



HAL
open science

Anisotropic norm-oriented mesh adaptation for a Poisson problem

Gautier Brèthes, Alain Dervieux

► **To cite this version:**

Gautier Brèthes, Alain Dervieux. Anisotropic norm-oriented mesh adaptation for a Poisson problem. Journal of Computational Physics, 2016, 322, pp.804 - 826. 10.1016/j.jcp.2016.07.008 . hal-01413431

HAL Id: hal-01413431

<https://inria.hal.science/hal-01413431>

Submitted on 9 Dec 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Anisotropic Norm-Oriented Mesh Adaptation for a Poisson problem

Gautier Brèthes^a, Alain Dervieux^{a, b}

^a*INRIA, Projet Tropics, 2004 route des lucioles - BP 93,
06902 Sophia Antipolis Cedex, France*

^b*Corresponding author: alain.dervieux@inria.fr*

Abstract

We present a novel formulation for the mesh adaptation of the approximation of a Partial Differential Equation (PDE). The discussion is restricted to a Poisson problem. The proposed norm-oriented formulation extends the goal-oriented formulation since it is equation-based and uses an adjoint. At the same time, the norm-oriented formulation somewhat supersedes the goal-oriented one since it is basically a solution-convergent method. Indeed, goal-oriented methods rely on the reduction of the error in evaluating a chosen scalar output with the consequence that, as mesh size is increased (more degrees of freedom), only this output is proven to tend to its continuous analog while the solution field itself may not converge. A remarkable quality of goal-oriented metric-based adaptation is the mathematical formulation of the mesh adaptation problem under the form of the optimization, in the well-identified set of metrics, of a well-defined functional. In the new proposed formulation, we amplify this advantage. We search, in the same well-identified set of metrics, the minimum of a norm of the approximation error. The norm is prescribed by the user and the method allows addressing the case of multi-objective adaptation like, for example in aerodynamics, adapting the mesh for drag, lift and moment in one shot. In this work, we consider the basic linear finite-element approximation and restrict our study to L^2 norm in order to enjoy second-order convergence. Numerical examples for the Poisson problem are computed.

Key words: goal-oriented mesh adaptation, anisotropic mesh adaptation, adjoint, metric, Poisson problem, finite elements

1. Introduction

This paper addresses anisotropic mesh adaptation. We focus on methods which build a somewhat optimal mesh defined by a parametrization using a Riemannian metric. A typical family of optimal metric-based methods for CFD is the family of Interpolation-based/Hessian-based methods. An attractive property of these methods is that they are based on a mathematical optimization principle.

Iso-distribution /equi-repartition Hessian-based methods tend to minimize a Sup or L^∞ norm of the (main term of) interpolation error with respect to a metric considered in a subset of metrics with a prescribed number of vertices. We refer to the two pioneering works [13, 16] for the methods, the two pioneering works [1, 31] for the analysis, and to [32, 4]. The methods minimizing the L^p norm ($p < \infty$) of the interpolation error of one or several *sensors* depending on the CFD solution allows to better capture features of different scales in the solution. Cf. [34, 22, 14, 23, 5]. Sensors are solution-dependant fields chosen by the user according to their ability to take into account mesh-resolution difficulties of the flow to compute.

The Hessian-based methods are particularly well adapted to the finite-element approximation of second-order elliptic PDEs. It is true, by the projection theorem, that a norm of the approximation error is bounded by the analog norm of the interpolation error but this concerns the H^1 norm while Hessian-based methods concentrate on L^∞ or L^p norms. More generally, while taking into account the features of the PDE *solution*, these methods do not take into account the features of the *PDE* itself. This is penalizing in the case of systems, for which several sensors need to be chosen and weighted by the user. However, if sensors are wisely chosen, a good convergence of the whole approximate solution field to the exact solution field is usually observed.

Taking into account the influence of the PDE on the error through an *equation-based estimate* has also been an important research topic. The formulation of *goal-oriented methods* is an important step for a more justified error evaluation. It has been introduced in [7]. It relies on an *a posteriori* estimate. A good synthesis concerning *a posteriori* estimates is [36], see also [17]. An interest of *a posteriori estimate* is that it is expressed in terms of the approximate solution, assumed to be available in a mesh adaptation loop. A second interest is that it does not require the use of higher-order (approximate) derivatives, in contrast to truncation analyses. These estimates show accurately where the mesh should be refined. A method for deducing a better anisotropic mesh from an *a posteriori* estimate is proposed in [18], while a theory for H^p norms in [3] and a joint analysis of H^p and L^p norms of the error are presented in [2]. These methods cannot focus on an arbitrary user-specified error norm but relies on a particular one, specified by the variational formulation of the PDE. A more popular option is to choose, as accuracy target, a particular scalar output depending on the PDE solution. Any scalar output can be considered, except that difficulties can arise for the so-called non-admissible ones, according to [6]. An *a posteriori* estimate also allows for building *correctors* for goal-oriented analyses [19, 30]. In [35], the goal-oriented approach is cleverly combined with the correction strategy of [30] and with the Hessian-based metric approach, still minimizing the interpolation error of a user-prescribed sensor.

A priori estimates generally rely on Taylor series, either through divided differences or through polynomial approximation of functions. Then, approximations of higher-order derivatives of solution need to be built from the approximate solution. This is a delicate job since there are not many proofs ensuring that a good approximation of a higher-order derivative of the exact solution can be recovered from the low-order approximate solution. However, since the development of the first recovery methods (see for example [38]), many numerical experiments tend to show that the method is useful and rather reliable.

A remarkable feature of the goal-oriented metric-based adaptation of [23, 9] is the complete and coherent mathematical formulation of the mesh adaptation problem. Indeed, it takes the form of the optimization of a well-defined functional, namely the error for a prescribed scalar output, to be minimized with respect to a parameter, the metric, belonging to a well-identified and compact set. This strategy is applied to the discrete case in [37]. In [23, 9], in order to analytically solve the optimum, an *a priori* analysis is developed. It restricts to the main asymptotic term of the local error in order to exhibit more easily the dependance with respect to metric.

Goal-oriented methods have strongly impacted the applications but, due to its formulation, a goal-oriented method has two inherent limitations. First, it does not naturally extend to several scalar outputs. This “multi-target” issue is well-known and a proposition for addressing it is made in [21]. Second, because they are specialized to a given scalar output, the features of the solution field which are not influencing this output might be neglected by the automatic mesh improvement. A goal-oriented method provides the convergence of the approximate prescribed scalar output to its continuous analog. But generally convergence does not hold for the whole flow field itself. To clarify

this point, let us consider the mesh adaptative computation of a sonic boom footprint at the ground. The functional depends only on the pressure at the ground level. Now, many details of the flow on upper part of the aircraft do not influence the pressure at the ground level. This vanishing influence is taken into account by the adjoint state which also vanishes on these upper regions. Then, in these regions, the adapted mesh is not refined and the approximation of the flow field does not converge. See an illustration in [26]. As already noted, another limitation of a goal-oriented method is the scalar character of the error to reduce. It leads to use integrals of solution fields as in $(u - u_h, g)$, u being the exact solution, u_h its approximation and g a field prescribed by user. Now, these integrals are generally not sufficiently sensitive to oscillating deviations between u and u_h .

In the new norm-oriented formulation proposed in this paper, the user can prescribe a norm or a semi-norm $|u - u_h|$ of the error in order to minimize it with respect to the mesh. As a typical example of semi-norm, this can be the sum of square deviations on particular outputs. Let us take an example in aerodynamics. The semi-norm $|u - u_h| \equiv |C_l(u) - C_l(u_h)|^2 + |C_d(u) - C_d(u_h)|^2 + |C_m(u) - C_m(u_h)|^2$ will account for minimizing the errors on lift, drag and moment measured from flow solution u_h with respect to mesh. The proposed method will ultimately address this kind of semi-norm, assuming that, as for the goal-oriented method, the possible issue of a non-admissible norm according to [6] is solved. As for the goal-oriented method, the norm-oriented method takes into account the PDE features and, in the case where a *norm* is prescribed, it produces an approximate solution field which does converge to the exact one in this norm.

Although the proposed method is a rather general method extending to more complex CFD models, see for example [27], we consider in this paper a 2D Poisson problem discretized by the usual linear finite-element method. This choice is motivated first by the rather complete set of theoretical works available for the finite-element approximation of a Poisson problem. This amount of theoretical background reduces as much as possible (although far from completely) the heuristics to introduce in building the mesh adaptation analysis. In contrast, convergence results (in which functional spaces?) are not available for Euler equations, for example. A second motivation is the easy availability of exact solutions defined in a simple way. This allows to build a kind of benchmark allowing to compare mesh adaptation methods. Let us mention also that the Poisson problem (with variable coefficient) is a central equation in two-fluid models. The proposed benchmark will be inspired by a few particular two-fluid configurations.

Although the FEM is designed for minimizing the H^1 norm (with first-order accuracy), the user may wish to enjoy a convergence with a different norm. The proposed method, demonstrated for L^2 norm (providing second-order accuracy), is defined in order to many types of norms (or semi-norms). The method relies on the use of a corrector field and on an *a priori* error estimate from which is extracted the asymptotically largest terms of the local error. After we have the best mesh for the prescribed norm, are we in the best of all possible worlds? Not really. We need that this norm be less than a prescribed level. The central difficulty, in our opinion, is the accurate evaluation of this error norm. Most estimates are conditioned by mesh convergence. Therefore, we compare our estimates with mesh convergence analysis and try to propose a least uncertain error evaluation. In our test cases, this evaluation is compared with the information produced by the analytic exact solution.

After a formulation of the problem, the derivation of two correctors is proposed in Sec.2. Next three sections are devoted to the three identified mesh adaptation formulations: Hessian-based, minimizing an interpolation error in Sec.3, goal-oriented formulation in Sec.4 and our proposal for a norm-oriented in Sec.5. Sec.6 is devoted to a numerical comparison between the two field-convergent formulations, *viz.* Hessian-based and norm-oriented and the paper is completed by a discussion of methods and numerical examples.

2. Two correctors for the Poisson problem

On the way for minimizing the error norm $\|u - u_h\|_{L^2}$, where u is the exact PDE solution and u_h its approximation, we have to replace $u - u_h$ by a so-called corrector which would be rather easy to evaluate.

2.1. Notations

Let $V = H_0^1(\Omega)$, Ω being a sufficiently smooth computational domain of \mathbb{R}^2 . The continuous PDE system is written in short:

$$u \in V \mid Au = f \quad \text{or} \quad u \in V \mid \forall \varphi \in V, \quad a(u, \varphi) = (f, \varphi) \quad (1)$$

where

$$A = -\text{div}\left(\frac{1}{\rho}\nabla\right) \quad ; \quad a(u, \varphi) = \int_{\Omega} \frac{1}{\rho} \nabla u \cdot \nabla \varphi \, dx dy$$

and where $\frac{1}{\rho}$ is a positive, possibly discontinuous, scalar field in $L^\infty(\Omega)$. Further, we assume that the bilinear form a is coercive in space V , *i.e.* there exists a positive α such that:

$$a(v, v) \geq \alpha |v|_V^2.$$

This model exemplifies the pressure equation in some multi-phase incompressible flow formulation for which mesh adaptation is useful, see e.g. [20]. Let $\Omega_h = \Omega$ for simplicity, τ_h a triangulation of Ω_h and V_h be the usual P_1 -continuous finite-element approximation space related to τ_h :

$$V_h = \{\varphi_h \in \mathcal{C}^0(\bar{\Omega}) \cap V, \varphi_h|_T \text{ is affine } \forall T \in \tau_h\}.$$

The finite-element discretisation of (1) is written in variational and operational form:

$$u_h \in V_h \quad \text{and} \quad \forall \varphi_h \in V_h, \quad a(u_h, \varphi_h) = (f_h, \varphi_h) \quad (2)$$

in such a way that u_h is a linear function of f_h which we denote:

$$u_h = A_h^{-1} f_h.$$

We denote by Π_h the usual interpolation operator:

$$\forall v \in \mathcal{C}^0(\bar{\Omega}) \cap V, \quad \Pi_h v \in V_h \quad \text{and,} \quad \forall \mathbf{x}_i \text{ vertex of } \Omega_h, \quad \Pi_h v(\mathbf{x}_i) = v(\mathbf{x}_i).$$

Scalar correctors, *i.e.* correctors for scalar outputs $j(u_h)$ depending on the solution, e.g. $j(u_h) = (g, u_h)$ with g prescribed, have been defined by Giles and Pierce, [19]. Our interest concerns the correction of the unknown field itself. Two options are now proposed.

2.2. A priori corrector for the PDE solution

A first rather simple *a priori* corrector can be derived from an analysis of the error RHS. We observe that:

$$a(u - u_h, \varphi_h) = (f - f_h, \varphi_h) \quad \forall \varphi_h \in V_h.$$

Assuming that the solution u is continuous, we get:

$$a(\Pi_h u - u_h, \varphi_h) = a(\Pi_h u - u, \varphi_h) + (f - f_h, \varphi_h) \quad \forall \varphi_h \in V_h. \quad (3)$$

We call $\Pi_h u - u_h$ the *implicit error*. By implicit, we mean that it can be obtained through solving a discrete system. It differs from the approximation error by an interpolation error:

$$u - u_h = \Pi_h u - u_h + u - \Pi_h u. \quad (4)$$

In order to find an approximate of the implicit error, we need to evaluate the RHS of (3) for any test function φ_h . The second term of RHS of (3) is easy to evaluate (we know f and f_h). The first term of RHS of (3) can be transformed as follows (assuming that $\frac{1}{\rho}$ is constant on each element):

$$\begin{aligned} a(\Pi_h u - u, \varphi_h) &= \sum_T \int_T \frac{1}{\rho} \nabla \varphi_h \cdot \nabla (\Pi_h u - u) \, dx dy \\ &= \sum_T \int_{\partial T} (\Pi_h u - u) \frac{1}{\rho} \nabla \varphi_h \cdot \mathbf{n} \, d\sigma. \end{aligned}$$

Then, we get:

$$a(\Pi_h u - u_h, \varphi_h) = \sum_{\partial T_{ij}} \frac{1}{\rho} \nabla (\varphi_h|_{T_i} - \varphi_h|_{T_j}) \cdot \mathbf{n}_{ij} \int_{\partial T_{ij}} (\Pi_h u - u) \, d\sigma + (f - f_h, \varphi_h) \quad (5)$$

where the sum is taken for all edges ∂T_{ij} separating triangles T_i and T_j of the triangulation. The unit vector \mathbf{n}_{ij} normal to ∂T_{ij} is pointing outward T_i .

Now, we do not know u but u_h . In order to evaluate the interpolation error, we first approximate the Hessian of u by an approximation $H_h(u_h)$ in V_h of the Hessian of u_h . This is done with a Zienkiewicz-Zhu-type ([38]) recovery method defined in the seventh chapter of [29]. Then, the evaluation of $\Pi_h u - u$ is built on the edge e_{ij} as a quadratic function vanishing at both extremities of e_{ij} and of second-derivative in direction e_{ij} defined from $H_h(u_h)$. We replace $\Pi_h u - u$ by $\pi_h u_h - u_h$ where $\pi_h u_h - u_h$ is defined on edge ij as follows:

$$\forall \mathbf{x} \in e_{ij}, \quad (\pi_h u_h - u_h)(\mathbf{x}) = \frac{1}{2} (H_h(u_h)(\mathbf{x}_i) + H_h(u_h)(\mathbf{x}_j)) (\mathbf{x} - \mathbf{x}_i) (\mathbf{x} - \mathbf{x}_j)$$

which allows a mid-edge integration on each triangle of Ω_h . This replacement is justified in [33]. We shall see, in the sequel, that it is useful to apply a similar estimate for the $f - f_h$ term, $f - f_h \approx -(\pi_h f_h - f_h)$. In order to approximate the implicit error $\Pi_h u - u_h$, solution of (5), we define our *a priori implicit corrector* by:

$$\begin{aligned} u'_{prio} &\in V_h, \quad \text{and} \quad \forall \varphi_h \in V_h, \\ a(\bar{u}'_{prio}, \varphi_h) &= \sum_{\partial T_{ij}} \left(\frac{1}{\rho} \nabla \varphi_h|_{T_i} - \nabla \varphi_h|_{T_j} \right) \cdot \mathbf{n}_{ij} \int_{\partial T_{ij}} (\pi_h u_h - u_h) \, d\sigma - (\varphi_h, \pi_h f_h - f_h). \end{aligned} \quad (6)$$

This corrector is an approximation of $\Pi_h u - u_h$. Replacing again the unknown interpolation error by its evaluation on the discrete approximation, we define our *a priori corrector* by:

$$u'_{prio} = \bar{u}'_{prio} - (\pi_h u_h - u_h). \quad (7)$$

Assuming the approximations made between (5) and (6) are small, u'_{prio} is built in such a way that:

$$u'_{prio} \approx u - u_h.$$

This corrector is easy to compute but of *a priori* low accuracy.

2.3. Finer-grid defect correction corrector for the PDE solution

A second option for a corrector relies on using a fictitious finer grid. Let us assume that the approximation is in its asymptotic mesh convergence phase for the mesh Ω_h under study, of size h . Then, this will be also true for a strictly two-times finer embedding mesh $\Omega_{h/2}$ and, for our second-order accurate scheme applied to a sufficiently smooth problem, the *heuristics* of Richardson analysis writes:

$$u_h = A_h^{-1}f_h \quad , \quad u_{h/2} = A_{h/2}^{-1}f_{h/2} \quad \Rightarrow \quad u - u_{h/2} \approx \frac{1}{4}(u - u_h) \quad (8)$$

where u_h and $u_{h/2}$ are respectively the solutions on Ω_h and $\Omega_{h/2}$. We have *formally*:

$$\Pi_h u - \Pi_h u_{h/2} \approx \frac{1}{4}(\Pi_h u - u_h)$$

which implies:

$$\Pi_h u - u_h \approx \frac{4}{3}(\Pi_h u_{h/2} - u_h).$$

Now,

$$A_{h/2}(u_{h/2} - P_{h \rightarrow h/2}u_h) = f_{h/2} - A_{h/2}P_{h \rightarrow h/2}u_h.$$

where $P_{h \rightarrow h/2}$ linearly interpolates coarse values on fine mesh. This allows to evaluate $\Pi_h u - u_h$ but needs to solve a finer grid system. Let us introduce the residual transfer $R_{h/2 \rightarrow h}$ which accumulates on coarser grid vertices the values at finer vertices in neighboring coarse elements multiplied with barycentric weights. In order to reduce the computational cost to solving a coarser grid system, we approximate $\Pi_h A_{h/2}^{-1}$ by $A_h^{-1}R_{h/2 \rightarrow h}$:

$$\Pi_h(u_{h/2} - P_{h \rightarrow h/2}u_h) \approx A_h^{-1}R_{h/2 \rightarrow h}(f_{h/2} - A_{h/2}P_{h \rightarrow h/2}u_h).$$

This motivates the definition of a finer-grid Defect-Correction (DC) corrector as follows:

$$A_h \bar{u}'_{DC} = \frac{4}{3}R_{h/2 \rightarrow h}(f_{h/2} - A_{h/2}P_{h \rightarrow h/2}u_h) \quad (9)$$

In the case of local singularities, statement (8) is not true for uniform meshes but we have some hints that it holds almost everywhere for a sequence of adapted meshes, according to [28]. The DC corrector \bar{u}'_{DC} approximates $\Pi_h u - u_h$ instead of $u - u_h$ and can be corrected as the previous one:

$$u'_{DC} = \bar{u}'_{DC} - (\pi_h u_h - u_h). \quad (10)$$

3. Interpolation error optimization

Main notions and notations are stated by recalling the basics of the Hessian-based approach.

3.1. Mesh parametrization

We shall work inside the framework proposed in [24, 25]. The main idea of this framework is to model discrete meshes by continuous Riemannian metric fields. It allows us to define the adaptation problem as a differentiable optimization problem, *i.e.*, to apply, on the class continuous metrics, a calculus of variations which cannot be applied on the class of discrete meshes. This framework lies in the class of metric-based methods. A continuous metric \mathcal{M} of the computational domain Ω is a

Riemannian metric field [11] $\mathcal{M} = (\mathcal{M}(\mathbf{x}))_{\mathbf{x} \in \Omega}$ where $\mathcal{M}(\mathbf{x})$ is a symmetric 2×2 matrix. We define the *total number of vertices* of \mathcal{M} as:

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

Given a continuous metric \mathcal{M} , we shall say that a discrete mesh \mathcal{H} of the same domain Ω is a *unit mesh with respect to \mathcal{M}* if each triangle $K \in \mathcal{H}$, defined by its list of edges $(\mathbf{a}_i \mathbf{b}_i)_{i=1..3}$, verifies:

$$\forall i \in [1, 3], \quad \int_0^1 \sqrt{t \mathbf{a}_i \mathbf{b}_i \mathcal{M}(\mathbf{a}_i + t \mathbf{a}_i \mathbf{b}_i) \mathbf{a}_i \mathbf{b}_i} \, dt \in \left[\frac{1}{\sqrt{2}}, \sqrt{2} \right].$$

The rest of the paper will try to find the best metric \mathcal{M} from an error analysis which is asymptotic with respect to mesh size.

3.2. Interpolation-based optimal metric

Let u be any sufficiently smooth function defined on Ω . Let \mathcal{M} be a mesh/metric of Ω . In the rest of the paper, \mathcal{M} replaces h as the discretization index. We consider only meshes \mathcal{M} involving enough nodes for justifying the replacement of the complete error by its main asymptotic part. The P^1 interpolation error $|\Pi_{\mathcal{M}}u - u|$ can be approximated in terms of second derivatives of u and of the metric \mathcal{M} by the *continuous interpolation error* defined in [24]:

$$|\Pi_{\mathcal{M}}u - u| \approx |u - \pi_{\mathcal{M}}u|$$

with:

$$|u - \pi_{\mathcal{M}}u|(\mathbf{x}) = \frac{1}{8} \text{trace}(\mathcal{M}^{-\frac{1}{2}}(\mathbf{x}) |H_u(\mathbf{x})| \mathcal{M}^{-\frac{1}{2}}(\mathbf{x})) \quad (11)$$

where $|H_u|$ is deduced from H_u by taking the absolute values of its eigenvalues. Starting from:

$$\|u - \pi_{\mathcal{M}}u\|_{\mathbf{L}^p(\Omega_h)} = \left(\int_{\Omega} \left(\text{trace}(\mathcal{M}^{-\frac{1}{2}}(\mathbf{x}) |H_u(\mathbf{x})| \mathcal{M}^{-\frac{1}{2}}(\mathbf{x})) \right)^p \, d\mathbf{x} \right)^{\frac{1}{p}}, \quad (12)$$

we define as optimal metric the one which minimizes the right hand side under the constraint of a total number of vertices equal to a parameter N . After solving analytically this optimization problem (see e.g. [34, 22, 14, 23, 5], we get the unique optimal $(\mathcal{M}_{\mathbf{L}^p}(\mathbf{x}))_{\mathbf{x} \in \Omega}$ as:

$$\mathcal{M}_{\mathbf{L}^p} = \mathcal{K}_p(H_u) \quad \text{with} \quad \mathcal{K}_p(H_u) = D_{\mathbf{L}^p} (\det |H_u|)^{-\frac{1}{2p+2}} |H_u| \quad \text{and} \quad D_{\mathbf{L}^p} = N \left(\int_{\Omega} (\det |H_u| \, d\mathbf{x})^{\frac{p}{2p+2}} \right)^{-1}, \quad (13)$$

where $D_{\mathbf{L}^p}$ is a global normalization term set to obtain a continuous metric with complexity N and $(\det |H_u|)^{-\frac{1}{2p+2}}$ is a local normalization term accounting for the sensitivity of the \mathbf{L}^p norm. In the case of an adaptation loop for solving a Partial Differential Equation, a continuous function u is not available but an approximate solution $u_{\mathcal{M}}$ is. In that case, the continuous interpolation error (11) is replaced by:

$$|u_{\mathcal{M}} - \pi_{\mathcal{M}}u_{\mathcal{M}}|(\mathbf{x}) = \frac{1}{8} \text{trace}(\mathcal{M}^{-\frac{1}{2}}(\mathbf{x}) |H_{u_{\mathcal{M}}}(\mathbf{x})| \mathcal{M}^{-\frac{1}{2}}(\mathbf{x})) \quad (14)$$

where $H_{u_{\mathcal{M}}}$ is an approximate Hessian evaluated with a recovery method.

According to the continuous metric framework, statement (13) defines directly a continuous optimal metric. In practice, solving (13) is done by approximation, *i.e.* in a discrete context with a couple (mesh, solution) denoted $(\mathcal{H}_{\mathcal{M}}, u_{\mathcal{M}})$ and iteratively obtained through the following fixed point:

Step 1: compute the discrete state $u_{\mathcal{M}}$ on mesh $\mathcal{H}_{\mathcal{M}}$,

Step 2: compute sensor $s_{\mathcal{M}} = s(u_{\mathcal{M}})$ and optimal metric $\mathcal{M}_{inter}^{opt} = \mathcal{K}_p(H_{\mathcal{M}}(s_{\mathcal{M}}))$,

Step 3: put $\mathcal{M} = \mathcal{M}_{inter}^{opt}$, generate a new mesh $\mathcal{H}_{\mathcal{M}} = \mathcal{H}_{\mathcal{M}_{inter}^{opt}}$ and go to 1, until convergence.

In the above algorithm, the continuous Hessian of s is replaced by an approximate Hessian $H_{\mathcal{M}}(s_{\mathcal{M}})$, evaluated by the patch-recovery approximation defined in [29]. In our Hessian-based numerical examples, the L^2 case, $p = 2$, has been considered. The above notation \mathcal{K}_p will also be used in the next sections for $p = 1$.

4. Implicit a priori error estimate

In contrast to a corrector as defined in Section 2, an asymptotic upper bound of the approximation error should allow an easier error reduction by minimisation of its norm with respect to the metric. In our PDE discretisation notations, we henceforward replace in the discretization index h by the index \mathcal{M} which holds for any unit mesh of the metric \mathcal{M} . The implicit *a priori* error system (5) then writes:

$$\begin{aligned} \forall \varphi \in V_{\mathcal{M}}, \\ a(\Pi_{\mathcal{M}}u - u_{\mathcal{M}}, \varphi) = \sum_{\partial T_{ij}} \frac{1}{\rho} (\nabla\varphi|_{T_i} - \nabla\varphi|_{T_j}) \cdot \mathbf{n}_{ij} \int_{\partial T_{ij}} (\Pi_{\mathcal{M}}u - u) \, d\sigma - (\varphi, \Pi_{\mathcal{M}}f - f) \end{aligned} \quad (15)$$

where T_{ij} are the triangles of a unit mesh for \mathcal{M} and the proposed corrector is expressed with the discrete solution:

$$\begin{aligned} \bar{u}'_{prio} \in V_{\mathcal{M}} \quad , \quad a(\bar{u}'_{prio}, \varphi) = K(\mathcal{M}, \varphi, u_{\mathcal{M}}) \quad \forall \varphi \in V_{\mathcal{M}} \quad \text{with} \\ K(\mathcal{M}, \varphi, u_{\mathcal{M}}) = \sum_{\partial T_{ij}} \frac{1}{\rho} (\nabla\varphi|_{T_i} - \nabla\varphi|_{T_j}) \cdot \mathbf{n}_{ij} \int_{\partial T_{ij}} (\pi_{\mathcal{M}}u_{\mathcal{M}} - u_{\mathcal{M}}) \, d\sigma - (\varphi, \pi_{\mathcal{M}}f - f), \\ u'_{prio} = \bar{u}'_{prio} - (\pi_{\mathcal{M}}u_{\mathcal{M}} - u_{\mathcal{M}}). \end{aligned} \quad (16)$$

We now restart from (5). The following result is proven in [8]:

Lemma 4.1. *We assume that the metric anisotropy is bounded by a positive number. For any smooth couple of functions (u, φ) , where u is not necessarily a solution of (1), we have the following bound:*

$$\left| \int_{\Omega} \frac{1}{\rho} \nabla(u - \Pi_{\mathcal{M}}u) \nabla \Pi_{\mathcal{M}}\varphi \, d\mathbf{x} \right| \leq K \int_{\Omega} \frac{1}{\rho} \rho_S(H(\varphi)) |u - \Pi_{\mathcal{M}}u| \, d\mathbf{x} \quad (17)$$

where $A \preceq B$ holds for a majoration asymptotically valid, *i.e.* $A \leq B + o(A)$. Expression $\rho_S(H(\varphi))$ holds for the largest (in absolute value) eigenvalue of the Hessian $H(\varphi)$ of φ . \square

The next section shows how to use this estimate.

5. Equation-based adaptation

5.1. Scalar output “goal-oriented” analysis

The goal-oriented analysis relies on the minimization of the error $\delta j_{goal}(\mathcal{M})$ done in the evaluation of the scalar output $j = (g, u)$, error which we simplify as follows:

$$\delta j_{goal}(\mathcal{M}) = |(g, u - u_{\mathcal{M}})| = |(g, \Pi_{\mathcal{M}}u - u_{\mathcal{M}} + u - \Pi_{\mathcal{M}}u)|. \quad (18)$$

Let us define the discrete adjoint state $u_{g,\mathcal{M}}^*$:

$$\forall \psi_{\mathcal{M}} \in V_{\mathcal{M}}, \quad a(\psi_{\mathcal{M}}, u_{g,\mathcal{M}}^*) = (\psi_{\mathcal{M}}, g). \quad (19)$$

Then:

$$(g, \Pi_{\mathcal{M}}u - u_{\mathcal{M}} + u - \Pi_{\mathcal{M}}u) = a(\Pi_{\mathcal{M}}u - u_{\mathcal{M}}, u_{g,\mathcal{M}}^*) + (g, u - \Pi_{\mathcal{M}}u)$$

and, using (3),

$$(g, \Pi_{\mathcal{M}}u - u_{\mathcal{M}} + u - \Pi_{\mathcal{M}}u) = a(\Pi_{\mathcal{M}}u - u, u_{g,\mathcal{M}}^*) + (f - \Pi_{\mathcal{M}}f, u_{g,\mathcal{M}}^*) + (g, u - \Pi_{\mathcal{M}}u)$$

thus

$$\delta j_{goal}(\mathcal{M}) \approx |a(\Pi_{\mathcal{M}}u - u, u_{g,\mathcal{M}}^*) + (f - \Pi_{\mathcal{M}}f, u_{g,\mathcal{M}}^*) + (g, u - \Pi_{\mathcal{M}}u)|$$

Recall that u is unknown. The second term $\Pi_{\mathcal{M}}u - u$, similar to the main term of the Hessian-based adaptation in Section 3.2, can be explicitly approached in the same way, *i.e.* introducing the continuous interpolation error (14):

$$\delta j_{goal}(\mathcal{M}) \leq |a(\Pi_{\mathcal{M}}u - u, u_{g,\mathcal{M}}^*)| + |(f - \Pi_{\mathcal{M}}f, u_{g,\mathcal{M}}^*)| + |g| |\pi_{\mathcal{M}}u_{\mathcal{M}} - u_{\mathcal{M}}|$$

For the second $\Pi_{\mathcal{M}}u - u$ term, we first apply Lemma 4.1 and then also introduce the continuous interpolation error. We get

$$\delta j_{goal}(\mathcal{M}) \leq \int_{\Omega} \left(\left[\frac{1}{\rho} \rho_S(H(u_{g,\mathcal{M}}^*)) + |g| \right] |\pi_{\mathcal{M}}u_{\mathcal{M}} - u_{\mathcal{M}}| + |u_{g,\mathcal{M}}^*| |\pi_{\mathcal{M}}f - f| \right) d\mathbf{x}.$$

It is then reasonable to try to minimize the RHS of this inequality instead of the LHS. But this still involves some difficulty due to the dependency of adjoint state $u_{g,\mathcal{M}}^*$ with respect to \mathcal{M} . We shall further simplify our functional by freezing, during a part of the algorithm, the adjoint state. The idea is that, when we change the parameter \mathcal{M} , the discrete adjoint $u_{g,\mathcal{M}}^*$ is close to its (non-zero) continuous limit and is thus not much affected, in contrast to the interpolation errors $|\pi_{\mathcal{M}}u_{\mathcal{M}} - u_{\mathcal{M}}|$ and $|\pi_{\mathcal{M}}f - f|$. We then consider, for a given \mathcal{M}_0 , the following optimum problem:

$$\min_{\mathcal{M}} \int_{\Omega} \left(\left[\frac{1}{\rho} \rho_S(H(u_{g,\mathcal{M}_0}^*)) + |g| \right] |\pi_{\mathcal{M}}u_{\mathcal{M}} - u_{\mathcal{M}}| + |u_{g,\mathcal{M}_0}^*| |\pi_{\mathcal{M}}f - f| \right) d\mathbf{x}.$$

This will produce an optimum:

$$\mathcal{M}_{opt,\mathcal{M}_0} = \arg \min_{\mathcal{M}} |tr(\mathcal{M}^{-1/2} \left(\left[\frac{1}{\rho} \rho_S(H(u_{g,\mathcal{M}_0}^*)) + |g| \right] |H_u| + |u_{g,\mathcal{M}_0}^*| |H_f| \right) \mathcal{M}^{-1/2})|.$$

Observing that, in the integrand,

$$H_{goal,0} = \left[\frac{1}{\rho} \rho_S(H(u_{g,\mathcal{M}_0}^*)) + |g| \right] |H_u| + |u_{g,\mathcal{M}_0}^*| |H_f|$$

is a positive symmetric matrix, we can apply the above calculus of variation and get:

$$\mathcal{M}_{opt,\mathcal{M}_0} = \mathcal{K}_1 \left(\left[\frac{1}{\rho} \rho_S(H(u_{g,\mathcal{M}_0}^*)) + |g| \right] |H_u| + |u_{g,\mathcal{M}_0}^*| |H_f| \right)$$

where \mathcal{K}_1 is defined in (13). This solution can then be introduced in a fixed-point loop and will produce the solution of:

$$\mathcal{M}_{opt,goal} = \mathcal{K}_1 \left(\left[\frac{1}{\rho} \rho_S(H(u_{g,\mathcal{M}_{opt,goal}}^*)) + |g| \right] |H_u| + |u_{g,\mathcal{M}_{opt,goal}}^*| |H_f| \right).$$

Let us precise how the discrete algorithm is organised:

Step 1: compute the discrete state $u_{\mathcal{M}}$ on mesh $\mathcal{H}_{\mathcal{M}}$,

Step 2: compute the discrete adjoint state $u_{\mathcal{M}}^*$,

Step 3: compute optimal metric $\mathcal{M}_{goal}^{opt}(u_{\mathcal{M}})$,

Step 4: put $\mathcal{M} = \mathcal{M}_{goal}^{opt}(u_{\mathcal{M}})$, generate a new mesh $\mathcal{H}_{\mathcal{M}} = \mathcal{H}_{\mathcal{M}_{goal}^{opt}(u_{\mathcal{M}})}$ and go to 1, until convergence.

The adaptation of this process to the Euler model of Gas Dynamics is studied in [26] for the steady case and in [10] for the unsteady case.

5.2. Norm-based functional

We are now interested by the minimization of the following expression with respect to the mesh \mathcal{M} :

$$\delta j(\mathcal{M}) = \|u - u_{\mathcal{M}}\|_{L^2(\Omega)}^2. \quad (20)$$

Introducing $g_{\mathcal{M}} = u - u_{\mathcal{M}}$, we get a formulation similar to the goal-oriented formulation:

$$\delta j(\mathcal{M}) = (g_{\mathcal{M}}, u - u_{\mathcal{M}}). \quad (21)$$

Let us define the discrete adjoint state $u_{\mathcal{M}}^*$:

$$\forall \psi_{\mathcal{M}} \in V_{\mathcal{M}}, \quad a(\psi_{\mathcal{M}}, u_{\mathcal{M}}^*) = (\psi_{\mathcal{M}}, g_{\mathcal{M}}). \quad (22)$$

Then, similarly to Section 5.1, we have to solve the following optimum problem.

$$\min_{\mathcal{M}} \int_{\Omega} \left(\left[\frac{1}{\rho} \rho_S(H(u_{\mathcal{M}}^*)) + |g_{\mathcal{M}}| \right] |\pi_{\mathcal{M}} u_{\mathcal{M}} - u_{\mathcal{M}}| + |u_{\mathcal{M}}^*| |\pi_{\mathcal{M}} f - f| \right) dx.$$

Exactly as for Section 5.1, we freeze the dependency of the adjoint state.

$$\min_{\mathcal{M}} \int_{\Omega} \left(\left[\frac{1}{\rho} \rho_S(H(u_{\mathcal{M}_0}^*)) + |g_{\mathcal{M}}| \right] |\pi_{\mathcal{M}} u_{\mathcal{M}} - u_{\mathcal{M}}| + |u_{\mathcal{M}_0}^*| |\pi_{\mathcal{M}} f - f| \right) dx.$$

$$\mathcal{M}_{opt,\mathcal{M}_0} = \mathcal{K}_1 \left(\left[\frac{1}{\rho} \rho_S(H(u_{\mathcal{M}_0}^*)) + |g_{\mathcal{M}}| \right] |H_u| + |u_{\mathcal{M}_0}^*| |H_f| \right).$$

In practice, the $g_{\mathcal{M}}$ RHS in (21-22) is replaced by the corrector $u'_{prio,\mathcal{M}}$. In order to get the final norm-oriented optimum $\mathcal{M}_{opt,norm}$, we apply:

Step 1: compute the discrete state $u_{\mathcal{M}^{(\alpha)}}$ on mesh $\mathcal{H}_{\mathcal{M}^{(\alpha)}}$,

Step 2: solve the linearised corrector system:

$$a(\bar{u}'_{prio,\mathcal{M}^{(\alpha)}}, \varphi) = \sum_{\partial T_{ij}} (\nabla \varphi|_{T_i} - \nabla \varphi|_{T_j}) \cdot \mathbf{n}_{ij} \int_{\partial T_{ij}} (\pi_{\mathcal{M}^{(\alpha)}} u_{\mathcal{M}^{(\alpha)}} - u_{\mathcal{M}^{(\alpha)}}) d\sigma - (\varphi, \pi_{\mathcal{M}^{(\alpha)}} f_{\mathcal{M}^{(\alpha)}} - f_{\mathcal{M}^{(\alpha)}}). \quad (23)$$

where $\pi_{\mathcal{M}^{(\alpha)}}u_{\mathcal{M}^{(\alpha)}} - u_{\mathcal{M}^{(\alpha)}}$ is expressed in terms of metric and Hessian, as in (6).

Put $u'_{prio,\mathcal{M}^{(\alpha)}} = \bar{u}'_{prio,\mathcal{M}^{(\alpha)}} - (\pi_{\mathcal{M}^{(\alpha)}}u_{\mathcal{M}^{(\alpha)}} - u_{\mathcal{M}^{(\alpha)}})$.

Step 3: then, solve the adjoint system:

$$a(\psi, u^*_{prio,\mathcal{M}^{(\alpha)}}) = (u'_{prio,\mathcal{M}^{(\alpha)}}, \psi) \quad (24)$$

Step 4: put:

$$\mathcal{M}^{(\alpha+1)} = \mathcal{K}_1(|u'_{prio}| + \frac{1}{\rho}\rho_S H(u^*_{prio})) |H_{u_{\mathcal{M}^{(\alpha)}}}| + |u^*_{prio}| |H_f|) \quad (25)$$

Step 5: generate a new mesh $\mathcal{H}_{\mathcal{M}^{(\alpha)}}$ and go to 1, until convergence to a fixed point $\mathcal{M}_{opt,norm} = \mathcal{M}^{(\infty)}$.

The new algorithm involves the solution of two extra linear systems with respect to a basic Hessian-based algorithm and one extra linear system with respect to the goal-oriented algorithm.

Remark: In contrast to the adjoint of the goal-oriented algorithm, these auxiliary variables are not consistent with a continuous adjoint (and do not converge towards it when mesh is refined). They are correctors and converge to zero. Then it seems that we cannot get a good initial condition to an FMG phase (from a coarse mesh to a finer one) on these variables. For doing it anyway, we rely again on the Richardson heuristics. We assume that our data and solution are sufficiently regular for applying the Aubin-Nitsche L^2 analysis. This implies the second-order L^2 convergence of approximate solution to exact one, *i.e.* $\|u_{\mathcal{M}} - u\|_{L^2} = O(h^2)$. This tends to indicate that our corrector norm would be also $O(h^2)$, *i.e.*, *formally*:

$$\|u'_{prio,\mathcal{M}}\|_{L^2} = O(h^2).$$

We write this in terms of the number N of vertices for metric \mathcal{M} as follows (again *formally*):

$$\|u'_{prio,\mathcal{M}}\|_{L^2} = O(N^{-1}).$$

Further, following a (*formal*) Richardson expansion:

$$u'_{prio,\mathcal{M}} \approx N^{-1}u'_{prio} \quad (26)$$

where u'_{prio} is the limit (assumed to exist) of $Nu'_{prio,\mathcal{M}}$ when mesh is refined. Similarly, we can put:

$$u^*_{prio,\mathcal{M}} \approx N^{-1}u^*_{prio}. \quad (27)$$

In particular, when passing from a mesh \mathcal{M}_1 with N_1 vertices, \mathcal{M}_2 with N_2 vertices, approximations of u'_{prio,\mathcal{M}_2} and u^*_{prio,\mathcal{M}_2} can be defined as

$$u'_{prio,\mathcal{M}_2} \approx \frac{N_1}{N_2}u'_{prio,\mathcal{M}_1} \quad ; \quad u^*_{prio,\mathcal{M}_2} \approx \frac{N_1}{N_2}u^*_{prio,\mathcal{M}_1} \quad (28)$$

and used as initial conditions inside the FMG method. \square

6. Numerical examples

We restrict our study to a benchmark of two-dimensional Poisson problems. We conjecture that the two following mesh adaptation methods produce L^2 convergent solutions to continuous. The first method, the Hessian-based method (with $p = 2$), is just heuristically relying on usual finite-element estimates. The second method, our novel norm-oriented method, is directly built on the minimisation of the L^2 error norm. We do not consider goal-oriented applications for which examples of computations can be found in [26] and [10]. As already remarked, the convergence of goal-oriented solutions to continuous is definitively questionable.

6.1. Numerical features

In [12], a mesh-adaptative full-multigrid (FMG) algorithm relying on the Hessian-based adaptation criterion is designed. We first describe in short this algorithm for the Hessian-based option. A sequence of numbers N_k of vertices is specified from a coarse mesh to finer one $N_0 = N, N_1 = 4N, N_2 = 16N, N_3 = 64N, \dots$. For each mesh size N_k , a sequence of adapted meshes of size N_k is built by iterating the following loop:

- (a) computing a solution,
- (b) computing the optimal metric,
- (c) building the adapted mesh.

In (a), a multi-grid V-cycle is applied to a sufficient convergence. In (b), approximations of the Hessians are performed as in [26]. When changing of mesh, an interpolation is applied in order to enjoy a good initial condition. A prescribed number of 4 adaptation iterations is applied at each mesh fineness N_k .

The extension of the above loop to norm-oriented adaptation consists of replacing the single Hessian evaluation by:

- the computation of the corrector, using MG and, as initial solution, the previous evaluation interpolated to current mesh and corrected according to (28),
- the computation of the adjoint, using MG and, as initial solution, the previous evaluation interpolated to current mesh and corrected according to (28),
- the evaluation of (25).

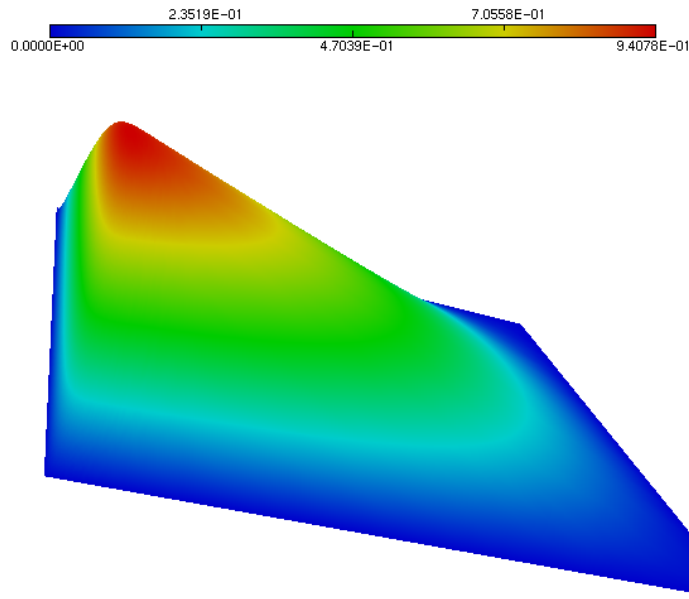


Figure 1: Fully 2D Boundary layer test case : sketch of the solution.

Let us discuss computer efficiency. In the demonstrator of [12], a particular feature is the stopping criterion of FMG which applies to the convergence of the solution of the unique solved system, *i.e.*

the system under study, $u = A^{-1}f$. It is then possible to enjoy a better and better initial condition and control the iterative and approximation errors convergence. Consequently, it was possible, in [12], to show that mesh adaptation carries large improvement not only in terms of accuracy for a given number of vertices but also in terms of accuracy for a given computational time.

In contrast, the method proposed in this paper involves three systems to solve: (1) the system under study, $u = A^{-1}f$, (2) the corrector system, (3) the adjoint system. We shall give an idea of the performances for one test case.

6.2. 2D Boundary layer

This test case is taken from [18]. We solve the Poisson problem $-\Delta u = f$ in $]0, 1[\times]0, 1[$ with Dirichlet boundary conditions and a right-hand side f chosen for having:

$$u(x, y) = [1 - e^{-\alpha x} - (1 - e^{-\alpha})x]4y(1 - y).$$

The coefficient α is chosen equal to 100. The graph of the solution is depicted in Figure 1. Before applying our mesh adaptative algorithm, it is interesting to evaluate the accuracy of our correctors. We choose a 161×161 uniform mesh and show, in Figure 2 and Figure 3, the cut of $u - u_h$ compared with the cut of u' . We observe that the *a priori* corrector does its job in a correct but inaccurate manner while the DC one is rather accurate. We have also observed that the DC corrector does not consume notably more CPU than the *a priori* one. Therefore, we keep this option for the rest of the test case. In Figure 4, we show a set of FMG calculations for the considered test case. The

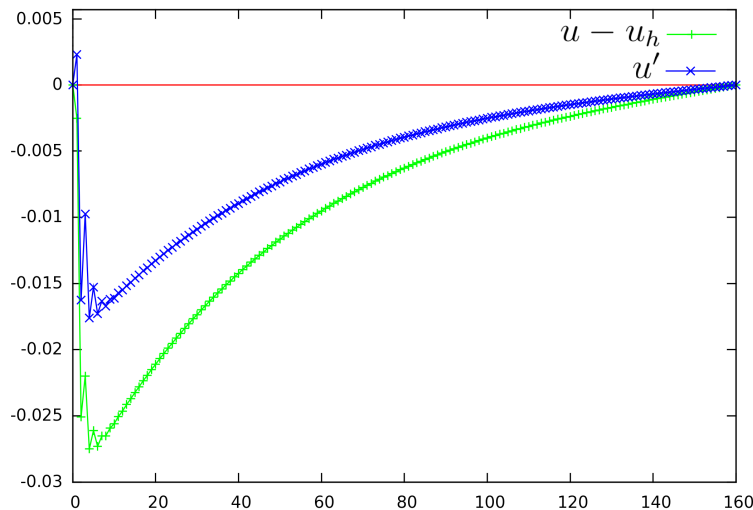


Figure 2: Fully 2D Boundary layer test case : comparison of error cuts for $y = 0.5$: plus signs (+) depict the approximation error $u - u_h$ and crosses (\times) depict the *a priori* corrector u'_{prio} . The corrector is able to correct about 60% of the approximation error.

numbers of vertices of the successive meshes are supported by the horizontal axis, from 120 vertices to 30,000 vertices. The vertical axis gives the L^2 -norm of the approximation error $|u - u_h|_{L^2}$ obtained on the mesh. Its variation with respect to number of vertices is compared in Figure 4 for the three following algorithms: (a) the uniform-mesh FMG, (b) the Hessian-based adaptative FMG and (c) the norm-oriented adaptative FMG. We observe that both adaptation methods carry an important improvement with respect to uniform-grid FMG (25921 vertices on finest mesh). For essentially the same number of vertices (32318), the Hessian option gives an error divided by 47. The norm-oriented option appears as better with an error divided by 208 with 29485 vertices.

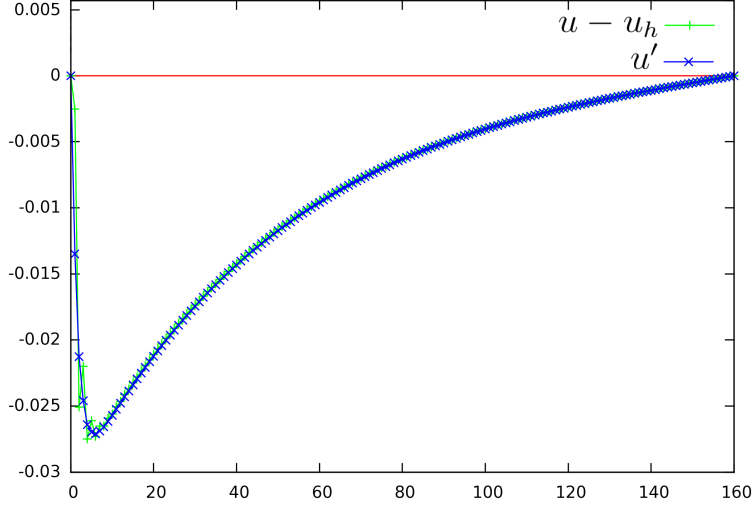


Figure 3: Fully 2D Boundary layer test case : comparison of error cuts for $y = 0.5$: plus signs (+) depict the approximation error $u - u_h$ and crosses (\times) depict the Defect-Correction corrector u'_{DC} . The corrector is able to correct about 95% of the approximation error.

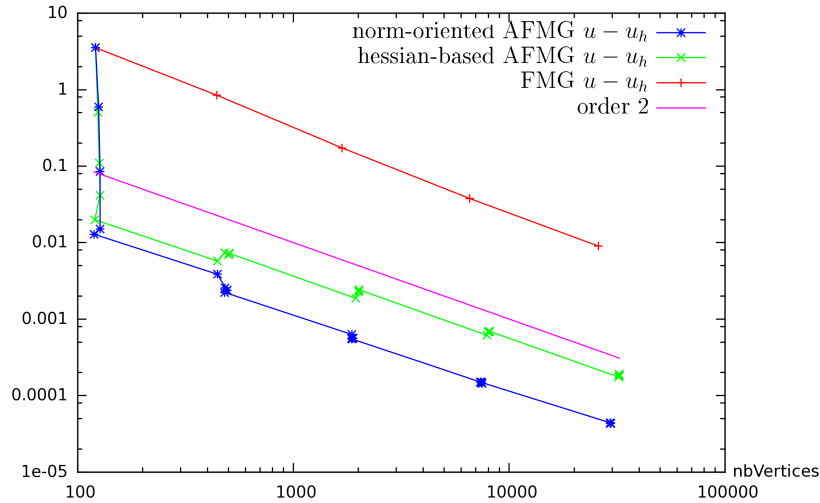


Figure 4: Fully 2D Boundary layer test case: convergence of the error norm $|u - u_h|_{L^2}$ as a function of number of vertices in the mesh for (+) non-adaptative FMG, (\times) Hessian-based adaptative FMG, (*) norm-oriented adaptative FMG.

Since the exact solution u is analytically available, we also propose an *a posteriori* measure of the correctors efficiency by comparing the convergence of our norm-oriented adaptation (equipped with either corrector) with the same method except that the corrector is replaced by $u - u_h$. Of course, that latter algorithm is not a mesh adaptation method since we assume that we already know the exact solution. In Figure 5, we observe that the error convergence for the three computations are very close to each other. This confirms the interest of the two proposed correctors.

Lastly, when using the proposed method in usual conditions, *i.e.* without knowing the analytic solution, it can be interesting to have an *evaluation of the final error* $u - u_h$ by using the DC corrector which appeared in experiments as more accurate. Figure 6 shows a comparison of the *evaluated error*

computed from the corrector with the *actual error*. We observe that this evaluation is useful, giving the good order of magnitude. For the finer mesh of 30,000 vertices, the evaluated error is $2.14 \cdot 10^{-5}$ while the actual error is $4.31 \cdot 10^{-5}$. However, for most meshes, the evaluation remains somewhat optimistic (smaller than the actual error).

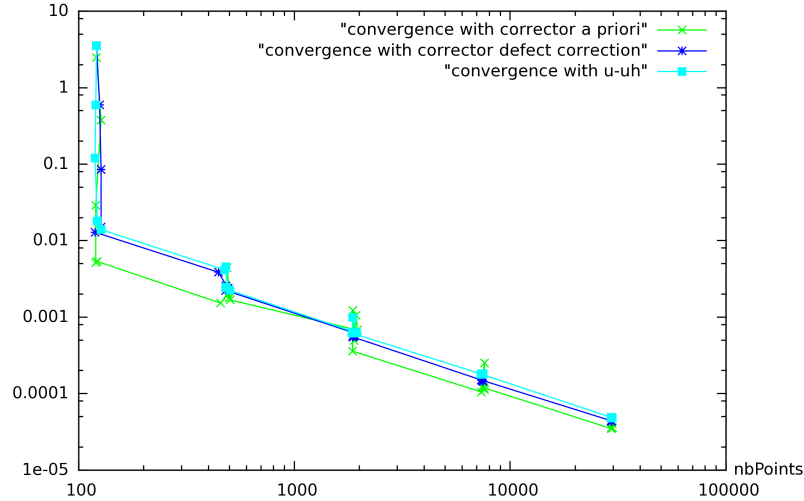


Figure 5: Fully 2D Boundary layer test case: convergence based on (×) the *a priori* corrector or on the (*) Defect-Correction one, compared with (□) a virtual adaptation controlled by $u - u_h$.

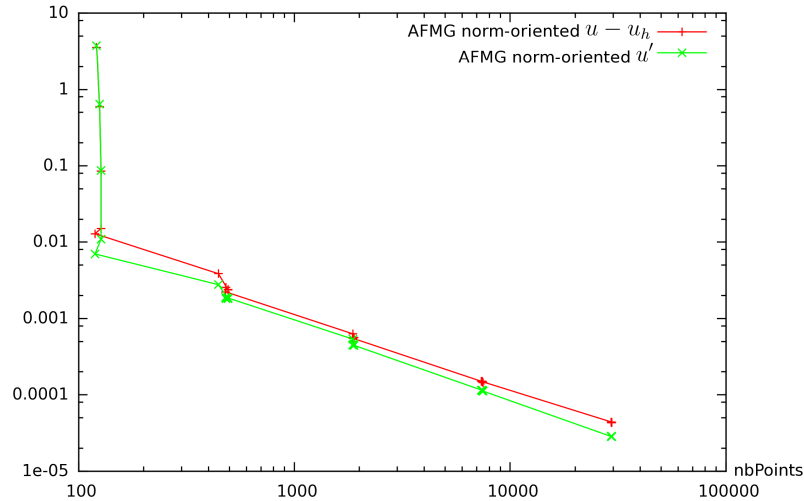


Figure 6: Fully 2D Boundary layer test case: comparison of the predicted error norm proposed by the method, curve with (□), with the exact error norm (×).

6.3. Bubble-like test case with thick interface

We are interested by a Poisson problem the solution of which is a function u equal to 1 on a disk and to 0 in the rest of the domain. This function is the prototype of the pressure in a multi-fluid flow involving capillary forces. The source term is a Dirac derivative. We smooth this

computation by defining a thickness ε for defining an annular region separating the two subdomains (outside the disk, inside the disk) and in which u is smoothly varying from 0 to 1: if (x, y) is located inside the annular region, $u(x, y)$ is given by the formula: $u(x, y) = \frac{1}{2} + \frac{1}{2} \sin(\frac{\pi\psi}{\varepsilon})$ with $\psi = 0.25 - \sqrt{(x_C - x)^2 + (y_C - y)^2}$. From this solution, a right-hand side f is computed. Given a mesh, vertex values of f_h are interpolated from the analytic f . As a result, for rather coarse meshes, the zone where f is not zero can be spuriously missed and f_h can be zero even in the neighborhood of the high values of f . We consider first a quite large thickness of $\varepsilon = 0.1$. An approximate solution u_h is shown in Figure 7. As for the previous test case, we first evaluate the accuracy of the corrector.

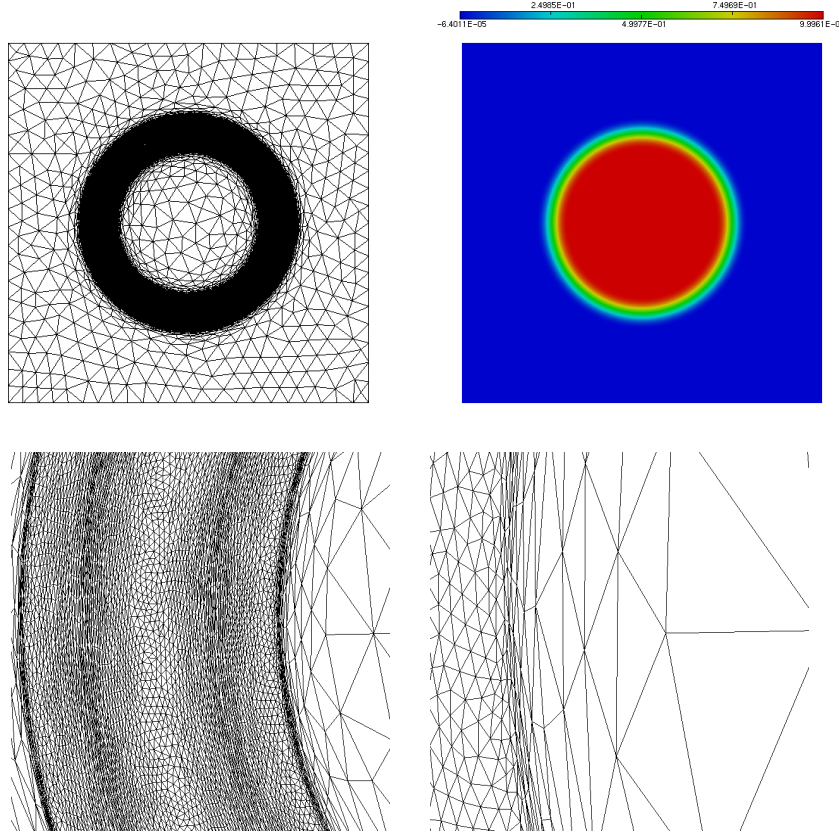


Figure 7: Circular layer test case: an adapted mesh and the corresponding numerical solution u_h . Palette from $-6.4 \cdot 10^{-5}$ to 0.9996.

We choose a uniform mesh 161×161 and show, in Figure 8 and Figure 9, the cut of $u - u_h$ compared with the cut after correction, that is $u - u_h - u'_h$. We observe that both *a priori* and Defect-Correction correctors do an accurate job.

The three methods are again compared in Figure 10: standard FMG, Hessian-based adaptive FMG and norm-oriented adaptive FMG. For the Hessian-based calculation, we observe a tendency for a slower convergence for finer meshes, finishing with an error which is worse than the uniform refinement. The proposed norm-oriented adaptive method behaves in a better way with a five times smaller error than for the uniform refinement.

6.4. Bubble-like test case with thin interface

In order to evaluate the robustness of the methods with respect to steeper gradients, we consider the same test case with a thinner transition: $\varepsilon = 0.02$.

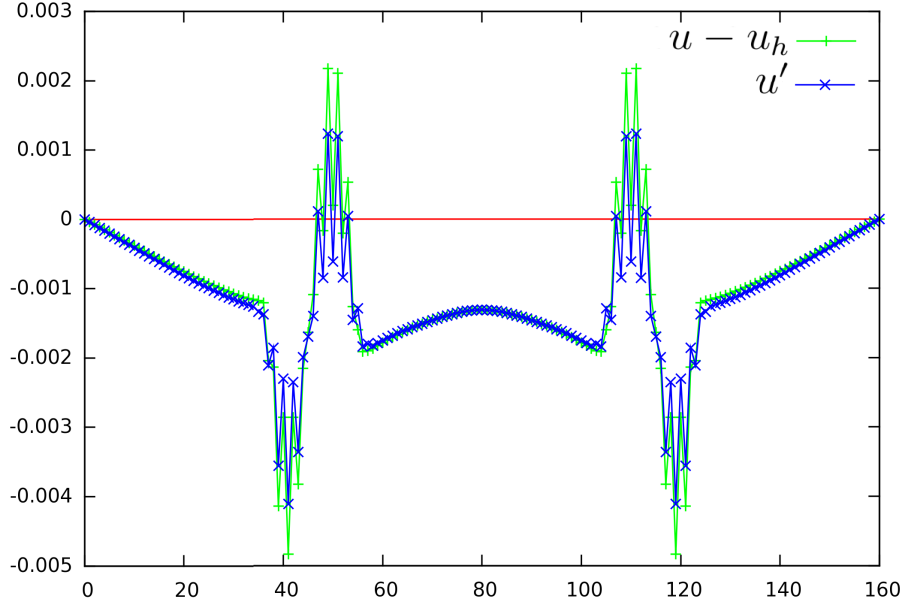


Figure 8: Thick bubble case: comparison of (+) error $u - u_h$ and (\times) *a priori* corrector.

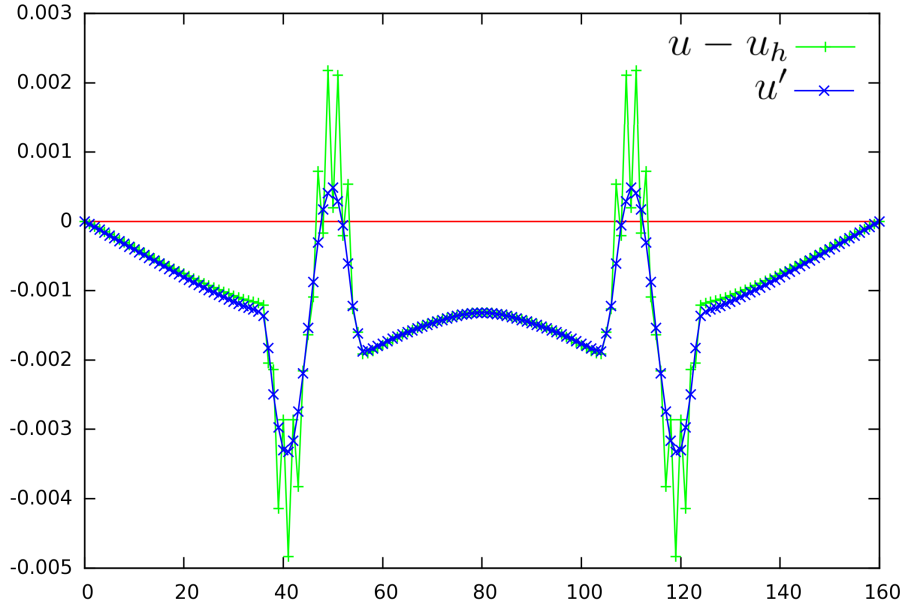


Figure 9: Thick bubble case: comparison of (+) error $u - u_h$ and (\times) Defect-Correction corrector.

The convergence of the three methods is shown in Figure 14. Due to the very thin support of the right-hand side f , the three methods start with a zero f_h . Then, either with adaptation or refinement, the error increases to several units. The convergence of the uniform FMG shows an acceptable slope but the error values remain relatively huge. By comparison with the thick-bubble convergence, we may infer that the slope of uniform FMG will be second-order with even higher number of vertices and a 0.1 % error may be not attained for meshes of less than 10 millions nodes. The Hessian-based final result is a little better but globally disappointing. The norm-oriented convergence starts chaotically

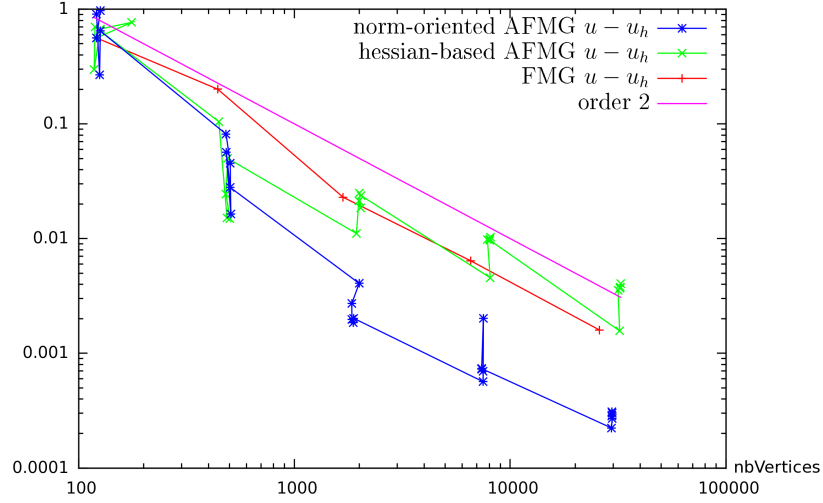


Figure 10: Thick bubble test case: convergence of the error norm $|u - u_h|_{L^2}$ as a function of number of vertices in the mesh for (+) non-adaptative FMG, (\times) Hessian-based adaptative FMG and ($*$) norm-oriented adaptative FMG.

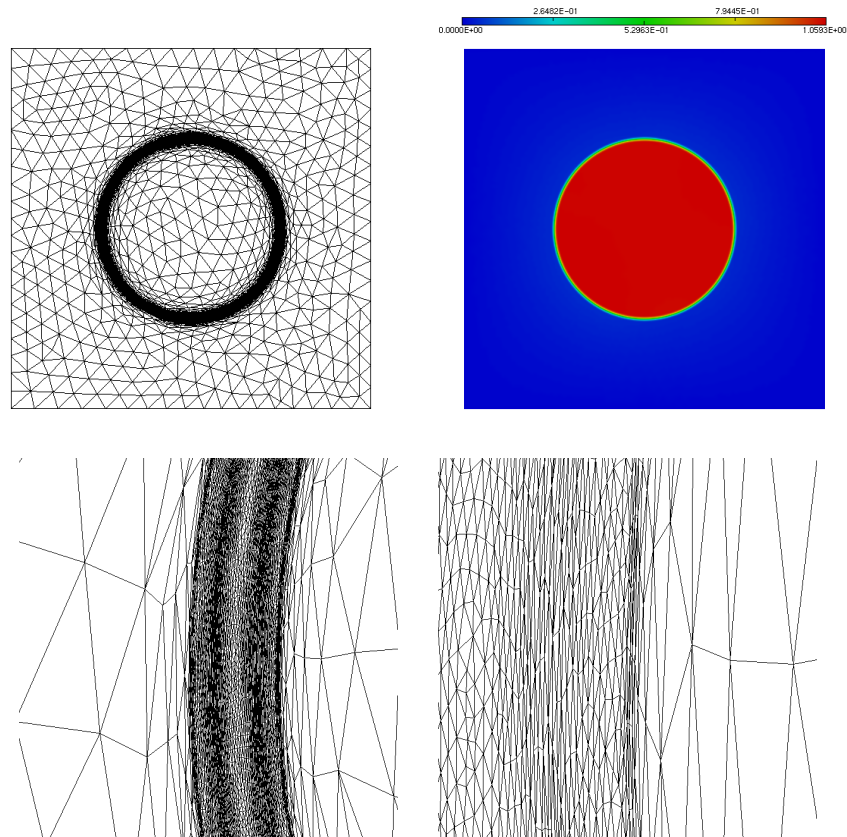


Figure 11: Thin bubble case: an adapted mesh and the corresponding numerical solution u_h . Palette from 0.000 to 1.0593.

before being monotone and second-order for meshes finer than 2000 vertices. The final L^2 errors produced by the three methods are 0.17589 with 25921 vertices for the uniform FMG, 0.03773 with 32127 vertices for the Hessian-based adaptation and 0.000585 for 29742 vertices for the norm-oriented

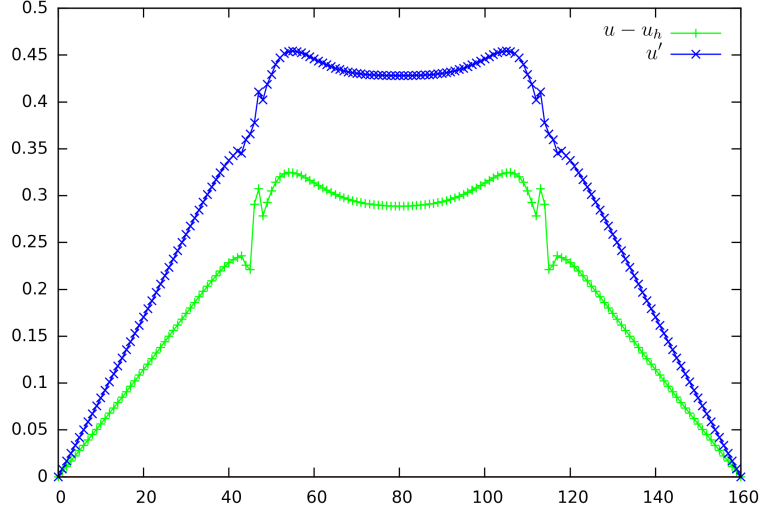


Figure 12: Thin bubble case: comparison of (+) error $u - u_h$ and (\times) *a priori* corrector.

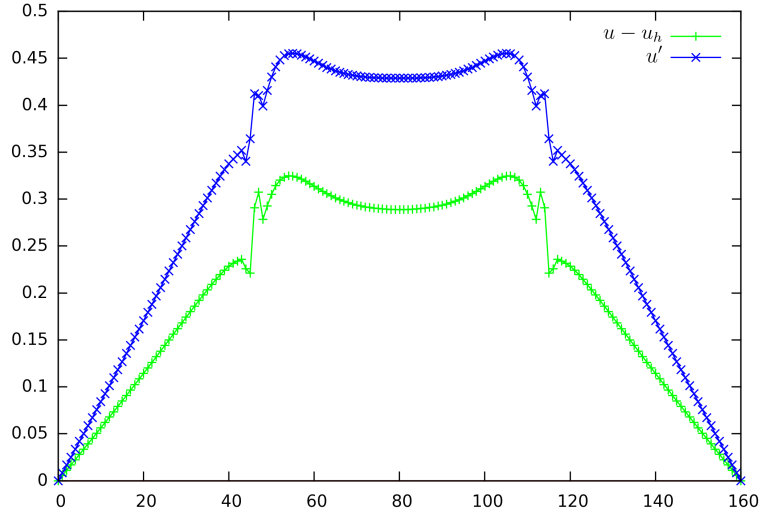


Figure 13: Thin bubble case: comparison of (+) error $u - u_h$ and (\times) Defect-Correction corrector.

calculation, 300 times smaller than the first result. In order to show the behavior of the algorithms for more complex solutions, we have computed the case of a solution involving three bubbles of different sizes. A comparison, in Figure 15, of the solutions with and without adaptation demonstrate that the uniform FMG (results of top range) is unable to produce even a rough approximation with 6561 vertices. Only the solution for 25921 vertices is rather good, with an error L^1 norm of 0.50 but with mean values in bubbles showing still 10% deviations. In contrast, the norm-oriented mesh-adapted solutions (bottom range of Figure 15) show a fast convergence. For 1681 vertices, the solution has the same quality as the uniform 25921-vertices one (error is slightly larger, 0.54). For 6561 and 25921 vertices, the accuracy seems good. The apparent convergence order is greater than two. In contrast, we observe that the Hessian-based calculation is rather disappointing.

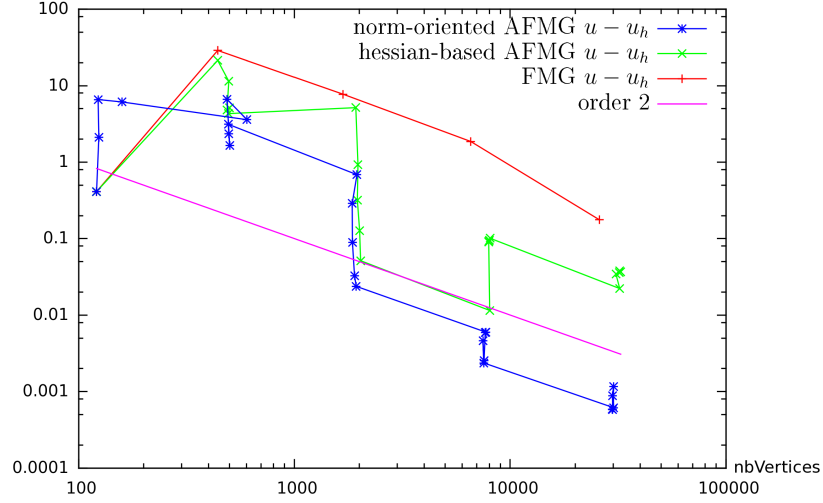


Figure 14: Thin bubble test case: convergence of the error norm $|u - u_h|_{L^2}$ as a function of number of vertices in the mesh for (+) non-adaptative FMG, (\times) Hessian-based adaptative FMG and ($*$) norm-oriented adaptative FMG.

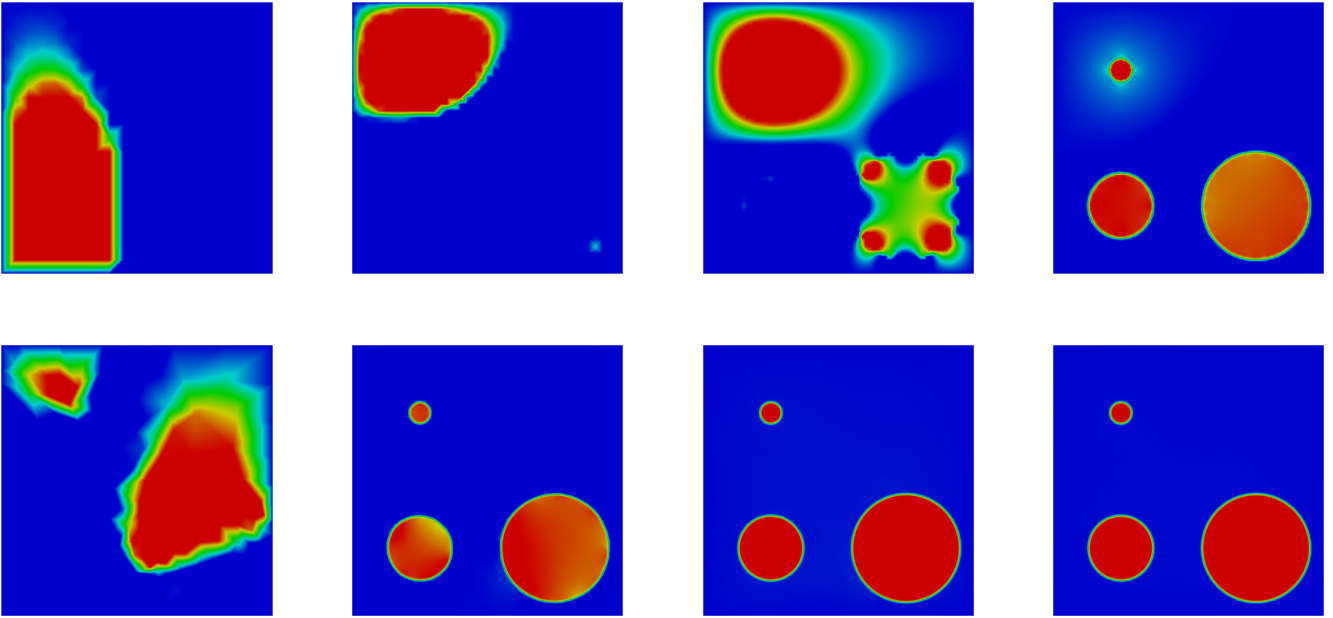


Figure 15: Multiple bubble test case: top: non-adaptative solutions for 441, 1681, 6561, 25921 nodes, and, bottom, the norm-oriented mesh adaptive solution for about the same number of nodes (palette retracted to values between the minimum (0.) and the maximum (1.) of the exact solution).

6.5. Poisson problem with discontinuous coefficient

This test case exemplifies the singularity which is met in the simulation of multi-fluid flows with a large deviation between the densities ρ_1 and ρ_2 of each phase. In the case where a projection algorithm is applied to solve the Navier-Stokes equations for incompressible flow, a Poisson problem

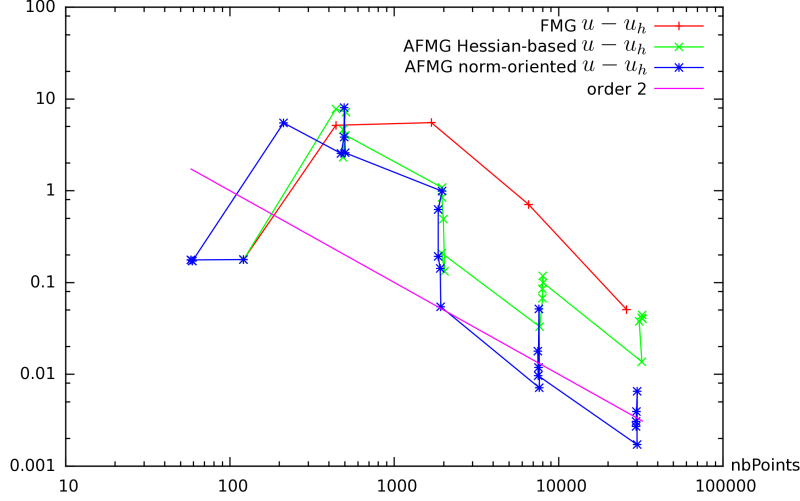


Figure 16: Multiple bubble test case: convergence of the error norm $|u - u_h|_{L^2}$ as a function of number of vertices in the mesh for (+) non-adaptative FMG, (x) Hessian-based adaptive FMG and (*) norm-oriented adaptive FMG.

with discontinuous coefficients has to be solved. An example can be found in [20]. The present case does not satisfy the smoothness assumptions introduced for deriving our method. However, a usual expectation in mesh adaptation is that the methods should also apply well on non-smooth contexts. We consider the equation of Poisson $-div(\frac{1}{\rho}\nabla u) = rhs$ with a discontinuous coefficient taking two different values $1/\rho_1$ and $1/\rho_2$ on two sub-domains Ω_1 and Ω_2 separated by an interface which is a sufficiently smooth curve for having a normal vector. This PDE is mathematically referred as a transmission problem and the solution is continuous across the interface but of discontinuous normal derivatives since:

$$1/\rho_1 \nabla u_1 \cdot \mathbf{n} = 1/\rho_2 \nabla u_2 \cdot \mathbf{n}$$

where u_1 and u_2 are the restrictions of the solution u on Ω_1 and Ω_2 . In our example, we define them as follows

$$u|_{\Omega_i} = u_i = \alpha_i + \beta_i(x^2 + y^2) \quad i = 1, 2.$$

Further, Ω_2 is the disk of center (0.5, 0.5) and of radius 0.2 in the computational domain $]0, 1[\times]0, 1[$ and we have:

$$\begin{aligned} 1/\rho_1 &= 1000. \quad ; \quad \alpha_1 = 1.23579... \quad ; \quad \beta_1 = -2.47158... \\ 1/\rho_2 &= 1. \quad \quad ; \quad \alpha_2 = 100. \quad \quad ; \quad \beta_2 = -2471.58... \end{aligned} \quad (29)$$

This is sketched in Figure 17. In the discrete model, the interface appears only as values of $1/\rho$ evaluated on the vertices of each grid. From the examination of cuts of the correctors in Figures 18 and 19, we observe that their *pointwise* accuracy is somewhat a disaster. Our analysis used intensively smoothness of functions and assumes, for the Defect Correction option that second-order convergence applies, which is not true for this discontinuous case. However, the injection of the correctors in the norm-oriented functional and adjoint performs adequately. The convergence of our process is not really affected by the poor accuracy of the correctors, as can be seen from the comparison of convergence with the correctors or with the exact error, Figure 21. The anisotropy of the finer mesh is illustrated in Figure 20. A convergence in terms of number of vertices with non-adaptative and Hessian-based adaptive is given in Figure 22. We observe that, without mesh

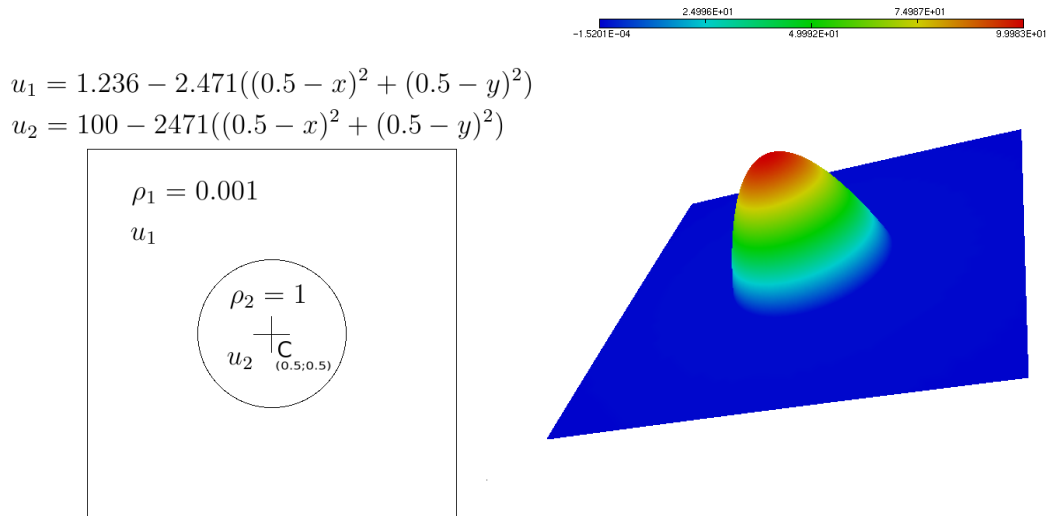


Figure 17: Poisson problem with discontinuous coefficient: sketch of exact solution definition and a typical computation of it.

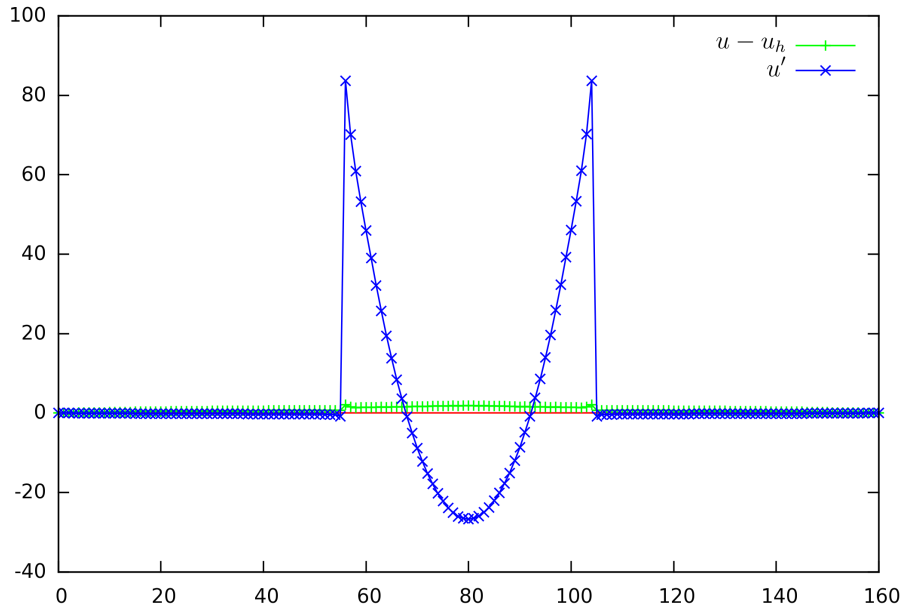


Figure 18: Poisson problem with discontinuous coefficient: comparison of (+) error $u - u_h$ and (\times) *a priori* corrector.

adaptation (crosses +), the convergence order is around 1. This behavior can be explained by the singularity of the solution. In contrast, the overall convergence order of the adaptative process is about two. Note that this convergence will finally deteriorate when we attain the limits of the stretching capabilities of the mesh generator. However, the second-order numerical convergence observed is a usual bonus obtained by anisotropic mesh adaptation which has already been noted in [28]. In [12], the convergence of an anisotropic adaptation has been compared with its isotropic analogous for the same test case. The anisotropic calculation was converging at order two while the isotropic one was converging at order $3/2$. A short analysis of these behaviors is proposed in [15].

Let us now compare with the Hessian-based method. This is the only one of our test cases for which the convergence of error in terms of number of nodes of the norm-oriented formulation is not neatly better (but it is as good) than the analog convergence of the Hessian-based formulation. The

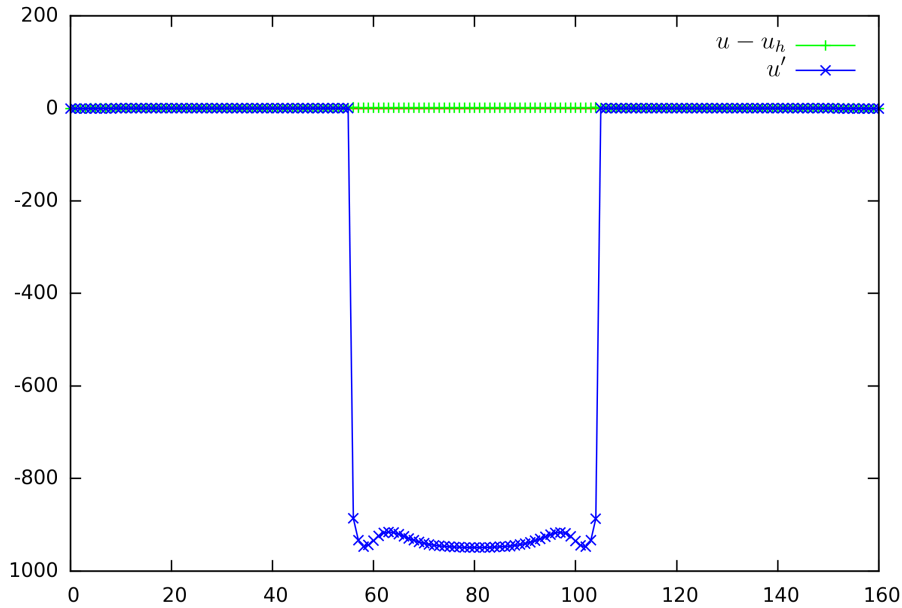


Figure 19: Poisson problem with discontinuous coefficient: comparison of (+) error $u - u_h$ and (\times) Defect-Correction corrector.

error plays a role of weighting in the new formulation. Then a possible explanation of this -relatively-disappointing behavior is that, with the realistic ratio of densities we have chosen, the error is very small close to the discontinuity.

Although not optimised for CPU, the novel algorithm may give a good increase of efficiency. We present, in Figure 23, the behavior of the error as a function of CPU. Three curves are compared. The upper curve corresponds to the non-adaptative FMG calculation. An error of about 0.06 is obtained on 100000 vertices after 25000 seconds of CPU. We present also the Hessian-based adaptative convergence obtained in [12] in which the convergence of MG is adequately controlled by a stopping criterion. Despite the remeshing and re-computing for adaptation, the total time is about twice the non-adaptative FMG time, for an error of 0.001 – 0.002, at least 30 times smaller. The third curve shows the CPU behavior of the norm-oriented algorithm. The total time is about ten times the non-adaptative FMG time, for an error of 0.001. After 25000 seconds of CPU, the error is about 10 times smaller than with the non-adaptative FMG at same time.

6.6. A 1D Boundary layer

The new method has shown a good behavior for all test cases we tried except one which we describe now. It is a boundary layer case with a 1D solution: $u(x, y) = u(x)$ of a Dirichlet-Neumann

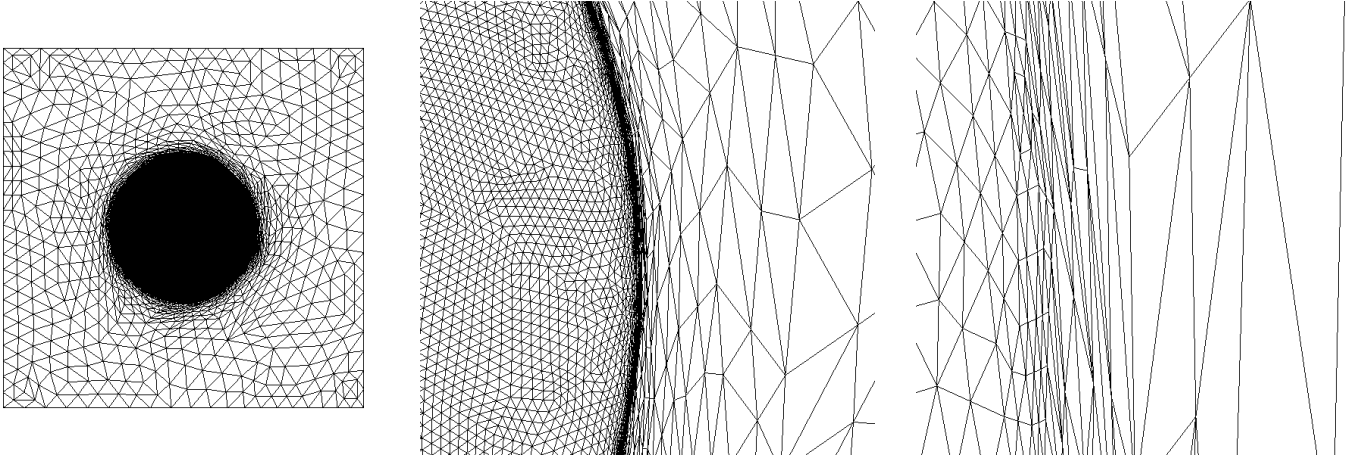


Figure 20: Poisson problem with discontinuous coefficient: views of final mesh: global view, zoom on right part, zoom of the zoom.

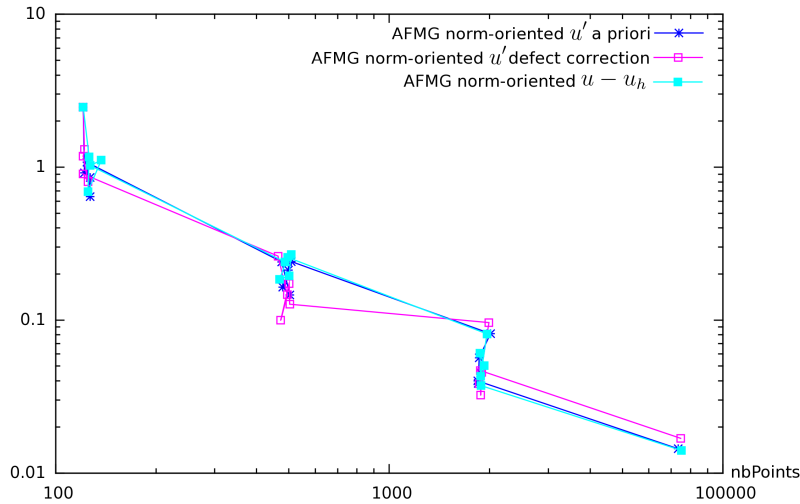


Figure 21: Poisson problem with discontinuous coefficient: convergence based on (×) the *a priori* corrector or on the (*) Defect-Correction one, compared with (□) a virtual adaptation controlled by $u - u_h$.

problem $-\Delta u = rhs$ with

$$rhs(x, y) = (\alpha^2(\exp(1/\alpha) - 1))^{-1} \exp(x/\alpha) \quad ; \quad \alpha = 0.03.$$

We check first the correctors. Both seem adequate on a uniform grid, as shown with a horizontal cut depicted on Figure 24 and Figure 25.

In Figure 26, the non-adaptative FMG produces an approximation error of 0.003 (30000 vertices). This convergence is relatively satisfactory, being a second order convergence. However, in order to reach a 10^{-7} error level, several hundred millions vertices will be necessary with this sequence of uniform meshes. A second curve is obtained with the adaptative FMG with the Hessian-based criterion. Final convergence is disappointing since the slope is first-order. The same problem appears with our new algorithm. A deeper examination of adaptation criteria has shown that the high derivatives of the right-hand side f are very close to boundary $x = 1$. It could not be seen by the algorithm, because of the weighting by the adjoint u^* , which is zero at this boundary. We have

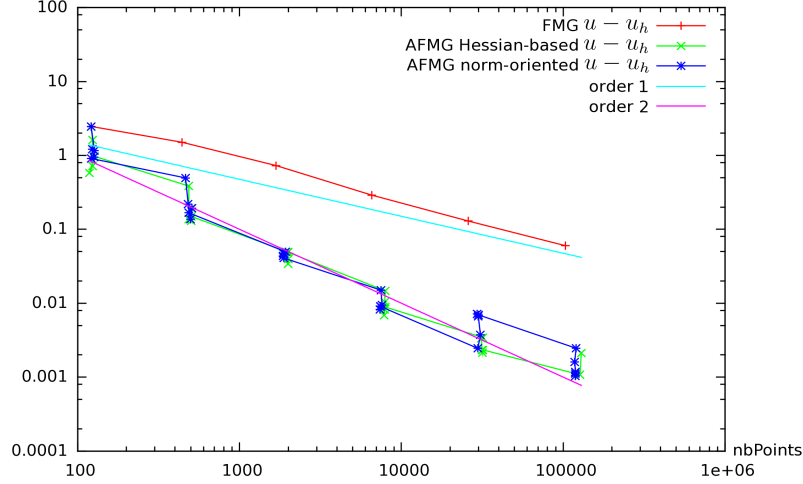


Figure 22: Poisson problem with discontinuous coefficient: convergence of the error norm $|u - u_h|_{L^2}$ as a function of number of vertices in the mesh for (+) non-adaptative FMG, (\times) Hessian-based adaptative FMG and ($*$) norm-oriented adaptative FMG.

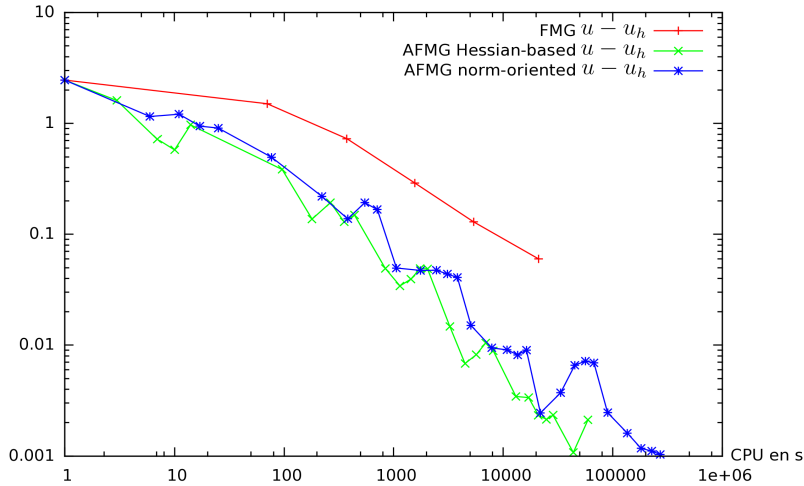


Figure 23: Poisson problem with discontinuous coefficient: efficiency analysis : CPU time for (+) non-adaptative FMG, (\times) Hessian-based adaptative FMG and ($*$) norm-oriented adaptative FMG.

replaced the norm-oriented optimum metric by its intersection with the metric based on the Hessian of f . Then, the convergence improved a lot. In contrast, introducing the same metric intersection in the other cases did not produce second-order convergence.

7. Conclusion

The norm-oriented mesh adaptation method is an answer to a well formulated problem. Considering a numerical scheme, here the most used FEM, and prescribing a norm, we want to find the mesh giving the smallest approximation error in that norm for a given number of vertices. The norm-oriented mesh adaptation method transforms the problem into an optimization problem which is mathematically well-posed. It relies on the following other features.

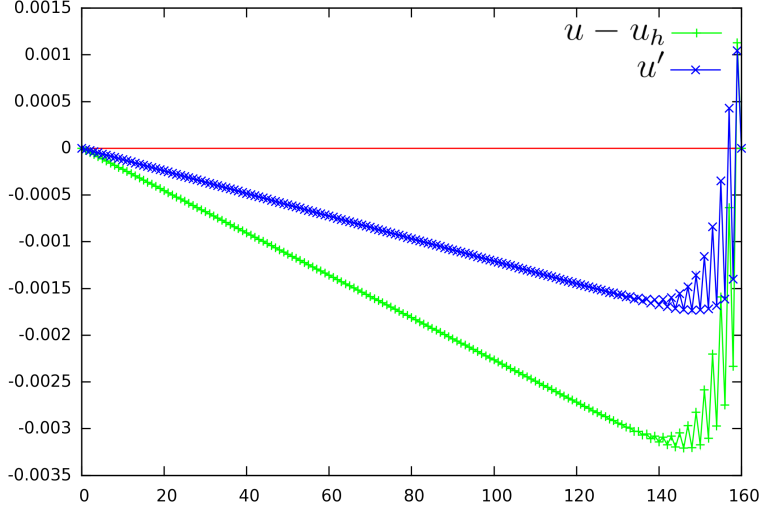


Figure 24: 1D Boundary Layer: comparison of error cuts for $y = 0.5$: plus signs (+) depict the approximation error $u - u_h$ and crosses (\times) depict the *a priori* corrector u'_{prio} . The corrector is able to correct about 55% of the approximation error.

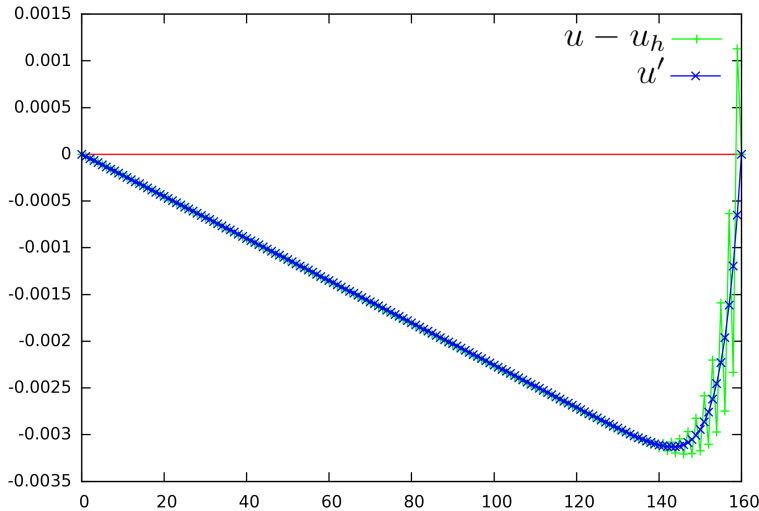


Figure 25: 1D Boundary Layer: comparison of error cuts for $y = 0.5$: plus signs (+) depict the approximation error $u - u_h$ and crosses (\times) depict the Defect-Correction corrector u'_{DC} . The corrector is able to correct about 80% of the approximation error.

A corrector represents the approximation error. We give two examples of correctors. An *a priori* corrector is built from the variational discrete statement. A Defect-Correction corrector is built from a finer-mesh defect correction principle. These correctors appear as not very accurate but sufficiently accurate for our purpose. According to the type of approximation, at least the second one, Defect-Correction is extendable to many models and schemes.

The norm-oriented algorithm is presented as a natural extension of the goal-oriented algorithm which, in our formulation, is itself a natural extension of the Hessian-based algorithm. More precisely, while the Hessian-based algorithm solves only the PDE under study in the mesh-adaptation loop, the goal-oriented algorithm also solves an adjoint system (with linearised operator, transposed). The norm-oriented algorithm solves three systems, a corrector (linearised system with an adhoc

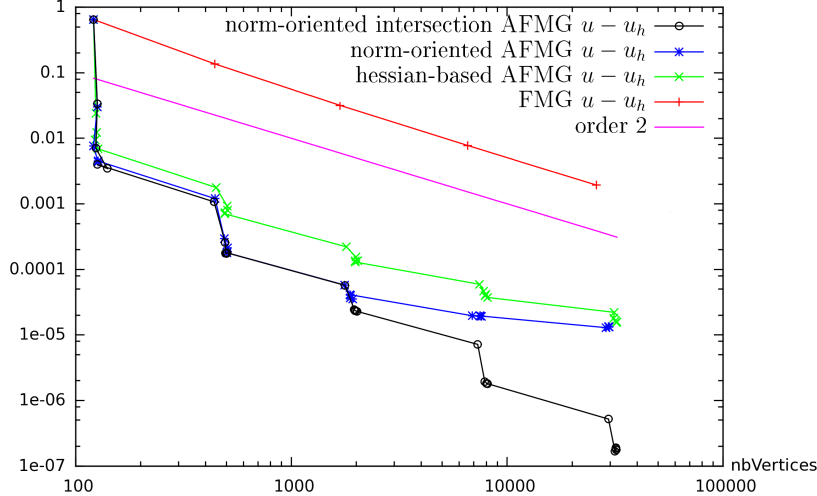


Figure 26: 1D Boundary Layer: convergence of the error norm $|u - u_h|_{L^2}$ as a function of number of vertices in the mesh for (+) non-adaptative FMG, (\times) Hessian-based adaptative FMG, ($*$) norm-oriented adaptative FMG, and (o) norm-oriented adaptative FMG with intersection with RHS Hessian.

RHS), an adjoint (linearised and transposed with the corrector as RHS) and the PDE itself. The three algorithms have in common an anisotropic *a priori* error analysis and a metric-based mesh parameterisation.

The Hessian-based method produces convergent solution fields but does not take into account the precise equation and discretization. The goal-oriented method takes into account equation and discretization but is too focused on a particular output and does not generally produce convergent solution fields. The norm-oriented method has the advantages of both.

In order to show the improvement obtained with respect to previous methods, we compare, in our experiments, the two field-convergent options, *i.e.* the Hessian-based method and the norm-oriented method. Our benchmark examines convergence to continuous thanks to the application of a Full-Multigrid (FMG) process. Approximation errors can then be compared as functions of the number of degrees of freedom. Although of larger complexity, the mesh adaptative algorithms are generally more efficient in terms of CPU time than the non-adaptative one. The advantage is reinforced by the fact that the same level of error is reached with a much lower number of vertices. Secondly, although the elliptic context is known as favourable to Hessian-based methods, the norm-oriented approach behaves notably better than Hessian-based in terms of approximation error for a given number of vertices. This is observable in particular for singular or stiff contexts.

We have presented an example of comparison of computational effort but our algorithm is not optimised. It is, therefore, in several cases, less efficient than Hessian-based adaptative FMG.

The method is rather general and it is now applied to more complex PDE models from CFD (Euler, Navier-Stokes), see [27], involving dominant advection effects for which a Hessian-based approach is much less efficient.

8. Acknowledgements

This work was partly done in the MAIDESC ANR project which is supported by the french ministry of Research under contract ANR-13-MONU-0010. The fellowship of Gautier Br ethes is supported by Lemma and r egion Provence-Alpes-C ote d'Azur.

References

- [1] A. Agouzal, K. Lipnikov, and Y. Vasilevski. Adaptive generation of quasi-optimal tetrahedral meshes. *East-West Journal*, 7(4):223–244, 1999.
- [2] A. Agouzal, K. Lipnikov, and Yu. Vassilevski. Hessian free metric based mesh adaptation via geometry of interpolation error. *J. Computational Mathematics and Mathematical Physics*, 50(1):124–138, 2010.
- [3] A. Agouzal and Yu. V. Vassilevski. Minimization of gradient errors of piecewise linear interpolation on simplicial meshes. *Comp.Meth. Appl.Mech.Engnr.*, 199:2195–2203, 2010.
- [4] F. Alauzet. *Adaptation de maillage anisotrope en trois dimensions. Application aux simulations instationnaires en Mécanique des Fluides*. PhD thesis, Université Montpellier II, Montpellier, France, 2003. (in French).
- [5] F. Alauzet, S. Borel-Sandou, L. Daumas, A. Dervieux, Q. Dinh, S. Kleinveld, A. Loseille, Y. Mesri, and G. Rogé. Multi-model and multi-scale optimization strategies. application to sonic boom reduction. *European Journal of Computational Mechanics*, 17(1-2):191–214, 2008.
- [6] E. Arian and M.D. Salas. Admitting the inadmissible: Adjoint formulation for incomplete cost functionals in aerodynamic optimization. *AIAA Journal*, 37(1):37–44, 1999.
- [7] R. Becker and R. Rannacher. A feed-back approach to error control in finite element methods: basic analysis and examples. *East-West J. Numer. Math.*, 4:237–264, 1996.
- [8] A. Belme. *Aérodynamique instationnaire et méthode adjointe*. PhD thesis, Université de Nice Sophia Antipolis, Sophia Antipolis, France, 2011. (in French).
- [9] A. Belme, A. Dervieux, and F. Alauzet. A fully anisotropic goal-oriented mesh adaptation for unsteady flows. In *Proceedings of the 5th ECCOMAS CFD Conf.*, 2010.
- [10] A. Belme, A. Dervieux, and F. Alauzet. Time accurate anisotropic goal-oriented mesh adaptation for unsteady flows. *J. Comp. Phys.*, 231(19):6323–6348, 2012.
- [11] M. Berger. *A panoramic view of Riemannian geometry*. Springer Verlag, Berlin, 2003.
- [12] G. Brèthes, O. Allain, and A. Dervieux. A mesh-adaptative metric-based Full-Multigrid for the Poisson problem. *I.J. Numer. Meth. Fluids*, 79(1), 2015. pre-print: <http://www-sop.inria.fr/members/Gautier.Brethes/article-ADA-MG.pdf>.
- [13] M.J. Castro-Díaz, F. Hecht, B. Mohammadi, and O. Pironneau. Anisotropic unstructured mesh adaptation for flow simulations. *Int. J. Numer. Meth. Fluids*, 25:475–491, 1997.
- [14] L. Chen, P. Sun, and J. Xu. Optimal anisotropic meshes for minimizing interpolation errors in L^p -norm. *Math. Comp.*, 76(257):179–204, 2007.
- [15] F. Courty, D. Leservoisier, P.L. George, and A. Dervieux. Continuous metrics and mesh adaptation. *Appl. Numer. Math.*, 56(2):117–145, 2006.
- [16] J. Dompierre, M.G. Vallet, M. Fortin, Y. Bourgault, and W.G. Habashi. Anisotropic mesh adaptation: towards a solver and user independent CFD. In *AIAA 35th Aerospace Sciences Meeting and Exhibit*, AIAA-1997-0861, Reno, NV, USA, Jan 1997.

- [17] A. Ern and M. Vohralík. Polynomial-degree-robust a posteriori estimates in a unified setting for conforming, nonconforming, discontinuous Galerkin, and mixed discretizations. In *La Serena Numerica II, Octavo Encuentro de Análisis Numérico de Ecuaciones Diferenciales Parciales*, Enero 14 - 16, 2015. HAL Preprint 00921583, (2014).
- [18] L. Formaggia and S. Perotto. Anisotropic a priori error estimates for elliptic problems. *Numer. Math.*, 94:67–92, 2003.
- [19] M.B. Giles and N.A. Pierce. Improved lift and drag estimates using adjoint Euler equations. *AIAA paper*, 99-3293, 1999.
- [20] D. Guégan, O. Allain, A. Dervieux, and F. Alauzet. An L^∞ - L^p mesh adaptive method for computing unsteady bi-fluid flows. *Int. J. Numer. Meth. Engng*, 84(11):1376–1406, 2010.
- [21] R. Hartman. Multitarget error estimation and adaptivity in aerodynamic flow simulations. *SIAM Journal on Scientific Computing*, 31(1):708–731, 2008.
- [22] W. Huang. Metric tensors for anisotropic mesh generation. *J. Comp. Phys.*, 204(2):633–665, 2005.
- [23] A. Loseille. *Adaptation de maillage anisotrope 3D multi-échelles et ciblée à une fonctionnelle pour la mécanique des fluides. Application à la prédiction haute-fidélité du bang sonique*. PhD thesis, Université Pierre et Marie Curie, Paris VI, Paris, France, 2008. (in French).
- [24] A. Loseille and F. Alauzet. Continuous mesh framework. Part I: well-posed continuous interpolation error. *SIAM Journal on Numerical Analysis*, 49(1):38–60, 2011.
- [25] A. Loseille and F. Alauzet. Continuous mesh framework. Part II: validations and applications. *SIAM Journal on Numerical Analysis*, 49(1):61–86, 2011.
- [26] A. Loseille, A. Dervieux, and F. Alauzet. Fully anisotropic goal-oriented mesh adaptation for 3D steady Euler equations. *J. Comp. Phys.*, 229(8):2866–2897, 2010.
- [27] A. Loseille, A. Dervieux, and F. Alauzet. Anisotropic norm-oriented mesh adaptation for compressible flows. In *53rd AIAA Aerospace Sciences Meeting*, AIAA-2015-2037, Kissimmee, Florida, Jan 2015.
- [28] A. Loseille, A. Dervieux, P.J. Frey, and F. Alauzet. Achievement of global second-order mesh convergence for discontinuous flows with adapted unstructured meshes. In *37th AIAA Fluid Dynamics Conference and Exhibit*, AIAA-2007-4186, Miami, FL, USA, Jun 2007.
- [29] F. Magoules. *Computational Fluid Dynamics*. CRC Press, Boca Raton, London, New York, Washington D.C., 2011.
- [30] N.A. Pierce and M.B. Giles. Adjoint recovery of superconvergent functionals from PDE approximations. *SIAM Review*, 42(2):247–264, 2000.
- [31] Y.V. Vasilevski and K.N. Lipnikov. An adaptive algorithm for quasi-optimal mesh generation. *Comput. Math. Math. Phys.*, 39(9):1468–1486, 1999.
- [32] Y.V. Vasilevski and K.N. Lipnikov. Error bounds for controllable adaptive algorithms based on a Hessian recovery. *Computational Mathematics and Mathematical Physics*, 45(8):1374–1384, 2005.

- [33] Yu. Vassilevski, V.G. Dyadechko, and K. Lipnikov. Hessian-based anisotropic mesh adaptation in domains with discrete boundaries. *Russ. J. Numer. Anal. Math. Modelling*, 20(4):391–402, 2005.
- [34] Yu. V. Vassilevski and A. Agouzal. An unified asymptotical analysis of interpolation errors for optimal meshes. *Doklady Mathematics*, 72(3):879–882, 2005.
- [35] D.A. Venditti and D.L. Darmofal. Adjoint error estimation and grid adaptation for functional outputs: application to quasi-one-dimensional flow. *J. Comp. Phys.*, 164(1):204–227, 2000.
- [36] R. Verfürth. *A Posteriori Error Estimation Techniques for Finite Element Methods*. Oxford University Press, Oxford, 2013.
- [37] M. Yano and D. Darmofal. An optimization framework for anisotropic simplex mesh adaptation: application to aerodynamics flows. *AIAA Paper*, 2012-0079, 2012.
- [38] O.C. Zienkiewicz and J.Z. Zhu. The superconvergent patch recovery and a posteriori error estimates. Part 1: The recovery technique. *Int. J. Numer. Meth. Engng*, 33(7):1331–1364, 1992.