MAIDESC-T5-D3-BIS: A continuous/tensorial synthesis for mesh adaptation

G. Brèthes[a], T. Coupez[b], A. Dervieux[a]

[a] *INRIA, Projet Tropics, 2004 route des lucioles - BP 93, 06902 Sophia Antipolis Cedex, France*
[b] *Ecole Centrale de Nantes, ICI-ECN,1 rue de la Noë, BP 92101, 44321 Nantes Cdex 3, France*

Abstract

This paper extends a series of papers dealing with a continuous analysis of metric-based anisotropic mesh adaption. It discusses an equation-based adaptation, addressing directly approximation error through the use of an adjoint state. The model problem is a Poisson problem. Continuous-metric methods were developed for this case and the novelty of this paper is to extend a discrete context introduced recently by one of the authors in order to rely on a possibly sharper analysis of the approximation error. The resulting optimal metric has a different anisotropic component. The novel formulation is compared with the continuous formulation for a few test cases involving high gradient layers and gradient discontinuities.

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

1. Introduction

This paper addresses anisotropic mesh adaptation for PDE's (Partial Differential Equations). We focus on methods which prescribe a somewhat optimal mesh under the form of a parametrization of it by a Riemannian metric. These Riemann metrics can be considered under a continuous standpoint, as proposed in [21, 22]. A dual way proposed in [15, 16] relies on mesh-based tensorial statistics.

Continuous and tensorial metrics both rely on the parametrization of the mesh by a spatial field defining in any point of the computational domain a matrix giving information on mesh size in all the spatial directions.

Both methods solve an optimality system, the continuous metric builds a continuous optimality system which has, afterwards, to be discretised and solved, while the tensorial metric builds a discrete optimality system to be solved directly. Also, the continuous metric theory defines the ideal metric to be choosen. The resulting ideal mesh produced by the optimization is a unit mesh for the ideal metric. In contrast, the tensorial metric obtained from optimization in [15, 16] defines the modification to apply to the current mesh in order to obtain the ideal mesh. Then the way to parameterize the final mesh with the two metrics is different, since the ideal mesh is of length unity (any edge has a length 1 for the metric) for the continuous metric, while the tensorial metric defines the ideal mesh from local directional amplifications of the background mesh. Similarly, the constraint imposing a prescribed number of nodes is formulated vertices by vertices for the continuous metric and edges by edges for the tensorial method.

Both methods apply to a typical family of optimal metric-based method for PDE's, the P_1-Interpolation-based/Hessian-based method. Hessian-based methods involves the equi-distribution method, which turns out to finding the metric which minimizes a L^∞ norm of the interpolation

error of one or several sensors depending on the PDE solution. Also involved in Hessian-based methods is the multiscale method, which minimizes the L^p interpolation error of the sensors. Cf. [13, 17, 3, 24, 18, 25, 19, 14, 4, 28]. These Hessian-based methods, while taking into account the features of the PDE solution, do not take into account the features of the PDE itself. However, if sensors are cleverly chosen, a good convergence of the whole approximate solution field to the exact solution field is usually observed.

Goal-oriented methods allow to take into account the equation, a combination with anisotropic Hessian-based adaption is proposed in [26]. Goal-oriented optimal methods [20, 9, 28], minimize with respect to the metric the approximation error committed on the evaluation of a scalar functional depending on the PDE solution. They do take into account the features of the PDE, typically through the use of an adjoint state. Goal-oriented methods needs also to rely on an error estimate (and its sensitivity to mesh).

Several methods have been proposed for reducing the approximation error through an estimate. A pioneering approach is the work Becker and Rannacher [7] which rely, as most estimate-based work, on an *a posteriori* estimate. A good synthesis concerning *a posteriori* estimates is [27]. An interest of *a posteriori* estimate is that it is expressed in terms of the approximate solution, assumed to be available in a mesh adaption 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. However, deducing the best anisotropic mesh from an *a posteriori* estimate remains difficult (see however [28]).

A priori estimates rely quasi systematically on Taylor series, either through divided differences, or through polynomial approximation of functions. Then approximations of higher order derivatives of solution need be built from the approximate solution. This is a delicate job since nothing ensures that a higher order derivative of the approximate solution is a good approximation of the corresponding higher order derivative of the exact solution. In contrast with the *a posteriori* option, the Taylor series can be easily used for proposing a somewhat optimal mesh. Further, *A priori* estimates can also provide correctors: an example is given in [12]. In [20, 9], in order to be able to solve the goal-oriented, metric-based, mesh optimization problem, the authors introduce an a priori analysis which restricts to the main asymptotic term of the local error. In this paper we use the tensorial formulation in order to build a novel *a priori* estimate for the Poisson equation.

With the metric-based goal-oriented formulation, metric-based mesh adaptation becomes a well-posed optimization problem for the reduction of a genuine approximation error. However, goal-oriented optimal methods are specialised to a given scalar output. Features of the solution field which are not related to this output may be neglected by the automatic mesh improvement. As a consequence, these methods do not provide a convergent solution field. In the present paper, we study a norm-based formulation (according to [12]) in which the user can prescribe a norm of error $|u - u_h|$ which the algorithm will minimize with respect to the metric parametrization of the mesh. As a consequence, with an adequate choice of the norm, the norm-oriented mesh adaptation produces convergent solution fields.

The continuous approach for Hessian-based, goal-oriented, and norm-oriented has been defined in other papers like [20, 9, 12].

The purpose of this paper is to analyse the possible novelties which can be derived from the application of a tensorial method to Hessian-based, goal-oriented, and norm-oriented problematics. The main feature of tensorial approach which we shall exloit is the tensorial inversion of main error term. In order to adapt this feature to L^1-Hessian, to goal-oriented, to norm-oriented problematics,

we unify the parametrization by choosing the unit-mesh formulation and by measuring the number of nodes on a vertex basis..

Presentation of plan: .....

2. Poisson problem approximation

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

$$Au = f \quad or \quad u \in V \quad \forall\ \phi \in V \quad a(u,\phi) = (f,\phi) \tag{1}$$

To fix the ideas and simplify notations,

$$A = -\sum \frac{\partial}{\partial x_k}\frac{\partial}{\partial x_k}$$

But the extension to a coercive case where $A = -\sum \frac{\partial}{\partial x_k}(a_{k\ell}(\mathbf{x})\frac{\partial}{\partial x_\ell})$ (where $a_{k\ell}$ is a scalar, possibly discontinuous, field) is not difficult. 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:

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

The finite-element discretisation of (1) is written:

$$u_h \in V_h \quad \forall\ \phi_h \in V_h \quad a(u_h,\phi_h) = (f,\phi_h). \tag{2}$$

We are interested first in getting estimates of the approximation error $u_h - u$. Let N be the dimension of V_h, that is the number of vertices in τ_h. We observe that (2) is equivalent to computing the array $\mathbf{u}_h$ of the degrees of freedom of the discrete solution:

$$\mathbf{u}_h \in \mathbb{R}^N \ ;\ \mathbf{A}_h\mathbf{u}_h = \mathbf{f}_h. \tag{3}$$

From the above array we derive u_h by

$$u_h = \sum_{i=1,N} \mathbf{u}_{h,i} N_i(\mathbf{x})$$

where the N_i are the canonic finite-element basis of V_h:

$$N_i \in V_h, \quad N_i(\mathbf{x}_j) = 1 \text{ if } i = j,\ 0 \text{ else.}$$

We also introduce the interpolation operator Π_h:

$$for \quad v \in V \cap H^2(\Omega), \quad \Pi_h v \in V_h, \quad (\Pi_h v - v)(\mathbf{x}_i) = 0 \quad \forall \mathbf{x}_i \text{ vertex of } \tau_h.$$

An *a priori* error analysis can be applied in order to build a corrector, which is a signed approximate, and not a upper bound, of the approximation error. We start from the discrete above statement

$$a(u_h,\phi_h) = (f_h,\phi_h) \quad \forall \phi_h \in V_h.$$

and observe that for the exact solution satisfies:

$$a(u,\phi_h) = (f,\phi_h) \quad \forall \phi_h \in V_h.$$

Then

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

Assuming that the solution u is sufficiently smooth, we get:

$$a(\Pi_h u - u_h, \phi_h) = a(\Pi_h u - u, \phi_h) + (f - f_h, \phi_h) \quad \forall \phi_h \in V_h. \tag{4}$$

We call $\Pi_h u - u_h$ the *implicit error*. It differs from the approximation error by an interpolation error:

$$u - u_h = u - \Pi_h u \quad + \quad \Pi_h u - u_h.$$

The rest of the section is devoted to finding an approximate of the implicit error. In practice, we need to evaluate the RHS for any basis function N_i. The second term of RHS is easy to evaluate (we know f and f_h). The first term of RHS can be transformed as follows:

$$\begin{aligned} a(\Pi_h u - u, \phi_h) &= \sum_T \int_T \nabla \phi_h \nabla (\Pi_h u - u) \, \mathrm{d}x\mathrm{d}y \\ &= \sum_T \int_{\partial T} (\Pi_h u - u) \nabla \phi_h \cdot \mathbf{n} \, \mathrm{d}\sigma. \end{aligned}$$

Then we get:

$$\begin{aligned} a(\Pi_h u - u, \phi_h) &= K(\phi, u_h) \qquad \text{with} \\ K(\phi, u_h) &= \sum_{\partial T_{ij}} \nabla(\phi_h|_{T_i} - \phi_h|_{T_j}) \cdot \mathbf{n}_{ij} \int_{\partial T_{ij}} (\Pi_h u - u) \, \mathrm{d}\sigma \end{aligned} \tag{5}$$

where the last sum is taken for all edges $ij = \partial T_{ij}$ (2D case) separating triangles T_{ij}^+ and T_{ij}^- 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 introduce the following interpolation estimator π:

$$for \;\; v \in V \cap \mathcal{C}^2(\bar{\Omega}), \quad \pi v - v = H(v) \delta x \delta y$$

where $H(v)$ holds for the Hessian matrix of u and $\delta x \delta y$ hold for local mesh sizes in Cartesian directions. Further, in $\pi v - v$, the second derivatives of the unknown u are approximated by approximate second derivatives of the discrete solution as for example in [5]. This more or less assumes that the second derivatives of the approximate solution are not a too bad approximation of the second derivatives of the unknown exact solution. This last statement is far from being stated. However, under this condition our *corrector* is defined by:

$$\begin{aligned} &a(u'_{prio}, \phi_h) = K(\phi_h, u_h) \qquad \text{with} \\ &K(\phi_h, u_h) = \sum_{\partial T_{ij}} (\nabla \phi_h|_{T_i} - \nabla \phi_h|_{T_j}) \cdot \mathbf{n}_{ij} \int_{\partial T_{ij}} (\pi_h u_h - u_h) \, \mathrm{d}\sigma. \end{aligned} \tag{6}$$

In practice, the term $\pi_h u_h - u_h$ is built on the edge T_{ij} as a quadratic function vanishing at both extremities of T_{ij}, and of second derivative in direction T_{ij} equal to the approximate second derivative in same direction of u_h.

The above relation (6) is routinely used for building mesh adaption loops, see [10]. The use of (6) for deriving a genuine corrector for u_h is a more delicate job, since the derivation on the test function cannot hide that we need a consistent approximation of the second derivative of the interpolation error which may mean that we need the convergence of a fourth derivative of the approximate solution.

3. Continuous metric parametrization

3.1. Mesh parametrization

We recall the continuous mesh framework, introduced in [21, 22]. The main idea of this framework is to model discrete meshes by Riemannian metric fields. It allows us to define a differentiable optimization problem [2, 6], *i.e.,* to apply on the class continuous metrics a calculus of variations which cannot be appied 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 [11] $\mathcal{M} = (\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$. 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: $r_i = h_i^3 \left(h_1 h_2 h_3\right)^{-1}$. The diagonalisation of $\mathcal{M}(\mathbf{x})$ writes:

$$\mathcal{M}(\mathbf{x}) = d^{\frac{2}{3}}(\mathbf{x})\, \mathcal{R}(\mathbf{x}) \begin{pmatrix} r_1^{-\frac{2}{3}}(\mathbf{x}) & & \\ & r_2^{-\frac{2}{3}}(\mathbf{x}) & \\ & & r_3^{-\frac{2}{3}}(\mathbf{x}) \end{pmatrix} {}^t\mathcal{R}(\mathbf{x}), \tag{7}$$

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

$$\mathcal{C}(\mathcal{M}) = \int_\Omega d(\mathbf{x})\, \mathrm{d}\mathbf{x} = \int_\Omega \sqrt{\det(\mathcal{M}(\mathbf{x}))}\, \mathrm{d}\mathbf{x}. \tag{8}$$

Given a continuous mesh $\mathcal{M}$, we shall say, following [21, 22], 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 $(\mathbf{e}_i)_{i=1\ldots3}$, verifies:

$$\forall i \in [1,3], \quad \ell_{\mathcal{M}}(\mathbf{e}_i) \in \left[\frac{1}{\sqrt{2}}, \sqrt{2}\right],$$

in which the length of an edge $\ell_{\mathcal{M}}(\mathbf{e}_i)$ is defined as follows:

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

The unit edge property of unit mesh writes also in short:

$$\text{For a unit mesh } \mathbf{x}^{\mathcal{M}} \text{ , any edge } \mathbf{x}_{ij}^{\mathcal{M}} \text{ satisfies } \left(\mathbf{x}_{ij}^{\mathcal{M}} - \mathbf{x}_{ij}^{\mathcal{M}}, \mathcal{M}(\mathbf{x}_{ij}^{\mathcal{M}} - \mathbf{x}_{ij}^{\mathcal{M}})\right) = 1.$$

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, but these meshes have properties sufficiently close to each others so that we consider these meshes as an equivalence class of meshes. We henceforward denote by $\mathcal{M}$ both the metric and the corresponding unit mesh.

3.2. Optimal continuous metric

We recall, following [21, 22], the main features of the metric-based analysis initiated in several papers like [17, 13, 3]. The continuous interpolation error of a function u defined on the computational domain is denoted now:

$$u - \pi_{\mathcal{M}} u = |tr(\mathcal{M}^{-\frac{1}{2}} |H_u| \mathcal{M}^{-\frac{1}{2}})| \tag{9}$$

where H_u is the Hessian of u. Let denote also $\mathcal{M}$ a unit mesh for metric $\mathcal{M}$. We shall use the estimate

$$|u - \Pi_{\mathcal{M}} u| \approx \frac{1}{8} |u - \pi_{\mathcal{M}} u|. \tag{10}$$

Once we have a continuous tensorial error kernel, we consider minimizing:

$$j_p(\mathcal{M}) = \|u - \pi_{\mathcal{M}} u\|_{\mathbf{L}^p(\Omega_h)} \tag{11}$$

and 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. In the case of a bounded p, after solving analytically this optimization problem, we get -without using the fact that H is anything but a positive symmetric matrix- the unique optimal $(\mathcal{M}_{\mathbf{L}^p}(\mathbf{x}))_{\mathbf{x}\in\Omega}$ as:

$$\mathcal{M}_{\mathbf{L}^p} = \mathcal{K}_p(1, H \text{ with } \mathcal{K}_p(1,u) = D_{\mathbf{L}^p} \left(\det(H)\right)^{\frac{-1}{2p+2}} H \text{ and } D_{\mathbf{L}^p} = N^{\frac{2}{2}} \left(\int_\Omega (\det(H))^{\frac{p}{2p+2}} \right)^{-\frac{2}{2}}, \tag{12}$$

where $D_{\mathbf{L}^p}$ is a global normalization term set to obtain a continuous mesh with complexity N and $(\det(H))^{\frac{-1}{2p+2}}$ is a local normalization term accounting for the sensitivity of the $\mathbf{L}^p$ norm.

A particular case: L^∞-norm/iso-distribution It is important to remark that error iso-distribution is taken into account by setting $p = \infty$, a limiting case for which we get:

$$(\det(H))^{\frac{-1}{2\infty+2}} = 1.$$

and

$$\mathcal{M}_{\mathbf{L}^\infty} = \mathcal{K}_\infty(1, H) \text{ with } \mathcal{K}_\infty(1,H) = D_{\mathbf{L}^\infty} H$$

where $D_{\mathbf{L}^\infty}$ is defined from the specification of the number of nodes of the mesh.

Another way to see it is to write that the error is uniform, indeed:

$$\mathcal{M}_{\mathbf{L}^\infty}(\mathbf{x}) = \text{const. (indep. of } \mathbf{x}) \; H \quad \Rightarrow \quad \text{trace}\big(\mathcal{M}_{\mathbf{L}^\infty}^{-\frac{1}{2}}(\mathbf{x}) H(\mathbf{x}) \mathcal{M}_{\mathbf{L}^\infty}^{-\frac{1}{2}}(\mathbf{x})\big) = \text{const. (indep. of } \mathbf{x}).$$

Main case under study: L^1-norm optimisation The rest of the paper concentrates with the case:

$$p = 1$$

Replacing the optimal metric $\mathcal{M}_{\mathbf{L}^1}$ in the L^1 norm shows that second-order convergence is obtained for smooth contexts. This can also be extended to non-smooth ones, cf. [23].

Let k a sufficiently smooth scalar function defined on Ω. We shall be, in the sequel, interested in minimizing the right-hand side of:

$$|(k, u - \Pi_{\mathcal{M}} u)_\Omega| \approx \int_\Omega \text{trace}\big(\mathcal{M}^{-\frac{1}{2}}(\mathbf{x}) |k(\mathbf{x}) H(\mathbf{x})| \mathcal{M}^{-\frac{1}{2}}(\mathbf{x})\big) \mathrm{d}\mathbf{x}. \tag{13}$$

The optimum metric is given by:

$$\mathcal{M}_{opt}^{1,k} = \mathcal{K}_1(k, H) \text{ with } \mathcal{K}_1(k,H) = D_{opt}^{1,k} \left(\det |kH|\right)^{\frac{-1}{4}} |kH| \text{ and } D_{opt}^{1,k} = N \left(\int_\Omega (\det |kH|)^{\frac{1}{4}} \right). \tag{14}$$

It is interesting to compare this result with the result of equidistribution, at least for the particular case of an interpolation error. We observe that:

$$\begin{aligned} \mathcal{M}_{opt}^{1,k} = \text{const.} \quad |k|^{\frac{3}{4}} |(\det |H|)^{\frac{-1}{4}} |H| \;&=\; \text{const.} \;\; |H_k| \\ H_k = |k|^{\frac{3}{4}} |(\det |H|)^{\frac{-1}{4}} H \end{aligned} \tag{15}$$

Which means that the error minimisation in $L^1_{weight,k}$ is equivalent to an equi-distribution process with a matrix H corrected by a scalar factor $|k|^{\frac{3}{4}} |(\det |H|)^{\frac{-1}{4}}$:

$$\mathcal{M}_{opt}^{1,k} = \text{const.} \;\; \mathcal{K}_\infty(|k|^{\frac{3}{4}} |(\det |H|)^{\frac{-1}{4}}, H).$$

3.3. Idea of the proofs

Pointwise optimization: For both norms, *same level of error in each direction* around a given point i of the computational domain (Loseille-Alauzet, SIAM 2011).

$$\mathcal{M}_{opt}^{i} = m_i \; |H_u^i| \quad \forall \; i \; \in \; \Omega.$$

The global optimization determines m_i :

$$\mathcal{M}_{L^\infty} = N^{\frac{2}{3}} \left(\int det(|H_u|) \right)^{-\frac{2}{3}} |H_u| \tag{16}$$

$$\mathcal{M}_{L^1} = N^{\frac{2}{3}} \left(\int det(|H_u|)^{\frac{2}{5}} \right)^{-\frac{2}{3}} det(|H_u|)^{-\frac{1}{5}} |H_u|. \tag{17}$$

To synthetize, the continuous mesh/metric method yields the mesh adaptation solution under the form of a continuous KKT system involving the continuous initial PDE, its continuous adjoint, and a stationaly condition explicitly solved by (16,17). In practice, the KKT system is discretized and then solved.

The rest of the paper examines the approach closer to [15, 16], refered in this paper as a tensorial method, which consists in the direct building of a discrete KKT system.

4. Approximation of metric properties

4.1. Generic mesh notations

Given a mesh $\mathcal{H}_{\mathbf{x}}$, we can define the following partitions:

- a *mesh-vertex* is a vertex of numero i and coordinates $\mathbf{x}_i$ of an element of the mesh.

- when there is an *edge* between vertex i and vertex j, we denote $\mathbf{x}_{ij} = \mathbf{x}_j - \mathbf{x}_i$.

- two tetrahedra m and n having a common face have face mn or face nm as common face.

- *elements* : triangles (i,j,k) or tetrahedra (i,j,k,l). Elements are divided in *sub-elements*: 6 *subtriangles* using medians and 24 *subtetrahedra* using median plans. The vertices of a subtetrahedron are : a mesh-vertex i , a center I_{ij} of an edge ij having i as extremity, the centroid g_{ijk} of a face ijk containing i and j, the element centroid G_{ijkl}. The measure of a subtetrahedron of the tetrahedron T is $1/24$ $meas(T)$.

- *cell* i : for a vertex i of the mesh, cell i is union of sub-elements having i as vertex of the sub-element. A cell measure is defined as

$$meas_{\mathbf{x}}(i) = \frac{1}{dim+1} \sum_{T_{\mathbf{x}} \ni i} \quad meas(T_{\mathbf{x}})$$

where $T_{\mathbf{x}}$ are elements of $\mathcal{H}_{\mathbf{x}}$ containing i.

- *2D-diamond* D_{ij} : union of the 4 subtriangles (of triangles ijk and ijl) having a side beared by edge ij.

- *face-diamond* $\bar{D}_{mn}$, where m and n are two tetrahedra having a common face ijk : union of 6 subtetrahedra having a subtriangle of the common face ijk as face.

- *edge-diamond* D_{ij}: union of subtetrahedra having having a side beared by edge ij.

The integral of a function e_{ij} defined on the edges can be approximated by:

$$err_{L^1} = \sum_i meas_{\mathbf{x}}(i) \Gamma(i)^{-1} \sum_j e_{ij}$$

or introducing the diamond partition $\Omega = \cup \bar{D}_{mn}$ where m and n are elements with a common face:

$$err_{L^1} = \frac{1}{3} \sum_{\bar{D}_{mn}} meas_{\mathbf{x}}(\bar{D}_{mn}) \ (e_{ij} + e_{ik} + e_{jk}).$$

where i,j,k are vertices of the face mn.

4.2. Discretizing an arbitrary continuous metric on a background mesh

In order to find the optimal metric we are given a background mesh $\mathbf{x}$. We assume that the unknown metric $\mathcal{M}$ is defined on the vertices $\mathcal{M}(\mathbf{x}_i) = \mathcal{M}^i$ of the background mesh and that it is P^1-continuously interpolated. The total number of nodes can be approximated on the mesh $\mathbf{x}$ by a quadrature of (8) as follows:

$$\mathcal{C}(\mathcal{M}) = \sum_i meas_{\mathbf{x}}(i) \sqrt{det(\mathcal{M}^i)}$$

To simplify, *we assume that the unit mesh is a deformation of* $\mathbf{x}$*, and that* $\mathbf{x}_{ij}^{\mathcal{M}}$ *and* $\mathbf{x}_{ij}$ *are colinear.* Then we can derive from the unit-mesh property a relation between the edge lengths of unknown mesh and the edge lengths of the background mesh:

$$(\mathbf{x}_{ij}^{\mathcal{M}}, \mathcal{M}\mathbf{x}_{ij}^{\mathcal{M}}) = 1 = (\mathbf{x}_{ij} \frac{|\mathbf{x}_{ij}^{\mathcal{M}}|}{|\mathbf{x}_{ij}|}, \mathcal{M}\mathbf{x}_{ij} \frac{|\mathbf{x}_{ij}^{\mathcal{M}}|}{|\mathbf{x}_{ij}|}) = (\mathbf{x}_{ij}, \mathcal{M}\mathbf{x}_{ij}) \frac{|\mathbf{x}_{ij}^{\mathcal{M}}|^2}{|\mathbf{x}_{ij}|^2}$$

$$\Rightarrow \mathbf{x}_{ij}^{\mathcal{M}} \approx \mathbf{x}_{ij} (\mathbf{x}_{ij}, \mathcal{M}\mathbf{x}_{ij})^{-\frac{1}{2}}.$$

In order now to evaluate the approximation error provoked by the application of the unit-mesh, we need to define an error model.

5. Second-order error of a metric on a background mesh

Let us define a generic family of error with values on mesh edges. We restrict to second-order or quadratic errors, on the model of P_1-interpolation error.

Definition : The directional second-order or quadratic error produced by the use of the unit mesh $\mathbf{x}_{\mathcal{M}}$ of metric $\mathcal{M}$ has an intensity defined on edge $\mathbf{x}_{ij}^{\mathcal{M}}$ by:

$$e_{ij}^{\mathcal{M}} = \bar{e}_{ij}\ |\mathbf{x}_{ij}^{\mathcal{M}}|^2.$$

in which $\bar{e}_{ij}$ depends only on location and direction of $\mathbf{x}_{ij}^{\mathcal{M}}$, typically:

$$e_{ij}^{\mathcal{M}} = \quad |\mathbf{x}_{ij}^{\mathcal{M}}|^2 \quad \bar{e}_{ij}(\tfrac{1}{2}(\mathbf{x}_i^{\mathcal{M}} + \mathbf{x}_j^{\mathcal{M}}), \tfrac{\mathbf{x}_{ij}^{\mathcal{M}}}{|\mathbf{x}_{ij}^{\mathcal{M}}|}).$$

Since we *a priori* do not know the optimal metric and mesh, it is useful to evaluate this error on a given background mesh $\mathbf{x}$. We use that the unit mesh is a deformation of $\mathbf{x}$ in such a way that $\mathbf{x}_{ij}^{\mathcal{M}}$ and $\mathbf{x}_{ij}$ are colinear. Then the intensity $e_{ij}^{\mathcal{M}}$ of the error with the unit mesh evaluated at middle of $\mathbf{x}_{ij}$ of the initial mesh writes:

$$e_{ij}^{\mathcal{M}} = \quad |\mathbf{x}_{ij}|^2 \quad (\mathbf{x}_{ij}, \mathcal{M}_{ij}\mathbf{x}_{ij})^{-1} \quad \bar{e}_{ij}(\frac{1}{2}(\mathbf{x}_i + \mathbf{x}_j), \frac{\mathbf{x}_{ij}}{|\mathbf{x}_{ij}|}) \tag{18}$$

where $\mathcal{M}_{ij}$ is evaluated on $\frac{1}{2}(\mathbf{x}_i + \mathbf{x}_j)$. The mesh adaptation problem will be set as the research of the discrete metric, defined on mesh verices and linearly interopolated, of a given number of nodes N

$$\mathcal{C}(\mathcal{M}) = N$$

and minimizing the discrete error norm:

$$j(\mathcal{M}) = \sum_i meas_{\mathbf{x}}(i) \frac{1}{\Gamma(i)} \sum_{ij \ni i}\ e_{ij}^{\mathcal{M}}. \tag{19}$$

In Section 6 we determine the optimal mesh for this type of error, as far as $\bar{e}_{ij}$ is identified. The rest of the present section is devoted to the description of three examples of quadratic errors.

5.1. First example: interpolation error

The above generic model of quadratic error applies to the P_1-interpolation error. Indeed, the weighted P_1-interpolation error of a quadratic function u on $\mathbf{x}_{ij}^{\mathcal{M}}$ can be discretized similarly to (9),(10) as follows:

$$\int |g||u - \Pi_h u| \mathrm{d}\Omega \preceq\ \tfrac{1}{8}\ \textstyle\sum_i\ meas_{\mathbf{x}}(i)\Gamma(i)^{-1}\ \sum_j\ e_{ij}^{\mathcal{M},g,u}(\mathbf{x}_{ij})$$

$$e_{ij}^{\mathcal{M},g,u} = \quad |\mathbf{x}_{ij}^{\mathcal{M}}|^2 \quad |g_{ij}|\ |H_{ij}| \cdot \tfrac{\mathbf{x}_{ij}^{\mathcal{M}}}{|\mathbf{x}_{ij}^{\mathcal{M}}|} \cdot \tfrac{\mathbf{x}_{ij}^{\mathcal{M}}}{|\mathbf{x}_{ij}^{\mathcal{M}}|}\ .$$

where $H_{ij} = H(\frac{1}{2}(\mathbf{x}_i^{\mathcal{M}} + \mathbf{x}_j^{\mathcal{M}}))$, $H(\mathbf{x})$ being the Hessian of u at point $\mathbf{x}$, and $g_{ij} = g(\frac{1}{2}(\mathbf{x}_i^{\mathcal{M}} + \mathbf{x}_j^{\mathcal{M}}))$. Here $\preceq$ holds for an inequality applying for a sufficiently fine mesh, with a multiplicative constant close to 1. It can be evaluated on a background mesh as follows:

$$e_{ij}^{\mathcal{M},g,u}(\mathbf{x}_{ij}) = \quad |\mathbf{x}_{ij}^{\mathcal{M}}|^2 \quad \bar{e_{ij}}(\mathbf{x}_{ij}) = \ (\mathbf{x}_{ij}, \mathcal{M}\mathbf{x}_{ij})^{-1}\ \ |\mathbf{x}_{ij}|^2 \quad \bar{e_{ij}}(\mathbf{x}_{ij})$$

with:

$$\bar{e_{ij}}(\mathbf{x}_{ij}) = |g_{ij}(\mathbf{x}_{ij})|\ |H_{ij}(\mathbf{x}_{ij})| \cdot \tfrac{\mathbf{x}_{ij}^{\mathcal{M}}}{|\mathbf{x}_{ij}^{\mathcal{M}}|} \cdot \tfrac{\mathbf{x}_{ij}^{\mathcal{M}}}{|\mathbf{x}_{ij}^{\mathcal{M}}|} \quad = \quad |g_{ij}(\mathbf{x}_{ij})|\ |H_{ij}(\mathbf{x}_{ij})| \cdot \tfrac{\mathbf{x}_{ij}}{|\mathbf{x}_{ij}|} \cdot \tfrac{\mathbf{x}_{ij}}{|\mathbf{x}_{ij}|}.$$

Then this first example takes place into the context of (18)(19).

5.2. Goal-oriented error

Quadratic errors can also be encountered in the case of a goal-oriented error analysis. Let u be the solution of (1) and $u_{\mathcal{M}}$ the discrete solution of (2) when the mesh is an unit mesh for metric $\mathcal{M}$. A typical 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 write as follows:

$$\delta j_{goal}(\mathcal{M}) = |(g, u - u_{\mathcal{M}})| = |(g, \Pi_{\mathcal{M}} u - u_{\mathcal{M}} + u - \Pi_{\mathcal{M}} u)|. \tag{20}$$

According to the Aubin-Nitsche analysis, this error is second-order with respect to mesh size. Let us define the discrete adjoint state u^*_{goal}:

$$\forall \psi_{\mathcal{M}} \in \ V_{\mathcal{M}}, \quad a(\psi_{\mathcal{M}}, u^*_{goal}) = (\psi_{\mathcal{M}}, g). \tag{21}$$

In the sequel, we use a fixed-point in which the adjoint is frozen with respect to the metric $\mathcal{M}$. Injecting (21) in (20) we get:

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

and, using (4),

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

thus

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

Recall that u is unknown. The second and third terms, similar to the main term of the Hessian-based adaptation in previous section can be explicitly approached in the same way.

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

The second and third terms give Hessian-like quadratic errors $e_{ij}^{\mathcal{M},u^*_{goal},f}$ and $e_{ij}^{\mathcal{M},g,u}$:

$$\begin{aligned} &|(f - \Pi_{\mathcal{M}} f, u^*_{goal})| + |g||\pi_{\mathcal{M}} u_{\mathcal{M}} - u_{\mathcal{M}}| \\ &\qquad \preceq \ \sum_i \ meas_{\mathbf{x}}(i)\Gamma(i)^{-1} \sum_{ij \ni i} \left(e_{ij}^{\mathcal{M},u^*_{goal},f} + e_{ij}^{\mathcal{M},g,u} \right) \\ &\qquad \preceq \ \sum_i \ meas_{\mathbf{x}}(i)\Gamma(i)^{-1} \sum_{ij \ni i} \ (\mathbf{x}_{ij}, \mathcal{M}\mathbf{x}_{ij})^{-1} \quad |\mathbf{x}_{ij}|^2 \left(\bar{e_{ij}}^{u^*_{goal},f} + \bar{e_{ij}}^{g,u} \right) \end{aligned}$$

with

$$\bar{e_{ij}}^{u^*_{goal},f}(\mathbf{x}_{ij}) = |u^*_{goal,ij}| \ |H^f_{ij}| \cdot \frac{\mathbf{x}_{ij}}{|\mathbf{x}_{ij}|} \cdot \frac{\mathbf{x}_{ij}}{|\mathbf{x}_{ij}|}$$

$$\bar{e_{ij}}^{g,u}(\mathbf{x}_{ij}) = |g_{ij}| \ |H^u_{ij}| \cdot \frac{\mathbf{x}_{ij}}{|\mathbf{x}_{ij}|} \cdot \frac{\mathbf{x}_{ij}}{|\mathbf{x}_{ij}|}$$

and

$$u^*_{goal,ij} = u^*_{goal}(\frac{\mathbf{x}_i + \mathbf{x}_j}{2})$$

$$g_{ij} = g(\frac{\mathbf{x}_i + \mathbf{x}_j}{2})$$

$$H^f_{ij} = H^f(\frac{\mathbf{x}_i + \mathbf{x}_j}{2})$$
$$H^u_{ij} = H^u(\frac{\mathbf{x}_i + \mathbf{x}_j}{2})$$

The first term is more complex. It can be estimated in a different way from the continuous method presented in [8] and used in [12].

$$|a(\Pi_{\mathcal{M}}u-u, u^*_{goal})| = |\int_\Omega \nabla(\Pi_{\mathcal{M}}u-u)\nabla\Pi_{\mathcal{M}}u^*_{g,\mathcal{M}}\mathrm{d}\mathbf{x}| \preceq \sum_{\partial T_{mn}} |\nabla u^*_{goal}|_{T_m} - \nabla u^*_{goal}|_{T_n}|\cdot\mathbf{n}_{mn} \int_{\partial T_{mn}} |\Pi_{\mathcal{M}}u-u|\,\mathrm{d}\sigma.$$

In the 3D case, the intersection ∂T_{mn} of two elements T_m and T_n is a common face with vertices i, j, k and an area $area(mn)$. The following quantity is known:

$$\kappa_{mn}(u^*_{goal}) = |\left(\nabla u^*_{goal}\right)|_{T_m}\cdot\mathbf{n}_{mn} - \left(\nabla u^*_{goal}\right)|_{T_n}\cdot\mathbf{n}_{mn}|.$$

The remaining expression can be expressed in terms of interpolation errors:

$$\int_{\partial T_{mn}} |\Pi_{\mathcal{M}}u - u| \approx \frac{1}{3} area(mn)(e^{\mathcal{M},u}_{ij} + e^{\mathcal{M},u}_{ik} + e^{\mathcal{M},u}_{kj})$$

with (for $\alpha\beta$=ij,ik and kj):

$$e^{\mathcal{M},u}_{\alpha\beta} = (\mathbf{x}_{\alpha\beta}, \mathcal{M}\mathbf{x}_{\alpha\beta})^{-1} \quad |\mathbf{x}_{\alpha\beta}|^2\ \bar{e}^u_{\alpha\beta}$$

$$\bar{e^{-}_{\alpha\beta}}^{u}(\mathbf{x}_{\alpha\beta}) = |g_{\alpha\beta}|\ |H^u_{\alpha\beta}|\cdot\frac{\mathbf{x}_{\alpha\beta}}{|\mathbf{x}_{\alpha\beta}|}\cdot\frac{\mathbf{x}_{\alpha\beta}}{|\mathbf{x}_{\alpha\beta}|}.$$

We get:

$$|a(\Pi_{\mathcal{M}}u - u, u^*_{goal})| \preceq \sum_{\bar{D}_{mn}} |\bar{D}_{mn}|\frac{area(mn)}{|\bar{D}_{mn}|}\frac{1}{3}(e^{\mathcal{M},u}_{ij} + e^{\mathcal{M},u}_{ik} + e^{\mathcal{M},u}_{jk})\ \kappa_{mn}(u^*_{goal})$$

Let us convert the RHS into an edge-by-edge sum:

$$|a(\Pi_{\mathcal{M}}u - u, u^*_{goal})| \preceq \sum_{\bar{D}_{mn}} \sum_{\alpha\beta=ij,ik,jk} area(mn)\ \frac{1}{3}e^{\mathcal{M}}_{\alpha\beta}\kappa_{mn}(u^*_{goal})$$

$$= \sum_{\text{edges } ij} \sum_{\bar{D}_{mn}\ni ij} area(mn)\ \frac{1}{3}e^{\mathcal{M}}_{ij}\kappa_{mn}(u^*_{goal}) = \sum_{\text{edges } ij} e^{\mathcal{M},a}_{ij}\ |D_{ij}|$$

where we recognize the edge-by-edge integral of a field $e^{\mathcal{M},a}_{ij}$ defined on edges, with the notation:

$$e^{\mathcal{M},a}_{ij} = \frac{1}{|D_{ij}|} \sum_{\bar{D}_{mn}\ni ij} area(mn)\ \frac{1}{3}e^{\mathcal{M}}_{ij}\kappa_{mn}(u^*_{goal}). \tag{22}$$

Equivalently (at the second order) we get the(18)(19) format:

$$|a(\Pi_{\mathcal{M}}u - u, u^*_{goal})| \preceq \sum_i meas_{\mathbf{x}}(i)\frac{1}{\Gamma(i)}\sum_{ij\ni i} e^{\mathcal{M},a}_{ij}.$$

Gathering the analyses of the three terms, introducing:

$$\bar{e_{ij}}^{\mathcal{M},a} = \ (\mathbf{x}_{ij}, \mathcal{M}\mathbf{x}_{ij}) \quad |\mathbf{x}_{ij}|^{-2} \ e_{ij}^{\mathcal{M},a}$$

we get:

$$\delta j_{goal}(\mathcal{M}) \ \preceq \ \sum_i \ meas_{\mathbf{x}}(i)\Gamma(i)^{-1} \ \sum_{ij \ni i} \ (\mathbf{x}_{ij}, \mathcal{M}\mathbf{x}_{ij})^{-1} \quad |\mathbf{x}_{ij}|^2 \left(\bar{e_{ij}}^{\mathcal{M},a} + \bar{e}_{ij}^{u^*_{goal},f} + \bar{e}_{ij}^{g,u} \right)$$

which takes place in the context of (18)(19).

5.3. Norm-oriented error

The norm-oriented analysis is defined in details in [1]. In short, this method focusses on the minimization of the following norm with respect to the mesh $\mathcal{M}$:

$$\delta j(\mathcal{M}) = ||u - u_{\mathcal{M}}||^2_{L^2(\Omega)}. \tag{23}$$

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

$$\delta j(\mathcal{M}) = (g_{\mathcal{M}}, u - u_{\mathcal{M}}). \tag{24}$$

But in the practical application $u - u_{\mathcal{M}}$ is not known. We approximate it by a function close to it, which we call a corrector. An example is the field $g_{\mathcal{M}} = \bar{u}'_{prio,\mathcal{M}} - (\pi_{\mathcal{M}} u_{\mathcal{M}} - u_{\mathcal{M}})$ in which $\pi_{\mathcal{M}} u_{\mathcal{M}} - u_{\mathcal{M}}$ is a Hessian-based approximation of the interpolation error and in which $\bar{u}'_{prio,\mathcal{M}}$ is the solution of:

$$a(\bar{u}'_{prio,\mathcal{M}}, \phi) = \sum_{\partial T_{ij}} \ (\nabla\phi|_{T_i} - \nabla\phi|_{T_j}) \cdot \mathbf{n}_{ij} \quad \int_{\partial T_{ij}} (\pi_{\mathcal{M}} u_{\mathcal{M}} - u_{\mathcal{M}}) \ \mathrm{d}\sigma \ - \ (\phi, \pi_{\mathcal{M}} f_{\mathcal{M}} - f_{\mathcal{M}}). \tag{25}$$

Another example with a RHS evaluated on a two-times finer grid is given in [1].

Let us define the discrete adjoint state u^*_{norm}:

$$\forall \psi_{\mathcal{M}} \in \ V_{\mathcal{M}}, \quad a(\psi_{\mathcal{M}}, u^*_{norm}) = (\psi_{\mathcal{M}}, g_{\mathcal{M}}). \tag{26}$$

Then, similarly to previous section we shall minimize:

$$\delta j_{norm}(\mathcal{M}) \approx |a(\Pi_{\mathcal{M}} u - u, u^*_{norm}) + (f - \Pi_{\mathcal{M}} f, u^*_{norm}) + (g_{\mathcal{M}}, u - \Pi_{\mathcal{M}} u)|$$

and, more precisely, minimize:

$$\mathcal{E}(\mathcal{M}) \ = \ \sum_i \ meas_{\mathbf{x}}(i)\Gamma(i)^{-1} \ \sum_{ij \ni i} \ (\mathbf{x}_{ij}, \mathcal{M}\mathbf{x}_{ij})^{-1} \quad |\mathbf{x}_{ij}|^2 \left(\bar{e_{ij}}^{\mathcal{M},a} + \bar{e}_{ij}^{u^*_{norm},f} + \bar{e}_{ij}^{g,u} \right)$$

with

$$\bar{e_{ij}}^{u^*_{norm},f} = |u^*_{norm,ij}| \ |H^f_{ij}| \cdot \frac{\mathbf{x}_{ij}}{|\mathbf{x}_{ij}|} \cdot \frac{\mathbf{x}_{ij}}{|\mathbf{x}_{ij}|}$$

$$\bar{e_{ij}}^{g,u} = |g_{ij}| \ |H^u_{ij}| \cdot \frac{\mathbf{x}_{ij}}{|\mathbf{x}_{ij}|} \cdot \frac{\mathbf{x}_{ij}}{|\mathbf{x}_{ij}|}$$

$$\bar{e_{ij}}^{\mathcal{M},a} = \ (\mathbf{x}_{ij}, \mathcal{M}\mathbf{x}_{ij}) \quad |\mathbf{x}_{ij}|^{-2} \ \frac{1}{|D_{ij}|} \ \sum_{\bar{D}_{mn} \ni ij} area(mn) \ \frac{1}{3} e_{ij}^{\mathcal{M}} \kappa_{mn}(u^*_{norm}) \tag{27}$$

and with $\kappa_{mn}(u^*_{norm}) = |\left(\nabla u^*_{norm}\right)|_{T_m} \cdot \mathbf{n}_{mn} - \left(\nabla u^*_{norm}\right)|_{T_n} \cdot \mathbf{n}_{mn}|$. This again takes place in the context of (18)(19).

6. Optimal metric

The purpose is to minimize with respect to the metric for a given number of vertices N a functional of the form:

$$\mathcal{E}(\mathcal{M}) \;=\; \sum_i meas_{\mathbf{x}}(i)\Gamma(i)^{-1}\sum_{\mathbf{x}_{ij}} \;(\mathbf{x}_{ij})^2(\mathbf{x}_{ij},\mathcal{M}\mathbf{x}_{ij})^{-1}\;\bar{e}_{ij}.$$

We solve this in two steps as in [20]: first we minimize the functional in a point of the computational domain and get a first property of the solution, second we finish determining the optimum by solving a sub-problem on the whole domain.

6.1. Pointwise optimal metric

The pointwise minimum problem can be set with a fixed number of vertices by considering a metric $\mathcal{M}$ with a product of eigenvalues equal to unity, i.e. $\lambda_1\lambda_2\lambda_2 = (h_1h_2h_3)^{-2} = 1$. This is equivalent to say that the metric is searched with an unknown scalar multiplicative constant. In the continuous metric formulation, according for example to [21, 22], an optimal error at a point is obtained when the directional error is uniform w.r.t. direction. In the discrete context, according to [15, 16], the error is uniform on the different edges $ij, ik, il, ...$ around a vertex i.:

$$e_{ij}^{\mathcal{M}} = e_{ik}^{\mathcal{M}} = ... = C_i \;\;\Leftrightarrow\;\; (\mathbf{x}_{ij})^2\;(\mathbf{x}_{ij},\mathcal{M}\mathbf{x}_{ij})^{-1}\;\;\bar{e}_{ij} = C_i\;\;\forall j \;\;\Leftrightarrow\;\; (\mathbf{x}_{ij})^{-2}\;(\mathbf{x}_{ij},\mathcal{M}\mathbf{x}_{ij})\;\;\bar{e}_{ij}^{-1} = C_i^{-1}\;\;\forall j.$$

Summing around a vertex gives:

$$\sum_{j\in\Gamma(i)} (\mathbf{x}_{ij})^{-2}\;\bar{e}_{ij}^{-1}\;(\mathbf{x}_{ij},\mathcal{M}\mathbf{x}_{ij}) = |\Gamma(i)|C_i^{-1}.$$

Then:

$$|\Gamma(i)|C_i^{-1} = \sum_{j\in\Gamma(i)} (\mathcal{M}\;\bar{e}_{ij}^{-\frac{1}{2}}|\mathbf{x}_{ij}|\mathbf{x}_{ij}, \bar{e}_{ij}^{-\frac{1}{2}}|\mathbf{x}_{ij}|\mathbf{x}_{ij}) = \mathcal{M}\;\;:\;\;\sum_{j\in\Gamma(i)} \bar{e}_{ij}^{-\frac{1}{2}}|\mathbf{x}_{ij}|\mathbf{x}_{ij}\otimes\bar{e}_{ij}^{-\frac{1}{2}}|\mathbf{x}_{ij}|\mathbf{x}_{ij}$$

now, remembering that $A : B \;=\; tr({}^tA.B)$, it is interesting to choose (among the possible solutions):

$$\mathcal{M}^i \;=\; \frac{|\Gamma(i)|C_i^{-1}}{dim}\left(\sum_{j\in\Gamma(i)} \bar{e}_{ij}^{-1}|\mathbf{x}_{ij}|^{-2}\mathbf{x}_{ij}\;\otimes\;\mathbf{x}_{ij}\right)^{-1}.$$

It remains to evaluate the optimal field C on the mesh vertices.

6.2. Global optimal metric

Let:

$$\mathcal{M}_1^i \;=\; \frac{|\Gamma(i)}{dim}\left(\sum_{j\in\Gamma(i)} \bar{e}_{ij}^{-1}|\mathbf{x}_{ij}|^{-2}\mathbf{x}_{ij}\;\otimes\;\mathbf{x}_{ij}\right)^{-1},$$

we look for a C_i which minimizes:

$$err_{L^1} = \sum_i meas_{\mathbf{x}}(i)\Gamma(i)^{-1}\sum_{\mathbf{x}_{ij}} \;(\mathbf{x}_{ij})^2(\mathbf{x}_{ij},C_i^{-1}\mathcal{M}_1^i\mathbf{x}_{ij})^{-1}\;\bar{e}_{ij}$$

or:

$$err_{L^1} = \sum_i\;\alpha_i\;C_i\;\;;\;\;\text{with}\;\;\alpha_i = meas_{\mathbf{x}}(i)\Gamma(i)^{-1}\sum_{\mathbf{x}_{ij}} \;(\mathbf{x}_{ij})^2(\mathbf{x}_{ij},\mathcal{M}_1^i\mathbf{x}_{ij})^{-1}\;\bar{e}_{ij}$$

to minimize under the constraint: $\sum_i meas_{\mathbf{x}}(i)\sqrt{det(C_i^{-1}\mathcal{M}_1^i)} = N$ or:

$$\sum_i \mu_i \, C_i^{-\frac{dim}{2}} = N \quad \text{with} \quad \mu_i \; = \; meas_{\mathbf{x}}(i)\sqrt{det(\mathcal{M}_1^i)}$$

This problem is easy to solve if we consider the following variable change $d_i = \mu_i C_i^{-\beta}$, with $\beta = dim/2$ giving:

$$\text{Min} \quad \sum_i \eta_i d_i^{-\frac{1}{\beta}} \quad \text{under the constraint} \quad \sum_i d_i = N, \tag{28}$$

with $\eta_i = \alpha_i \mu_i^{\frac{1}{\beta}}$. The solution of (28) writes:

$$d_i = \left(\sum_j \eta_j^{\frac{\beta+1}{\beta}} \right)^{-1} \eta_i^{\frac{\beta+1}{\beta}} \; N.$$

Lemma: *The optimal metric is defined by:*

$$\mathcal{M}^i \; = \; C_i^{-1} \;\; \mathcal{M}_1^i$$

with

$$\mathcal{M}_1^i \; = \; \frac{|\Gamma(i)}{dim} \left(\sum_{j \in \Gamma(i)} \bar{e}_{ij}^{-1} |\mathbf{x}_{ij}|^{-2} \mathbf{x}_{ij} \; \otimes \; \mathbf{x}_{ij} \right)^{-1},$$

$$C_i^{-1} = \mu_i^{-\frac{2}{dim}} \left(\sum_j \eta_j^{\frac{dim+2}{dim}} \right)^{-\frac{2}{dim}} \eta_i^{\frac{2dim+4}{dim^2}} \; N^{\frac{2}{dim}},$$

$$\alpha_i = \frac{meas_{\mathbf{x}}(i)}{\Gamma(i)^{-1}} \sum_{\mathbf{x}_{ij}} \frac{(\mathbf{x}_{ij})^2}{(\mathbf{x}_{ij}, \mathcal{M}_1^i \mathbf{x}_{ij})^{-1}} \, \bar{e}_{ij} \quad ; \quad \mu_i \; = \; meas_{\mathbf{x}}(i)\sqrt{det(\mathcal{M}_1^i)} \quad ; \quad \eta_i = \alpha_i \mu_i^{\frac{dim}{2}}. \square$$

7. Numerical examples : EN COURS

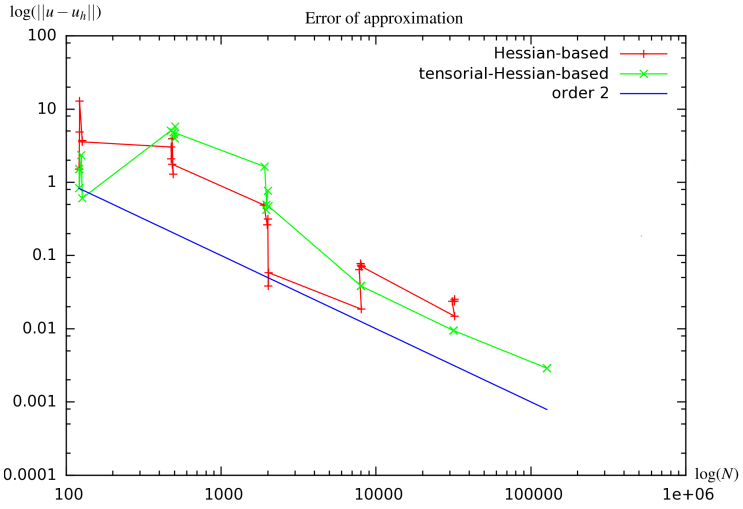

Figure 1: hess-tenso

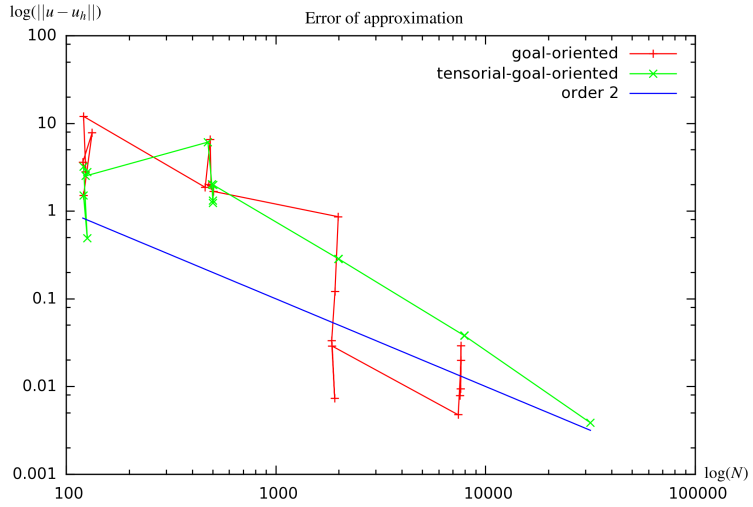

Figure 2: goal-tenso

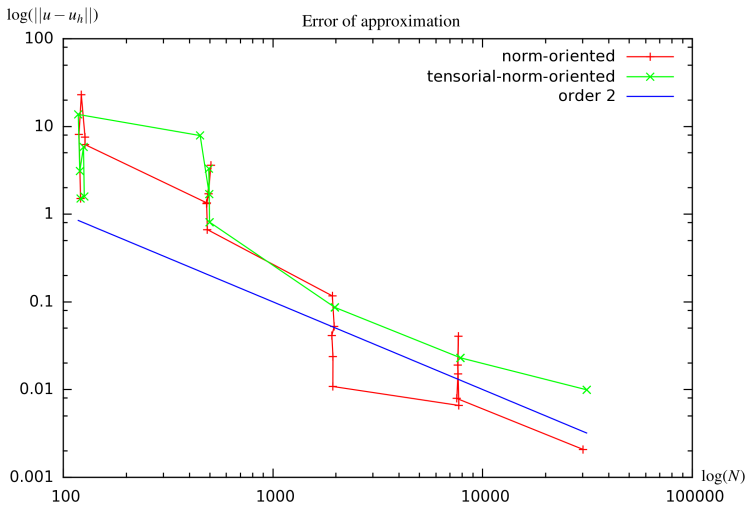

Figure 3: norm-tenso

8. Conclusion

We have proposed an adaptation of the dicrete tensorial approach for metric-based mesh adaptation and compared it to the continuous metric method for an elliptic model. The tensorial option assumes that the iterated mesh is locally of same edge directions as the background mesh. The main theoretical difference is in the treatment of goal and norm oriented error analysis. The error analysis is simpler and does not require an anisotropy bound as in the continuous approach.

9. Acknowledgements

10. Annexe

Lemmas Lemma 1: Given $L_i \geq 0$, if

$$\sum L_i \delta\mu_i = 0 \quad \forall \delta\mu_i \quad s.t. \quad \sum \delta\mu_i = 0$$

Then $L_i = L$ (indep of i).
Proof: if $L_k > L_m$ then $L_k.(-1) + L_m.(+1)$ is not zero.

Lemma 2: $\mathcal{M}$ matrix, $\mathbf{x}$ vecteur:

$$(\mathcal{M}\ \mathbf{x}, \mathbf{x}) = \mathcal{M}\ :\ (\mathbf{x} \otimes \mathbf{x})$$

Lemma 3: assuming $\mathcal{M}$ is symmetric

$$\mathcal{M} = \frac{1}{dim}\,(\mathbf{x} \otimes \mathbf{x})^{-1} \ \Rightarrow\ \mathcal{M}\ :\ (\mathbf{x} \otimes \mathbf{x}) = 1.$$

Proof: $A : B = tr({}^t A.B)$.

References

[1] PhD thesis.

[2] P.-A. Absil, R. Mahony, and R. Sepulchre. *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, Princeton, NJ, 2008.

[3] A. Agouzal, K. Lipnikov, and Y. Vasilevskii. Adaptive generation of quasi-optimal tetrahedral meshes. *East-West Journal*, 7(4):223–244, 1999.

[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 and A. Loseille. High order sonic boom modeling by adaptive methods. *J. Comp. Phys.*, 229:561–593, 2010.

[6] V. Arsigny, P. Fillard, X. Pennec, and N. Ayache. Log-Euclidean metrics for fast and simple calculus on diffusion tensors. *Magn. Reson. Med.*, 56(2):411–421, 2006.

[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 and A. Dervieux. Anisotropic Norm-Oriented mesh adaptation for a Poisson problem. *to appear*, 2015. pre-print: http://www-sop.inria.fr/members/Gautier.Brethes/article-NO.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] T. Coupez. Metric construction by length distribution tensor and edge based error for anisotropic adaptive meshing. *J. Comp. Phys.*, 230:2391–2405, 2011.

[16] T. Coupez, G. Jannoun, N. Nassif, H.C. Nguyen, H. Digonnet, and E. Hachem. Adaptive time-step with anisotropic meshing for incompressible flows. *J. Comp. Phys.*, 241:195–211, 2013.

[17] 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.

[18] C. Gruau and T. Coupez. 3D tetrahedral, unstructured and anisotropic mesh generation with adaptation to natural and multidomain metric. *Comput. Methods Appl. Mech. Engrg.*, 194(48-49):4951–4976, 2005.

[19] W. Huang. Metric tensors for anisotropic mesh generation. *J. Comp. Phys.*, 204(2):633–665, 2005.

[20] 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).

[21] 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.

[22] A. Loseille and F. Alauzet. Continuous mesh framework. Part II: validations and applications. *SIAM Journal on Numerical Analysis*, 49(1):61–86, 2011.

[23] 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.

[24] Y.V. Vasilevski and K.N. Lipnikov. An adaptive algorithm for quasi-optimal mesh generation. *Comput. Math. Math. Phys.*, 39(9):1468–1486, 1999.

[25] 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.

[26] D.A. Venditti and D.L. Darmofal. Anisotropic grid adaptation for functional outputs: application to two-dimensional viscous flows. *J. Comp. Phys.*, 187(1):22–46, 2003.

[27] R. Verfürth. *A Posteriori Error Estimation Techniques for Finite Element Methods.* Oxford University Press, Oxford, 2013.

[28] M. Yano and D. Darmofal. An optimization framework for anisotropic simplex mesh adaptation: application to aerodynamics flows. *AIAA Paper*, 2012-0079, 2012.