation of the predictor and $\mathbf{u}_{n+1}^c$ is the approximation of the corrector. In the system (26)–(27) the quantities $\mathbf{u}(t_{n+1})$ and $\mathbf{f}^{(p+1)}(t_n)$ are unknown. Therefore we solve equation (26) w.r.t. $\mathbf{u}(t_{n+1})$ and insert it into equation (27). Then we get the following approximation of the local truncation error

$$\mathbf{d}_{n+1} = \mathbf{u}_{n+1}^c - \mathbf{u}(t_{n+1}) = C_c \frac{\tau_{n+1}^{p+1}}{(p+1)!} \mathbf{f}^{(p+1)}(t_n) = \frac{C_c}{C_c - C_p}(\mathbf{u}_{n+1}^c - \mathbf{u}_{n+1}^p). \tag{28}$$

As predictor we want to use the leapfrog method (see [GS00]) given by

$$\mathbf{u}_{n+1}^p = \mathbf{u}_n + \left(1 + \frac{\tau_n}{\tau_{n+1}}\right) \tau_{n+1} \dot{\mathbf{u}}_n - \left(\frac{\tau_{n+1}}{\tau_n}\right)^2 (\mathbf{u}_n - \mathbf{u}_{n-1}).$$

Expanding this formula into a Taylor expansion gives us the constant C_p, which reads as

$$C_p = -\left(1 + \frac{\tau_n}{\tau_{n+1}}\right).$$

The error constant C_c we receive from the Taylor expansion of the generalised-α method. We obtain

$$C_c = \left(\frac{\tau_n}{\tau_{n+1}}\right)^2 \frac{\alpha_m - 1 + 3(1-\gamma)(\alpha_f - 1)^2}{\alpha_m} + 3\frac{\gamma \alpha_f^2}{\alpha_m} - 1.$$

The chemical reaction E5   As numerical example we again choose the chemical reaction problem E5. The parameter ρ is chosen to be 0, 1/4, and 1/2, resp. For $\rho > 1/2$ we obtain too many stepsize rejections and the algorithm becomes ineffective. We compare the generalised-α methods with other implicit and linear-implicit second order solvers like ROS2 (see [VSBH99]), ROS2S (see [HHR12]), and the method of Ellsiepen (see [EH01]). It can be observed from Figure 3 that the generalised-α with the new stepsize controller is not as effective as the other second order methods.

5   Summary and Outlook

In this article we gave a short analysis of the generalised-α method for first and second order ODEs and introduced a new adaptive timestep control. In case of the onestep versions this controller is better than other adaptive second order methods. For multistep methods the adaptivity is more complicated. We have seen that only in the case $\rho \in [0, 1/2]$ an effective method is achieved. In future works adaptivity should also be possible if $\rho > 1/2$.
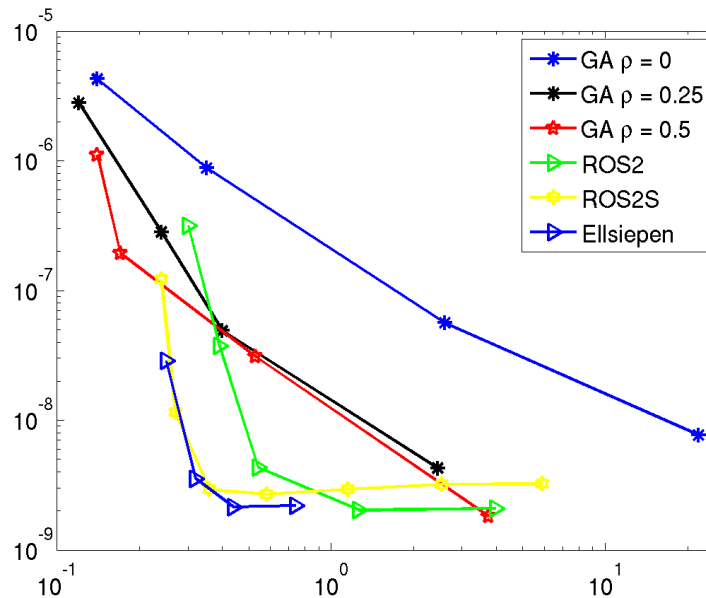
Figure 3: Comparison of generalised-α methods for first order ODEs: CPU time versus error

REFERENCES

[CH93]   J. Chung and G.M. Hulbert. A time integration algorithm for structural dynamics with improved numerical dissipation: The generalized-α method. *J. Appl. Mech.*, 60(2):371–375, 1993.

[DP03]   W. Dettmer and D. Perić. An analysis of the time integration algorithms for the finite element solutions of incompressible Navier-Stokes equations based on a stabilised formulation. *Comput. Methods Appl. Mech. Engrg.*, 192(9-10):1177–1226, 2003.

[EBB02]   S. Erlicher, L. Bonaventura, and O. S. Bursi. The analysis of the generalized-α method for non-linear dynamic problems. *Comput. Mech.*, 28(2):83–104, 2002.

[EH01]   P. Ellsiepen and S. Hartmann. Remarks on the interpretation of current non-linear finite-element-analyses as differential-algebraic equations. *International Journal for Numerical Methods in Engineering*, 51:679–707, 2001.

[EHL75]   W.H. Enright, T.E. Hull, and B. Lindberg. Comparing numerical methods for stiff systems of O.D.E:s. *BIT, Nord. Tidskr. Inf.-behandl.*, 15:10–48, 1975.

[GLS88]   K. Gustafsson, M. Lundh, and G. Söderlind. A PI stepsize control for the numerical solution of ordinary differential equations. *BIT*, 28(2):270–287, 1988.

[GS00]   P.M. Gresho and R.L. Sani. *Incompressible Flow and the Finite Element Method*. Wiley, Chichester, 2000.

[HHR12]   Ahmad-Wahadj Hamkar, Stefan Hartmann, and Joachim Rang. A stiffly accurate Rosenbrock-type method of order 2. *Appl. Num. Math.*, 62(12):1837–1848, 2012.

[HNW93]   E. Hairer, S. P. Nørsett, and G. Wanner. *Solving ordinary differential equations. I: Nonstiff problems.*, volume 8 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 1993.

[HW96]   E. Hairer and G. Wanner. *Solving ordinary differential equations. II: Stiff and differential-algebraic problems*, volume 14 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 1996.

[JR10]   Volker John and Joachim Rang. Adaptive time step control for the incompressible Navier–Stokes equations. *Comput. Methods Appl. Mech. Eng.*, 199:514–524, 2010.

[JWH00]   Kenneth E. Jansen, Christian H. Whiting, and Gregory M. Hulbert. A generalized-α method for integrating the filtered Navier-Stokes equations with a stabilized finite element method. *Comput. Methods Appl. Mech. Eng.*, 190(3-4):305–319, 2000.

[Lan01]   J. Lang. *Adaptive Multilevel Solution of Nonlinear Parabolic PDE Systems*, volume 16 of *Lecture Notes in Computational Science and Engineering*. Springer-Verlag, Berlin, 2001.

[Ran04]   Joachim Rang. *Stability estimates and numerical methods for degenerate parabolic differential equations*. PhD thesis, Institut für Mathematik, TU Clausthal, 2004. appeared also as book from Papierflieger Verlag Clausthal-Zellerfeld.

[SW92]   K. Strehmel and R. Weiner. *Linear-implizite Runge–Kutta-Methoden und ihre Anwendung*, volume 127 of *Teubner-Texte zur Mathematik*. Teubner, Stuttgart, 1992.

[VSBH99]   J.G. Verwer, E.J. Spee, J.G. Blom, and W. Hundsdorfer. A second-order Rosenbrock method applied to photochemical dispersion problems. *SIAM J. Sci. Comput.*, 20(4):1456–1480, 1999.

FAST TIME IMPLICIT DISCRETIZATION FOR COMPRESSIBLE FLOW EQUATIONS VIA A DISCONTINUOUS GALERKIN METHOD

FLORENT RENAC[*] AND EMERIC MARTIN[†]

[*]ONERA The French Aerospace Lab
92320 Châtillon Cedex, France
e-mail: florent.renac@onera.fr, www.onera.fr

[†]ONERA The French Aerospace Lab
92320 Châtillon Cedex, France
e-mail: emeric.martin@onera.fr, www.onera.fr

Key words: discontinuous Galerkin method, implicit-time integration, compressible flows

Abstract. In this study, we investigate efficient time integration techniques for a high-order accurate discontinuous Galerkin method. The method is associated to a Jacobian-free Newton-Krylov algorithm. This method is known to resolve the problem of strong restriction on the time step due to the so-called Courant-Friedrichs-Levy condition for stability of the discontinuous Galerkin scheme associated to an explicit time discretization. However, the shortcoming of implicit time integration methods is the extremely high computational cost and memory requirement induced by the large number of degrees of freedom in practical applications. In the present work, we focus on efficient preconditioning techniques. In a first time, we will review and compare techniques in the context of Jacobian-free Newton-Krylov algorithm such as block-Jacobi, LU SGS and ILU(0) preconditioners. Then, we will exploit the possibility of using approximate Jacobians as preconditioning matrix to reduce the strong computational cost and memory requirement associated to a high-order discontinuous Galerkin method. Steady-state and time-dependent solutions of the compressible Euler equations in two and three space dimensions will be considered to assess the performances of the present method.

A GUARANTEED ERROR BOUND SEPARATING ALGEBRAIC AND DISCRETIZATION CONTRIBUTIONS IN NON-OVERLAPPING DOMAIN DECOMPOSITION METHODS

C. REY∗, P. GOSSELET∗ AND V. REY∗

∗LMT-Cachan
ENS Cachan/CNRS/UPMC/PRES UniverSud Paris
61, avenue du Président Wilson, F-94235 Cachan Cedex, France
e-mail: {first name}.{last name}@lmt.ens-cachan.fr, www.lmt.ens-cachan.fr

Key words: Algebraic error, Discretization error, Stopping criterion, Verification

Abstract. For the last decades, three trends have grown and reinforced each other: the fast growth of hardware computational capacities, the requirement of finer and larger finite element models for industrial simulations and the development of efficient computational strategies amongst which non-overlapping domain decomposition (DD) methods [2, 3, 4] are very popular since they have proved to be scalable in many applications. One main shortcoming in DD lies on the absence of verification of the discretized models in order to warranty the quality of numerical simulations (global or goal-oriented error estimators). In a recent work [1], we introduced a first error estimator in a non-overlapping domain decomposition framework and outlined its connection with two iterative non-overlapping domain decomposition solvers (FETI and BDD). It is fully parallel in the sense that it involves a simple preprocessing of interface tractions and the use of standard black-box sequential error estimators [5, 6, 8] independently on each subdomain. It yields a guaranteed upper bound on the error whatever the state (converged or not) of the iterative solver associated to the interface continuity. It has been numerically observed that our first DD-error estimator enables to recover the same efficiency factor as the standard sequential. However, its main drawback is its inability to separate the algebraic error (coming from the DD iterative solver) from the discretization error per subdomains.

In this talk, we present some of our recent work that aims at separating the algebraic error and the discretization error. We introduce a new guaranteed upper bound that enables to introduce such a separation. This leads to the definition of new convergence criteria of DD iterative solvers based on the estimation of the discretization error instead of purely algebraic criteria. Works in progress are related to (i) goal-oriented error estimator (ii) nonlinear problems.

REFERENCES

[1] A. Parret-Fréaud, C. Rey, P. Gosselet, F. Feyel *Fast estimation of discretization error for FE problems solved by domain decomposition*, Comput. Methods Appl. Mech. Engrg. **199** (2010), 3315–3323.

[2] P. Gosselet, C. Rey *Non-overlapping domain decomposition methods in structural mechanics*, Arch. Comput. Meth. Engng. **13** (2006), 515–572.

[3] C. Farhat, F. X. Roux, *Implicit Parallel Processing in Structural Mechanics*, Computational Mechanics Advances, **2**, (1994), 1–124.

[4] P. Le Tallec, *Domain-decomposition methods in computational mechanics*, Computational Mechanics Advances, **1**, (1994), 121–220.

[5] P. Ladevèze, D. Leguillon, *Error estimate procedure in the finite element method and application*, SIAM J. Numer. Anal. **20** (1983) 485-509.

[6] N. Parés, P. Díez, A. Huerta, *Subdomain-based flux-free a posteriori error estimators*, Comp. Methods Appl. Mech. Eng. **195** (2006) 297-323.

[7] M. Vohralik, *A posteriori error estimates for lowest-order mixed finite element discretization of convection-diffusion-reaction equations*, SIAM J. Numer. Anal. **45** (2007), 1570-1599.

[8] P. Ladevèze, L. Chamoin, E. Florentin, *A new non-intrusive technique for the construction of admissible stress fields in model verification*, Comput. Methods Appl. Mech. Eng. **199** (2010) 766-777.

GOAL-ORIENTED ERROR ESTIMATOR FOR THE FRACTIONAL STEP θ TIME-STEPPING SCHEME

Thomas Richter[*], Dominik Meidner[†]

[*]University of Heidelberg
Institute of Applied Mathematics
INF 294, 69120 Heidelberg
e-mail: thomas.richter@iwr.uni-heidelberg.de

[†]Dominik Meidner
TU Munich
e-mail: meidner@ma.tum.de

Key words: error estimation, fractional step time-stepping, DWR

Abstract.   In this contribution we present an adjoint based a posteriori error estimator for (nonlinear) parabolic problems discretized with the fractional step θ time-stepping scheme. This scheme combines several highly desired attributes: it is second order accurate, strongly A-stable and shows very little numerical dissipation. The drawback of this time-stepping scheme is its time-stepping character based on a finite difference approximation which makes it ill-suited for variational and in particular adjoint error estimation techniques.

We will propose a Petrov-Galerkin scheme, that is shown to be algebraically equivalent to the fractional step θ time-stepping scheme for linear problems and that can be regarded as an approximation of this scheme for general nonlinear problems.

The error estimator is split into two parts: the first is a traditional residual based estimator of the Galerkin scheme, the second measures the defect in Galerkin orthogonality given by the numerical quadrature error. Both estimator parts require the solution of an adjoint in time solution.

EXPLICIT EXPRESSIONS OF DUAL LOADS FOR ACCURATE ERROR ESTIMATION AND BOUNDING IN GOAL ORIENTED ADAPTIVITY

J.J. Ródenas[*], **E. Nadal**[*], **O.A. González-Estrada**[†], **F.J. Fuenmayor**[*], **S.P.A. Bordas**[†], **and P. Kerfriden**[†]

[*]Centro de Investigación en Tecnología de Vehículos (CITV)
Universitat Politècnica de València ,C/ Vera, S/n, E-46022-Valencia, Spain
e-mail: {jjrodena, ffuenmay}@mcm.upv.es, ennaso@upvnet.upv.es

[†]Institute of Mechanics & Advanced Materials,
Cardiff University, School of Engineering, Queen's Building, The Parade, Cardiff CF24 3AA
Wales, UK.
e-mail: {bordass, estradaoag}@cardiff.ac.uk, pierre.kerfriden@gmail.com

Key words: Quantities of Interest, Error estimator, Goal oriented adaptivity

Abstract. Recently, Goal Oriented Adaptivity (GOA) has been an active research area because of its advantages in terms of computational cost and accuracy. This technique consists in solving two Finite Element (FE) problems: the primal one, which is the actual problem and the dual one, which is an auxiliary problem depending on the Quantity of Interest (QoI).

To improve the quality of the error estimate in the QoI we consider a recovery-based procedure which enforces local equilibrium for an accurate stress representation. The proposed procedure requires the explicit expressions for the dual loads which, traditionally, are not obtained in the FE framework. Our objective in this paper is to obtain those explicit expressions for the dual problem for the extraction of linear QoI in the context of linear elasticity. The ZZ-type error estimator is used to evaluate the error in the QoI at element level, yielding a high quality, as shown in the numerical tests.

1 Introduction

The Finite Element (FE) solution is a numerical approximation to the unknown exact solution of a Boundary Value Problem (BVP), thus, there exists an *error* due to the discretization. The most widely used way to estimate the discretization error is to evaluate it in terms of the global energy. A great effort has been devoted since the very beginning by researchers in order to obtain good approximations or even sharp upper bounds for the global error measurement in energy norm [1, 2, 3, 4, 5, 6, 7].

These error estimators were based on the evaluation of an approximation to the true error in energy norm. However, for practitioners this quantity is, in general, not very useful from an industrial point of view. In practice, analysts run simulations in order to evaluate stresses, displacements, *etc...* in a particular area of the domain. In the late 90s, a new paradigm appeared [8, 9, 10] where, instead of evaluating the error of the solution in terms of energy, the error is evaluated in terms of a Quantity of Interest (QoI) in a Domain of Interest (DoI). That is, some relevant quantity are considered as the main output. Then, we directly control the error of the QoI in the DoI. The error estimation of a QoI requires solving two problems simultaneously, the first one is called primal problem and is the one we are interested in. The second problem, called dual or adjoint problem, serves to extract the information for the error in the QoI. Both problems are geometrically identical and differ on the applied loads. Those of the dual problem depend on the DoI and the QoI. The construction of the dual problem will be explained later in more detail.

Our approach to obtain estimations of the error in the QoI, in contrast to previous techniques, is based on the use of equilibrated recovered fields obtained for the solution of both, the primal and the dual problem. The proposed procedure begins with the evaluation of displacement recovered fields considering: the fulfilment of boundary and internal equilibrium equations, Dirichlet constraints and, for singular problems, the splitting of the displacement and stress fields into singular and smooth parts, as described in [11]. Similar recovery techniques considering stresses were previously used to obtain upper bounds of the error in energy norm in [12, 13]. For the recovery procedure we need the analytical expressions defining the loads for the primal and dual problems. Thus, for the dual problem, we must obtain the analytical expressions related to the QoI required during the recovery process.

Numerical tests using 2D benchmark problems with exact solution are used to investigate the quality of the proposed technique. Results for different quantities of interest show that the technique provides excellent error estimates which can be used in goal oriented adaptive procedures.

2 Problem Statement

2.1 Primal problem

In this section we briefly present the model for the 2D linear elasticity problem. Denote, in vectorial form, $\boldsymbol{\sigma}$ and $\boldsymbol{\varepsilon}$ as the stresses and strains, $\mathbf{D}$ as the elasticity matrix of the constitutive relation $\boldsymbol{\sigma} = \mathbf{D}\boldsymbol{\varepsilon}$, and $\mathbf{u}$ the unknown displacement field, which take values in $\Omega \subset \mathbb{R}^2$. $\mathbf{u}$ is the solution of the boundary value problem given by:

$$\mathbf{S}\mathbf{u} = -\mathbf{b} \qquad \text{in } \Omega \qquad (1)$$

$$\boldsymbol{\epsilon}(\mathbf{u}) = \mathbf{L}\mathbf{u} \qquad \text{in } \Omega \qquad (2)$$

$$\mathbf{G}\boldsymbol{\sigma}(\mathbf{u}) = \mathbf{t} \qquad \text{on } \Gamma_N \qquad (3)$$

$$\mathbf{u} = \bar{\mathbf{u}} \qquad \text{on } \Gamma_D, \qquad (4)$$

Γ_N and Γ_D denote the Neumann and Dirichlet boundaries with $\partial\Omega = \overline{\Gamma_N \cup \Gamma_D}$ and $\Gamma_N \cap \Gamma_D = \emptyset$, $\mathbf{b}$ are body loads and $\mathbf{t}$ are the tractions imposed along Γ_N. $\mathbf{S} = \mathbf{L}^T\mathbf{D}\mathbf{L}$, being $\mathbf{L}$ the differential operator, and $\mathbf{G}$ is the projection operator that projects the stress field into tractions over any boundary with outward normal vector $\mathbf{n} = \{n_x \ n_y\}^T$:

$$\mathbf{L} = \begin{bmatrix} \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} \end{bmatrix} \qquad (5)$$

$$\mathbf{G} = \begin{bmatrix} n_x & 0 & n_y \\ 0 & n_y & n_x \end{bmatrix} \qquad (6)$$

Consider the initial stresses $\boldsymbol{\sigma}_0$ and strains $\boldsymbol{\varepsilon}_0$, the symmetric bilinear form $a : (V + \bar{\mathbf{u}}) \times V \to \mathbb{R}$ and the continuous linear form $\ell : V \to \mathbb{R}$ defined by:

$$a(\mathbf{u}, \mathbf{v}) := \int_\Omega \boldsymbol{\sigma}^T(\mathbf{u})\boldsymbol{\varepsilon}(\mathbf{v})\mathrm{d}\Omega = \int_\Omega \boldsymbol{\sigma}^T(\mathbf{u})\mathbf{D}^{-1}\boldsymbol{\sigma}(\mathbf{v})\mathrm{d}\Omega \qquad (7)$$

$$\ell(\mathbf{v}) := \int_\Omega \mathbf{v}^T\mathbf{b}\mathrm{d}\Omega + \int_{\Gamma_N} \mathbf{v}^T\mathbf{t}\mathrm{d}\Gamma + \int_\Omega \boldsymbol{\sigma}^T(\mathbf{v})\boldsymbol{\varepsilon}_0\mathrm{d}\Omega - \int_\Omega \boldsymbol{\varepsilon}^T(\mathbf{v})\boldsymbol{\sigma}_0\mathrm{d}\Omega. \qquad (8)$$

With these notations, the variational form of the problem reads [14]:

$$\text{Find } \mathbf{u} \in (V + \bar{\mathbf{u}}) : \forall \mathbf{v} \in V \quad a(\mathbf{u}, \mathbf{v}) = \ell(\mathbf{v}) \qquad (9)$$

where V is the standard test space for the elasticity problem such that $V = \{\mathbf{v} \mid \mathbf{v} \in [H^1(\Omega)]^2, \mathbf{v}|_{\Gamma_D}(\mathbf{x}) = \mathbf{0}\}$.

Let $\mathbf{u}^h$ be a finite element approximation of $\mathbf{u}$. The solution for the discrete counterpart of the variational problem in (9) lies in a subspace $(V^h + \bar{\mathbf{u}}) \subset (V + \bar{\mathbf{u}})$ associated with a mesh of finite elements of characteristic size h, and it is such that:

$$\forall \mathbf{v}^h \in V^h \subset V \qquad a(\mathbf{u}^h, \mathbf{v}^h) = \ell(\mathbf{v}^h). \qquad (10)$$

Consider the linear elasticity problem given in (9) and its approximate FE solution $\mathbf{u}^h \in V^h \subset V$. This problem is related to the original problem to be solved, that henceforth will be called the *primal problem*.

2.2 Dual problem

Now, let us define $Q : V \to \mathbb{R}$ as a bounded linear functional representing some quantity of interest, acting on the space V of admissible functions for the problem at hand. The objective is to estimate the error in $Q(\mathbf{u})$ when calculated using the value of the approximate solution $\mathbf{u}^h$:

$$Q(\mathbf{u}) - Q(\mathbf{u}^h) = Q(\mathbf{u} - \mathbf{u}^h) = Q(\mathbf{e}). \tag{11}$$

As will be shown later, $Q(\mathbf{v})$ may be interpreted as the work associated with a displacement field $\mathbf{v}$ and a distribution of forces specific to each type of quantity of interest. If we particularise $Q(\mathbf{v})$ for $\mathbf{v} = \mathbf{u}$, this force distribution will allow us to extract information concerning the quantity of interest associated with the solution of the problem in (9).

A standard procedure [15] to evaluate $Q(\mathbf{e})$ consists in solving the auxiliary *dual* problem (also called *adjoint* or *extraction* problem) defined as:

$$\text{Find } \mathbf{w} \in V : \forall \mathbf{v} \in V \quad a(\mathbf{v}, \mathbf{w}) = Q(\mathbf{v}). \tag{12}$$

An exact representation for the error $Q(\mathbf{e})$ in terms of the solution of the dual problem can be simply obtained by substituting $\mathbf{v} = \mathbf{e}$ in (12) and remarking that for all $\mathbf{w}_Q^h \in V^h$, due to the Galerkin orthogonality, $a(\mathbf{e}, \mathbf{w}_Q^h) = 0$ such that:

$$Q(\mathbf{e}) = a(\mathbf{e}, \mathbf{w}) = a(\mathbf{e}, \mathbf{w}) - \overset{0}{\underset{}{a(\mathbf{e}, \mathbf{w}^h)}} = a(\mathbf{e}, \mathbf{w} - \mathbf{w}^h) = a(\mathbf{e}, \boldsymbol{\epsilon}). \tag{13}$$

Therefore, the error in evaluating $Q(\mathbf{u})$ using $\mathbf{u}^h$ is given by:

$$Q(\mathbf{u}) - Q(\mathbf{u}^h) = Q(\mathbf{e}) = a(\mathbf{e}, \boldsymbol{\epsilon}) = \int_\Omega \left(\boldsymbol{\sigma}_p - \boldsymbol{\sigma}_p^h \right) \mathbf{D}^{-1} \left(\boldsymbol{\sigma}_d - \boldsymbol{\sigma}_d^h \right) \mathrm{d}\Omega, \tag{14}$$

where $\boldsymbol{\sigma}_p$ is the stress field associated with the solution of the primal problem and $\boldsymbol{\sigma}_d$ is the one associated with the dual problem. Using the Zienkiewicz and Zhu (ZZ) error estimator [16] and (14) we can derive an estimate for the error in the QoI which reads:

$$Q(\mathbf{e}) \approx Q(\mathbf{e}_{es}) = \int_\Omega \left(\boldsymbol{\sigma}_p^* - \boldsymbol{\sigma}_p^h \right) \mathbf{D}^{-1} \left(\boldsymbol{\sigma}_d^* - \boldsymbol{\sigma}_d^h \right) \mathrm{d}\Omega, \tag{15}$$

where $\boldsymbol{\sigma}_p^*$ and $\boldsymbol{\sigma}_d^*$ represent the recovered stress fields for the primal and dual problems, respectively. Here, we expect to have a sharp estimate of the error in the QoI if the recovered stress fields are accurate approximations to their exact counterparts.

In order to obtain accurate representations of the exact stress fields both for the primal and dual solutions, we propose the use of a locally equilibrated displacement recovery technique, called SPR-CD, based on the ideas in [17, 11, 13]. This technique, which is an enhancement of the Superconvergent Patch Recovery (SPR) proposed in [18], enforces the fulfillment of the internal, boundary equilibrium equations and Dirichlet boundary conditions locally on patches. For problems with singularities the stress field is also decomposed into two parts: smooth and singular, which are separately recovered.

3 Quantities of Interest

The recovery procedure based on the SPR technique and denoted as SPR-CD, fully described in [19], relies on the *a priori* known values of $\mathbf{b}$, $\mathbf{t}$, $\boldsymbol{\varepsilon}_0$, $\boldsymbol{\sigma}_0$ and Dirichlet boundary conditions to impose the internal and boundary equilibrium equations and the exact displacements over Γ_D. Regarding the loads, these values are already available for the primal problem ($\mathbf{b}_p$ and $\mathbf{t}_p$). However, the body forces $\mathbf{b}_d$, boundary tractions $\mathbf{t}_d$, etc... are not known for the dual problem. We can easily derive expressions associated to certain linear QoIs, *e.g.* the mean values of displacements and stresses in a sub-domain of interest Ω_i, which can be interpreted in terms of $\mathbf{b}_d$ and $\mathbf{t}_d$. This approach was first introduced in [20] and presented later in [21]. Similarly, in [22] the authors defined the relation between the natural quantities of interest and dual loading data.

3.1 Mean displacement in Ω_i

Let us assume that the objective is to evaluate the mean value of the displacements along the direction α in a sub-domain of interest $\Omega_i \subset \Omega$. The functional for the quantity of interest can be written as:

$$Q(\mathbf{u}) = \bar{u}_\alpha|_{\Omega_i} = \frac{1}{|\Omega_i|} \int_{\Omega_i} \mathbf{u}^T \mathbf{c}_{u_\alpha} \mathrm{d}\Omega, \tag{16}$$

where $|\Omega_i|$ is the volume of Ω_i and $\mathbf{c}_{u_\alpha}$ is a vector used to select the appropriate combination of components of $\mathbf{u}$. For example, $\mathbf{c}_{u_\alpha} = \{1,0\}^T$ if α is parallel to the x-axis. Now, considering $\mathbf{v} \in V$ in (16) results in:

$$Q(\mathbf{v}) = \int_{\Omega_i} \mathbf{v}^T \left(\frac{\mathbf{c}_{u_\alpha}}{|\Omega_i|} \right) \mathrm{d}\Omega = \int_{\Omega_i} \mathbf{v}^\mathrm{T} \mathbf{b}_\mathrm{d} \mathrm{d}\Omega. \tag{17}$$

Note that the term $\mathbf{c}_{u_\alpha}/|\Omega_i|$ formally corresponds to a vector of body forces in the problem defined in (9). Therefore, we can say that $\mathbf{b}_d = \mathbf{c}_{u_\alpha}/|\Omega_i|$ is a constant vector of body loads that applied in the sub-domain of interest Ω_i can be used in the dual problem to extract the mean displacements.

3.2 Mean displacement along Γ_i

For the case where the quantity of interest is the functional that evaluates the mean value of the displacements along a given boundary Γ_i the expression reads:

$$Q(\mathbf{u}) = \bar{u}_\alpha|_{\Gamma_i} = \frac{1}{|\Gamma_i|} \int_{\Gamma_i} \mathbf{u}^T \mathbf{c}_{u_\alpha} \mathrm{d}\Gamma, \tag{18}$$

$|\Gamma_i|$ being the length of Γ_i and $\mathbf{c}_{u_\alpha}$ a vector used to select the appropriate component of $\mathbf{u}$. Again, considering $\mathbf{v} \in V$ in (18) we have:

$$Q(\mathbf{v}) = \int_{\Gamma_i} \mathbf{v}^T \left(\frac{\mathbf{c}_{u_\alpha}}{|\Gamma_i|} \right) \mathrm{d}\Gamma = \int_{\Gamma_i} \mathbf{v}^\mathrm{T} \mathbf{t}_\mathrm{d} \mathrm{d}\Gamma \tag{19}$$

Note that the term $\mathbf{c}_{u_\alpha}/|\Gamma_i|$ can be interpreted as a vector of tractions applied along the boundary in the problem defined in (9). Thus, $\mathbf{t}_d = \mathbf{c}_{u_\alpha}/|\Gamma_i|$ is a vector of tractions applied on Γ_i that can be used in the dual problem to extract the mean displacements along Γ_i.

3.3   Mean stresses and strains in Ω_i

In the case that our QoI is the mean stress (or strains) in Ω_i we can define the QoI (20) where $\mathbf{c}_{\sigma_\alpha}^T$ is a vector to choose any linear combination of the stress (strain) components.

$$Q(\mathbf{u}) = \bar{\sigma}_\alpha|_{\Omega_i} = \frac{1}{|\Omega_i|} \int_{\Omega_i} \mathbf{c}_{\sigma_\alpha}^T \boldsymbol{\sigma} \mathrm{d}\Omega = \int_{\Omega_i} \frac{\mathbf{c}_{\sigma_\alpha}^T}{|\Omega_i|} \boldsymbol{\sigma} \mathrm{d}\Omega \tag{20}$$

Comparing the last integral in (20) with (8) we can define $\boldsymbol{\epsilon}_{0,d} = \mathbf{c}_{\sigma_\alpha}^T/|\Omega_i|$ corresponding to the term of initial strains that we need to apply in the dual problem to extract the value of $\bar{\sigma}_\alpha|_{\Omega_i}$. A similar formulation can be derived for the case of the mean strains in Ω_i such that $\boldsymbol{\sigma}_{0,d} = \mathbf{c}_{\epsilon_\alpha}^T/|\Omega_i|$. Note that the loads for the dual problem of this QoI could also be obtained applying the divergence theorem, yielding tractions along the boundary of the DoI which are equivalent to the initial strains.

3.4   Mean tractions along Γ_i

Let $\mathbf{t} = \{t_n, t_t\}^T$, with t_n and t_t the normal and tangential components of the tractions vector $\mathbf{t}$. Let us assume that we want to evaluate, for example, the mean normal tractions along boundary Γ_i. The functional that defines the mean tractions along the boundary Γ_i can be expressed as

$$Q(\mathbf{u}) = \bar{\mathbf{t}}_n = \frac{1}{|\Gamma_i|} \int_{\Gamma_i} \mathbf{t}^T \mathbf{c} \mathrm{d}\Gamma \tag{21}$$

Using (21) and considering the extraction vector $\mathbf{c}$ and the rotation matrix $\mathbf{R}_\Gamma$ that aligns the tractions normal to the boundary Γ_i we have:

$$\begin{aligned}
\bar{\mathbf{t}}_n &= \frac{1}{|\Gamma_i|} \int_{\Gamma_i} \mathbf{t}^T \mathbf{c} \mathrm{d}\Gamma = \frac{1}{|\Gamma_i|} \int_{\Gamma_i} \{t_n \quad t_t\} \begin{Bmatrix} 1 \\ 0 \end{Bmatrix} \mathrm{d}\Gamma = \\
&= \frac{1}{|\Gamma_i|} \int_{\Gamma_i} \{t_x \quad t_y\} \mathbf{R}_\Gamma^T \begin{Bmatrix} 1 \\ 0 \end{Bmatrix} \mathrm{d}\Gamma = \int_{\Gamma_i} \{t_x \quad t_y\} \frac{\mathbf{R}_\Gamma^T}{|\Gamma_i|} \begin{Bmatrix} 1 \\ 0 \end{Bmatrix} \mathrm{d}\Gamma = \int_{\Gamma_i} \{t_x \quad t_y\} \mathbf{u}_d \mathrm{d}\Gamma
\end{aligned} \tag{22}$$

In (22) the term $\mathbf{u}_d = \mathbf{R}_\Gamma^T \mathbf{c}/|\Gamma_i|$ corresponds to a vector of displacements used as Dirichlet boundary conditions for the dual problem used to extract the mean value of the normal tractions along Γ_i.

Note that in this case $\bar{\mathbf{w}}_Q = \mathbf{u}_d \neq 0$, then (13) does not hold. We redefine the dual problem in (12), $\forall \mathbf{v} \in V$, such that:

$$\begin{aligned}
a(\mathbf{v}, \mathbf{w}_Q) &= 0 \quad \text{in } \Omega \\
\mathbf{w}_Q &= \mathbf{u}_d \quad \text{on } \Gamma_i
\end{aligned} \tag{23}$$

The dual solution can be expressed as $\mathbf{w}_Q = \mathbf{w}_Q^0 + \bar{\mathbf{w}}_Q$, where $\mathbf{w}_Q^0 = 0$ on Γ_i. Assuming that $\bar{\mathbf{w}}_Q = \mathbf{u}_d$ is in the FE solution space, the FE approximation for (23) is also decomposed into two parts $\mathbf{w}_Q^h = \mathbf{w}_Q^{h0} + \bar{\mathbf{w}}_Q$ where, again, $\mathbf{w}_Q^{h0} = 0$ on Γ_i. Therefore, we have for the dual problem:

$$\forall \mathbf{v} \in V \quad a(\mathbf{v}, \mathbf{w}_Q^0) = -a(\mathbf{v}, \bar{\mathbf{w}}_Q) \tag{24}$$

Substituting $\mathbf{v} = \mathbf{e}$ in (24), using the Galerkin orthogonality property, $a(\mathbf{e}, \mathbf{w}_Q^{h0}) = 0$, and considering that $\mathbf{e}_Q^0 = \mathbf{e}_Q$ we write:

$$a(\mathbf{e}, \mathbf{w}_Q^0 - \mathbf{w}_Q^{h0}) = a(\mathbf{e}, \mathbf{e}_Q^0) = a(\mathbf{e}, \mathbf{e}_Q) = -a(\mathbf{e}, \bar{\mathbf{w}}_Q) \tag{25}$$

Similarly, the QoI can also be rewritten by means of the divergence theorem. Thus, generalizing (22) $\forall \mathbf{v} \in V$ we have:

$$Q(\mathbf{v}) = \int_{\Gamma_i} (\mathbf{G}\boldsymbol{\sigma}(\mathbf{v}))^T \mathbf{u}_d \, \mathrm{d}\Gamma = \int_{\Omega_i} \boldsymbol{\sigma}(\mathbf{v})^{\mathrm{T}} \epsilon(\mathbf{u}_d) \, \mathrm{d}\Omega = a(\mathbf{v}, \mathbf{u}_d) \tag{26}$$

Thus, $Q(\mathbf{e}) = a(\mathbf{e}, \mathbf{u}_d) = a(\mathbf{e}, \bar{\mathbf{w}}_Q)$ and substituting in (25) we obtain the error for this QoI: $Q(\mathbf{e}) = -a(\mathbf{e}, \mathbf{e}_Q)$.

3.5 Generalized stress intensity factor in Ω_i

The Generalized Stress Intensity Factor (GSIF) K is the characterizing parameter in problems with singularities. The GSIF is a multiplicative constant that depends on the loading of the problem and linearly determines the intensity of the displacement and stress fields in the vicinity of the singular point. In the particular case that the corners that produce the singularities have an angle of 2π, this parameter is called the Stress Intensity Factor (SIF).

Let us consider the general singular problem of a V-notch domain subjected to loads in the infinite. The analytical solution for this singular elasticity problem can be found in [23] where, considering a polar reference system centred in the corner, the displacement and stress fields at points sufficiently close to the corner can be described as:

$$\mathbf{u}(r, \phi) = K_{\mathrm{I}} r^{\lambda_{\mathrm{I}}} \boldsymbol{\Psi}_{\mathrm{I}}(\lambda_{\mathrm{I}}, \phi) + K_{\mathrm{II}} r^{\lambda_{\mathrm{II}}} \boldsymbol{\Psi}_{\mathrm{II}}(\lambda_{\mathrm{II}}, \phi) \tag{27}$$

$$\boldsymbol{\sigma}(r, \phi) = K_{\mathrm{I}} \lambda_{\mathrm{I}} r^{\lambda_{\mathrm{I}}-1} \boldsymbol{\Phi}_{\mathrm{I}}(\lambda_{\mathrm{I}}, \phi) + K_{\mathrm{II}} \lambda_{\mathrm{II}} r^{\lambda_{\mathrm{II}}-1} \boldsymbol{\Phi}_{\mathrm{II}}(\lambda_{\mathrm{II}}, \phi) \tag{28}$$

where r is the radial distance to the corner, λ_m (with $m = \mathrm{I}, \mathrm{II}$) are the eigenvalues that determine the order of the singularity, $\boldsymbol{\Psi}_m$ and $\boldsymbol{\Phi}_m$ are sets of trigonometric functions that depend on the angular position ϕ, and K_m are the Generalised Stress Intensity Factors (GSIFs). For the evaluation of the GSIF we consider the expression shown in [24]:

$$K^{(1,2)} = -\frac{1}{C} \int_{\Omega^*} \left[\left(u_x^{(2)} \frac{\partial q}{\partial x} \right) \sigma_{xx}^{(1)} + \left(u_y^{(2)} \frac{\partial q}{\partial y} \right) \sigma_{yy}^{(1)} + \left(u_x^{(2)} \frac{\partial q}{\partial y} + u_y^{(2)} \frac{\partial q}{\partial x} \right) \sigma_{xy}^{(1)} \right.$$
$$\left. - \left(\sigma_{xx}^{(2)} \frac{\partial q}{\partial x} + \sigma_{xy}^{(2)} \frac{\partial q}{\partial y} \right) u_x^{(1)} - \left(\sigma_{xy}^{(2)} \frac{\partial q}{\partial x} + \sigma_{yy}^{(2)} \frac{\partial q}{\partial y} \right) u_y^{(1)} \right] \mathrm{d}\Omega, \tag{29}$$

where $u^{(1)}$, $\sigma^{(1)}$ are the displacement and stress fields from the FEM solution, $u^{(2)}$, $\sigma^{(2)}$ are the auxiliary fields associated with the extraction functions for the GSIFs in mode I or mode II, q is an arbitrary function used to define the extraction zone, which is one at the singular point and 0 on the boundaries. x_j refers to the local coordinates system at the crack tip. For more details we refer the reader to [24].

Rearranging terms of the integral in (29), we can obtain:

$$K^{(1,2)} = \int_\Omega \mathbf{u}^{(1)T} \left(-\frac{1}{C}\right) \begin{bmatrix} \sigma_{xx}^{(2)} q_{,1} + \sigma_{xy}^{(2)} q_{,2} \\ \sigma_{xy}^{(2)} q_{,1} + \sigma_{yy}^{(2)} q_{,1} \end{bmatrix} + \boldsymbol{\sigma}^{(1)T} \left(-\frac{1}{C}\right) \begin{bmatrix} u_x^{(2)} q_{,1} \\ u_y^{(2)} q_{,2} \\ u_y^{(2)} q_{,1} + u_x^{(2)} q_{,2} \end{bmatrix} \mathrm{d}\Omega \quad (30)$$

where $q_{,1} = \partial q/\partial x$ and $q_{,2} = \partial q/\partial y$. Rewriting the previous expression we obtain:

$$K^{(1,2)} = \int_\Omega \left(\mathbf{u}^{(1)T}\mathbf{A} + \boldsymbol{\sigma}^{(1)T}\mathbf{B}\right) \mathrm{d}\Omega \quad (31)$$

Thus, if we replace $\mathbf{u}$ by a vector of arbitrary displacements $\mathbf{v}$, the quantity of interest can be defined as:

$$Q(\mathbf{v}) = \int_\Omega \mathbf{v}^T \mathbf{b}_d \mathrm{d}\Omega + \int_\Omega \mathbf{L}\mathbf{v}^T \mathbf{D}\boldsymbol{\varepsilon}_{0d} \mathrm{d}\Omega \quad (32)$$

where $\mathbf{A}$ has been replaced by the dual body forces $\mathbf{b}_d$ and the term $\mathbf{B}$ has been replaced by the vector of initial strains $\boldsymbol{\varepsilon}_{0d}$. It must be taken into account that these expressions can be used either for mode I or mode II.

4 Numerical examples

To verify the influence of the analytical dual loads we compare the standard SPR with our new approach, called SPR-CD, using a singular problem. Plane strain and 2D linear elastic behavior are considered. A bilinear (Q4) h-adaptive refinement process, guided by the error in the quantity of interest, has been considered in all examples. To assess the performance of the recovery procedure and error indicators we have considered some quantities: i) the global effectivity θ that indicates the relation between the exact error $Q(\mathbf{e})$ and the estimated error $Q(\mathbf{e}_{es})$:

$$\theta = \frac{Q(\mathbf{e}_{es}) - Q(\mathbf{e})}{|Q(\mathbf{e})|}, \quad (33)$$

with positive values meaning overestimation of the error and negative values underestimation, and ii) the error in the QoI η_{QoI} which is the relation between the estimated and exact value of the QoI according to the next expression:

$$\eta_{QoI} = \frac{Q(\mathbf{u}^h) + Q(\mathbf{e}_{es})}{Q(\mathbf{u})} \quad (34)$$

4.1 Problem 1. L-Shape plate

The problem model is in Figure 1a. The model is loaded on the boundary with the tractions corresponding to the first symmetric term of the asymptotic expansion that describes the exact solution under mode I or II loading conditions around the singular vertex. The exact displacement and stress fields for the singular elastic problem can be found in [23]. Material parameters are elastic modulus $E = 1000$ and Poisson's ratio $\nu = 0.3$. As we are solving a singular problem, for the recovery we use the *singular+smooth* technique described in [11]. For this problem we consider the GSIF as the QoI, that is K_I or K_II. When K_I is the QoI the primal problem is loaded with $K_\mathrm{I} = 1$ and $K_\mathrm{II} = 0$, and with $K_\mathrm{I} = 0$ and $K_\mathrm{II} = 1$ when K_II is the QoI. In Figure 1 we present a set of h-adapted meshes for K_I. We represent in Figure 2, for K_I and K_II, the results of the proposed

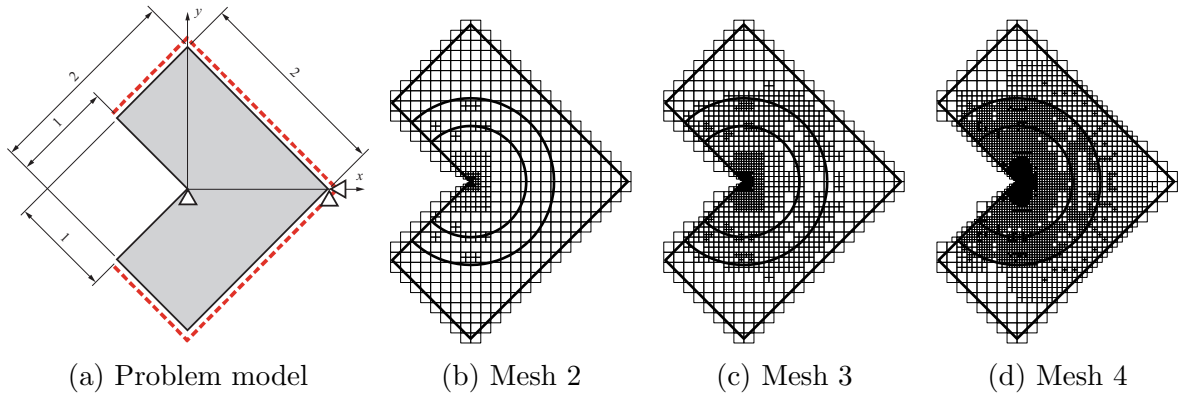| (a) Problem model | (b) Mesh 2 | (c) Mesh 3 | (d) Mesh 4 |

Figure 1: Problem 1. L-Shape plate. Sequence of the h-adaptive refinement process guided by the error in the QoI K_I.

recovery procedure (SPR-CD) using the analytical expressions of the dual loads, and the standard SPR. The smoother and more accurate behavior of the novel procedure,is clearly shown in both the effectivity of the error estimator and the indicator for the QoI.

5 Conclusions

In this work we have presented a methodology to obtain the analytical expressions for the loads of the dual problem, which are required by the equilibrated displacement recovery technique we are using to locally equilibrate the recovered dual stress field. The error estimation is performed by using a ZZ-type error estimator, thus, the quality of the recovered solutions is critical. Numerical results have shown the importance of equilibrating the recovered solutions for the primal and dual problem in order to provide the sharp error estimates presented.
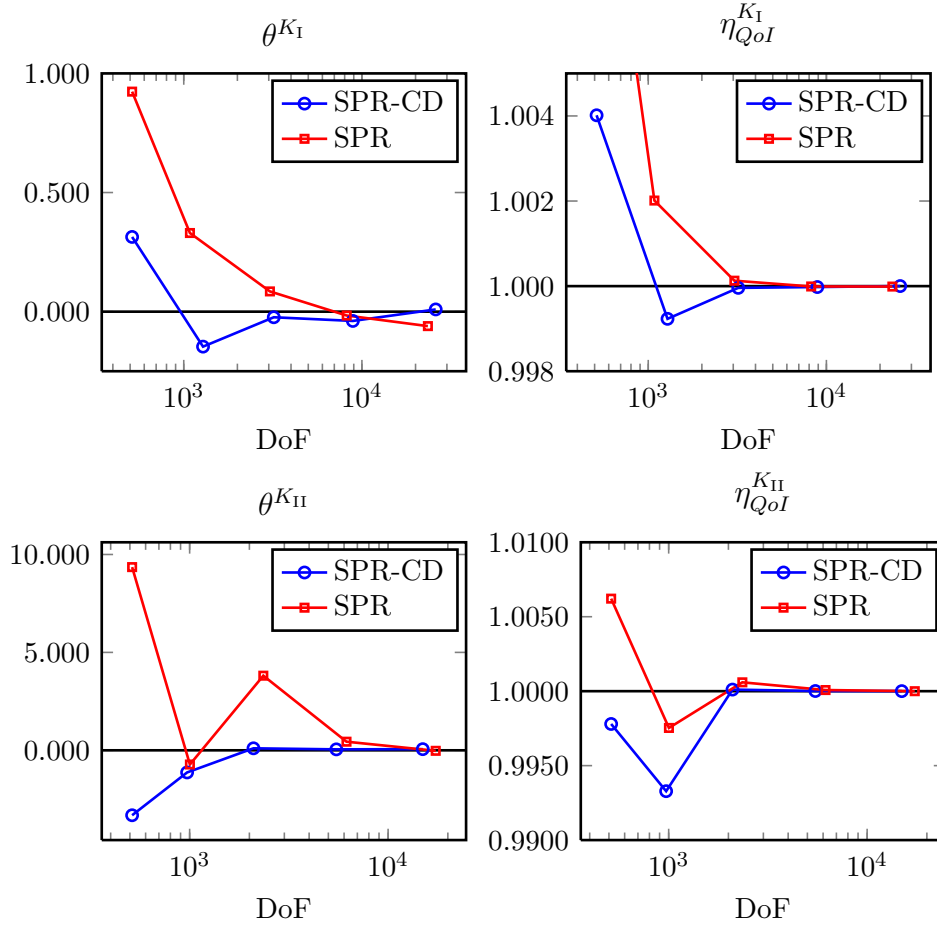
Figure 2: Problem 1. K_{I} and K_{II}. Evolution of the global effectivites

ACKNOWLEDGEMENTS

REFERENCES

[1] Babuška I. Error-Bounds for Finite Element Method. *Numerische Mathematik* 1970; **16**:322–333.

[2] Babuška I, Rheinboldt WC. A-posteriori error estimates for the finite element method. *International Journal for Numerical Methods in Engineering* 1978;

12(10):1597–1615.

[3] Ladevèze P, Leguillon D. Error estimate procedure in the finite element method and applications. *SIAM Journal on Numerical Analysis* 1983; **20**(3):485–509.

[4] Díez P, Parés N, Huerta A. Recovering lower bounds of the error by postprocessing implicit residual a posteriori error estimates. *International Journal for Numerical Methods in Engineering* Mar 2003; **56**(10):1465–1488.

[5] Pereira OJBA, de Almeida JPM, Maunder EAW. Adaptive methods for hybrid equilibrium finite element models. *Computer Methods in Applied Mechanics and Engineering* 1999; **176**(1-4):19–39.

[6] Almeida OJB, Moitinho JP. A posteriori error estimation for equilibrium finite elements in elastostatic problems. *Computer Assisted Mechanics and Engineering Sciences* 2001; **8**(2-3):439–453.

[7] Moitinho JP, Maunder EAW. Recovery of equilibrium on star patches using a partition of unity technique. *International journal for . . .* 2009; **79**:1493–1516.

[8] Ainsworth M, Oden JT. *A posteriori Error Estimation in Finite Element Analysis*. John Wiley & Sons: Chichester, 2000.

[9] Paraschivoiu M, Peraire J, Patera AT. A posteriori finite element bounds for linear-functional outputs of elliptic partial differential equations. *Computer Methods in Applied Mechanics and Engineering* 1997; **150**(1-4):289–312.

[10] Ladevèze P, Rougeot P, Blanchard P, Moreau JP. Local error estimators for finite element linear analysis. *Computer Methods in Applied Mechanics and Engineering* 1999; **176**(1-4):231–246.

[11] Ródenas JJ, González-Estrada OA, Tarancón JE, Fuenmayor FJ. A recovery-type error estimator for the extended finite element method based on singular+smooth stress field splitting. *International Journal for Numerical Methods in Engineering* 2008; **76**(4):545–571.

[12] Díez P, Ródenas JJ, Zienkiewicz OC. Equilibrated patch recovery error estimates: simple and accurate upper bounds of the error. *International Journal for Numerical Methods in Engineering* 2007; **69**(10):2075–2098.

[13] Ródenas JJ, González-Estrada OA, Díez P, Fuenmayor FJ. Accurate recovery-based upper error bounds for the extended finite element framework. *Computer Methods in Applied Mechanics and Engineering* 2010; **199**(37-40):2607–2621.

[14] Verfürth R. A review of a posteriori error estimation techniques for elasticity problems. *Computational Methods in Applied Mechanics and Engineering* 1999; **176**:419–440.

[15] Oden JT, Prudhomme S. Goal-oriented error estimation and adaptivity for the finite element method. *Computers & Mathematics with Applications* 2001; **41**(5-6):735–756.

[16] Zienkiewicz OC, Zhu JZ. A simple error estimator and adaptive procedure for practical engineering analysis. *International Journal for Numerical Methods in Engineering* 1987; **24**(2):337–357.

[17] Ródenas JJ, Tur M, Fuenmayor FJ, Vercher A. Improvement of the superconvergent patch recovery technique by the use of constraint equations: the SPR-C technique. *International Journal for Numerical Methods in Engineering* 2007; **70**(6):705–727.

[18] Zienkiewicz OC, Zhu JZ. The superconvergent patch recovery and a posteriori error estimates. Part 1: The recovery technique. *International Journal for Numerical Methods in Engineering* 1992; **33**(7):1331–1364.

[19] Nadal E, González-Estrada OA, Ródenas JJ, Bordas SPA, Fuenmayor FJ. Error estimation and error bounding in energy norm based on a displacement recovery technique. *6th European Congress on Computational Methods in Applied Sciences and Engineering (ECCOMAS 2012)*, Eccomas, 2012.

[20] Ródenas JJ. Goal Oriented Adaptivity: Una introducción a través del problema elástico lineal. *Technical Report*, CIMNE, PI274, Barcelona, Spain 2005.

[21] González-Estrada OA, Ródenas JJ, Nadal E, Bordas SPA, Kerfriden P. Equilibrated patch recovery for accurate evaluation of upper error bounds in quantities of interest. *Adaptive Modeling and Simulation. Proceedings of V ADMOS 2011*, Aubry D, Díez P, Tie B, Parés N (eds.), CINME: Paris, 2011.

[22] Verdugo F, Díez P, Casadei F. Natural quantities of interest in linear elastodynamics for goal oriented error estimation and adaptivity. *Adaptive Modeling and Simulation. Proceedings of V ADMOS 2011*, Aubry D, Díez P, Tie B, Parés N (eds.), CIMNE: Paris, 2011.

[23] Szabó BA, Babuška I. *Finite Element Analysis*. John Wiley & Sons: New York, 1991.

[24] Ródenas JJ, Giner E, Tarancón JE, González-Estrada OA. A recovery error estimator for singular problems using singular+smooth field splitting. *Fifth International Conference on Engineering Computational Technology*, Topping BHV, Montero G, Montenegro R (eds.), Civil-Comp Press: Stirling, Scotland, 2006.

SHAPE SENSITIVITY ANALYSIS INCLUDING QUALITY CONTROL WITH CARTESIAN FINITE ELEMENT MESHES

J.J. RÓDENAS, J.E. TARANCÓN, O. MARCO AND E. NADAL

Centro de Investigación de Tecnología de Vehículos (CITV)
Universitat Politècnica de València ,Camino de Vera, s/n, E-46022 Valencia, Spain
e-mail: {jjrodena,jetaranc}@mcm.upv.es {onmaral,ennaso}@upvnet.upv.es

Key words: Cartesian Grid-FEM, NURBS, velocity field, shape sensitivity analysis

Abstract. The gradient-based optimization methods used for optimization of structural components require that the information of the gradients (sensitivity) of the magnitudes of interest with respect to the design variables is calculated with sufficient accuracy. The aim of this paper is to present a module for calculation of shape sensitivities with geometric representation by NURBS (Non-Uniform Rational B-Splines) for a program created to analyze 2-D linear elasticity problems, solved by FEM using cartesian grids independent of the geometry, CG-FEM.

First, it has been implemented the ability to define the geometry using NURBS, which have become in recent years in the most used geometric technology in the field of engineering design. In order to be able to represent exact geometries, a scheme based on matrix representation of this type of curve and proper integration is proposed. Moreover, the procedures for shape sensitivities calculation, for standard FEM, have been adapted to an environment based on cartesian meshes independent of geometry, which implies, for instance, a special treatment of the elements trimmed by the boundary and the implementation of new efficient methods of velocity field generation, which is a crucial step in this kind of analysis.

Secondly, an error estimator, as an extension of the error estimator in energy norm developed by Zienkiewicz and Zhu, has been proposed for its application to the estimation of the discretization error arising from shape sensitivity analysis in the context of cartesian grids.

The results will show how using NURBS curves involves significant decrease of geometrical error during FE calculation, and that the calculation module implemented is able to efficiently provide accurate results in sensitivity analysis thanks to the use of the CG-FEM technology.

1 INTRODUCTION

This paper presents an approach for calculation of shape sensitivities based on the use of cartesian meshes independent of the geometry. Gradient based optimization processes require this kind of information, and its accuracy influences the evolution of the process. Specifically, this paper focuses on 2-D optimization problems, with exact representation of the geometries, governed by the linear elasticity equations, using FEM to determine the sensitivity of the quantities of interest.
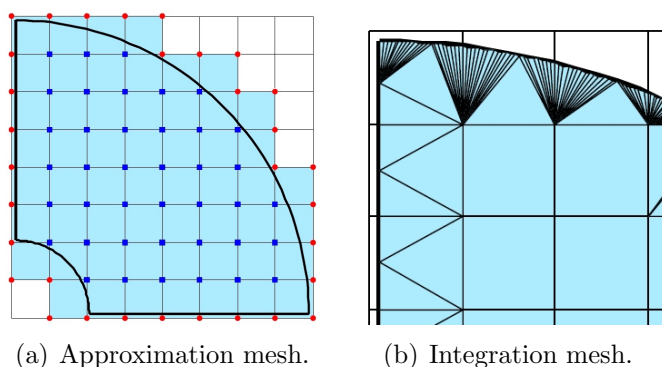

(a) Approximation mesh.   (b) Integration mesh.

Figure 1: Example of meshes related to CG-FEM.

Meshes independent of the geometry are used as a tool to alleviate the meshing and remeshing burden, for example, in the Generalized FEM (GFEM)[1] or Extended FEM (XFEM)[2]. The analysis proceducre makes use of two meshes, an approximation mesh, which is a mesh that covers the original domain and is used for the construction of the approximation basis, and an integration mesh intended for numerical evaluation of all integrals.

The approximation mesh needs to be a FE mesh satisfying only the requirement of covering completely the problem extension, as shown in Fig. 1(a), while the integration mesh is obtained by the division into subdomains of each element of the approximation mesh separately, taking into account the local geometry of the domain as shown in Fig. 1(b). For this subdivision the Delaunay triangulation is used, generating integration subdomains whose number will depend on the curvature of the edge crossing the elements.

In addition to this, the elements are disposed following a cartesian grid pattern in order to achieve significant computational savings, absolutely necessary in optimization analysis, where iterative analyses leading to considerable data flows are required.

Nowadays engineering analysis and high-performance computing are also demanding greater precision and tighter integration of the overall modeling-analysis process. In this regard, we will reduce errors by focusing on one, and only one, geometric model, which can be utilized directly as an analysis model. There are a number of candidate computational geometry technologies that may be used. The most widely used in engineering design are NURBS (Non-Uniform Rational B-splines), the industry standard. NURBS are convenient for free-form surface modeling, can exactly represent conic sections such as circles, cylinders, spheres, etc., and there exist many efficient and numerically stable algorithms to generate them. In order to achieve an accurate geometrical representation we present in this paper a combination of exact geometrical modeling, using NURBS

technology, and proper numerical integration while maintaining standard polynomial FE interpolation (isoparametric formulation).

Some of the most popular optimization methods are the gradient-based methods, based on the calculation of derivatives (sensitivities). To evaluate these gradients, a sensitivity analysis with respect to design variables is necessary. The design variables are defined by the analyst and describe the geometry of the component to be optimized. As a prelude to the calculation of sensitivities, it is necessary to define how to vary the position of material points of the domain in relation to the design variables, i.e. the sensitivity of the coordinates of the particles. This sensitivity can be interpreted as a velocity field, and its quality will affect the accuracy of the results. In this work we adapt the calculation of sensitivities to an CG-FEM environment, dribbling the problems arising from the use of meshes independent of the geometry and taking advantage of the cartesian grid structure. To evaluate the quality of the calculations we will implement an error estimator based on SPR techniques.

2 EXACT GEOMETRICAL REPRESENTATION

This section gives a brief introduction to NURBS[3][4]. In addition, it explains practical features when operating with cartesian meshes. A NURBS curve is defined as follows

$$P(t) = \frac{\sum_{i=0}^{n} N_{i,p}(t) w_i B_i}{\sum_{i=0}^{n} N_{i,p}(t) w_i} \qquad 0 \le t \le 1 \tag{1}$$

where $B_i = (x_i, y_i)$ represents the coordinate positions of a set of $i = 0, \ldots, n$ control points, w_i is the corresponding weight, and $N_{i,p}$ is the degree p B-spline basis function defined on the knot vector

$$K_t = \{t_0, t_1, \ldots, t_{n+p+1}\} \tag{2}$$

The i-th $(i = 0, \ldots, n)$ B-spline basis function can be defined recursively as

$$N_{i,p}(t) = \frac{(t - t_i) N_{i,p-1}(t)}{t_{i+p} - t_i} + \frac{(t_{i+p+1} - t) N_{i+1,p-1}(t)}{t_{i+p+1} - t_{i+1}} \qquad N_{i,0}(t) = \begin{cases} 1 & t_i \le t \le t_{i+1} \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

The polynomial space spanned by the B-spline basis can be converted into the piecewise polynomial representation spanned by the power basis so that the matrix representation for B-spline curves is always possible. There are some situations where it may be advantageous to generate the coefficients of each of the polynomial pieces, e.g., when we have to evaluate the curve at a large number of points or when we have to intersect the geometry with the mesh, as we will see later. Explicit matrix forms we have used would make it easier and faster because polynomial evaluation is more efficient in a power basis. In this work we have used a recursive procedure to get these matrices[5], then if we can represent

each section of the NURBS by standard parameter $u = \frac{t-t_i}{t_{i+1}-t_i}$ being $t \in [t_i, t_{i+1})$ the range that defines each one of them, we can write the matrix representation of a NURBS as

$$P(j, u) = \frac{U(u)M(j)w(j)B(j)}{U(u)M(j)w(j)} \tag{4}$$

where $U = \{1 \ u \ u^2 \dots \ u^p\}$, $M(j)$ the coefficient matrix corresponding to the span j, $B(j)$ the coordinates of the control points that influence the span j and $h(j)$ the weights for these control points.

One of the major tasks observed by using cartesian meshes independent of the geometry is the evaluation of its intersection with the geometric entities. Although the mesh is formed only by straight lines, the task of processing the intersections is a great computational effort. Although NURBS are rational curves, intersecting with straight lines implies that we can transform the NURBS rational expression in a non-rational polynomial expression, and therefore any algorithm to find polynomial roots will be valid.

As seen in Fig. 1(b) the integration mesh is composed by the internal elements and the subdomains created using a Delaunay triangulation depending on the curvature of the boundary. This triangulation attempts to capture the curvature of the geometry in order to solve the numerical integrals. However a linear triangulation, as in Fig. 1(b), will not suffice if we want to take advantage of exact geometries, so we use a coordinate transformation that allows us to accurately represent the problem domain. To achieve this the *transfinite interpolation*[6], commonly used in p-adaptivity, is ideal because it performs a mapping using area coordinates of triangles to locate the integration points considering the exact geometry, this computational effort will be located only in the boundary elements which reduces the number of triangular mappings to be done.

3 SHAPE SENSITIVITY ANALYSIS

In this paper we solve 2-D elasticity problems where discrete equilibrium equation is

$$\mathbf{K}\mathbf{u} = \mathbf{f} \tag{5}$$

and its derivative with respect to any design variable a_m is given by

$$\frac{\partial \mathbf{K}}{\partial a_m}\mathbf{u} + \mathbf{K}\frac{\partial \mathbf{u}}{\partial a_m} = \frac{\partial \mathbf{f}}{\partial a_m} \qquad \text{and rearranging} \qquad \mathbf{K}\frac{\partial \mathbf{u}}{\partial a_m} = \frac{\partial \mathbf{f}}{\partial a_m} - \frac{\partial \mathbf{K}}{\partial a_m}\mathbf{u} \tag{6}$$

where $\mathbf{K}$ and $\frac{\partial \mathbf{K}}{\partial a_m}$ are the global stiffness and stiffness sensitivity matrices and $\frac{\partial \mathbf{f}}{\partial a_m}$ are the equivalent forces sensitivities, considered null value because it is assumed that the applied forces will not change with the introduction of a disturbance of differential order.

Let us consider the formulation of isoparametric 2-D elements. The stiffness matrix of the element is given by

$$\mathbf{k}^e = \int_{\Omega^e} \mathbf{B}^T \mathbf{D} \mathbf{B} |\mathbf{J}| \mathrm{d}\Omega \tag{7}$$

where Ω^e is the domain in local element coordinates, $\mathbf{B}$ is the nodal strains-displacements matrix, $\mathbf{D}$ is the stiffness matrix that relates stresses with strains and $|\mathbf{J}|$ is the determinant of the Jacobian matrix.

Considering the derivative of $\mathbf{D}$ with respect to design variables is zero we will have

$$\frac{\partial \mathbf{k}^e}{\partial a_m} = \int_{\Omega^e} \left[\frac{\partial \mathbf{B}^T}{\partial a_m} \mathbf{D} \mathbf{B} + \mathbf{B}^T \mathbf{D} \frac{\partial \mathbf{B}}{\partial a_m} \right] |\mathbf{J}| \mathrm{d}\Omega + \int_{\Omega^e} \left[\mathbf{B}^T \mathbf{D} \mathbf{B} \frac{\partial |\mathbf{J}|}{\partial a_m} \right] \mathrm{d}\Omega \qquad (8)$$

where all members can be evaluated numerically, although the calculation of $\frac{\partial \mathbf{B}}{\partial a_m}$ and $\frac{\partial |\mathbf{J}|}{\partial a_m}$ will require the sensitivity of the nodal coordinates, known as velocity field, defined as

$$\frac{\partial}{\partial a_m} \{x_i, y_i\} = \mathbf{V}_m (x_i, y_i) \qquad (9)$$

The velocity field along the contour can be easily evaluated from the parametric description of the boundary. Theoretically, the velocity field is subject to only two requirements: regularity and linear dependency[7] with respect to the design variables.

To ensure the quality of the velocity field is even more important when using cartesian meshes independent of the geometry, since the integration mesh has elements intersected with the boundary, leading to internal and external nodes (blue squares and red circles respectively in Fig. 1(a)) needed to be assigned an appropriate velocity field satisfying the boundary conditions imposed.

From now on we will use the problem of a 1/4 of cylinder under internal pressure (Figs. 2(a) and 2(b)), to illustrate the velocity field generation process. In the sensitivity analysis of this example is considered only one design variable corresponding to the outer radious of the cylinder, thus taking $a_m = b$. The strategy will be to impose a velocity field on the boundary evaluated (Fig. 2(c)), for instance, using a finite difference scheme, and then to perform an interpolation to the rest of the internal domain and an extrapolation to the external nodes; so we present the next alternatives:
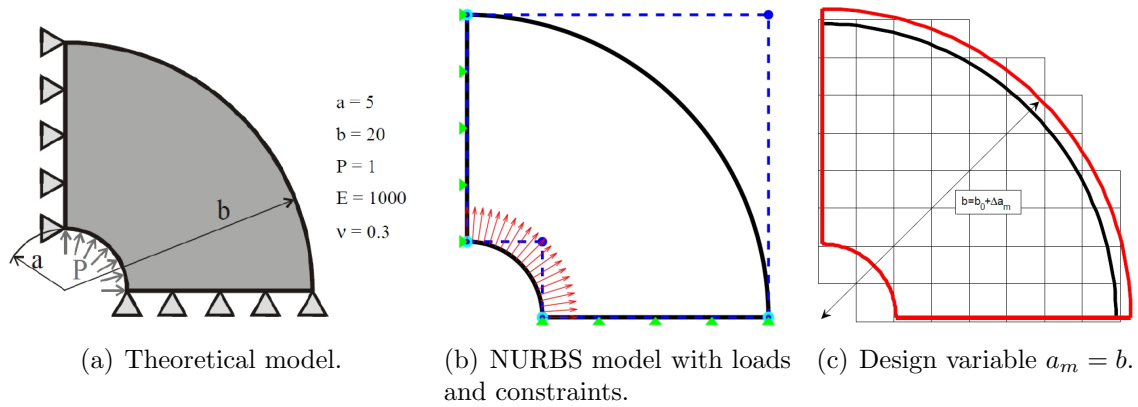

(a) Theoretical model.   (b) NURBS model with loads and constraints.   (c) Design variable $a_m = b$.

Figure 2: Cylinder under internal pressure. Problem definition.

Complete domain velocity field. This option implies to interpolate by picking points over the boundary and weighting the inverse of their distance with respect every internal node. For the external nodes, knowing the information in the boundary and in the internal domain it is possible to fit local polynomial surfaces on patches of elements around these nodes in order to get a proper extrapolation of the information. An output for the example is Fig. 3(a).

Contour adjacent elements velocity field. Knowing that any velocity field, satisfying some conditions, would be suitable for calculation of sensitivities, at least theoretically, we could obviate the internal domain and follow the previous strategy but only in the elements intersected with the boundary. This would lead to important computational savings without sacrificing much accuracy. See Fig. 3(b).

Physical approach. We can set up a load case where the Dirichlet boundary conditions are a velocity field[8], then, solving a FEM problem, we could obtain a solution that represents a velocity field on all active nodes. This solution will be very smooth but with the computational cost related to solving a system of equation. See Fig. 3(c). Note that iterative solvers can be used to solve the system of equations. If the intermediate solutions satisfy Dirichlet boundary conditions then they can be used as velocity fields as they meet the theoretical requirements thus reducing the computational cost to obtain a valid velocity field.
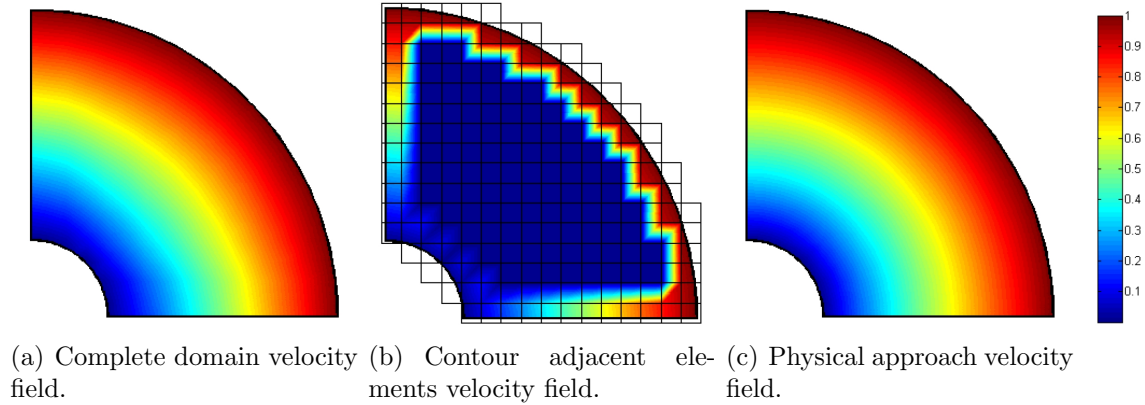

(a) Complete domain velocity field. (b) Contour adjacent elements velocity field. (c) Physical approach velocity field.

Figure 3: Comparison between different velocity fields for the cylinder example.

4 ERROR ESTIMATION BASED ON RECOVERY TECHNIQUES

In general it is not possible to know the exact error of the solution associated to the FE discretization in the problem analyzed, since this requires knowing the exact solution of the problem. Currently different methods have been developed to estimate the discretization error of the FE solution.

For linear elasticity problems the energy norm $\|\mathbf{u}\|$ is commonly used for quantifying the discretization error of the solution obtained from the FE analysis. In order to evaluate an estimate, $\|\mathbf{e}_{es}\|$, of the exact value of the discretization error in energy norm, $\|\mathbf{e}_{ex}\|$, Zienkiewicz and Zhu[9] developed the *ZZ estimator*, which is currently in wide use, proposing the use of the following expression:

$$\|\mathbf{e}_{es}\|^2 = \int_{\Omega} \left(\boldsymbol{\sigma}^* - \boldsymbol{\sigma}^h\right)^T \mathbf{D}^{-1} \left(\boldsymbol{\sigma}^* - \boldsymbol{\sigma}^h\right) \mathrm{d}\Omega \tag{10}$$

where domain Ω can refer to either the whole domain or a local (element) subdomain, $\boldsymbol{\sigma}^h$ represents the stresses evaluated using the FE method, $\boldsymbol{\sigma}^*$ is the so called smoothed or recovered stress field, that is a better approximation to the exact solution than the FE stresses $\boldsymbol{\sigma}^h$. In defining the $\boldsymbol{\sigma}^*$ field inside of each element the following expression is normally used:

$$\boldsymbol{\sigma}^* = \mathbf{N}\underline{\boldsymbol{\sigma}}^* \tag{11}$$

where $\mathbf{N}$ are the same shape functions used in the interpolation of the FE displacements field and $\underline{\boldsymbol{\sigma}}^*$. is the vector of smoothed nodal stresses in the element.

4.1 The SPR Technique and its Modifications

It becomes evident from (10) and (11) that the precision of the ZZ estimator will be a function of the smoothing technique used to obtain the nodal values $\underline{\boldsymbol{\sigma}}^*$. Because of its accuracy, robustness, simplicity and low computational cost, one of the most popular techniques used to evaluate $\underline{\boldsymbol{\sigma}}^*$ is the *Superconvergent Patch Recovery* (SPR) technique developed by Zienkiewicz and Zhu[10].

The components of $\underline{\boldsymbol{\sigma}}^*$ are obtained from a polynomial expansion, σ_p^*, defined over a set of contiguous elements called patch which consists of all of the elements sharing a vertex node, of the same complete order q as that of the shape functions $\mathbf{N}$. For each of the stress components of this polynomial, σ_p^* is found using the following expression:

$$\sigma_p^* = \mathbf{p}\mathbf{a} \tag{12}$$

where $\mathbf{p}$ contains the terms of the polynomial and $\mathbf{a}$ is the vector of polynomial unknown coefficients.

The data of the FE stresses calculated at the superconvergence points are used to evaluate $\mathbf{a}$ by means of a least-square fit. Once the $\mathbf{a}$ parameters have been calculated for each stress component, the values of $\underline{\boldsymbol{\sigma}}^*$ are obtained by substituting the nodal coordinates into the polynomial expressions σ_p^*.

After the publication of the SPR technique, a great number of articles followed which proposed modifications generally based on an approximate satisfaction of equilibrium equations to improve its performance. In this paper we will use an implementation of the SPR-C technique[11] which is a modification of the original SPR technique with the

–

objective of improving the values of $\underline{\boldsymbol{\sigma}}^*$. In this approach constraint equations are applied over the $\mathbf{a}$ coefficients that define the stress interpolation polynomials in the patch σ_p^*, so that these polynomials can satisfy both field equations (internal equilibrium equation and compatibility equation) and boundary conditions (boundary equilibrium equation, symmetry boundary condition, etc.) as far as the representation of $\boldsymbol{\sigma}^*$ by means of polynomials can allow. The constraint equations over the a coefficients are applied using the Lagrange's multipliers method.

4.2 Error Estimation in Sensitivities

As explained before, the sensitivity of the FE solution will not be exact, regardless of the method used for calculating sensitivities. So we will need to determine a magnitude to quantify the discretization error in sensitivities associated to the FE discretization.

Following [12], we will use the sensitivity of the square of the energy norm with respect to the design variable considered as magnitude for the quantification of the discretization error in sensitivities. This sensitivity is defined as the variation of the square of the energy norm with respect to the design variables, i.e.:

$$\chi_m = \frac{\partial \|\mathbf{u}\|^2}{\partial a_m} = \sum^n \int_{\Omega^e} \boldsymbol{\sigma}^T \mathbf{D}^{-1} \left(2 \left(\frac{\partial \boldsymbol{\sigma}}{\partial a_m} \right) + \frac{\boldsymbol{\sigma}}{|\mathbf{J}|} \frac{\partial |\mathbf{J}|}{\partial a_m} \right) |\mathbf{J}| \, \mathrm{d}\Omega^e \tag{13}$$

The discretization error in sensitivities can be evaluated deriving (10) with respect to the design variables, yielding

$$e\left(\chi_m\right)_{es} = \sum^n \int_{\Omega^e} \left(\left(\boldsymbol{\sigma}^* - \boldsymbol{\sigma}^h\right)^T \mathbf{D}^{-1} \left(2 \left[\left(\frac{\partial \boldsymbol{\sigma}}{\partial a_m} \right)^* - \frac{\partial \boldsymbol{\sigma}^h}{\partial a_m} \right] + \frac{\left(\boldsymbol{\sigma}^* - \boldsymbol{\sigma}^h\right)}{|\mathbf{J}|} \frac{\partial |\mathbf{J}|}{\partial a_m} \right) |\mathbf{J}| \right) \mathrm{d}\Omega^e \tag{14}$$

where, considering $\underline{\mathbf{u}}^e$ as displacements at nodes of each element e, $\frac{\partial \boldsymbol{\sigma}^h}{\partial a_m}$ is given by:

$$\frac{\partial \boldsymbol{\sigma}^h}{\partial a_m} = \mathbf{D}\mathbf{B}\frac{\partial \underline{\mathbf{u}}^e}{\partial a_m} + \mathbf{D}\frac{\partial \mathbf{D}}{\partial a_m}\underline{\mathbf{u}}^e \tag{15}$$

In addition it is important to point out that in order to smoothing the derivatives of the stress field with respect to the design variables $\left(\frac{\partial \boldsymbol{\sigma}}{\partial a_m} \right)^*$, we will apply recovery techniques similar to those used with the stress field $\boldsymbol{\sigma}^*$.

The computational cost will be reduced since it only requires the determination of $\left(\frac{\partial \boldsymbol{\sigma}}{\partial a_m} \right)^*$ and the direct application of the above equation. The remaining quantities involved in the above expression ($|\mathbf{J}|$, $\frac{\partial |\mathbf{J}|}{\partial a_m}$ and $\boldsymbol{\sigma}^*$) will be available through previous calculations of the analysis, the calculation of sensitivities and the estimation of the discretization error in energy norm. We have to say that the smoothing of the stress field will be carried out with the SPR-C technique while the field of stress derivatives will be recovered with the standard SPR technique.

5   NUMERICAL RESULTS

Before starting with numerical comparations we have to define some magnitudes to help us to evaluate and to understand the results obtained. First of all, we define the estimated relative error in sensitivities. This is more easily interpretable than the absolute error, then we will have

$$\eta\left(\chi_m\right)_{es} = \sqrt{\left|\frac{e\left(\chi_m\right)_{es}}{\chi_m + e\left(\chi_m\right)_{es}}\right|} \qquad (16)$$

To evaluate the effectivity of the error estimator in sensitivities we will use

$$\theta\left(\chi_m\right) = \sqrt{\left|\frac{e\left(\chi_m\right)_{es}}{e\left(\chi_m\right)_{ex}}\right|} \qquad (17)$$

In addition, we can find in [12] a relationship between the discretization error in energy norm and the discretization error in sensitivities. This relationship between the two types of errors can be used as an indicator quality of the results obtained from the velocity field generation process, being defined as

$$\frac{e(\chi_m)_{ex}}{\|e(\mathbf{u})_{ex}\|^2} \approx const. \qquad (18)$$

Now we will show the analyses performed to evaluate the proper behavior of the work developed and the accuracy of the results. For proper evaluation of the results we will evaluate the parameters defined in the previous paragraphs.

The exact solution for the problem defined in Fig.2 is: energy norm $\|\mathbf{u}_{ex}\|^2 = 2\Pi = 0.055815629478779$ and $\chi_{m_{ex}} = -5.082398781807488 \cdot 10^{-4}$.

For this problem we will compare the complete domain velocity field, the contour adjacent elements velocity field and a velocity field calculated by the physical approach of FE. Also we will use a field imposed analytically so that we can judge the goodness of the methods developed. This field is calculated directly at nodes following the expression $\mathbf{V}_m = (r \cdot A - B) \cdot \Delta a_m$ where the coefficients A and B will define a growing linear function from the axis of the cylinder depending on the nodal distance r.

The first analysis will be an h-adaptive refinement procedure with linear elements to observe the performance differences between inner and outer arcs created with NURBS or with standard splines defined with with different sets of interpolation points, 3 to 9.

In Figs. 4(a) and 4(c), we can evaluate the results in magnitudes related to the sensitivity analysis, where we can see that the convergence rate and the quality constant are better in the geometry defined by NURBS. But Fig. 4(b) is very important as it shows the error in energy norm and we see how for splines the convergence rate tends to zero as it approaches a value of error, related with the geometrical definition error, while for the geometry defined by NURBS the value of the error continues decreasing steadily.
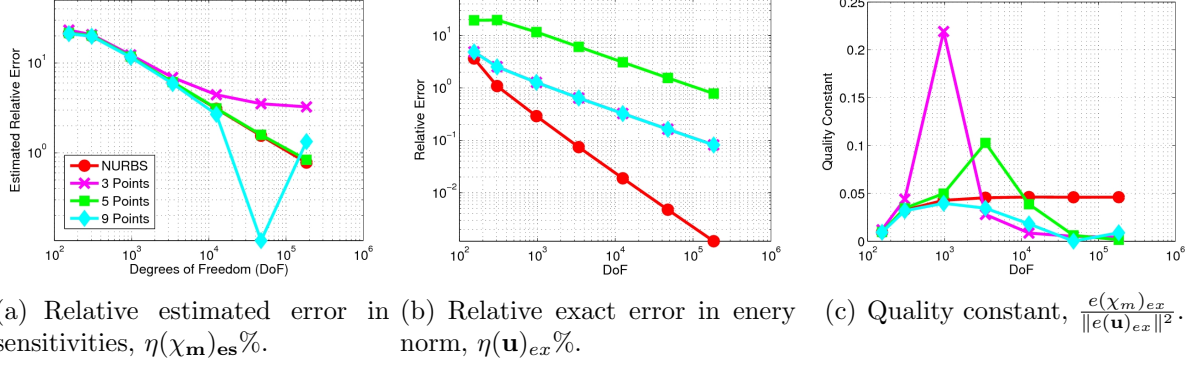
(a) Relative estimated error in sensitivities, $\eta(\chi_{\mathbf{m}})_{\mathbf{es}}\%$.

(b) Relative exact error in enery norm, $\eta(\mathbf{u})_{ex}\%$.

(c) Quality constant, $\frac{e(\chi_m)_{ex}}{\|e(\mathbf{u})_{ex}\|^2}$.

Figure 4: Effect of using NURBS. h-adaptive refinement and analytical velocity field.

After seeing these results, the remaining analyses in this section will use NURBS geometry. The simulations will be two analyses with linear elements and mesh refinement, one with uniform refinement and the other with h-adapted meshes. Both will compare the analytical velocity field and the one obtained by the physical approach of FE, with the complete domain velocity field and the contour adjacent elements velocity field.

Regarding uniform refinement, Fig. 5(a) shows similar convergence rate for the latest meshes, but when comparing the quality constant and the effectivity, in Figs. 5(b) and 5(c), only the field generated by the physical approach is able to match convergence rate and to maintain quality constant stable with respect the analytically imposed.
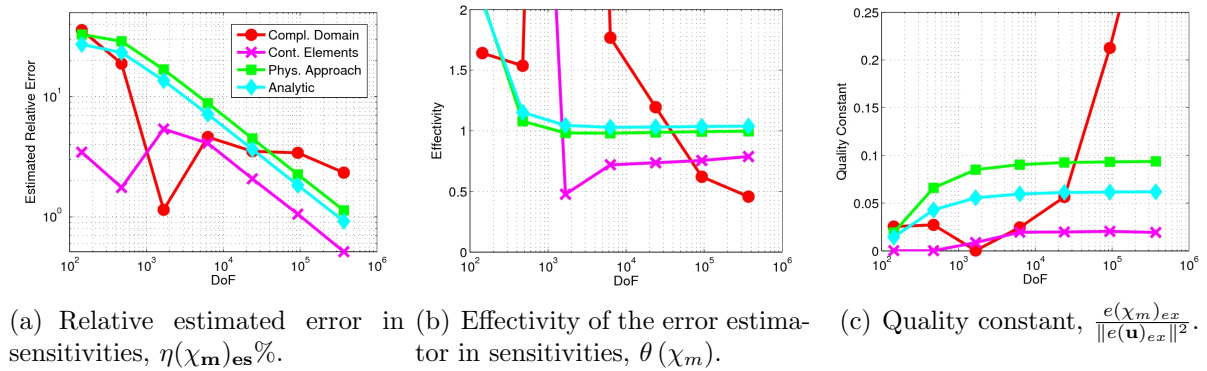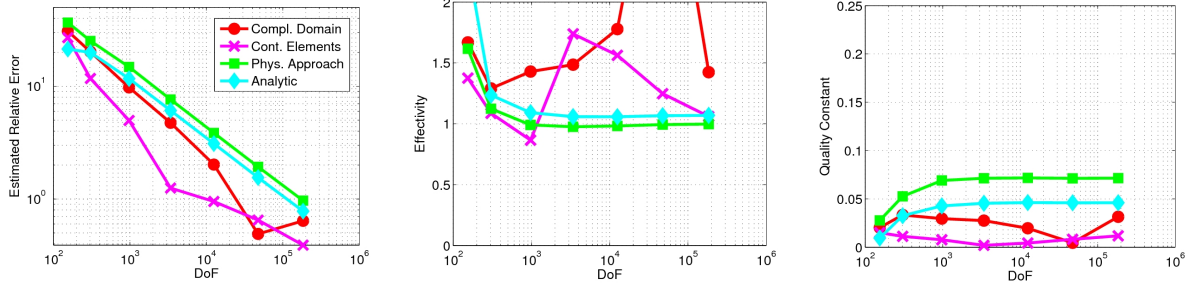

(a) Relative estimated error in sensitivities, $\eta(\chi_{\mathbf{m}})_{\mathbf{es}}\%$.

(b) Effectivity of the error estimator in sensitivities, $\theta(\chi_m)$.

(c) Quality constant, $\frac{e(\chi_m)_{ex}}{\|e(\mathbf{u})_{ex}\|^2}$.

Figure 5: Effect of velocity field. Uniform refinement. Geometry defined by NURBS.

If we use h-adaptive meshes then we will get similar results. The value of the convergence rate is good overall, (Fig. 6(a)). The effectivity index of the physical approach is the only capable of keeping up with the analytical velocity field, Fig. 6(b), but the quality constant of the velocity fields is acceptable for all of them, Fig. 6(c).

(a) Relative estimated error in sensitivities, $\eta(\chi_{\mathbf{m}})_{\mathbf{es}}\%$.

(b) Effectivity of the error estimator in sensitivities, $\theta\left(\chi_m\right)$.

(c) Quality constant, $\frac{e(\chi_m)_{ex}}{\|e(\mathbf{u})_{ex}\|^2}$.

Figure 6: Effect of velocity field. *h*-adaptive refinement. Geometry defined by NURBS.

6 CONCLUSIONS

This paper presents a module for the calculation of shape sensitivities with geometric representation by NURBS for a program created to analyze linear elastic problems, solved by FEM using 2-D cartesian meshes independent of geometry. Evaluating the results obtained, the theoretical bases available in the literature on the calculation of sensitivities have been adapted properly to an environment for which they were not developed in the beginning, thus overcoming the requirements arising from the use of cartesian meshes independent of the geometry and, on the other hand, various methods to generate robust and efficient velocity fields have been implemented as well. To evaluate the quality of the calculations we have implemented an error estimator based on SPR techniques. In general we can say that among the velocity field evaluation techniques analyzed, which provides better results in problems with a smooth solution is the physical approach of FE. The creation of the velocity field using this technique requires solving a problem with the same size of the original problem and that is usually done by a direct solver for small size problems but for large size problems the use of iterative procedures could be interesting.

REFERENCES

[1] T. Strouboulis, K. Copps and I. Babuška. *The Generalized Finite Element Method: an Example of its Implementation and Illustration of its Performance.* Int. Journal for Num. Methods in Eng., (2000) 47:1401-1417.

[2] N. Moës, J. Dolbow and T. Belytschko. *A Finite Element Method for Crack Growth without Remeshing.* Int. Journal for Num. Methods in Eng., (1999) 46:131-150.

[3] D.F. Rogers. *An Introduction to NURBS with Historical Perspective.* Morgan Kaufmann, (2001).

[4] L. Piegl and W. Tiller. *The NURBS Book.* Springer, (1997).

[5] K. Qin. *General Matrix Representations for B-splines.* The Visual Computer, (2000) 16:1-13.

[6] W.J. Gordon and C.A. Hal. *Construction of a Curvilinear Coordinate Systems and Applications to Mesh Generation.* Int. Journal for Num. Methods in Eng., (1973) 7:461-477.

[7] K.K. Choi and K.H. Chang. *A Study of Design Velocity Field Computation for Shape Optimal Design.* Finite Elements in Analysis and Design, (1994) 15:317-341.

[8] D. Belegundu, S. Zhang, Y. Manicka and R. Salagame. *The Natural Approach for Shape Optimization with Mesh Distortion Control.* Penn State University Report. College of Engineering. University Park, Pennsylvania, (1991).

[9] O.C. Zienkiewicz, J.Z. Zhu. *A Simple Error Estimation and Adaptive Procedure for Practical Engineering Analysis.* Int. Journal for Num. Methods in Eng., (1987) 24:337-357.

[10] O.C. Zienkiewicz, J.Z. Zhu. *The Superconvergent Patch Recovery and a-Posteriori Error Estimates. Part I: The Recovery Technique.* Int. Journal for Num. Methods in Eng., (1992) 33:1331-1364.

[11] J.J. Ródenas, M. Tur, F.J. Fuenmayor and A. Vercher. *Improvement of the Superconvergent Patch Recovery Technique by the Use of Constraint Equations: The SPR-C Technique.* Int. Journal for Num. Methods in Eng., (2007) 70:705-727.

[12] F.J. Fuenmayor, J.L. Oliver and J.J. Ródenas. *Extension of the Zienkiewicz-Zhu Error Estimator to Shape Sensitivity Analysis.* Int. Journal for Num. Methods in Eng., (1997) 40:1413-1433.

[13] J.J. Ródenas, F.J. Fuenmayor and J.E. Tarancón. *A Numerical Methodology to Assess the Quality of the Design Velocity Field Computation Methods in Shape Sensitivity Analysis.* Int. Journal for Num. Methods in Eng., (2003) 59:1725-1747.

PROBABILITY AND VARIANCE-BASED STOCHASTIC DESIGN OPTIMIZATION OF A RADIAL COMPRESSOR CONCERNING FLUID-STRUCTURE INTERACTION

DIRK ROOS*, KEVIN CREMANNS & TIM JASPER

Institute of Modelling and High-Performance Computing
Niederrhein University of Applied Sciences
Reinarzstr. 49, D-47805 Krefeld, Germany
e-mail: dirk.roos[@]hs-niederrhein.de

Key words: robust design optimization, robustness evaluation, reliability analysis, fluid-structure interaction, surrogate models, adaptive design of experiment, importance sampling, directional sampling

Abstract. Since the engineering of turbo machines began the improvement of specific physical behaviour, especially the efficiency, has been one of the key issues. However, improvement of the efficiency of a turbo engine, is hard to archive using a conventional deterministic optimization, since the geometry is not perfect and many other parameters vary in the real approach.

In contrast, stochastic design optimization is a methodology that enables the solving of optimization problems which model the effects of uncertainty in manufacturing, design configuration and environment, in which robustness and reliability are explicit optimization goals. Therein, a coupling of stochastic and optimization problems implies high computational efforts, whereby the calculation of the stochastic constraints represents the main effort. In view of this fact, an industrially relevant algorithm should satisfy the conditions of precision, robustness and efficiency.

In this paper an efficient approach is presented to assist reducing the number of design evaluations necessary, in particular the number of nonlinear fluid-structure interaction analyses. In combination with a robust estimation of the safety level within the iteration and a final precise reliability analysis, the method presented is particularly suitable for solving reliability-based structural design optimization problems with ever-changing failure probabilities of the nominal designs.

The applicability for real case applications is demonstrated through the example of a radial compressor, with a very high degree of complexity and a large number of design parameters and random variables.
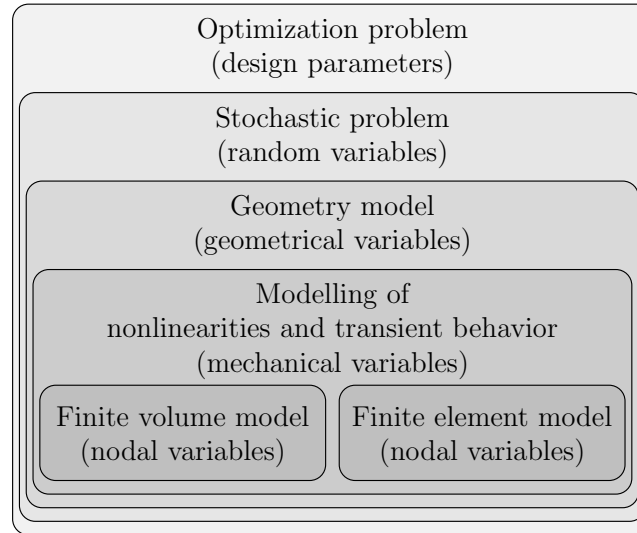
Figure 1: Coupled numerical models and different variable spaces of a stochastic design optimization of a fluid-structure interaction analysis based on a parametric geometry model (according Chateauneuf, 2008).

1 INTRODUCTION

1.1 Stochastic design optimization

In engineering problems, randomness and uncertainties are inherent and may be involved in several stages, for example in the system design with material parameters and in the manufacturing process and environment. Stochastic optimization, also referred to as reliability-based and variance-based optimization is known as the most adequate and advantageous methodology for system or process design and aims at searching for the best compromise design between design improvement and robustness or reliability assurance. Herein, the optimization process is carried out in the space of the design parameters and the robustness evaluation and reliability analysis are performed in the space of the random variables. Consequently, during the optimization process the design variables are repeatedly changed, whereby each design variable vector corresponds to a new random variable space. Therefore usually, a high number of numerical calculations are required to evaluate the stochastic constraints at every nominal design point. This repeated search becomes the main problem, especially when numerical nonlinear multi-domain simulations and CAD models are involved.

Unfortunately, in real case applications of the virtual prototyping process, it is not

always possible to reduce the complexity of the physical models to obtain numerical models which can be solved quickly. Although progress has been made in identifying numerical methods to solve stochastic design optimization problems and high performance computing, in cases such as those that have several nested numerical models, as shown in Fig. 1, the actual costs of using these methods to explore various model configurations for practical applications is too high. Therefore, methods for efficiently solving stochastic optimization problems based on the introduction of simplifications and special formulations for reducing the numerical efforts are required. Note: an extended version of this paper is published in Roos et al. (2013).

1.2 Application to aerodynamic optimization

In comparative studies on the application of the deterministic optimization for aerodynamic optimization (see e.g. Sasaki et al., 2001, Shahpar, 2000) usually stochastic programming algorithms or response surface methods (see e.g. Pierret and van den Braembussche, 1999) are used in turbomachinery design, for example in the development of engine components, such as at Vaidyanathan et al. (2000). In Shyy et al. (2001) a comprehensive overview is represented.

Another very comprehensive study of the use of the combination of genetic algorithms and neural networks for two-dimensional aerodynamic optimization of profiles is presented in Dennis et al. (1999) combine a genetic algorithm with an gradient-based optimization method.

Furthermore, an increasing application of stochastic analysis on turbo machinery (e.g. at Garzon, 2003, Garzon and Darmofal, 2003, Lange et al., 2010, Parchem and Meissner, 2009) underlines the importance of integrating the uncertainty analysis into the aerodynamic design process.

2 RELIABILITY AND VARIANCE-BASED DESIGN OPTIMIZATION

2.1 Deterministic optimization

Optimization is defined as a procedure to achieve the best outcome of a given objective function while satisfying certain restrictions. The deterministic optimization problem

$$
\begin{aligned}
f(d_1, d_2, \ldots d_{n_d}) &\rightarrow \min \\
e_l(d_1, d_2, \ldots d_{n_d}) &= 0; \; l = 1, n_e \\
u_m(d_1, d_2, \ldots d_{n_d}, \boldsymbol{\gamma}) &\geq 0; \; m = 1, n_u \\
d_{l_i} &\leq d_i \leq d_{u_i} \\
d_i &\in [d_{l_i}, d_{u_i}] \subset \mathbb{R}^{n_d}
\end{aligned}
\tag{1}
$$

Figure 2: Different solution points $\tilde{d}_i^I$ or $\tilde{d}_i$ as result of a deterministic vs. stochastic design optimization in the space of given randomly distributed design parameters.

Figure 3: Comparison of the deterministic optimal point $\tilde{d}_i^I$ and the solution of a stochastic optimization $\tilde{d}_i$ with corresponding most probable failure point x_j^* in the space of the randomly distributed von Mises stress and the yield stress.

is defined by the objective function $f : \mathbb{R}^{n_d} \to \mathbb{R}$ subject to the restrictions, defined as equality and inequality constraints e_l and u_m. The variables $d_1, d_2, \ldots d_{n_d}$ are the optimization or design variables and the vector of the partial safety factors $\boldsymbol{\gamma}$ ensures the system or design safety within the constraint equations u_m, for example defining a safety distance $u(\mathbf{d}, \gamma) = y_g/\gamma - y_d \geq 0$ between a defined limit state value y_g and the nominal design value y_d of a physical response parameter $y = f(\mathbf{d})$. In structural safety assessment, a typical constraint for the stress is given as

$$u(\mathbf{d}, \gamma) = \sigma_{y,k}/\gamma - \sigma_d \geq 0 \tag{2}$$

ensuring the global safety distance

$$\Delta_\gamma = \sigma_{y,k} - \sigma_d = \sigma_{y,k} - \frac{\sigma_{y,k}}{\gamma} = \sigma_{y,k} \left(1 - \frac{1}{\gamma}\right)$$

between the defined quantile value $\sigma_{y,k}$ of the yield stress and the nominal design stress σ_d with the global safety factor γ, as shown in Fig. 3. Whereby, in the real approach with given uncertainties, σ_d corresponds to the mean von Mises equivalent stress $\bar{\sigma}_e$ at the current design point.

2.2 Stochastic chance-constrained optimization

Stochastic optimization algorithms use the quantification of uncertainties to produce solutions that optimize the expected performance of a process or design, ensuring the target variances of the model responses and failure probability. So, the deterministic optimization problem (1) can be enhanced by additional stochastic restrictions. For example, the expression for system reliability

$$1 - \frac{P(\mathcal{F})}{P^t(\mathcal{F})} \geq 0 \tag{3}$$

ensures that the probability of failure

$$P(\mathcal{F}) = P[\{\mathbf{X} : g_k(\mathbf{x}) \leq 0\}] = \int \cdots \int_{g_k(\mathbf{x}) \leq 0}^{n_r} f_{\mathbf{X}}(\mathbf{x}) d\mathbf{x} \tag{4}$$

cannot exceed a given target probability $P^t(\mathcal{F})$, considering the vector of all random influences

$$\mathbf{X} = [X_1, X_2, ..., X_{n_r}]^T \tag{5}$$

with the joint probability density function of the random variables $f_{\mathbf{X}}(\mathbf{x})$ and $k = 1, 2, ..., n_g$ limit state functions $g_k(\mathbf{x}) \leq 0$.

These enhancements of the problem (1) are usually referred to reliability-based design optimization, in which we ensure that the design variables d_i satisfy the given constraints (3) to some specified probabilities. As a consequence, now the design parameters

$$\mathbf{d} = E[\mathbf{X}] \tag{6}$$

are the means of the n_r random influences $\mathbf{X}$ with every changing density function during the optimization process. As a result of the random influences, now the objective and the constraints are non-deterministic functions.

2.3 Reliability analysis using adaptive response surface method

For an efficient probability assessment of $P(\mathcal{F})$, according to Eq. (4), a multi-domain adaptive design of experiment in combination with directional sampling (see e.g. Ditlevsen et al., 1990) is introduced in Roos (2011) to improve the accuracy and predictability of surrogate models, commonly used in applications with several limit state conditions. Furthermore, the identification of the failure domains using the directional sampling procedure, the pre-estimation and the priori knowledge of the probability level is no longer required. Therefore this adaptive response surface

method is particularly suitable to solve reliability-based design optimization problems considering uncertainties with ever-changing failure probabilities of the nominal designs.

However, a reliability analysis method based on surrogate models, is generally suitable for a few random variables only. In case of the proposed probability assessment method, an efficient application is given up to $n_r = 10, ..., 25$, depending on the number of relevant unsafe domains. Therefore, a variance-based sensitivity analysis should be used to find a reduced space of the important random influences.

2.4 Global variance-based sensitivity analysis

In general, complex nested engineering models, as shown in Fig. 1 contain not only first order (decoupled) influences of the design parameters or random variables but also higher order (coupled) effects on the response parameter of a numerical model. A global variance-based sensitivity analysis, as introduced by Saltelli et al. (2008), can be used for ranking variables $X_1, X_2, \ldots, X_{n_r}$ with respect to their importance for a specified model response parameter

$$Y = f(X_1, X_2, \ldots, X_{n_r})$$

depending on a specific surrogate model $\tilde{Y}$. In order to quantify and optimize the prognosis quality of these meta models, in Most and Will (2008) and Most (2011) the so-called coefficient of prognosis

$$\text{COP} = \left(\frac{\mathbf{E}[Y_{\text{Test}} \cdot \tilde{Y}_{\text{Test}}]}{\sigma_{Y_{\text{Test}}} \sigma_{\tilde{Y}_{\text{Test}}}} \right)^2; \quad 0 \leq \text{COP} \leq 1 \tag{7}$$

of the meta model is introduced. In contrast to the commonly used generalized coefficient of determination R^2 based on a polynomial regression model, in Eq. (7) variations of different surrogate models $\tilde{Y}$ are analyzed to maximize the coefficient of prognosis themselves. This procedure results in the so-called meta model of optimal prognosis, used as surrogate model $\tilde{Y}$ with the corresponding input variable subspace which gives the best approximation quality for different numbers of samples, based on a multi-subset cross validation obtained by latin hypercube sampling (see e.g. Huntington and Lyrintzis, 1998).

The single variable coefficients of prognosis are calculated as follows

$$\text{COP}_i = \text{COP} \cdot \tilde{S}_{T_i} \tag{8}$$

with the total sensitivity indices

$$\tilde{S}_{T_i} = \frac{E(V(\tilde{Y}|\mathbf{X}_{\sim i}))}{V(\tilde{Y})} \tag{9}$$
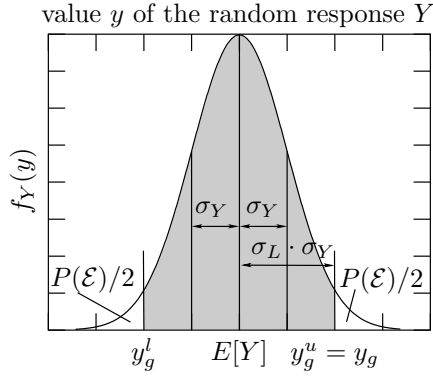
Figure 4: Relationship between density function $f_Y(y)$ of a model response, sigma level and exceedance probability, depending on chosen limit state conditions $y_g^{u,l}$.
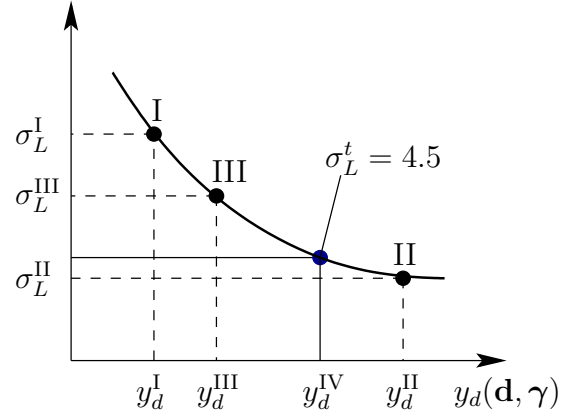

Figure 5: Convergence of a sequential stochastic chance-constrained optimization with successive interpolation of the nominal response limit y_d to ensure a target sigma level σ_L^t.

which have been introduced by Homma and Saltelli (1996), where $E(V(\tilde{Y}|\mathbf{X}_{\sim i}))$ is the remaining variance of $\tilde{Y}$ that would be left, on average, if the parameter of X_i is removed from the model. In Eq. (9) $\mathbf{X}_{\sim i}$ indicates the remaining set of input variables.

2.5 Probability estimation based on moments

For an accurate calculation of the reliability it would be interesting to expand the probability density function of the model responses about a critical threshold. Unfortunately, the density functions are unknown, especially close to the unsafe domain with high failure probability. Existing methods such as polynomial expansions, maximum entropy method or saddlepoint expansion, as reviewed in Hurtado (2008), are frequently used within the reliability-based structural optimization replacing the expensive reliability analysis.

A more simple, non-intrusive approach for a rough estimation of the failure probability is the calculation of the minimal sigma level σ_L for a performance-relevant random response parameter Y defined by an upper and lower limitstate value $y_g^{u,l} := \{Y|g(X) = 0\}$ as follows

$$E[Y] \pm \sigma_L \cdot \sigma_Y \overset{!}{\lessgtr} y_g^{u,l}$$

The sigma level can be used in conjunction with standard deviation to measure the deviation of response values Y from the mean $E[Y]$. For example, for a pair of
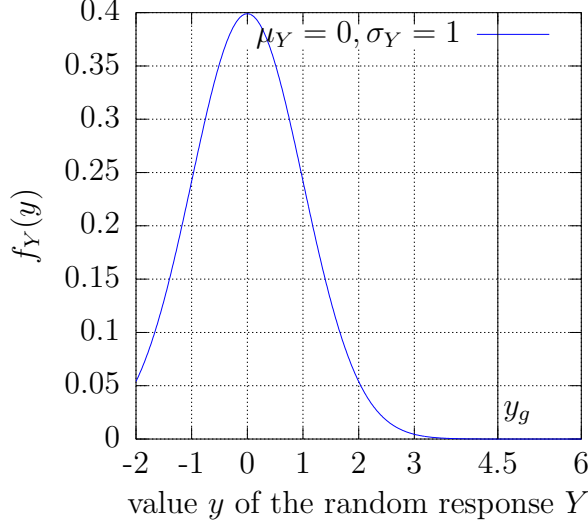
Figure 6: Gaussian density function $f_Y(y)$ of random response with upper specification limit $y_g := \{Y|g(X) = 0\}$.
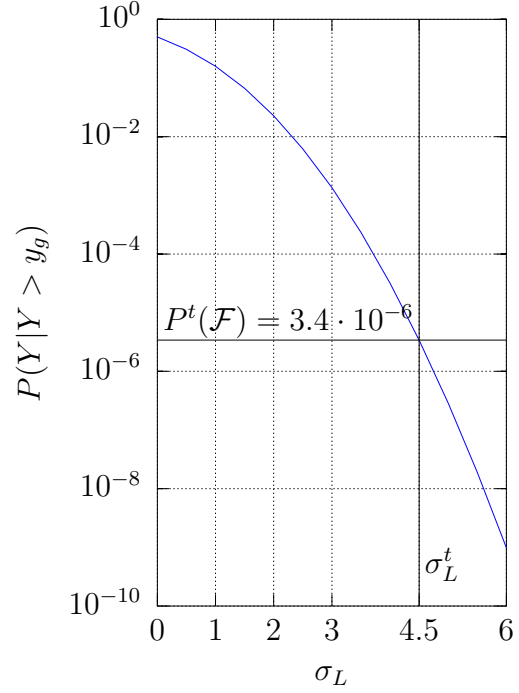
Figure 7: Sigma level variation and associated probability of failure (assumption: normal distribution for all important random responses).

quantiles (symmetrical case) and the mean value we obtain the assigned sigma level

$$\sigma_L = \frac{y_g - E[Y]}{\sigma_Y} \tag{10}$$

of the limit state violation, as explained in Fig. 4. Therewith, the non-exceedance probability results in

$$P(\mathcal{E}) = P(\{Y|Y \lesseqgtr y_g^{u,l}\}) = f(\sigma_L)$$

as a function of the sigma level, depending on the current distribution type of Y. In the same manner failure probability

$$P(\mathcal{F}) = P(\{Y|Y > y_g\}) = f(\sigma_L) \tag{11}$$

is given as a function of the sigma level. For example, assuming a normal distribution of the random response Y with $\mu_Y = 0$ and $\sigma_Y = 1$, as shown in Fig. 6, the failure probability is given as a nonlinear function

$$P(\mathcal{F}) = \Phi(-\sigma_L) = \Phi(-y_g)$$

of the sigma level, as illustrated in Fig. 7. Therewith, a probability of $P^t(\mathcal{F}) = 3.4 \cdot 10^{-6}$ is achieved when the performance target σ_L^t is 4.5 σ away from the mean value.

Other values of acceptable annual probabilities of failure $P^t(\mathcal{F})$ depending on the consequence of failure, significance warning or without warning before occurrence of failure and (non-)redundant structures can be found in engineering standards, e.g. in DNV (1992).

2.6 Methods solving stochastic optimization problems
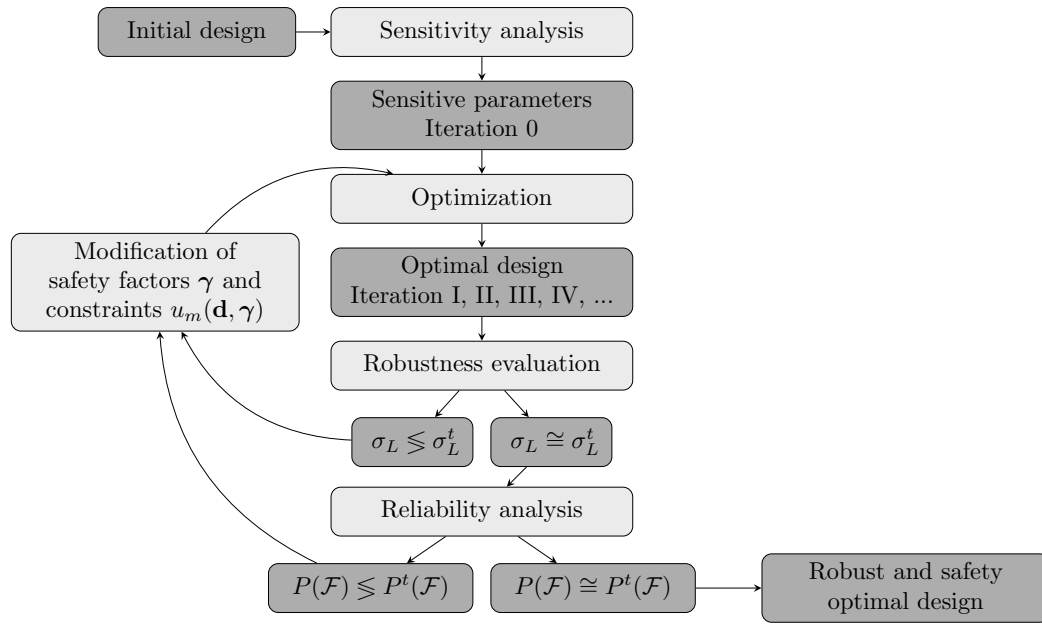
Figure 8: Basic concept of a decoupled loop of a reliability-based and variance-based stochastic design optimization using global variance-based sensitivity analysis and robustness evaluation to reduce the design parameter and random variable space.

In general, problem (1) to (6) is solved as a combination of a deterministic optimization in the n_d-dimensional design space and a stochastic analysis in the n_r-dimensional random space. Derivative-free global optimization methods are typically recommended to solve the sequential deterministic optimization problem, according to Eq. (1) for highly nonlinear numerical models, especially fluid-structure interaction models with probability-based constraints, whose objective and constraint function value may be computed with some noise or are non-computable in any design points.

Evolutionary computation, as a special class of global optimization strategies, imitates the natural processes like biological evolution or swarm intelligence. Based on the principle "survival of the fittest" a population of artificial individuals searches the design space of possible solutions in order to find a better solution for the optimization problem. In this paper an evolution strategy using a class of evolutionary algorithms is used. This strategy uses normally distributed mutations, recombination, selection of the best offspring individuals, and the principle of self-adaptation of strategy parameters, as described in Bäck (1995).

As an alternative derivative-free optimization method, especially useful for expensive numerical computations, we use the adaptive response surface methodology, as introduced in Etman et al. (1996), Toropov and Alvarez (1998), Abspoel et al. (1996), Stander and Craig (2002), Kurtaran et al. (2002).

Mainly, there are three methods for solving these kinds of coupled problems (1) to (6). The simplest and most direct solution method is a coupled approach in which a full reliability analysis is performed for every optimization function evaluation (see e.g. Choi et al., 2001). This involves a nesting of two distinct levels of optimization within each other, one at the design level and one at the reliability analysis level. This coupled procedure leads in general to an inefficient double loop with a large number of design evaluations.

The single-loop method (see e.g. Kharmanda et al., 2002) simultaneously minimizes the objective function and searches for the β-point, satisfying the probabilistic constraints only at the optimal solution, but needs a sensitivity analysis to analytically compute the design gradients of the probability constraint.

An alternative method, used in the following, is the sequential approach (see e.g. Chen et al., 2003). The general concept is to iterate between optimization and uncertainty quantification, updating the constraints based on the most recent probabilistic assessment results, using safety factors or other approximation methods. This effective iterative decoupled loop approach can be enhanced by updating the constraints during the internal optimization using sigma levels and statistical moments

$$\frac{\sigma_{L_k}}{\sigma_L^t} - 1 \geq 0; \ \sigma_{L_k} = \frac{y_{gk} - E[Y_k]}{\sigma_{Y_k}}; \ k = 1, n_g$$

in place of the exceedance probability of the Eq. (3). Essentially, by means of transformation in Eq. (11) of the probability-based highly nonlinear and non-differentiable constraints to linear ones, these functions may be more well conditioned for the optimization approach and we can expect a better performance of the solution process. Of course, the transformation in Eq. (11) can only be used as a rough estimation of the safety level and we have to calculate the probabilities of failure using the
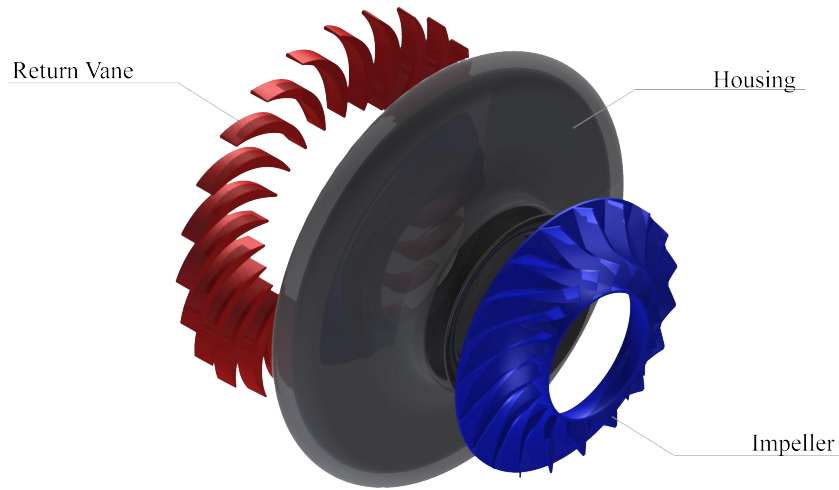
Figure 9: Parametric CAD model of a one stage radial compressor, consisting of a impeller and returnvane

reliability analysis, at least at the iteration end.

As shown in Fig. 8, in the initial iteration step a variance-based sensitive analysis identifies the most important multivariate dependencies and design parameters. After this, the deterministic optimization step results in an optimal solution for which the sigma level is calculated using a robustness evaluation, based on a latin hypercube sampling. The size of violation of the target sigma level is used to interpolate the constraints using modified safety factors. Whereby, as an important fact, the interpolation order increases continuously with each iteration step, so in practice three or four iteration steps may meet our optimization requirements in terms of robustness and safety. Fig. 5 shows a typical convergence of a sequential stochastic chance-constrained optimization.

Furthermore, the optimization steps and the final reliability analysis run mostly efficiently in the space of the current significant parameters. So every size of problem definition (number of design and random parameters) is solvable within all sigma levels.

The following numerical example with a very high degree of complexity is given to demonstrate the solving power of this sequential stochastic chance-constrained optimization by adapting the constraint $u_m(\mathbf{d}, \gamma)$ depending on interpolated nominal response values y_d.

3 NUMERICAL EXAMPLE

3.1 Fluid-structure interaction model

The stochastic optimization method presented here is applied to a CAD and CAE parameter-based design optimization of a radial compressor shown in Fig. 9, including material, process and geometry tolerances. In the example presented the target of the optimization process is to maximize the efficiency of the turbine engine with respect to a limitation of the maximal v. Mises stress. Additional constraints are defined by resonance of any eigen frequency with the rotational velocity of the rotor. In total 36 optimization parameters and 49 random influences are defined.

The Calculations were done with the software ANSYS Workbench and the probabilistic and optimization tasks were performed with the optiSLang software package.

As the method was already explained in Sec. 2, the results of the example are summarized. For a extended version see Roos et al. (2013).

3.2 Decoupled stochastic optimization loop

Through the sensitivity analysis the design parameters were reduced to 10 design variables with a relevant coefficient of optimal prognosis. The mean efficiency of the initial radial compressor was 86%. The best design of the latin hypercube sampling with an efficiency of 88.9% is used as start design of an evolutionary optimization based on the surrogate model of the meta model of optimal prognosis and gives with one additional design evaluation an efficiency of 89.3%. The distance of the design stress to the 5% quantile of the yield strength is a result of the first global safety factor of $\gamma^I = 1.5$ of the first iteration step. The target sigma level is $\sigma_L^t = 4.5$ to ensure a probability of failure $P(\mathcal{F}) = 3.14 \cdot 10^{-6}$. In the following, only the results of each iteration are shown in the Tab. 1.

Of course, the probability levels of violation of the limit state conditions or of the initial efficiency are only a rough estimation and at least a reliability analysis of the final design is recommended, especially for small probability levels. With the identification of the random sub domain directional sampling on adaptive moving least square is used for reliability analysis (see Sec. 2.3). The moving least square approximation is based on $N = 56$ design evaluations of an adaptive D-optimal design of experiment, as shown in Figs. 10 and 11. The assigned failure probability $\bar{P}(\mathcal{F}) = 2.5 \cdot 10^{-6} \leq P^t(\mathcal{F}) = 3.4 \cdot 10^{-6}$ indicates an optimized six sigma design.

Finally, the Figs. 12 and 13 show the flow along the return vane blades. It is distinctly and visibly how the separations have been reduced in the optimized design and a more uniform flow is present.

	Safety factor γ^i, sigma level σ_L^i, σ_d^i and efficiency η_i			
i	γ^i	σ_L^i	σ_d^i	η^i
0	2.4	-	$1.27 \cdot 10^8$	86%
1	1.5	5.13	$1.67 \cdot 10^8$	90.5%
2	1.32	3.6	$1.75 \cdot 10^8$	90.8%
3	1.426	4.1	$1.71 \cdot 10^8$	90.0%
4	1.46	4.48		91%

Table 1: Results for each iteration step i.
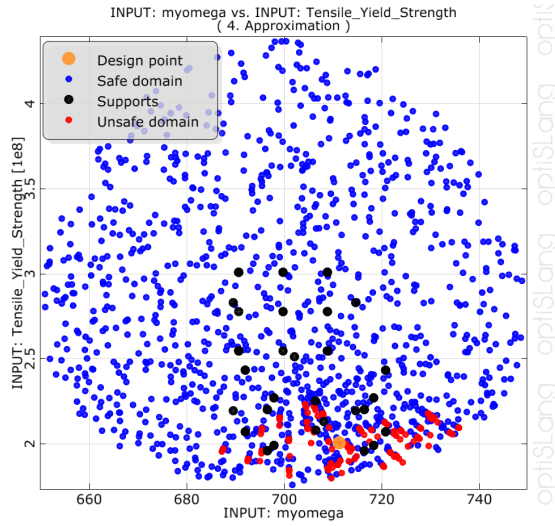
Figure 10: Anthill plot of the analyzed $N = 56$ design evaluations of the reliability analysis within iteration step IV between efficiency η and yield stress σ_y.
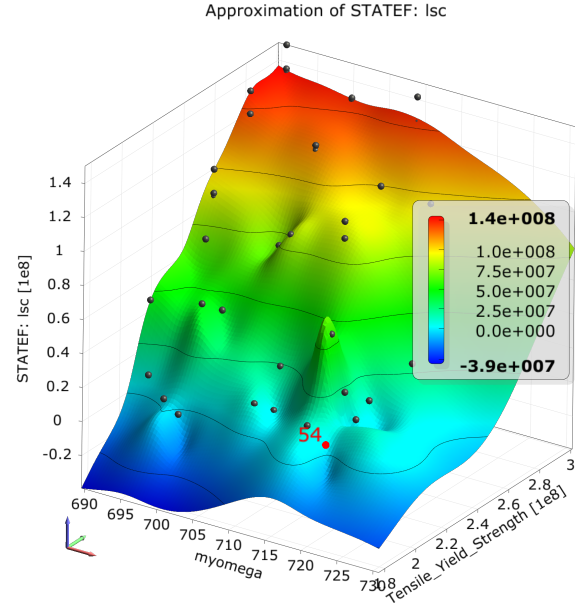
Figure 11: Response surface plot of the reliability analysis design IV.

4   CONCLUSIONS

In this paper an efficient iterative decoupled loop approach is provided for reducing the necessary number of design evaluations. The applicability of this method for
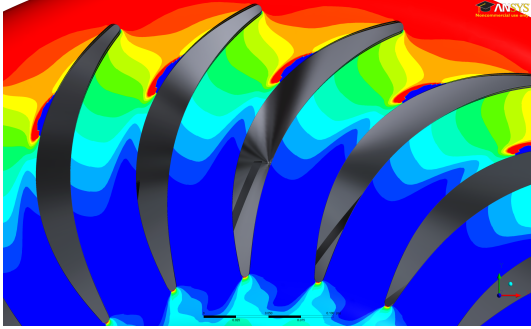
Figure 12: Flow angle of the initial design at the returnvane blades with separations along the blades

Figure 13: Flow angle of the optimized design at the returnvane blades with a much more uniform flow

real case applications is demonstrated for a radial compressor. Using the approach presented, it is possible to improve the efficiency by about 5%. In addition we obtain an optimized design which is insensitive to uncertainties and considers the target failure probability.

5 ACKNOWLEDGEMENT

References

S.J. Abspoel, L.F.P. Etman, J. Vervoort, R.A. van Rooij, A.J.G Schoofs, and J.E. Rooda. Simulation based optimization of stochastic systems with integer design variables by sequential multipoint linear approximation. *Structural and Multidisciplinary Optimization*, 22:125–138, 1996.

T. Bäck. Evolution strategies: an alternative evolutionary algorithm. In *Artificial Evolution*, pages 3–20. Springer-Verlag, 1995.

A. Chateauneuf. *Advances in solution methods for reliability-based design optimization*, volume 1 of *Structures and Infrastructures: Structural Design Optimization Considering Uncertainties*, chapter 9, pages 217 – 246. Taylor & Francis, London, UK, 2008.

W. Chen, H. Liu, J. Sheng, and H. C. Gea. Application of the sequential optimization and reliabilty assessment method to structural design problems. In *Proceedings of DETC'03, ASME 2003 Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Chicao, Illinois USA, September 2 – 6 2003.

K. K. Choi, J. Tu, and Y. H. Park. Extensions of design potential concept for reliability-based design optimization to nonsmooth and extreme cases. *Structural and Multidisciplinary Optimization*, 22:335–350, 2001.

B. H. Dennis, G. S. Dulikravich, and Z.-X. Han. Constrined shape of optimization of airfoil cascades using a navier-stokes solver and a genetic/sqp algorithm. In *ASME 99-GT-441*, 1999.

O. Ditlevsen, R. E. Melchers, and H. Gluver. General multi-dimensional probability integration by directional simulation. *Computers & Structures*, 36:355–368, 1990.

DNV. Structural reliability analysis of marine structure. Technical Report Classification Notes, No. 30.6, Det Norske Veritas Classification AS, Computer Typesetting by Division Ship and Offshore, Norway, 1992.

L.F.P. Etman, J.M.T.A. Adriaens, M.T.P. van Slagmaat, and A.J.G. Schoofs. Crashworthiness design optimization using multipoint sequential linear programming. *Structural Optimization*, 12:222–228, 1996.

Victor E. Garzon. *Probabilistic Aerothermal Design of Compressor Airfoils*. PhD thesis, Massachusetts Institute of Technology, 2003.

Victor E. Garzon and David L. Darmofal. Impact of geometric variability on axial compressor performance. *Journal of Turbomachinery*, 125:692–703, 2003.

T. Homma and A. Saltelli. Importance measures in global sensitivity analysis of nonlinear models. *Reliability Engineering & System Safety*, 52(1):1 – 17, 1996.

D. E. Huntington and C. S. Lyrintzis. Improvements to and limitations of latin hypercube sampling. *Probabilistic Engineering Mechanics*, 13(4):245 – 253, 1998.

J. E. Hurtado. *Structural robustness and its relationship to reliability*, volume 1 of *Structures and Infrastructures: Structural Design Optimization Considering Uncertainties*, chapter 16, pages 435 – 470. Taylor & Francis, London, UK, 2008.

G. Kharmanda, A. Mohamed, and M. Lemaire. Efficient reliability-based design optimization using a hybrid space withapplication to finite element analysis. *Structural and Multidisciplinary Optimization*, 24:233 – 245, 2002.

H. Kurtaran, A. Eskandarian, D. Marzougui, and N.E. Bedewi. Crashworthiness design optimization using successive response surface approximations. *Computational Mechanics*, 29:409–421, 2002.

A. Lange, M. Voigt, K. Vogeler, H. Schrapp, E. Johann, and V. Gümmer. Probabilistic CFD simulation of a high-pressure compressor stage taking manufacturing variability into account. *ASME Conference Proceedings*, 2010(44014):617–628, 2010.

T. Most. Efficient sensitivity analysis of complex engineering problems. In M. H. Faber, J. Köhler, and K. Nishijima, editors, *11th International Conference on Applications of Statistics and Probability in Civil Engineering*, Zurich, 2011. Balkema.

T. Most and J. Will. Metamodel of optimal prognosis - an automatic approach for variable reduction and optimal metamodel selection. In *Proceedings of the 5th Weimar Optimization and Stochastic Days*, Weimar, Germany, November 20-21, 2008. DYNARDO GmbH.

R. Parchem and B. Meissner. Engine multidisciplinary optimization deployed on a two-stage turbine. In Ernst Kesseler, editor, *Advances in Collaborative Civil Aeronautical Multidisciplinary Design Optimization*, pages 289 – 331. AIAA, Amsterdam, The Netherlands, 2009.

S. Pierret and R. van den Braembussche. Turbomachinery blade design using a navier-stokes solver and artificial neural network. *Journal of Turbomachinery*, 121:326–332, 1999.

D. Roos. Multi-domain adaptive surrogate models for reliability analysis. In H. Budelmann, A. Holst, and D. Proske, editors, *Proceedings of the 9th International Probabilistic Workshop*, pages 191 – 207. Technical University Carolo-Wilhelmina zu Braunschweig, Braunschweig, Germany, November 17-18 2011.

D. Roos, K. Cremanns, and T. Jasper. Probability and variance-based stochastic design optimization of a radial compressor concerning fluid-structure interaction. In S. Idelsohn, M. Papadrakakis, and B. Schrefler, editors, *V International Conference on Coupled Problems in Science and Engineering (Coupled Problems 2013)*, Conference Centre, Santa Eulalia, Ibiza, Spain, June 17-19, 2013. (Paper accepted for presentation).

A. Saltelli et al. *Global Sensitivity Analysis. The Primer*. John Wiley & Sons, Ltd, Chichester, England, 2008.

D. Sasaki, S. Obayashi, and H.-J. Kim. Evolutionary algorithm vs. adjoint method applied to sst shape optimization. In *The Annual Conference of CFD Society of Canada, Waterloo*, 2001.

S. Shahpar. A comparative study of optimisation methods for aerodynamic design of turbomachinery blades. In *Proceedings of ASME TURBOEXPO Nr. 2000-GT-523*, 2000.

Shyy, N. Wei, R. Papila, Vaidyanathan, and K. Tucker. Global design optimization for aerodynamics and rocket propulsion components. *Progress in Aerospace Sciences*, 71:59–118, 2001.

N. Stander and K.J. Craig. On the robustness of a simple domain reduction scheme for simulation-based optimization. *Eng. Comput.*, 19(4):431–50, 2002.

V. V. Toropov and L.F. Alvarez. Development of mars – multipoint approximation method based on the response surface fitting. Technical report, AIAA, 1998.

Vaidyanathan, N. Rajkumar, W. Papila, K. Shyy, R. Tucker, L. Griffin, Haftka N., and Fitz-Coy. Neural network and response surface methodology for rocket engine component optimization. *AIAA*, pages 2000–4880, 2000.

ON A TIME-SEQUENTIAL ADAPTIVE STRATEGY IN SPACE-TIME FOR FINITE STRAIN CONSOLIDATION PROBLEMS

Kenneth Runesson[*] and Fredrik Larsson[*]

[*]Chalmers University of Technology
Department of Applied Mechanics
SE41320 Göteborg, Sweden
e-mail: keru@chalmers.se, www.chalmers.se/am

Key words: Goal-oriented adaptivity, time-sequential strategy

Abstract.   The paper outlines a time-sequential space-time adaptive FE-strategy applied to finite strain coupled consolidation, which can be viewed as a prototype model of a class of nonlinear and time-dependent poro-mechanics problems. The natural variational setting for the consolidation problem allows for space-time FE using dG- or cG-methods in time depending on the expected character (quasistatic or dynamic). We discuss goal-oriented error computation and the combined space-time adaptivity while accounting for non-linearities in the model as well as the output functional.

One key ingredient in the proposed strategy is to introduce a hierarchical decomposition in space-time of the discrete function space(s) in which the approximate dual solution is sought. As a result, it is possible to decompose the estimated error from the discretization in space and time in a unified fashion within the same algorithm. This decomposition of error contributions allows for efficient adaptive mesh-refinement in space and time separately. Moreover, other sources of error (model and solution errors) can be identified.

Traditionally, controlling the global error in space-time problems involves storing the complete solution and, when adopting an adaptive algorithm, complete re-computation of the solution for each iteration of the space-time mesh. The main idea proposed in this contribution is to increase the computational efficiency of the adaptive scheme by avoiding recursive adaptations of the entire space-time mesh; rather, the space-mesh and the time-step defining each finite space-time slab are defined in a truly sequential fashion. The procedure involves the solution of an initial, approximate, dual solution on a coarse "background" space-time mesh which is kept fixed during the space-time re-meshing for the primal problem.

The overall performance of the proposed strategy is investigated using a few numerical examples.

A VARIATIONAL FORMULATION OF DAMAGE FOR THERMO-VISCOELASTIC PROBLEMS IN LARGE STRAINS

A.E. SELKE[*], L. STAINIER[*] AND E.A. FANCELLO[†]

[*] Institut de Recherche en Génie Civil et Mécanique – GeM (UMR CNRS 6183)
Ecole Centrale de Nantes
1 rue de la Noë, 44321 Nantes, France
e-mail: Augusto.Emmel-Selke@ec-nantes.fr
Laurent.Stainier@ec-nantes.fr

[†] Grupo de Analise e Projeto Mecânico (GRANTE)
Campus Univesitario Reitor João David Ferreira Lima – Trindade
88040-900 Florianopolis, Brazil
e-mail: fancello@grante.ufsc.br

Key words: Variational formulation, Damage, Thermo-viscoelasticity, Large strains.

Abstract. A variational formulation of damage for thermomechanically coupled problems is proposed. Based on variational methods for dissipative materials developed in the last few years (especially since [1]), this formulation allows for the accurate simulation of a wide variety of damage models, encompassing the behavior of various classes of polymers, the target application of the present paper.

A full development of the proposed formulation is presented in the first section. From an energy-like scalar-valued functional, the balance equations of the problem are derived, including the coupling terms between the different physics. The treatment of thermal effects follows [2]. A distinction between an internal and an external temperature allows for a factorization that, contrary to more classic formulations, keeps a symmetric structure to the problem. In order to describe rate dependence phenomena, the deformation energy potential is constructed from a generalized Kelvin-Voigt/Maxwell rheological model. Eigenvalues of strains are used as independent variables so that large strains are properly treated, following [3]. Having established the ingredients for a variational formulation of thermo-viscoelasticity, we move on to include damage effects. After a brief description of the choices made in [4] for the modeling of low cycle metal fatigue, a general framework for the inclusion of different damage models is presented.

In the second section, different damage models are applied. Although the framework we present is sufficiently general to consider anisotropic models, only isotropic examples of damage are presented in this article. The simplest case, where a scalar damage variable acts only upon the elastic part of the deformations, is presented first. We then apply the damage variable to both elastic and viscous parts of the deformation, which yields a more complex behavior. The possibilities of developing even more complex damage models, with different

damage variables acting upon each rheological Maxwell branch, is also discussed – it should be noted, though, that such models bring about increasingly tougher challenges in parameter identification. Different damage evolution laws are tested to show the versatility of the formulation.

The third and final section consists of the validation of the proposed model for the simulation of some characteristic phenomena of polymer damage behavior. A final discussion of further possible applications of the formulation concludes the present article.

REFERENCES

[1] Ortiz, M. and Stainier, L. The variational formulation of viscoplastic constitutive updates. *Computer Methods in Applied Mechanics and Engineering* (1999) **171**:419-444.

[2] Yang, Q., Stainier, L. and Ortiz, M. A variational formulation of the coupled thermo-mechanical boundary-value problem for general dissipative solids. *Journal of the Mechanics and Physics of Solids* (2006) **54**:401-424.

[3] Fancello, E., Ponthot, J-P. and Stainier, L. A variational formulation of constitutive models and updates in nonlinear finite viscoelasticity. *International Journal for Numerical Methods in Engineering* (2006) **65**:1831-1864.

[4] Kintzel, O. and Mosler, J. An incremental minimization principle suitable for the analysis of low-cycle fatigue in metals: A coupled ductile-brittle damage model. *Computer Methods in Applied Mechanics and Engineering* (2011) **200**:3127-3138.

METHODS AND TOOLS FOR PARALLEL ANISOTROPIC MESH ADAPTATION AND ANALYSIS

MARK S. SHEPHARD[*], E. SEEGYOUNG SEOL[*], CAMERON W. SMITH[*], MISBAH MUBARAK[*], ALEKSANDR OVCHARENKO[*] AND ONKAR SAHNI[*]

[*]Scientific Computation Research Center
Rensselaer Polytechnic Institute
Troy, New York 12180
e-mail: shephard@rpi.edu, smithc11@rpi.edu, seols@rpi.edu, mubarm@rpi.edu,
shurik.asa@gmail.com, sahni@rpi.edu

Key words: Mesh adaptation, boundary layer, parallel adaptation, dynamic load balancing.

Summary. It is well known that adaptive methods provide the most effective means to obtain reliable solutions and control the amount of computation required. However, for many classes of problems the best adaptive method still requires a level of computation that demands massively parallel computing. This paper presents a set of technologies for parallel adaptive simulation that includes a parallel mesh infrastructure, dynamic load balancing procedures and parallel anisotropic mesh adaptation. Examples of anisotropically adapted meshes for real-world fluid flow problems, including boundary layer meshes, are given.

1   INTRODUCTION

Adaptive anisotropic unstructured mesh technologies support the effective analysis of complex physical behaviours modelled by partial differential equations over general three-dimensional domains. Although adaptively defined anisotropic meshes can have two to three orders of magnitude fewer elements than a more uniform mesh for the same level of accuracy, there are many cases where the desired size of the adapted meshes are so large that they can only be solved on parallel computers. The execution of a simulation on a parallel computer requires the mesh to be distributed over the nodes and cores of the parallel computer.

The design of an infrastructure supporting adaptive unstructured meshes on massively parallel computers must consider the management of mesh information, the modification operations to be carried out on the meshes, and the scalability of the algorithms. The most basic functionality of the mesh infrastructure is to support the distribution of the mesh over the cores of the parallel computer. In adaptive simulations additional functionality is needed to adapt the mesh in parallel as the simulation process proceeds. A key requirement of effective parallel simulation is maintaining equal distribution, or load balance, of the computations, especially for the typically dominant analysis-related computations. Since mesh adaptation locally increases and/or decreases the mesh density, methods are needed to redistribute the mesh in order to regain load balance for the subsequent analysis step.

This paper presents a set of three components that are needed to support geometry-based parallel adaptive simulations. The first component is a parallel mesh infrastructure designed to support evolving meshes of any size on massively parallel computers. The second component

supports dynamic load balancing procedures that are capable of regaining the load balance of meshes as they adaptively evolve. The third component is a generalized mesh modification procedure that can execute anisotropic mesh adaptation, including boundary layer meshes, in parallel on distributed meshes.

2 PARALLEL MESH INFRASTRUCTURE

It is clear that simulations on meshes with millions and billions of elements require the mesh to be distributed over the compute cores of the massively parallel computers that execute the simulation. The most common form of mesh decomposition over distributed memory compute cores is to partition into a set of parts where the individual parts are groups of mesh entities. Between neighboring parts, the number of mesh entities on the common boundary is kept as small as possible in order to minimize communications. In addition to the mesh maintaining parallel distribution information, it must also maintain information relating it to the high-level problem domain definition to effectively support general mesh adaptation. This information allows mesh adaptation processes to account for the actual shape of the problem domain as the mesh is adapted and not be restricted to the geometric approximation defined by the initial mesh [12]. Rebalancing the workload as the mesh is adapted requires effective methods to migrate mesh entities between parts and update the inter-part communication links. The Parallel Unstructured Mesh Infrastructure (PUMI) is being developed to support the needs of adaptively evolving meshes on massively parallel computers [22], [23].

The geometric model is the high-level (mesh independent) definition of the domain that consists of a set of topological entities with adjacencies and associated shape information. A general non-manifold boundary representation [26] is needed to support the full range of requirements that often include multiple material domains and reduced dimension entities as elements in the mesh. PUMI interacts with the geometric model through a functional interface that supports the ability to interrogate the geometric model for the adjacencies of the model entities and geometric information about the shape of the entities [2], [24]. The use of such a representation allows the mesh adaptation procedures to interact with the geometric domain through simple topologically driven geometric interrogations to ensure that the mesh modifications are consistent with the actual geometric domain [12]. Other interactions with the geometric domain definition support the proper transformation of the input boundary condition fields onto the mesh and using geometric interrogations to the original geometry to support element integrations for mesh entities bounded by curved domain boundaries [7]. Effective execution of mesh adaptation and field transformation procedure requires a complete mesh representation in which the complexity of any mesh adjacency interrogation is $O(1)$ (i.e., not a function of mesh size [3], [22]). Meeting this requirement does not require the explicit storage of every one of the four levels/orders of mesh entities (vertex, edge, face and region) or all of the 12 possible adjacencies. However, it is critical to use a complete mesh representation [20], [22] so that information on any mesh entity or mesh entity adjacency can be obtained in $O(1)$ time.

2.1 Partition Model

When a mesh is distributed to N parts, each part is assigned to a process or processing core. A part is a subset of topological mesh entities of the entire mesh denoted by P_i, $0 \leq i < N$. Figure 1 depicts a 2-D mesh that is distributed to three parts. Each part is treated as a serial mesh with the addition of part boundaries to describe groups of mesh entities that form the links between parts. Mesh entities on part boundaries, called *part boundary entities,* are duplicated on all parts for which they bound other higher order mesh entities. Mesh entities that are not on any part boundaries exist on a single part and are called *interior mesh entities* (with respect to the part). The boundary between the parts in different processes is an inter-process part boundary and the boundary between the parts on the same process is an intra-process part boundary.
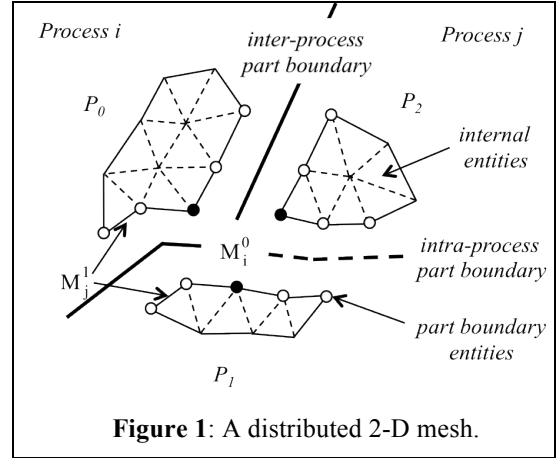
Figure 1: A distributed 2-D mesh.

For each mesh entity, the *residence part set* [22], [23] is a set of part ID(s) where a mesh entity exists based on adjacency information: If mesh entity M_i^d is not adjacent to any higher dimension entities, the residence part set of M_i^d is the ID of the single part where M_i^d exists. Otherwise, the residence part set of M_i^d is the set of part IDs of the higher order mesh entities that are adjacent to M_i^d. Note that part boundary entities share the same residence part if their locations with respect to the part boundaries are the same.

In the 2-D mesh illustrated in Figure 1, the part boundary entities are the vertices and edges that are adjacent to mesh faces on multiple parts. The residence part set of vertex M_i^0 and edge M_j^1 are $\{P_0, P_1, P_2\}$ and $\{P_0, P_1\}$, respectively.

For the purpose of representation of a partitioned mesh and efficient parallel operations, a *partition model* is developed.

- Partition (model) entity: A topological entity, P_i^d, which represents a group of mesh entities of dimension d or less, which have the same residence part set. One part among all parts in residence part set is designated as the owning part.
- Partition classification: Unique association of mesh entities to a partition model entity.

Figure 2 depicts the partition model of the distributed mesh. The mesh vertex M_i^0, depicted in Figure 1, duplicated on three parts, is classified on the partition vertex P_0^0 such that P_0^0 represents mesh vertex duplicated on part $\{P_0, P_1, P_2\}$, P_0^1, P_1^1, and P_2^1 represent mesh edges and vertices duplicated on part $\{P_0, P_1\}$, $\{P_0, P_2\}$, and $\{P_1, P_2\}$, respectively. At the mesh entity level, the proper partition classification is needed to maintaining up-to-date residence part set and owning part information which is key to effective support of an evolving distributed mesh. As illustrated in Figure 3, the partition model can be viewed as a part of hierarchical domain decomposition.
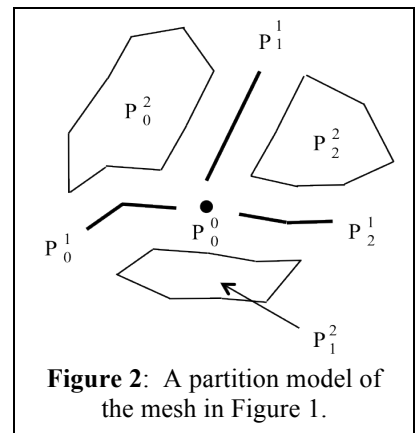

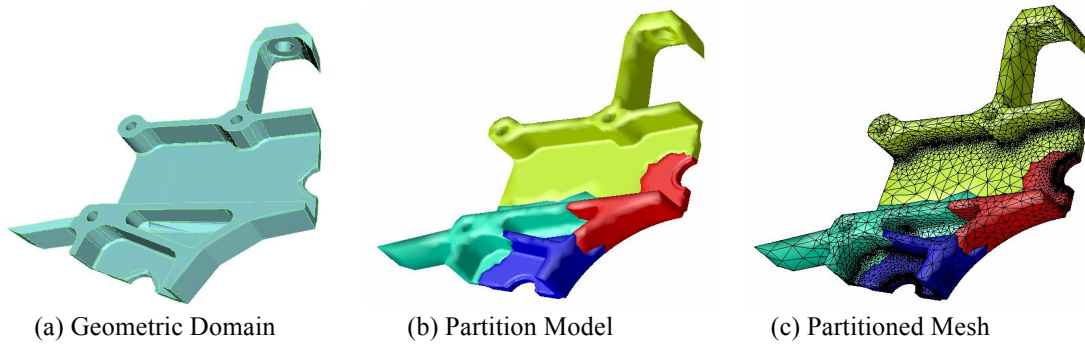**Figure 2**: A partition model of the mesh in Figure 1.

(a) Geometric Domain   (b) Partition Model   (c) Partitioned Mesh

Figure 3: Domain representation hierarchy: (a) geometric model, (b) partition model and (c) partitioned mesh.

2.2 Migration

Mesh migration is a procedure that moves mesh entities from one part to another. Migration supports: (i) partitioning, (ii) dynamic load balancing, and (iii) obtaining mesh entities needed for mesh modification operations. In the mesh migration procedure, a partition object is the basic unit to assign a destination part id. It can be either a mesh entity not on the boundary of any higher dimension mesh entities or a group of mesh entities contained in a single part called p-set [27]. Figure 4 presents the pseudo code of migration algorithm while figure 5 shows the steps involved.

```
INPUT: a list of partition objects and destination parts. See Figure 5(a).
OUTPUT: a mesh with new partitioning

Step 1. Collect all mesh entities that will be effected by migration.
Step 2. For entities collected in step 1, determine residence part set and
update partition classification based on new partitioning. See Figure 5(b).
Step 3: Among entities collected in step 1, collect entities to remove
based on residence part set.
Step 4. Exchange p-set and entities along with partition classification.
See Figure 5(c).
Step 5.  For entities collected in step 1, and newly created entities on
destination parts, update part boundary links.
Step 6: Delete entities collected in step 3. See Figure 5(d).
```

Figure 4: Pseudo code of migration algorithm.

2.3 Ghosting

A class of operations on mesh entities near a part boundary such as error estimation and element shape optimization often require data from adjacent mesh entities that are internal to neighboring parts. This information could be obtained by the repeated communication requests between parts. However, such communications adversely affect scalability and performance. Thus it is desirable to minimize them and localize the data for part boundary computations. In cases where this needed data is static for the entire operation, such as solution fields used in error estimation, or change slowly, such as coordinates in element shape optimization, an alternative is to provide all the needed information through local

copies of the remote data. The common approach creates one or more layers of *ghost* copy of the needed un-owned mesh and field data near part boundaries [8], [25].
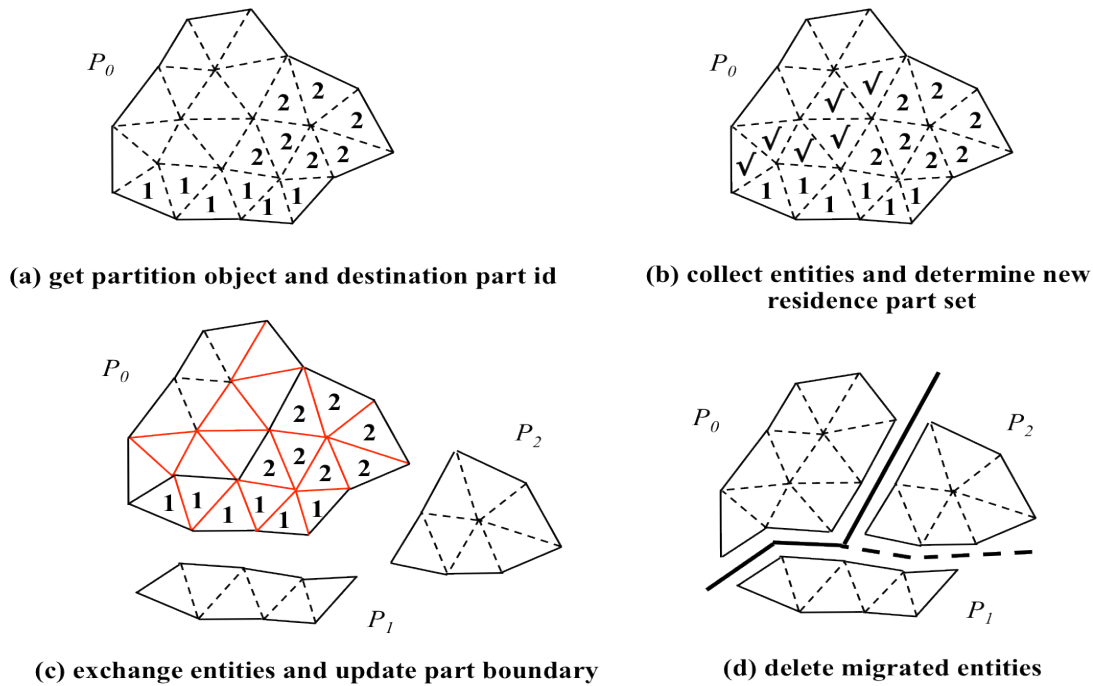

(a) get partition object and destination part id

(b) collect entities and determine new residence part set

(c) exchange entities and update part boundary

(d) delete migrated entities

Figure 5: Migration steps demonstrated on a 2-D mesh.

To support a full range of ghost requirements for unstructured meshes, a generalized N-layer ghosting parallel ghosting algorithm has been developed and implemented [15] in PUMI. The algorithm supports the creation of layers of copied mesh entities of desired order and their bounding entities based on specification of a bridge dimension using mesh adjacencies. The key components of the ghost creation process are:

- Ghost dimension: Permissible options in a topological mesh representation can be regions, faces or edges. As ghost entities are specified through a bridge dimension, the lowest possible dimension of a ghost entity can be an edge since the minimum bridge is a vertex. Vertices are ghosted if they are part of higher dimension ghost entities. For example, in a 1-D mesh, the only possible ghost dimension is an edge and the vertices that are on the boundary of ghost edges can be ghosted to create the ghost edges.

- Bridge dimension: Ghost entities are specified similar to second order adjacencies using a bridge entity. The bridge entity must be of lower topological order than the ghost and can be a face, edge or a vertex. Two common examples of ghosted entities are: (i) the mesh regions that are bounded by faces classified on a partition model face, and (ii) the mesh regions that are bounded by vertices classified on partition model vertices, edges or faces. A less common, but supported case would be the mesh edges that are bounded by vertices classified on partition model vertices, edges or faces.

- Number of layers: Number of layers of ghost entities. Layers are measured from the part boundary.

For example, to get two ghost layers of regions, bridged by bounding vertices, the ghost dimension is set to region, the bridge dimension set to vertices and number of layers is 2. When multiple ghost layers are requested by an application, the ghosting process starts with the first (innermost) layer of ghosts adjacent to part boundary. Figure 6 shows zero, one and two layers of ghosted mesh regions based on a vertex bridge.
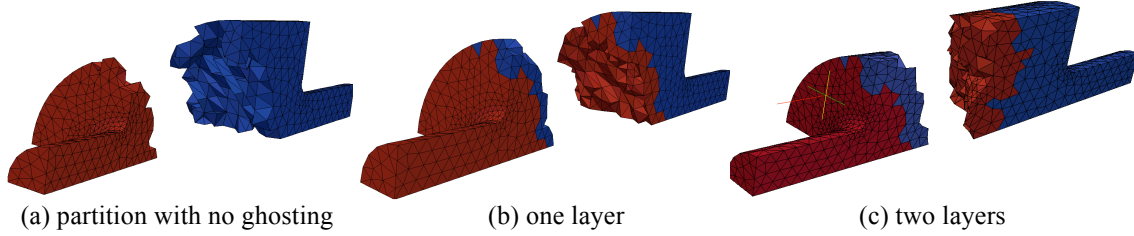

(a) partition with no ghosting   (b) one layer   (c) two layers

Figure 6: Application of region ghosting based on a vertex-based bridge.

Figure 7 illustrates the ghosting procedure with ghost dimension 2, bridge dimension 1 and the number of layers 1: (a) initial mesh (b) collect mesh faces to ghost which are bounded by partition model edges (c) for mesh faces and their downward adjacent entities collected in step b, determine the destination part id(s) to migrate to based on residence part set (d) exchange entities (e) at the original copy, update ghost copy information.
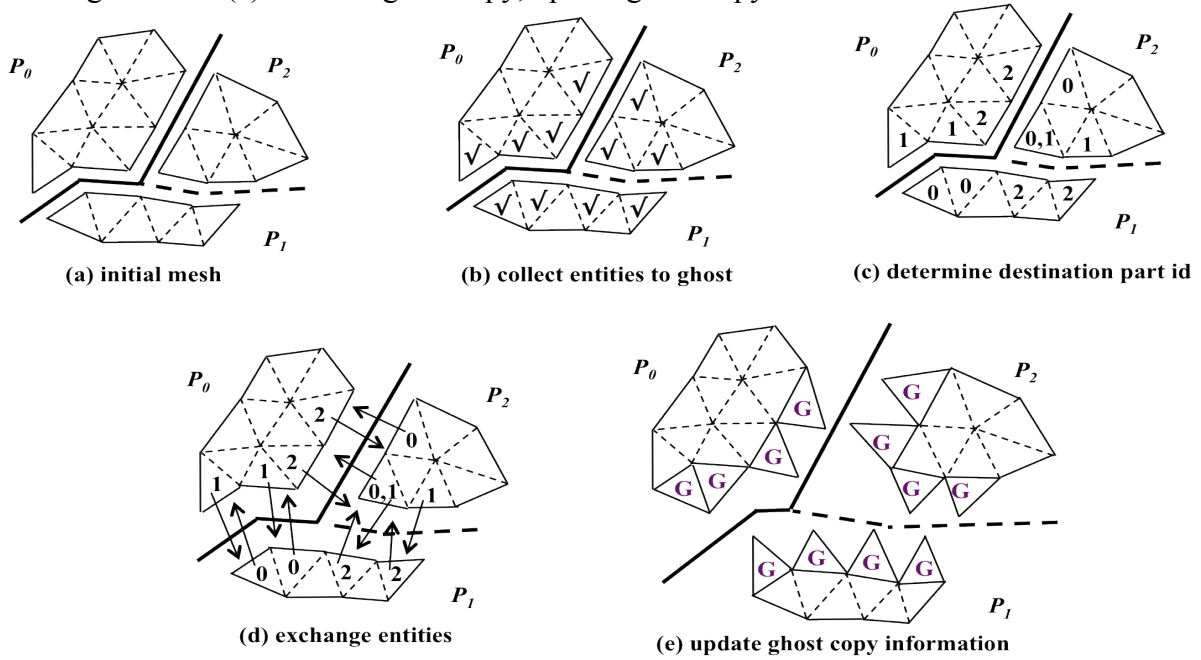

(a) initial mesh   (b) collect entities to ghost   (c) determine destination part id

(d) exchange entities   (e) update ghost copy information

Figure 7: 2-D Ghosting steps.

A ghost entity stores information about its owner entity and the part where the owner entity exists. At a minimum, a ghost entity's owner must also store information about its ghost copies that exist. This synchronizes the ghost copies synchronized with their owner entities and eliminates the need for inter-part communication if there are any queries about ghost entity ownership. The inter-part communication required in the ghost creation process is optimized by utilizing a general-purpose package that sends messages within a fixed process

neighborhood by packing small MPI messages and avoiding unnecessary calls to MPI collective operations [18].

3   DYNAMIC LOAD BALANCING

Another key component of supporting unstructured mesh simulation workflows is dynamic load balancing. At a minimum, the mesh needs to be partitioned such that resulting adapted mesh fits within the memory of each node onto which the mesh is partitioned. PDE analysis additionally requires consideration of the workload by accounting for the different entity types defining the computation load of the phases of the analysis. In both cases peak imbalance determine performance since one heavily loaded processor can force all the others to sit idle while it completes. Small valleys (with load below average) leave a few processes idle having a minimum affect of scaling. Therefore, the reduction of peaks for each workflow step is critical for parallel performance and scaling. A dynamic partitioning algorithm must also account for the connectivity of the unstructured mesh such that the part boundaries are optimized to minimize the amount of communications across neighboring parts.

The most powerful partitioning procedures for meshes are the graph and hypergraph-based methods as they can explicitly account for application defined balance criteria via graph node weights, and one piece of the mesh connectivity information via the definition of graph edges. Hypergraph-based methods can further optimize the mesh partition at the cost of increased run-time over the graph-based methods [4]. Graph based methods balance the weighted values of the graph nodes while trying to minimize the number of graph edges between parts. When partitioning an unstructured mesh, the graph nodes are selected to be the appropriate set of mesh entities, where in most cases the set of graph nodes are all the mesh entities of the highest order (mesh regions in 3D and mesh faces in 2D). The graph edges are defined by the mesh adjacencies that happen to be of importance to the simulation step for which the partition is being constructed. For example in the case of linear finite elements where the unknowns are at mesh vertices, creating a graph edge between each mesh regions that shares a vertex is important, while in a face-based finite volume procedure graph edges should be defined between the pair of mesh regions that share a face.

In addition to the use of standard graph-based procedures, consideration is being given to drive selected dynamic load balancing operations directly from the adjacencies of the mesh entities since it is the selected adjacency information that defines the graph edges in standard graph-based partitioners. Two advantages of a tool that performs parallel Partitioning using Mesh Adjacencies (ParMA) are: (i) it can more easily account for the balancing of mesh entities of different dimensions at the same time, and (ii) could potentially be more computationally and memory efficient since, by working with the existing mesh topological information, it avoids the need to construct a separate partition graph.

The ability to use ParMA to effectively improve mesh partitions in-turn to improve the scalability of finite element solvers has been clearly demonstrated [28]-[31], where consideration was given to the balance of mesh regions, critical for equation formation, as well as mesh vertices, critical for equation solution of linear finite elements. Recent extensions to ParMA have generalized these procedures such that mesh entities of all orders, with assigned priority, can be considered [23]. For example, in the case of quadratic finite elements, there are unknowns at both the mesh vertices and mesh edges. Thus in that case

there is a high priority given to balancing the mesh edges and vertices, since equation solution is the dominating computational step, while the mesh regions are given a lower priority, since their balance controls the scaling of the equation formation step.

A second area of application of ParMA currently being investigated is repartitioning before a mesh adaptation step. Typically the mesh before a mesh adaptation step is well balanced, however, since the mesh adaptation procedures are going to refine the elements is some areas and coarsen them in others, the adapted mesh would be dramatically out of balance after adaptation, to the point that exceeding available memory becomes quite likely. Thus before the mesh is adapted, the new mesh size information is used to assign weights to the current mesh entities (>1 in areas where the mesh will get finer, <1 in areas where the mesh will get coarser) and the mesh in rebalanced. The execution of this process yields a mesh that is very close to being well balanced after mesh adaptation [29]. A full graph-based predictive partitioning [29] is used before mesh adaptation to ensure it will execute without problems and again at the end to refine the balance for the next analysis step. Noting that the only goal of the current predictive load balancing is to load balance the subsequent analysis after the mesh is adapted, it is not considering the scaling of the adaptive process itself, and that even with its current goals, the load balance must be improved between mesh adaptation and the next analysis step. This indicates that there is potential for improving the process. The primary idea under current consideration, potentially as the mesh is being adapted, is to merge neighboring parts in which the number of mesh entities after adaptation will be less than, or equal to, that of a balanced part, and to split parts that will be heavily refined into a number of parts such that each has about the average number of mesh entities after adaptation.

4   PARALLEL ANISOTROPIC MESH ADAPTATION

Many physical problems of interest involve directional solution features. To address such cases adaptive mesh control methods are designed to match an anisotropic mesh size field defined through the application of a posteriori correction indicators [1], [5], [9], [19]. In the case of viscous flow problems is it is important to supplement the general anisotropic mesh adaptation procedures with ones that can maintain a semi-structured boundary layer mesh on selected boundaries [5], [10], [11], [17], [21]. The two approaches to creating the adaptive anisotropic meshes given an adaptively defined anisotropic mesh size field, including adapted boundary layers, are complete domain re-meshing methods, and methods that use local mesh modification. Adaptive re-meshing accounts well for curved domains and mesh resolution. However, this is at the cost of re-meshing the entire domain. A more serious concern of the use of global re-meshing, especially for problems where accurate transfer of the solution fields to the new mesh is required, is both the cost and accuracy of general mesh-to-mesh solution transfers. Conversely, mesh adaptation based on local mesh modification can be a faster method that when coupled with local solution transfer methods, can provide more accurate solution transfer. However, the set of local mesh modification operators must be rich enough to be able to produce the desired anisotropic mesh configurations, while accounting for the curved domain geometry (e.g., as defined by the CAD). A local mesh modification-based procedure that meets these requirements builds off a complete set of mesh modification operations that include compound operators [13] and that maintains semi-structured boundary layers (if any) [17], [21], while local operations also ensure that the adapted mesh conforms to

the curved domain geometry [12], [13].

The parallel implementation of the general mesh modification operators work directly with the partitioned mesh by querying the PUMI provided partition model to coordinate operations during mesh refinement. Mesh coarsening and swapping operations are supported by PUMI mesh migration functions to move required mesh entities between parts [6]. The parallel migration procedures have been extended to include mesh sets that require stacks of semi-structured mesh entities to be migrated together thus supporting the parallel execution of semi-structured boundary layer adaptation [17].

Figure 8 shows a simple example of parallel mesh adaptation including a boundary layer. Figure 8(a) shows an initial coarse mesh that includes a boundary layer while Figure 8(b) shows an adapted mesh on the same simple geometry. In more general cases, the adaptation of the boundary layer can locally reduce the anisotropy to the point where it is desirable to convert the top portions of the boundary layer to be a regular unstructured mesh such that more general unstructured mesh modification operations can be applied. To support this functionality the mesh adaptation procedures need additional extensions to deal with the pyramid elements that are introduced by local trimming of the boundary layer mesh (Figure 9) [17].
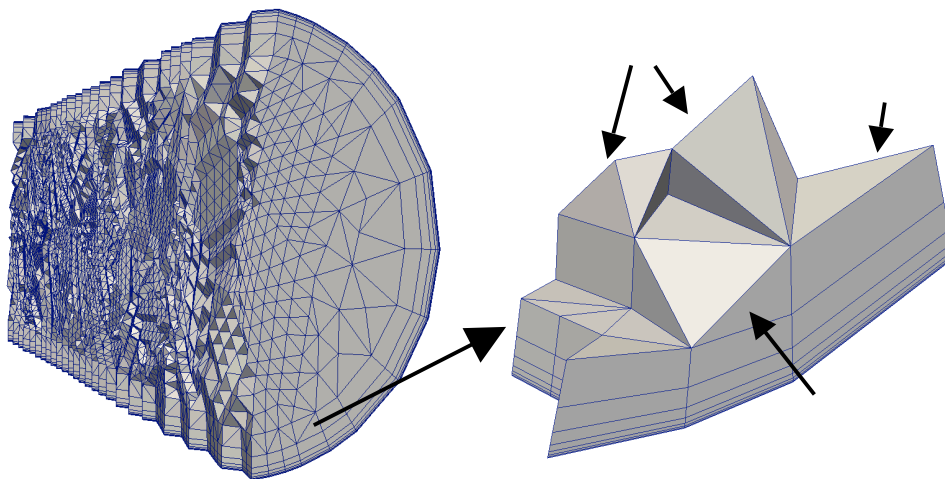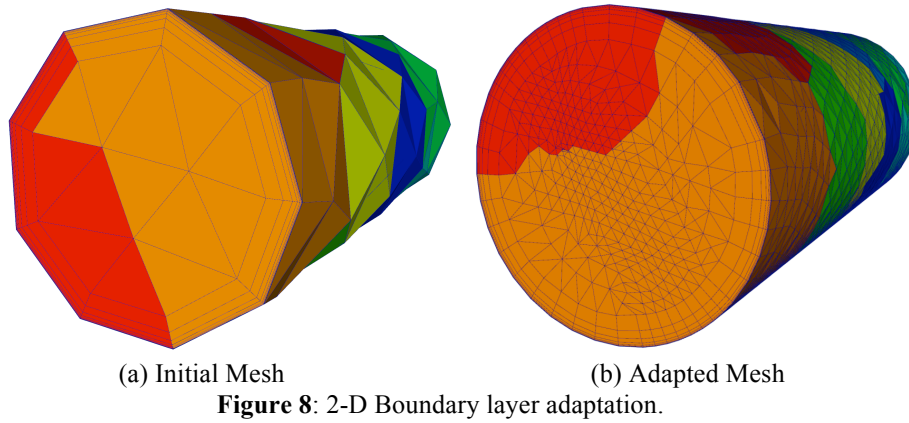

(a) Initial Mesh   (b) Adapted Mesh
Figure 8: 2-D Boundary layer adaptation.

Figure 9: An example requiring the introduction of pyramid elements to allow the trimming boundary layers.

5   PARALLEL ADAPTIVE EXAMPLES

The first example is a viscous flow simulation of a NASA CIAM scramjet case run with a free stream Mach number of 6.2, and a free stream reference temperature of 203.5 Kelvin. The initial boundary layer mesh has 2.9M regions with a mid-section cut-away view of the boundary layer mesh, including close-up of the inlet, is shown in the top two images in Figure 10. The adapted boundary layer mesh has 43M regions and is shown in the bottom two images in Figure 10.
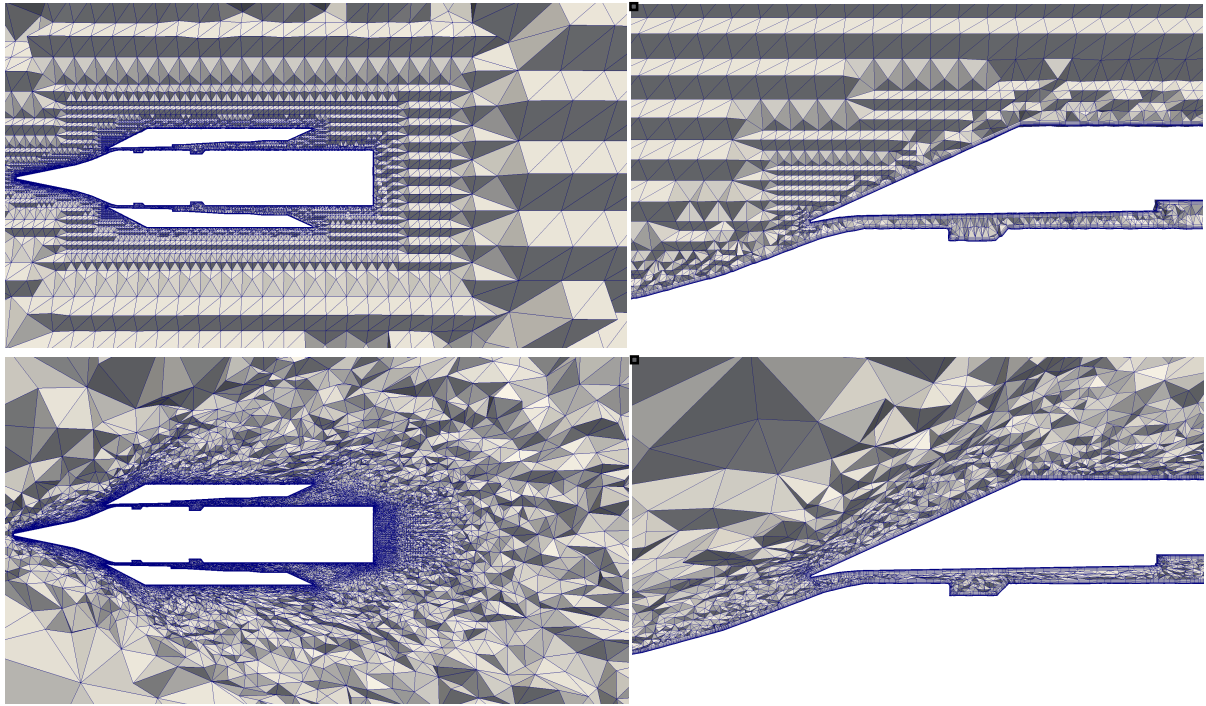

Figure 10: Initial and anisotropic adapted meshes for a scramjet engine.

The second example is a multiphase flow in which a fluid in being injected into air. In this example the interfaces between the fluid and air is modeled using a level set method [16]. The mesh adaptation procedure is keyed to perform anisotropic mesh adaptation at the zero level set that represents the dynamic two-fluid interface. Figure 11 shows the anisotropically adapted mesh at three different time steps in the simulation.

6   CLOSING REMARKS

This paper provides an overview of a set of procedures to perform parallel anisotropic mesh adaptation of unstructured meshes that can include semi-structured boundary layer meshes. The parallel mesh infrastructure, PUMI, parallel partitioning using mesh adjacencies, ParMA and mesh adaptation procedures, MeshAdapt, have been implemented using a component-based approach in which all interactions are controlled through functional interfaces. This approach allows these tools to be efficiently coupled with various unstructured mesh analysis codes and other mesh related components such as mesh generators, dynamic load balancers, etc. See http://www.scorec.rpi.edu/software.php for

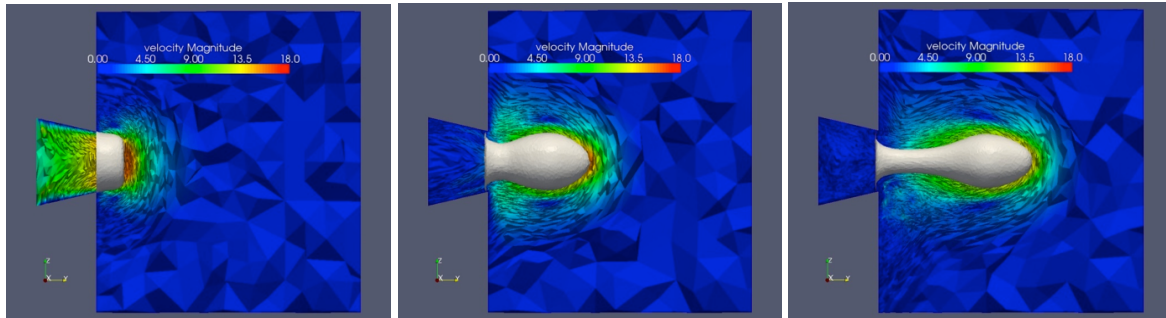information on, and access to, these components.

Figure 11: Anisotropic adapted meshes for a two-phase flow problem.

REFERENCES

[1] Alauzet, F., Li, X., Seol, E.S. and Shephard, M.S., "Parallel anisotropic 3D mesh adaptation by mesh modification", *Engineering with Computers*, 21 (2006) 247–258.

[2] Beall, M.W., Walsh, J. and Shephard, M.S., "A comparison of techniques for geometry access related to mesh generation," *Engineering with Computers,* 20(3):210-221, 2004.

[3] Beall, M.W. and Shephard, M.S., "A general topology-based mesh data structure," *Int. J. Numerical Methods in Engineering*, 40(9):1573–1596, 1997.

[4] Boman, E.G., Devine, K.D., Fisk, L.A., Heaphy, R., Hendrickson, B., Leung, V., Vaughan, C., Catalyurek, U., Bozdag, D. and Mitchell, W., "Zoltan home page," September 2011, http://www.cs.sandia.gov/Zoltan.

[5] Botasso, C.L., "Anisotropic mesh adaption by metric-driven optimization", *Int. J. Numer. Meth. Engng.*, 60 (2004) 597–639.

[6] de Cougny, H.L. and Shephard, M.S., "Parallel Refinement and Coarsening of Tetrahedral Meshes", *Int. J. Numer. Meth. Engng.*, 46:1101-1125, 1999.

[7] Dey, S., Shephard M.S. and Flaherty, J.E., "Geometry Representation Issues Associated with p-Version Finite Element Computations", *Comp. Meth. App. Mech. and Eng.*, 150(1-4):29-55, 1997.

[8] Dreher, J. and R. Grauer, R., "Racoon: A parallel mesh-adaptive framework for hyperbolic conservation laws", *Parallel Computing*, 31(8-9):913–932, 2005.

[9] Frey, P.L. and Alauzet, F., "Anisotropic mesh adaptation for CFD computations", *Computer Meth. Applied Mechanics and Engineering*, 194:5068–5082, 2005.

[10] Kallinderis Y. and Kavouklis, C., "A dynamic adaptation scheme for general 3-D hybrid meshes", *Comput. Methods Appl. Mech. Engrg*. 194:5019–5050, 2005.

[11] Khawaja, A., Minyard, T. and Kallinderis, Y., "Adaptive hybrid grid methods", *Comput. Methods Appl. Mech. Engrg.*, 189:1231–1245, 2000.

[12] Li, X., Shephard, M.S. and Beall, M.W., "Accounting for curved domains in mesh adaptation," *Int. J. Numerical Methods in Engineering*, vol. 58, no. 2, pp. 247–276, 2003.

[13] Li, X., Shephard, M.S. and Beall, M.W., "3-D Anisotropic Mesh Adaptation by Mesh Modifications", *Comp. Meth. Appl. Mech. Engng.*, 194(48-49):4915-4950, 2005.

[14] Lu, Q., Shephard, M.S., Tendulkar, S. and M.W. Beall, M.W., "Parallel Curved Mesh Adaptation for Large Scale High-Order Finite Element Simulations", *Proc. 21st International Meshing Roundtable, Springer*, NY, pp. 419-436, 2012.

[15] Mubarak, M., "A parallel ghosting algorithm for the flexible distributed mesh database (FMDB), Scientific Computation Research Center, RPI, Troy, NY, 2011, http://www.scorec.rpi.edu/reports/view_report.php?id=548.

[16] Nagrath, S., Jansen, K.E., and Lahey, R.T., Jr., "Computation of incompressible bubble dynamics with a stabilized finite element level set method", *Computer Methods in Applied Mechanics and Engineering*, 194:4565-4587, 2005.

[17] Ovcharenko, A., Chitale, K., Sahni, O., Jansen, K.E., Shephard, M.S., Tendulkar, S. and Beall, M.W., "Parallel Adaptive Boundary Layer Meshing for CFD Analysis", *Proc. 21st International Meshing Roundtable*, Springer, NY, pp. 437-455, 2012.

[18] Ovcharenko, A., Ibanez, D., Delalondre, F., Sahni, O., Jansen, K.E., Carothers, C.D. and Shephard, MS. "Neighborhood Communication Paradigm to Increase Scalability in Large-Scale Scientific Applications" *Parallel Computing*, 38(3):140-156, 2012.

[19] Park, M.A., "Parallel Anisotropic Tetrahedral Adaptation", 46th AIAA Aerospace Sciences Meeting and Exhibit, AIAA 208-917, 2008.

[20] Remacle, J.F., and Shephard, M.S., "An algorithm oriented mesh database," *Int. J. Num. Methods in Engineering*, 58(2):349–374, Sep. 2003.

[21] Sahni, O., Jansen, K.E., Shephard, M.S., Taylor, C.A. and Beall, M.W., "Adaptive boundary layer meshing for viscous flow simulations", *Engineering with Computers*, 24:267–285, 2008.

[22] Seol, E.S. and Shephard, M.S., "Efficient distributed mesh datastructure for parallel automated adaptive analysis," Eng. with Computers, 22(3-4):197–213, Nov. 2006.

[23] Seol, S., Smith, C.W., Ibanez, D.A. and Shephard, M.S., "A Parallel Unstructured Mesh Infrastructure", http://www.scorec.rpi.edu/reports/view_report.php?id=591, 2012.

[24] Shephard, M.S., "Meshing environment for geometry-based analysis", *Int. J. Numerical Methods in Engineering*, 47(1-3):169-190, 2000.

[25] Stewart, J.R. and H.C. Edwards, H.C, "A framework approach for developing parallel adaptive multiphysics applications", *Finite Elements Analysis and Design*, 40(12):1599–1617, 2004.

[26] Weiler, K.J., "The radial-edge structure: a topological representation for non-manifold geometric boundary representations," *Geometric Modeling CAD Appl.*, pp. 3–36, 1988.

[27] Xie, X, Seol, S., Shephard. M.S., "Generic Components for Petascale Adaptive Unstructured Mesh Simulations", *Eng. with Computers*, DOI 10.1007/s00366-012-0288-4, Accepted Sep. 2012.

[28] Zhou, M., Sahni, O., Xie, T., Shephard, M.S. and Jansen, K.E., "Unstructured Mesh Partition Improvement for Implicit Finite Element at Extreme Scale", *Journal of Supercomputing*, 59(3): 1218-1228, 2012.

[29] Zhou, M., Xie, T., Seol, S., Shephard, M.S. Sahni, O. and Jansen, K.E. "Tools to Support Mesh Adaptation on Massively Parallel Computers", *Engineering with Computers*, 28(3):287-301, 2012

[30] Zhou, M., Sahni, O., Shephard, M.S., Devine, K.E. and Jansen, K.E., "Controlling unstructured mesh partitions for massively parallel simulations", *SIAM J. Sci. Comp*., 32(6):3201-3227, 2010.

[31] Zhou, M., Sahni, O., Shephard, M.S., Carothers,C.D. and K.E. Jansen, J.E., "Adjacency based reordering algorithm for acceleration of finite element computations", *Scientific Programming*, 18(2):107-123, 2010.

ERROR ESTIMATION FOR THE CONVECTIVE CAHN – HILLIARD EQUATION

Görkem Şimşek[*], **Kristoffer G. van der Zee**[*,†], **E. Harald van Brummelen**[*,††]

[*]Multiscale Engineering Fluid Dynamics
Eindhoven University of Technology
5600 MB Eindhoven, Netherlands
e-mail: G.Simsek@tue.nl

[†] e-mail: K.G.v.d.Zee@tue.nl
[††] e-mail: E.H.v.Brummelen@tue.nl

Key words: Cahn–Hilliard equation, mixed finite element method, a-posteriori error analysis

Abstract. The Cahn–Hilliard phase-field (or diffuse-interface) model has a wide range of applications where the interest is the modelling of phase segregation and evolution of multiphase flow systems. In order to capture the physics of these systems, diffuse-interface models presume a nonzero interface thickness between immiscible constituents, see [1]. The multiscale nature inherent in these models (interface thickness and domain size of interest) urges the use of space-adaptivity in discretization.

In this contribution we consider the a-posteriori error analysis of the convective Cahn–Hilliard [4] model for varying Péclet number and interface-thickness (diffusivity) parameter. The adaptive discretization strategy uses mixed finite elements, a stable time-stepping algorithm and residual-based a-posteriori error estimation [2, 5]. This analysis for the convective model forms a basic step in our research and will be helpful to the coupled Cahn–Hilliard/Navier–Stokes system [3] which is the desired model for future research.

1 INTRODUCTION

1.1 The Model

Let $\Omega \subset \mathbb{R}^d$ be a bounded domain with $d = 1, 2, 3$ and $\partial\Omega$ be the boundary which has an outward unit normal $\mathbf{n}$. The convective Cahn-Hilliard equation can be written as follows:

Find the real valued functions $(c, \mu) : \Omega \times [0, T] \to \mathbb{R}$ for $T > 0$ such that

$$\partial_t c - \frac{1}{Pe} \triangle \mu + \nabla \cdot (\mathbf{u}c) = 0 \quad \text{in} \quad \Omega_T := \Omega \times (0, T]$$

$$\mu = \phi'(c) - \epsilon^2 \nabla c \quad \text{in} \quad \Omega_T$$

$$c(\cdot, 0) = c_0 \quad \text{in} \quad \Omega$$

$$\partial_{\mathbf{n}} c = \partial_{\mathbf{n}} \mu = 0 \quad \text{on} \quad \partial\Omega_T := \partial\Omega \times (0, T],$$

where $\partial_t(\cdot) = \partial(\cdot)/\partial t$, $\partial_{\mathbf{n}}(\cdot) = \mathbf{n} \cdot \nabla(\cdot)$ is the normal derivative, ϕ is the real-valued free energy function, $\mathbf{u}$ is a given function such that $\nabla \cdot \mathbf{u} = 0$ in Ω and $\mathbf{u} \cdot \mathbf{n} = 0$ on $\partial\Omega$, Pe is the *Péclet* number and ϵ is the interface thickness.

The nonlinear energy function $\phi(c)$ is of the double well form and we consider the following C^2-continuous function :

$$\phi(c) := \begin{cases} (c+1)^2 & c < -1, \\[2mm] \frac{1}{4}\left(c^2 - 1\right)^2 & c \in [-1, 1], \\[2mm] (c-1)^2 & c > 1. \end{cases}$$

1.2 Weak Formulation

In order to obtain the weak formulation, we consider the following function space and the corresponding norm as a suitable space for μ:

$$V := L^2(0, T; H^1(\Omega)), \quad \|v\|_V^2 := \int_0^T \|v(t)\|_{H^1_{(\Omega)}}^2 \, dt$$

and the space suitable for the phase variable c is

$$W := \left\{ v \in V : v_t \in V' \right\},$$

where $V' := L^2(0, T; [H^1(\Omega)]')$ is the dual space of V with the norm $\|v_t\|_W^2 := \|v\|_V^2 + \|v_t\|_{V'}^2$, where

$$\|v_t\|_{V'}^2 := \int_0^T \|v_t(t)\|_{[H^1(\Omega)]'}^2 \, dt.$$

Then the weak form of the problem becomes:

Find $(c, \mu) \in W_{c_0} \times V$:

$$\langle c_t, w \rangle + (u\nabla c, w) + \frac{1}{Pe}\left(\nabla \mu, \nabla w\right) = 0 \quad \forall w \in H^1(\Omega)$$

$$(\mu, v) - (\phi(c), v) + \epsilon^2 \left(\nabla c, \nabla v\right) = 0 \quad \forall v \in H^1(\Omega),$$

for $t \in [0, T]$, where W_{c_0} is the subspace of W of which the trace at $t = 0$ coincide with c_0.

To derive an a-posteriori error representation, we will employ the mean-value-linearized adjoint problem. The dual problem can be defined in terms of dual variables (p, χ) where the dual variable p is a function in the space

$$W^{\bar{q}} := \{v \in W : v(T) = \bar{q}\}.$$

Then the dual problem can be written:
Find $(p, \chi) \in W^{\bar{q}} \times V$:

$$
\begin{aligned}
-\partial_t p + u\nabla p + \epsilon^2 \triangle \chi - \phi'(c, \hat{c})\chi &= q_1 && \text{in} \quad \Omega \times [0, T) \\
\chi - \frac{1}{Pe}\triangle p &= q_2 && \text{in} \quad \Omega \times [0, T) \\
p &= \bar{q} && \text{on} \quad \Omega \times \{t = T\} \\
\partial_{\mathbf{n}} p = \partial_{\mathbf{n}} \chi &= 0 && \text{on} \quad \partial\Omega \times [0, T],
\end{aligned}
$$

where the nonlinear function $\phi'(c, \hat{c})$ is a mean-value-linearized function

$$\phi'(c, \hat{c}) = \int_0^1 \phi''\left(sc + (1 - s)\hat{c}\right) ds.$$

REFERENCES

[1] Anderson, D.M., McFadden, G.B. and Wheeler, A.A. Diffuse-Interface Methods in Fluid Mechanics. *Annu. Rev. Fluid Mech.*(1998) **30**:139–65

[2] Bartels, S., Müller, R. A-posteriori error controlled local resolution of evolving interfaces for generalized Cahn–Hilliard equations. *Interfaces and Free Boundaries* (2010) **12**:45–73

[3] Boyer, F., Lapuerta, C., Minjeaud, S., Piar, B. and Quintard, M. Cahn-Hilliard/NavierStokes Model for the Simulation of Three-Phase Flows. *Transport in Porous Media* (2010) **82**:463-483

[4] Kay, D., Styles, V. and Süli, E. Discontinuous Galerkin Finite Element Approximation of the Cahn–Hilliard Equation with Convection. *SIAM J. Numer. Anal.*(2009) **47**:2660–2685

[5] Van der Zee, K. G., Oden, J. T., Prudhomme, S. and Hawkins-Daarud, A. Goal-oriented error estimation for Cahn–Hilliard models of binary phase transition. *Numer. Methods Partial Differ. Equations* (2011) **27**:160–196

VI International Conference on Adaptive Modeling and Simulation
ADMOS 2013
J. P. Moitinho de Almeida, P. Díez, C. Tiago and N. Parés (Eds)

HEMODYNAMICS OF A STENOSED CAROTID BIFURCATION

LUÍSA C. SOUSA *[1,2], CATARINA F. CASTRO[1,2], CARLOS C. ANTÓNIO[1,2], RUI CHAVES[2], ROSA SANTOS[3], PEDRO CASTRO[3] AND ELSA AZEVEDO[3]

[1]Department of Mechanical Engineering, FEUP

Institute of Mechanical Engineering (IDMEC)

University of Porto

Dr. Roberto Frias, s/n, 4200-465 Porto, Portugal

e-mail: lcsousa@fe.up.pt; ccastro@fe.up.pt; cantonio@fe.up.pt

[2]Institute of Mechanical Engineering (IDMEC)

University of Porto

Dr. Roberto Frias, s/n, 4200-465 Porto, Portugal

e-mail: ec03210@fe.up.pt

[3]Faculty of Medicine (FMUP), University of Porto

Hospital São João

Alameda Prof. Hernâni Monteiro, 4200-369 Porto, Portugal

email: rosampsantos2@gmail.com; pedromacc@gmail.com; elsaazevedo@netcabo.pt

Key words: Carotid Artery Bifurcation; Stenosis; Imaging; 3D Reconstruction And Mesh Generation; Doppler Ultrasound.

Abstract. A methodology for computational 3D reconstruction and structured hexahedral meshing for patient-specific hemodynamics analysis of the carotid artery bifurcation with a stenosis is presented. The purpose of this work is the use of anatomically realistic blood flow simulations by the finite element method (FEM) derived from in vivo medical imaging to make patient specific studies of flow phenomena associated with the development of atherosclerosis disease. Blood flow is described by the incompressible Navier-Stokes equations and the simulation is carried out under pulsatile conditions. The study of a diseased carotid bifurcation illustrates the extremely complex hemodynamical behaviour along the cardiac cycle.

1 INTRODUCTION

A long standing hypothesis that correlates fluid dynamic forces and atherosclerotic disease has led to numerous analytical, numerical, and experimental studies over the years. The observation that atherosclerotic disease is focal typically occurring at sites of complex hemodynamics, such as arterial bifurcations, junctions, or regions of high curvature inspired these studies. High wall shear stress (WSS), damage the arterial wall and regions of low or oscillatory shear stress cause monocyte adhesion to the endothelium, an early stage in

atherogenesis [1]. The carotid bulb is one of the first sites in the carotid bifurcation to develop late atherosclerotic inflammation consistent with reports that show this to be a region of low WSS. Usually flow separation occurs at the carotid bulb due to the increase in cross-sectional area, being this region of separation greater during deceleration phase of systole, when the fluid undergoes the largest reversal of momentum.

Recent non-invasive medical imaging data acquisition made feasible to construct three dimensional models of blood vessels. B-mode ultrasound is a non-invasive method of examining the intima and walls of peripheral arteries providing measures of the intima-media thickness (IMT) at various sites (common carotid artery, bifurcation, internal carotid artery) and of plaques that may indicate early presymptomatic disease. It also allows measurements of blood flow velocities providing accurate information on flow fields. Validated computational fluid dynamics models using data obtained by these currently available measurement techniques can be very valuable in the early detection of vessels at risk and prediction of future disease progression.

Computational modelling of blood flow in realistic arterial geometries has the potential to provide a complete set of hemodynamic data that cannot be acquired by measurement alone. This needs to be performed by combining the latest computational fluid dynamics approaches with the velocity measurements and flow images obtained using ultrasound techniques.

In this work flow characteristics in a patient-specific carotid bifurcation with a stenosis are investigated by using direct numerical simulation. A semi-automatic methodology for patient-specific reconstruction and structured meshing of the right carotid bifurcation is presented. As hexahedral meshes compared to tetrahedral/prismatic meshes converge better, and for the same accuracy of the result less computational time is required [2-4] a tool to generate suitable structured hexahedral meshes for vascular modelling frameworks from Doppler ultrasound images is considered.

Blood flow simulation models [5] using pulsatile inlet conditions based on in vivo colour Doppler ultrasound measurements of blood velocity, allow to compare numerical results with experimental data collected in clinical practice. The three-dimensional, unsteady, incompressible Navier–Stokes equations are solved with the assumptions of rigid vessel walls and constant viscosity (Newtonian fluid).

The ultimate aim of this study is the reconstruction of geometry and flow environment from in-vivo patient data, particularly at the extra-cranial carotid artery, using Doppler ultrasound data.

2 METHODOLOGY

To perform the computational investigation of patient specific arterial morphology and blood flow behaviour using a finite element code four steps are necessary: acquiring the in vivo anatomical data of the arterial segment, image surface reconstruction, 3D finite element mesh definition and blood flow simulation. Data was obtained in Hospital de São João, a university hospital in Oporto, Portugal. Informed consent of each volunteer was obtained using a protocol for the acquisition of a set of longitudinal and sequential transverse Doppler images and velocity measurements at carotid artery bifurcation. Using a standard commercial colour ultrasound scanner (General Electric vivid e) a set of longitudinal and transversal B-

mode images of the common carotid artery (CCA), its bifurcation and proximal segments of internal carotid artery (ICA) and external carotid artery (ECA) of a patient was acquired This set of images cover the bifurcation region, about 10 cm long.

For each volunteer velocity measurements were made at different locations in the common, internal and external carotid arteries allowing the definition of the boundary conditions and the validation of the blood flow simulation.

The acquired Doppler ultrasound images made possible the 3D geometry reconstruction and mesh generation. A selected 2D longitudinal image shown in figure 1 was manually segmented by three medical experts and a rough outline of the intima-media region boundaries was defined and imported into the modeling commercial software FEMAP (FEMAP, Siemens PLM, USA & Canada).
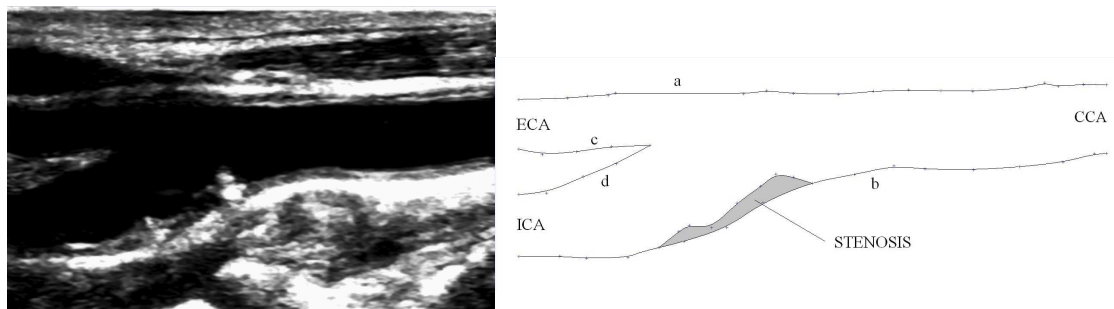
Figure 1: Carotid arterybifurcation of patient P1: (a) input image, (b) estimated boundaries

A computational 3D geometry reconstruction and a structured hexahedral mesh of the lumen were constructed. The centerlines of CCA, ECA and ICA were defined at the adopted mid-plane by creating a curve associated to equidistant points from splines a to b, a to c and d to b. Then, cylindrical geometries are assumed for CCA, ECA and ICA, except at their junctions, and their cross-sections modified according to the drawn lumen boundary and acquired cross-sectional ultrasound images. Artery surfaces are defined as vessels presenting curved axes and cross-sectional shape and diameter variability. As shown in figure 2 cross-sections of internal carotid artery and external carotid artery junctions are the result of overlapped cross-sections, defining non-circular sections.
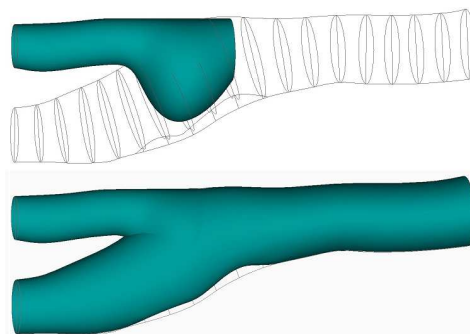

Figure 2: Definition of carotid bifurcation surface

Using software FEMAP, mesh generation of the previously defined surface was performed by dividing the domain in six parts being each part meshed independently and maintaining finite elements continuity at each contact surface as shown in Figure 3. The generation of the volume mesh with hexahedral elements started by building a 2D mesh (quadrilateral) on the confining cross-sections defined at the bifurcation, as artificial separations of CCA, ECA and ICA branches. Then CCA, ECA and ICA branches are treated independently by sweeping or extruding the 2D mesh in order to generate a volume mesh of hexahedrons.
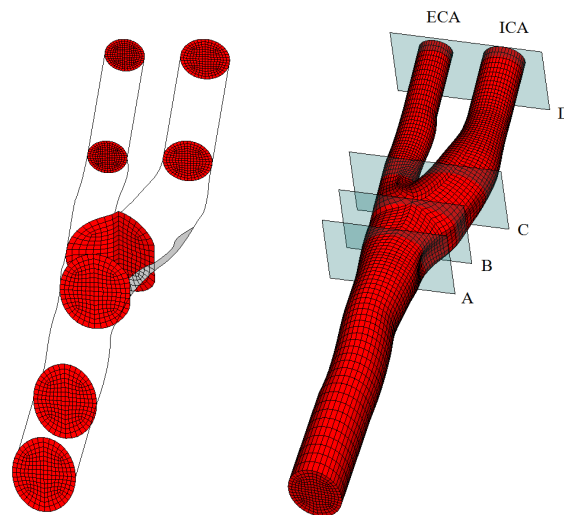
Figure 3: Structured mesh and cross-section slices for spatial velocity variation A–A, B–B, C–C and D–D (figure 4).

The use of computational meshes with well-organized elements along the main flow direction assures faster convergence and more accurate numerical solutions as blood motion in vessels is highly directional [3-5]. More accurate solutions are also obtained with a finer mesh in the bifurcation, near stenosis and near wall regions.
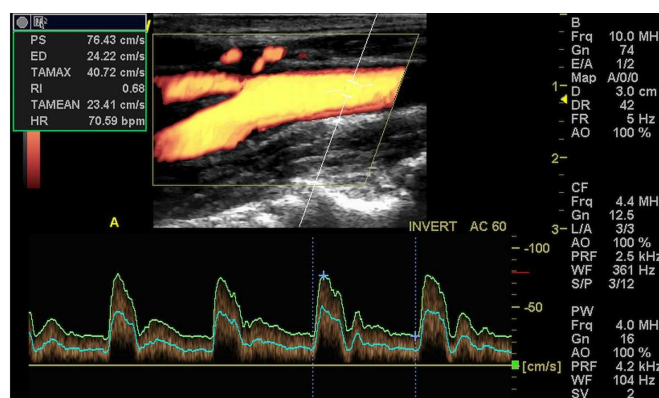
Figure 3: Measured flow wave form in the common carotid distal the flow divider

It is desirable to impose boundary conditions a few diameters upstream and downstream the region of interest therefore the polygonal surface obtained is not directly usable for generating a suitable computational mesh. Cylindrical 4-diameter extensions are added in the direction of the centerlines in order to reflect the vessels geometry as it approaches the domain of interest.

With the inlet flow corresponding to the systolic peak a mesh sensitivity analysis was carried out under steady conditions. Mesh refinement was performed until changes in velocities and maximum nodal WSS became less than 1.5%; a mesh of 55 thousand hexahedrons was chosen and a temporal convergence was performed with a temporal refinement until changes in velocities and maximum nodal WSS became insignificant between the adopted and finer time steps (less than 1.5%). A large amount of computational work is involved and with the chosen mesh the transient study was performed with a constant time step equal to 2.5×10^{-3}s.

3 RESULTS AND DISCUSSION

The accuracy and efficiency of the blood simulation is validated comparing velocities given by numerical calculations with experimental data collected in clinical practice. In figure 5 velocities are compared with Doppler ultrasound measurements at different cross-section locations. At all positions in the carotid bifurcation there is a good agreement between the obtained flow velocities and those obtained experimentally in clinical practice as deviations are less than 5% at all positions.
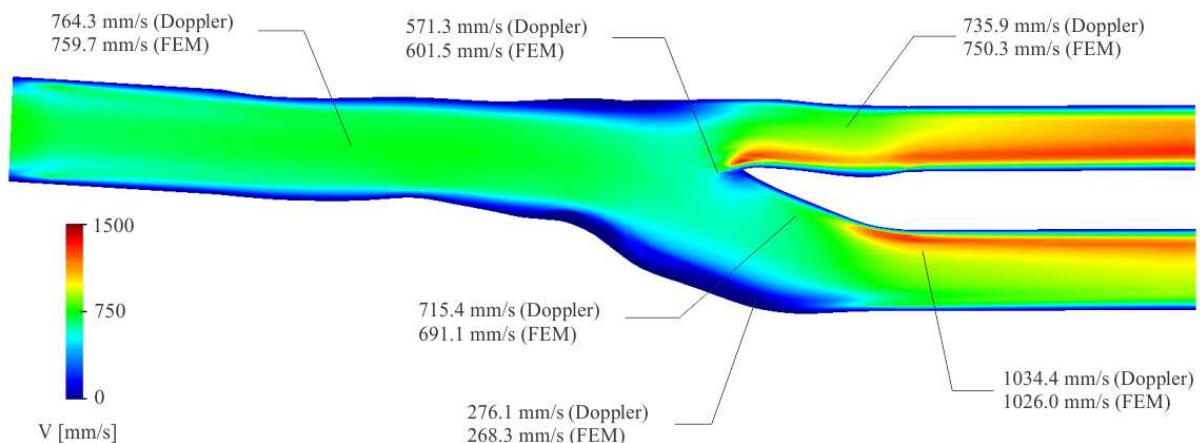

Figure 5: Velocity field in the mid plane of the bifurcation at systolic phase: Power-Doppler measurements and calculated velocities

Numerical velocity field in some sections is presented in Figure 6, for two cardiac phases, near peak systolic and at mid-deceleration phase. Within the stenosis, section A-A, no large velocity gradients can be noticed as there is no variation in lumen section. The increase in lumen section of the carotid bulb just downstream the stenosis contributes to the tendency for flow separation observed in sections B-B and C-C. This recirculation zone, downstream the stenosis is higher during deceleration phase due to the fact that flow parthens change drastically during this phase of the cardiac cycle. In the external carotid artery high velocity

gradients occur in section C-C due to the apex-induced separation; high velocity gradients can also be observed in section D-D for both internal and external carotid arteries as the blood vessel diameter decreases significantly.
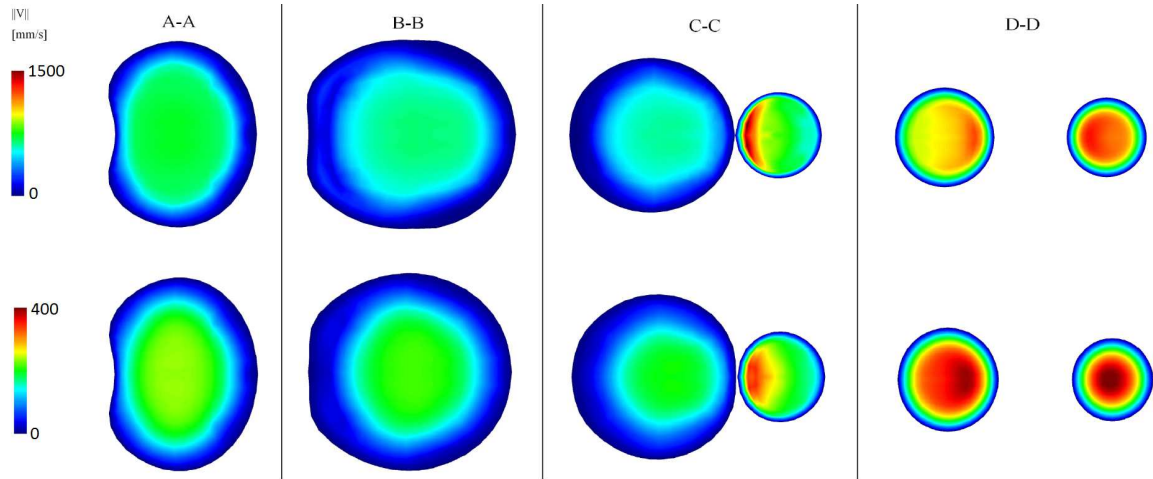

Figure 6: Velocity distribution at four sections near peak systolic and at mid-deceleration phase (section locations in figure 3).

In order to elucidate the role of carotid hemodynamics on plaque vulnerability WSS distribution at the two same flow phases is studied. In Figure 7 WSS distribution near peak systolic time shows high shear stress at the inner wall of the ECA, corresponding to high velocity-gradients with a skewed velocity profile. However the highest values are found at the inner wall of ICA, downstream the stenosis, due to the reduction of the lumen diameter.
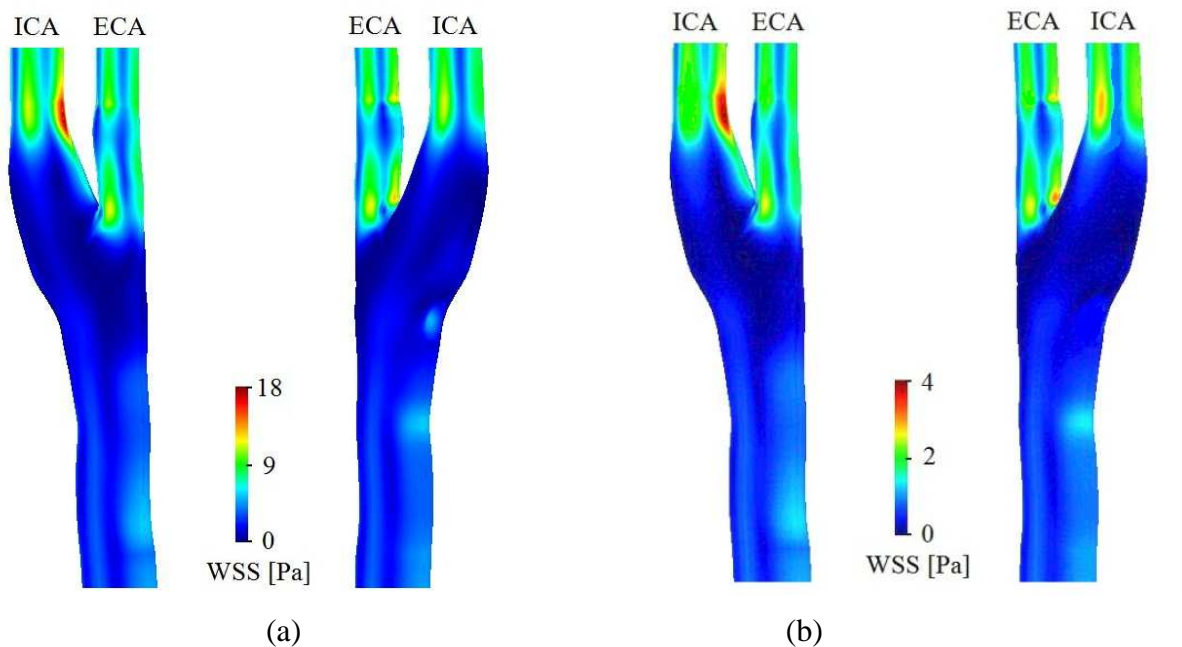

(a)   (b)

Figure 7: WSS distribution at two flow phases: (a) near peak systolic, and (b) diastole.

Recirculation zones are characterized by patches of low WSS in ICA, where the largest regions are associated with the apex-induced separation near the outer sinus wall and with the stenosis-induced separation along inner and outer walls. At diastolic time high shear stress zones can also be observed, however patterns at the two time instants are different providing further evidence that flow patterns change drastically during the cardiac cycle. One striking similarity prevails between both instants: the largest continuous region of low WSS seems to exist in the carotid bulb, where late atherosclerotic inflammation develops.

This work addresses the hemodynamical environment of a diseased carotid bifurcation concluding to be extremely complex during systolic phase and different from that at diastolic phase.

ACKNOWLEDGMENTS

REFERENCES

[1] De Paola N, Gimbrone Jr. MA, Davies PF, Dewey Jr. CF. Vascular endothelium responds to fluid shear stress gradients. Arteriosclerosis Thrombosis 1992; 12:1254–1257.

[2] De Santis, G., Mortier, P., De Beule, M., Segers, P., Verdonck, P., Verhegghe, B. Patient-specific computational fluid dynamics: structured mesh generation from coronary angiography. *Med. Biol. Eng. Comput.* (1010) **48**(4):371-80.

[3] Antiga L., Piccinelli M., Botti L., Ene-Iordache B., Remuzzi A., Steinman D. An image-based modeling framework for patient-specific computational hemodynamics. *Med. Biol. Eng. Comput.* (2008) **46**:1097-1112.

[4] Verma C.G., Singh C, Fischer P., Lee S., Loth F. An All-Hex Meshing Strategy for Bifurcation Geometries in Vascular Flow Simulation. Proceedings, *14th International Meshing Roundtable*, USA. Springer-Verlag (2005) 11-14:363-376.

[5] Sousa L. C., Castro C.F., António C.C. and Chaves R. Blood flow simulation and vascular reconstruction. *Journal of Biomechanics* (2012) **45**:2549-2555, DOI information: 10.1016/j.jbiomech.2012.07.033S.R.

NUMERICAL INTEGRATION OF WEAK FORM IN EMBEDDED INTERFACE METHODS

Y. Sudhakar[*], Wolfgang A. Wall[*] and J. P. Moitinho de Almeida[†]

[*]Institute for Computational Mechanics, Technische Universität München
Boltzmannstrasse 15, 85747 Garching, Germany. e-mail: {sudhakar,wall}@lnm.mw.tum.de

[†]Instituto Superior Técnico, Universidade Técnica de Lisboa
Av. Rovisco Pais, 1049-001 Lisboa, Portugal. e-mail: moitinho@civil.ist.utl.pt

Key words: Divergence theorem, embedded interface methods, integration over polyhedra, XFEM

Abstract. The numerical integration of weak form over the elements that are crossed by discontinuities in embedded interface methods is addressed in this work. Since these methods lead to complex shaped cut volumes, integration of weak form requires an efficient method for integration of polynomials over arbitrary polyhedra. Most widely, volume decomposition [1] or moment fitting methods [2] are used for such integrations. In this work, we present an efficient and robust method, based on the divergence theorem, for integration of polynomials over polyhedra. For a scalar function $\mathcal{F}$, using the divergence theorem, the integration over $\mathcal{R} \subset \mathbb{R}^3$ whose boundary is given by $\mathcal{S}$ can be written as,

$$\int_{\mathcal{R}} \mathcal{F}dV = \int_{\mathcal{S}} \mathcal{G}n_x \, dA; \quad where \quad \mathcal{G} = \int_{\kappa}^{\mathcal{S}} \mathcal{F}dx \tag{1}$$

where κ is an arbitrary reference point. $\mathcal{G}$ is evaluated by integrating $\mathcal{F}$ using one-dimensional Gauss quadrature, and then to compute the required integral, $\mathcal{G}$ is integrated using another set of Gauss quadratures defined on surfaces of the polyhedra. The method is extremely easy to implement, and we show through numerical examples that it is efficient as well.

REFERENCES

[1] Sukumar, N. Möes, N. Moran, B. and Belytschko, T. Extended finite element method for three-dimensional crack modelling. *Int. J. Num. Meth. Engng.* (2000) **48**:1549–1570.

[2] Sudhakar, Y. and Wall, W.A. Quadrature schemes for arbitrary convex/concave volumes and integration of weak form in enriched partition of unity methods. *Comput. Methods. Appl. Mech. Eng.*, Submitted.

VI International Conference on Adaptive Modeling and Simulation
ADMOS 2013
J. P. Moitinho de Almeida, P. Díez, C. Tiago and N. Parés (Eds)

INNOVATION TOOL USING TAGUCHI-METHODS FOR DEVELOPMENT OF A NEW PRODUCT WITH OPTIMUM CONDITION

IKUO TANABE [*]

[*] Nagaoka University of Technology (NUT)
Department of Mechanical Enginnering
1603-1 Kamitomioka, Nagaoka, Niigata, 940-2188 JAPAN
e-mail: tanabe@mech.nagaokaut.ac.jp

Key words: Taguchi-methods, Tool, Optimum Condition, Innovation, Development, Trial

Abstract. As a development with short-term and lower cost are strongly required in 21st century. Therefore the innovation tool using Taguchi-methods for development of a new product with optimum condition was developed and evaluated. There are two trials in the innovation tool using Taguchi-methods. First trial investigates rough fuctions regarding all levers of all control factors, then important control factors and meaningless control factors are sorted. Second trial decides the optimum combination of the control factors by more detail trial using only important control factors. The optimum condition for polishing a minute die was investigated for evaluating this innovation tool in the experiment. It is concluded from the result that (1) Innovation tool using the Taguchi-methods was useful for development with short-term and lower cost, and (2) This tool could quickly and exactly decide the optimum polishing condition.

1 INTRODUCTION

Recently a developments with short-term and lower cost are strongly required for shorten products life cycle. Therefore FEM simulation is used for predicting the result of design process instead of doing experiments. On the other hand, Taguchi-methods [1], [2], [3], [4], [5] is also used for deciding optimum process conditions. However these methods are not enough to develop a new product with short time, lower cost, high quality and high accuracy.
In this study, the innovation tool using Taguchi-methods for development of a new product with optimum condition was developed and evaluated. The Taguchi-methods has several properties; the selected optimum combination for the control factors has very high robustness, influences for the each level of the control factors were shown by the each SN ratio and Sensitivity with decibel unit. Therefore the Taguchi-methods was used for the innvovation tool. In this research, there are two trials using the innovation tool; the first trial investigates rough fuctions regarding all levers of all control factors, and important control factors and meaningless control factors were sorted by using the results of the first trial. Second trial decides the optimum combination of the control factors by more detail trial using only important control factors. The optimum condition for polishing a minute die was investigated

for evaluating this innovation tool in the experiment. This new method will be more proper than the conventional Taguchi-methods [6] or other methods [7], [8] for searching the optimum condition.

2 EXPLANATION OF TAGUCHI-METHODS

Flow chart regarding explanation of the Taguchi-methods for products is shown in Figure 1. Taguchi-methods is used to decide optimum processing conditions with narrow dispersion for robust design. Control factors are equal to the design factors (See the control factors in Table 1). Noise factors are occurred for the error of function on the product (See the noise factors in Table1). Most designer can understand that the final functions of the developed product are strongly influenced for the each lever of each control factor under several noise factors. All combinations using all control factors are compressed by an orthogonal table (See the orthogonal array in Table 2). Then the experiment or the CAE analysis with influence of noise factors is performed by the orthogonal table. At last, the average and the standard deviation regarding all combinations using all parameters are calculated for the SN ratio and Sensitivity.

Then most of users write the effective figure of the control factors and zealously search the combination of the control factors for large SN ratio. A product using the combination isn't nearly influenced by noise factors. Specifically decision of optimum combination using the parameters for high robustness was completely finished.

Properties regarding the Taguchi-methods were that the selected optimum combination for the control factors has very high robustness, and influences for the each level of the control factors were shown by the each SN ratio and Sensitivity with decibel unit. Therefore most designers can selected the optimum lever of each control factor by checking the effective figure of the control factors for both SN ratio and Sensitivity.

The control and the noise factors are shown in Table 1. These factors are important factors in the Taguchi-methods. Each factor has several levels. The control factors are equal to the design factors. Noise factors relate to the error of function with regard to the product.
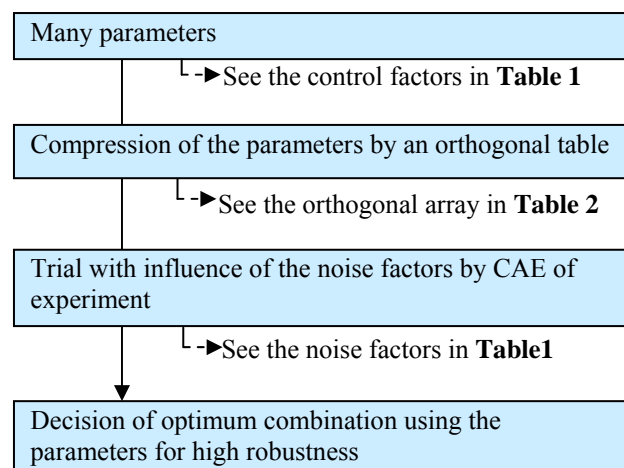

Figure 1 : Flow-chart of the Taguchi-methods

Table 1: Control factors and noise factors in the Taguchi-methods

Control factors				
Name	A	B	C	D
Levels	A_1	B_1	C_1	D_1
	A_2	B_2	C_2	D_2
	A_3	B_3	C_3	D_3
Noise factors				
Name	N			
Levels	N_1		N_2	N_3

Table 2 : Orthogonal array, SN ratio and sensitivity in the Taguchi-methods

Trial No.	Control factors				Result with noise factors			SN ratio (db)	Sensitivity (db)
	A	B	C	D	N_1	N_2	N_3		
1	A_1	B_1	C_1	D_1	2.7	2.6	2.4	24.5	8.2
2	A_1	B_2	C_2	D_2	2.3	2.2	2.0	23.0	6.7
3	A_1	B_3	C_3	D_3	2.1	1.9	2.0	26.0	6.0
4	A_2	B_1	C_2	D_3	3.3	3.1	3.0	26.2	9.9
5	A_2	B_2	C_3	D_1	4.6	4.4	4.5	33.1	13.1
6	A_2	B_3	C_1	D_2	3.3	3.3	3.0	25.3	10.1
7	A_3	B_1	C_3	D_2	2.1	2.3	2.4	23.4	7.1
8	A_3	B_2	C_1	D_3	3.1	3.2	3.1	34.7	9.9
9	A_3	B_3	C_2	D_1	4.7	5.1	4.9	27.8	13.8

The total of all combinations using all control factors is 81 (=3^4) kinds (Table 1); however, these combinations are compressed to 9 kinds in the orthogonal table (Table 2). The influence of the noise factors is investigated 3 times for each combination of the control factors. Therefore the number of trial in this case is 27 (= 9 kinds×3 times). SN ratio and sensitivity are calculated by equations (1) and (2).

$$SN\ ratio\ (db) = 10\ log\ (\mu^2 \diagup \sigma^2) \tag{1}$$

$$Sensitivity\ (db) = 10\ log\ \mu^2 \tag{2}$$

Where μ is average of the evaluation value, and σ is standard deviation of the evaluation value in the results of trial. The evaluation value is the final properties or the final functions. Then most of users write the effective figure (Figure 2) of the control factors and search the combination of the control factors for the largest SN ratio. At that time, a product using the combination is not nearly influenced by noise factors. These effective figures are used for decision of optimum combination using several parameters in design of a product with high
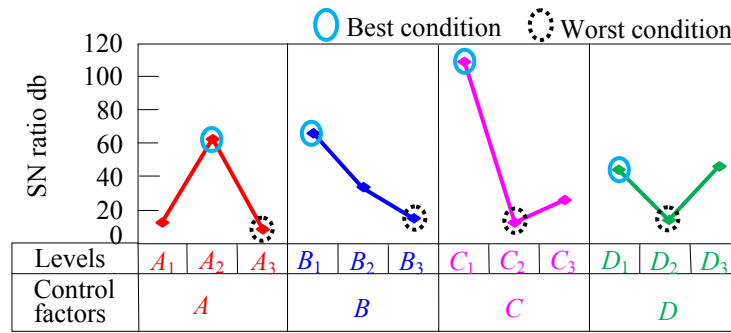
(a) Effective figure for SN ration
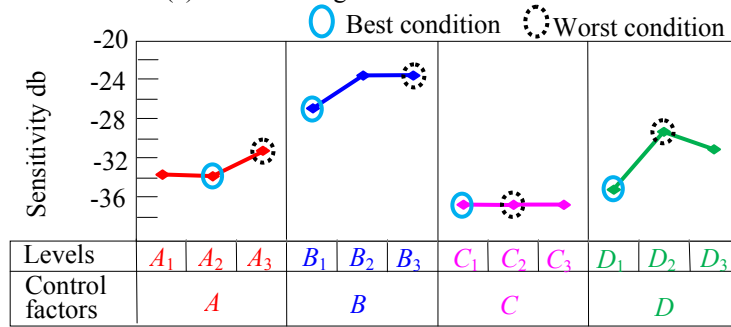
(b) Effective figure for Sensitivity

Figure 2 : Relationship between SN ratio or Sensitivity and each lever of each control factor
(In the case, the best condition was supposed at the smallest final function possible)

robustness. When the SN ratio becomes large, the distribution of the final functions becomes small and the product has high robustness.

Finally, the average and the standard deviation regarding all combinations using all parameters are calculated by the SN ratio and Sensitivity with respect to the 9 kinds. The addition theorem in the Taguchi-methods is used to calculate the results for all combinations. For example, when m is a control factor and n is the level for the factor, the SN ratio SN_{mn} and Sensitivity S_{mn} for the control factor m and the level n are calculated by the addition theorem. Moreover the SN ratio $SN_{a4 \cdot b2 \cdot c1 \cdot d3 \cdot e2 \cdot f1 \cdot g2}$ and the Sensitivity $S_{a4 \cdot b2 \cdot c1 \cdot d3 \cdot e2 \cdot f1 \cdot g2}$ for a4, b2, c1, d3, e2, f1, g2 using control factors (a, b, c, d, e, f, and g) and levels (1, 2, 3, 4, 5 and 6) are calculated by equations (3) and (4), respectively.

$$SN_{a4 \cdot b2 \cdot c1 \cdot d3 \cdot e2 \cdot f1 \cdot g2} = SN_{a4} + SN_{b2} + SN_{c1} + SN_{d3} + SN_{e2}$$
$$+ SN_{f1} + SN_{g2} - (7-1)\ SN_{ave} \quad (3)$$

$$S_{a4 \cdot b2 \cdot c1 \cdot d3 \cdot e2 \cdot f1 \cdot g2} = S_{a4} + S_{b2} + S_{c1} + S_{d3} + S_{e2} + S_{f1} + S_{g2}$$
$$- (7-1)\ S_{ave} \quad (4)$$

Where SN_{ave} and S_{ave} are averages of all SN ratios and Sensitivities, respectively. SN ratios and Sensitivities of the final properties or the final functions for all combinations of all control factors are quickly estimated before the trials.

3　INNOVATION TOOL USING TAGUCHI-METHODS

There are several strong points in the Taguchi-methods. Trial combinations are compressed to small size by using the orthogonal table, then the Taguchi-methods can estimate the final results for all combinations. Therefore the development with high quality is quickly performed.

Everyone can check influences for all levers on all control factors in the effective figure (Figure 2). SN ratio is used for robustness of the final properties or the final functions, and Sensitivity is used for evaluation regarding the final properties or the final functions. The final properties or the final functions have the several influences of the noise factors. Therefore the trial using the Taguchi-methods can arrive at the results with high robustness.

Flow-chart of the innovation tool using the Taguchi-methods is shown in Figure 3. The tool consists of two trials using the Taguchi-methods; these are "First trial for selection of the several important parameters" and "Second trial for decision of the optimum condition". In the First trial, all levers of all control factors in your laboratory should try for the final properties or the final functions. This trial is for picking out the important parameters and for throwing away the meaningless parameters. If difference of influence on the each level regarding a control factor in the effective figure of "the Sensitive" is very little, the control factor is judged to the meaningless parameter. And when SN ratio is very small, the level of the control factor is judged to low robustness. Only important parameters selected in the First trial are used in the Second trial. In this trial, each important parameter are checked in more detail. If the important parameters require the larger or smaller level of a control factor for optimum condition, the new equipment for the larger or smaller level of a control factor is supplied in here. And if the important parameters require the level with high precision of a control factor for optimum condition, the new equipment with high precision is also then supplied in here. This second trial becomes the final trial, because optimum condition is decided by the second trial using Innovation Tool using Taguchi-methods with the best condition in the laboratory.
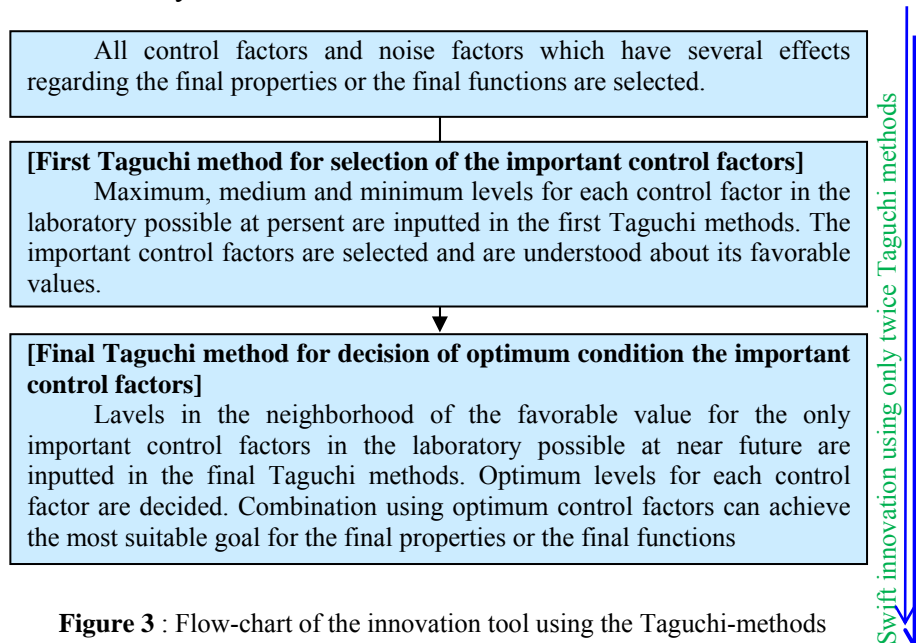
All control factors and noise factors which have several effects regarding the final properties or the final functions are selected.

[First Taguchi method for selection of the important control factors]
Maximum, medium and minimum levels for each control factor in the laboratory possible at persent are inputted in the first Taguchi methods. The important control factors are selected and are understood about its favorable values.

[Final Taguchi method for decision of optimum condition the important control factors]
Levels in the neighborhood of the favorable value for the only important control factors in the laboratory possible at near future are inputted in the final Taguchi methods. Optimum levels for each control factor are decided. Combination using optimum control factors can achieve the most suitable goal for the final properties or the final functions

Swift innovation using only twice Taguchi methods

Figure 3 : Flow-chart of the innovation tool using the Taguchi-methods

4 EVALUATION USING THE POLISHING TOOL

4.1 First trial for selection of the several important parameters regarding the polishing tool

Optimum polishing condition is investigated for evalguchi-methods. Polishing tool and polishing procedure are shown in Figures 4 and 5, respectively[9]. This poliuation of the Innovation Tool using Tashing tool consists of the pipe and the ball head with diamond grains. Base material of the ball head is epoxy resin. Slurry consists of water, a polymer and diamond grains. The polishing tool is installed on the spindle of CNC milling machine, is rotating and moving in three dimensional directions by NC control. Several diamond grains in the polishing head and in the slurry can cut on the work piece. The polishing trace becomes very shallow because of soft ball head. However surface roughness of the work piece becomes
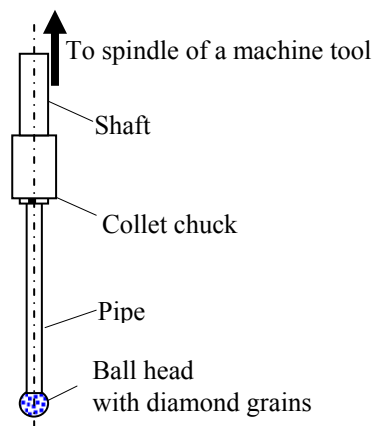
Figure 4: Schematic view of the polishing tool
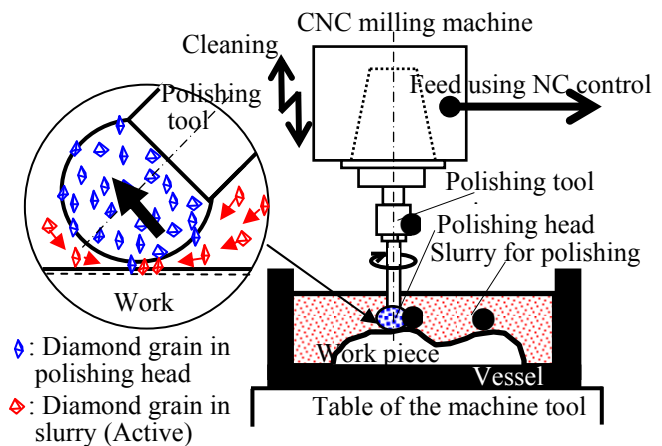
Figure 5: Schematic view of polishing (Principle)

Table 3: Control factors and noise factors for the polishing in the First trial

	Control factors				
Name	Spindle speed min⁻¹	Feed speed mm/min	Polishing pressure MPa	Polishing pitch mm	Room Temp. ℃
Levels	300	0.5	60	0.030	20
	2400	1.0	120	0.045	25
	8000	5.0	180	0.060	30
	Noise factors				
Name	Measuring position				Point 1 · · · Point 2 Point 3 · · · ←5×5mm Point 4 · · · Point 5
Levels	Point 1	Point 2	Point 3	Point 4	Point 5

647

(a) Effective figure for SN ration

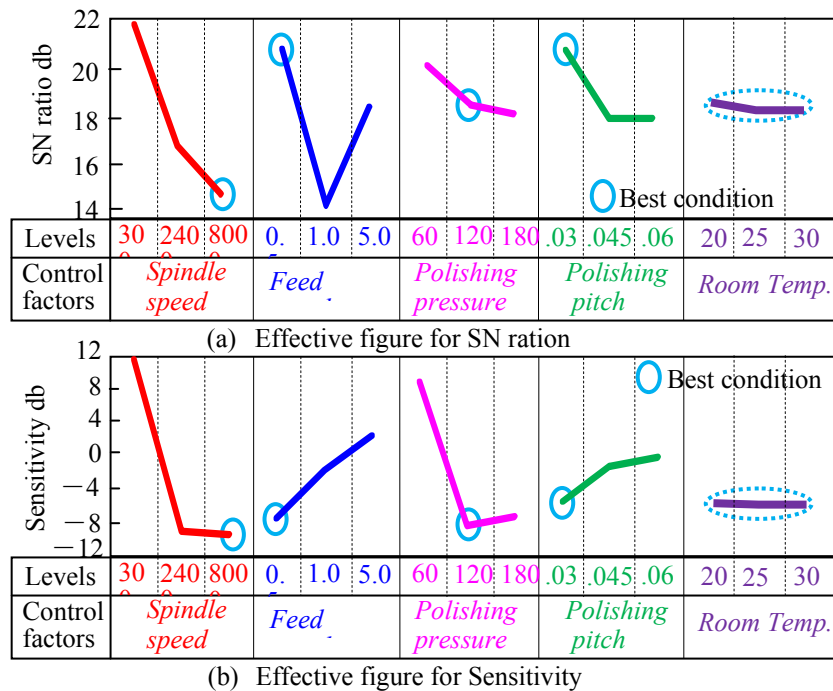(b) Effective figure for Sensitivity
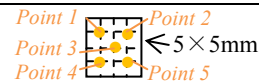
Figure 6 : Relationship between SN ratio or Sensitivity and
each lever of each control factor in the First trial

very small because of shallow trace. After all, the polishing tool can polish to mirror-like surface. Particularly the ball head of the polishing tool has small diameter which is smaller than 1 mm. Therefore the polishing tool can polish a minute die.

Control and noise factors for the first trial are shown in Table 3. Control factors are several polishing conditions. These control factors are experimental parameters as thoroughly as possible I could in my laboratory, and the levels of the each control factor are maxmum, minimum and middle values. Namely we can try all possibility for picking out the important parameters and for throwing away the meaningless parameters. Several measuring points are used for the noise factor. Surface roughness after polishing is used for the final function.
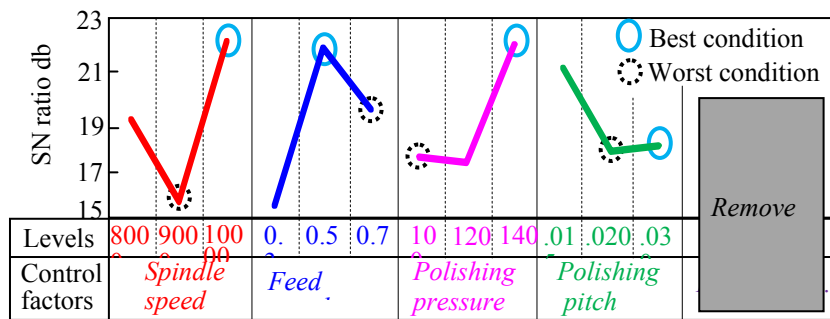
The effective figures for SN ratio and Sensitivity in the First trial are shown in Figure 6. Spindle speed, feed speed, polishing pressure and polishing pitch were selected for the important parameters by the effective figure, and the room temperature was thrown away from the control factors because of the little differences on influence of the each level regarding the room temperature in the effective figure of Sensitive. Optimum condition is that spindle speed was 8000 min[-1], feed speed was 0.5 mm/min, polishing pressure was 120 MPa and polishing pitch was 0.03 mm in the First trial. Therefore these control factors are investigated in more detail in the next final trial.

4.2 Second trial for decision of the optimum condition regarding polishing tool

Control and noise factors for the second trial are shown in Table 4. The only important parameters in the first trial are used for the control factors in the second trial. These control

Table 4: Control factors and noise factors for the polishing in the Second trial

Control factors				
Name	Spindle speed min⁻¹	Feed speed mm/min	Polishing pressure MPa	Polishing pitch mm
Levels	8000	0.3	100	0.015
	9000	0.5	120	0.020
	10000	0.7	140	0.030
Noise factors				
Name	Measuring position			Point 1 ── Point 2
Levels	Point 1 Point 2 Point 3 Point 4 Point 5			Point 3 ←5×5mm Point 4 ── Point 5

(a) Effective figure for SN ration
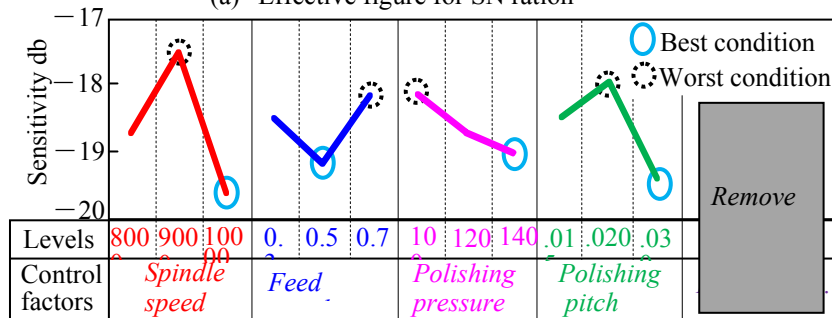
(b) Effective figure for Sensitivity

Figure 7 : Relationship between SN ratio or Sensitivity and each lever of each control factor in the Second trial

factors have large influence for surface roughness, and the levels of the each control factor are established nearby the optimum condition of the first trial. The effective figures for SN ratio and Sensitivity in the Second trial are shown in Figure 7. The optimum condition is that spindle speed was 10000 min-1, feed speed was 0.5mm/min, polishing pressure was 140 MPa and polishing pitch was 0.030mm in the Second trial. The SN ratio becomes very large and the Sensitivity becomes very small at the optimum condition. At that time, this polishing can make smooth and fine surface and the polishing has very high robustness. And average and standard deviation of surface roughness at best and worst polishing condition were also estimated by equations (3) and (4).

4.3 Evaluation regarding decision of the optimum condition for polishing tool

The optimum condition for polishing tool is evaluated in the experiment. Polishing condition used in the experiment is shown in Table 5. Work piece material is carbide. Specifications of the polishing tool and the slurry are similar to the previous experiment. Best and worst conditions in the Second trial are included for the polishing conditions.

Surface roughness of the polishing with best and worst conditions is shown in Figure 8. The results of the experiment are similar to the calculated results by the Innovation Tool using Taguchi-methods. The optimum condition for polishing tool was decided by only twice trials. Therefore the Innovation tool using the Taguchi-methods was useful for development with short-term and lower cost.

Table 5: Best and worst conditions for the polishing

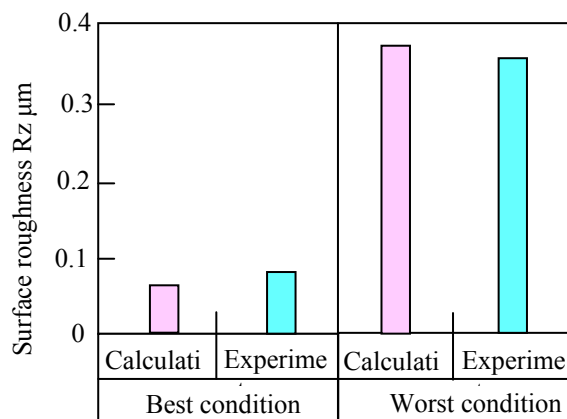| Polishing condition		Best condition	Worst condition
Spindle speed min^{-1}		10000	9000
Feed speed mm/min		0.5	0.7
Polishing pressure MPa		140	100
Polishing pitch mm		0.3	0.2
Polishing tool	Material of polishing head (Ball head)	Epoxy resin	
	Diameter of polishing head (Ball head)	ϕ 1.0 mm	
	Diamond grain in ball head (♯=Mesh size)	#2500	
	Pipe	0.7 mm	
Slurry	Base liquid	Water	
	Diamond grain in slurry (♯=Mesh size)	#2500	
	Rate of grain (slurry : diamond)	10 wt% (9:1)	
	Ratio of PEO（Poly-ethylene-oxide） for water	2 wt%	
Work piece		Carbide	

Figure 8 : Surface roughness of the polishing with best and worst conditions (By the Innovation Tool using Taguchi-methods)

5   CONCLUSIONS

- The innovation tool using Taguchi-methods was useful for decision of optimum condition.
- The proposed method effectively predicted the optimum polishing condition in experiment for evaluation.
- The predicted results conformed to the results of the actual polishing.

REFERENCES

[1] Makino, T., Optimization of Exhaust Port using Computer Simulation, *Proceedings of the 13th Quality Engineering Society Conference* (2005): 6-9

[2] Fujikawa, S., Optimum Parameter Design using the Taguchi-methods for Finite-Element Analysis of 3D Forging Deformation, *Journal of the Japan Society of Technology, for Plasticity*, Vol. 40, No. 466 (1999-11): 1061-1065

[3] Tatebayashi, K., Computer Aided Engineering Combined with Taguchi-methods, *Proceeding of the 2005 Annual Meeting of the Japan Society of Mechanical Engineering*, No. 05-1 Vol. 8 (2005-9): 224-225

[4] Nagakura, S., et al., *Iwanami Physics and Chemistry Dictionary* (in Japanese) (1988), Iwanami Shoten: 1234

[5] Sugai, H., et al., Prediction of Optimum Machining Condition in Press Forming Using Taguchi-methods and FEM Simulation, Transactions of the JSME (in Japanese), Vol.72 No.721 (2006): 3044-3051

[6] MIURA, K., Optimization of Clamping Mechanism of Injection Molding Machine Using Simulation,   Journal of Quality Engineering Society (in Japanese), Vol.16 No1, (2008): 78-84

[7] SVATEK M., MARTINEC J. and PROVAZNIKOVA M., Optimization of Continuous Chemical   Production, *Proceedings of the 10th International Conference on MITIP* (*Modern Information Technology in the Innovation Processes of the Industrial Enterprises*) , (2008):62-67

[8] CESAROTTI V., SILVIO B. D. and INTRONA V., Optimizing Control Parameters of Industrial Processes   with Attribute Response Through Design of Experiments, *Proceedings of the 10th International Conference on MITIP* (*Modern Information Technology in the Innovation Processes of the Industrial Enterprises*) , (2008):32-37

[9] IYAMA T., TANABE I., Optimization of Lapping Slurry in Automatic Lapping System for Dies with Cemented Carbide and Its Evaluation, Transactions of the JSME (in Japanese), Vol.75 No.749, (2009): 210-215

ANISOTROPIC ADAPTIVE NEARLY BODY-FITTED MESHES FOR CFD

T. TOULORGE[*], D.-L. QUAN[*], E. MARCHANDISE[*] AND J.-F. REMACLE[*]

[*]Université catholique de Louvain (UCL)
Institute of Mechanics, Materials and Civil Engineering (iMMC)
Avenue G. Lemaitre, 4
B-1348 Louvain-la-neuve, BELGIUM
e-mail: thomas.toulorge@uclouvain.be

Key words: anisotropic adaptive, embedded interfaces, level-set, computational fluid dynamics, incompressible

Abstract. This paper presents a numerical study of a recent technique that consists in modeling embedded geometries by a level-set representation in combination with local anisotropic mesh refinement. This method proves beneficial in CFD simulations involving complex geometries, as it suppresses the need for the tedious process of body-fitted mesh generation, without altering the finite element formulation nor the prescription of boundary conditions. The first part of the study deals with a simple Laplace problem featuring a planar interface on which a Dirichlet boundary condition is imposed. It is shown that the appropriate amount of local isotropic refinement yields the optimal convergence, unlike uniform refinement. Anisotropic refinement further ensures geometric convergence and limits the growth of the number of unknowns. The second part deals with the adaptive strategy for CFD problems. We show that the methodology yields accurate flow solutions, despite very limited user interaction.

1 INTRODUCTION

Because of the increasingly complex geometries involved in flow problems of industrial relevance, numerical methods based on unstructured meshes have become popular in CFD. However, the corresponding meshing methods require a high-quality CAD description of the geometry, which is not part of the traditional workflow in fields like architecture or medicine. Many professionals also lack the expertise required to build appropriate meshes for flow problems. Nevertheless, recent progresses in meshing technology could overcome these barriers.

In this work, we use anisotropic adaption to generate a nearly body-fitted mesh. The mesh is locally refined depending on a level-set function that describes the geometry

without resorting to a CAD model [1, 2]. Dirichlet boundary conditions can then be imposed in a strong manner by node collocation, just as with classical body-fitted meshes. Unlike other treatments of embedded geometries, this technique only requires a standard finite element formulation, without basis enrichment or Lagrange multipliers that alter its numerical properties.

2   ADAPTIVE STRATEGY FOR NEARLY BODY-FITTED MESHES

A metric-based anisotropic mesh adaptation procedure is performed. It generates a uniform unit mesh [3] in a prescribed Riemannian metric space that corresponds to an anisotropic adapted mesh in the Euclidean space. Anisotropic mesh adaptation is performed in the vicinity of the interface Γ described by the level-set function $\phi(\mathbf{x})$, i.e. in a band $\{\mathbf{x}$ s.t. $|\phi(\mathbf{x})| \leq E\}$ of thickness $2E$ around Γ. With a linear discretization, the approximation error on the level-set function $\phi(\mathbf{x})$ is of second order. An appropriate metric field $\mathcal{M}$ can thus be constructed from the gradient vector $\nabla\phi(\mathbf{x}) = (\phi_x\ \phi_y\ \phi_z)^T$ and the Hessian matrix $\mathcal{H}(\phi(\mathbf{x}))$ of $\phi(\mathbf{x})$. More details about the construction of the metric can be found in [4].

In a first step, we apply this method to an academic 2D Laplace problem in a square with an embedded planar surface [5]. The solution is compared to the results obtained on anisotropic meshes with results obtained on uniform refined meshes and isotropic adaptive refined meshes (see Fig. 1). We show that an appropriate level of local refinement around the geometry recovers the optimal grid convergence rate for the solution, whereas uniform refinement yields first-order convergence as can be seen in the left plot of Fig. 2.

We also show in the right plot of Fig. 2 that controlling the anisotropic character of the adaption further enables the error of the geometrical discretization to decrease at optimal rate, which is not the case for isotropic refinement. This affects particularly the computation of integral quantities, such as lift and drag in CFD. Anisotropic adaptive refinement also slows down the growth of the number of unknowns, which limits the computational overhead.

3   ADAPTIVE MESHES FOR CFD

The adaptive strategy for CFD combines the presented nearly body-fitted adaptive mesh strategy with an iterative anisotropic adaption to the flow solution. A second mesh metric is constructed by calculating a scaled eigenspace of the Hessian matrix of the norm of the velocity. Indeed, as we are using linear finite element interpolation for the solution of the Navier-Stokes equations, the interpolation error is equivalent to second order derivatives and it has been shown that a large proportion of the discretization error is governed by this error indicator. This second mesh metric is then intersected with the level-set based anisotropic mesh metric.

We present two incompressible flow problems involving respectively a cylinder and more complex geometry case, namely an array of cylinders. The overall approach for the CFD
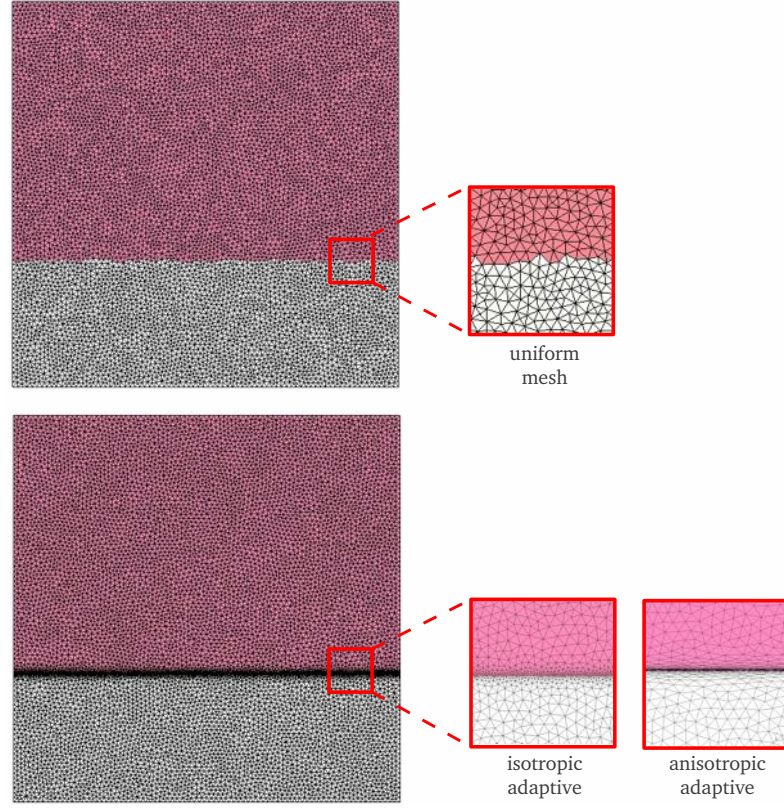
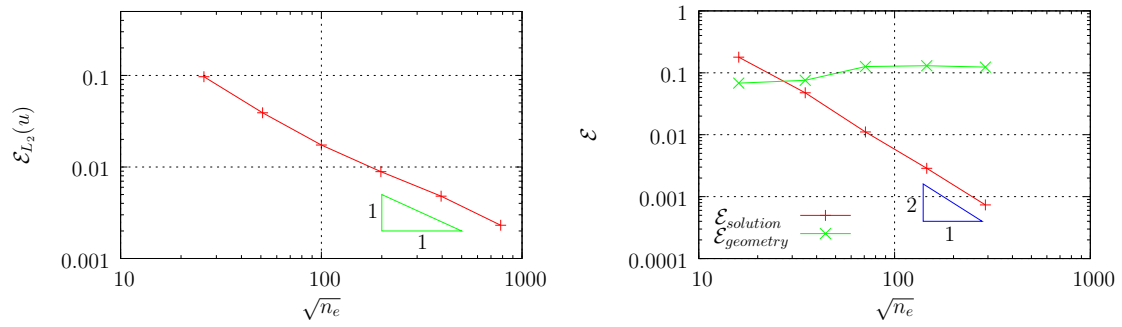**Figure 1**: Uniform refined mesh and adaptive refined meshes

Figure 2: Solution error on uniform refined meshes (left) and geometry error on isotropic adaptive refined meshes (right)

problems can be explained as follows: the problem is first solved on a very coarse mesh with anisotropic elements in the vicinity of the interfaces. The mesh is then successively adapted to both the geometry and to the flow field. For the unsteady case, the solution at time steps which correspond to maximal values of the lift coefficient is used for iteratively adapting the mesh (see Fig. 3).
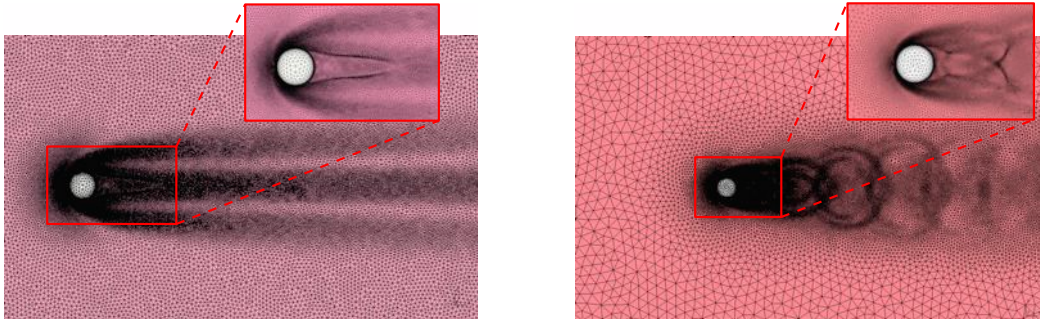
Figure 3: Adaptive mesh for steady flow (left) and unsteady flow (right) over a 2D circular cylinder
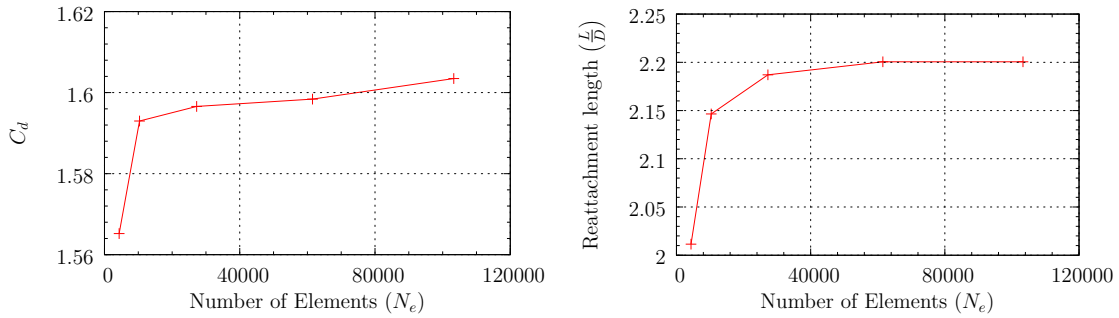
Figure 4: Convergence of drag coefficient and reattachment length at $Re = 40$

While both drag and reattachment length in the steady flow over the cylinder converge to the expected value in Fig. 4, the unsteady case also demonstrates the accuracy of the method as can be seen in Fig. 5.

Concerning the application to a complex geometry, we consider the benchmark described in Geller et al. [6]. The solution is in good agreement with the reference results.

4 CONCLUSION

The use of the standard finite element solver for solving CFD problems on "nearly body-fitted meshes" proves that the optimal rate of convergence can be obtained, and that the methodology yields accurate flow solutions, despite very limited user interaction.
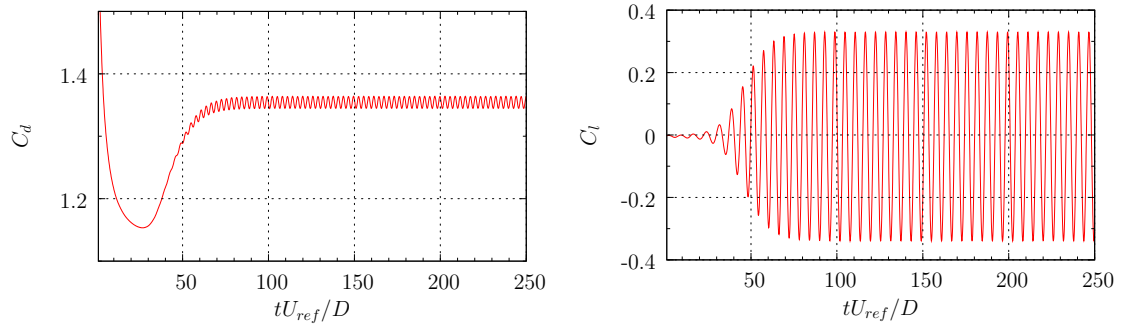
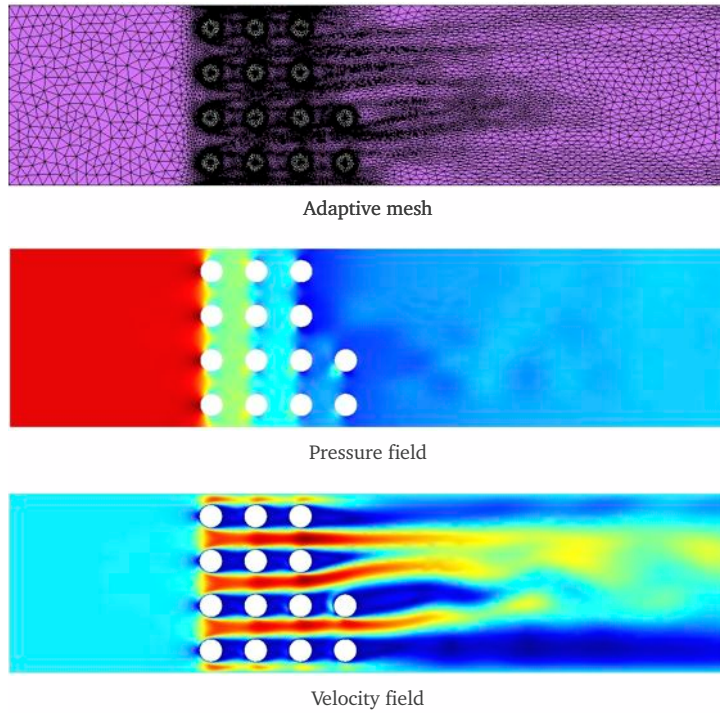**Figure 5**: Lift and drag coefficients at $Re = 100$

Adaptive mesh

Pressure field

Velocity field

Figure 6: Computational mesh for flow over array of cylinders at $Re_E = 200$
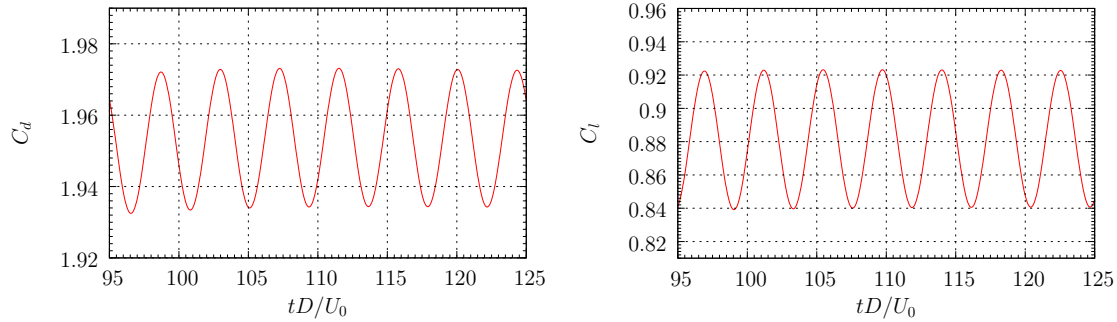
Figure 7: Lift and drag coefficients at $Re_E = 200$

REFERENCES

[1] Claisse, A., Ducrot, V., Frey, P. Levelsets and anisotropic mesh adaptation. *Discrete and Continuous Dynamical Systems*, (2009) **23**(1–2):165–183.

[2] Hachem, E. Kloczko, T. ,Digonnet, H. , Coupez T. Stabilized finite element solution to handle complex heat and fluid flows in industrial furnace using the immersed volume method, *International Journal for Numerical Methods in Fluids* (2010), **68**:99–121.

[3] Frey, P.J., Alauzet, F. Anisotropic mesh adaptation for CFD computations. *Computer Methods in Applied Mechanics and Engineering*, (2005) **194**(48–49):5068–5082.

[4] Quan, D.-L , Toulorge, T., Marchandise, E., Remace, J.-F., Bricteux, G. An hp-anisotropic mesh adaptation procedure for embedded geometries, *submitted to Computer Methods in Applied Mechanics and Engineering* (2013).

[5] Dolbow, J.E., Harari, I. An efficient finite element method for embedded interface problems. *International Journal for Numerical Methods in Engineering*, (2009) **78**(2):229–252.

[6] Geller, S., Krafczyk, M., Toelke J., Turek S., Hron J. Benchmark computations based on lattice-Boltzmann, finite element and finite volume methods for laminar flows. *Computer and fluids* (2006), **35**:888–897.

ADAPTIVE MODELING FOR PARTITIONED-DOMAIN CONCURRENT MULTISCALE CONTINUUM MODELS

KRISTOFFER G. VAN DER ZEE[*],
SERGE PRUDHOMME[†] **AND J. TINSLEY ODEN**[‡]

[*] Multiscale Engineering Fluid Dynamics
Technische Universiteit Eindhoven
e-mail: k.g.v.d.zee@tue.nl   web: www.tue.nl/vanderzee

[†] Department of Mathematics and Industrial Engineering
École Polytechnique de Montréal
e-mail: serge.prudhomme@polymtl.ca

[‡]Institute for Computational Engineering and Sciences (ICES)
The University of Texas at Austin
e-mail: oden@ices.utexas.edu

Key words: adaptive modeling, partitioned-domain concurrent multiscale modeling, a posteriori error estimates, shape derivatives

Abstract. In this contribution adaptive modeling strategies are considered for the control of modeling errors in so-called partitioned-domain concurrent multiscale models. In these models, the exact fine model is considered intractable to solve throughout the entire domain. It is therefore replaced by an approximate multiscale model where the fine model is only solved in a small subdomain, and a coarse model is employed in the remainder.

We review two approaches to adaptively improve the approximate model in a general framework assuming that the fine and coarse model are described by (local) continuum models separated by a sharp interface. In the classical approach [1] an a posteriori error estimate is computed, and the model is improved in those regions with the largest contributions to this estimate. In the recent shape-derivative approach [2] the interface between the fine and coarse model is perturbed so as to decrease a shape functional associated with the error. Several numerical experiments illustrate the strategies.

[1] J. T. Oden and S. Prudhomme. Estimation of modeling error in computational mechanics. *J. Comput. Phys.*, 182:496–515, 2002.

[2] H. Ben Dhia, L. Chamoin, J. T. Oden, and S. Prudhomme. A new adaptive modeling strategy based on optimal control for atomic-to-continuum coupling simulations. *Comput. Methods Appl. Mech. Engrg.*, 200:2675–2696, 2011.

PHASE-FIELD-DRIVEN MODEL ADAPTIVITY

TIMO VAN OPSTAL[*], KRIS VAN DER ZEE[*], PABLO SELESON[†] AND SERGE PRUDHOMME[‡]

[*] Multiscale Engineering Fluid Dynamics section
Mechanical Engineering department
Eindhoven University of Technology
P.O. Box 513, 5600MB, Eindhoven, Netherlands
e-mail: t.m.v.opstal@tue.nl, www.tue.nl/staf/t.m.v.opstal

[†] Institute for Computational Engineering and Sciences
The University of Texas at Austin
201 East 24th St. Stop C0200, Austin, TX 78712, USA
e-mail: pablo@ices.utexas.edu, ices.utexas.edu

Département de Mathématiques et de Génie Industriel
Ecole Polytechnique de Montréal
C.P. 6079, succ. Centre-Ville, Montréal, Québec, Canada H3C 3A7
e-mail: serge@ices.utexas.edu, ices.utexas.edu

Key words: Adaptive modeling, diffuse interfaces, blending techniques, phase fields

Abstract.

Methods for coupling two compatible models have been developed during the last decade or so to simulate problems for which one may identify a small subregion where the assumptions of a *coarse-scale model* break down and whose physical behavior should rather be described by some *fine-scale* model. Examples of blending techniques to couple continuum models with non-local particle models can be found, for instance, in [1, 2]. The main motivation in using such approaches is that fine-scale models are usually too expensive to be employed in the entire domain Ω due to their small length- and time-scale features and their nonlinear behavior. Therefore, the idea is to use the fine model only in a subdomain $\omega \subset \Omega$, where it is deemed necessary, and the coarse-scale model in the remainder of Ω, except in a layer separating these two within which one imposes a gradual transition, via a so-called weighting or *blending function*, from the fine-scale to the coarse-scale model. More concretely, the blending function has value unity in the region of the fine-scale model, zero in the region of the coarse-scale model, and monotonically varies from unity to zero between these two regions.

Questions that naturally arise are how to quantify the errors incurred by substituting a hybrid model for the fine-scale model and how to choose the domain of the fine-scale

model in an optimal way. Note that errors and optimality should be measured in terms of a quantity of interest $Q = Q(u)$, a functional of the solution u of the problem, that characterizes the goal of the simulations. Goal-oriented adaptivity provides a framework to estimate, and substantially control, these approximation errors. Finding the optimal configuration of the coupled problem can be achieved by considering the blending function as an unknown and determining its optimal shape. We develop here a phase-field formulation to solve for the blending function as the problem drives the solution either to the value one or to the value zero with a smooth, narrow transition in between. In addition, a phase-field model satisfies a gradient flow structure, for which an energy is minimized. By adding to the energy functional the error in the quantity of interest, one can thus determine the evolution of the blending function that drives down the error with respect to the goal.

The phase-field formulation for model blending adaptivity will be explored on a simple problem that couples a fourth-order partial differential equation model (the fine-scale model that incorporates non-local effects) with a second-order partial differential equation model (the coarse-scale model that ignores those non-local effects). In particular, we will present some mathematical properties of the formulation, provide some numerical results, and discuss the viability of the methodology.

REFERENCES

[1] Prudhomme, S., Ben Dhia, H., Bauman, P.T., Elkhodja, N., and Oden, J.T. Computational analysis of modeling error for the coupling of particle and continuum models by the Arlequin method. *Comput. Methods Appl. Mech. and Engrg.* (2008) **197**(41–42):3399—3409.

[2] Seleson, P., Beneddine, S., and Prudhomme S. A force-based coupling scheme for peridynamics and classical elasticity. *Computational Material Science*, (2012) in press.

ERROR ASSESSMENT FOR TIMELINE-DEPENDENT QUANTITIES OF INTEREST IN TRANSIENT ELASTODYNAMICS

F. VERDUGO[*], N. PARÉS [†] AND P. DÍEZ[*]

[*]Laboratori de Càlcul Numèric (LaCàN),
Universidat Politècnica de Catalunya (UPC),
Jordi Girona, 1-3, E-08034, Barcelona, Spain.
e-mails {francesc.verdugo,pedro.diez}@upc.edu

[†]Laboratori de Càlcul Numèric (LaCàN)
Escola Universitària d'Enginyeria Tècnica Industrial de Barcelona (EUETIB),
Compte d'Urgell, 187, E-08036, Barcelona, Spain.
e-mail: nuria.pares@upc.edu

Key words: goal-oriented error assessment, elastodynamics, transient dynamics, adjoint problem, quantity of interest, timeline-dependent quantity of interest, modal analysis

Abstract. This work presents a new approach to assess the error in specific quantities of interest in the framework of linear elastodynamics. In particular, a new type of quantities of interest (referred as timeline-dependent quantities) is proposed. These quantities are scalar time-dependent outputs of the transient solution which are better suited to time-dependent problems than the standard scalar ones available in the literature. The proposed methodology furnishes error estimates for both the standard scalar and the new timeline-dependent quantities of interest. The key ingredient is the modal-based approximation of the associated adjoint problems which allows efficiently computing and storing the adjoint solution.

1 INTRODUCTION

Assessing the reliability and/or improving the efficiency of the finite element based approximations has motivated the development of a huge variety of error assessment techniques. The pioneering references on this topic focus in steady-state elliptic problems, e.g. linear elasticity or steady heat transfer. In the context of elliptic problems, the early works consider the energy norm as an error measure [1, 2, 3]. Much later, functionals outputs or *quantities of interest* are introduced to assess the error [4, 5, 6, 7]. The estimates assessing the error in quantities of interest are usually referred in the literature

as *goal-oriented*. These techniques are extended to deal with other linear and non-linear problems, as well as to time-dependent problems.

An important issue associated with goal-oriented estimates for elastodynamics (and also for other time-dependent problems) is the definition of the quantity of interest itself. Typically, the quantity is expressed in terms of a (linear) functional, which transforms the solution of the problem into a single representative scalar value. In many cases, a single scalar value does not provide enough pieces of information about the whole time-space solution. This suggests introducing a new type of quantities of interest. The output of such a quantity of interest is not anymore a scalar quantity but a time-dependent function. The major novelty of this article is the introduction of this new type of quantities. They are referred as *timeline-dependent* quantities of interest in contrast with the standard *scalar* quantities.

2 PROBLEM STATEMENT

2.1 Governing equations

Consider a visco-elastic body occupying an open bounded domain $\Omega \subset \mathbb{R}^d$, $d \leq 3$, with boundary $\partial\Omega$. The boundary is divided in two disjoint parts, Γ_{N} and Γ_{D} such that $\partial\Omega = \overline{\Gamma}_{\mathrm{N}} \cup \overline{\Gamma}_{\mathrm{D}}$ and the time interval under consideration is $I := [0, T]$. Under the assumption of small perturbations, the evolution of displacements $\mathbf{u}(\mathbf{x}, t)$ and stresses $\boldsymbol{\sigma}(\mathbf{x}, t)$, $\mathbf{x} \in \Omega$ and $t \in I$, is described by the visco-elastodynamic equations,

$$\rho(\ddot{\mathbf{u}} + a_1\dot{\mathbf{u}}) - \boldsymbol{\nabla} \cdot \boldsymbol{\sigma} = \mathbf{f} \quad \text{in } \Omega \times I, \tag{1a}$$

$$\mathbf{u} = \mathbf{0} \quad \text{on } \Gamma_{\mathrm{D}} \times I, \tag{1b}$$

$$\boldsymbol{\sigma} \cdot \mathbf{n} = \mathbf{g} \quad \text{on } \Gamma_{\mathrm{N}} \times I, \tag{1c}$$

$$\mathbf{u} = \mathbf{u}_0 \quad \text{at } \Omega \times \{0\}, \tag{1d}$$

$$\dot{\mathbf{u}} = \mathbf{v}_0 \quad \text{at } \Omega \times \{0\}. \tag{1e}$$

where an upper dot indicates partial derivation with respect to time, that is $(\dot{\bullet}) := \frac{\mathrm{d}}{\mathrm{d}t}(\bullet)$, and $\mathbf{n}$ denotes the outward unit normal to $\partial\Omega$. The problem data are the mass density $\rho = \rho(\mathbf{x}) > 0$, the first Rayleigh coefficient $a_1 \geq 0$, the body force $\mathbf{f} = \mathbf{f}(\mathbf{x}, t)$ and the traction $\mathbf{g} = \mathbf{g}(\mathbf{x}, t)$ acting on the Neumann boundary $\Gamma_{\mathrm{N}} \times I$. The initial conditions for displacements and velocities are $\mathbf{u}_0 = \mathbf{u}_0(\mathbf{x})$ and $\mathbf{v}_0 = \mathbf{v}_0(\mathbf{x})$ respectively. For the sake of simplicity and without any loss of generality, Dirichlet conditions (1b) are taken as homogeneous.

The set of equations (1) is closed with the constitutive law,

$$\boldsymbol{\sigma} = \boldsymbol{\mathcal{C}} : \boldsymbol{\varepsilon}(\mathbf{u} + a_2\dot{\mathbf{u}}), \tag{2}$$

where the parameter $a_2 \geq 0$ is the second Rayleigh coefficient, the tensor $\boldsymbol{\mathcal{C}}$ is the standard 4th-order elastic Hooke tensor and the kinematic relation (corresponding to small perturbations) $\boldsymbol{\varepsilon}(\mathbf{w}) := \frac{1}{2}(\boldsymbol{\nabla}\mathbf{w} + \boldsymbol{\nabla}^{\mathrm{T}}\mathbf{w})$ is considered.

2.2 Numerical approximation

In the following developments, $\hat{\mathbf{u}}$ is assumed to be an approximation of the solution of the boundary value problem (1). For technical reasons, $\hat{\mathbf{u}}$ must have C^0-continuity in space and C^1-continuity in time. Most typically, the approximation computed with the standard Newmark method, say $\mathbf{u}^{H,\Delta t}$, does not fulfill these continuity requirements and has to be post-processed to obtain a suitable smooth in time function $\hat{\mathbf{u}}$.

The numerical approximation $\hat{\mathbf{u}}$ is computed here as a post process of the Newmark solution using the method of the linear accelerations [8]. This post-process consist basically in integrate in time a piecewise linear interpolation of the Newmark accelerations furnishing the smooth velocity $\dot{\hat{\mathbf{u}}}$ and then integrating in time again furnishing the smooth displacement $\hat{\mathbf{u}}$, see [8, 9] for details.

2.3 Scalar and timeline-dependent quantities of interest

A quantity of interest is represented by a functional $L^{\mathcal{O}}(\cdot)$ extracting a single scalar value, $s_T := L^{\mathcal{O}}(\mathbf{u}) \in \mathbb{R}$, of the space-time solution $\mathbf{u}$. A typical expression for this functional is given by

$$L^{\mathcal{O}}(\mathbf{u}) := \int_0^T (\mathbf{f}^{\mathcal{O}}(t), \dot{\mathbf{u}}(t)) \, \mathrm{d}t + \int_0^T (\mathbf{g}^{\mathcal{O}}(t), \dot{\mathbf{u}}(t))_{\Gamma_{\mathrm{N}}} \, \mathrm{d}t + (\rho \mathbf{v}^{\mathcal{O}}, \dot{\mathbf{u}}(T)) + a(\mathbf{u}^{\mathcal{O}}, \mathbf{u}(T)), \quad (3)$$

where $\mathbf{f}^{\mathcal{O}}$, $\mathbf{g}^{\mathcal{O}}$, $\mathbf{v}^{\mathcal{O}}$ and $\mathbf{u}^{\mathcal{O}}$ are the data characterizing the quantity of interest. The functions $\mathbf{f}^{\mathcal{O}}$ and $\mathbf{g}^{\mathcal{O}}$ extract global or localized averages of velocities in Ω and Γ_{N}, respectively, over the whole time simulation $[0, T]$ whereas $\mathbf{v}^{\mathcal{O}}$ and $\mathbf{u}^{\mathcal{O}}$ assess averages of velocities and strains or displacements respectively at the final simulation time T.

The quantity of interest associated with the adjoint solution, namely s_T, is obviously unknown and it is approximated by the quantity of interest associated with the approximated solution $\hat{\mathbf{u}}$, that is $s_T \approx \hat{s}_T := L^{\mathcal{O}}(\hat{\mathbf{u}})$. Goal oriented error estimates aims at assessing the quality of the approximation $\hat{s}_T$ by means of approximating the error $s_T^{\mathrm{e}} = s_T - \hat{s}_T$. Consequently, the problem of goal-oriented consists in finding approximations of the value s^{e}.

This work extends the paradigm of classical goal-oriented error estimation by introducing the new concept of *timeline-dependent quantities of interest*. Timeline-dependent quantities of interest are defined as an extension of (3) as

$$L_{\mathrm{TL}}^{\mathcal{O}}(\mathbf{u})(t) := \int_0^t (\mathbf{f}^{\mathcal{O}}(\tau), \dot{\mathbf{u}}(\tau)) \, \mathrm{d}\tau + \int_0^t (\mathbf{g}^{\mathcal{O}}(\tau), \dot{\mathbf{u}}(\tau))_{\Gamma_{\mathrm{N}}} \, \mathrm{d}\tau + (\rho \mathbf{v}^{\mathcal{O}}, \dot{\mathbf{u}}(t)) + a(\mathbf{u}^{\mathcal{O}}, \mathbf{u}(t)). \quad (4)$$

Note that the time-line dependent quantity $s(t) := L_{\mathrm{TL}}^{\mathcal{O}}(\mathbf{u})(t)$ associated with the function $\mathbf{u}$ is a time dependent function instead of a single scalar value, see figure 1.

The aim of timeline-dependent goal-oriented error estimation strategies is assessing the quality of $\hat{s}(t) = L_{\mathrm{TL}}^{\mathcal{O}}(\hat{\mathbf{u}}; t)$, that is the difference between the exact quantity of interest $s(t) = L_{\mathrm{TL}}^{\mathcal{O}}(\mathbf{u}; t)$ and the approximation obtained with the numerical simulation $\hat{s}(t)$.
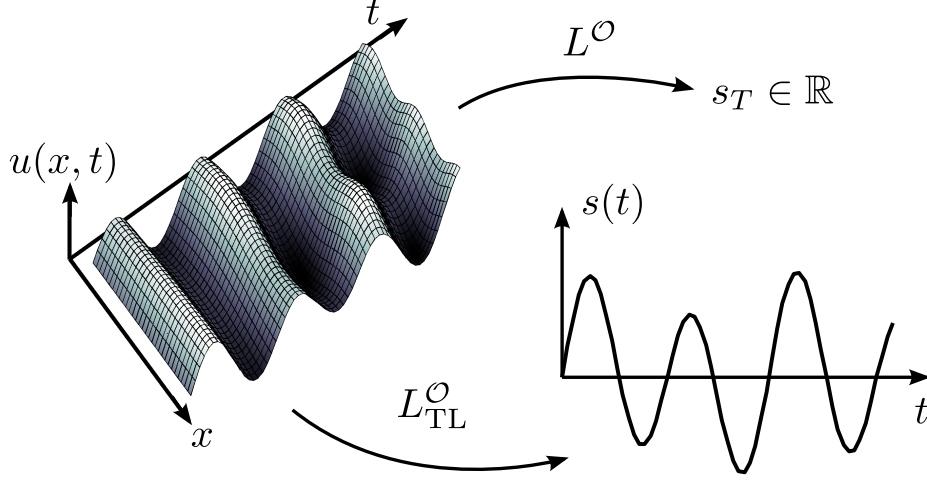
Figure 1: Illustration of scalar and timeline-dependent quantities of interest. The functional $L^{\mathcal{O}}$ maps the time-space solution $\mathbf{u}$ into a scalar value $s_T \in \mathbb{R}$. The operator $L^{\mathcal{O}}_{\mathrm{TL}}$ transforms $\mathbf{u}$ into a time-dependent function $s(t)$.

Thus, the goal of goal-oriented error estimates for timeline-dependent quantities is finding approximation of the time-dependent function

$$s^{\mathrm{e}}(t) := s(t) - \hat{s}(t).$$

3 ASSESSING SCALAR AND TIMELINE-DEPENDENT QUANTITIES OF INTEREST

This section is devoted to present a novel approach to assess the error both in the scalar quantity of interest, s_T^{e}, and in the timeline-dependent quantity, $s^{\mathrm{e}}(t)$, using the modal analysis to obtain a proper approximation of the adjoint solution.

3.1 Assessing Scalar quantities

Assessing the error in quantities of interest requires introducing an auxiliary problem associated with the functional $L^{\mathcal{O}}(\cdot)$, usually denoted by *adjoint* or *dual* problem [9]. The strong form of the adjoint problem associated with the quiantity defined in (3) is, see [9] for details,

$$\rho(\ddot{\mathbf{u}}^{\mathrm{d}} - a_1 \dot{\mathbf{u}}^{\mathrm{d}}) - \boldsymbol{\nabla} \cdot \boldsymbol{\sigma}^{\mathrm{d}} = -\mathbf{f}^{\mathcal{O}} \quad \text{in } \Omega \times I, \tag{5a}$$

$$\mathbf{u}^{\mathrm{d}} = \mathbf{0} \quad \text{on } \Gamma_{\mathrm{D}} \times I, \tag{5b}$$

$$\boldsymbol{\sigma}^{\mathrm{d}} \cdot \mathbf{n} = -\mathbf{g}^{\mathcal{O}} \quad \text{on } \Gamma_{\mathrm{N}} \times I, \tag{5c}$$

$$\mathbf{u}^{\mathrm{d}} = \mathbf{u}^{\mathcal{O}} \quad \text{at } \Omega \times \{T\}, \tag{5d}$$

$$\dot{\mathbf{u}}^{\mathrm{d}} = \mathbf{v}^{\mathcal{O}} \quad \text{at } \Omega \times \{T\}, \tag{5e}$$

with the constitutive law

$$\boldsymbol{\sigma}^{\mathrm{d}} := \boldsymbol{\mathcal{C}} : \boldsymbol{\varepsilon}(\mathbf{u}^{\mathrm{d}} - a_2 \dot{\mathbf{u}}^{\mathrm{d}}). \tag{6}$$

Note that the terms affected by a_1 and a_2 have opposite sign that the ones in the original problem (1). Consequently, the adjoint problem has to be integrated backwards in time, starting from the *final conditions* (5d) and (5e).

The solution of the adjoint problem $\mathbf{u}^{\mathrm{d}}$ allows representing the error in the quantity of interest in terms of residuals. That is

$$L^{\mathcal{O}}(\hat{\mathbf{e}}) = R(\mathbf{u}^{\mathrm{d}}) \tag{7}$$

where $R(\cdot) := L(\cdot) - B(\hat{\mathbf{u}}, \cdot)$ is the weak residual associated with the numerical approximation $\hat{\mathbf{u}}$. The forms $B(\cdot, \cdot)$ and $L(\cdot)$ are defiend as

$$B(\mathbf{v}, \mathbf{w}) := \int_I (\rho(\ddot{\mathbf{v}} + a_1 \dot{\mathbf{v}}), \dot{\mathbf{w}}) \, \mathrm{d}t + \int_I a(\mathbf{v} + a_2 \dot{\mathbf{v}}, \mathbf{w}) \, \mathrm{d}t + (\rho \dot{\mathbf{v}}(0^+), \dot{\mathbf{w}}(0^+)) + a(\mathbf{v}(0^+), \mathbf{w}(0^+)),$$

and

$$L(\mathbf{w}) := \int_I l(t; \dot{\mathbf{w}}(t)) \, \mathrm{d}t + (\rho \mathbf{v}_0, \dot{\mathbf{w}}(0^+)) + a(\mathbf{u}_0, \mathbf{w}(0^+)).$$

where the standard linear and bilinear forms are introduced

$$a(\mathbf{v}, \mathbf{w}) := \int_\Omega \boldsymbol{\varepsilon}(\mathbf{v}) : \boldsymbol{\mathcal{C}} : \boldsymbol{\varepsilon}(\mathbf{w}) \, \mathrm{d}\Omega \quad , \quad l(t; \mathbf{w}) := (\mathbf{f}(t), \mathbf{w}) + (\mathbf{g}(t), \mathbf{w})_{\Gamma_{\mathrm{N}}},$$

along with the scalar products

$$(\mathbf{v}, \mathbf{w}) := \int_\Omega \mathbf{v} \cdot \mathbf{w} \, \mathrm{d}\Omega \quad \text{and} \quad (\mathbf{v}, \mathbf{w})_{\Gamma_{\mathrm{N}}} := \int_{\Gamma_{\mathrm{N}}} \mathbf{v} \cdot \mathbf{w} \, \mathrm{d}\Gamma.$$

Note that the error representation (7) allows obtaining the error in the quantity of interest provided that the exact solution of the adjoint problem is available. Conversely, if an accurate approximation of the adjoint solution is available, say $\tilde{\mathbf{u}}^{\mathrm{d}}$, the error in the quantity of interest is estimated as

$$s_T^e = L^{\mathcal{O}}(\hat{\mathbf{e}}) \approx \hat{R}(\tilde{\mathbf{u}}^{\mathrm{d}}) =: \tilde{s}_T^e. \tag{8}$$

As previously announced, the adjoint problem (5) is of the same type as the original one (1). Thus the adjoint approximation $\tilde{\mathbf{u}}^{\mathrm{d}}$ can be solved with any of the approximation methods available for elastodynamics. Here, the adjoint approximation is computed with modal analysis. For particular quantities of interest, modal analysis is a very efficient way to compute the adjoint problem. Moreover, the modal description of the adjoint solution is a key ingredient in assessing the error in timeline-quantities of interest.

The modal analysis requires computing the M fist vibration modes $\mathbf{q}_i^H$ and frequencies ω_i^H of the problem, $i = 1, \ldots, M$, solution of the generalized eigenvalue problem: find $\mathbf{q}_i^H \in \boldsymbol{\mathcal{V}}_0^H$ such that

$$a(\mathbf{q}^H, \mathbf{w}) = (\omega^H)^2 (\rho \mathbf{q}^H, \mathbf{w}) \quad \forall \mathbf{w} \in \boldsymbol{\mathcal{V}}_0^H, \tag{9}$$

where $\boldsymbol{\mathcal{V}}_0^H$ is the finite element space (H stands for characteristic element size of the underlying computational mesh). Eigenpairs are sorted from low to high frequencies, namely $\omega_1^H \le \omega_2^H \cdots \le \omega_{N_{\mathrm{dof}}}^H$, and eigenvectors are normalized to be orthonormal with respect the product $(\rho \cdot, \cdot)$, i.e.

$$(\rho \mathbf{q}_i^H, \mathbf{q}_j^H) = \delta_{ij}, \quad 1 \le i, j \le N_{\mathrm{dof}}. \tag{10}$$

For thechnical reasons (Galerkin cancellation), the adjoint approximation $\tilde{\mathbf{u}}^{\mathrm{d}}$ cannot be computed by means of the eigenpairs $(\mathbf{q}_i^H, \omega_i^H)$. The reason in the eigenvectors have to belong to a richer space than $\boldsymbol{\mathcal{V}}_0^H$. For that reason, new enhanced eigenpairs $(\tilde{\mathbf{q}}_i, \tilde{\omega}_i)$ are computed starting form the original ones $(\mathbf{q}_i^H, \omega_i^H)$ using the post-processing technique proposed in [10]. Once the enhanced eigenpairs are available, the adjoint approximation is computed as the expansion of the enhanced eigenvectors

$$\tilde{\mathbf{u}}^{\mathrm{d}}(\mathbf{x}, t) := \sum_{i=1}^{M} \tilde{\mathbf{q}}_i(\mathbf{x}) \tilde{y}_i(t). \tag{11}$$

Finaly, the time dependent coeficciets are computed solving the scalar ordinary differential equations

$$\ddot{\tilde{y}}_i - [a_1 + a_2 (\tilde{\omega}_i)^2] \dot{\tilde{y}}_i + (\tilde{\omega}_i)^2 \tilde{y}_i = \tilde{l}_i, \tag{12a}$$

$$\tilde{y}_i(T) = \tilde{u}_i, \tag{12b}$$

$$\dot{\tilde{y}}_i(T) = \tilde{v}_i, \tag{12c}$$

where $\tilde{l}_i(t) := (\mathbf{f}^{\mathcal{O}}(t), \tilde{\mathbf{q}}_i) + (\mathbf{g}^{\mathcal{O}}(t), \tilde{\mathbf{q}}_i)_{\Gamma_{\mathrm{N}}}$, and $\tilde{u}_i$ and $\tilde{v}_i$ are the coefficients best fitting $\mathbf{u}^{\mathcal{O}}$ and $\mathbf{v}^{\mathcal{O}}$ in the enhanced eigenvector basis, that is

$$\mathbf{u}^{\mathcal{O}} \approx \sum_{i=1}^{N_{\mathrm{dof}}} \tilde{\mathbf{q}}_i(\mathbf{x}) \tilde{u}_i \quad \text{and} \quad \mathbf{v}^{\mathcal{O}} \approx \sum_{i=1}^{N_{\mathrm{dof}}} \tilde{\mathbf{q}}_i(\mathbf{x}) \tilde{v}_i. \tag{13}$$

Once the approximation $\tilde{\mathbf{u}}^{\mathrm{d}}$ is available, the error in the quantity of interest is assessed using equation (8).

3.2 Assessing timeline-dependent quantities

Recall that, for a given time $t \in I$, $s(t) = L_{\mathrm{TL}}^{\mathcal{O}}(\mathbf{u})(t)$. In that sense, for this particular value of t, $s(t)$ is seen as a scalar quantity of interest taking t as the *final* time. This

scalar quantity of interest is characterized as $L^{\mathcal{O}}(\cdot) = L^{\mathcal{O}}_{\text{TL}}(\cdot)(t)$. The associated adjoint problem is analogous to the one presented for the scalar quantity of interest and reads:

$$\rho(\ddot{\mathbf{u}}^{\text{d}}_t - a_1 \dot{\mathbf{u}}^{\text{d}}_t) - \boldsymbol{\nabla} \cdot \boldsymbol{\sigma}^{\text{d}}_t = -\mathbf{f}^{\mathcal{O}} \quad \text{in } \Omega \times [0, t], \tag{14a}$$

$$\mathbf{u}^{\text{d}}_t = \mathbf{0} \quad \text{on } \Gamma_{\text{D}} \times [0, t], \tag{14b}$$

$$\boldsymbol{\sigma}^{\text{d}}_t \cdot \mathbf{n} = -\mathbf{g}^{\mathcal{O}} \quad \text{on } \Gamma_{\text{N}} \times [0, t], \tag{14c}$$

$$\mathbf{u}^{\text{d}}_t = \mathbf{u}^{\mathcal{O}} \quad \text{at } \Omega \times \{t\}, \tag{14d}$$

$$\dot{\mathbf{u}}^{\text{d}}_t = \mathbf{v}^{\mathcal{O}} \quad \text{at } \Omega \times \{t\}, \tag{14e}$$

with the constitutive law

$$\boldsymbol{\sigma}^{\text{d}}_t := \boldsymbol{\mathcal{C}} : \boldsymbol{\varepsilon}(\mathbf{u}^{\text{d}}_t - a_2 \dot{\mathbf{u}}^{\text{d}}_t). \tag{15}$$

Note that the solution of this problem is denoted by $\mathbf{u}^{\text{d}}_t$ emphasizing that there is a different solution for each time t. Consequently, equation (14) describes a family of problems, one for each time t.

For a particular instance of time t, the error representation of the timeline-dependent quantity of interest $s^{\text{e}}(t)$ is similar to the standard scalar case but taking the adjoint solution $\mathbf{u}^{\text{d}}_t$ related with the particular value $t \in I$, namely

$$s^{\text{e}}(t) = \hat{R}_t(\mathbf{u}^{\text{d}}_t), \tag{16}$$

where $\hat{R}_t(\mathbf{w}) := L_t(\mathbf{w}; t) - B_t(\hat{\mathbf{u}}, \mathbf{w})$ and

$$B_t(\mathbf{v}, \mathbf{w}) := \int_0^t (\rho(\ddot{\mathbf{v}}(\tau) + a_1 \dot{\mathbf{v}}(\tau)), \dot{\mathbf{w}}(\tau)) \, \mathrm{d}\tau + \int_0^t a(\mathbf{v}(\tau) + a_2 \dot{\mathbf{v}}(\tau), \dot{\mathbf{w}}(\tau)) \, \mathrm{d}\tau$$
$$+ (\rho \dot{\mathbf{v}}(0^+), \dot{\mathbf{w}}(0^+)) + a(\mathbf{v}(0^+), \mathbf{w}(0^+)),$$
$$L_t(\mathbf{w}) := \int_0^t l(\tau; \dot{\mathbf{w}}(\tau)) \, \mathrm{d}\tau + (\rho \mathbf{v}_0, \dot{\mathbf{w}}(0^+)) + a(\mathbf{u}_0, \mathbf{w}(0^+)).$$

Hence, an estimate for $s^{\text{e}}(t)$ is obtained injecting an enhanced adjoint approximation $\tilde{\mathbf{u}}^{\text{d}}_t$ in equation (16)

$$s^{\text{e}}(t) \approx \hat{R}_t(\tilde{\mathbf{u}}^{\text{d}}_t). \tag{17}$$

Obviously, it is not possible in practice to independently compute the infinite solutions $\tilde{\mathbf{u}}^{\text{d}}_t$ (one for each time $t \in I$) and then using them in equation (16) to assess $s^{\text{e}}(t)$. However, taking $\mathbf{f}^{\mathcal{O}}$ and $\mathbf{g}^{\mathcal{O}}$ constant in time (which accounts for a number of interesting cases), the different functions $\mathbf{u}^{\text{d}}_t$ corresponding to different time instances are all equivalent after a time translation. Thus, if $\mathbf{u}^{\text{d}}_t$ is properly computed for a particular value of t, for instance $t = T$, the general functions $\mathbf{u}^{\text{d}}_t$ for $t \neq T$ are easily recovered as a direct post-process of $\mathbf{u}^{\text{d}}_T$. This fundamental result, shown in the following theorem, is the crucial observation that allows the error estimation technique to be brought to fruition.

Theorem 1 *For a given t, let $\mathbf{u}_t^d$ be the solution of the adjoint problem defined by equations (14). Assume that data $\mathbf{f}^{\mathcal{O}}$ and $\mathbf{g}^{\mathcal{O}}$ in (4) are constant in time, i.e. $\mathbf{f}^{\mathcal{O}}(\mathbf{x}, t) = \mathbf{f}^{\mathcal{O}}(\mathbf{x})$ and $\mathbf{g}^{\mathcal{O}}(\mathbf{x}, t) = \mathbf{g}^{\mathcal{O}}(\mathbf{x})$.*

Then, $\mathbf{u}_t^d$ is related with the adjoint solution associated with the final time T, $\mathbf{u}_T^d$, via the time translation

$$\mathbf{u}_t^d(\tau) = \mathbf{u}_T^d(\tau + T - t). \tag{18}$$

Theorem (1) allows to efficiently recover the family of enhanced approximations $\tilde{\mathbf{u}}_t^d$ from the enhanced approximation $\tilde{\mathbf{u}}_T^d$ as

$$\tilde{\mathbf{u}}_t^d(\tau) = \tilde{\mathbf{u}}_T^d(\tau + T - t). \tag{19}$$

Consequently, the approximation $\tilde{\mathbf{u}}_T^d$ is the base for assessing the error both in the scalar and timeline-dependent quantities, providing in the latter case more meaningful information. The translation (19) is done very efficiently by means of the modal description of $\tilde{\mathbf{u}}_T^d$:

$$\tilde{\mathbf{u}}_t^d(\tau) = \sum_{i=1}^{M} \tilde{\mathbf{q}}_i \tilde{y}_i(\tau + T - t). \tag{20}$$

Recall that, functions $\tilde{y}_i$ may be known analytically in many cases and therefore computing the translation $y_i(\tau + T - t)$ is inexpensive in that cases.

Finally, the error in the timeline-dependent quantity is assessed using the computed adjoint approximations $\tilde{\mathbf{u}}_t^d$ in equation (16).

4 NUMERICAL EXAMPLE

This example illustrates the performance of the proposed error estimates in a 2D wave propagation problem. The problem definition is taken from [9] where it is used to test an error estimate providing error bounds in quantities of interest.

(a) Problem geometry   (b) Time-dependent external load
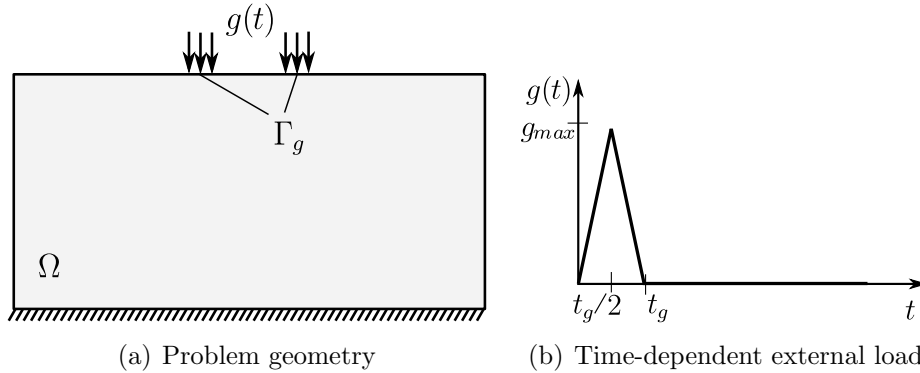
Figure 2: Example 1: Problem statement.

The problem geometry is the rectangular plate sketched in figure 2(a). The plate is initially at rest ($\mathbf{u}_0 = \mathbf{v}_0 = \mathbf{0}$) and loaded with the time dependent traction

$$\mathbf{g} = \begin{cases} -g(t)\mathbf{e}_2 & \text{on } \Gamma_g, \\ 0 & \text{elsewhere,} \end{cases} \tag{21}$$

where $\mathbf{e}_2 := (0, 1)$ and $g(t)$ is the impulsive time-dependent function defined in figure 2(b) with parameters $g_{\max} = 30$ Pa and $t_g = 0.005$ s. No body force is acting in this example ($\mathbf{f} = \mathbf{0}$).

Table 1 details the geometrical parameters and material data, where E and ν are the Young's modulus and Poisson's ratio respectively and the parameter ξ is the dimensionless damping factor. In the examples included her we take $a_1 = 0$, and its corresponding value is $\xi := \frac{1}{2}\omega_1 a_2$, see [9, 11]. Three different values of the viscosity parameter a_2 are considered. The solution of the problem consists of elastic waves propagating along the plate, see [9] for a qualitative description of the solution.

Table 1: Example 1: Problem parameterization

	Geometry			Material properties	
Ω	$(-0.5, 0.5) \times (0, 0.5)$	m²	E	8/3	Pa
Γ_g	$[(0.075, 0.125) \cup (-0.075, -0.125)] \times (0.5)$	m	ν	1/3	
T	0.25	s	ρ	1	kg/m³
			a_1	0	s
			a_2	$\{0, 10^{-4}, 10^{-2}\}$	s
			ξ	$\{0, 0.0247, 2.47\}$	%

The timeline-dependent quantity considered in this example is

$$s(t) = (\rho\mathbf{q}_1, \dot{\mathbf{u}}(t)).$$

The quantity s_T is associated with the *exact* first eigenvector of the generalized eigenvalue problem (9) in the Sobolev space $\boldsymbol{\mathcal{V}}_0$. In the following, the unknown function $\mathbf{q}_1$ is replaced by a reference eigenvector $\mathbf{q}_1^{H,p+1}$ solution of the eigenvalue problem (9) in the discrete space $\boldsymbol{\mathcal{V}}_0^{H,p+1}$. The space $\boldsymbol{\mathcal{V}}_0^{H,p+1}$ is obtained increasing by one the interpolation order of $\boldsymbol{\mathcal{V}}_0^H$.

Figure 3 shows the reference and approximated timeline quantities $s(t)$ and $\hat{s}(t) := (\rho\mathbf{q}_1, \dot{\hat{\mathbf{u}}}(t))$ and the reference and estimated errors $s^e(t)$ and $\hat{s}^e(t)$ for mesh id. 1 and time step id. 3, see table 2. The proposed estimate $\tilde{s}^e(t)$ is really close to the reference value $s^e(t)$ in all cases, also for $a_2 = 0$. It can be observed that, in this example, the quantity of interest associated to the lowest eigenvector $\mathbf{q}_1$ is nearly unaffected by the change in the damping coefficient a_2. However, the time dependent errors $s^e(t)$ and its approximations $\tilde{s}^e(t)$ are smoothed out as the coefficient a_2 increases.

Table 2: Example 1: Space and time discretizations

Mesh id.	N_{nod}	# Elements	Type	H [m]
1	3051	5899	Triangle	$3.2 \cdot 10^{-3}$
2	12000	23596	”	$1.6 \cdot 10^{-3}$
3	47595	94384	”	$7.9 \cdot 10^{-4}$

Time step id.	# steps	Δt [s]
1	100	$2.5 \cdot 10^{-3}$
2	200	$1.3 \cdot 10^{-3}$
3	400	$6.2 \cdot 10^{-4}$
4	800	$3.1 \cdot 10^{-4}$

5 CONCLUSIONS

This article presents a new type of goal-oriented error estimates assessing the error in timeline-dependent quantities of interest. Timeline-dependent quantities are outputs of the solution describing the time evolution of some space-post-processed functional. Compared to the traditional scalar quantities of interest, this approach fits better the requirements of end-users in dynamic problems. Assessing the error in timeline-dependent quantities involves a family of infinite adjoint problems (one for each time instant in the time interval under consideration). However, all these adjoint problems are similar and they can be recovered from a common parent problem (associated with the a scalar quantity of interest) by means of a simple translation (shift) of the time variable.

The second novelty in this paper is the approximation of the adjoint problem using a decomposition into vibration modes. This allows efficiently precomputing and storing the adjoint solution. Thus, the error estimate is computed along the time integration of the original problem. This approach applies both for the scalar and timeline quantities, but it is specially indicated for the latter because it simplifies the implementation of the time shift.

The error estimation strategies proposed in this work are based on an explicit approach. The error estimate is computed injecting an enhanced approximation of the adjoint solution into the residual of the direct problem. The enhancement is based on a local postprocess of the computed eigenvectors, performed only once and not at each time step. This approach is very efficient for some quantities of interest in which the adjoint solution is fairly represented in a modal description.

The numerical examples show that the proposed estimates have a good effectivity for both the scalar and timeline quantities of interest, accounting both for space and time discretization errors. Contrary to other error estimates for linear visco-elastodynamics, the proposed estimates do not degenerate in the limit case of pure elasticity (i.e. when
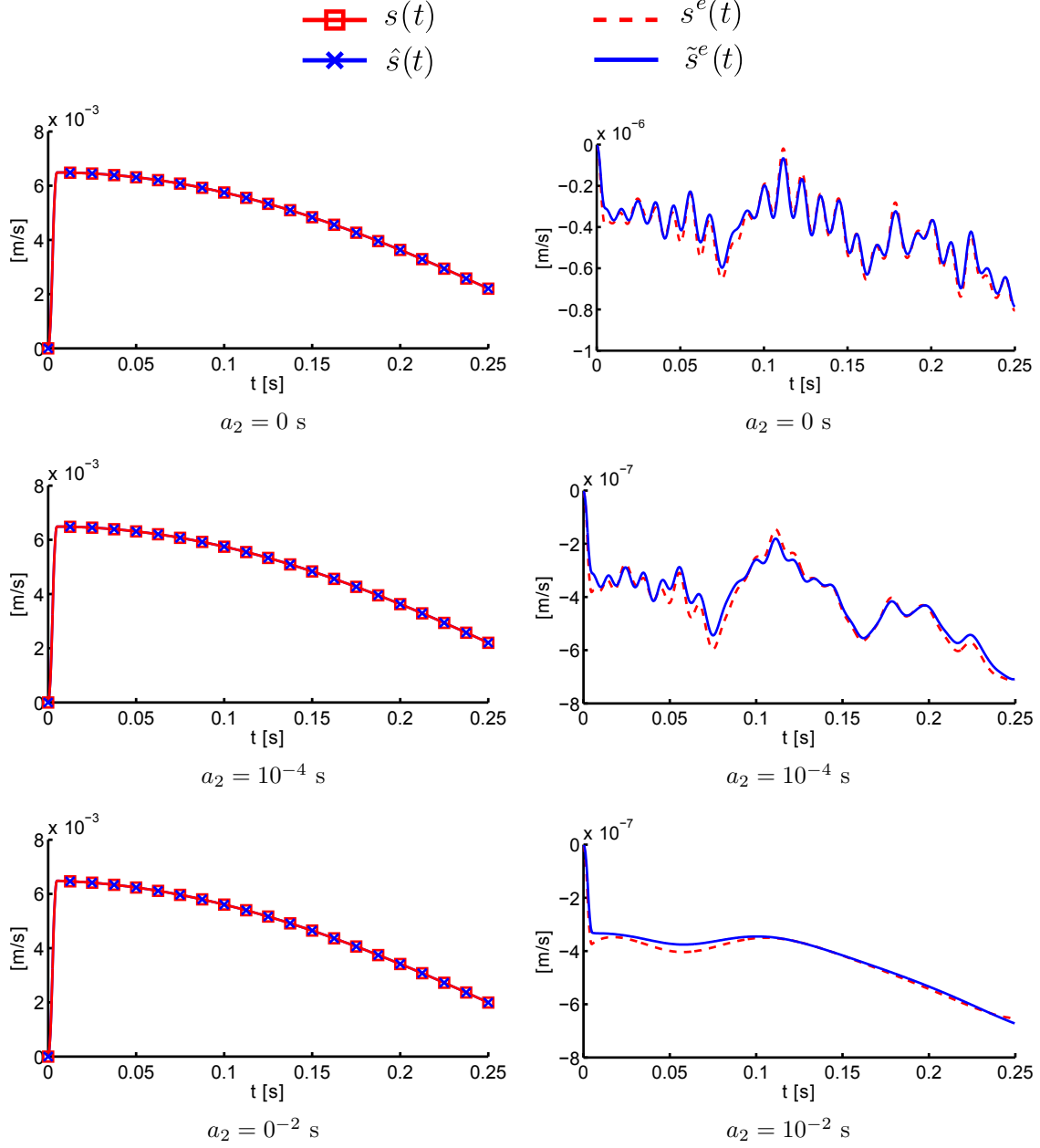
Figure 3: Example 1: Approximated and reference timeline-dependent quantity (left) and estimated and reference errors in the timeline-dependent quantity (right) for the three values of the damping parameter a_2 ($a_2 = 0$ s, top; $a_2 = 10^{-2}$ s, center; $a_2 = 10^{-4}$ s, bottom).

no damping is introduced in the formulation).

In current ongoing work, the proposed error estimation techniques are used as driving indicators for mesh adaptivity.

REFERENCES

[1] I Babuŝka and WC Rheinboldt. Error estimates for adaptive finite element computations. *SIAM J. Numer. Anal.*, 18:736–754, 1978.

[2] P Ladevèze and D Leguillon. Error estimate procedure in the finite element method. *SIAM J. on Numerical Analysis*, 20:485–509, 1983.

[3] OC Zienkiewicz and JZ Zhu. A simple error estimator and adaptative procedure for practical engineering analysis. *Int. J. Numer. Meth. Engrg.*, 24:337–357, 1987.

[4] M Paraschivoiu, J Peraire, and AT Patera. A posteriori finite element bounds for linear-functional outputs of elliptic partial differential equations. *Comput. Methods Appl. Mech. Engrg.*, 150:289–321, 1997.

[5] N Parés, J Bonet, A Huerta, and J Peraire. The computation of bounds for linear-functional outputs of weak solutions to the two-dimensional elasicity equations. *Comput. Methods Appl. Mech. Engrg.*, 195:406–429, 2006.

[6] F Cirak and E Ramm. A posteriori error estimation and adaptivity for linear elasticity using the reciprocal theorem. *Comput. Methods Appl. Mech. Engrg.*, 156:351–362, 1998.

[7] S Prudhomme and JT Oden. On goal–oriented error estimation for elliptic problems: application to the control of pointwise errors. *Comput. Methods Appl. Mech. Engrg.*, 176:313–331, 1999.

[8] P Ladevèze and JP Pelle. *La maîtrise du calcul en mécanique linéaire et non linéaire.* Lavoisier, 2001.

[9] F. Verdugo and P. Diez. Computable bounds of functional outputs in linear visco-elastodynamics. *Comput. Methods Appl. Mech. Engrg.*, 245–246:313–330, 2012.

[10] N.E. Wiberg, R. Bausys, and P. Hager. Adaptive h-version eigenfrequency analysis. *Computers and structures*, 71:565–584, 1999.

[11] J Waeytens. *Contrôle des calculs en dynamique: bornes strictes et pertinents sur une quantié d'intérêt.* PhD thesis, LMT-Cachan, 2010.

ON THE FEASIBILITY OF GOAL-ORIENTED ERROR ESTIMATION FOR SHIP HYDRODYNAMICS

JEROEN WACKERS, GANBO DENG AND MICHEL VISONNEAU

Laboratoire de recherche en Hydrodynamique, Energétique et Environnement Atmosphérique,
Ecole Centrale de Nantes, CNRS-UMR 6598, 44321 Nantes Cedex 3, France
e-mail: jeroen.wackers@ec-nantes.fr

Key words: Error Estimation, Anisotropic Refinement, High-Reynolds Flows

Abstract. An initial study is made of the possibilities for goal-oriented error estimation and anisotropic grid refinement in the simulation of water flow around ships. A finite-volume discretisation for the adjoint solution is presented together with least-squares computation of the local residuals. The paper shows the difficulties in the computation of adjoints and residuals for high-Reynolds flows, but indicates that error estimation and grid refinement for such flows may be possible.

1 INTRODUCTION

Goal-oriented flow simulation, for the purpose of this article, denotes simulation methods where numerical or physical parameters and techniques can be automatically adjusted by the solver in order to provide accurate and efficient computation of a given single output parameter. The concept originates from the idea that flow computations are often performed to answer a specific question and that computational resources should therefore be applied for answering this question, and for nothing else. And also, if non-expert users are to base crucial design decisions only on the results of CFD computations, they need simulation software which provides reliable results at least partially in an automatic way. Thus, goal-oriented simulation has two key aspects: the simulation has to be adaptive to provide efficient results and the precision of these results must be estimated to guarantee reliability. In this paper, we shall study both error estimation and adaptive grid refinement.

While adjoint-based error estimation and mesh refinement is common for the simulation of structures [1] and has been successfully applied to compressible Euler flow [5], its use for incompressible Reynolds-averaged Navier-Stokes flows at high Reynolds numbers is not straightforward and few results have been reported. Notably, Stück and Rung [8] use the adjoint solution for hull form optimisation. Particular difficulties for these flows are:

- The RANS equations contain turbulence models which are often not taken into account for the computation of the adjoint solution. This may have an influence on the quality of the error estimation.

- To model turbulent boundary layers, meshes are used with very high aspect-ratio cells near the walls. The adjoint equations and especially the evaluation of the local truncation error used in the error estimation are sensible to these meshes.

- The flow is incompressible, which is very hard for grid refinement. There are no obvious local zones which require great precision (such as shock waves and contact discontinuities in compressible flow); to get accuracy, the flow needs to be more or less well resolved everywhere. Finding the resulting optimum grid sizes requires a delicate balance which puts strong requirements on the quality of the refinement criterion.

The goal of this article is to provide an initial investigation of the possibilities for adjoint-based error estimation and grid refinement for the computation of water flow around ships. A continuous adjoint solver is under development for ISIS-CFD, the unstructured Navier-Stokes solver developed by ECN-CNRS. This solver is combined with local truncation error estimation by high-order integration of the RANS equations over the grid cells [4] for error estimation. In the light of the difficulties outlined above, these techniques are investigated critically. Grid refinement for the flows of interest and the unstructured hexahedral meshes that we use, is necessarily anisotropic [10]: grid cells to be refined can be divided in only one direction, as well as in several. For goal-oriented anisotropic refinement, we show a first test with an approximate implementation of the technique proposed by [5] where the criterion is based on the Hessians of the fluxes.

Section 2 describes the ISIS-CFD flow solver. Then section 3 introduces the continuous adjoint equations and briefly describes their discretisation. Sections 4 and 5 give, respectively, an overview of the error estimation and grid refinement techniques. Finally, section 6 shows initial tests on laminar and turbulent flows which shed some light on the possible efficiency of these methods.

2   THE ISIS-CFD FLOW SOLVER

ISIS-CFD, available as a part of the FINE$^{\text{TM}}$/Marine computing suite, is an incompressible unsteady Reynolds-averaged Navier-Stokes (RANS) method [3, 7]. The solver is based on the finite volume method to build the spatial discretisation of the transport equations. Pressure-velocity coupling is obtained through a Rhie & Chow SIMPLE-type method: in each time step, the velocity updates come from the momentum equations and the pressure is given by the mass conservation law, transformed into a pressure equation.

The discretisation is face-based. While all unknown state variables are cell-centered, the systems of equations used in the implicit time stepping procedure are constructed face by face. Fluxes are computed in a loop over the faces and the contribution of each face is

then added to the two cells next to the face. This technique poses no specific requirements on the topology of the cells. Therefore, the grids can be completely unstructured, cells with an arbitrary number of arbitrarily-shaped faces are accepted. The code is fully parallel using the MPI (Message Passing Interface) protocol.

An automatic adaptive grid refinement technique is included in the solver ISIS-CFD [9, 10]. The method supports the isotropic and anisotropic refinement of unstructured hexahedral meshes. Earlier refinements can be undone in order to adapt the grid to unsteady problems. The refinement criterion, which indicates where the grid must be refined, can be modified very easily; different refinement criteria have already been tested. And finally, the grid refinement is performed in parallel and includes an automatic dynamic load balancing in order to redistribute the refined grid over the processors when some partitions have been refined more than the others.

3   ADJOINT EQUATIONS AND DISCRETISATION

This section discusses briefly the adjoint to the incompressible RANS equations and its discretisation. The RANS system of equations itself, $\mathcal{N}(\mathbf{U}) = 0$ can be expressed as follows:

$$[u_i u_j + p\delta_{i,j} - \mu((u_i)_j + (u_j)_i)]_j = 0, \tag{1a}$$

$$(u_j)_j = 0, \tag{1b}$$

with u_i the velocity components, p the pressure, μ the (variable) viscosity coming from a turbulence model, and δ the Kronecker delta function. The Einstein summation convention is used, indices outside brackets denote differentiation. $\mathbf{U} = [u_1, u_2, u_3, p]^T$ is the exact solution of this system. Finally, $\mathbf{U}_h$ denotes an approximate (numerical) solution of (1).

We are interested in the error for an output functional on the solution $J(\mathbf{U})$, when it is computed from $\mathbf{U}_h$ instead of $\mathbf{U}$. Linearisation gives:

$$J(\mathbf{U}_h) - J(\mathbf{U}) \approx (\mathbf{g}, \mathbf{U}_h - \mathbf{U})_\Omega, \qquad \text{with } \mathbf{g} = [\frac{\partial J}{\partial u_1}, \frac{\partial J}{\partial u_2}, \frac{\partial J}{\partial u_3}, \frac{\partial J}{\partial p}]. \tag{2}$$

$(\cdot, \cdot)_\Omega$ denotes an inner product, integrated over the flow domain. The adjoint solution $\mathbf{z}$ is then defined by:

$$(\mathbf{z}, \mathcal{N}(\mathbf{U}_h) - \mathcal{N}(\mathbf{U}))_\Omega = (\mathbf{g}, \mathbf{U}_h - \mathbf{U})_\Omega, \qquad \forall(\mathbf{U}_h - \mathbf{U}). \tag{3}$$

For the RANS equations, the adjoint system reads:

$$-u_i(z_j)_i - u_i(z_i)_j + (z_p)_j - \mu((z_i)_j + (z_j)_i)_j = g_j, \tag{4a}$$

$$(z_j)_j = g_p, \tag{4b}$$

where $\mathbf{z} = [z_1, z_2, z_3, z_p]^T$.

Like the primal system, the adjoint equations are discretised with a finite-volume technique and a segregated Rhie & Chow method; this is possible because the continuity equation (4b) is the same as in the primal system. Space is lacking here to describe the discretisation in full, but a few remarks are given:

- The term $-u_i(z_i)_j$ makes these equations non-conservative and impossible to put in conservative form. Therefore, the standard finite-volume technique where the equations are integrated over the cells to produce expressions containing only the fluxes over the cell faces, cannot be used. To construct a finite-volume like discretisation, we use constant cell-centred values for the velocities u_i when integrating over a cell. Thus, for all convective terms, the fluxes over a face are different for the left and right cells, since the u_i on the two sides of the face are different.

- The convective term $-u_i(z_j)_i$ is discretised with upwind reconstruction. However, the term $-u_i(z_i)_j$ originates from the linearisation of the mass fluxes which, in the primal system, use the Rhie & Chow reconstruction that is closer to the central discretisation. We found that, for the adjoint system to be stable, also the adjoint terms $-u_i(z_i)_j$ require a central discretisation. Despite this discretisation, the current formulation lacks robustness for high-Reynolds flows in regions of very low velocity, where iterative errors sometimes increase very slowly leading to an eventual divergence of the computation. This is a problem which could come either from the discretisation or from the segregated solution procedure; we are working on its solution.

- The velocities in the convective operators are the velocities u_i coming from the primal system, but due to the minus signs in (4a) the adjoint 'flow' is backwards. This must be taken into account in the upwind discretisations.

As for the primal system, the z_i-equations are solved with Gauss-Seidel while PGCStab is used for the z_p-equation. The z_i-corrections are underrelaxed, but not the z_p-correction.

4 ESTIMATING RESIDUALS

Equation (3) can be used directly to estimate the error in $J(\mathbf{U}_h)$ or even to obtain an improved approximation by subtracting the error estimation from the computed value of J. For this, besides the adjoint solution, we need to compute the residuals $\mathcal{N}(\mathbf{U}_h)$, the result of applying the exact RANS equations to the approximate numerical solution. Since we cannot evaluate these exact equations, the residuals are approximated with a higher-order finite-volume discretisation, similar to [4]. This approximation, valid on structured and unstructured grids, is performed as follows:

- Flux vectors in the face centres and in the nodes of the faces are computed with least-squares polynomials. For each node, third-order polynomials for each component
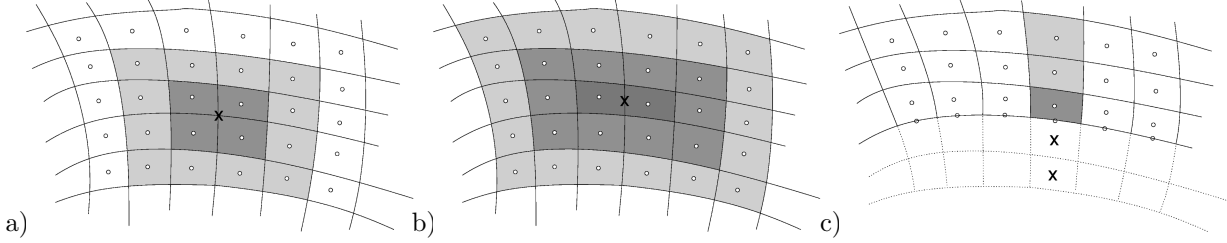
Figure 1: Stencils used in the residual estimation, for reconstruction in nodes (a), face centres (b), and for extrapolating two layers of ghost cells on the boundary (c). Examples on structured grids.

of U_h are fitted through the values in the neighbour cells of the nodes and the neighbours' neighbours (figure 1a). For the faces, the two cells next to the face are used as well as their neighbours and neighbours' neighbours (figure 1b). These are the minimum stencil sizes which ensure that sufficient points are available for fitting third-order polynomials, i.e. four points in all directions. From these fitted polynomials, we get the state vector and derivatives in the nodes and face centres, which are used to compute fluxes.

- Then, third-order accurate quadrature integration of the flux over the faces is performed with the face centre and nodal values. In 2D, third-order accuracy is obtained by assigning a weight of $\frac{2}{3}$ to the face and $\frac{1}{6}$ to each of the two face nodes. Weighting coefficients for arbitrary faces in 3D are under study.

Since the least-squares polynomials are fourth-order accurate, the resulting finite-volume approximation of $\mathcal{N}(\mathbf{U}_h)$ is at least third-order accurate for the convection (which involves one differentiation) and second-order for the diffusion (which requires two differentiations). This is an order more than the primary discretisation of ISIS-CFD, so it is sufficient. Theoretically, the same order of accuracy can be obtained more easily by fitting third-order polynomials through the cell centres and extracting first and second derivatives from these polynomials, which are then substituted directly in (1) to find the residuals. In practice, we have found that this alternative procedure is less accurate so it is not used.

Finally, tests on manufactured solutions revealed strong errors near curved boundaries with large aspect-ratio cells, which are typical for high-Reynolds flows. These errors appear because the cells on the surface are missing the neighbours' neighbour cells in the wall direction; they can be reduced by using symmetric stencils for the least-squares fits, even in the boundary cells. To obtain these, two layers of ghost cells are created at the boundaries whose values are set by extrapolation using 1D third-order polynomials fitted to three cell values and the known value on the face (figure 1). These ghost cells are then used like standard cells for the reconstruction in the nodes and faces. While no new information is added through the extrapolation, this procedure significantly reduces the errors near boundaries.

5 GOAL-ORIENTED REFINEMENT

Apart from error estimation, the adjoint solution can be used for grid refinement. A straightforward way to do this would be to use the local integrand $\mathbf{z} \cdot (\mathcal{N}(\mathbf{U}_h) - \mathcal{N}(\mathbf{U}))$ from equation (3) as a refinement criterion. However, this expression is a scalar so it cannot be used for anisotropic grid refinement, as it cannot specify different cell sizes in different directions.

A method for anisotropic grid refinement which is suitable for our metric-tensor refinement criteria [10] is presented by Loseille et al. [5] for the Euler equations. They express the local residuals in terms of projection errors for the fluxes. Using this, they show that these projection tensors can be minimised if a refined mesh is based on metric tensors that are the Hessian matrices of second spatial derivatives of the flux components. The optimum mesh for computing J is obtained by weighting these Hessians with the gradients of the adjoint solution.

Loseille et al. show excellent results for their method applied to compressible Euler flow. We have made an approximate implementation of this method for RANS flows: the viscous terms are added to the fluxes without further analysis, while separate treatment is advised [2]. For the computation of the flux Hessians, we use third-order cell-centred polynomials which are least-squares fitted to the fluxes computed in the cell centres. An initial test for RANS will be presented in the next section.

6 NUMERICAL TESTS

6.1 Error estimation for 2D profiles

The quality of the adjoint error estimation is tested with two 2D wing section test cases. The first, laminar case is the NACA0012 profile at 4º angle of attack and $Re = 1000$. Turbulent flow is computed around the Nakayama B profile [6] at 4º angle of attack and $Re = 1.2 \cdot 10^6$. Adjoints are computed for two functionals: the drag force and the integral of u_1 over a rectangle in the wake, which resembles the evaluation of the propeller plane flow. The rectangle is centred at $[0.9, -0.02]$ and has dimensions 0.02×0.04 for the NACA wing and 0.01×0.02 for Nakayama. The computed functionals are corrected with the estimated error in order to improve the estimate.

For both cases, computations have been performed on structured C-topology and unstructured HEXPRESS grids with low-Reynolds boundary layers. Figure 2 gives an impression of the solutions. While the boundary layer is of course much thinner for the turbulent case, both flows are similar. No particular problems were encountered for the computation of the primal and the adjoint solutions. However, the computations of the residuals have significant errors near the walls (section 4). Also, the evaluation of the disk integral is complicated because the meshes are not specially refined in the disk zone, so on the coarser structured grids there were too few cells in the disk to perform the integration.

Computed and corrected functional values are given in figure 3. The adjoint error estimation is based on linearisation (section 3) so it is only valid near the asymptotic
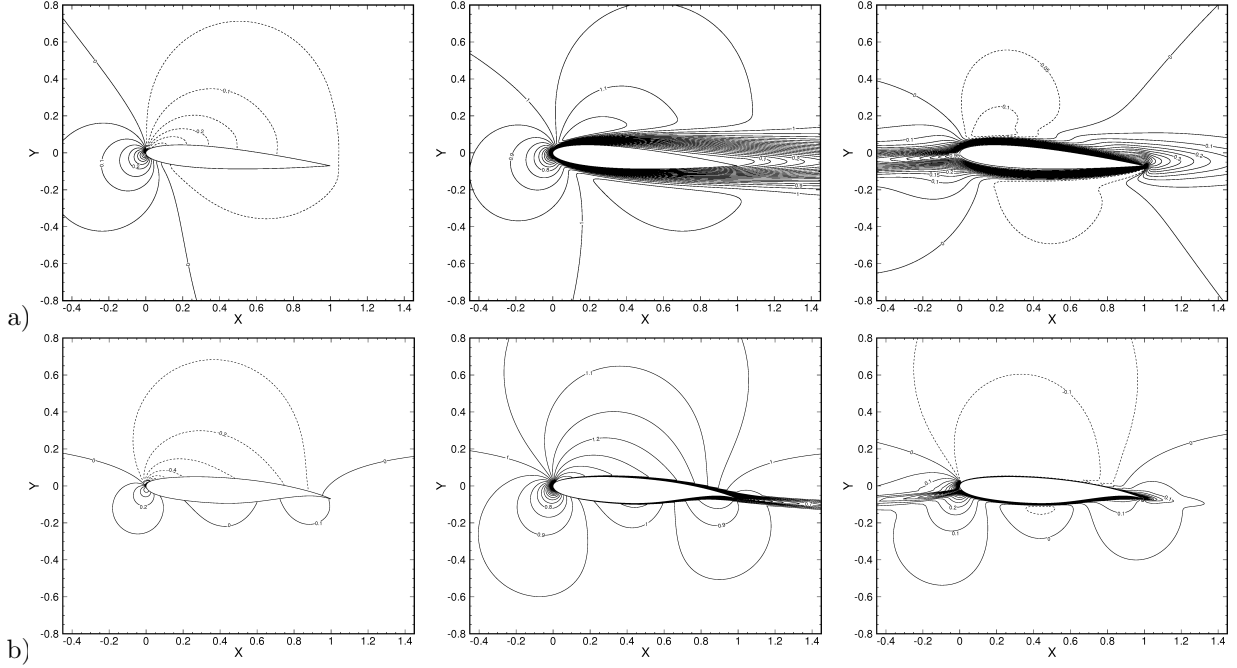
Figure 2: Flow around the NACA0012 (a) and Nakayama B (b) airfoil: pressure, horizontal velocity, and z_1 for the drag functional.

range. True asymptotic convergence is only reached for the NACA0012 drag functional on structured meshes; here the correction seems to improve the solution on the finer meshes even though the influence of near-wall errors in the residuals is non-negligible. Also for the turbulent Nakayama case, improvement may be obtained on the two finest meshes despite the catastrophic failure on the coarser meshes. On unstructured meshes in the laminar case, the error is systematically overestimated by a factor two. And while the solution is not improved for the turbulent case, the estimation is of the same order as the numerical error.

For the rectangle integral, the solutions are not in the asymptotic range so it is difficult to say whether the solutions are improved by the correction or not. However, also here the magnitude of the corrections is the same as the difference between the solutions.

In conclusion, it seems unrealistic to improve functional computations by error correction. However, the adjoint technique might be useful as an error estimator, certainly if the reliability of the residual computation is further improved.

6.2   Goal-oriented grid refinement for the KVLCC2

As an initial test of adjoint-based refinement, we compute the flow without free-surface effects around the KVLCC2 tanker at model scale, $Re = 6.4 \cdot 10^6$, (see figure 4). The turbulence model is EASM without rotation correction [3]. The refinement criterion consists of the flux Hessians, weighted by the gradients of the adjoint solution (section 5).
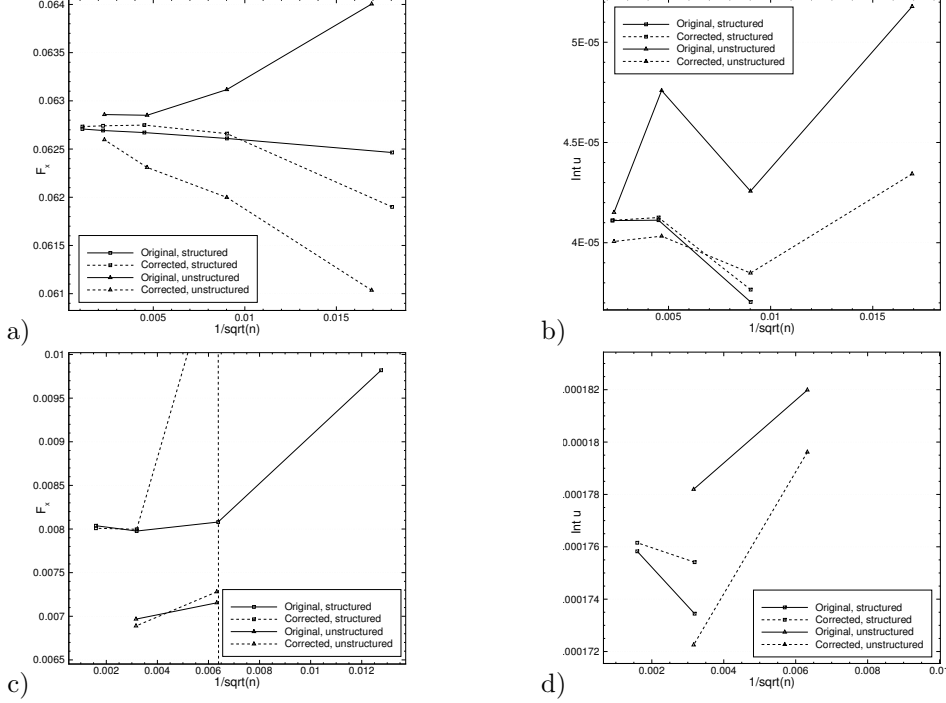
Figure 3: Original and corrected functionals for the NACA0012 airfoil: drag (a) and rectangle integral (b); for the Nakayama B airfoil: drag (c) and rectangle integral (d).

To prevent the divergence of the adjoint solver in low-velocity regions (section 3) which increases on finer grids, the adjoint solution is computed on the original coarse grid (265k cells), based on the converged primal solution for this grid. Automatic refinement is then performed in several steps until the solution and the mesh are converged; for these steps the flux Hessians are computed on the refined grid but the original adjoint solution is kept. In the future, we plan to compute also the adjoint solution on the refined mesh. The final mesh has about 1M cells, an automatic procedure in the solver was used to adjust the threshold for the criterion in order to obtain this number of cells.

The functional is the integral of the axial velocity over the propeller disk with (non-dimensional) radius 0.01541 and thickness 0.01, centered at x/L=0.0175 and z/L=-0.04688. This functional is chosen due to its importance for propeller design, however it does not guarantee that all flow details in the propeller plane are well captured: only the integral value has to be right!

The flow solution is presented in figure 5 and compared with a reference solution

Figure 4: Hull of the KVLCC2 tanker, with the stern facing left and the propeller plane indicated.
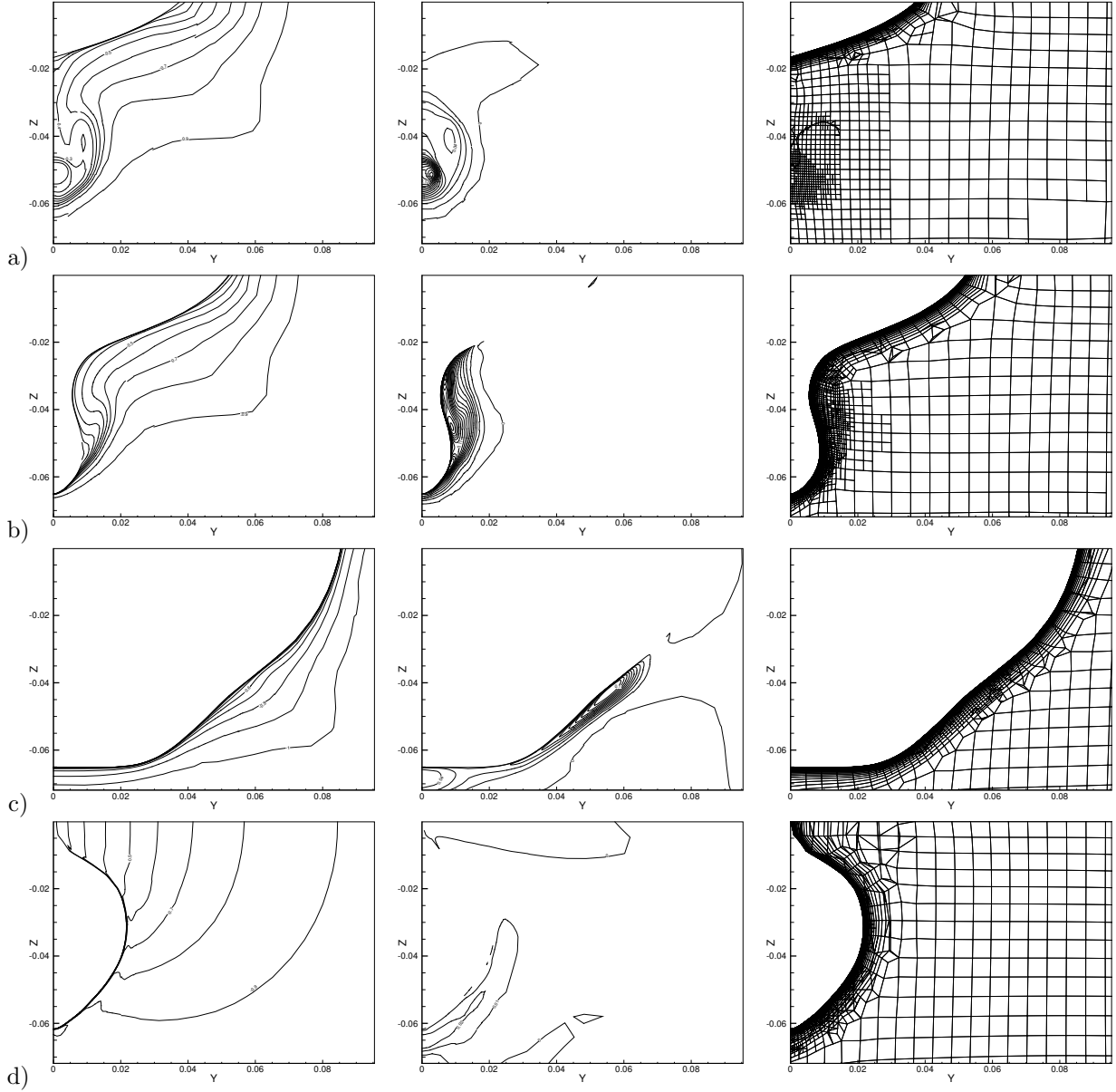
Figure 5: Goal-adaptive refinement for the KVLCC2: axial velocity, z_1 for the propeller plane functional, and the refined mesh in x-cross sections at the propeller plane $x/L = 0.0175$ (a), $x/L = 0.05$ (b), $x/L = 0.17$ (c), and near the bow at $x/L = 1.0$ (d).

Table 1: Computed values of the disk integral functional for the KVLCC2 test case.

Refined grid	Original grid	Reference fine grid
$-3.141 \cdot 10^{-6}$	$-2.621 \cdot 10^{-6}$	$-3.053 \cdot 10^{-6}$
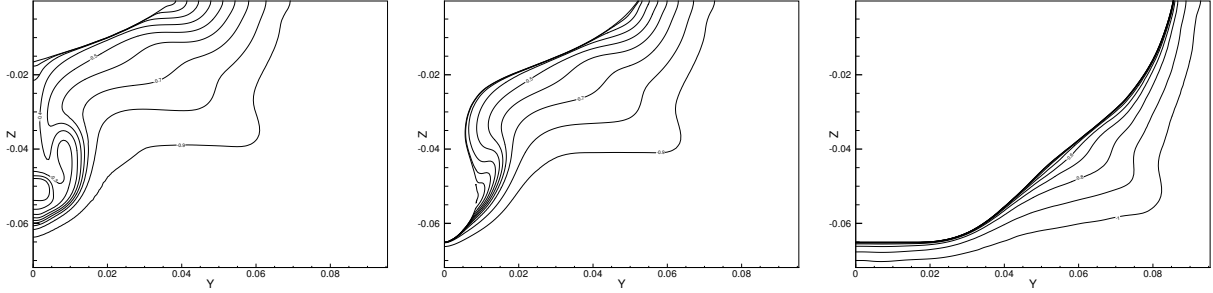
Figure 6: KVLCC2 reference fine grid solution: axial velocity at the propeller plane $x/L = 0.0175$, $x/L = 0.05$, and $x/L = 0.17$.

obtained on a very fine non-adapted grid (7M cells, figure 6). We see how the adjoint solution on the refined grid is non-uniform in the propeller plane, then gets high values around the rear of the ship and continues upstream at the ship's side. Near the bow it has diminished but it is still noticeable. This is reflected in the refined grid, which has fine cells mainly in the boundary layer region. Compared with refined grids created using a pressure Hessian criterion (see [9, 10]) the refinement is concentrated very close to the ship. Also, of course, there is little refinement at the front of the ship, although some refinement below the hull is visible in figure 5d.

The velocity profile in the propeller plane contains the correct 'hook' shape of low velocity around $y/L = 0.01$, $z/L = -0.04$, even though this hook is less noticeable than for the reference solution. Note that, according to the adjoint solution, the flow near the propeller hub has a much bigger influence on the integral than the hook. Further upstream, the velocity in the boundary layer is well resolved, but the velocity away from the hull is only computed approximately, since this velocity is unimportant for the integral according to the adjoint solution. The value of the disk integral functional is computed well (table 1), it is much closer to the fine-grid solution than the value computed on the original coarse grid without refinement. Thus, despite the difficulties in the computation of the adjoint, this initial result is promising.

7   CONCLUSION

This paper investigates the use of goal-oriented error estimation and adaptive grid refinement for high-Reynolds incompressible RANS flows. Error estimates and corrections for a functional are obtained by weighing the local residuals with an adjoint solution to the RANS equations. Tests on 2D airfoils show that it is probably unrealistic to improve

computed functionals by adding the error estimate, but that the estimations are of the same order as the actual errors and could thus be used as error estimators. Notable difficulties are the evaluation of the residuals near walls and the correction of functionals that are far away from asymptotic convergence.

An example is shown of anisotropic grid refinement based on the weighing of flux Hessians with the gradient of the solution. Whether this is the optimal choice for goal-oriented refinement and whether significant gains in efficiency can be made with respect to non goal-oriented grid refinement, remains to be investigated.

REFERENCES

[1] D. Aubry, P. Díez, B. Tie, and N. Parés. *Proceedings of the V International Conference on Adaptive Modelling and Simulation (ADMOS 2011)*, Paris, France (2011).

[2] A. Dervieux, A. Belme, H. Alcin, and F. Alauzet. Goal-oriented mesh adaptation for vortex shedding flows. *Proceedings of ECCOMAS 2012*, Vienna, Austria (2012).

[3] R. Duvigneau, M. Visonneau, and G.B. Deng. On the role played by turbulence closures in hull shape optimization at model and full scale. *J Mar Sci Techn* (2003) **8**(1):1–25.

[4] A. Hay and M. Visonneau. Error estimation using the error transport equation for finite-volume methods and arbitrary meshes. *Int J Comput Fluid Dyn* (2006) **20**(7):463–479.

[5] A. Loseille, A. Dervieux, and F. Alauzet. Fully anisotropic goal-oriented mesh adaptation for 3D steady Euler equations. *J Comput Phys* (2010) **229**:2866–2897.

[6] A. Nakayama. Characteristics of the flow around conventional and supercritical airfoils. *J Fluid Mech* (1985) **160**:155–179.

[7] P. Queutey and M. Visonneau. An interface capturing method for free-surface hydrodynamic flows. *Comput Fluids* (2007) **36**(9):1481–1510.

[8] A. Stück, T. Rung. Adjoint RANS with filtered shape derivatives for hydrodynamic optimisation. *Comput Fluids* (2011) **47**(1):22–32.

[9] J. Wackers, G.B. Deng and M. Visonneau. Tensor-based grid refinement criteria for ship flow simulation. *Proceedings of the 12th Numerical Towing Tank Symposium (NuTTS '10)*, Duisburg, Germany (2010).

[10] J. Wackers, G.B. Deng, A. Leroyer, P. Queutey, and M. Visonneau. Adaptive grid refinement for hydrodynamic flows. *Comput Fluids* (2012) **55**:85–100.

VI International Conference on Adaptive Modeling and Simulation
ADMOS 2013
J. P. Moitinho de Almeida, P. Díez, C. Tiago and N. Parés (Eds)

ADAPTIVE SPACE-TIME FINITE ELEMENT METHOD FOR A NONLINEAR EVOLUTIONARY MONOTONE PROBLEM FROM APPLIED SUPERCONDUCTIVITY

A. WAN[*], M. LAFOREST[*] AND F. SIROIS[†]

[*]Département de mathématiques et génie industriel
École Polytechnique de Montréal
C.P. 6079, succursale centre-ville
Montréal, QC, Canada, H3C 3A7
e-mail: andy.wan@polymtl.ca, marc.laforest@polymtl.ca

[†]Département de génie électrique
École Polytechnique de Montréal
C.P. 6079, succursale centre-ville
Montréal, QC, Canada, H3C 3A7
e-mail: f.sirois@polymtl.ca

Key words: adaptivity, error estimation, space-time, finite element method, Maxwell's equations, nonlinear parabolic PDE, superconductivity

Abstract. The steady improvement of the performances of high temperature superconductors (HTS) brings them within reach of new applications, such as HTS motors, transformers and fault current limiters. To optimize the design of these devices, one must be able to predict the magnetic and electric fields in complex 3-D geometries, but doing so efficiently and accurately is still a challenging task. Within the engineering community, phenomenological models relating the electric field and current density of HTS lead to a novel nonlinear evolutionary monotone PDE based on Maxwell's equations, which is effectively a generalization of the classical p-Laplacian problem. Unfortunately, these models possesses sharp moving fronts that lead to the use of prohibitively small time steps in numerical simulations, even in 2-D domains.

In this work, we propose a new numerical space-time method that allows for local space and time adaptivity without the restrictive global timestep constraint. We present an a posteriori error estimator for the computation of the AC loss, a key design parameter for HTS devices. Numerical results are presented in one and two space dimensions attesting to the efficiency of the numerical method.

MOVING NODES ADAPTION COMBINED TO MESHLESS METHODS FOR SOLVING CFD OPTIMIZATION PROBLEMS

HONG WANG[∗], JACQUES PÉRIAUX[∗†] AND ZHI-LI TANG[††]

[∗]University of Jyväskylä
P.O. Box 35 FI-40014, Jyväskylä, Finland
e-mail: hong.m.wang@jyu.fi

[†]International Center for Numerical Methods in Engineering (CIMNE)
Edificio C1, Gran Capitán s/n, 08034 Barcelona, Spain
e-mail: jperiaux@gmail.com

[††]Nanjing University of Aeronautics and Astronautics
Nanjing, 210016, China
e-mail: tangzhili@nuaa.edu.cn

Key words: Adaptive meshless method, Sub-clouds, Nash algorithms, Drag minimization, Multi-objective problem

Abstract. In past decades, many adaptive mesh schemes have been developed and become important tools for designers to simultaneously increase accuracy of their computations and reduce the cost of numerical computations in many engineering problems. In most of the cases the adaptation is done by subdividing cells or elements into finer cells or elements. Maintaining mesh quality during optimization procedure is still a critical constraint to satisfy for accurate design. In the discretized approach using meshless methods, there are no cells or elements but only a cloud of points which flexibility is an advantage compared to the mesh topology constraint. This attractive property facilitates the coupling of meshless methods with adaptive techniques for inverse or optimization problems.

In this paper, an algebraic adaptive meshless scheme based on a weighted reference radius equi distribution is presented. Cloud nodes adaption combined to meshless methods are used to solve inverse and drag minimization Computational Fluid Dynamics (CFD) problems.

The adaptive meshless method coupled with advanced Evolutionary Algorithms (EAs) is considered as a first test case to rebuild via prescribed surface pressure target the shape of the circular arc bump or ogive operating at supersonic shocked flow regimes. The objective functions could be chosen as the distance between candidate and prescribed

pressure coefficients minimized in L_2 norm and uniform level of errors minimized in L_2 norm.

Numerical results demonstrate numerically that adaptive meshless methodology presented in this paper can provide efficiently optimization solutions with a desired accuracy in aerodynamics. Results will be compared with other adaption methods, namely the so called goal oriented method.

1 METHODOLOGY

Based on the success of developing an efficient dynamic cloud technique which maintains the primary clouds of points qualities with rigid moving boundary problems, it is expected to obtain reliable results maintaining the same number of points in the computational domain at each time step or at each modified body shape or position in a design optimization procedure. It is therefore very important to bring an adaptive meshless method to adjust clouds of points automatically.

In past decades, many adaptive mesh schemes have been developed and become important tools for designers to increase the reliability and reduce the cost of numerical computations in many engineering problems. Through an effective adaptive scheme, the discretization error can be reduced via an automatic refinement of the computational region where the accuracy of the numerical solution is low, and therefore the prescribed accuracy [1, 2, 3] is achieved. Hsu et al. [4] proposed an algebraic mesh adaptation scheme based on the concept of arc equidistribution.

1.1 An adaptive meshless method

In this research, an algebraic adaptive meshless scheme based on a weighted reference radius equidistribution is presented. To illustrate this, the difference in 2D Cartesian coordinate system between center point i and its satellite point k can be written as:

$$\begin{cases} \Delta x_{ij} = x_j - x_i \\ \Delta y_{ij} = y_j - y_i \end{cases} \tag{1}$$

For each satellite point j, the distance between center point i and point j is

$$R_{ij} = \sqrt{\Delta x_{ij}^2 + \Delta y_{ij}^2} \tag{2}$$

and the reference radius of cloud C_i is defined as the longest distance between the point i and its satellite point j as

$$R_i = max(R_{i1}, R_{i2}, ..., R_{iM_i}) \tag{3}$$

where M_i is the total number of satellite points around point i. By using the concept of a weighted reference radius equidistribution for mesh from Hsu et al. [4], a weighted

reference radius equidistribution for clouds of points based on the pressure gradient in the flow field can be expressed as:

$$\tilde{R}_i = \frac{1}{1 + \beta \mid \nabla P \mid_i} R_i \tag{4}$$

where β is a constant that controls the sensitivity to the pressure gradient. Thus, the movement of point j should be:

$$\begin{cases} \Delta x'_j = \left(\frac{1}{1+\beta|\nabla P|_i} - 1 \right) \Delta x_j + \Delta x_i \\ \Delta y'_j = \left(\frac{1}{1+\beta|\nabla P|_i} - 1 \right) \Delta y_j + \Delta y_i \end{cases} \tag{5}$$

considering the movement in point $i(\Delta x_i, \Delta y_i)$.

The adaptive meshless method with a moving technique is implemented and tested on a circular arc bump geometry. The geometry of the channel is depicted in Figure 1 and discretized with 2940 clouds of points distributed in the computational domain. Supersonic flow conditions are Mach number 1.6 and angle of attack 0.0°. These flow conditions are high enough to form a normal shock slightly in front of the bump. This shock bends into an oblique shock, which eventually becomes the foremost oblique shock in a lambda shock structure near the upper wall of the channel. The normal shock segment of the lambda shock has a region of subsonic flow behind it, and the rear-most oblique part of the lambda shock downward intersects the lower wall near the trailing edge of the bump. The pressure contours are shown in Figure 2.

After three rounds of moved points adaptation based on the gradient of pressure in the flow field, Figure 3 presents the moved clouds of points distribution in the computational domain. A higher resolution of the shock can be observed in Figure 4. This comparison demonstrates the advantages of the adaptive meshless method.

1.2   Application to shape reconstruction problems

Since the adaptive meshless method achieves higher solution without adding any points in the flow field, one shape reconstruction test case with the above circular arc bump
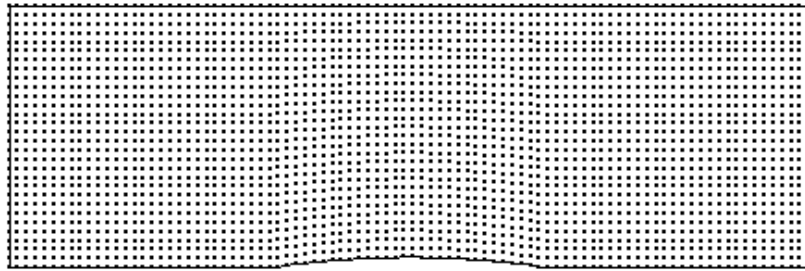
Figure 1: Original clouds of points for the channel with a circular arc bump.
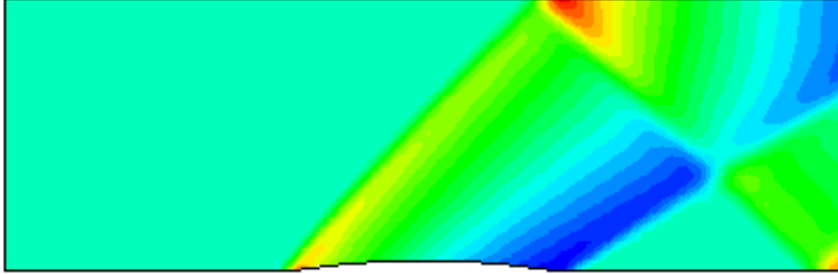
Figure 2: Original pressure contours for the channel with a circular arc bump.
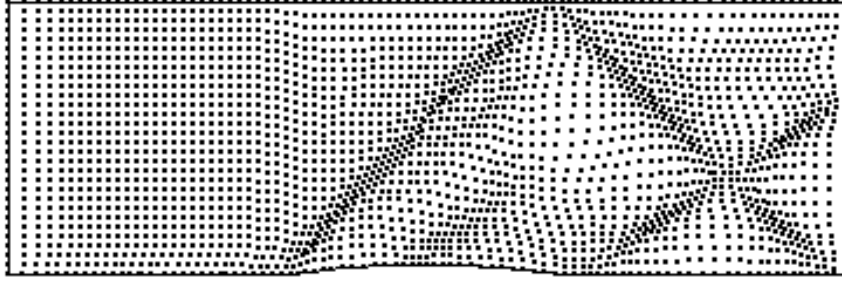
Figure 3: Adapted clouds of points for the channel with a circular arc bump.
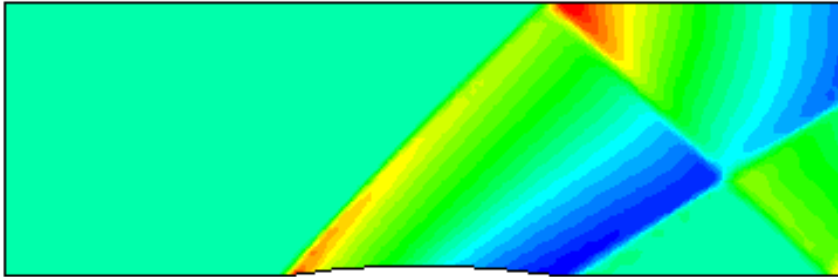
Figure 4: Adapted pressure contours for the channel with circular arc bump.

using adaptive clouds of points is considered in this study. The flow conditions and computational points are the same as in Section 1.1.

Let the thickness of the circular arc bump h be selected as one design parameter. The objective function is defined according to surface pressure coefficients as:

$$\min f(h) = \sum_{i=1}^{M} |C_p(h) - C_p(h^*)|_i^2 \tag{6}$$

where M is the total number of points distributed on the surface of the upper and the

lower channel. The search space for the reconstruction is the interval $h \in [2.0, 8.0]$, and h^* is the targeted design parameter. The parameters value in the GA software are the following: 30 the size of population, 0.85 for the probability of crossover and 0.01 for the probability of mutation. The stopping criteria are the fitness value $f(h) < 10^{-06}$ and the number of generation as 50.

Figure 5 shows the convergence history of the objective function reconstruction procedure. Convergence to zero of the fitness function means that GAs coupled with cloud movement have, within 40 generations, successfully rebuilt the circular arc bump with the targeted thickness. Figure 6 is the comparison of surface pressure coefficients of the targeted value and the obtained result. The red solid line stands for the targeted pressure distribution of 5% bump, and blue dots present the obtained pressure distribution of 5.01% bump. The obtained result is in good agreement with the targeted pressure distribution.
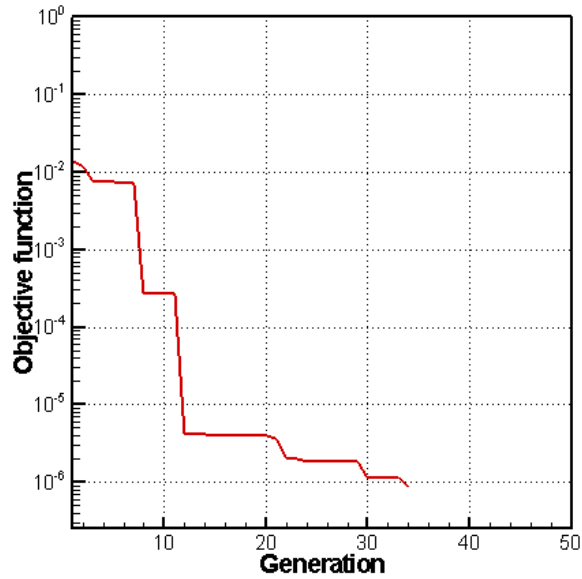
Figure 5: Convergence history of the objective function.

2   CONCLUSIONS

To conclude, the adaptive meshless method coupled with GAs has rebuilt the shape of the targeted circular arc bump based on the prescribed surface pressure. The results presented in this chapter are preliminary and will be consolidated by a measure of the uniformity of level of errors in the flow field to be minimized using sub-clouds and Nash algorithms as a multi-objective problems. The objective functions could be chosen as

f_1, the distance between candidate and prescribed pressure coefficients minimized in L_2 norm, and f_2, uniform level of errors minimized in L_2 norm. Furthermore, distributed optimization coupled with distributed levels of errors can both be run on High Performance Computing (HPC) in the future.
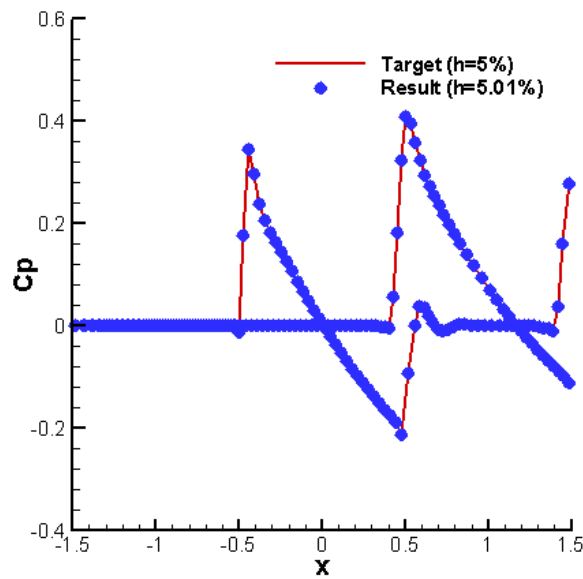
Figure 6: Comparison of pressure distribution of the targeted value and the obtained result.

REFERENCES

[1] Z. Q. Zhu, P. Wang, and X. B. Lu. Adaptive muhigrid solution of the 2D–Euler equations on an unstructured grid. *Acta Mechanica*, 144:43–56, 2000.

[2] A. Anguloa, L. Pérez Pozob, and F. Perazzo. A posteriori error estimator and an adaptive technique in meshless finite points method. *Engineering Analysis with Boundary Elements*, 33(11):1322–1338, 2009.

[3] F. Perazzoa, R. Löhnerb, and L. Perez-Pozo. Adaptive methodology for meshless finite point method. *Advances in Engineering Software*, 39(3):156–166, 2008.

[4] A. T. Hsu and J. K. Lytle. A simple algebraic grid adaptation scheme with applications to two– and three–dimensional flow problems. *AIAA Paper*, 1984, 1989.

ABOUT THE GENERATION OF UNSTRUCTURED MESH FAMILIES FOR GRID CONVERGENCE ASSESSMENT BY MIXED MESHES

JOCHEN WILD[*]

[*] German Aerospace Center (DLR)
Institute of Aerodynamics and Flow Technology
Lilienthalplatz 7, 38108 Braunschweig, Germany
e-mail: jochen.wild@dlr.de, http://www.dlr.de/as

Key words: Grid Generation, Unstructured Meshes, Anisotropic Meshes, Mixed Meshes, CFD.

Abstract. This work describes properties of the mixed mesh approach that are especially suitable for generating families of meshes to assess the grid refinement convergence of CFD solvers. The paper outlines how a regular grid refinement is achieved throughout the domain. The distributions of characteristic grid quality metrics are compared and a grid convergence study is outlined for a commonly used case for outer aerodynamics, the Boeing CRM configuration of the 5th AIAA Drag Prediction Workshop.

1 INTRODUCTION

The assessment of the accuracy of a simulation method is a crucial step during the verification and validation process of the simulation software. Even the best mathematically derived formulation has to prove the rate of convergence with increasing mesh resolution for designated applications.

For applications of CFD for aerodynamics of aircrafts a series of five workshops has been organized under the governance of the American Institute of Aeronautics and Astronautics (AIAA) [1]. A major focus of the comparisons of solvers for simulation accuracy was laid on the grid convergence, for which families of grids have been provided, both structured and unstructured ones. A major conclusion of all workshops has been the highly demanding generation of unstructured mesh families.

While for structured meshes the generation of self-similar grids with different but regularly refined grid resolutions is straight forward, for unstructured hybrid meshes this is a more complicated task. Due to the – in most cases – fully automatic generation a distinct control on local mesh resolution and its influence into other parts of the mesh is hard to control.

This paper outlines the usability of the mixed mesh approach for the purpose of generating families of meshes for grid convergence assessment. Block-unstructured mixed meshes provide structured meshes in the near field of the aerodynamic body where viscous effects dominate and a high resolution normal to the wall is needed. In contrast to pure block-structured meshes, limitations of topologies are overcome by locally using unstructured mesh element types, mainly prismatic elements. For the outer field an a priori defined anisotropic

field triangulation is applied to allow for maximum flexibility and to minimize the effort of user input.

2 THE MIXED MESH FRAMEWORK

The meshing framework used is the formerly structured multi-block grid generation MegaCads developed at DLR [2]. In recent years unstructured capabilities have been introduced. Among these are a parabolic marching procedure to generate prismatic layers based on the same mathematical approach as used in the elliptic smoothing of structured blocks [3], the linking to a number of volume triangulation codes including the SIMMETRIX software [4] used within this work, and a memory efficient way to specify a priori a smooth anisotropic metric field for the triangulation smoothly adopting to the underlying structured and quasi-structured elements [5].

2.1 Mixed mesh approach

(a) (b) (c)

(d) (e)
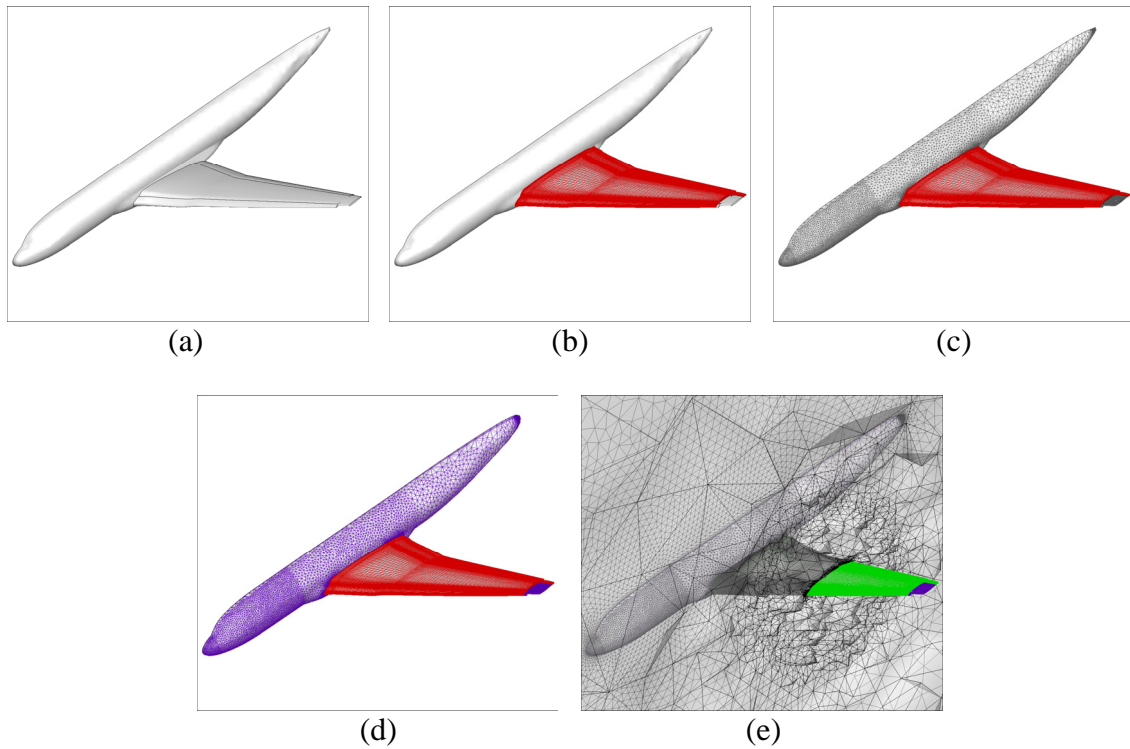
Figure 1: Generating a mixed mesh: (a) surface description; (b) structured hexahedral grids for boundary resolution; (c) surface triangulation of remaining geometry parts; (d) extrusion of prismatic layers; (e) anisotropic field triangulation

The big shortcoming of hybrid unstructured grids is the low anisotropy of surface triangles resulting in a large number of grid points, which is agglomerated through the number of prismatic layers for the boundary layer resolution. This low anisotropy leads to an unnecessary high resolution in span direction, especially for high aspect ratio wings. Recovering the experience with the application of structured grids, it is known, that the aspect ratios of the surface quadrilaterals can be much higher, additionally resulting in well aligned

body conforming meshes. The shortcoming of an overall use of structured meshes is the increasing complexity of the targeted configurations, where structured meshing reaches its limitations, mainly due to grid topology issues.

The simple idea of mixed meshes is to make as much use as possible of the advantages of structured meshes skipping the disadvantages of complicated topology generation. In **Figure 1** the general procedure is sketched for a transport aircraft high-lift configuration. Boundary layer regions are meshed by structured hexahedral mesh blocks, except when the topology gets difficult to be generated. The advantageous C-type meshes for wake resolutions are used here as far as possible. Whenever the block-structured topology gets too complicated, the mixed mesh approach switches to unstructured elements. For the remaining parts of boundary layer regions near surfaces, quasi-structured prismatic elements are used. For the outer flow field the volume is filled by tetrahedrons.

In the past the author showed that mixed meshes significantly reduce mesh sizes without reduction of accuracy [6],[7]. Since the methods work flow starts with the block-structured meshes, the generation method is best described as block-unstructured mixed mesh method. Also commercial grid generation software in the meantime has implemented mixed meshing capabilities, e.g. ICEM CFD [8] and PointWise [9], but those methods rely on isotropic triangulation methods.

2.2 Mixed mesh family around Boeing CRM

Figure 2 shows a view on the grid family around the Boeing Common Research Model (CRM) [10] in the configuration without tail as used in the 5th Drag Prediction Workshop. The coloring depicts the different types of grid elements. For the grid family, three levels of different mesh density have been generated using the anisotropic mixed meshing approach. The number of cells in the structured part is multiplied by a factor of 1.5 for adjacent grid levels in each direction, while the cell sizes are reduced by the same magnitude wherever specified. The portion of the fluid volume meshed by structured elements is not changed in order to obtain self-similar meshes.

The used method derives a smooth anisotropic metric field based on the anisotropy of the adjacent structured hexahedrons and quasi-structured prisms. This anisotropic metric field is inherently responsible to achieve the self-similarity and grid family properties in the unstructured domain of the meshed volume, since all information of grid resolution is directly derived from the structured and quasi-structured grid parts.

Table 1: Characteristics of the mesh family around the Boeing CRM configuration

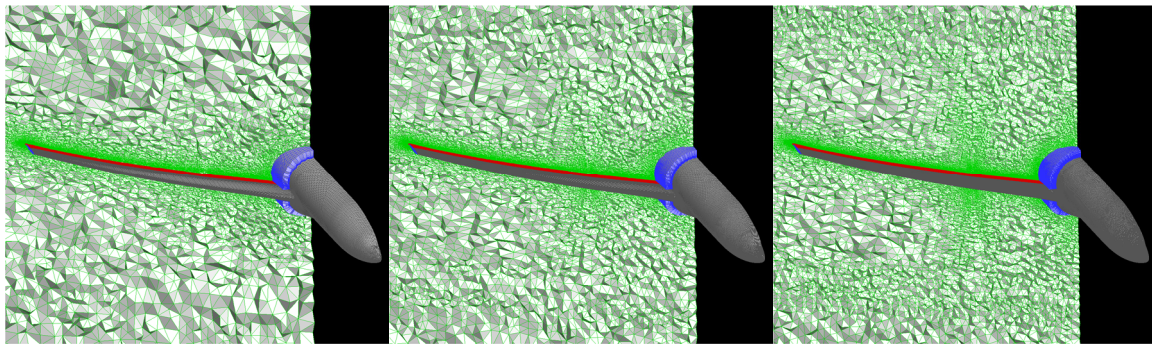| CRM mesh size	coarse	scaling ↔	medium	scaling ↔	fine
grid points	1,368,229	2.68	3,666,721	2.85	10,450,269
surface triangles	38,548	2.20	84,686	2.20	186,370
surface quadrilaterals	27,753	2.24	62,680	2.28	142,134
tetrahedrons	4,222,454	2.19	9,267,026	2.18	20,276,862
Prisms	340,935	3.16	1,078,816	3.26	3,513,840
hexahedrons	419,756	3.46	1,452,208	3.46	5,019,198

Figure 2: Family of grids around the Boeing CRM configuration, left to right: coarse – medium – fine; colors indicate cell type: red=hexahedrons, blue=prisms, green=tetrahedrons

In **Table 1** some characteristic features with respect to grid family properties are shown. The number of surface elements scales between the grid levels close to the theoretical value of $1.5^2=2.25$. Prisms and hexahedrons counts scale also close to the theoretical value of $1.5^3=3.375$. Interestingly, the scale of the number of tetrahedrons is close to the theoretical value for surface elements. Therefore, also the overall point number scales less than theoretically expected or achieved in pure structured meshes. But, interestingly the inverse of the point number to the power of two-third, which is later used for Richardson extrapolation, scales almost exactly by a factor of 2 between grid levels, thus indicating a doubling of grid density.

3   GRID CONVERGENCE OF THE BOEING CRM CASE

3.1 Grid quality indicators

The selection of indicators for a priori grid quality assessment has to respect the type of solver later on used for the simulation. For finite element methods (FEM) most commonly used are indicators that look for the shape of single elements and detecting badly shaped cells like slivers, needles, and hats that inherently disrupt the numerical accuracy of the solution. For the targeted finite volume (FVM) flow solver these element based metrics are of less significance since for the flux computations needed the relationship of neighboring elements is at least as important. Knupp [11] introduced algebraic grid quality metrics based on the Jacobian of elements, which are the transformation matrices from computational into physical space. They are more representative, since the averaging of the cell based Jacobians for a common grid node provide an indication for the smoothness of the surrounding cells. In the following two of the metrics provided by Knupp's MESQUITE library are used to show the comparability of the generated grids in the sense of self-similarity for a family of grids.

Figure 3 shows the histogram of the variations of the "local size" quality metric for the three generated meshes around the Boeing CRM. A value of one indicates that all elements surrounding a grid node have the same size. The counts of grid nodes in the histogram are normalized by the number of overall grid points and are plotted on a logarithmic axis to more precisely inspect the behavior of the different grid levels at values of the lower quality metric values. The figure shows that the distributions of the "local size" quality metric is nearly

identical for all three mesh levels indicating that the characteristics of the grid are not depending on the grid size in this respect.
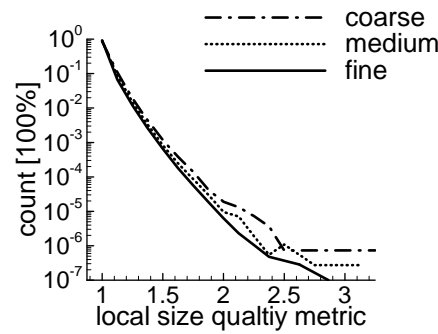

Figure 3: histograms of local size variation of the cells around a grid node for three mesh levels of the grids around the Boeing CRM.

Figure 4 shows a histogram for the vertex based condition number, which is an average of the condition numbers of the Jacobians of the elements surrounding a grid node. Since the targeted meshes are anisotropic high values of are expected since the value directly reflects the anisotropy of the grid. The histogram shows that the generated grids show similar distributions of anisotropy in the mesh and therefore the needed self-similarity.
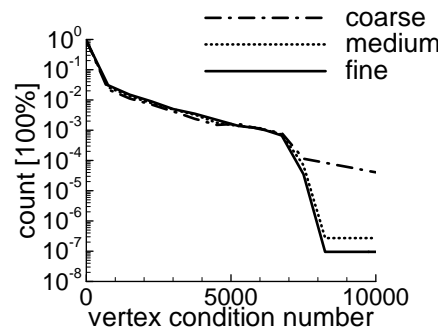
Figure 4: Histogram of the vertex based condition number variation for three mesh levels of the grids around the Boeing CRM.

Another important indicator of the suitability of a grid for CFD is the local number of neighbors to a grid node. Meshes often degrade the simulation quality by having local hot-spots of the neighboring node count. Especially hybrid meshes where the anisotropy of quasi-structured cells is not respected can have up to 200 neighboring nodes and the flux computations are comprised by this. **Figure 5** shows the distribution of the number of neighboring nodes and – again – shows that the characteristics of the three mesh levels are very similar.
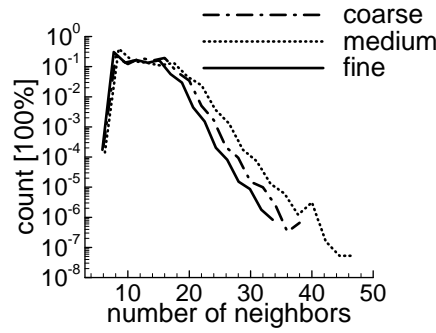
Figure 5: Histogram of the variation of the number of neighbors of a grid node for three mesh levels of the grids around the Boeing CRM.

3.2 Grid convergence assessment

The assessment of the grid convergence is calculated using the DLR TAU-code [12], a second order finite volume CFD solver. The assessment is made based on overall aerodynamic characteristics and on the contributions of local parts to show the suitability of the mixed mesh approach for such studies in global and local effects. Data for comparison is included from wind tunnel tests in the NASA National Transonic Facility [13] and from the result of the 5th AIAA Drag Prediction Workshop (DPW5) [14].
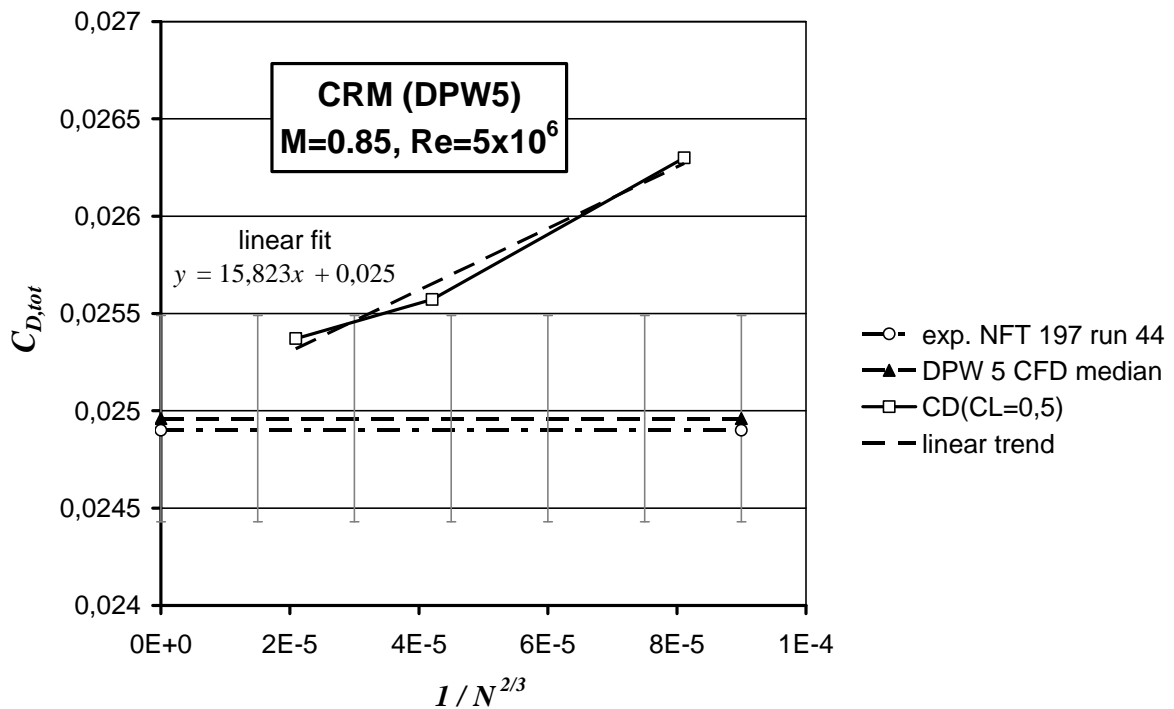
Figure 6: Grid convergence of the total drag coefficient compared to experimental data and the average of CFD calculations performed for the 5th AIAA Drag Prediction Workshop.

Figure 6 shows the grid convergence of the overall drag coefficient compared to experimental data and the average result of DPW5. The value of the drag coefficient is plotted against the inverse of the number of nodes to the power of two-third, according to the Richardson extrapolation for 2nd order CFD codes. Along with the average the standard deviation of the CFD results is shown by error bars. It should be highlighted that the error bar is also derived from the values obtained by Richardson extrapolation to infinite grid density. It is worth to highlight that for the presented meshing strategy already the medium mesh is only slightly off the standard deviation of the CFD calculations and the extrapolated value of the drag coefficient ($C_D = 0.0250$) is only 0.4% off the experimental value. The comparison of the drag coefficients for the distinct grid refinements shows a slight deviation from the theoretical 2nd order behavior of the CFD method.
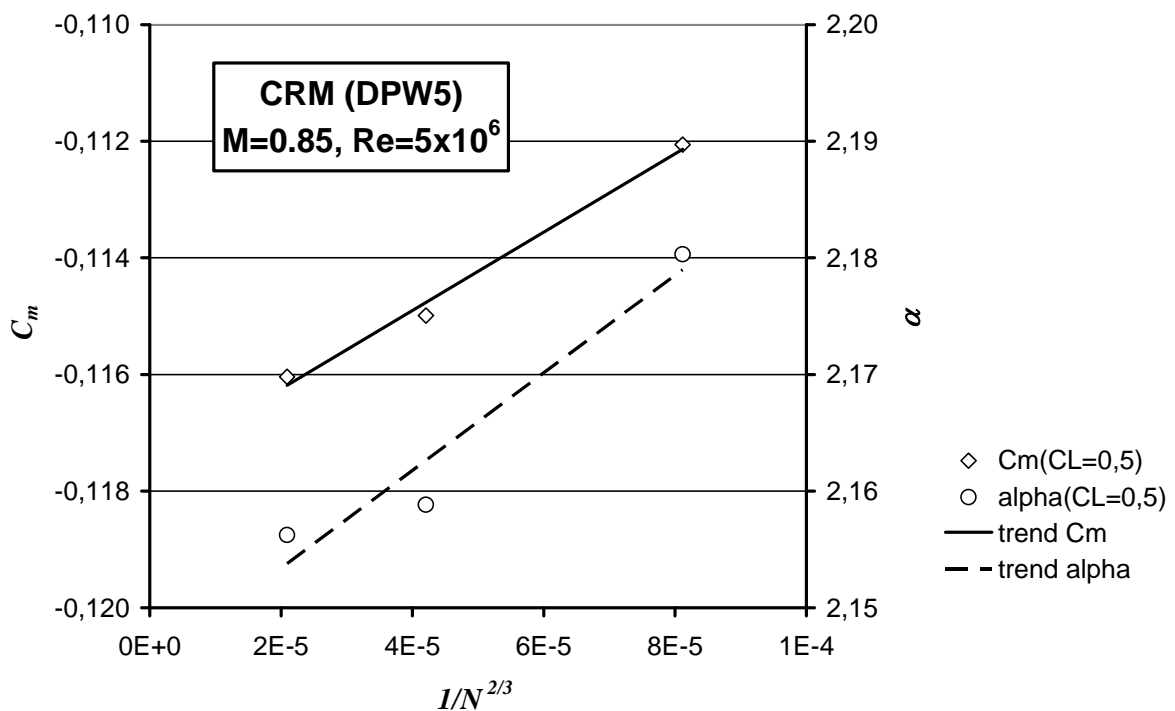
Figure 7: Grid convergence of the pitching moment coefficient and the angle of attack.

Figure 7 shows the grid convergences of the pitching moment coefficient and the angle of attack to obtain a lift coefficient of $C_L = 0.5$. Experimental data is omitted here, since already in the description of DPW5 a large deviation between CFD and experiment was observed. An evaluation of the CFD results similar to the analysis of the drag coefficient is not available, but from the data available it is confirmed that both characteristics fall well into the scatter of CFD results. The pitching moment shows a similar accordance to the 2nd order trend as the drag coefficient, while the angle of attack shows a larger deviation form a 2nd order behavior.

There are in a first view two aspects that can be addressed to more deeply analyze the slight deviation from the theoretical grid convergence in order to see whether the mesh sequence is well suited for the grid convergence study. First, it is worth to differentiate

between the partial contributions the total aircraft drag is composed from. **Figure 8** shows the grid convergence of the drag coefficient separated into pressure and viscous parts. Comparing the trend lines against the distinct values there is no obvious difference in the behavior between the two principal contributors to the overall drag coefficient. This result is especially important to conclude on the meshing strategy to generate the grid family. The grid refinement regarding the structured and quasi-structured parts to resolve the viscous layers of the flow behave the same way, otherwise the grid convergence of the distinct contributions to the drag coefficient would show a behavior different from the pressure part.
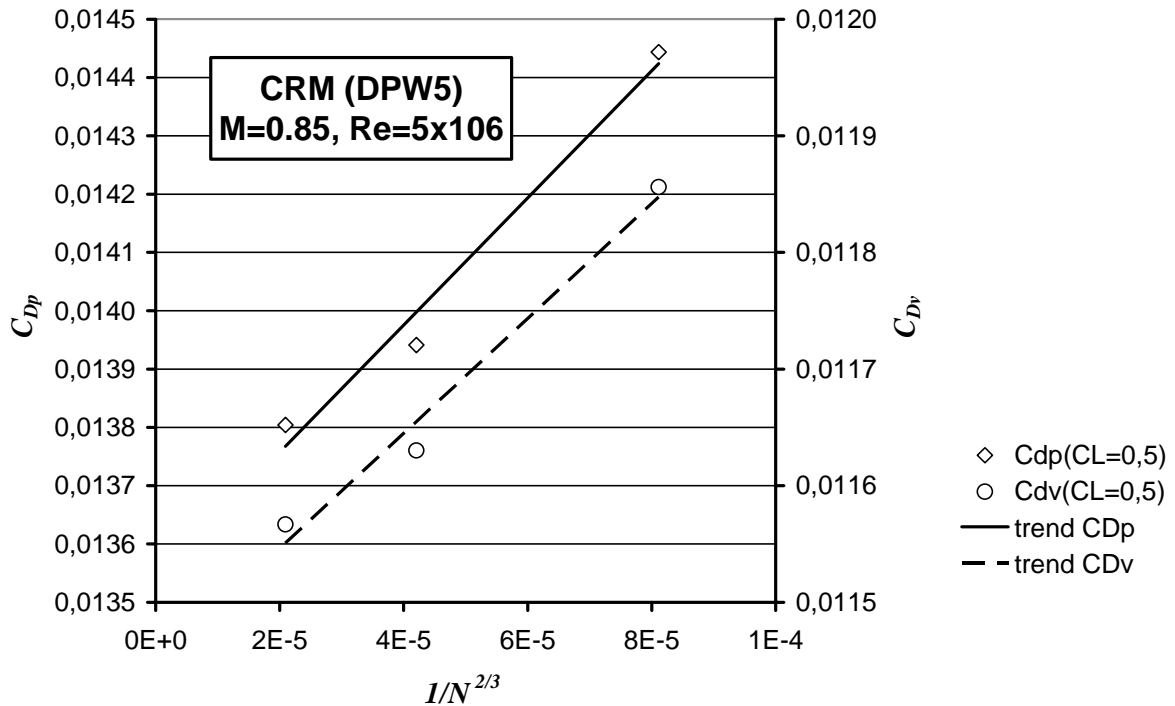
Figure 8: Grid convergence of the drag coefficient divided into pressure and viscous parts.

The second influence to analyze concerns the contribution of the different model bodies to the overall drag estimate. In terms of the meshing strategy this reflects the choice of structured hexahedrons or quasi-structured prisms. **Figure 9** shows the drag split for the two bodies, wing and fuselage respectively. As formerly described, the wing surrounding aside the most outboard wingtip is meshed with structured hexahedrons including the wake of the win while the fuselage is covered by layers of prismatic cells. The figure shows that the deviation from the ideal 2nd order trend line is significantly larger for the fuselage body, while the wing is very close to the 2nd order behavior. This result is not as astonishing as the grid conformity between grid levels can be easily achieved for the structured parts, which includes the complete structured mesh parts as well as the quasi-structured direction of the prism stacks. So this result is not contradictory to the missing influence of the mesh type on the previous shown differentiation between pressure and viscous parts. The behavior observed here highlights that the self-similarity is already limited to some extent within the surface

triangulation of the body.

The behavior observed for the drag coefficient is even more pronounced regarding the contributions of the different bodies to the pitching moment coefficient shown in **Figure 10**. For the contribution of the wing the grid convergence shows a nearly exact 2nd order behavior. It can be therefore concluded that the deviation from the theoretical order can be attributed mainly to the contribution of the fuselage. This confirms the observations obtained for the drag coefficient even more pronounced.
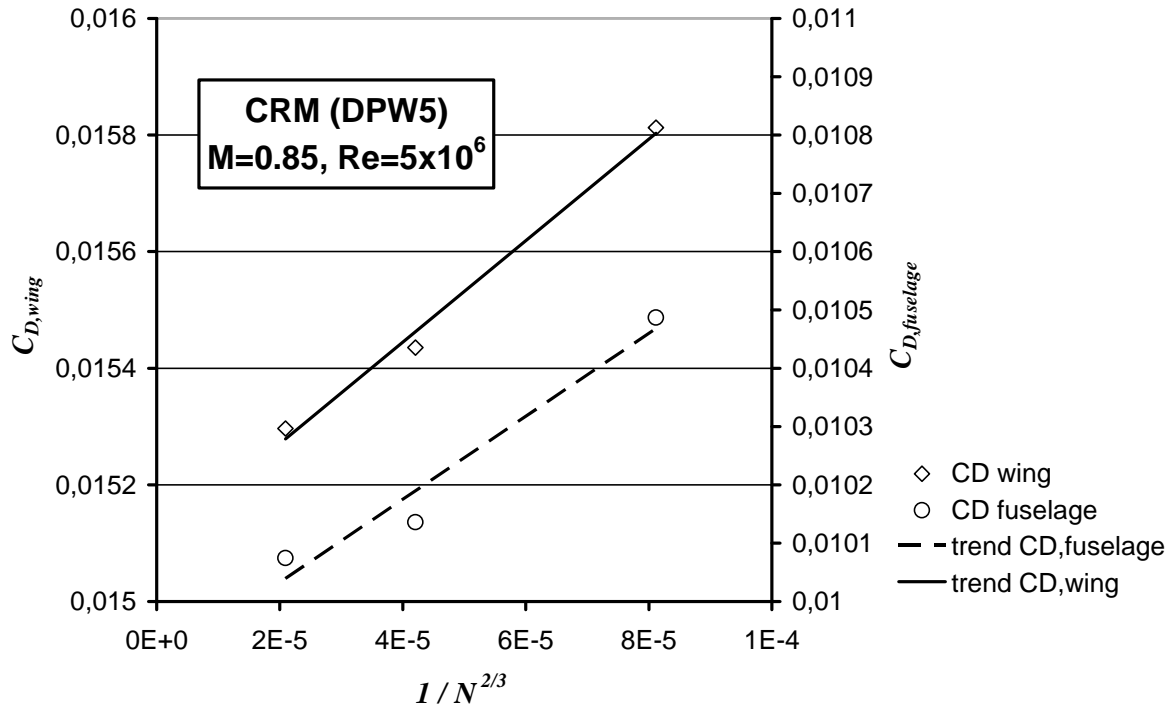
Figure 9: Grid convergence of the drag coefficient separated to contributions of wing and fuselage.

It can be concluded that the formal conformance of grid levels for a grid family can be obtained by the presented meshing approach, but special attendance must be paid to the unstructured parts of the mesh. The generated mesh family fulfills the requirements for a grid convergence study. The results confirm that a basic recommendation for the grid size require that the number of surface element and unstructured tetrahedrons scale with a power of 2 between the grid levels, while the number of structured and quasi-structured elements scale with the power of 3 depending on the refinement ratio.

4 CONCLUSIONS

The mixed meshing approach allows for the generation of mesh families for the assessment of grid convergence behavior of CFD solvers. The use of block-structured cells in near wall regions inherently possesses this capability. The derivation and usage of an adjacent smooth anisotropic metric field for the flow field triangulation promotes the self-similarity of the

structured part of the mesh into the unstructured part. Nevertheless, the result obtained for the differentiation between wing and fuselage indicates that the appropriate meshing of the fuselage is not negligible for the accuracy of the CFD simulation of wing body configurations.
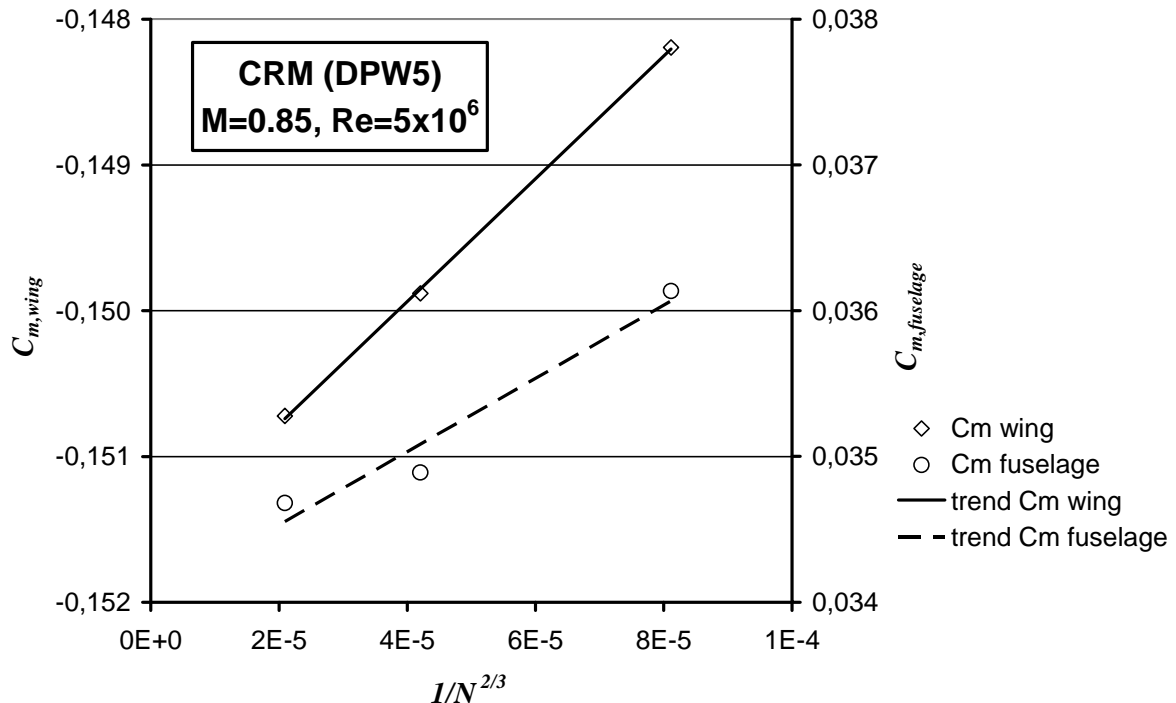

Figure 10: Grid convergence of the pitching moment coefficient separated to contributions of wing and fuselage

REFERENCES

[1] Mavriplis, D.J., Vassberg, J.C., Tinoco, E.N., Mani, M., Brodersen, O., Eisfeld, B., Wahls, R.A., Morrison, J.H., Zickuhr, T. Levy, D., and Murayama, M. "Grid Quality and Resolution Issues from the Drag Prediction Workshop Series", *AIAA Journal of Aircraft* 2009 **46**(3):935-950.
[2] Brodersen, O., Hepperle, M., Ronzheimer, A., Rossow, C.-C. and Schöning, B. "The Parametric Grid Generation System MegaCads", *Proceedings 5th International Conference on Numerical Grid Generation in Computational Field Simulation*, National Science Foundation, (1996), pp. 353-362.
[3] Wild, J. Application Of Smooth Mixed Meshes Based On Metric Identity In Aerospace Analysis And Design. In: Grimella RV (ed) *Proceedings of the 17th Int. Meshing Roundtable*, Springer, Berlin Heidelberg, (2008), pp. 387-398.
[4] Simmetrix Inc., http://www.simmetrix.com, v8.0 -130213, (accessed 2013)
[5] Wild, J. 3D Anisotropic Delaunay Meshing for Ideal Interfacing to Block-Unstructured Mixed Meshes using a Sparse Octree for Metric Size Propagation. *European Congress on Computational Methods in Applied Sciences and Engineering (ECCOMAS 2012)* proceedings on CD-ROM (2012) paper no. 2541.

[6] Wild, J. Acceleration of Aerodynamic Optimization Based on RANS-Equations by Using Semi-Structured Grids, in K.C. Giannakoglou, W. Haase (eds.) *ERCOFTAC Design Optimization: Methods & Applications*, conference proceedings on CD-ROM (2004).

[7] Wild, J. Smooth Mixed Meshes for Acceleration of RANS CFD in Aircraft Analysis and Design, *48th AIAA Aerospace Science Meeting and Exhibit* (2011) AIAA-2011-1267.

[8] Ansys Inc. Home > Products > ANSYS ICEM CFD. http://www.ansys.com/ (accessed 2013).

[9] Pointwise Inc. Pointwise's Unstructured and Hybrid Meshing.
http://www.pointwise.com/ (accessed 2013).

[10] Vassberg, J.V., DeHaan, M.A., Rivers, S.M. and Wahls, R.A. Development of a Common Research Model for Applied CFD Validation Studies. *26th AIAA Applied Aerodynamics Conference*. (2008) AIAA-2008-6919,.

[11] Knupp, P.M. Algebraic Mesh Quality Metrics. *SIAM Journal of Scientific Computing* (2001) **23**(1):193-218.

[12] Schwamborn, D., Gerhold, Th. and Heinrich, R. The DLR TAU-Code: Recent Applications in Research and Industry. *ECCOMAS CFD 2006*, proceedings on CD-ROM (2006).

[13] Rivers, M. and Dittberner, A. Experimental Investigations of the NASA Common Research Model in the NASA Langley National Transonic Facility and NASA Ames 11-Ft Transonic Wind Tunnel. *49th Aerospace Sciences Meeting* (2011) AIAA 2011-1126. Data on Common Research Model available at
http://commonresearchmodel.larc.nasa.gov/, (accessed March 2013).

[14] Levy, D.W., Laflin, K.R., Tinoco, E.N., Vassberg, J.C., Mani, M., Rider, B., Rumsey, C.L., Wahls, R.A., Morrison, J.H., Brodersen, O.P., Crippa, S., Mavripilis, D.J. and Murayama, M. Summary of Data from the Fifth AIAA CFD Drag Prediction Workshop. *51st AIAA Aerospace Sciences Meeting* (2013) AIAA 2013-0046.

ADJOINT BASED A POSTERIORI ERROR ESTIMATES USING DATA COMPRESSION

Tim Wildey[*], **Eric C. Cyr**[†], **and John Shadid**[‡]

[*]Optimization and Uncertainty Quantification Department,
[†]Scalable Algorithms Department,
[‡]Multiphysics Simulation Technologies Department,
Sandia National Labs, Albuquerque, NM.
Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia
Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S.
Department of Energy's National Nuclear Security Administration under contract
DE-AC04-94AL85000.

Key words: data compression, adjoint problem, error analysis, Navier-Stokes

Abstract. One of the major computational burdens in the application of adjoint techniques to time-dependent nonlinear problems is the need to store the full forward approximation to define the adjoint of the linearized forward problem and to evaluate the dual-weighted residual [1, 2]. An approach for mitigating the storage cost is checkpointing, whereby the forward solution is stored at a series of carefully selected time nodes. During an adjoint computations these checkpoints are used to reconstruct the forward solution, recomputing on each subinterval as needed. A notable implementation of the checkpointing scheme is the REVOLVE [3] algorithm. This method minimizes the number of recomputations of the forward solution at any timestep allowing for a fixed storage budget and fixed number of time steps. Recent versions of this algorithm permit a varying number of timesteps with bounded maximum number of recomputations, albeit with a slowly growing storage cost [4]. These approaches successfully reduce the storage cost when the exact solution to the forward problem is required. However, the cost of recomputing the forward solution may be to great of a burden for some applications.

We relax the assumption that the computed forward solution is needed to evaluate the adjoint when used in goal-oriented error estimation. We show that the accuracy of the forward solution has limited effect on the accuracy of the error estimate, and use this to develop a method for approximation of the forward solution that gives reasonably good error estimates. To this end, a number of data compression algorithms are proposed where the storage cost of the approximate foward solution is small compared to storing the exact forward solution. Yet the accuracy of the resulting error estimates are of good quality and no recomputations of the forward solution are required.

REFERENCES

[1] BANGERTH, W., AND RANNACHER, R. *Adaptive Finite Element Methods for Differential Equations.* Birkhauser Verlag, 2003.

[2] ESTEP, D., LARSON, M. G., AND WILLIAMS, R. D. Estimating the error of numerical solutions of systems of reaction-diffusion equations. *Mem. Amer. Math. Soc. 146*, 696 (2000), viii+109.

[3] GRIEWANK, A., AND WALTHER, A. Algorithm 799: revolve: an implementation of checkpointing for the reverse or adjoint mode of computational differentiation. *ACM Trans. Math. Softw. 26*, 1 (Mar. 2000), 19–45.

[4] WANG, Q., MOIN, P., AND IACCARINO, G. Minimal repetition dynamic checkpointing algorithm for unsteady adjoint calculation. *SIAM Journal on Scientific Computing 31*, 4 (2009), 2549–2567.

ADAPTIVE TIME-STEPPING FOR CAHN-HILLIARD-TYPE EQUATIONS WITH APPLICATION TO DIFFUSE-INTERFACE TUMOR-GROWTH MODELS

X. Wu, G.J. van Zwieten, K.G. van der Zee, and G. Simsek

Multiscale Engineering Fluid Dynamics (MEFD)
Eindhoven University of Technology
P.O. Box 513, 5600 MB Eindhoven, Netherlands
e-mail: X.Wu@tue.nl

Key words: Adaptive time-stepping, Second-order time-accurate algorithms, Diffuse-interface tumor-growth model, Cahn-Hilliard equation

Abstract. Many tumor-growth phenomena can be considered as multiphase problems. Employing the continuum theory of mixtures, phase-field tumor-growth models can be derived with diffuse interfaces. The chosen form of the Helmholtz free-energy leads to equations of the Cahn-Hilliard type. Such nonlinear fourth-order partial-differential equations are time-dependent, and their solutions exhibit alternating fast and slow variations in time. It is therefore of prime importance to use adaptive time-stepping to efficiently simulate the entire dynamics of the system [5].

In this contribution, we consider a thermodynamically consistent four-species model of tumor growth in which the energy is non-increasing and total mass is conserved [6]. In order to inherit these two main characteristics of the system at the discrete level, we propose a gradient-stable time-stepping scheme with second-order accuracy [8]. Mixed finite elements are used for spatial discretization. For this discretization, we discuss various adaptive time-stepping strategies in time. Furthermore, we present illustrative numerical results.

1 INTRODUCTION: DIFFUSE-INTERFACE TUMOR-GROWTH MODEL

We consider the diffuse-interface tumor-growth model proposed in Hawkins-Daarud, van der Zee and Oden [6] of the form:

find (u, μ_u, n, μ_n) such that

$$
\begin{aligned}
u_t &= \Delta\mu_u + P(u)(\mu_n - \mu_u) && \text{for } (x,t) \in \Omega \times (0,T] \\
\mu_u &= -\epsilon^2 \Delta u + \Psi'(u) && \text{for } (x,t) \in \Omega \times (0,T] \\
n_t &= \Delta\mu_n - P(u)(\mu_n - \mu_u) && \text{for } (x,t) \in \Omega \times (0,T] \\
\mu_n &= \frac{n}{\delta} && \text{for } (x,t) \in \Omega \times (0,T] \\
\nabla u \cdot \boldsymbol{n} = \nabla\mu_u \cdot \boldsymbol{n} &= \nabla\mu_n \cdot \boldsymbol{n} = 0 && \text{for } x \in \partial\Omega \times (0,T] \\
u(x,0) = u_0, \quad n(x,0) &= n_0 && \text{for } x \in \Omega
\end{aligned}
$$

where u represents the phase of *tumor* and at the same time serves to model the interface, n denotes the phase of *nutrients*, μ_u and μ_n are the chemical potentials corresponding to u and n, respectively, and $P(u) \geq 0$ is a nonnegative *proliferation* function defined as

$$
P(u) := \begin{cases} \delta\hat{P}(1 - u^2) & u \in [-1, 1] \\ 0 & \text{otherwise} \end{cases} \tag{1}
$$

with $\delta > 0$ and $\hat{P} \geq 0$. It is assumed that the domain Ω is a bounded subset of $\mathbb{R}^d$ with $d = 1, 2$ or 3, with Lipschitz boundary $\partial\Omega$. For the sake of simplicity we consider natural boundary conditions, and constant mobility and diffusion (equal to 1). Furthermore, we do not consider chemotaxis; see [6] for details on chemotaxis.

The nonlinear free-energy density function $\Psi(u)$ is a double well potential. We consider the following $C^{2,1}$-continuous Ψ:

$$
\Psi(u) := \begin{cases} (u+1)^2 & u < -1 \\ \dfrac{1}{4}(u^2 - 1)^2 & u \in [-1, 1] \\ (u-1)^2 & u > 1 \end{cases} \tag{2}
$$

Notice that when $\hat{P} = 0$, the system decouples in a Cahn-Hilliard and a diffusion equation.

2   ENERGY DISSIPATION AND MASS CONSERVATION

The total free energy of the tumor-growth model is defined as

$$
E(u, n) := \int_\Omega \left(\frac{\epsilon^2}{2} |\nabla u|^2 + \Psi(u) + \frac{1}{2\delta} n^2 \right) \tag{3}
$$

Similar to the Cahn–Hilliard equation, the tumor-growth model dissipates the total free energy:

$$
\frac{\mathrm{d}}{\mathrm{d}t} E(u(t), n(t)) = -\|\nabla\mu_u\|^2 - \|\nabla\mu_n\|^2 - \int_\Omega P(u)(\mu_n - \mu_u)^2 \leq 0
$$

which is proved in [6]. Furthermore, the total "mass" is conserved, i.e.

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_\Omega (u+n)\,d\boldsymbol{x} = \int_\Omega (\Delta\mu_u + \Delta\mu_n)d\boldsymbol{x} = \int_{\partial\Omega}(\nabla\mu_u\cdot\boldsymbol{n} + \nabla\mu_n\cdot\boldsymbol{n})dS = 0$$

where we employed the homogeneous Neumann boundary conditions.

3 SECOND-ORDER ACCURATE SCHEME

Many problems of interest in the physical and engineering sciences require the understanding of dynamical features which evolve over long-time periods. However, most of standard time-stepping schemes are conditionally energy stable in the sense that the free energy is dissipated at the discrete level only for small enough time steps. This stability issue is the motivation for a large amount of literature on the development of schemes that are provably energy-stable.

Here, we focus on the time-discretization. Therefore we present our ideas without discretizing in space. Of course, one may obtain fully discrete schemes by using finite-difference of finite-element methods in space.

Elliott and Stuart [2, Eq. (5.4)] and Eyre [4] proposed a first-order accurate unconditional energy-stable scheme for gradient-flow systems based on the splitting of E into a convex (contractive) and concave (expansive) part, i.e.

$$E = E_c - E_e$$

where both E_c and E_e are convex.

Following the same idea of energy splitting, we propose in [8] a unconditionally energy-stable second-order time-accurate schemes for the tumor-growth model as follows:

$$\frac{u_{k+1} - u_k}{\tau} = \Delta\tilde{\mu}_u + \tilde{P}_{k+1/2}(\tilde{\mu}_n - \tilde{\mu}_u) \tag{4a}$$

$$\tilde{\mu}_u = \tilde{\Psi}'(u_k, u_{k+1}) - \epsilon^2\Delta\frac{u_{k+1} + u_k}{2} - \alpha_1\tau\Delta(u_{k+1} - u_k) + \alpha_2\tau(u_{k+1} - u_k) \tag{4b}$$

$$\frac{n_{k+1} - n_k}{\tau} = \Delta\tilde{\mu}_n - \tilde{P}_{k+1/2}(\tilde{\mu}_n - \tilde{\mu}_u) \tag{4c}$$

$$\tilde{\mu}_n = \frac{n_{k+1} + n_k}{2\delta} \tag{4d}$$

where

$$\tilde{P}_{k+1/2} := P\left(\frac{3}{2}u_k - \frac{1}{2}u_{k-1}\right) \tag{5}$$

$$\tilde{\Psi}'(u_k, u_{k+1}) = \Psi'_{c,k+1} - \frac{u_{k+1} - u_k}{2}\Psi''_{c,k+1} - \Psi'_{e,k} - \frac{u_{k+1} - u_k}{2}\Psi''_{e,k} \tag{6}$$

This scheme is a modification of the Crank-Nicolson method which includes splitting, stabilization and extrapolation. In particular $\tilde{\Psi}'(u_k, u_{k+1})$ is a novel second-order accurate

splitting employing an implicit Taylor expansion of the convex part Ψ_c' and an explicit Taylor expansion of the concave part Ψ_e'. The α_1-stabilization is an artificial diffusivitiy, while the α_2-stabilization can be thought of as *artificial convexity*. Furthermore, we apply the extrapolation technique for the treatment of $P(u)$, which allows the scheme to be linear (for quadratic $\Psi_c(u)$). The initial step of the scheme done by setting $u_{-1} = u_0$.

We summarize the properties of the above scheme in the following theorem, whose proof can be found in [8].

Theorem: Let the free energy density Ψ be $C^{2,1}$-continuous and have a convex splitting $\Psi = \Psi_c - \Psi_e$ with $C^{2,1}$-continuous Ψ_c and Ψ_e and finite second derivatives, i.e., $|\Psi_c''| \leq L_c$ and $|\Psi_e''| \leq L_e$ for some constants $L_c, L_e \geq 0$. Let the proliferation function P be $C^{0,1}$-continuous and satisfy $0 \leq P \leq \bar{P}$. If the stabilization is large enough, i.e. $\alpha_1 \geq (L_c + L_e)^2/16$ and $\alpha_2 \geq \bar{P}(L_c + L_e)^2/16$, then the time-stepping scheme (4) has the following properties:

1. Unconditional energy-stability: $E(u_{k+1}, n_{k+1}) \leq E(u_k, n_k)$

2. Total mass conservation: $\displaystyle\int_\Omega (u_{k+1} + n_{k+1})d\boldsymbol{x} = \int_\Omega (u_0 + n_0)d\boldsymbol{x}$

3. Second-order accuracy.

Remark: Variable mobilities can also be considered in the tumor-growth model. If they are treated by extrapolation similar to [8], then this theorem also holds, but with the constraint:

$$\alpha_1 \geq \bar{M}(L_c + L_e)^2/16$$

4   ADAPTIVE TIME-STEPPING

In our talk we will consider various adaptive time-stepping strategies: a classical indicator, a comparison indicator and a goal-oriented indicator:

- Classical indicator: $\|u_{k+1} - u_k\| \leq TOL$
  The classical adaptive method has been studied and discussed by Johnson in [7], and extended in the context of parabolic problems by Eriksson et al in [3]. The indicator is based on the a posteriori error estimate in [3], controlling the size of the Galerkin discretization error.

- Comparison indicator: $\|u_{k+1}^{1st} - u_{k+1}^{2nd}\| \leq TOL$
  Based on the optimal a priori error estimates for a class of one-step methods, the idea of the comparison with a higher order method has been discussed by Johnson in [7]. The adaptive strategy is to compare the result between our second-order accurate scheme and the first order accurate scheme in [6]. If the value of the indicators is larger than some tolerance, we refine the time step size.

- Goal-oriented indicator
  A blockwise adaptive approach using adjoint solutions for time-dependent problems has been developed by Carey et al in [1]. Here, we will apply the same idea to time adaptvity, and develop the adjoint-based estimator for the tumor-growth model. Since energy-dissipation is of significant importance, this is selected as our quantity of interest.

We demonstrate that, in various cases, the computing time is reduced by more than one order of magnitude compared to constant time steps.

REFERENCES

[1] V. Carey, D. Estep, A. Johansson, M. Larson, and S. Tavener, *Blockwise adaptivity for time dependent problems based on coarse scale adjoint solutions*, SIAM Journal on Scientific Computing, 32 (2010), pp. 2121–2145.

[2] C. M. Elliott and A. Stuart, *The global dynamics of discrete semilinear parabolic equations*, SIAM journal on numerical analysis, 30 (1993), pp. 1622–1663.

[3] K. Eriksson, C. Johnson, and A. Logg, *Adaptive computational methods for parabolic problems*, vol. 1 of Encyclopedia of Computational Mechanics, Wiley, New York, 2004.

[4] D. Eyre, *An unconditionally stable one-step scheme for gradient systems*, tech. rep., Department of Mathematics, University of Utah, Salt Lake City, Utah, USA, 1997. unpublished.

[5] H. Gomez and T. J. R. Hughes, *Provably unconditionally stable, second-order time-accurate, mixed variational methods for phase-field models*, Journal of Computational Physics, 230 (2011), pp. 5310–5327.

[6] A. Hawkins-Daarud, K. G. van der Zee, and J. Tinsley Oden, *Numerical simulation of a thermodynamically consistent four-species tumor growth model*, International Journal for Numerical Methods in Biomedical Engineering, (2012).

[7] C. Johnson, *Error estimates and adaptive time-step control for a class of one-step methods for stiff ordinary differential equations*, SIAM Journal on Numerical Analysis, 25 (1988), pp. 908–926.

[8] X. Wu, G. J. van Zwieten, and K. G. van der Zee, *Stabilized second-order convex splitting schemes for cahnhilliard models with application to diffuse-interface tumor-growth models*, Submitted, (2012).

EFFECTIVE APPLICATION OF THE EQUILIBRATED RESIDUAL METHOD IN ERROR ESTIMATION OF THE 3D-BASED hp-APPROXIMATED MODELS OF COMPLEX STRUCTURES

GRZEGORZ ZBOIŃSKI[*][†]

[*]Polish Academy of Sciences (PAS)
Institute of Fluid Flow Machinery (IFFM)
14 Fiszera Street, 80-231 Gdańsk, Poland
e-mail: zboi@imp.gda.pl

[†]University of Warmia and Mazury
Faculty of Technical Sciences
11 Oczapowskiego Street, 10-719 Olsztyn, Poland

Key words: A Posteriori Error Estimation, Equilibrated Residual Method, hp-Approximation, Complex Structures

Abstract. This paper recalls our previous research on adaptive modeling and analysis of structures of complex mechanical description. Such complex description results from the application of at least two different models for the structure mechanical characterization. The geometry of the structures can be either complex, i.e. composed of solid, shell and transition parts, or simple – with one geometrical part employed. The numerical models applied in such structures' adaptive modelling and analysis is based on 3D-based hierarchical modelling and hierarchical hp-approximations. The corresponding control of the model and discretization adaptivities takes advantage of the a posteriori error estimation, which is based on the equilibrated residual method (ERM). The method is generalized for the special needs of 3D-based hierarchical models, and is applied to the assessment of the total and approximation error estimators/indicators. The modelling error indicators are obtained as the differences between their total and approximation counterparts. The necessary modifications of the original ERM are the first subject of this paper. These modifications concern both theoretical and implementation aspects. The second subject presented in this paper concerns the parametric studies of the global estimators or indicators of the total, modeling, and approximation error components. Various factors affecting effectivity of the estimation are taken into account.

1 INTRODUCTION

The paper completes our hitherto theoretical research efforts concerning the a posteriori error estimation in the case of complex structures. Such structures include at least two different mechanical models (theories) applied in mechanical characteristics of the structure. Within such structures we apply 3D-based approach utilizing only three-dimensional degrees of freedom. The 3D-based models for complex structures were presented in [1, 2]. The considered estimation method for such models is based on the equilibrated residual approach (ERM) [3] and is applied to the assessment of the global, total and approximation errors. The global modeling error is obtained as a difference of the former two errors. The results from the a posteriori error estimation are assigned for adaptivity control of hierarchical modeling and adaptive analysis of solid mechanics problems. These problems may correspond to either simple or complex mechanical description.

The global modeling error estimate and the element contributions to it allow for the adaptive hierarchical modeling within first order shell, hierarchical shell and the corresponding transition (either shell-to-shell or solid-to-shell) domains of the complex structures. Both, the change of the mechanical model or q-adaptivity are possible – with q denoting the transverse order of approximation within the hierarchical shell models. In the recalled approach also adaptive 2D, 3D or mixed (2D/3D) hp-approximations are possible in the shell, solid and transition domains of the complex structures, respectively, with h standing for the averaged element dimension and p denoting the longitudinal or three-dimensional order of approximation. The element contributions to the estimated global approximation error serve these two types (h and p) of adaptivity.

Taking the above context into account, it is very important to have the estimation method which can satisfy the specific needs of the complex structures 3D-based modeling and analysis and delivers sufficiently accurate estimated values of the global errors, and acceptable element contributions to them as well. In order to satisfy the mentioned needs we adopt the existing algorithms of the equilibrated residual method. So far the method was applied to either the approximation error estimation within three-dimensional elasticity [4] or the total error estimation of the conventional hierarchical shell models [5, 6]. Also the approximation error estimation for the 3D-based first-order shell models is available [1, 7, 8]. Here we extent the application of the residual equilibrated method onto the estimation of the total error of the 3D-based first-order shell model, as well as the estimation of the total and approximation errors of the 3D-based hierarchical shell and transition models, both skipped in our previous works. In particular we show how to apply this method to the 3D-based (constrained) shell model of the first order and the corresponding constrained transition models as well. Such an application needs different equilibration procedure than the three-dimensional equilibration applied to the 3D-elasticity or hierarchical shells, where the equilibration is performed in the global directions. The adopted approach requires introduction of the local nodal coordinate systems, and different treatment of the constrained and unconstrained directions. Note

that some key aspects of this approach were presented in [9]. The present paper provides readers with some additional, theoretical and algorithmic, remarks or hints which were not published in the cited paper.

In order to assess the quality of the equilibrated residual method we compare three versions of the method. The differences between them result from the different definitions of the element local problems in these three versions. The collection of solutions to such problems constitutes the estimate of the exact global solution. In the first of the applied versions, we average the interelement stress fluxes. In the second case we perform linear (at the element vertex nodes) equilibration of these fluxes. In the third version we constrain local problems at element vertices by means of the displacement values obtained from the global numerical solution. Then, for the most effective case (the third one), we perform unique parametric studies of the modeling, approximation and total error estimations. These studies include such important factors as: the problem type, the applied mechanical model, and the mechanical complexity of the model. Our studies are completed with an analysis of the results. This analysis leads to practical hints concerning the appropriate definitions of the local problems, so as to assure the most effective error estimation within complex structures. Note that the numerical results presented here illustrate some of the tabular data published in [9].

2   MODIFICATION OF THE ERM FOR COMPLEX STRUCTURES

In order to apply the equilibrated residual methods (ERM) to the 3D-based models of complex structures, one has to include the following changes.

Firstly, the equilibration procedure has to be performed in the local nodal directions for the shell vertex nodes of the shell elements (see [7, 9]) and such nodes of the shell parts of the transition elements as well (compare [9]). These nodal directions are consistent with the shell mid-surface normal and tangent directions. In the normal direction different shape function definition has to be applied in the equilibration procedure so as to take into account the Reissner-Mindlin kinematic constraints of lack of elongation of the normals to the mid-surface. In the case of the solid nodes of the solid or hierarchical shell elements, and in such nodes of the solid or hierarchical shell parts of the solid-to-shell or shell-to-shell transition elements as well, the equilibration can be performed in the standard way [1, 3, 4], i.e. in the global directions. The standard 3D vertex shape functions are applied in this case.

And secondly, in the local ERM problems of the elements of the regular meshes, the nodal forces due to the equilibrated interelement stress fluxes have to be defined as global ones. In the case of the solid nodes, one can use the global splitting factors and global components of the interelement stresses for these forces determination. The applied shape functions are 3D vertex ones. On the contrary, in the case of the shell nodes, one has to utilize the local splitting factors and local components of the interelement stresses. One may apply the same shape functions as before, apart from the third local direction, where one has to apply the modified vertex shape functions, accounting for the Reissner-Mindlin

constraints. Finally, the products of the local factors and stresses have to be transformed to the global system of coordinates.

In the case of the irregular meshes, resulting from the local element subdivisions, one has to take into account that for the hanging (constrained) vertex nodes of the element obtained with the subdivision, the splitting factors are expressed through the corresponding factors of the vertex unconstrained (active) nodes of the undivided parent (or neighbouring) element. Examples of the corresponding relations, for the solid nodes, in the case of 2D- and 3D-problems, can be found in [4, 10] and [1], respectively. In this paper we extend this approach onto the shell nodes of the first-order shell and transition elements.

3   PARAMETRIC STUDIES OF THE ESTIMATORS

Here we take advantage the benchmark examples used by us elsewhere [1, 2, 9]. The first two examples concern a bending-dominated plate and a bending-dominated half-cylindrical shell. Both structures are aligned horizontally. The third example corresponds to a membrane-dominated cylindrical shell. The length of straight edges of the plate and shells are equal to $2l = 3.14 \cdot 10^{-2}$ m, the curved edges of the shells are equal to $2l = \pi r = 3.14 \cdot 10^{-2}$ m, with $r = 1.0 \cdot 10^{-2}$ m. The thickness of the structures equals $t = 0.15 \cdot 10^{-2}$ m. The plate is clamped. Also the straight edges of the first shell are clamped, while the curved ones are free. There is no rotation along the curved edges of the second shell. The two bending-dominated examples are loaded vertically with the uniform traction $p = 4.0 \cdot 10^4$ N/m^2, while the membrane-dominated shell is loaded with the internal pressure of the same magnitude. Due to the symmetry of the geometry, loading and boundary conditions we analyse only a quarter of the bending-dominated structures and one-eighth of the membrane-dominated one.

3.1   Dependence of effectivities on the local problem definition

We analyze the residual method local problems of three types in this section. In figs. 1 and 2 we present the results obtained with averaging of the interelement stresses, for two definitions of the discretization parameters H, P and Q in the local problems. Here H, P and Q are the local problem counterparts of the global parameters h, p and q. The first definition is: $H = h$, $P = p$, $Q = q + 1$, while the second one reads: $H = h$, $P = p + 1$, $Q = q + 1$. Subsequently, figs. 3 and 4 correspond to equilibration of the interelement tractions in the local problems for the two definitions, while figs. 5 and 6 correspond to constraining the vertex nodes of the elements with the values of displacements from the global problem. Note that in the first four cases the elements are constrained with six global displacements so as to remove three rigid body translations and three rigid body rotations. The presented results concern the approximation, modeling and total global error effectivity indices and correspond to the plate example. The calculations were performed for the hierarchical shell model MI of the second order ($I \equiv q = 2$) and
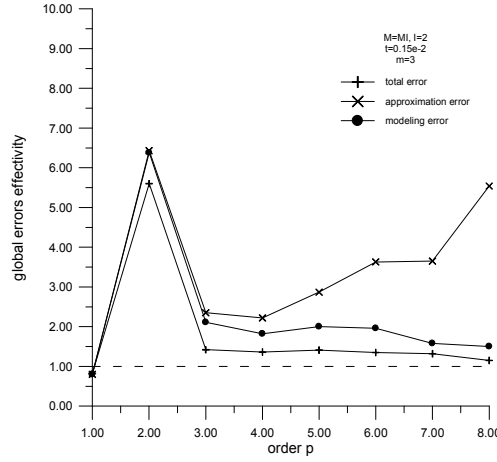
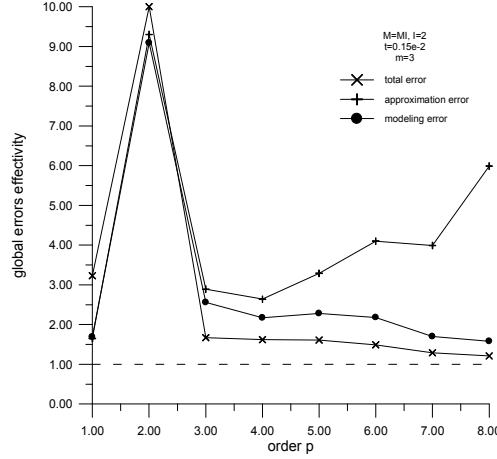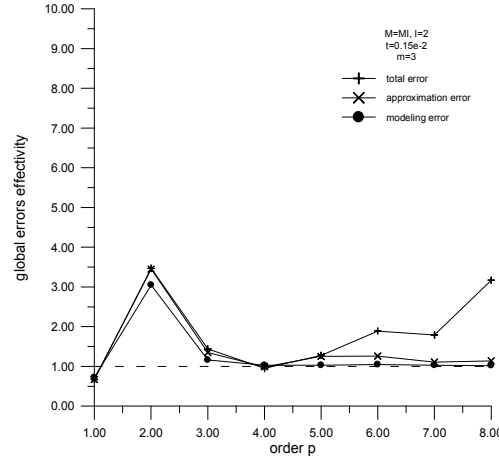**Figure 1**: Effectivity indices in the case of stress averaging ($H = h$, $P = p$, $Q = q + 1$)

Figure 2: Effectivity indices in the case of stress averaging ($H = h$, $P = p + 1$, $Q = q + 1$)

the uniform mesh division into $2m^2$ prismatic elements ($m \equiv l/h = 3$).

Analyzing all of the presented results, one can notice the worsening of the approximation error effectivity indices with low values of p. Also high values of p lead to worse values of the approximation error effectivities. This two observations can be related to the presence of the numerical locking for $p = 2$ and the influence of the boundary layer phenomena for $p \geq 6$, respectively. Please note that the total and modelling error effectivities are affected by the locking phenomenon only.

The second observation is that, even though the equilibration delivers better effectivities than the averaging, neither the averaging nor the equilibration provide the satisfactory results of the residual-based global error estimation, as the effectivity indices are far above the desired value of 1.0. The best results are obtained for the constraining the local problems with the global displacements, with $H = h$, $P = p + 1$ and $Q = q + 1$. Because of that, our further numerical tests will be limited to this particular case.
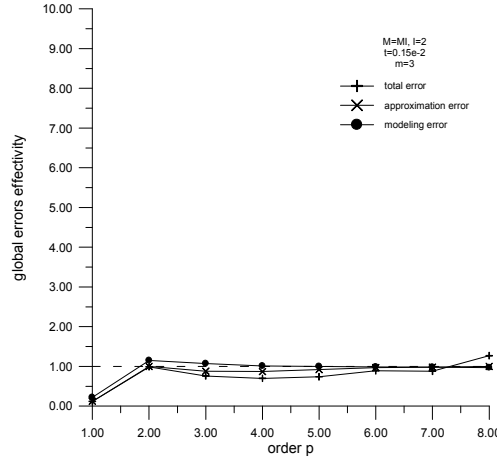
Figure 3: Effectivity indices in the case of stress equilibration ($H = h$, $P = p$, $Q = q + 1$)
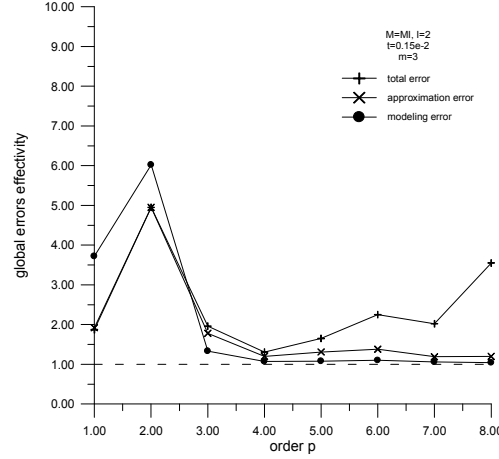
Figure 4: Effectivity indices in the case of stress equilibration ($H = h$, $P = p + 1$, $Q = q + 1$)

3.2 Influence of the problem type on effectivities

In order to determine the influence of the problem type (plate or shell, bending or membrane dominance) on effectivities we compare the corresponding results for the plate (the bending-dominated one), the bending-dominated shell, and the membrane-dominated shell as well. In the latter case we consider two problems, corresponding to hierarchical shell models MI of the first ($I \equiv q = 1$) and second ($I \equiv q = 2$) order, as for the membrane-dominated structures the improper solution limit phenomenon does not appear. In the case of the two bending-dominated examples only $I \equiv q = 2$ is possible, because of this phenomenon appearance. In all examples we apply $m = 4$. The results, corresponding to the four respective cases, are presented in figs. 7, 8, 9 and 10.

Analyzing the results one can notice that for the bending- and membrane-dominated shell examples the approximation error effectivity is worse than for the plate. Only, the

Figure 5: Effectivity indices in the case of vertex constraints ($H = h$, $P = p$, $Q = q + 1$)
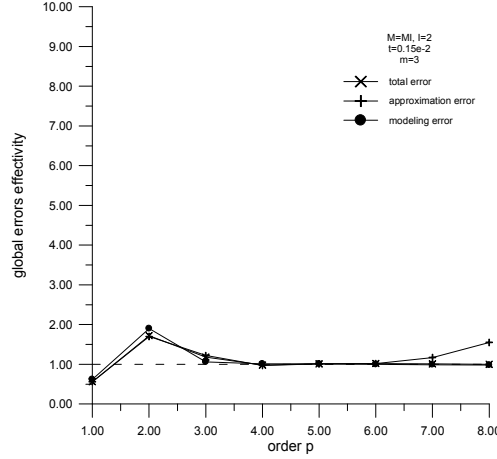
Figure 6: Effectivity indices in the case of vertex constraints ($H = h$, $P = p + 1$, $Q = q + 1$)

bending-dominated shell example is sensitive to the locking ($p = 2, 3$) and boundary layer phenomena ($p \geq 6$), in the way qualitatively similar to the plate example. Both membrane-dominated examples produce similar results. No influence of the locking and the boundary layer is observed.

3.3 Influence of the model on effectivities

Here we compare the results presented in the previous subsection, corresponding to our three model problems and 3D-based hierarchical shell model MI, with the analogous results obtained for the 3D-based Reissner-Mindlin model RM of the plate and shells. The respective results for $m = 4$, in the case of the plate and bending- and membrane-dominated shells are shown in figs. 11, 12, 13, respectively.

It can be noticed that the RM model results are less sensitive to the locking phenomenon for the applied density of the mesh, due to better regularity of this model in
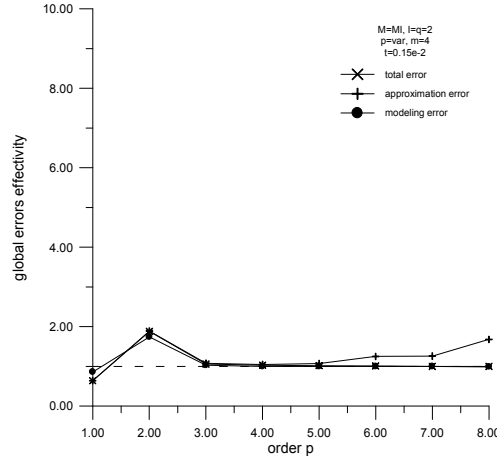
Figure 7: Plate problem effectivities versus the order p ($M = M2$, $t = 0.15 \cdot 10^{-2}$ m)
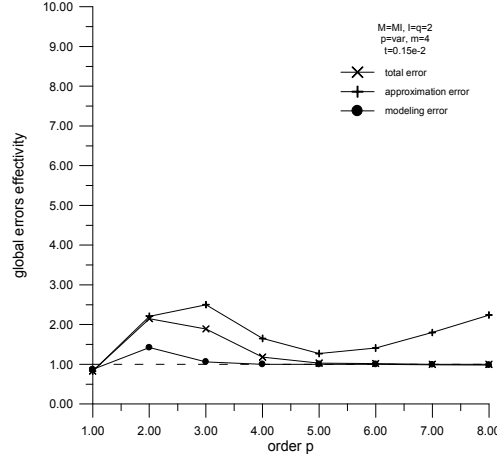

Figure 8: Half-cylindrical shell problem effectivities vs. p ($M = M2$, $t = 0.15 \cdot 10^{-2}$ m)

comparison to the hierarchical MI model. Moreover the influence of the boundary layers is not present for this model. This observation is consistent with the theory for the Reissner-Mindlin model.

The second observation is that in the case of the two bending-dominated examples the total and modeling error effectivities are below 1.0 and equal to about 0.8.

3.4 Effectivities versus model complexity

In this subsection we introduce complex mechanical description of our plate and shell examples. This means that the hierarchical shell model MI ($I \equiv q = 2$), the first-order shell model RM ($q = 1$) and the 3D-based shell-to-shell transition model MI/RM ($q = 1, 2$) are employed for each model structure. In the case of the plate, the RM square domain is symmetric and located in the interior of the plate. The lengths of a quarter of this domain are equal to $l_{RM} = l/2$. In the case of the bending-dominated half-cylindrical
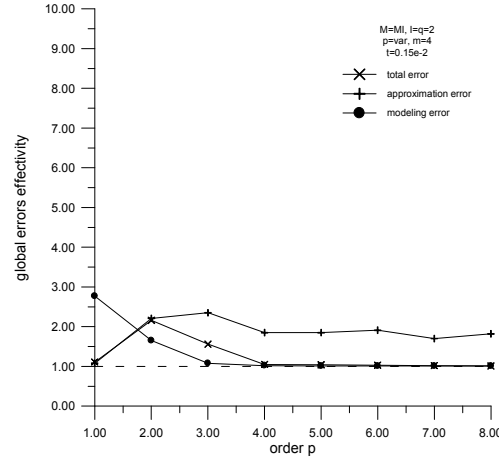
Figure 9: Cylindrical shell problem effectivities versus p ($M = M2$, $t = 0.15 \cdot 10^{-2}$ m)
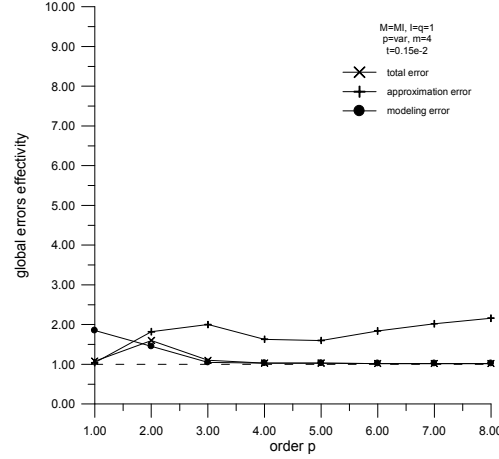

Figure 10: Cylindrical shell problem effectivities versus p ($M = M1$, $t = 0.15 \cdot 10^{-2}$ m)

shell, the MI zone is aligned along the straight clamped boundary of the shell. The curved boundary of a quarter of the shell is divided by two, i.e. $l_{RM} = l/2 = \pi r/4$. In the case of the membrane-dominated shell, the straight boundary of one-eighth of the shell is divided by two, i.e. $l_{RM} = l/2$, and the MI zone is aligned along the external curved boundary of the shell. The MI zones of the plate and shells are joined with the RM zones with one layer of the transition elements MI/RM, forming the transition domain TR.

Comparing the complex models' results, presented in figs. 14, 15, 16 and 17, with the corresponding results for the pure MI models (figs. 7, 8, 9 and 10), one can see their close similarity. The only difference is that now, in the case of the two bending-dominated examples, the total and modeling error effectivities are slightly below 1.0, due to the presence of the RM zones in the complex models of the structures.
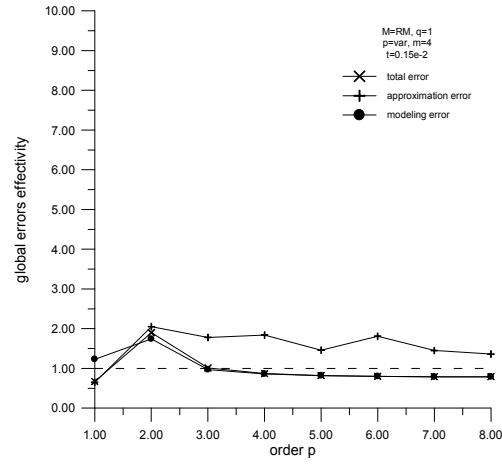
Figure 11: Plate problem effectivities versus the order p ($M = RM$, $t = 0.15 \cdot 10^{-2}$ m)
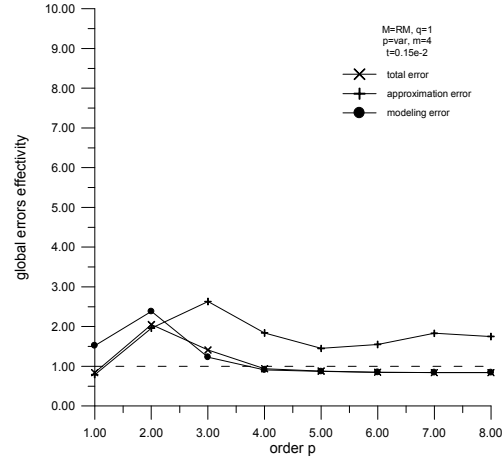

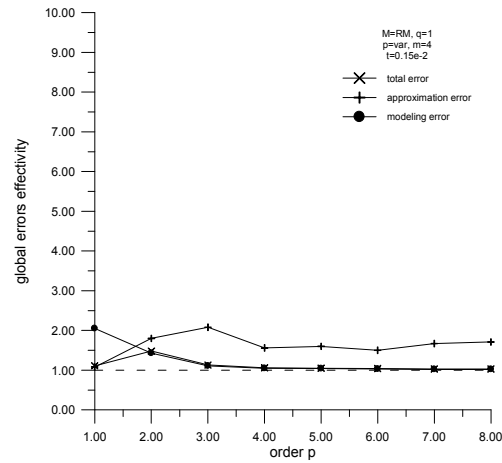**Figure 12**: Half-cylindrical shell effectivities versus p ($M = RM$, $t = 0.15 \cdot 10^{-2}$ m)

Figure 13: Cylindrical shell problem effectivities versus p ($M = RM$, $t = 0.15 \cdot 10^{-2}$ m)
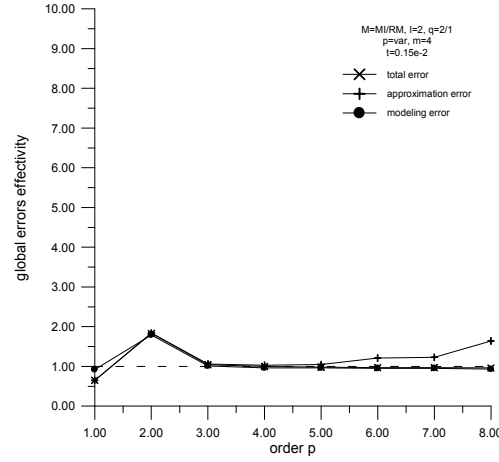
Figure 14: Plate problem effectivities versus p ($TR = M2/RM$, $t = 0.15 \cdot 10^{-2}$ m)
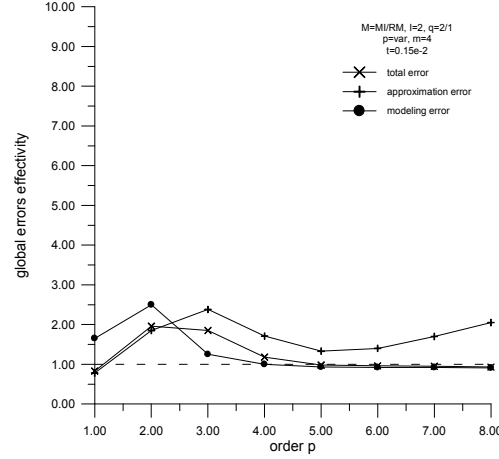

Figure 15: Half-cylindrical shell effectivities versus p ($TR = M2/RM$, $t = 0.15 \cdot 10^{-2}$ m)

4   CONCLUSIONS

The theoretical findings, which concern the algorithms of the residual-based error estimation of the structures of 3D-based complex mechanical description, are as follows.

- The equilibration procedure, proposed in the case of the complex structures, needs distinction between the solid and shell vertex nodes, as in the latter case the application of the modified shape functions, which account for the Reissner-Mindlin constraints, is necessary. Also the application of the local directions, perpendicular and tangent to the shell mid-surface, is necessary in this case.

- The definitions of the vertex nodal forces, entering the ERM local problems and representing the equilibrated interelement stress fluxes, are also dependent on the vertex node type. In the case of the shell nodes, the local directions and the modified shape functions are applied again.
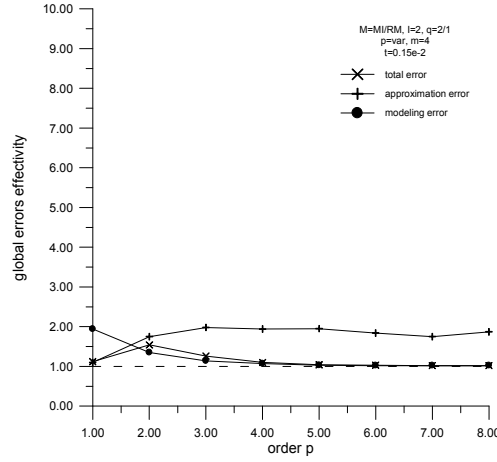
Figure 16: Cylindrical shell problem effectivities vs. p ($TR = M2/RM$, $t = 0.15 \cdot 10^{-2}$ m)
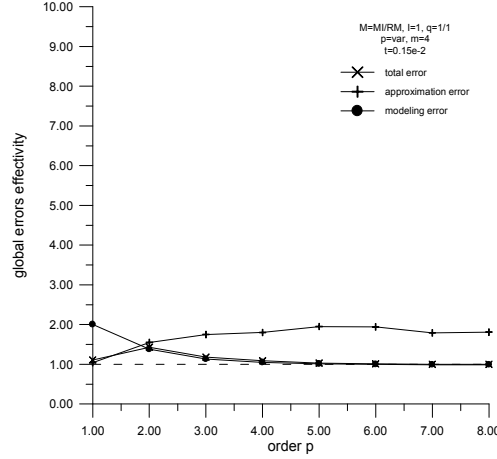

Figure 17: Cylindrical shell problem effectivities vs. p ($TR = M1/RM$, $t = 0.15 \cdot 10^{-2}$ m)

- In the case of the element local problems with hanging (constrained) nodes, the splitting functions and factors are expressed with the corresponding factors of the bigger undivided elements. The distinction between the solid and shell nodes may be in use again, i.e. both the local and global splitting factors may enter the calculation of the forces acting in the element vertex hanging nodes. Such complex situations happen for the h-refined transition elements.

The conclusions, concerning parametric studies of the error estimation with the element residual methods, can be formulated as follows.

- The version of the element residual method based on the constraints defined with the global problem vertex displacements, with $H = h$, $P = p+1$, $Q = q+1$ applied in the local element problems, delivers the effectivity results closest to the desired value of 1.0. This version is better than the two approaches based on the averaging

or equilibration of the interelement stresses in the local problems.

- In the case of this constrained version of the ERM local problems, all three global effectivities are above 1.0 for the purely hierarchical models MI of the structures. In the case of the bending-dominated RM and complex models of the structures, the total and modelling error effectivities are below 1.0.

REFERENCES

[1] Zboiński, G. *Hierarchical Modeling and Finite Element Approximation for Adaptive Analysis of Complex Structures.* D.Sc. Thesis (in Polish), 520/1479/2001, Institute of Fluid Flow Machinery, Gdańsk (Poland), (2001).

[2] Zboiński, G. Adaptive *hpq* finite element methods for the analysis of 3D-based models of complex structures. Part 1. Hierarchical modeling and approximation. *Comput. Methods Appl. Mech. Engrg.* (2010) **199**:2913-2940.

[3] Ainsworth, M. and Oden, J.T. *A posteriori error estimation in finite element analysis.* Wiley & Sons, Inc., New York, (2000).

[4] Ainsworth, M. and Oden, J.T. and Wu W. A posteriori error estimation for *hp* approximation in elastostatics. *Appl. Numer. Math.* (1994) **14**:23-55.

[5] Oden, J.T. and Cho, J.R. *Local A Posteriori Error Estimation for Hierarchical Models for Plate- and Shell-Like Structures.* TICAM Report 95-02, The University of Texas at Austin, Austin (TX), (1995).

[6] Oden, J.T. and Cho, J.R. Adaptive finite element methods of hierarchical shell models for plate- and shell-like structures, *Comput. Methods Appl. Mech. Engrg.* (1996) **136**:317-345.

[7] Zboiński, G. A posteriori error estimation for *hp*-approximation of the 3D-based first order shell model. Part 1. Theoretical aspects. *Applied Mathematics, Informatics and Mechanics* (2003) **8**(1):104-125 (http://www.viam.hepi.edu.ge/Ami/).

[8] Zboiński, G. A posteriori error estimation for *hp*-approximation of the 3D-based first order shell model. Part 2. Implementation aspects. *Applied Mathematics, Informatics and Mechanics* (2003) **8**(2):58-83 (http://www.viam.hepi.edu.ge/Ami/).

[9] Zboiński, G. Adaptive *hpq* finite element methods for the analysis of 3D-based models of complex structures. Part 2. A posteriori error estimation. Submitted to: *Computer Methods in Applied Mechanics and Engineering.*

[10] Ainsworth, M. and Oden, J.T. A unified approach to a posteriori error estimation using element residual methods. *Numer. Math.* (1993) **65**:23-50.

This book contains the Abstracts of the papers presented at ADMOS 2013, the sixth International Conference on Adaptive Modeling and Simulation, held in Lisbon from June 3 to 5, 2013.

Numerical Modeling and Simulation is increasingly used as a complement to Experimental Modeling and Analysis and as a design or certification tool in engineering applications. However, after more than thirty years of worldwide research efforts around Adaptive Modeling and Simulation, the problem of properly assessing and controlling the quality of the numerical solutions is still relevant, as the design of sophisticated engineering systems requires increasingly complex and coupled modeling, which leads to increasingly time-consuming computations. Adaptive approaches, which provide reliable and cost efficient modeling and coherent coupling of different scales and mechanisms in a unique model, are more strategic and indispensable than ever.

TÉCNICO LISBOA

UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH
UPC

E-CAero

ATCoMe
ITN in Advanced Tecniques in Computational Mechanics
PEOPLE

CIMNE