Continuous Mesh Model and Well-Posed Continuous Interpolation Error Estimation

Adrien Loseille, Frédéric Alauzet

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Continuous Mesh Model and Well-Posed Continuous Interpolation Error Estimation

Adrien Loseille — Frédéric Alauzet

N° 6846

March 23, 2009

Thème NUM

Rapport de recherche

ISSN 0249-6399 ISRN INRIA/RR--6846--FR+ENG

Continuous Mesh Model and Well-Posed Continuous Interpolation Error Estimation

Adrien Loseille* , Frédéric Alauzet†

Thème NUM — Systèmes numériques
Projet Gamma

Rapport de recherche n° 6846 — March 23, 2009 — 51 pages

Abstract: In the context of mesh adaptation, Riemannian metric spaces have been used to prescribe orientation, density and stretching of anisotropic meshes. Such structures are used to compute lengths in adaptive mesh generators. In this report, a Riemannian metric space is shown to be more than a way to compute a distance. It is proven to be a reliable continuous mesh model. In particular, we demonstrate that the linear interpolation error can be derived continuously for a continuous mesh.

In its tangent space, a Riemannian metric space reduces to a constant metric tensor so that it simply spans a metric space. Metric tensors are then used to continuously model discrete elements. On this basis, geometric invariants have been extracted. They connect a metric tensor to the set of all the discrete elements which can be represented by this metric. As the behavior of a Riemannian metric space is obtained by patching together the behavior of each of its tangent spaces, the global mesh model arises from gathering together continuous element models. We complete the continuous-discrete analogy by providing a continuous interpolation error estimate and a well-posed definition of the continuous linear interpolate. The later is based on an exact relation connecting the discrete error to the continuous one.

From one hand, this new continuous framework freed the analysis of the topological mesh constraints. On the other hand, powerful mathematical tools are available and well defined on the space of continuous meshes: calculus of variations, differentiation, optimization, ..., whereas these tools are not defined on the space of discrete meshes.

Key-words: Unstructured mesh, continuous mesh, Riemannian metric space, interpolation error, linear interpolate.

* Email : Adrien.Loseille@inria.fr
† Email : Frederic.Alauzet@inria.fr

Unité de recherche INRIA Rocquencourt
Domaine de Voluceau, Rocquencourt, BP 105, 78153 Le Chesnay Cedex (France)
Téléphone : +33 1 39 63 55 11 — Télécopie : +33 1 39 63 53 30

Modèle continu de maillage et erreur d'interpolation continue bien posée

Résumé : Les espaces métriques riemanniens sont classiquement utilisés en adaptation de maillage dans le but de prescrire l'orientation, les étirements et la densité des maillages anisotropes. Ils définissent alors le calcul des distances dans les mailleurs adaptatifs. Dans ce rapport, on montre, au-delà de la simple définition du calcul des distances, qu'un espace métrique riemannien est un modèle continu de maillage. On montre que ce modèle est bien posé sur le plan théorique. En particulier, on démontre qu'il est possible de dériver de façon continue l'erreur d'interpolation linéaire.

Localement, ces espaces se comportent dans leurs plans tangents comme des espaces métriques euclidiens. On utilise ces derniers pour modéliser les éléments discrets. À partir de cette modélisation, on montre qu'il existe un ensemble d'invariants géométriques qui lient la métrique aux éléments discrets qu'elle représente. Tout comme le comportement global d'un espace riemannien est obtenu en recollant les comportements locaux de ses espaces tangents, un maillage va être modélisé par le recollement des modèles d'éléments continus. Enfin, on complète l'analogie entre la vision continue et la vision discrète en proposant une estimation de l'erreur d'interpolation continue et une définition bien posée de l'opérateur d'interpolation linéaire continu. La définition de cet interpolé repose sur une propriété d'exactitude locale aboutissant à une relation d'équivalence entre l'erreur d'interpolation discrète et l'erreur d'interpolation continue.

D'une part, ce nouveau cadre théorique permet de se libérer des contraintes liées à la topologie des maillages discrets. D'autre part, on dispose naturellement sur l'espace des maillages continus d'outils d'analyse puissants et bien posés qui ne sont pas définis sur l'espace des maillages discrets: calcul des variations, différentiation, optimisation, ...

Mots-clés : Maillage non structuré, maillage continu, espaces métriques riemanniens, erreur d'interpolation, interpolé linéaire.

Introduction

In this article, a continuous mesh concept is introduced using the notion of Riemannian metric space of the differential geometry. The notion of continuous mesh aims at modeling unstructured discrete computational meshes classically used in numerical simulations solved by the finite element, the finite volume, the discontinuous Galerkin,... numerical methods. The final purpose of this model is twice. From a theoretical point of view, it enables us to prove the consistency of classical metric-based mesh adaptation. To this end, it provides exact relations between the linear interpolation error and the mesh prescription. From a practical point of view, the objective is to derive a continuous interpolation error estimate that can be used to automatically perform anisotropic mesh adaptation. The proposed continuous formalism permits a complete abstraction of the notion of mesh. In particular the geometric data (e.g. the vertex coordinates) and the topology description (e.g. the mesh entities) do not exist anymore. As regards interpolation error estimates in the continuous mesh framework, they are free of any *a priori* hypothesis on the mesh such as alignment or density requirements. As a consequence, this report aims at proving that Riemannian metric space is more than a way to define distance computation within anisotropic mesh generators. It is a reliable mesh model that is well suited for metric-based mesh adaptation.

Fundamentals of anisotropic mesh adaptation. When dealing with mesh adaptation, the prescription and the generation of adapted meshes are crucial issues. There exists a large class of methods to prescribe and to generate adapted meshes depending on the problem at hand along with the mesh specificity: uniform, isotropic, anisotropic, ... The simplest algorithms consist in refining or coarsening the current mesh according to patterns. However, such strategies encounter several bottlenecks. In particular, mesh coarsening can only be applied to regions already refined by patterns, *i.e.,* only added patterns can be removed, and they do not allow anisotropic meshes to be generated. Moreover, the quality of the sequence of refined meshes is strongly related to the quality of the initial mesh and this quality can only decrease during the refinement process. A generic and elegant way to generate anisotropic meshes is to use the notion of metric and Riemannian metric space. An adapted anisotropic mesh in this framework is simply the image in the Euclidean space of a uniform mesh in a Riemannian metric space.

To this end, the distance in the adaptive mesh generator is computed in the Riemannian metric space instead of the Euclidean one. This technique generalizes the case of uniform meshes where distances are computed in the classical Euclidean space. Then, the main idea to generate an anisotropic adapted mesh is to complete a unit mesh in the given Riemannian metric space. It consists in generating a mesh where the edges length are equal to one with respect to the prescribed Riemannian metric space. This approach was initiated by Hecht, see developments in [21] and in Vallet's thesis [35]. This method is commonly called metric-based mesh adaptation. There are actually a lot of available codes that are based on the metric concept. Let us mention `Bamg` [20] and `BL2D` [23] in 2D, `Yams` [16] for discrete surface mesh adaptation and `Forge3d` [10], `Fun3d` [22], `Gamic` [18], `MeshAdap` [24],

`Mmg3d` [12], `Mom3d` [34], `Tango` [5] and [29] in 3D. It is worth mentioning that all these codes have arisen from different mesh generation methods. The method in [18, 20] is based on a global constrained Delaunay kernel. In [23], the Delaunay method and the frontal approaches are coupled. [16] is based on local mesh modifications. And, [10] is based on the minimal volume principle. Another benefit is that this mesh prescription approach is naturally derived from anisotropic interpolation error estimate. Indeed, interpolation error involved a Hessian matrix from which a metric is easily derived. If the use of a Hessian matrix to defined a metric tensor is now classical for generating anisotropic meshes, it remains to evaluate the impact of the practical algorithm used to generate the mesh. In other words, is it a relevant choice to use metric fields for the control of the interpolation error ? A lot of numerical examples for real life problems [5, 13, 14, 17, 26, 29, 30, 34] tend to answer affirmatively to this question. In this work, this question is theoretically studied by using a continuous mesh model.

The proposed continuous framework. In this paper, a fully continuous mesh model is introduced based on the notion of Riemannian metric space. The continuous mesh is a function that prescribes at each point a density, anisotropic quotients and orientations. This model is completely generic and geometric. Several geometric relations connect the continuous mesh to the set of its discrete representatives. The main pro of this approach is to be independent of the mesh generation algorithm used to generate adapted meshes.

This model is then used to study the interpolation error. Some mathematical developments lead to a single relation that connects an infinite set of discrete elements to a unique continuous estimate. From this estimate, the interpolation error for a given continuous mesh can be accurately predicted whatever the considered smooth function. Contrary to classical approaches inherited from [7], there are optimal conditions leading to alignment requirements between the mesh and the Hessian function. The derivation of a general interpolation error estimate has already been studied for a single element in the case of a quadratic function [6]. These studies enable the best shaped element for a given norm to be derived. However, it is, from a practical point of view, impossible to generate a mesh only composed of optimal elements. Moreover, the variation of the function needs to be taken into account when the function is no more quadratic. In this work, estimates are derived locally both for the continuous mesh and the functions thanks to Taylor expansion. If the Taylor expansion of a function is common, the Taylor expansion of a continuous mesh consists in working in the tangent space of the Riemannian metric space where the metric tensor is constant. Then both the variations of the mesh and the function are integrated on the whole computational domain. As a consequence, estimates are still very accurate for non quadratic functions and non uniform continuous meshes. This point is demonstrated in the numerical examples. Theoretically, the use of a discrete support is no more mandatory to compute the interpolation error. Several fully continuous and analytical examples are given to exemplify this feature. More practically, only a background mesh that supports a discrete representation of the continuous mesh is needed when dealing with real-life appli-

cations. There is no need to generate a discrete mesh that corresponds to the continuous one.

Overview. In a first section, we recall the differential geometry notions that are recurrently used in the continuous approach. Next, the continuous element and mesh models are introduced. Geometrical invariants, that connect continuous elements and discrete elements, are proved. In a third section, these results are used to exactly predict the linear interpolation error for discrete and continuous elements. Then, the notion of continuous linear interpolation is well established and exactly estimated. Finally, numerical experiments emphasize the possibility to compute, for a given function and a given continuous mesh, the continuous interpolation error without any discrete supports. The correlation between continuous and discrete estimations of the interpolation error is finally shown.

1 Metric notion for mesh adaptation

In order to be self-contained, we recall the differential geometry notions that are used in the sequel. It mainly concerns the computation of lengths for different kinds of metric spaces: Euclidean, Euclidean metric space and Riemannian metric space. A complete review and the mathematical study of these spaces are available in [2, 3, 11].

Notations. Bolded symbols, as $\mathbf{a}, \mathbf{b}, \mathbf{u}, \mathbf{v}, \mathbf{x}, \ldots$, denote vectors or points of $\mathbb{R}^n$. Vector coordinates are denoted by $\mathbf{x} = (x_1, \ldots, x_n)$. And, the natural dot product between two vectors $\mathbf{u}$ and $\mathbf{v}$ of $\mathbb{R}^n$ is:

$$\langle \mathbf{u},\, \mathbf{v} \rangle = \sum_{i=1}^{n} u_i\, v_i.$$

1.1 Euclidean metric space

An **Euclidean metric space** $(\mathbb{R}^n, \mathcal{M})$ of dimension n is a finite vector space where the dot product is defined by means of a symmetric definite positive form $\mathcal{M}$:

$$\langle \mathbf{u}, \mathbf{v} \rangle_{\mathcal{M}} = \langle \mathbf{u}, \mathcal{M}\mathbf{v} \rangle = {}^t\mathbf{u}\mathcal{M}\mathbf{v}\,, \quad \text{for } (\mathbf{u}, \mathbf{v}) \in \mathbb{R}^n \times \mathbb{R}^n\,.$$

The form $\mathcal{M}$ is usually written as a $n \times n$ matrix that is:

(i) (symmetric) $\forall (\mathbf{u}, \mathbf{v}) \in \mathbb{R}^n \times \mathbb{R}^n\,, \quad \langle \mathbf{u}, \mathcal{M}\mathbf{v} \rangle = \langle \mathbf{v}, \mathcal{M}\mathbf{u} \rangle$

(ii) (positive) $\forall \mathbf{u} \in \mathbb{R}^n\,, \quad \langle \mathbf{u}, \mathcal{M}\mathbf{u} \rangle \geq 0$

(iii) (definite) $\langle \mathbf{u}, \mathcal{M}\mathbf{u} \rangle = 0 \Longrightarrow \mathbf{u} = \mathbf{0}$.

These properties ensure that $\mathcal{M}$ defines a dot-product. In the following, the matrix $\mathcal{M}$ is simply called a **metric tensor** or a **metric**. The simplest example of an Euclidean metric space is given by the identity matrix I_n which spans the canonical Euclidean space $\mathbb{R}^n$.

The dot product defined by $\mathcal{M}$ spans a normed vector space $(\mathbb{R}^n, \|.\|_{\mathcal{M}})$ and a metric vector space $(\mathbb{R}^n, d_{\mathcal{M}}(.,.))$ supplied by the following norm and distance definition:

- $\forall \mathbf{u} \in \mathbb{R}^n\,, \quad \|\mathbf{u}\|_{\mathcal{M}} = \sqrt{\langle \mathbf{u}, \mathcal{M}\mathbf{u}\rangle}$
- $\forall (\mathbf{u}, \mathbf{v}) \in \mathbb{R}^n \times \mathbb{R}^n\,, \quad d_{\mathcal{M}}(\mathbf{u}, \mathbf{v}) = \|\mathbf{u} - \mathbf{v}\|_{\mathcal{M}}$.

In these spaces, the length $\ell_{\mathcal{M}}$ of a segment $\mathbf{ab} = [\mathbf{a}, \mathbf{b}]$ is given by the distance between its extremities:

$$\ell_{\mathcal{M}}(\mathbf{ab}) = d_{\mathcal{M}}(\mathbf{a}, \mathbf{b}).$$

Note that this property is generally wrong for a general Riemannian metric space defined hereafter.

In an Euclidean metric space, angles and volumes are still well posed. These features are of main interest when dealing with mesh generation. These quantities are generally introduced to define quality functions. Given a bounded subset K of $\mathbb{R}^n$, the volume of K computed with respect to metric tensor $\mathcal{M}$ is:

$$|K|_{\mathcal{M}} = \int_K \sqrt{\det \mathcal{M}}\, \mathrm{d}K = \sqrt{\det \mathcal{M}}\, |K|_{I_n}\,. \tag{1}$$

The angle between two non-zero vectors $\mathbf{u}$ and $\mathbf{v}$ is defined by the unique real-value $\theta \in [0, \pi]$ verifying:

$$\cos(\theta) = \frac{\langle \mathbf{u}, \mathbf{v}\rangle_{\mathcal{M}}}{\|\mathbf{u}\|_{\mathcal{M}} \|\mathbf{v}\|_{\mathcal{M}}}\,.$$

Geometric interpretation. We will often refer to the geometric interpretation of a metric tensor. This geometric view plays an important role in the continuous mesh model. In the vicinity $\mathcal{V}(\mathbf{a})$ of a point $\mathbf{a}$, the set of points, that are at a distance ε of $\mathbf{a}$, is given by:

$$\Phi_{\mathcal{M}}(\varepsilon) = \left\{ \mathbf{x} \in \mathcal{V}(\mathbf{a}) \mid {}^t(\mathbf{x} - \mathbf{a})\, \mathcal{M}\, (\mathbf{x} - \mathbf{a}) \leq \varepsilon^2 \right\}.$$

We note that it is sufficient to describe $\Phi_{\mathcal{M}}(1)$ as $\Phi_{\mathcal{M}}(\varepsilon)$ can be deduced from $\Phi_{\mathcal{M}}(1)$ for all ε by homogeneity:

$$\Phi_{\mathcal{M}}(\varepsilon) = \left\{ \varepsilon^{-1}\, \mathbf{x} \in \mathcal{V}(\mathbf{a}) \mid \mathbf{x} \in \Phi_{\mathcal{M}}(1) \right\}.$$

To describe $\Phi_{\mathcal{M}}(1)$, the spectral decomposition $\mathcal{M} = \mathcal{R}\, \Lambda\, {}^t\mathcal{R}$ is used. $\mathcal{R}$ is an orthonormal matrix verifying ${}^t\mathcal{R}\mathcal{R} = \mathcal{R}\,{}^t\mathcal{R} = I_n$. It is composed of the eigenvectors of $\mathcal{M}$. Λ is a diagonal matrix composed of the eigenvalues of $\mathcal{M}$. Eigenvalues $(\lambda_i)_{i=1,n}$ are strictly positive. In the eigenvectors frame, the initial quadratic form ${}^t(\mathbf{x} - \mathbf{a})\, \mathcal{M}\, (\mathbf{x} - \mathbf{a})$ becomes ${}^t(\tilde{\mathbf{x}} - \tilde{\mathbf{a}})\, \Lambda\, (\tilde{\mathbf{x}} - \tilde{\mathbf{a}})$. Consequently, $\Phi_{\mathcal{M}}(1)$ can be rewritten in this basis:

$$\begin{aligned}
\Phi_{\mathcal{M}}(1) &= \left\{ \tilde{\mathbf{x}} \in \mathcal{V}(\tilde{\mathbf{a}}) \mid \sum_{i=1}^{n} \lambda_i\, (\tilde{x}_i - \tilde{a}_i)^2 \leq 1 \right\} \\
&= \left\{ \tilde{\mathbf{x}} \in \mathcal{V}(\tilde{\mathbf{a}}) \mid \sum_{i=1}^{n} \left(\frac{\tilde{x}_i - \tilde{a}_i}{h_i} \right)^2 \leq 1 \right\}.
\end{aligned}$$

The last relation defines an ellipsoid centered at **a** with its axes aligned with the principal directions of $\mathcal{M}$. Sizes along these directions are given by $h_i = \lambda_i^{-\frac{1}{2}}$. We denote by $\mathcal{E}_\mathcal{M}$ this ellipsoid. Figure 1 depicts $\mathcal{E}_\mathcal{M}$. In the sequel, the set $\Phi_\mathcal{M}(1)$ is called the **unit ball** of metric $\mathcal{M}$ and it is denoted by $\mathcal{B}_\mathcal{M}$.

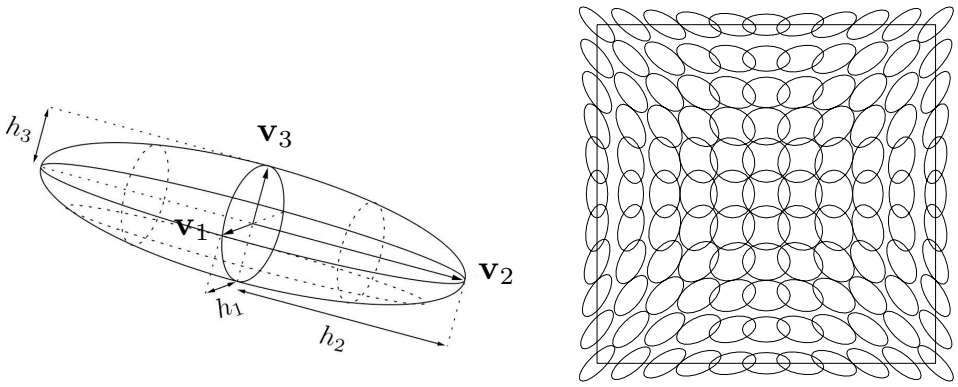

Figure 1: *Left, geometric interpretation of* $\mathcal{B}_\mathcal{M} = \Phi_\mathcal{M}(1)$. $\mathbf{v}_i$ *are the eigenvectors of* $\mathcal{M}$ *and* h_i^{-2} *are the eigenvalues of* $\mathcal{M}$. *Right, geometric visualization of a Riemannian metric space* $(\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in[0,1]\times[0,1]}$. *At each point* $\mathbf{x}$ *of the domain, the unit ball of metric* $\mathcal{M}(\mathbf{x})$ *is drawn.*

Natural metric mapping. The last information handled by a metric tensor $\mathcal{M}$ is the definition of an application that maps the unit ball $\mathcal{B}_{I_n}$ of identity metric I_n onto the unit ball $\mathcal{B}_\mathcal{M}$ of metric $\mathcal{M}$. This mapping is given by the application $\mathcal{M}^{-\frac{1}{2}} : \mathbb{R}^n \mapsto \mathbb{R}^n$. $\mathcal{M}^{-\frac{1}{2}}$ is defined by the spectral decomposition $\mathcal{M}^{-\frac{1}{2}} = \mathcal{R}\,\Lambda^{-\frac{1}{2}}\,{}^t\mathcal{R}$, where $\Lambda^{-\frac{1}{2}}$ is the diagonal matrix composed of the inverse of the square root of the eigenvalues of $\mathcal{M}$. This mapping provides another description of the ellipsoid $\mathcal{E}_\mathcal{M}$:

$$\mathcal{E}_\mathcal{M} = \left\{\mathcal{M}^{-\frac{1}{2}}\,\mathbf{x} \mid \|\mathbf{x}\|_2^2 = 1\right\}.$$

1.2 Riemannian metric space

When a metric tensor field is varying smoothly in the whole domain Ω, a **Riemannian metric space** is defined. We denote this space by $\mathbf{M} = (\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$. The continuous mesh model is based on such a space. To give a practical visualization of a Riemannian metric space, the unit ball of the metric at some points of the domain has been drawn, see Figure 1 (right).

The main operation performed in this space is the computation of edges lengths. It is important to note that, in a Riemannian metric space, computing the length of a segment (*i.e.,* an edge) differs from evaluating the distance between the extremities of this segment. Indeed, the straight line is no more the shortest path between two points which is given by

a geodesic. To take into account the variation of the metric along the edge, the edge length is evaluated with an integral formula:

Definition 1 (Edge length computation)**.** *In a Riemannian metric space* $\mathbf{M} = (\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$*, the length of edge* $\mathbf{ab}$ *is computed using the straight line parameterization* $\gamma(t) = \mathbf{a} + t\,\mathbf{ab}$*, where* $t \in [0,1]$*:*

$$\ell_{\mathcal{M}}(\mathbf{ab}) = \int_0^1 \|\gamma'(t)\|_{\mathcal{M}}\,\mathrm{d}t = \int_0^1 \sqrt{t\mathbf{ab}\,\mathcal{M}(\mathbf{a}+t\,\mathbf{ab})\,\mathbf{ab}}\,\mathrm{d}t. \tag{2}$$

Figure 2 depicts iso-values of segment length from the origin for different Riemannian metric spaces. The plotted function is $f(\mathbf{x}) = \ell_{\mathcal{M}}(\mathbf{ox})$ where $\mathbf{o}$ is the origin of the plane. The iso-values are isotropic for the Euclidean space. They are anisotropic in the case of an Euclidean metric space defined by $\mathcal{M}$. The two principal directions of $\mathcal{M}$ clearly appear. In the case of a Riemannian metric space $\mathbf{M}$, all previous symmetries are lost.

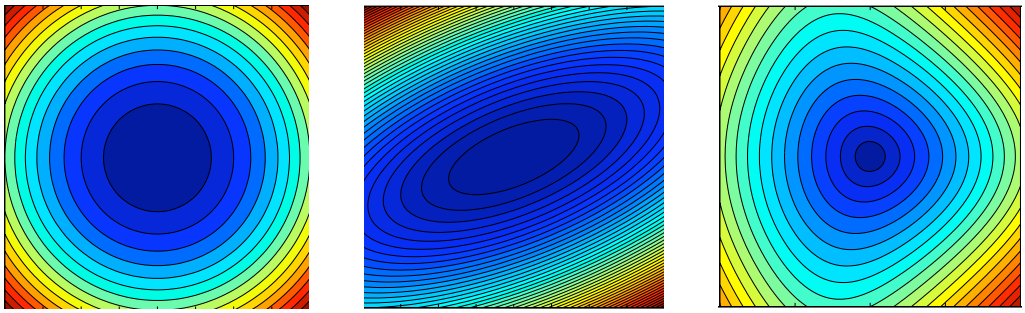

Figure 2: *Iso-values of the function* $f(\mathbf{x}) = \ell_{\mathcal{M}}(\mathbf{ox})$ *where* $\mathbf{o}$ *is the origin, i.e., segment length issued from the origin, for different Riemannian metric spaces. Left, in the canonical Euclidean space* $([-1,1]\times[-1,1], I_2)$*, middle, in an Euclidean metric space* $([-1,1]\times[-1,1], \mathcal{M})$ *with* $\mathcal{M}$ *constant and, right, in a Riemannian metric space* $(\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in[-1,1]^2}$ *with a varying metric tensor field.*

2 Continuous mesh model

In this section, the definition of a continuous element is first introduced. It is defined locally by considering a constant metric tensor. Then, a continuous mesh model is defined based on the notion of Riemannian metric space. In both cases, we give the set of all discrete elements or all discrete meshes that are well represented by these continuous models. These classes of equivalence are based on the notion of unit element and of unit mesh.

2.1 Continuous element and class of unit elements

In the continuous framework, a metric tensor $\mathcal{M}$ is a **continuous element**.

Remark 1. *A continuous element $\mathcal{M}$ is a metric tensor and it also defines an Euclidean metric space $(\mathbb{R}^n, \mathcal{M})$. In the sequel, we may use one these equivalent definitions of a continuous element.*

The class of all discrete elements, that are represented by this continuous model, is given by the following definition:

Definition 2 (Unit element)**.** *An element K is* **unit** *with respect to a continuous element $\mathcal{M}$ if the length of all its edges is unit in the metric $\mathcal{M}$. If K is given by its list of edges $(\mathbf{e_i})_{i=1..n(n+1)/2}$, then :*

$$\forall i = 1, ..., \frac{n(n+1)}{2}, \;\; \ell_{\mathcal{M}}(\mathbf{e_i}) = 1\,.$$

The volume of K is given by:

$$|K|_{\mathcal{M}} = \frac{\sqrt{n+1}}{2^{n/2}\,n!} \;\; \textit{and} \;\; |K|_{I_n} = \frac{\sqrt{n+1}}{2^{n/2}\,n!}\sqrt{\det(\mathcal{M})}.$$

Volume and length values are deduced from the Euclidean space example where the metric $\mathcal{M}$ is the identity matrix I_n. In this case, the element is the regular n-simplex. It is composed of $n+1$ vertices connected by $\frac{n(n+1)}{2}$ edges of unit length. We can prove by induction that the volume V_n of the regular n-simplex is given by:

$$V_n = \frac{\sqrt{n+1}}{2^{n/2}\,n!}.$$

To this end, we use the following induction formulas:

$$t_n^2 + R_{n-1}^2 = 1 \text{ and } \quad V_n = \tfrac{1}{n} V_{n-1} t_n,$$

where t_n is the length from one vertex to the center of the opposite face. The formulas are initialized by the 2D area or the 3D volume of the regular triangle or tetrahedron:

$$V_2 = \frac{\sqrt{3}}{4}, \quad t_2 = \frac{\sqrt{3}}{2}, \quad V_3 = \frac{\sqrt{2}}{12} \quad \text{and} \quad t_3 = \sqrt{\frac{2}{3}}.$$

Figure 3 gives two examples of unit elements for two different metric tensors. More generally, the relationships between unit discrete elements and a continuous element are stated in the following proposition:

Proposition 1 (Equivalence classes)**.** *Let $\mathcal{M}$ be a continuous element, there exists a non-empty infinite set of unit elements with respect to $\mathcal{M}$. Conversely, given an element $K = (\mathbf{e_i})_{i=1..n(n+1)/2}$ such that $|K|_{I_n} \neq 0$, then there is a unique continuous element $\mathcal{M}$ for which element K is unit with respect to $\mathcal{M}$.*
The relation unit with respect to $\mathcal{M}$ *defines a class of equivalence among the set of all discrete elements.*

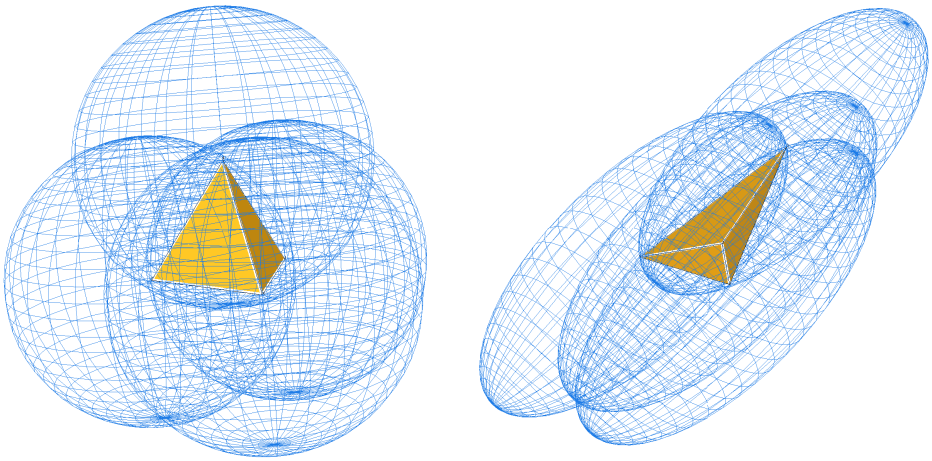

Figure 3: *3D examples of a unit element with respect to the identity metric (left) and to an anisotropic metric tensor (right). In each case, the unit ball of the metric is drawn at each vertex of the unit element.*

Proof. We first examine the uniform case where $\mathcal{M} = I_3$. The general case is deduced from it by using the mapping $\mathcal{M}^{-\frac{1}{2}}$. Let K_0 be a regular tetrahedron, thereby K_0 is unit with respect to I_3. Whatever the rotation matrix $\mathcal{R}$ verifying ${}^t\mathcal{R}\mathcal{R} = \mathcal{R}{}^t\mathcal{R} = I_3$, the tetrahedron $\mathcal{R}\,K_0$ is still unit for I_3. Consequently, the class of all unit elements for the continuous element I_3 is:

$$\mathcal{K} = \{K \,|\, \forall \mathcal{R} \in \mathcal{O}_3 : \; K = \mathcal{R}K_0\} \;\text{ with } \mathcal{O}_3 = \left\{\mathcal{R} \,|\, {}^t\mathcal{R}\mathcal{R} = \mathcal{R}{}^t\mathcal{R} = I_3\right\}.$$

The equivalence class of the unit elements with respect to $\mathcal{M}$ is then given by the set:

$$\left\{\mathcal{M}^{-\frac{1}{2}}K \,\middle|\, \forall K \in \mathcal{K}\right\}.$$

Conversely, given a non-degenerated discrete element $K = (\mathbf{e_i})_{i=1..n(n+1)/2}$, such that $|K|_{I_3} \neq 0$, let us demonstrate that there exists a unique metric for which K is unit. It is sufficient to solve the following linear system:

$$(S) \; \begin{cases} \ell^2_{\mathcal{M}}(\mathbf{e_1}) = 1 \\ \vdots \\ \ell^2_{\mathcal{M}}(\mathbf{e_{n(n+1)/2}}) = 1\,. \end{cases}$$

The determinant of (S) is equal to $|K|_{I_3} \neq 0$. Consequently, (S) admits a unique solution. □

Figure 4 depicts some unit elements with respect to a continuous element.

2.2 Geometric invariants

So far, only Definition 2 has established relationships between unit elements and continuous elements. Other properties exist. They connect the geometric properties of unit elements to the linear algebra properties of metric tensors. The following proposition gives geometric invariants that hold for all unit elements with respect to a continuous element.

Proposition 2 (Geometric invariants)**.** *Let $\mathcal{M}$ be a continuous element and K be a unit element with respect to $\mathcal{M}$. We denote by $(\mathbf{e}_i)_i$ its edges list, see conventions in Figure 5, and by $|K|$ its Euclidean volume. Then, the following invariants hold:*

- *standard invariants:*

$$\forall\,(\mathbf{e}_i,\mathbf{e}_j),\qquad \begin{cases} {}^t\mathbf{e}_i\,\mathcal{M}\,\mathbf{e}_i = 1, \\ 2\,{}^t\mathbf{e}_i\,\mathcal{M}\,\mathbf{e}_j + 1 = 0 \;\; \textit{if } i \neq j. \end{cases} \tag{3}$$

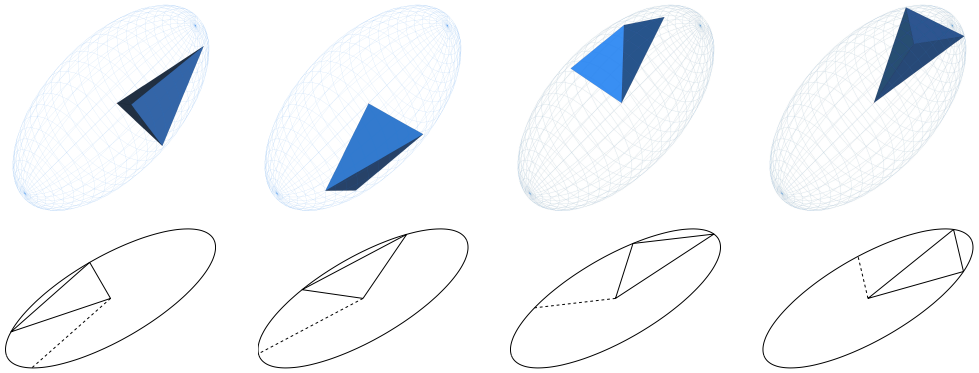

Figure 4: *Several unit elements with respect to a continuous element in 2D and 3D.*

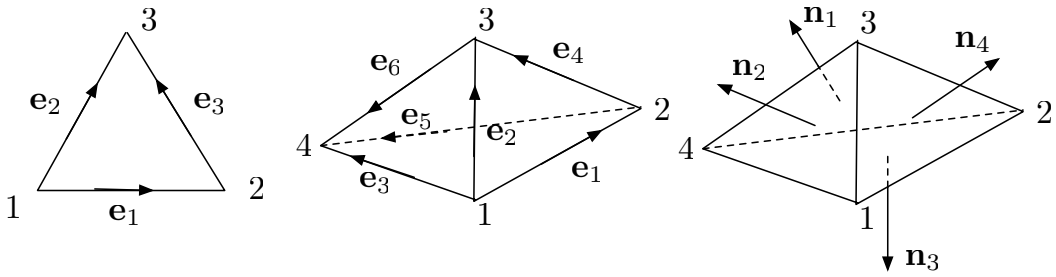

Figure 5: *Conventions used to enumerate the edges and the faces of a triangle and of a tetrahedron.*

- *invariant related to the Euclidean volume $|K|$:*

$$|K| = \frac{\sqrt{3}}{4}\,\det(\mathcal{M}^{-\frac{1}{2}})\ \textit{in } 2D \quad \textit{and} \quad |K| = \frac{\sqrt{2}}{12}\,\det(\mathcal{M}^{-\frac{1}{2}})\ \textit{in } 3D. \tag{4}$$

- *invariant related to the square length of the edges for all symmetric matrix H:*

$$\begin{aligned} &\sum_{i=1}^{3} {}^t\mathbf{e}_i\, H \mathbf{e}_i = \frac{3}{2}\mathrm{trace}(\mathcal{M}^{-\frac{1}{2}} H \mathcal{M}^{-\frac{1}{2}})\ \textit{in } 2D, \\ &\sum_{i=1}^{6} {}^t\mathbf{e}_i\, H \mathbf{e}_i = 2\,\mathrm{trace}(\mathcal{M}^{-\frac{1}{2}} H \mathcal{M}^{-\frac{1}{2}})\ \textit{in } 3D. \end{aligned} \tag{5}$$

Proof. The first invariant of Relation (3) comes from the definition of a unit element.

The second invariant of Relation (3) states that the angle between two edges of a unit element face is constant in the metric. Let $(\mathbf{e}_i, \mathbf{e}_j)$ be a couple of edges of element K, this couple defines a face. We denote by $\mathbf{e}_k$ the third edge of this face. According to the conventions depicted in Figure 5, these edges verify: $\mathbf{e}_i + \mathbf{e}_j - \mathbf{e}_k = \mathbf{0}$. Expanding the following relation

$$^t(\mathbf{e}_i + \mathbf{e}_j + \mathbf{e}_k)\,\mathcal{M}\,(\mathbf{e}_i + \mathbf{e}_j - \mathbf{e}_k) = 0,$$

leads to the second invariant of Relation (3).

Invariant (4) is proved by a direct integration. Given a unit element K for $\mathcal{M}$, there exists a unique regular tetrahedron K_0, which is unit with respect to the identity matrix I_3, such that $K = \mathcal{M}^{-\frac{1}{2}} K_0$. The volume of K is then given by:

$$|K| = \int_K 1\,\mathrm{d}\mathbf{x} = \int_{K_0} \det(\mathcal{M}^{-\frac{1}{2}})\,\mathrm{d}\mathbf{x} = \det(\mathcal{M}^{-\frac{1}{2}})\,|K_0|,$$

where $|K_0| = \frac{\sqrt{2}}{12}$. The same proof applies in 2D.

Invariant (5) is first proved in the simpler case where $H = I_3$ and $\mathcal{M} = I_3$. The general case will be deduced from this proof. Let us consider the regular tetrahedron $K_0 = (\mathbf{v_1}, \mathbf{v_2}, \mathbf{v_3}, \mathbf{v_4})$ unit for I_3 defined by the list of vertices:

$$\mathbf{v_1} = (0,0,0)\,,\ \mathbf{v_2} = (1,0,0)\,,\ \mathbf{v_3} = \left(\frac{1}{2}, \frac{\sqrt{3}}{2}, 0\right) \text{ and } \mathbf{v_4} = \left(\frac{1}{2}, \frac{\sqrt{3}}{6}, \sqrt{\frac{2}{3}}\right).$$

The proof does not depend on this specific choice of coordinates. We first demonstrate the following preliminary result: *For all lines (D) passing through one of the vertices of K_0, the sum of the square lengths of the edges projected on (D) is invariant.* Without loss of generality, we assume that line (D) passes through the vertex $\mathbf{v_1}$ of K_0. If line (D) is defined by the vector

$$\mathbf{n} = (\cos(u)\cos(v), \cos(u)\sin(v), \sin(u))\,,$$

with $(u,v) \in \mathbb{R}^2$, then the length of the first three edges of K_0 projected on (D) are given by:

$$\begin{aligned} a &= \mathbf{e_1} \,.\, \mathbf{n} = \cos(u)\cos(v), \\ b &= \mathbf{e_2} \,.\, \mathbf{n} = \frac{1}{2}\cos(u)\cos(v) + \frac{\sqrt{3}}{2}\cos(u)\sin(v), \\ c &= \mathbf{e_3} \,.\, \mathbf{n} = \frac{1}{2}\cos(u)\cos(v) + \frac{\sqrt{3}}{6}\cos(u)\sin(v) + \sqrt{\frac{2}{3}}\sin(u), \end{aligned}$$

with conventions of Figure 5. A direct trigonometric calculus shows that the sum of the square length of all the edges projected on (D) is equal to 2. Indeed, it comes:

$$\begin{aligned} \Sigma &= a^2 + b^2 + c^2 + (b-c)^2 + (c-a)^2 + (a-b)^2 \\ &= 3\,a^2 + 3\,b^2 + 3\,c^2 - 2ab - 2ac - 2bc. \end{aligned}$$

After expanding and factorizing, Σ reads:

$$\Sigma = 2\cos(u)^2\cos(v)^2 + 2\cos(u)^2\sin(v)^2 + 2\sin(u)^2 = 2.$$

When $\mathcal{M}$ is different from I_3, we use the mapping $\mathcal{M}^{-\frac{1}{2}}$ that maps the unit ball of I_3 onto the unit ball of $\mathcal{M}$. As regards line (D), we select the specific line which has for direction vector one the main direction of $\mathcal{M}$, e.g. $\mathbf{u}_j$, and which is passing through $\mathbf{v_1}$. The lengths a, b and c are thus multiplied by h_j which is the size prescribed by $\mathcal{M}$ in the direction $\mathbf{u}_j$. Consequently, the square length of the edges projected on (D) are multiplied by h_j^2. It comes:

$$\Sigma_j = \sum_{i=1}^{6} |\mathbf{e}_i \,.\, \mathbf{u}_j|^2 = h_j^2\,\Sigma = 2\,h_j^2.$$

Considering the previous relation for all the principal directions of $\mathcal{M}$ and summing the Σ_j complete the proof:

$$\sum_{i=1}^{6} \|\mathbf{e}_i\|_2^2 = \sum_{j=1}^{3}\sum_{i=1}^{6} |\mathbf{e}_i \,.\, \mathbf{u}_j|^2 = 2\left(h_1^2 + h_2^2 + h_3^2\right) = 2\,\mathrm{trace}(\mathcal{M}^{-1}). \tag{6}$$

We now consider the more general case where H is symmetric definite positive. The matrices $H^{\frac{1}{2}}$ and $H^{-\frac{1}{2}}$ are well defined and are symmetric. We first prove that if K is a unit element for $\mathcal{M}$ then $H^{\frac{1}{2}}K$ is a unit element with respect to $\mathcal{M}^{\frac{1}{2}}\,H^{-1}\,\mathcal{M}^{\frac{1}{2}}$. Indeed, if we consider the edge $\tilde{\mathbf{e}}_\mathbf{j} = H^{\frac{1}{2}}\mathbf{e_j}$ of $H^{\frac{1}{2}}K$ where $\mathbf{e_j}$ is a unit length edge of K with respect to $\mathcal{M}$, it comes:

$${}^t\tilde{\mathbf{e}}_\mathbf{j}\,\mathcal{M}^{\frac{1}{2}}\,H^{-1}\,\mathcal{M}^{\frac{1}{2}}\,\tilde{\mathbf{e}}_\mathbf{j} = {}^t(H^{\frac{1}{2}}\mathbf{e_j})\,\mathcal{M}^{\frac{1}{2}}\,H^{-1}\,\mathcal{M}^{\frac{1}{2}}\,(H^{\frac{1}{2}}\mathbf{e_j}) = 1.$$

Then, from Relation (6) we get the last invariant:

$$\sum_{i=1}^{6} {}^t\mathbf{e}_i\,H\mathbf{e}_i = \sum_{i=1}^{6} \|\tilde{\mathbf{e}}_i\|_2^2 = 2\,\mathrm{trace}(\mathcal{M}^{-\frac{1}{2}}H\mathcal{M}^{-\frac{1}{2}})\,.$$

□

Other geometric invariants can be found in [25].

2.3 Continuous mesh and class of unit meshes

A continuous mesh of a domain $\Omega \subset \mathbb{R}^n$ is a Riemannian metric space $\mathbf{M} = (\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\mathbf{\Omega}}$. We recall the spectral decomposition of $\mathcal{M}(\mathbf{x})$:

$$\begin{aligned}\mathbf{M}:\ \mathbf{x}\in\Omega\mapsto\mathcal{M}(\mathbf{x}) &= \mathcal{R}(\mathbf{x})\begin{pmatrix}\lambda_1(\mathbf{x}) & & \\ & \ddots & \\ & & \lambda_n(\mathbf{x})\end{pmatrix}{}^t\mathcal{R}(\mathbf{x})\\ &= \mathcal{R}(\mathbf{x})\begin{pmatrix}h_1^{-2}(\mathbf{x}) & & \\ & \ddots & \\ & & h_n^{-2}(\mathbf{x})\end{pmatrix}{}^t\mathcal{R}(\mathbf{x}),\end{aligned}$$

where $\mathcal{R}(\mathbf{x})$ is an orthonormal matrix providing the local orientation, $(\lambda_i(\mathbf{x}))_{i=1,n}$ are the local eigenvalues and $(h_i(\mathbf{x}))_{i=1,n}$ are the local sizes along the principal directions of the continuous mesh. Practically, another decomposition is used that points out the local characteristics of the continuous mesh. This decomposition is given by the following proposition.

Proposition 3 (Continuous mesh)**.** *A continuous mesh* $\mathbf{M} = (\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\mathbf{\Omega}}$ *locally writes:*

$$\mathcal{M}(\mathbf{x}) = d^{\frac{2}{n}}(\mathbf{x})\,\mathcal{R}(\mathbf{x})\begin{pmatrix}r_1^{-\frac{2}{n}}(\mathbf{x}) & & \\ & \ddots & \\ & & r_n^{-\frac{2}{n}}(\mathbf{x})\end{pmatrix}{}^t\mathcal{R}(\mathbf{x}),$$

where

- *the density* d *is equal to:* $d = \left(\prod_{k=1}^{n} h_k\right)^{-1} = \left(\prod_{k=1}^{n}\lambda_k\right)^{\frac{1}{2}}$,
- *the* n *anisotropic quotients* r_i *are equal to:* $r_i = h_i^n\left(\prod_{k=1}^{n} h_k\right)^{-1}$.

Proof. The proof consists in computing $d^{\frac{2}{n}}\, r_i^{-\frac{2}{n}}$:

$$d^{\frac{2}{n}}\, r_i^{-\frac{2}{n}} = \left(\prod_{k=1}^{n} h_k\right)^{-\frac{2}{n}} h_i^{-2}\left(\prod_{k=1}^{n} h_k\right)^{\frac{2}{n}} = h_i^{-2} = \lambda_i.$$

□

The density d controls only the local level of accuracy of the continuous mesh. Increasing or decreasing d does not change the anisotropic properties or the orientation, see Figure 6 (left). In 3D, anisotropic quotients arises from the quotient of different parallelepipeds, see Figure 6 (right).

We also define the complexity $\mathcal{C}$ of a continuous mesh:

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

This real-value parameter is useful to quantify the global level of accuracy of the continuous mesh $(\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\mathbf{\Omega}}$. It can also be interpreted as the continuous counterpart of the number of vertices of a discrete mesh. This quantity also leads to the definition of sequence of continuous embedded meshes. Two continuous embedded meshes have the same anisotropic ratios and orientations. They only differ from their complexity:

Definition 3 (Embedded continuous meshes)**.** *Two continuous meshes* $(\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\mathbf{\Omega}}$ *and* $(\mathcal{N}(\mathbf{x}))_{\mathbf{x}\in\mathbf{\Omega}}$ *are embedded if a constant* c *exists such that:*

$$\forall \mathbf{x} \in \Omega, \quad \mathcal{N}(\mathbf{x}) = c\,\mathcal{M}(\mathbf{x}).$$

Conversely, from a continuous mesh $\mathbf{M} = (\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\mathbf{\Omega}}$, we can deduce a continuous mesh $\mathbf{N} = (\mathcal{N}(\mathbf{x}))_{\mathbf{x}\in\mathbf{\Omega}}$ of complexity N having the same anisotropic properties (anisotropic orientations and ratios) by considering:

$$\mathcal{N}(\mathbf{x}) = \left(\frac{N}{\mathcal{C}(\mathbf{M})}\right)^{\frac{2}{n}} \mathcal{M}(\mathbf{x}).$$

In the context of error estimation, this notion enables the study of the order of convergence with respect to an increasing complexity N. Consequently, the complexity $\mathcal{C}(\mathbf{M})$ is also the continuous counterpart of the classical parameter h used for uniform meshes while studying convergence. In the continuous mesh framework, the uniform refinement consisting in dividing by two each edge of a uniform mesh of size h writes:

$$\mathcal{M}_i = 4^i \begin{pmatrix} \frac{1}{h^2} & & \\ & \frac{1}{h^2} & \\ & & \frac{1}{h^2} \end{pmatrix},$$

where i is the level of refinement. $(\mathcal{M}_i)_{i=1\ldots k}$ defines a sequence of continuous embedded meshes. Consequently, this simple practical adaptive strategy has a simple continuous interpretation in term of embedded continuous meshes. Such a uniform refinement is exemplified on Figure 7. However, when dealing with anisotropic meshes, a unique size h is no more sufficient to give a quantitive information on the accuracy. The size h is then replaced by the continuous mesh complexity.

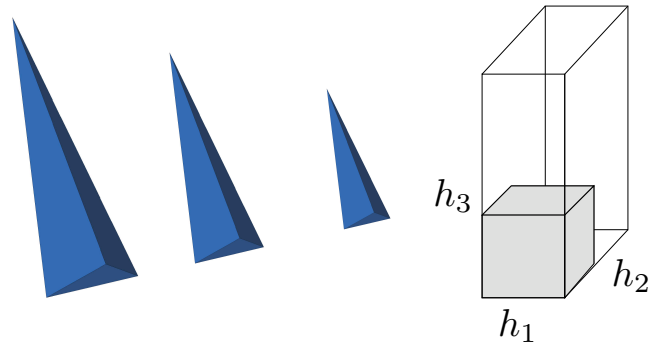

Figure 6: *Left, different unit elements where only the density increases from left to right. Right, the geometric interpretation of anisotropic quotients as quotients of parallelepipeds volumes.*

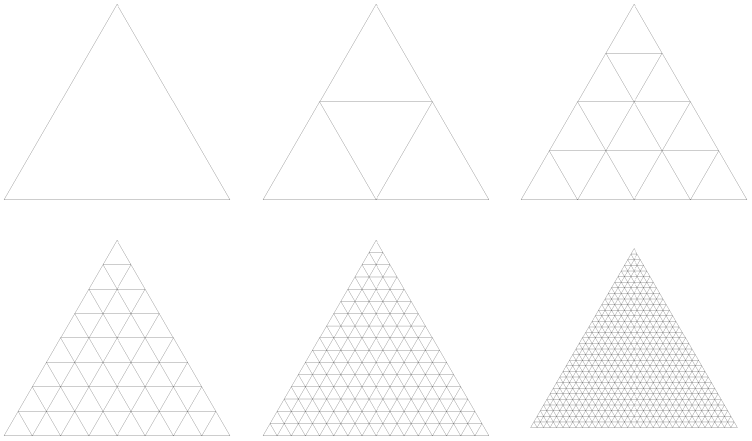

Figure 7: *In a sequence of uniformly refined uniform meshes, a single size information h is sufficient to describe the accuracy of the current mesh in the whole domain. On the contrary when the mesh involves strong differences in sizes and orientations, this size is replaced by another measure, the complexity.*

Unit mesh. The notion of unit mesh is far more complicated than the notion of unit element as the existence of a mesh composed only of unit regular simplexes with respect to a given continuous mesh is not guaranteed. For instance, if the continuous mesh is not compatible with the domain size, then it clearly does not exist such discrete mesh. To avoid this problem, let us look at the existence of a discrete mesh composed only with unit regular simplexes with respect to a continuous mesh in $\mathbb{R}^n$. To simplify even more the problem, we first consider the continuous mesh $(I_n(\mathbf{x}))_{\mathbf{x}\in\mathbb{R}^n}$.

It is well known that $\mathbb{R}^3$ cannot be filled only with the regular tetrahedron while it is possible to fill $\mathbb{R}^2$ with the equilateral triangle. Consequently, even for the simplest continuous mesh $(I_3(\mathbf{x}))_{\mathbf{x}\in\mathbb{R}^3}$, there is no discrete mesh composed only of the unit regular tetrahedron. Therefore, the notion of unit mesh has to be released:

Definition 4 (Unit mesh)**.** *A discrete mesh $\mathcal{H}$ of a domain $\Omega \subset \mathbb{R}^n$ is* **unit** *for a continuous mesh $(\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$ if all its elements are quasi-unit.*

Now, let us give a meaning to quasi-unit in three dimensions. A first way to release the definition of unity is to take into account technical constraints imposed by mesh generators. To converge (and to avoid cycling) while analyzing edges length, the meshing algorithm considers an admissible edge length interval of the form $[\frac{1}{\alpha}, \alpha]$ with $\alpha > 0$ [16]. If the symmetry property is required, *i.e.,* $\frac{\alpha}{2} = \frac{1}{\alpha}$, then we obtain $\alpha = \sqrt{2}$. Therefore, as regards the meshing requirement, a tetrahedron K defined by its list of edges $(\mathbf{e_i})_{i=1\ldots6}$ is said quasi-unit if $\forall i \in [1,6],\ \ell_{\mathcal{M}}(\mathbf{e_i}) \in [\frac{1}{\sqrt{2}}, \sqrt{2}]$. Nevertheless, we do not know if this definition provide the existence of a unit mesh for the continuous mesh $(I_3(\mathbf{x}))_{\mathbf{x}\in\mathbb{R}^3}$. In the following, this question of existence is studied by means of the space filling tetrahedra.

Non-regular space filling tetrahedra. The study of space filling tetrahedra is an old geometrical question [28, 31]. In the past, it has been demonstrated that there exist sets of non-regular space filling tetrahedra: the Sommerville tetrahedra [32] and the Goldberg tetrahedra family [19].

The Sommerville tetrahedra are based on particular splittings of the unit cube, see Figure 8. We recall these tetrahedra thanks to their vertices coordinates, only the last vertex distinguishes them. K is denoted $(\mathbf{v_1}, \mathbf{v_2}, \mathbf{v_3}, \mathbf{v_4})$ with $\mathbf{v_1} = (0,0,0)$, $\mathbf{v_2} = (\frac{1}{2}, -\frac{1}{2}, \frac{1}{2})$, $\mathbf{v_3} = (\frac{1}{2}, \frac{1}{2}, \frac{1}{2})$ and

- $\mathbf{v_4} = (\frac{1}{2}, 0, 0)$ for the Sommeville tetrahedron 1
- $\mathbf{v_4} = (1, 0, 0)$ for the Sommeville tetrahedron 2
- $\mathbf{v_4} = (\frac{1}{2}, -\frac{1}{2}, -\frac{1}{2})$ for the Sommeville tetrahedron 3
- $\mathbf{v_4} = (\frac{1}{2}, 0, -\frac{1}{4})$ for the Sommeville tetrahedron 4.

The Goldberg tetrahedra are based on the splitting of a prism the basis of which is the equilateral triangle, see Figure 9. Their coordinates depend on an initial choice of two lengths a and e. We specify one of the Goldberg tetrahedra for the specific choice $a = \frac{1}{3}$ and $e = 1$:

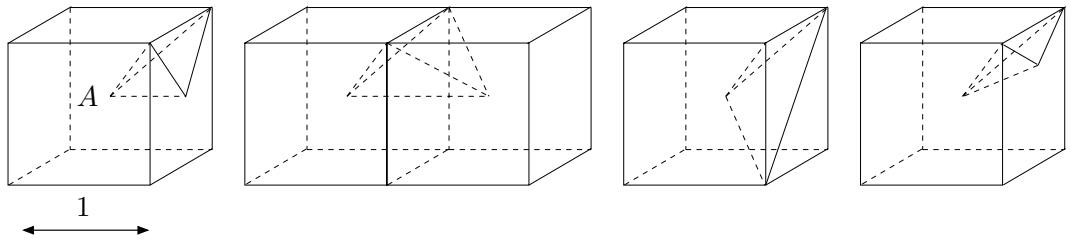

Figure 8: *From left to right, Sommerville tetrahedra* 1, 2, 3 *and* 4.

- $\mathbf{v_1} = (0,0,0)$, $\mathbf{v_2} = (0,0,1)$, $\mathbf{v_3} = (0,1,\frac{1}{3})$ and $\mathbf{v_4} = (\frac{\sqrt{3}}{2},\frac{1}{2},\frac{2}{3})$ for the Goldberg tetrahedron.

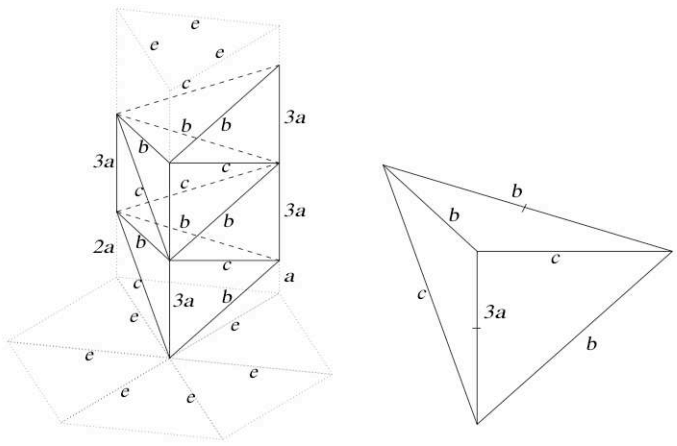

Figure 9: *Goldberg's tetrahedra family. They are parameterized by the prescription of the lengths* a *and* e*, with* $b^2 = a^2 + e^2$ *and* $c^2 = 4a^2 + e^2$*. Left, gathering tetrahedra together fills a prism which basis is the regular triangle of side length* e*.*

We propose now to compare these space filling tetrahedra to the unit regular tetrahedron. To this end, all these tetrahedra are scaled such that their volumes are equal to $\frac{\sqrt{2}}{12}$. The resulting edges lengths for each tetrahedron are specified in Table 1. We notice that the proposed notion of quasi-unit element is only verified for the Sommerville tetrahedra 1 and 2, and the Goldberg tetrahedron. Therefore, there exists space filling tetrahedra that are quasi-units for the metric I_3 in the sense proposed above.

Now, the case of a constant anisotropic metric $\mathcal{M}$ is studied. We consider the pattern around a vertex, *i.e.,* the vertex ball, composed only with the second Sommerville tetrahedron. This pattern exists as it fills $(\mathbb{R}^3, \mathcal{M})$. The vertex ball is mapped back in the natural Euclidean space thanks to the application $\mathcal{M}^{\frac{1}{2}}$. Then, we notice that the non-regularity of the second Sommerville tetrahedron leads necessarily to the creation (in the Euclidean space) of several different anisotropic tetrahedra. However, all these different tetrahedra have the same edges lengths and the same volume in the metric $\mathcal{M}$. Consequently, filling space with only one tetrahedra is possible for all isotropic metrics of the form αI_3, but a set of tetrahedra is required to fill the Euclidean space anisotropically.

Controlling the volume. Unfortunately, the weaker constraint on the edges length can lead to the generation of quasi-unit elements with a null volume. For instance in $(\mathbb{R}^3, I_3)$, the regular tetrahedron with edges length equal to $\sqrt{2}$ is quasi-unit for I_3. However, if one of its vertex is projected orthogonally on the opposite face, then a quasi-unit element of null volume is obtained. Indeed, three edges are of length $\sqrt{2}$ and the three other are of length $\frac{\sqrt{3}}{6} \approx 0.816 \in \left[\frac{1}{\sqrt{2}}, \sqrt{2}\right]$. In consequence, controlling only the edges length is not sufficient, the volume must also be controlled to release the notion of unit element. Practically, this is achieved by prescribing a quality function:

$$Q_{\mathcal{M}}(K) = \frac{36}{3^{\frac{1}{3}}} \frac{|K|_{\mathcal{M}}^{\frac{2}{3}}}{\sum_{i=1}^{6} \ell_{\mathcal{M}}^2(\mathbf{e_i})} \in [0,1]\,. \tag{7}$$

For the perfect regular tetrahedron, whatever its edges length, the quality function is equal to 1. For a null volume tetrahedron, $Q_{\mathcal{M}}$ is 0. The qualities of the space filling tetrahedra are given in Table 1. Notice that $Q_{\mathcal{M}}$ only quantifies the gap to the regular tetrahedron shape.

We deduce the following definition of quasi-unit element, which is also practically used by mesh generators,

Tetrahedron	Coeff.	Edges length						Quality
Sommerville 1	$\sqrt{2}$	0.70	1.22	1.22	1.0	1.0	1.41	0.800
Sommerville 2	$2^{\frac{1}{6}}$	1.12	0.970	0.970	0.970	0.970	1.12	**0.954**
Sommerville 3	$2^{\frac{1}{6}}$	0.970	0.970	0.970	1.12	**1.59**	1.12	0.763
Sommerville 4	$12^{-\frac{1}{3}}2^{\frac{3}{2}}$	**0.691**	1.07	1.07	1.12	1.12	1.23	0.886
Goldberg	$3^{-\frac{1}{2}}2^{\frac{1}{6}}$	0.932	0.990	1.12	1.12	0.990	0.990	0.950

Table 1: *Space filling tetrahedra characteristics. Coeff. is the coefficient that scales the tetrahedron onto a unit volume tetrahedron, i.e.,* $|K| = \sqrt{2}/12$. *The edges length and the tetrahedron quality* Q_{I_3} *given by Formula (7) are provided.*

Definition 5 (Quasi-unit element)**.** *A tetrahedron K defined by its list of edges $(\mathbf{e_i})_{i=1...6}$ is said* **quasi-unit** *for $\mathcal{M}$ if*

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

In our case, $\alpha = 0.8$ is an acceptable value as it enables the Sommerville tetrahedra 1 and 2, and the Goldberg tetrahedron to be generated.

Remark 2. *Instead of considering $Q_{\mathcal{M}}$, the quality function $\frac{1}{Q_{\mathcal{M}}}$ can be considered. As the variation range becomes $[1, \infty[$, the discrimination of bad elements is made easier.*

3 Continuous linear interpolation error

In the previous section, a continuous framework has been introduced to model elements and meshes. Now, we aim at applying this framework in the context of error estimation. Let $(\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$ be a continuous mesh of a domain Ω and let u be a non linear function which is assumed to be only twice continuously differentiable. We seek a well-posed definition of the continuous linear interpolation error $\|u - \pi_{\mathcal{M}} u\|_{\mathbf{L}^1(\Omega)}$ related to a continuous mesh $(\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$ which implies a well-posed definition of a linear continuous interpolate $\pi_{\mathcal{M}} u$. More precisely, we would like the continuous linear interpolation error to be a reliable mathematical model of $\|u - \Pi_h u\|_{\mathbf{L}^1(\Omega_h)}$ where Π_h is defined by a mesh $\mathcal{H}$ of a discretized domain Ω_h which is a unit mesh with respect to $(\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$. This means that considering $\|u - \pi_{\mathcal{M}} u\|_{\mathbf{L}^1(\Omega)}$ is equivalent to consider $\|u - \Pi_h u\|_{\mathbf{L}^1(\Omega_h)}$.

The error analysis is first done locally, *i.e,* in a tangent space of $(\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$ at a given point $\mathbf{a}$. In the tangent space, the continuous mesh $(\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$ reduces to the continuous element $\mathcal{M}(\mathbf{a})$, *i.e.,* the analysis is performed locally at the element level. The function u is approximated by its local quadratic taylor expansion. Indeed, terms of order greater than two can be neglected while studying the linear interpolation error. Then, a global error estimate is derived by taking into account the variation of the continuous mesh and of the function.

3.1 Interpolation error in $\mathbf{L}^1$ norm for quadratic functions

In this section, we consider a **quadratic function** u defined on a domain $\Omega \subset \mathbb{R}^3$. The function is given by its matrix representation:

$$\forall \mathbf{x} \in \Omega, \quad u(\mathbf{x}) = \frac{1}{2}\, {}^t\mathbf{x}\, H\, \mathbf{x},$$

where H is a symmetric matrix representing the Hessian of u. For every symmetric matrix H, $|H|$ denotes the positive symmetric matrix deduced from H by taking the absolute values of its eigenvalues. The function u is linearly interpolated on an tetrahedron K defined by its vertices list: $K = (\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3, \mathbf{v}_4)$. $|K|$ denotes the Euclidean volume of K. We denote by $\Pi_h u$ the linear interpolate of u on K. We can now state the following result:

Proposition 4. *For every quadratic function u, its linear interpolation error in $\mathbf{L}^1$ norm on an element K verifies:*

$$\|u - \Pi_h u\|_{\mathbf{L}^1(K)} \leq \frac{|K|}{40} \sum_{i=1}^{6} {}^t\mathbf{e}_i |H| \mathbf{e}_i,$$

where $\{\mathbf{e}_i\}_{i=1,6}$ is the set of edges of K.
The previous inequality becomes an equality when u is elliptic or parabolic.

Proof. The proof consists in deriving an exact error estimate of the point-wise interpolation error within the element K: $e(\mathbf{x}) = (u - \Pi_h u)(\mathbf{x})$ for $\mathbf{x} \in K$. This error is then integrated over K. To derive e, we use the standard reference element technique. Reference element K_{ref} is defined by its four vertices coordinates:

$$\widehat{\mathbf{v}}_1 = {}^t(0,0,0),\ \widehat{\mathbf{v}}_2 = {}^t(1,0,0),\ \widehat{\mathbf{v}}_3 = {}^t(0,1,0) \text{ and } \widehat{\mathbf{v}}_4 = {}^t(0,0,1).$$

All the computations are done on K_{ref} and the result is then mapped onto the current element K by using the following affine mapping:

$$\mathbf{x} = \mathbf{v}_1 + B_K\, \widehat{\mathbf{x}} \quad \text{with} \quad B_K = (\mathbf{v}_2 - \mathbf{v}_1, \mathbf{v}_3 - \mathbf{v}_1, \mathbf{v}_4 - \mathbf{v}_1), \quad \mathbf{x} \in K, \quad \hat{\mathbf{x}} \in K_{ref}.$$

The matrix B_k is given as a function of the following edges:

$$\mathbf{e}_1 = \mathbf{v}_2 - \mathbf{v}_1, \quad \mathbf{e}_2 = \mathbf{v}_3 - \mathbf{v}_1 \quad \text{and} \quad \mathbf{e}_3 = \mathbf{v}_4 - \mathbf{v}_1,$$

so that $B_K = (\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$. The quadratic function u reads in the frame of K_{ref}:

$$u(\mathbf{x}(\widehat{\mathbf{x}})) = \frac{1}{2}\, {}^t\mathbf{v}_1\, H\, \mathbf{v}_1 + \frac{1}{2}\, {}^t\mathbf{v}_1\, H\, B_K\, \widehat{\mathbf{x}} + \frac{1}{2}\, {}^t\widehat{\mathbf{x}}\, {}^t B_K\, H\, \mathbf{v}_1 + \frac{1}{2}\, {}^t\widehat{\mathbf{x}}\, {}^t B_K\, H\, B_K\, \widehat{\mathbf{x}}.$$

As we consider the linear interpolation, linear and constant terms of $u(\mathbf{x}(\widehat{\mathbf{x}}))$ are exactly interpolated. Without loss of generality, these terms are neglected and only quadratic terms are kept. Indeed, if we consider $\tilde{u}(\mathbf{x}) = \frac{1}{2}\, {}^t\widehat{\mathbf{x}}\, {}^t B_K\, H\, B_K\, \widehat{\mathbf{x}}$, then it comes:

$$e(\mathbf{x}) = (u - \Pi_h u)(\mathbf{x}) = (\tilde{u} - \Pi_h \tilde{u})(\mathbf{x}).$$

We can now consider $\tilde{u}$ instead of u. However for the sake of clarity, we keep on writing u and not $\tilde{u}$. We rewrite u in a matrix form:

$$u(\mathbf{x}(\widehat{\mathbf{x}})) = \frac{1}{2}\, {}^t \begin{pmatrix} \hat{x} \\ \hat{y} \\ \hat{z} \end{pmatrix} \begin{pmatrix} {}^t\mathbf{e}_1 H \mathbf{e}_1 & {}^t\mathbf{e}_1 H \mathbf{e}_2 & {}^t\mathbf{e}_1 H \mathbf{e}_3 \\ {}^t\mathbf{e}_2 H \mathbf{e}_1 & {}^t\mathbf{e}_2 H \mathbf{e}_2 & {}^t\mathbf{e}_2 H \mathbf{e}_3 \\ {}^t\mathbf{e}_3 H \mathbf{e}_1 & {}^t\mathbf{e}_3 H \mathbf{e}_2 & {}^t\mathbf{e}_3 H \mathbf{e}_3 \end{pmatrix} \begin{pmatrix} \hat{x} \\ \hat{y} \\ \hat{z} \end{pmatrix}.$$

u in K_{ref} reads:

$$\begin{aligned} u(\mathbf{x}(\widehat{\mathbf{x}})) = \frac{1}{2}\, [\quad & ({}^t\mathbf{e}_1 H \mathbf{e}_1)\, \hat{x}^2 \quad + \quad ({}^t\mathbf{e}_2 H \mathbf{e}_2)\, \hat{y}^2 \quad + \quad ({}^t\mathbf{e}_3 H \mathbf{e}_3)\, \hat{z}^2 \quad + \\ & 2({}^t\mathbf{e}_1 H \mathbf{e}_2)\, \hat{x}\hat{y} \quad + \quad 2({}^t\mathbf{e}_1 H \mathbf{e}_3)\, \hat{x}\hat{z} \quad + \quad 2({}^t\mathbf{e}_2 H \mathbf{e}_3)\, \hat{y}\hat{z} \quad]. \end{aligned}$$

u is now linearly interpolated on K_{ref}. Its linear interpolate $\Pi_h u(\widehat{\mathbf{x}})$ writes $a\widehat{x} + b\widehat{y} + c\widehat{z} + d$, where coefficients $(a, b, c, d) \in \mathbb{R}^4$ satisfies the following linear system ensuring the $\mathcal{P}_1$ exactness, *i.e.,* $\Pi_h u(\mathbf{v}_i) = u(\mathbf{v}_i)$ for all $i \in [1, 4]$:

$$\begin{cases} \Pi_h u(\mathbf{v}_1) = d = u(\mathbf{x}((0,0,0)) = 0, \\ \Pi_h u(\mathbf{v}_2) = a = u(\mathbf{x}((1,0,0)) = \frac{1}{2}\,({}^t\mathbf{e}_1 H \mathbf{e}_1), \\ \Pi_h u(\mathbf{v}_3) = b = u(\mathbf{x}((0,1,0)) = \frac{1}{2}\,({}^t\mathbf{e}_2 H \mathbf{e}_2), \\ \Pi_h u(\mathbf{v}_4) = c = u(\mathbf{x}((0,0,1)) = \frac{1}{2}\,({}^t\mathbf{e}_3 H \mathbf{e}_3). \end{cases}$$

The solution of the previous linear system gives the final expression of $\Pi_h u$:

$$\Pi_h u(\mathbf{x}(\widehat{\mathbf{x}})) = \frac{1}{2}\left[({}^t\mathbf{e}_1 H \mathbf{e}_1)\,\hat{x} + ({}^t\mathbf{e}_2 H \mathbf{e}_2)\,\hat{y} + ({}^t\mathbf{e}_3 H \mathbf{e}_3)\,\hat{z}\right].$$

The exact point-wise interpolation error $e(\mathbf{x})$ is then given by:

$$\begin{aligned} e(\mathbf{x}(\widehat{\mathbf{x}})) = \frac{1}{2}[\quad & ({}^t\mathbf{e}_1 H \mathbf{e}_1)\,(\hat{x}^2 - \hat{x}) \quad + \quad ({}^t\mathbf{e}_2 H \mathbf{e}_2)\,(\hat{y}^2 - \hat{y}) \quad + \quad ({}^t\mathbf{e}_3 H \mathbf{e}_3)\,(\hat{z}^2 - \hat{z}) \\ + \quad & 2\,({}^t\mathbf{e}_1 H \mathbf{e}_2)\,\hat{x}\hat{y} \quad + \quad 2\,({}^t\mathbf{e}_1 H \mathbf{e}_3)\,\hat{x}\hat{z} \quad + \quad 2\,({}^t\mathbf{e}_2 H \mathbf{e}_3)\,\hat{y}\hat{z} \quad]. \end{aligned}$$

From this equality, estimate in $\mathbf{L}^1$, $\mathbf{L}^2$ or $\mathbf{H}^1$ can be deduced by considering the change of variables given by the mapping B_K. Indeed, for every function F, its integration over K can be computed through its expression in K_{ref}:

$$\int_K F(\mathbf{x})\,\mathrm{d}x\mathrm{d}y\mathrm{d}z = \int_{K_{ref}} F(\mathbf{x}(\widehat{\mathbf{x}}))\,|\det(B_K)|\,\mathrm{d}\hat{x}\mathrm{d}\hat{y}\mathrm{d}\hat{z},$$

As $6\,|K| = \det(B_k)$, previous equality becomes:

$$\int_K F(\mathbf{x})\,\mathrm{d}x\mathrm{d}y\mathrm{d}z = 6|K| \int_{K_{ref}} F(\mathbf{x}(\widehat{\mathbf{x}}))\,\mathrm{d}\hat{x}\mathrm{d}\hat{y}\mathrm{d}\hat{z}.$$

Consequently, the interpolation error in $\mathbf{L}^1$ norm is evaluated by a direct integration of $|e(\mathbf{x})|$. When u is concave or convex, we have: $|(u - \Pi_h u)(\mathbf{x})| = (u - \Pi_h u)(\mathbf{x})$ in the convex case and $|(u - \Pi_h u)(\mathbf{x})| = -(u - \Pi_h u)(\mathbf{x})$ in the concave case. The error reads:

$$\begin{aligned} \|u - \Pi_h u\|_{\mathbf{L}^1(K)} = \frac{|K|}{40} \quad & \Big| \quad 2({}^t\mathbf{e}_1\,H\,\mathbf{e}_2 + {}^t\mathbf{e}_1\,H\,\mathbf{e}_3 + {}^t\mathbf{e}_2\,H\,\mathbf{e}_3) \\ & - \quad 3({}^t\mathbf{e}_1\,H\,\mathbf{e}_1 + {}^t\mathbf{e}_2\,H\,\mathbf{e}_2 + {}^t\mathbf{e}_3\,H\,\mathbf{e}_3)\,\Big|. \end{aligned}$$

Using the conventions of Figure 5, the crossed terms can be expressed only in terms of $\mathbf{e}_i\,H\,\mathbf{e}_i$ for $i = 1, .., 6$:

$$\begin{cases} 2\,{}^t\mathbf{e}_1\,H\,\mathbf{e}_2 = {}^t\mathbf{e}_1\,H\,\mathbf{e}_1 + {}^t\mathbf{e}_2\,H\,\mathbf{e}_2 - {}^t\mathbf{e}_4\,H\,\mathbf{e}_4, \\ 2\,{}^t\mathbf{e}_1\,H\,\mathbf{e}_3 = {}^t\mathbf{e}_1\,H\,\mathbf{e}_1 + {}^t\mathbf{e}_3\,H\,\mathbf{e}_3 - {}^t\mathbf{e}_5\,H\,\mathbf{e}_5, \\ 2\,{}^t\mathbf{e}_2\,H\,\mathbf{e}_3 = {}^t\mathbf{e}_2\,H\,\mathbf{e}_2 + {}^t\mathbf{e}_3\,H\,\mathbf{e}_3 - {}^t\mathbf{e}_6\,H\,\mathbf{e}_6. \end{cases}$$

We deduce:

$$\left|2({}^t\mathbf{e}_1\, H\, \mathbf{e}_2 + {}^t\mathbf{e}_1\, H\, \mathbf{e}_3 + {}^t\mathbf{e}_2\, H\, \mathbf{e}_3) - 3({}^t\mathbf{e}_1\, H\, \mathbf{e}_1 + {}^t\mathbf{e}_2\, H\, \mathbf{e}_2 + {}^t\mathbf{e}_3\, H\, \mathbf{e}_3)\right| = \left|\sum_{i=1}^{6} {}^t\mathbf{e}_i\, H\, \mathbf{e}_i\right|.$$

If u is hyperbolic, the following inequality is used:

$$\frac{1}{2}|\mathbf{x}\, H\, \mathbf{x}| \leq \frac{1}{2}\mathbf{x}\, |H|\, \mathbf{x},$$

to conclude the proof in the general case. □

The same proof applies in 2D. For a quadratic function u, the linear interpolation error on a triangle K is given by:

$$\|u - \Pi_h u\|_{\mathbf{L}^1(K)} \leq \frac{|K|}{24} \sum_{i=1}^{3} {}^t\mathbf{e}_i |H| \mathbf{e}_i.$$

Error estimates in $\mathbf{L}^2$ norm and in $\mathbf{H}^1$ norm can also be derived. We refer to [4, 27] and references therein for their evaluations. These error estimates are classically used to exhibit mesh quality functions and to obtain the best element shape minimizing the interpolation error. In the continuous mesh framework, the interpolation error estimate in $\mathbf{L}^1$ norm is used to prove some exactness properties of the continuous linear interpolate.

Remark 3 (Safety principle)**.** *Even if it is possible to define exactly the linear interpolation error in* $\mathbf{L}^1$ *norm for hyperbolic functions, we do not consider these expressions from a practical point of view. We prefer to consider* $|H|$ *instead of* H *transforming the function into an elliptic or a parabolic one. It comes to in over-estimating the error for hyperbolic functions. Indeed, it seems that we do not take any advantages of considering the null error directions. However, the maximal error directions are those of the gradient of* u*. These directions correspond to the eigenvectors direction of* H*. All these choices are illustrated for the* $2D$ *example where the function* $x^2 - y^2$ *is considered. In that case, the following elliptic approximates can be used:*

$$\begin{array}{rcl} |x^2 - y^2| & \leq & x^2 + y^2 \\ |x^2 - y^2| & \leq & \frac{r}{2}(x-y)^2 + \frac{2}{r}(x+y)^2 \text{ for all } r > 0 \\ |x^2 - y^2| & \leq & \frac{r}{2}(x+y)^2 + \frac{2}{r}(x-y)^2 \text{ for all } r > 0\,. \end{array}$$

The first one is the approximation retained in this paper. The others are elliptic form aligned with null directions of the hyperbolic function. The interesting point to note is that, whatever the considered bound, the unit balls of the three elliptic bounds have the same area. Consequently, even if the safety principle consists in over-estimating the interpolation error in the hyperbolic case, it does not result in over-meshing.

3.2 Linear interpolation on a continuous element

Let $\mathcal{M}$ be a **continuous element** and u be a **quadratic positive function** (see Remark 3). We study the interpolation error for the class of all unit discrete elements in $\mathcal{M}$, given by Definition 2. Figure 4 depicts for a given metric tensor $\mathcal{M}$ some unit elements. We can now state the main result:

Theorem 1. *For all unit element K with respect to $\mathcal{M}$, the interpolation error of u in $\mathbf{L}^1$ norm does not depend on the element shape and is only a function of the Hessian H of u and of the metric $\mathcal{M}$.*

- *In 3D, for all unit elements K in $\mathcal{M}$, the following equality holds:*

$$\|u - \Pi_h u\|_{\mathbf{L}^1(K)} = \frac{\sqrt{2}}{240} \det(\mathcal{M}^{-\frac{1}{2}}) \operatorname{trace}(\mathcal{M}^{-\frac{1}{2}} H \mathcal{M}^{-\frac{1}{2}}). \tag{8}$$

- *In 2D, for all unit elements K in $\mathcal{M}$, the following equality holds:*

$$\|u - \Pi_h u\|_{\mathbf{L}^1(K)} = \frac{\sqrt{3}}{64} \det(\mathcal{M}^{-\frac{1}{2}}) \operatorname{trace}(\mathcal{M}^{-\frac{1}{2}} H \mathcal{M}^{-\frac{1}{2}}).$$

Proof. According to Proposition 4, the interpolation error in $\mathbf{L}^1$ norm of a quadratic positive function u on an element K is:

$$\|u - \Pi_h u\|_{\mathbf{L}^1(K)} = \frac{|K|}{40} \sum_{i=1}^{6} {}^t\mathbf{e}_i H \mathbf{e}_i \,.$$

Then, if K is unit with respect to $\mathcal{M}$, the previous interpolation error is expressed by:

$$\|u - \Pi_h u\|_{\mathbf{L}^1(K)} = \frac{\sqrt{2}}{240} \det(\mathcal{M}^{-\frac{1}{2}}) \operatorname{trace}(\mathcal{M}^{-\frac{1}{2}} H \mathcal{M}^{-\frac{1}{2}}).$$

thanks to the geometric invariants related to the volume, Relation (4), and to the square lengths of the edges, Relation (5). □

We note the strong analogy with classical interpolation error estimate for Lagrange interpolation [8]:

- The term $\det(\mathcal{M}^{-\frac{1}{2}})$ stands for the Jacobian of the affine transformation from the reference element $\hat{K}$ onto the current element K. In our continuous framework, it is the Jacobian of the affine mapping between the reference continuous element unit ball $\mathcal{B}_{I_3}$ onto the current continuous element unit ball $\mathcal{B}_{\mathcal{M}}$.
- The term $\operatorname{trace}(\mathcal{M}^{-\frac{1}{2}} H \mathcal{M}^{-\frac{1}{2}})$ stands for the semi-norm involved in classical error estimates. Generally, this semi-norm contains the anisotropic behavior of the estimate. In the continuous framework, the trace-term gives the alignment correlation between the principal directions of the Hessian H and the principal directions of the metric $\mathcal{M}$.

Relation (8) shows that the infinite set of discrete elements that are unit for a given metric $\mathcal{M}$ achieves the same interpolation error, and moreover, shows that this interpolation error is only expressed with continuous quantities: the continuous element $\mathcal{M}$ and the Hessian of the function u. Consequently, Theorem 1 points out that the metric alone contains enough information to describe completely the linear interpolation error in $\mathbf{L}^1$ norm. In other words, this theorem confirms that the use of metric-based mesh adaptation is particularly well suited to control anisotropically the interpolation error. In the past, this efficiency has been observed practically on real life problems, see for instance [5, 13, 14, 17, 26, 29, 30, 34].

3.3 Continuous linear interpolate

The main difficulty in defining the continuous linear interpolate is to connect a discrete error computed on an element to a local continuous error that is defined point-wise. Indeed, the discrete interpolation error in norm $\mathbf{L}^1$ is integrated on the element K. On the contrary, a continuous mesh is a function $\mathbf{x} \mapsto \mathcal{M}(\mathbf{x})$ defined at each point $\mathbf{x}$ of Ω.

Suppose now that the continuous mesh $(\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$ is varying and that the function u is no more quadratic but only twice continuously differentiable. If Equality (8) of Theorem 1 does not hold anymore, all the terms of the right-hand-side $\mathcal{M}$ and H are well defined continuously. The definition of a continuous interpolate follows up from this consideration.

We denote by u_Q the quadratic approximation of a smooth function u. At point $\mathbf{a}$, u_Q is defined in the vicinity of $\mathbf{a}$ as the truncated second order Taylor expansion of u:

$$\forall \mathbf{x} \in \mathcal{V}(\mathbf{a}), \quad u_Q(\mathbf{a};\mathbf{x}) = u(\mathbf{a}) + \nabla u(\mathbf{a})(\mathbf{x}-\mathbf{a}) + \frac{1}{2}\langle(\mathbf{x}-\mathbf{a}),\, H(\mathbf{a})(\mathbf{x}-\mathbf{a})\rangle.$$

When no confusion is possible, the notation $u_Q(\mathbf{a};\mathbf{x})$ is replaced by $u_Q(\mathbf{x})$. u_Q is a complete quadratic form composed of a constant term, a linear term and finally a quadratic term.

The first result of this section provides an equivalence formula between discrete and continuous views locally around a point $\mathbf{a}$ of the domain. In the vicinity of $\mathbf{a}$, u_Q approximates u and $(\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$ reduces to $\mathcal{M}(\mathbf{a})$ in the tangent space. We can now state the main result:

Theorem 2 (Discrete-continuous equivalence)**.** *Let u be a twice continuously differentiable fonction of a domain Ω and $(\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$ be a continuous mesh of Ω. Then, there exists a unique continuous linear interpolate function $\pi_{\mathcal{M}}$ such that:*

$$\forall \mathbf{a} \in \Omega, \quad |u - \pi_{\mathcal{M}} u|(\mathbf{a}) = 2\,\frac{\|u_Q - \Pi_h u_Q\|_{\mathbf{L}^1(K)}}{|K|},$$

for every K unit element with respect to $\mathcal{M}(\mathbf{a})$.

The proof is given hereafter, we first look at the consequences of this theorem.

Corollary 1. *Let u be a twice continuously differentiable fonction of a domain Ω and $(\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$ be a continuous mesh of Ω. Then, the following continuous linear interpolation estimate holds in 3D:*

$$\begin{aligned}\forall \mathbf{a}\in\Omega\,,\quad |u-\pi_{\mathcal{M}}u|(\mathbf{a}) &= \frac{1}{10}\,\mathrm{trace}\big(\mathcal{M}(\mathbf{a})^{-\frac{1}{2}}\,|H(\mathbf{a})|\,\mathcal{M}(\mathbf{a})^{-\frac{1}{2}}\big)\\ &= \frac{1}{10}\,\Big(\,d(\mathbf{a})^{-\frac{2}{3}}\sum_{i=1}^{3} r_i(\mathbf{a})^{\frac{2}{3}}\;{}^t\mathbf{u}_i(\mathbf{a})\,|H(\mathbf{a})|\,\mathbf{u}_i(\mathbf{a})\,\Big).\end{aligned}$$

In 2D, the estimate is:

$$\begin{aligned}\forall \mathbf{a}\in\Omega\,,\quad |u-\pi_{\mathcal{M}}u|(\mathbf{a}) &= \frac{1}{8}\,\mathrm{trace}\big(\mathcal{M}(\mathbf{a})^{-\frac{1}{2}}\,|H(\mathbf{a})|\,\mathcal{M}(\mathbf{a})^{-\frac{1}{2}}\big)\\ &= \frac{1}{8}\,\Big(\,d(\mathbf{a})^{-1}\sum_{i=1}^{2} r_i(\mathbf{a})\;{}^t\mathbf{u}_i(\mathbf{a})\,|H(\mathbf{a})|\,\mathbf{u}_i(\mathbf{a})\,\Big).\end{aligned}$$

Proof. In 3D, for all unit elements K with respect to $\mathcal{M}(\mathbf{a})$, the error estimation (8) can be rewritten as follow for the quadratic function u_Q approximating u in the vicinity of $\mathbf{a}$:

$$\frac{\|u_Q-\Pi_h u_Q\|_{\mathbf{L}^1(K)}}{|K|} = \frac{1}{20}\,\mathrm{trace}(\mathcal{M}(\mathbf{a})^{-\frac{1}{2}}\,|H(\mathbf{a})|\,\mathcal{M}(\mathbf{a})^{-\frac{1}{2}}).$$

Then, expressing $\mathcal{M}(\mathbf{a})$ as a function of the continuous mesh parameters given by the decomposition of Proposition 3 leads to:

$$\frac{\|u_Q-\Pi_h u_Q\|_{\mathbf{L}^1(K)}}{|K|} = \frac{1}{20}\,\Big(\,d(\mathbf{a})^{-\frac{2}{3}}\sum_{i=1}^{3} r_i(\mathbf{a})^{\frac{2}{3}}\;{}^t\mathbf{u}_i(\mathbf{a})\,|H(\mathbf{a})|\,\mathbf{u}_i(\mathbf{a})\,\Big)$$

where the $(\mathbf{u}_i(a))_{i=1,3}$ stand for the eigenvectors of $\mathcal{M}(\mathbf{a})$. □

This result shows that the continuous point-wise linear interpolation can be decomposed into the product of two terms:

- a first term that control the accuracy, this density term is directly connected to the size of the continuous element,
- a second term that measures alignment deviation between the continuous element orientation and the anisotropy features of the function u.

It is possible to give a geometric interpretation of this estimate. This interpretation illustrates the impact of a continuous mesh on the error iso-values and, consequently, gives some clue toward the control of the error by means of a continuous mesh. The term $\mathcal{M}^{-\frac{1}{2}}(\mathbf{a})\,H(\mathbf{a})\,\mathcal{M}^{-\frac{1}{2}}(\mathbf{a})$ corresponds to the frame change related to the continuous mesh local orientation. Given a symmetric matrix $|H(\mathbf{a})|$, the corresponding quadratic form is:

$$f = \frac{1}{2}\,{}^t\mathbf{x}\,|H(\mathbf{a})|\,\mathbf{x}.$$

The matrix $\mathcal{M}^{-\frac{1}{2}}(\mathbf{a})\,|H(\mathbf{a})|\,\mathcal{M}^{-\frac{1}{2}}(\mathbf{a})$ corresponds to a new quadratic form observed in the space deformed by $\mathcal{M}(\mathbf{a})$. Indeed, if we consider the change of coordinates $\tilde{\mathbf{x}} = \mathcal{M}^{\frac{1}{2}}(\mathbf{a})\,\mathbf{x}$, we define a new quadratic form $\tilde{f}$:

$$\tilde{f}(\tilde{\mathbf{x}}) = {}^t\mathbf{x}\,|H(\mathbf{a})|\,\mathbf{x} = {}^t(\mathcal{M}^{-\frac{1}{2}}(\mathbf{a})\,\tilde{\mathbf{x}})\,|H(\mathbf{a})|\,\mathcal{M}^{-\frac{1}{2}}(\mathbf{a})\,\tilde{\mathbf{x}} = {}^t\tilde{\mathbf{x}}\left(\mathcal{M}^{-\frac{1}{2}}(\mathbf{a})\,|H(\mathbf{a})|\,\mathcal{M}^{-\frac{1}{2}}(\mathbf{a})\right)\tilde{\mathbf{x}}.$$

Iso-values of $\tilde{f}$ and f are different when they are seen in the canonical Euclidean space. In fact, viewing a quadratic form for different continuous meshes changes its iso-values in the Euclidean space, *i.e.,* the real physical space. It is then possible to control the error iso-values by modifying $\mathcal{M}$. This is the main principle of mesh adaptation, but here formulated in a continuous framework. Classical metric-based mesh adaptation consists in finding a metric field that provides isotropic iso-values of the error function, see pionneer work [7].

These results demonstrate that both the interpolation error and the linear interpolate Π_h have continuous counterparts. It is then a step forward in finding a complete analogy between the discrete and the continuous views. From a practical point of view, we deduce the following analogy. Given a unit mesh $\mathcal{H}$ of a domain Ω_h with respect to a continuous mesh $(\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$, the global interpolation error is:

$$\|u - \Pi_h u\|_{\mathbf{L}^1(\Omega_h)} = \sum_{K\in\mathcal{H}} \|u - \Pi_h u\|_{\mathbf{L}^1(K)}. \tag{9}$$

In the continuous case, the discrete summation becomes an integral:

$$\|u - \pi_{\mathcal{M}} u\|_{\mathbf{L}^1(\Omega)} = \int_\Omega |u - \pi_{\mathcal{M}} u|(\mathbf{x})\,\mathrm{d}\mathbf{x}. \tag{10}$$

Note that there is no global guarantee on the continuous interpolation error reliability given by Relation (10). For instance, there is no *a priori* relationship between (9) and (10). The only guarantee is the local equivalence given by Theorem 2. However, the local guarantee becomes global when the mesh is unit with respect to a constant metric tensor (this does not necessary implied that the mesh is uniform) and when the function is quadratic. In this specific case, by neglecting error due to the boundary discretization, we have the equality:

$$2\,\|u - \Pi_h u\|_{\mathbf{L}^1(\Omega_h)} = \|u - \pi_{\mathcal{M}} u\|_{\mathbf{L}^1(\Omega)},$$

for all unit meshes $\mathcal{H}$ with respect to $(\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$. The numerical examples of Section 4 will numerically demonstrate the efficiency of the continuous model. In particular, we will observe that:

- the model is accurate and the equivalence (9)≈(10) is observed for non quadratic functions and non-constant continuous meshes,
- the error due to the fact that mesh generator generates edges with length not stricly equal to one is negligible. In particular, edges length range given in Definition 5 ensures reliable numerical results.

The constant 2 involved in Theorem 2 arrises in the perfect case where the mesh is only composed of perfect unit elements. Note this is only possible in 2D whereas it is always false in 3D due the impossibility to tessel the space only with the regular tetrahedron. In a more practical situation, this constant needs to be evaluated from one unit mesh in order to estimate the deviation between the continuous mesh complexity with respect to the number of vertices of the unit mesh. In other words, the number of vertices N_v of a unit mesh verifies the following function:

$$N_v = C\,N,$$

where N is the continuous mesh complexity and C a constant. The constant C depends on the domain shape, the mesh generator used and the unit mesh resulting quality. Consequently, it reflects how far the current mesh is from the perfect unity. Several examples are given in the numerical examples section and illustrates this relation.

To conclude this section, the proof of Theorem 2 is given. This proof is based on the exact expression of the continuous linear interpolate:

Proposition 5. *The continuous interpolate $\pi_{\mathcal{M}}u$ evaluated at $\mathbf{a} \in \Omega$ for a continuous mesh $(\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$ and for a smooth function u is given by:*

$$\pi_{\mathcal{M}}u(\mathbf{a}) = p^*(0),$$

where p^* is the unique linear polynomial solution of:

$$p^* = \min_{p\in\mathcal{P}^1} \|u_Q - p\|_{\mathbf{L}^2(\mathcal{B}_{\mathcal{M}})},$$

where u_Q is the quadratic model of u at $\mathbf{a}$ and $\mathcal{B}_{\mathcal{M}}$ is the unit ball of $\mathcal{M}$ at $\mathbf{a}$. $\pi_{\mathcal{M}}$ is given by

$$\pi_{\mathcal{M}}u(\mathbf{a}) = u(\mathbf{a}) + \nabla u(\mathbf{a}) + \frac{1}{c_n}\mathrm{trace}(\mathcal{M}^{-\frac{1}{2}}(\mathbf{a})\,H(\mathbf{a})\,\mathcal{M}^{-\frac{1}{2}}(\mathbf{a})),$$

where c_n is a constant that depends only on the space dimension:

$$c_2 = \frac{1}{8} \quad \textit{and} \quad c_3 = \frac{1}{10}.$$

Proof. The quadratic model u_Q of u at point $\mathbf{a}$ defined by:

$$u_Q(\mathbf{a};\mathbf{x}) = u(\mathbf{a}) + \nabla u(\mathbf{a})(\mathbf{x}-\mathbf{a}) + \frac{1}{2}\langle(\mathbf{x}-\mathbf{a}),\, H(\mathbf{a})(\mathbf{x}-\mathbf{a})\rangle,$$

becomes after the translation $\mathbf{x} \mapsto \mathbf{x} + \mathbf{a}$

$$u_Q(\mathbf{x}) = \frac{1}{2}\,{}^t\mathbf{x}\,H(\mathbf{a})\,\mathbf{x} + \nabla u(\mathbf{a})\,\mathbf{x} + u(\mathbf{a}).$$

The linear polynomial p^* is given by:

$$p^* : \mathbf{x} \in \mathcal{B}_{\mathcal{M}} \mapsto {}^t\mathbf{g}\,\mathbf{x} + c,$$

where $c \in \mathbb{R}$ and $\mathbf{g} \in \mathbb{R}^3$. As the space of linear polynomials $\mathcal{P}^1$ and the unit ball $\mathcal{B}_{\mathcal{M}}$ are convex, p^* exists and is unique. We seek for p^* verifying the following condition:

$$\forall p \in \mathcal{P}^1\,, \int_{\mathcal{B}_{\mathcal{M}}} (u_Q(\mathbf{x}) - p^*(\mathbf{x}))\, p(\mathbf{x})\, \mathrm{d}\mathbf{x} \quad = \quad 0.$$

In particular, it is true for the following basis of $\mathcal{P}^1$:

$$\mathbf{x} \mapsto 1, \quad \mathbf{x} \mapsto x_1, \quad \mathbf{x} \mapsto x_2 \ \text{ and } \ \mathbf{x} \mapsto x_3.$$

The previous condition leads to:

$$\begin{array}{rcl} \displaystyle\int_{\mathcal{B}_{\mathcal{M}}} (u_Q(\mathbf{x}) - p^*(\mathbf{x}))\, \mathrm{d}\mathbf{x} & = & 0, \\ \displaystyle\int_{\mathcal{B}_{\mathcal{M}}} (u_Q(\mathbf{x}) - p^*(\mathbf{x}))\, x_i\, \mathrm{d}\mathbf{x} & = & 0, \end{array} \tag{11}$$

for $i = \{1, 2, 3\}$. The initial integration domain $\mathcal{B}_{\mathcal{M}}$ is mapped onto the unit sphere $\mathcal{B}_{I_3}$ by using the following one-to-one change of variables:

$$\begin{array}{ccc} \mathcal{B}_{\mathcal{M}} & \longrightarrow & \mathcal{B}_{I_3} \\ \mathbf{x} & \longmapsto & \mathbf{y} = \mathcal{M}(\mathbf{a})^{\frac{1}{2}}\, \mathbf{x}. \end{array}$$

Functions u_Q and p^* becomes:

$$u_Q(\mathbf{x}) = \tilde{u}_Q(\mathbf{y}) = \tfrac{1}{2}\, {}^t\mathbf{y}\, \mathcal{M}(\mathbf{a})^{-\frac{1}{2}}\, H(\mathbf{a})\, \mathcal{M}(\mathbf{a})^{-\frac{1}{2}} \mathbf{y} + {}^t\mathbf{y}\, \mathcal{M}(\mathbf{a})^{-\frac{1}{2}}\, \nabla u(\mathbf{a}) + u(\mathbf{a}),$$
$$p^*(\mathbf{x}) = \tilde{p}^*(\mathbf{y}) = {}^t\mathbf{y}\, \mathcal{M}(\mathbf{a})^{-\frac{1}{2}}\, \mathbf{g} + c.$$

We now consider the following basis:

$$\mathbf{y} \mapsto 1, \quad \mathbf{y} \mapsto y_1, \quad \mathbf{y} \mapsto y_2 \ \text{ and } \ \mathbf{y} \mapsto y_3,$$

Equations (11) become:

$$\begin{array}{rcl} \displaystyle\int_{\mathcal{B}_{I_3}} (\tilde{u}_Q(\mathbf{y}) - \tilde{p}^*(\mathbf{y}))\, \det(\mathcal{M}(\mathbf{a})^{-\frac{1}{2}})\, \mathrm{d}\mathbf{y} & = & 0, \\ \displaystyle\int_{\mathcal{B}_{I_3}} (\tilde{u}_Q(\mathbf{y}) - \tilde{p}^*(\mathbf{y}))\, y_i\, \det(\mathcal{M}(\mathbf{a})^{-\frac{1}{2}})\, \mathrm{d}\mathbf{y} & = & 0, \end{array}$$

for $i = \{1, 2, 3\}$. By using integration formula of Annexe A, it comes in 3D:

$$\left(\frac{2}{15}\, \mathrm{trace}(\mathcal{M}(\mathbf{a})^{-\frac{1}{2}}\, H(\mathbf{a})\, \mathcal{M}(\mathbf{a})^{-\frac{1}{2}}) + \frac{4}{3}\, (u(\mathbf{a}) - c)\right) \det(\mathcal{M}(\mathbf{a})^{-\frac{1}{2}}) = 0,$$
$$\left(\frac{4}{15}\, \mathcal{M}(\mathbf{a})^{-\frac{1}{2}}\, (\nabla u(\mathbf{a}) - \mathbf{g})\right) \det(\mathcal{M}(\mathbf{a})^{-\frac{1}{2}}) = 0,$$

from which the expression of p^* is deduced:

$$\left\{ \begin{array}{rcl} \mathbf{g} & = & \nabla u(\mathbf{a}), \\ c & = & u(\mathbf{a}) + \dfrac{1}{10}\text{trace}(\mathcal{M}(\mathbf{a})^{-\frac{1}{2}}\, H(\mathbf{a})\, \mathcal{M}(\mathbf{a})^{-\frac{1}{2}}). \end{array} \right.$$

Finally, for the $2D$ case, integration formula of Annexe A provides:

$$\left(\frac{1}{8}\,\text{trace}(\mathcal{M}(\mathbf{a})^{-\frac{1}{2}}\, H(\mathbf{a})\, \mathcal{M}(\mathbf{a})^{-\frac{1}{2}}) + (u(\mathbf{a}) - c)\right) \det(\mathcal{M}(\mathbf{a})^{-\frac{1}{2}}) = 0,$$
$$\left(\frac{1}{4}\,\mathcal{M}(\mathbf{a})^{-\frac{1}{2}}\,(\nabla u(\mathbf{a}) - \mathbf{g})\right) \det(\mathcal{M}(\mathbf{a})^{-\frac{1}{2}}) = 0.$$

and the expression of p^* is given by:

$$\left\{ \begin{array}{rcl} \mathbf{g} & = & \nabla u(\mathbf{a}), \\ c & = & u(\mathbf{a}) + \dfrac{1}{8}\text{trace}(\mathcal{M}(\mathbf{a})^{-\frac{1}{2}}\, H(\mathbf{a})\, \mathcal{M}(\mathbf{a})^{-\frac{1}{2}}). \end{array} \right.$$

□

The proof of Theorem 2 is deduced from the definition of $\pi_{\mathcal{M}} u(\mathbf{a})$.

In the case where the continuous mesh is constant and the function u quadratic, we verify (9)=(10).

Notice that using the $\mathbf{L}^2$ projection of the quadratic model u_Q of u is necessary to ensure the specific equivalence between the discrete linear interpolate Π_h and the continuous linear interpolate $\pi_{\mathcal{M}}$ of Theorem 2. However, the continuous linear interpolate is still well defined if we use the function u instead of u_Q. It seems then possible to define the continuous linear interpolate for far less regular functions. For instance, one may consider only functions that are locally $\mathbf{L}^2$. An open problem is then to find a discrete linear interpolation operator which enables discrete and discontinuous approaches to be linked. Some works involving other interpolation operators have already been considered, we can cite the developments in [15] that derive interpolation error estimate based on the Clément's interpolate [9].

4 Numerical examples

Using the continuous framework introduced above, the continuous interpolation error in $\mathbf{L}^1$ norm of a function over a domain Ω can be computed analytically for any function u and any continuous mesh $(\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$ that are defined by analytic functions. We exemplify this continuous view on examples in 2D and 3D. This calculus does not require any discrete support, e.g. any mesh. To validate the approach, each continuous error is compared to the discrete interpolation error computed on a unit mesh with respect to the continuous one.

Note. In this section, the analytical integrations are computed by using symbolic calculus, when possible, with Maple.

4.1 Continuous interpolation error computation

Embedded continuous meshes. We consider the set of continuous embedded meshes $\mathbf{M}(\alpha) = (\mathcal{M}_\alpha(\mathbf{x}))_{\mathbf{x}\in\Omega_1}$. $\mathbf{M}(\alpha)$ is defined on the square domain $\Omega_1 = [0,1]\times[0,1]$ and is given by:

$$\mathcal{M}_\alpha(x,y) = \alpha \begin{pmatrix} h_1^{-2}(x,y) & 0 \\ 0 & h_2^{-2}(x,y) \end{pmatrix},$$

where $h_1(x,y) = 0.1(x+1) + 0.05(x-1)$ and $h_2(x,y) = 0.2$. The parameter α is used to control the level of accuracy of the mesh. The continuous mesh becomes coarser when α decreases but anisotropic quotients and orientation remain constant. This trend is given by the computation of the complexity $\mathcal{C}(\mathbf{M}(\alpha))$:

$$\mathcal{C}(\mathbf{M}(\alpha)) = N(\alpha) = \iint_{\Omega_1} \frac{1}{h_1 h_2}(x,y)\,\mathrm{d}x\mathrm{d}y = \frac{200}{3}\ln(2)\alpha.$$

The parameter α defines embedded continuous meshes accordingly to Definition 3. The continuous interpolation error on $\mathbf{M}(\alpha)$ is computed for two analytical functions u_1 and u_2:

- u_1 is a quadratic function given by:

$$u_1(x,y) = 6x^2 + 2xy + 4y^2\,,$$

- u_2 is a non quadratic function given by:

$$u_2(x,y) = \mathrm{e}^{(2x^2+y)}.$$

As regards the function u_1, the point-wise continuous interpolation error on $\mathbf{M}(\alpha)$ is:

$$\begin{aligned}(u_1 - \pi_{\mathcal{M}} u_1)(x,y) &= \frac{1}{8}\,\mathrm{trace}(\mathcal{M}_\alpha^{-\frac{1}{2}}(x,y)\,|H_{u_1}|(x,y)\,\mathcal{M}_\alpha^{-\frac{1}{2}}(x,y)) \\ &= \frac{3\,(0.15x+0.05)^2}{2\,\alpha} + \frac{0.04}{\alpha} \\ &= \frac{27\,x^2 + 18\,x + 35}{800\,\alpha}.\end{aligned}$$

The previous expression is then integrated over Ω_1:

$$\iint_{\Omega_1} |u_1 - \pi_{\mathcal{M}} u_1|(x,y)\,\mathrm{d}x\mathrm{d}y \;=\; \frac{53}{800\,\alpha} \;=\; \frac{53}{21}\frac{\ln(2)}{N(\alpha)}.$$

For function u_2, the point-wise continuous interpolation error on $\mathbf{M}(\alpha)$ is:

$$\begin{aligned}(u_2 - \pi_{\mathcal{M}} u_2)(x,y) &= \frac{1}{8}\,\mathrm{trace}(\mathcal{M}_\alpha^{-\frac{1}{2}}(x,y)\,|H_{u_2}|(x,y)\,\mathcal{M}_\alpha^{-\frac{1}{2}}(x,y)) \\ &= \frac{e^{4x^2+y}}{8\,\alpha}\left((0.15x+0.05)^2\,(4+16x^2) + 0.05\right).\end{aligned}$$

By a direct integration over Ω_1, it comes:

$$\iint_{\Omega_1} |u_2 - \pi_{\mathcal{M}} u_2|(x,y)\,\mathrm{d}x\mathrm{d}y \quad \approx \quad \frac{0.2050950191}{\alpha} \quad \approx \quad \frac{13.673\,\ln(2)}{N(\alpha)}.$$

Before comparing the continuous evaluation of interpolation error to the discrete one, some remarks can be made. According to the continuous mesh $\mathbf{M}(\alpha)$, the continuous interpolation for a smooth function u writes:

$$\|u - \pi_{\mathcal{M}} u\|_{\mathbf{L}^1(\Omega_1)} = \frac{C_u}{N(\alpha)},$$

where the constant C_u depends on u and $N(\alpha)$ is the complexity of $\mathbf{M}(\alpha)$. The previous expression gives a quantitative information on the order of convergence of the interpolation error on a sequence of continuous embedded meshes issued from $\mathbf{M}(1)$. Indeed, with a simple analogy with uniform meshes, we have $N(\alpha) = O(h^{-2}(\alpha))$, so that:

$$\|u - \pi_{\mathcal{M}} u\|_{\mathbf{L}^1(\Omega_1)} = C_u\, h^2(\alpha).$$

Consequently, the continuous interpolation error model predicts an order of convergence of two on a sequence of embedded continuous meshes. From the discrete view, it is well known that a uniform refinement leads to a second order of convergence for the linear interpolation error with respect to a smooth function. This fact is also given by the continuous analysis.

Non-embedded continuous mesh. Now, let us give a more complex example in order to demonstrate the ability of the continuous mesh model to predict the order of convergence. In this example, a discrete study of the prediction of the interpolation error is impossible whereas a clear convergence order is exhibited with the continuous mesh model.

We consider the following set of continuous meshes $\mathbf{I}(\alpha) = (\mathcal{M}_\alpha(\mathbf{x}))_{\mathbf{x}\in\Omega_2}$. $\mathbf{I}(\alpha)$ is defined on the square domain $\Omega_2 = [0.5,1] \times [0.5,1]$ and is given by:

$$\mathcal{M}_\alpha(x,y) = F(x,y) \begin{pmatrix} \alpha^2\, h_1^{-2}(x,y) & 0 \\ 0 & \alpha\, h_2^{-2}(x,y) \end{pmatrix} {}^t F(x,y),$$

where

$$F(x,y) = \frac{1}{\sqrt{x^2+y^2}} \begin{pmatrix} x & -y \\ y & x \end{pmatrix},$$

and

$$h_1^{-2}(x,y) = 4(x^2+y^2) \quad \text{and} \quad h_2(x,y)^{-2} = \frac{1}{2\sqrt{x^2+y^2}}.$$

Note that this set is no more embedded accordingly to Definition 3. On the contrary, the continuous meshes are rather non uniformly embedded as one size is scaled by α and the other is scaled by α^2. The equivalent discrete refinement process is no more homogeneous,

i.e., the factor of division of each edge while increasing α depends on the edge coordinates. Consequently, the order of convergence seems unpredictable *a priori* contrary to the embedded case. However, we show that we are able to predict the asymptotic convergence order of the continuous interpolation error for the set spanned by $\mathbf{I}(\alpha)$ by using the continuous analysis. The complexity of $\mathbf{I}(\alpha)$ is given by:

$$\mathcal{C}(\mathbf{I}(\alpha)) = N(\alpha) = \iint_{\Omega_2} \frac{1}{h_1 h_2}(x,y)\,\mathrm{d}x\mathrm{d}y \approx 0.364\,\alpha^{\frac{3}{2}}.$$

We consider the interpolation error of the quadratic function u_3:

$$u_3(x,y) = x^2 + y^2.$$

The point-wise continuous interpolation error on $\mathbf{I}(\alpha)$ is given by:

$$\begin{aligned}(u_3 - \pi_{\mathcal{M}} u_3)(x,y) &= \frac{1}{8}\,\mathrm{trace}(\mathcal{M}_\alpha^{-\frac{1}{2}}(x,y)\,|H_{u_3}|(x,y)\,\mathcal{M}_\alpha^{-\frac{1}{2}}(x,y)) \\ &= \frac{1}{16\,(x^2+y^2)\,\alpha^2} + \frac{\sqrt{x^2+y^2}}{2\,\alpha}.\end{aligned}$$

The previous expression is then integrated over Ω_2, it results:

$$\iint_{\Omega_2} |u_3 - \pi_{\mathcal{M}} u_3|(x,y)\,\mathrm{d}x\mathrm{d}y = \frac{0.133}{N(\alpha)^{\frac{4}{3}}} + \frac{0.014}{N(\alpha)^{\frac{2}{3}}}.$$

The inhomogeneity in the scaling of the sizes leads to two terms with different order of convergence: $\frac{8}{3}$ and $\frac{4}{3}$ for the first and the second terms, respectively. Consequently, the asymptotic order of convergence for the continuous interpolation error of u_3 on $I(\alpha)$ is only $O(N^{-\frac{2}{3}})$ leading to a convergence order of $\frac{4}{3} \approx 1.33$. This is less than the second order reached on the previous set of meshes defined by $\mathbf{M}(\alpha)$. Indeed, for a sufficiently large value of N, the term of order $O(N^{-\frac{4}{3}})$ becomes negligible with respect to the low order term $O(N^{-\frac{2}{3}})$. However, this approximation is only true asymptotically. Practically, the complexity allowing this simplification depends on the constant 0.014 and 0.133. According to Figure 10 (left), as soon as the complexity becomes greater than 1000, the asymptotic order of convergence is fully represented by $\frac{0.014}{N(\alpha)^{\frac{2}{3}}}$. Note that this value is reachable in practice on discrete meshes.

Remark 4. *Depending on the constants involved in the estimation of the order of convergence, a convergence order different from the asymptotic one can be observed. If we suppose that the error for a function u on $\mathbf{I}(\alpha)$ now writes:*

$$\iint_{\Omega_2} |u - \pi_{\mathcal{M}} u|(x,y)\,dx\,dy = \frac{133}{N(\alpha)^{\frac{4}{3}}} + \frac{0.014}{N(\alpha)^{\frac{2}{3}}}.$$

Then, the asymptotic complexity such that the term of order $\frac{4}{3}$ becomes negligible is difficult to reach in practice with discrete meshes (greater than 10^6), see Figure 10 (right). Consequently, the observed order of convergence for acceptable complexities ($\mathcal{C}(\mathbf{I}(\alpha)) \in [1, 10^6]$) is

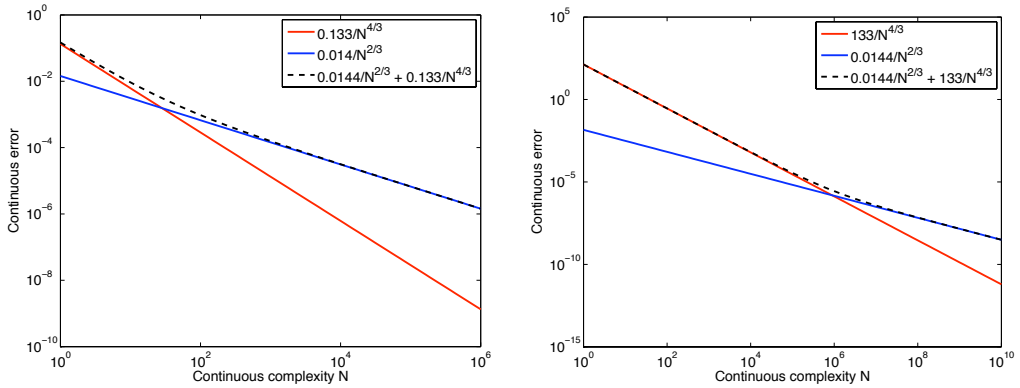

Figure 10: *Left, convergence history obtained for the function u_3 on the set of inhomogeneous continuous meshes spans by $\mathbf{I}(\alpha)$ defined on Ω_2. Right, convergence history for the error function of Remark 4. In that case, the coefficients involved in the error expression lead to observe $O(N^{-\frac{4}{3}})$ as predominant term in the reachable range of complexity $[1, 10^6]$ whereas the asymptotic convergence order is $O(N^{-\frac{2}{3}})$.*

of the order of $O(N^{-\frac{4}{3}})$. This example shows that predicting both the order of convergence and the magnitude of the constants is crucial to get a reliable asymptotic prediction of the interpolation error.

In the next section, the analytic evaluation of the constant C_u along with the convergence order are compared to discrete estimations obtained by generating unit discrete meshes with respect to $\mathbf{M}(\alpha)$ and $\mathbf{I}(\alpha)$ for different values of α.

4.2 Numerical interpolation error computation

Sequences of unit meshes are generated with respect to $\mathbf{M}(\alpha)$ and $\mathbf{I}(\alpha)$. Interpolation errors in $\mathbf{L}^1$ norm are first computed on these meshes and then are compared to continuous error estimations.

Unit meshes. To validate the previous continuous evaluation of the interpolation error, a series of discrete unit meshes with respect to $\mathbf{M}(\alpha) = (\mathcal{M}_\alpha(\mathbf{x}))_{\mathbf{x}\in\Omega}$ is generated. These meshes are considered for $\alpha = \{1, 2, 4, 8, 16, 32\}$. We denote by $(\mathcal{H}_\alpha)_{\alpha\in[1...32]}$ this sequence of discrete meshes. They have been generated using `Yams` [16]. Figure 11 depicts all these meshes. The histograms reporting the meshes edges length and elements quality are given in Table 2. These histograms point out the gap between the generated unit meshes and a perfect unit mesh. We notice that an almost perfect quality is reached for each mesh and that more than 80% of the edges length lie in the range $[\frac{\sqrt{2}}{2}, \sqrt{2}]$ as soon as $\alpha \geq 4$. Unit

$0.20 < L < 0.50$	5	1.94 %	$0.20 < L < 0.50$	1	0.19 %	$0.20 < L < 0.50$	6	0.71 %
$0.50 < L < 0.71$	53	20.54 %	$0.50 < L < 0.71$	130	25.15 %	$0.50 < L < 0.71$	82	9.69 %
$0.71 < L < 0.90$	116	44.96 %	$0.71 < L < 0.90$	204	39.46 %	$0.71 < L < 0.90$	352	41.61 %
$0.90 < L < 1.11$	68	26.36 %	$0.90 < L < 1.11$	133	25.73 %	$0.90 < L < 1.11$	279	32.98 %
$1.11 < L < 1.41$	16	6.20 %	$1.11 < L < 1.41$	49	9.48 %	$1.11 < L < 1.41$	127	15.01 %
$1 < Q < 2$	160	99.38 %	$1 < Q < 2$	328	99.39 %	$1 < Q < 2$	541	99.27 %
$2 < Q < 3$	1	0.62 %	$2 < Q < 3$	2	0.61 %	$2 < Q < 3$	4	0.73 %
$\mathcal{M}_1$			$\mathcal{M}_2$			$\mathcal{M}_4$		
			$0.20 < L < 0.50$	9	0.28 %	$0.20 < L < 0.50$	8	0.12 %
$0.50 < L < 0.71$	281	16.6 %	$0.50 < L < 0.71$	276	8.44 %	$0.50 < L < 0.71$	905	13.69 %
$0.71 < L < 0.90$	564	33.31 %	$0.71 < L < 0.90$	1436	43.89 %	$0.71 < L < 0.90$	2491	37.67 %
$0.90 < L < 1.11$	542	32.01 %	$0.90 < L < 1.11$	1071	32.73 %	$0.90 < L < 1.11$	2081	31.47 %
$1.11 < L < 1.41$	306	18.07 %	$1.11 < L < 1.41$	480	14.67 %	$1.11 < L < 1.41$	1127	17.04 %
$1 < Q < 2$	328	99.18 %	$1 < Q < 2$	328	99.77 %	$1 < Q < 2$	328	99.70 %
$2 < Q < 3$	2	0.82 %	$2 < Q < 3$	2	0.23 %	$2 < Q < 3$	2	0.30 %
$\mathcal{M}_8$			$\mathcal{M}_{16}$			$\mathcal{M}_{32}$		

Table 2: *Quality* $1/Q_{\mathcal{M}}$ *and edges length for the unit meshes with respect to* $\mathbf{M}(\alpha)$ *in* $2D$ *for* $\alpha = \{1, 2, 4, 6, 8, 16, 32\}$.

meshes sequence for the continuous mesh $\mathbf{I}(\alpha)$ are depicted in Figure 12. Similar conclusions arise.

Complexity vs. number of vertices. We first study the correlation between the discrete number of vertices N_v and the continuous complexity N. This preliminary study is necessary in the continuous discrete comparison. Indeed, the continuous interpolation error estimate involves the continuous complexity whereas the discrete number of points is used in the discrete error computation. As regards the sequence of discrete meshes $(\mathcal{H}_\alpha)_\alpha$ with respect to $(\mathbf{M}(\alpha))_\alpha$, the discrete number of nodes N_v is plotted as a function of the complexity N in Figure 13 (top right). This function is linear and the slope evaluated numerically gives:

$$\forall \mathcal{H}_\alpha\,, \quad N_v(\alpha) = 1.54\, N(\alpha).$$

Note that this constant handles the discrepancy between the perfect unit mesh and the generated constrained discrete mesh with respect to the continuous one. Constraints arise from the domain boundary, the used mesh generator and the smoothness of the continuous mesh. This constant is in the theoretical framework equal to 2, see Theorem 2. When the constant is equal to 2, it ensures the following equality between discrete and continuous errors:

$$\frac{C_u}{N} = 2\frac{C_u}{N_v}.$$

In the case of the set of inhomogeneous continuous meshes defined by $\mathbf{I}(\alpha)$, the constant is equal to 3.37.

Discrete interpolation error. The discrete interpolation error of a function u on a mesh $\mathcal{H}_\alpha$ reads:

$$\|u - \Pi_h u\|_{\mathbf{L}^1(\mathcal{H}_\alpha)} = \sum_{K \in \mathcal{H}_\alpha} \|u - \Pi_h u\|_{\mathbf{L}^1(K)},$$

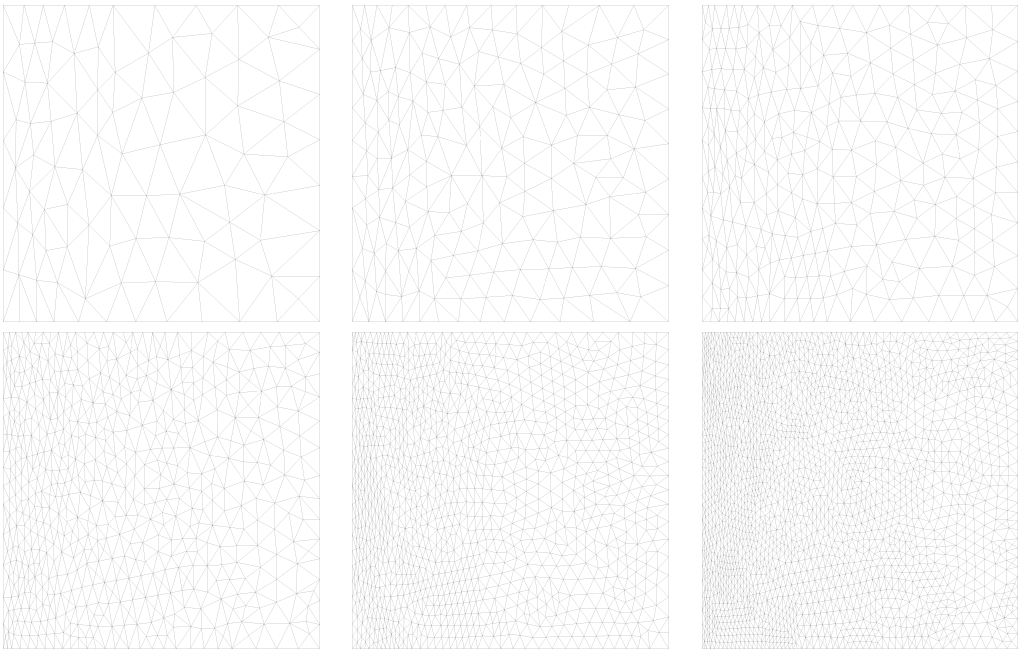

Figure 11: *Unit meshes with respect to* $\mathbf{M}(\alpha)$ *on* Ω_1 *for* $\alpha = \{1, 2, 4, 8, 16, 32\}$ *from left to right and from top to bottom.*

where K stands for an element of $\mathcal{H}_\alpha$. The element-wise interpolation error $\|u - \Pi_h u\|_{\mathbf{L}^1(K)}$ is computed by means of a 5-order Gauss quadrature numerical integration. The discrete interpolation error is compared to the continuous one by considering N_v instead of N in the continuous estimates. Figure 13 plots discrete and continuous interpolation errors for functions u_1 and u_2 on $\mathbf{M}$, and for function u_3 on $\mathbf{I}$. An excellent correlation is obtained between the continuous and the discrete evaluations. The slight differences between the continuous and the discrete expressions come from the practical difficulty to generate a perfect unit mesh. Indeed, mesh generators generate unit meshes in the sense of Definition 5 implying that elements are quasi-unit. In consequence, most of the edges length lie in the range $[\frac{\sqrt{2}}{2}, \sqrt{2}]$. Bounds of the interpolation error can be deduced from this edge length tolerance. For every function u and for every continuous mesh, we have :

$$\|u - \pi_{\mathcal{M}_{min}} u\|_{\mathbf{L}^1} \leq \|u - \pi_{\mathcal{M}} u\|_{\mathbf{L}^1} \leq \|u - \pi_{\mathcal{M}_{max}} u\|_{\mathbf{L}^1},$$

where

$$\mathcal{M}_{min} = 2\,\mathcal{M} \quad \text{and} \quad \mathcal{M}_{max} = \frac{1}{2}\,\mathcal{M}.$$

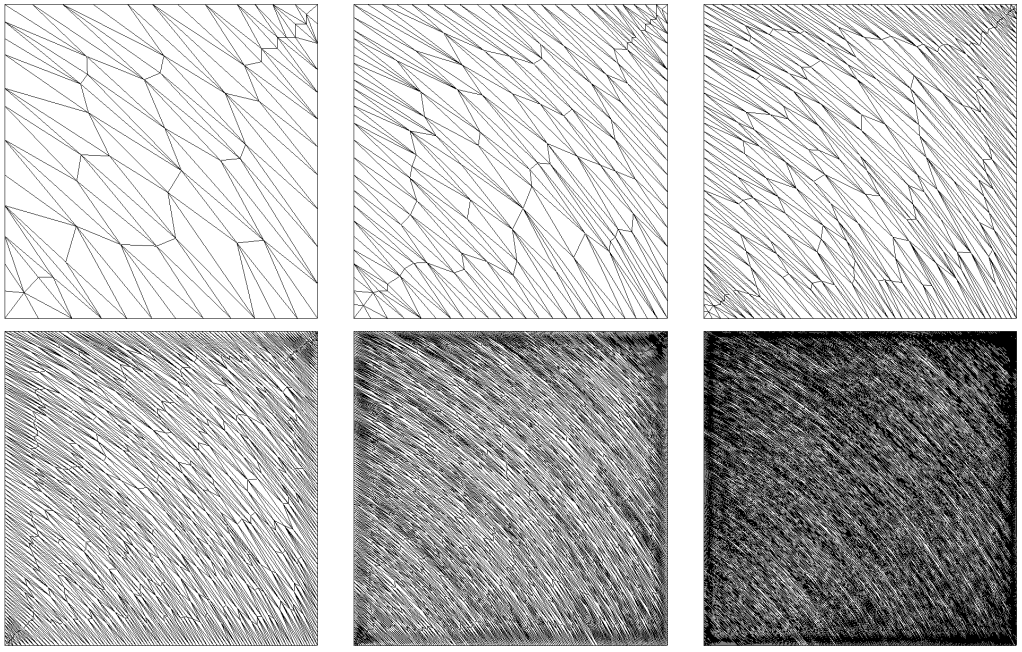

Figure 12: *Unit meshes with respect to* $\mathbf{I}(\alpha)$ *on* Ω_2 *for* $\alpha = \{16, 32, 64, 128, 256, 512\}$ *from left to right and from top to bottom.*

Unit meshes composed of edges length that lie in the acceptable range will verify:

$$\frac{1}{2}\left\|u - \pi_{\mathcal{M}} u\right\|_{\mathbf{L}^1} \leq \left\|u - \Pi_h u\right\|_{\mathbf{L}^1} \leq 2\left\|u - \pi_{\mathcal{M}} u\right\|_{\mathbf{L}^1}.$$

These bounds are plotted in Figure 13. Note that previous bounds impact only the constant evaluation C_u and not the order of convergence of the interpolation error.

4.3 3D examples

In this section, a 3D analytical and a 3D numerical examples are presented.

A fully analytical example. We first consider the set of 3D continuous embedded meshes $\mathbf{M}(\alpha) = (\mathcal{M}_\alpha(\mathbf{x}))_{\mathbf{x}\in\Omega_3}$ defined on the domain $\Omega_3 = [0,1]\times[0,1]\times[0,1]$ which are given by:

$$\mathcal{M}_\alpha(x,y,z) = \alpha \begin{pmatrix} h_1^{-2}(x,y,z) & 0 & 0 \\ 0 & h_2^{-2}(x,y,z) & 0 \\ 0 & 0 & h_3^{-2}(x,y,z) \end{pmatrix},$$

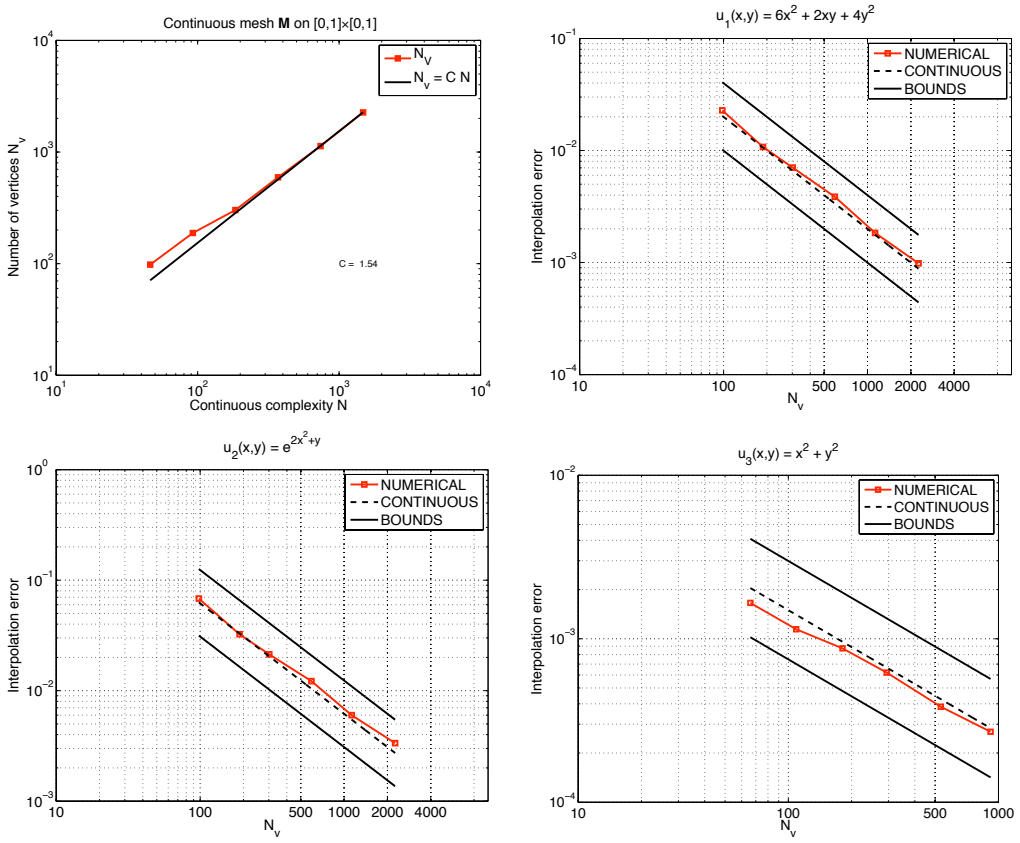

Figure 13: *Top left, comparison between the continuous complexity and the number of vertices for a sequence of discrete meshes* $(\mathcal{H}_\alpha)_\alpha$ *with respect to* $(\mathbf{M}(\alpha))_\alpha$*. Top right and bottom, comparison between continuous interpolation error* $\|u - \pi_{\mathcal{M}} u\|_{\mathbf{L}^1(\Omega)}$ *and discrete interpolation error* $\|u - \Pi_h u\|_{\mathbf{L}^1(\Omega)}$ *for function* u_1 *(top right),* u_2 *(bottom left) on* $\mathbf{M}(\alpha)$ *and* u_3 *(bottom right) on* $\mathbf{I}(\alpha)$*. Black plain lines represent interpolation error when considering* $\mathcal{M}_{max}$ *and* $\mathcal{M}_{min}$*.*

where $h_1(x,y,z) = 0.1(x+1) + 0.05(x-1)$, $h_2(x,y,z) = 0.2$ and $h_3(x,y,z) = 0.2(z+2)$. The parameter α is used to control the level of accuracy of the mesh. The continuous mesh becomes coarser when α decreases. This trend is given by the computation of the complexity $\mathcal{C}(\mathbf{M}(\alpha))$:

$$\mathcal{C}(\mathbf{M}(\alpha)) = N(\alpha) = \iiint_{\Omega_3} \frac{1}{h_1 h_2 h_3}(x,y,z)\,\mathrm{d}x\mathrm{d}y\mathrm{d}z = \frac{1000}{3}\ln(2)(\ln(3) - \ln(2))\alpha^{\frac{3}{2}}.$$

We consider the interpolation error of the function u_4 given by:

$$u_4(x,y,z) = e^{2x+y+z}.$$

The point-wise continuous linear interpolation error is

$$(u_4 - \pi_{\mathcal{M}} u_4)(x,y,z) = \frac{1}{10^5} \frac{e^{2\,x+y+z}\left(441\,x^2 + 798\,x + 2361 + 400\,z^2 + 1600\,z\right)}{\alpha}.$$

Integration over Ω_3 leads to:

$$\begin{aligned} \iiint_{\Omega_3} |u_4 - \pi_{\mathcal{M}} u_4|(x,y,z)\,\mathrm{d}x\mathrm{d}y\mathrm{d}z &= \frac{\left(-2133 - 1772\,e^4 + 7466\,e^2 - 10322\,e^6 + 6761\,e^8\right) e^{-4}}{4.\ 10^5 \alpha} \\ &\approx \frac{0.73}{\alpha} \approx \frac{126.215}{N(\alpha)^{\frac{2}{3}}}. \end{aligned}$$

Sequence of unit meshes with respect to $\mathbf{M}(\alpha)$ for $\alpha \in \{2,4,8,16,32\}$ have been generated using `Yams` [16] and `Gamanic` [18], see Figure 14. In this example, the constant linking the number of vertices and the continuous complexity is 2.07, see Figure 15 (left). Consequently, the perfect theoretical case is almost reached. The comparison between discrete and continuous interpolation error is depicted in Figure 15 (right). We observe an excellent correlation between the continuous model and the numerical computation.

A fully numerical example. The theory is now applied to a more realistic 3D example where the continuous interpolation error results from numerical computations and nor from an analytical evaluation which is not possible in that case. More precisely, the function and the continuous mesh (*i.e.,*the metric field) are linearly interpolated on a discrete mesh. Then, the continuous interpolation error is evaluated by using these discrete data. This case is more representative of the practical use of the continuous mesh model. The aim of this example is to point out that, even with numerical computations, the continuous mesh model turns out to be a reliable model to predict interpolation error.

If u is the function of interest and $\mathbf{I}(\alpha) = (\mathcal{M}_\alpha(\mathbf{x}))_{\mathbf{x}\in\Omega}$ is the continuous mesh, the continuous interpolation error is computed on the discrete mesh $\mathcal{H}_0$ as follow:

$$\int_\Omega |u(\mathbf{x}) - \pi_{\mathcal{M}} u(\mathbf{x})|\,\mathrm{d}\mathbf{x} \approx \frac{1}{10} \sum_{K\in\mathcal{H}_0} \mathrm{trace}(\mathcal{M}_\alpha(K)^{-\frac{1}{2}} |H_u(K)| \mathcal{M}_\alpha(K)^{-\frac{1}{2}})\,|K|, \quad (12)$$

where $\mathcal{M}_\alpha(K)$ and $|H_u(K)|$ are the linear discrete representation on the element K of the continuous mesh and of the Hessian of u, and where a simple 1-order Gauss quadrature numerical integration has been considered.

We first define the continuous mesh $\mathbf{I}(\alpha) = (\mathcal{M}_\alpha(\mathbf{x}))$ on the domain $\Omega = [-1,1] \times [-1,1] \times [-1,1] \subset \mathbb{R}^3$. The analytical expression of $\mathcal{M}_\alpha$ is inherited from the following quadratic form:

$$q(x,y,z) = 2x^2 - yx + 6y^2 + z^2.$$

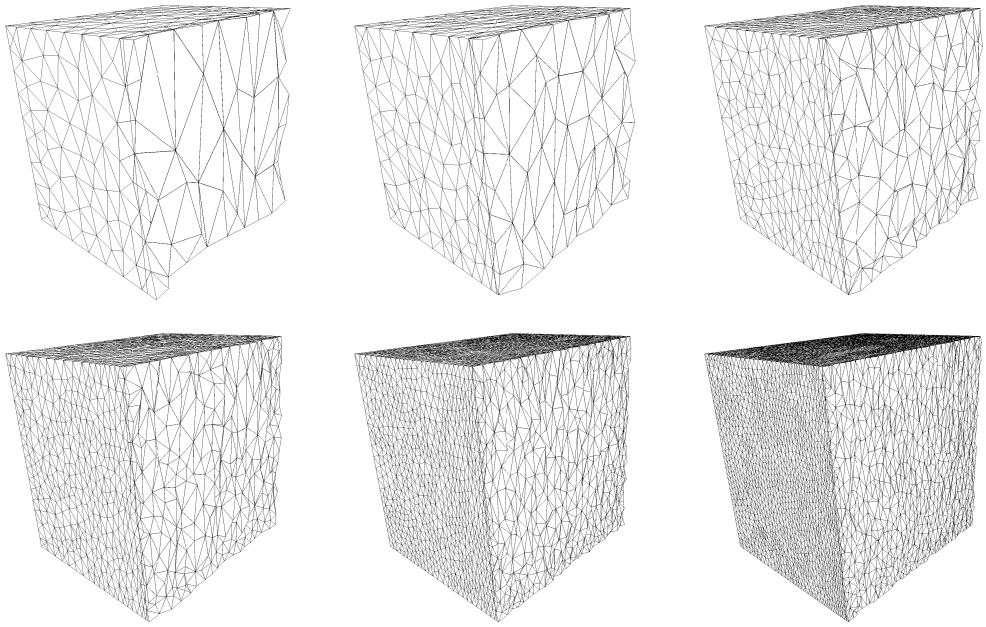

Figure 14: *3D discrete meshes with respect to* $\mathbf{M}(\alpha)$ *for* $\alpha = \{1, 2, 4, 8, 16, 32\}$ *from left to right and from top to bottom.*

The gradient vector is denoted $\mathbf{g}(x, y, z) = {}^t\left(\frac{\partial q}{\partial x}, \frac{\partial q}{\partial y}, \frac{\partial q}{\partial z}\right)$. The local Frenet frame is given by $F = (\mathbf{n}, \mathbf{t}_1, \mathbf{t}_2)$, where the vectors $\mathbf{t}_1, \mathbf{t}_2$ lies in the plane orthogonal to $\mathbf{n} = \mathbf{g}/\|\mathbf{g}\|_2$. We assume that $(\mathbf{t}_1, \mathbf{t}_2)$ is an orthonormal basis of $\mathbb{R}^2$. We also consider the Hessian matrix H_q of q:

$$H_q = \begin{pmatrix} 4 & -1 & 0 \\ -1 & 12 & 0 \\ 0 & 0 & 2 \end{pmatrix}.$$

From this matrix, we define the sub-matrix:

$$\overline{H}_q = \begin{pmatrix} {}^t\mathbf{t}_1 H_q \mathbf{t}_1 & {}^t\mathbf{t}_1 H_q \mathbf{t}_2 \\ {}^t\mathbf{t}_1 H_q \mathbf{t}_2 & {}^t\mathbf{t}_2 H_q \mathbf{t}_2 \end{pmatrix},$$

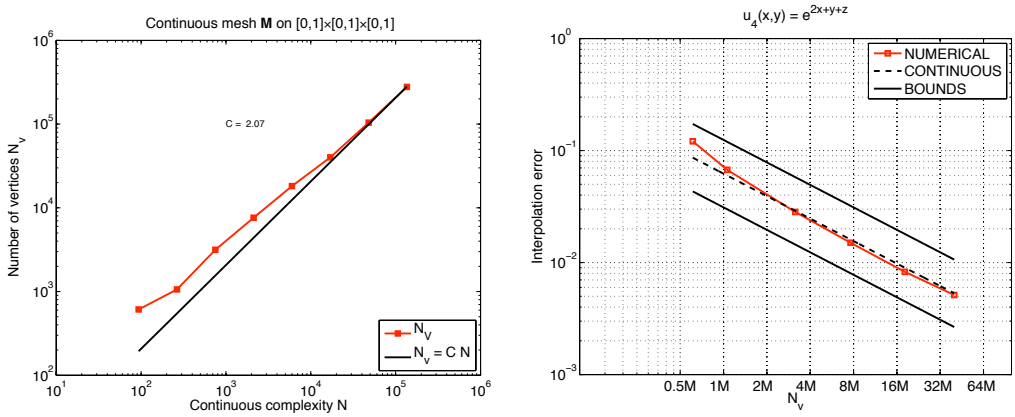

Figure 15: *Left, comparison between the continuous complexity and the number of vertices for a sequence of unit meshes with respect to* $\mathbf{M}(\alpha)$ *on* Ω_3*. Right, comparison between the continuous interpolation error* $\|u_4 - \pi_{\mathcal{M}} u_4\|_{\mathbf{L}^1(\Omega_3)}$ *and the discrete interpolation error* $\|u_4 - \Pi_h u_4\|_{\mathbf{L}^1(\Omega_3)}$*.*

which is the projection of H_q onto the plane defined by $(\mathbf{t}_1, \mathbf{t}_2)$. We denote by $\overline{\lambda_1}$ and $\overline{\lambda_2}$ the eigenvalues of $\overline{H}_q$. The continuous mesh $\mathbf{I}(\alpha)$ parametrized by α is then given by:

$$\mathcal{M}_\alpha = F \begin{pmatrix} \alpha^2 \|\mathbf{g}\|_2^2 & & \\ & \alpha \dfrac{|\overline{\lambda_1}|}{2\|\mathbf{g}\|_2} & \\ & & \alpha \dfrac{|\overline{\lambda_2}|}{2\|\mathbf{g}\|_2} \end{pmatrix} {}^t F.$$

The complexity of $\mathbf{I}(\alpha)$ is:

$$\mathcal{C}(\mathbf{I}(\alpha)) = \frac{\alpha^2}{2} \int_\Omega \sqrt{\|\mathbf{g}\|_2^2 \, |\overline{\lambda_1}| \, |\overline{\lambda_2}|}\,,$$

which can be evaluated numerically. This continuous mesh has the physical feature to follow the iso-surfaces of the function q and to adapt the sizes to the local curvature of the iso-surfaces. The iso-surfaces of q are represented in Figure 16 (left). A sequence of discrete meshes $(\mathcal{H}_\alpha)_\alpha$ for a complexity of

$$[4\,000,\ 8\,000,\ 16\,000,\ 32\,000,\ 64\,000,\ 128\,000]\,,$$

have been generated. Resulting unit meshes are depicted in Figures 17 and 18. In this case, the number of vertices is almost 2.3 times the continuous complexity.

The interpolation error of the function

$$u_5(x, y, z) = x^2 + y^2 + z^2\,.$$

is computed on this sequence of discrete meshes. These discrete errors are compared to continuous interpolation errors evaluated on a fine uniform mesh of constant size $h = 0.004$ using Relation (12). This size corresponds to the minimal size prescribed by the continuous mesh $\mathbf{I}(128\,000)$. The correlation between the continuous and the discrete interpolation error is shown in Figure 16 (right). As expected, the order of convergence is well predicted but the numerical computations induces an over error estimation. Consequently, if the slope (*i.e.,* the convergence order) of the error is well approximated, the perfect fitting due the approximated evaluations of C_u results in a slightly shifted curve.

4.4 Some conclusions

These examples reveal that the interpolation error can be computed continuously without any discrete support. From this point of view, a discrete mesh is simply the projection of the continuous one. Discrepancies between continuous and discrete interpolation errors are due to projection errors. Practically, this gap depends on the mesh generator used and the difficulty to generate a unit mesh.

When the continuous interpolation error cannot be anymore evaluated analytically, we have shown that a numerical evaluation using a discrete mesh still provides accurate predictions of either the convergence order or the error magnitude.

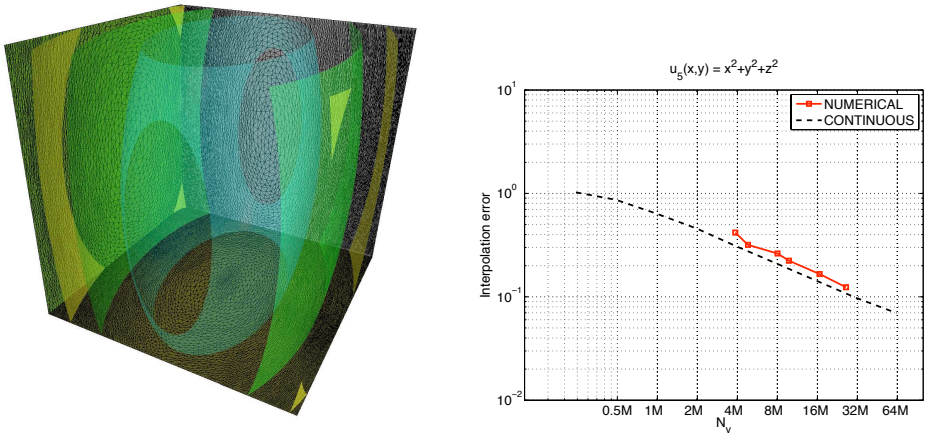

Figure 16: *Left, iso-surfaces of q. Right, comparison between the continuous and the discrete interpolation error for u_5 on $\mathbf{I}(\alpha)$. In that case, the continuous interpolation error is computed numerically by means of the linear approximation of the continuous mesh on a fine uniform discrete mesh.*

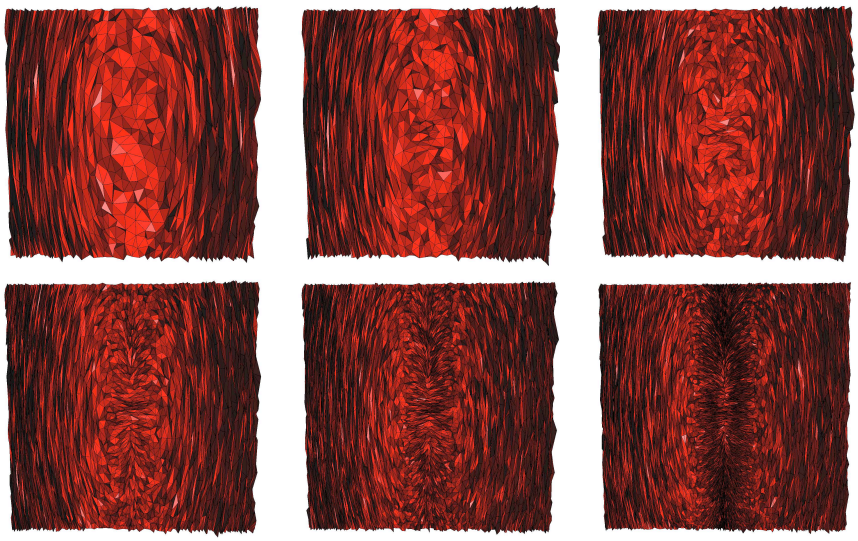

Figure 17: *A cut through the plane $y = 0$ of the discrete meshes with respect to $\mathbf{I}(\alpha)$ of complexity* 4000, 8000, 16000, 32000, 64000, 128000 *from top to bottom, from left to right.*

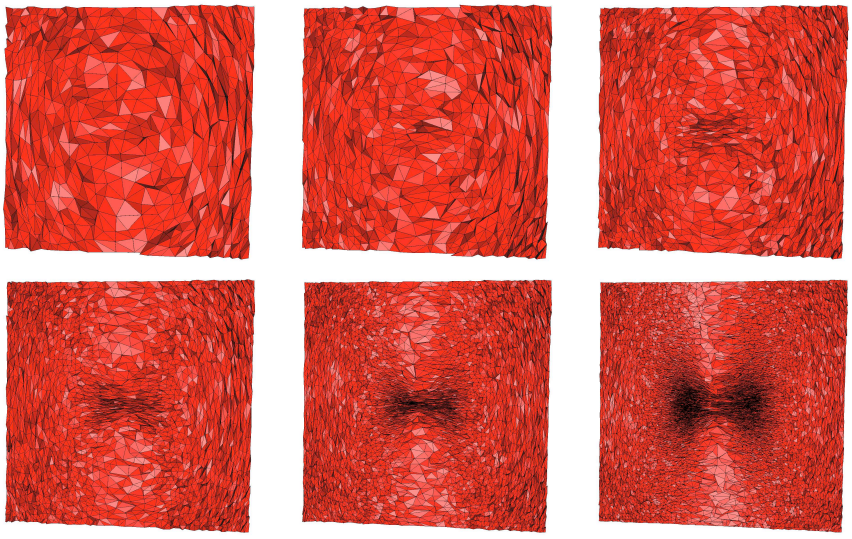

Figure 18: *A cut through the plane $x = 0$ of the discrete meshes with respect to $\mathbf{I}(\alpha)$ of complexity* 4000, 8000, 16000, 32000, 64000, 128000 *from top to bottom, from left to right.*

5 Perspective uses

We give some possible extensions related to the use of continuous meshes. In particular, we emphasize the available mathematical tools and their possible use with the concept of continuous mesh. These extensions are related to the wish of describing the optimal mesh for a given functional.

Optimal mesh adaptation. Given a smooth function u and a continuous mesh $\mathbf{M}$, the linear continuous interpolation error reads, at least asymptotically:

$$\|u - \pi_{\mathcal{M}} u\|_{\mathbf{L}^1(\Omega)} = \frac{C_u}{N^{\frac{k}{n}}},$$

where n is the space dimension, k the expected order of convergence and C_u a constant revealing alignment correlation between $\mathbf{M}$ and the Hessian of u. We have seen previously that the previous relation leads to a second order of convergence on a sequence of embedded continuous meshes having an increasing complexity. In the case of a sequence of non embedded continuous meshes, the order of convergence is related to the specific refinement strategy. Deriving an optimal mesh adaptation strategy consists in increasing the convergence order while reducing the constant C_u. As regards the convergence order, it is fixed by the chosen refinement (embedded or inhomogeneous) along with the order of the interpolation. Consequently, the remaining possibility to reduce the error is to diminish the constant C_u. This problem is rewritten as an optimization problem set up on the space of continuous meshes:

$$\min_{\mathbf{M}} F(\mathbf{M}) \tag{13}$$

In the case of the linear interpolation error, the functional is equal to:

$$F(\mathbf{M}) = \int_{\Omega} |u - \pi_{\mathcal{M}} u|.$$

To avoid the trivial solution giving a null error, a constraint is imposed on the complexity:

$$\mathcal{C}(\mathbf{M}) = N. \tag{14}$$

In the following, two methods to solve (13)-(14) are briefly presented. The first one is based on calculus of variations while the second uses differentiable optimization. Note that these methods are not defined on the space of discrete meshes.

Calculus of variations. This approach consists in deriving analytically a necessary condition for the optimal solution of (13)-(14). For instance, the necessary Euler-Lagrange condition states that there exists λ such that: $\forall \delta\mathbf{M},\ \delta F(\mathbf{M};\delta\mathbf{M}) = \lambda \delta\mathcal{C}(\mathbf{M};\delta\mathbf{M})$. This method needs to be able to exhibit the variation of the functionals F and $\mathcal{C}$ with respect to

$\mathbf{M}$. This can be done, at least formally, on the space of continuous meshes. The variation of F is:

$$\delta F(\mathbf{M};\delta\mathbf{M}) = \lim_{\varepsilon\to 0} \frac{F(\mathbf{M}+\varepsilon\,\delta\mathbf{M}) - F(\mathbf{M})}{\varepsilon}.$$

where

$$\mathbf{M} + \varepsilon\,\delta\mathbf{M} = (\mathcal{M}(\mathbf{x}) + \varepsilon\delta\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\mathbf{\Omega}}.$$

The previous variation is well defined on the space of continuous meshes for $\varepsilon \geq 0$.

Numerical optimization. In the case of a non-linear functional F, deriving an optimality condition can be unpractical. Consequently, a numerical optimization approach is advised. Such an approach is still possible on the set of continuous meshes. Let us consider a classic descent step: from a continuous mesh $\mathbf{M}_i$, a new continuous mesh $\mathbf{M}_{i+1}$ is found:

$$\mathbf{M}_{i+1} = \mathbf{M}_i - \alpha\mathbf{d}_i,$$

where $\mathbf{d}_i$ is a descent direction ensuring that $F(\mathbf{M}_i) > F(\mathbf{M}_{i+1})$. It generally involves derivatives information or approximation of F with respect to $\mathbf{M}_i$. The difficulty in this approach is to ensure that at each step the new continuous mesh $\mathbf{M}_{i+1}$ is well defined, *i.e.,* whatever the point $\mathbf{x} \in \Omega$, $\mathcal{M}_{i+1}(\mathbf{x})$ is a metric tensor. At first, it seems that a lot of additional non linear constraints are necessary to ensure this property. For instance, we may impose all principal determinants of $\mathcal{M}_{i+1}(\mathbf{x})$ to be strictly positive. An elegant solution to avoid additional constraints is the use the log-Euclidean framework [1]. This framework ensures that a classic descent direction step will result in a valid metric tensors field.

6 Conclusion

In this paper, we have proposed a continuous framework to model a mesh and its elements. We have investigated how a discrete mesh can be viewed as the discrete projection of a continuous mesh. Far to be able to give a general continuous mesh model for any discrete meshes, we only consider a model based on the notion of Riemannian metric space. This model corresponds to a wide class of existing 2D and 3D mesh generators [5, 10, 12, 16, 18, 20, 22, 23, 24, 29, 34]. Practically, these mesh generators compute edges length in a Riemannian metric space and aim at generating unit meshes, *i.e.,* meshes with only unit length edges and unit volume elements with respect to the prescribed Riemannian metric field.

The continuous mesh model is mathematically given by rewriting the symmetric definite positive matrix representing the metric tensor at each point of the computational domain. Continuous mesh features as density, orientation and stretching ratio are then exhibited. Geometric invariants that connect discrete elements and continuous elements have been provided. These invariants point out all the information contained in a metric. The discrete projection is then based on the notion of unit mesh. A unit mesh with respect to a continuous

mesh is provided by the use of an adaptive mesh generator. Consequently, the discrete image depends on the method used to generate the mesh.

This model has then been considered to study the interpolation error. We have demonstrated that the interpolation error estimate can be formulated in this continuous framework. The main result states that the linear interpolation error on the class of unit elements is only a function of the associated continuous element and of the considered function. In other words, the notion of continuous mesh handles sufficient information to completely model the interpolation error which is *a priori* only discrete. The numerical examples section has illustrated that, for a given continuous mesh (known analytically), we are able to compute the interpolation error of a given function on this continuous mesh without any discrete support. Consequently, the continuous mesh model demonstrates the well-foundedness of the metric-based mesh adaptation. The numerical examples have also shown that the continuous framework is able to predict perfectly the convergence order of the interpolation error on a sequence of embedded continuous meshes.

The continuous mesh concept is both a theoretical and a practical tool. Theoretically, it can be used to predict order of convergence for complex refinement strategies. The correlation between the continuous and the discrete estimates depends on the correlation between the continuous complexity and the discrete number of nodes. This dependance is linear. Consequently, one unit mesh is sufficient practically to fit the continuous model on the discrete one. This constant contains non linear features induced by the geometry of the domain or the specific used mesh generator. The numerical examples have demonstrated that the continuous mesh concept can also be used numerically and still gives reliable results. This feature is mandatory to deal with and to predict interpolation error on real-life problem involving complex geometries. In that case, symbolic calculus and analytical integration are impractical.

The on-going work is to use this model to propose numerical and analytical mesh adaptation strategies. Indeed, as a mathematical meaning of the continuous interpolation error $u - \pi_{\mathcal{M}} u$ for a given continuous mesh $(\mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$ of a domain Ω and a given function u has been exhibited in this paper, it is then possible to tackle the following global optimization problem:

$$\min_{\mathcal{M}} \left(\int_{\Omega} |u - \pi_{\mathcal{M}} u|^p \right)^{\frac{1}{p}},$$

subject to the constraint $\mathcal{C}(\mathcal{M}) = N$. This problem is simply the continuous equivalent of seeking the optimal discrete mesh minimizing globally the $\mathbf{L}^p$ norm of the interpolation error of the function u. The main interest of adopting a continuous view is to be able to use well defined mathematical tools, such as the calculus of variations or differential optimization that are not available on the space of discrete meshes.

Acknowledgment. The authors are greatly indebted to M. Mehrenberger for his fruitful discussions and advice for proving geometric invariants and to G. Olivier for her careful reading and comments on this work.

A Integration formula on ellipsoids

All the results provided here can be found in [33]. We briefly recall the main results and the way to find them.

Notations. $\mathbf{x}$ is a vector $\mathbb{R}^n$, its coordinates are denoted $\mathbf{x} = (x_1, \ldots, x_n)$. We recall that $\mathcal{B}_{I_n}$ is the unit ball of the identity matrix metric of $\mathbb{R}^n$. It is fully described by the set of points verifying:

$$\{\mathbf{x} = (x_1, \ldots, x_n) \,|\, \|\mathbf{x}\|_2 \leq 1\}.$$

We denote by $V_n(R)$ the volume of a sphere of $\mathbb{R}^n$ with a radius R. With this notation, it comes $V_n(1) = |\mathcal{B}_{I_n}|$. Given an index vector $\mathbf{p} = (p_1, \ldots, p_n)$ and a polynomial function

$$E(\mathbf{x}, \mathbf{p}) = \prod_i (x_i^2)^{p_i},$$

the expression of the following integral can be computed:

$$W_n(\mathbf{p}) = \int_{\mathcal{B}_{I_n}} E(\mathbf{x}, \mathbf{p})\, \mathrm{d}\mathbf{x}. \tag{15}$$

We propose an induction on n to compute (15). We first consider the following one-to-one change of variables:

$$(x_1, \ldots, x_n) \longrightarrow (x_1, \ldots, x_{n-1}, \cos(\theta)),\ 0 \leq \theta \leq \pi,$$

the Jacobian of which is $|\sin(\theta)| = \sin(\theta)$. If we write: $\tilde{\mathbf{p}} = (p_1, \ldots, p_{n-1})$ and $\tilde{\mathbf{x}} = (x_1, \ldots, x_{n-1})$, using the previous change of variables leads to

$$W_n(\mathbf{p}) = \int_0^\pi \left(\int_{V_{n-1}(\sin(\theta))} E(\tilde{\mathbf{x}}, \tilde{\mathbf{p}})\, \mathrm{d}\tilde{\mathbf{x}} \right) (\cos^2(\theta))^{p_n} \sin(\theta)\, \mathrm{d}\theta.$$

The domain $V_{n-1}(\sin(\theta))$ is mapped onto $V_{n-1}(1)$ by simply dividing each coordinate by $\sin(\theta)$. This new change of variables is given by:

$$(x_1, \ldots, x_{n-1}) \longrightarrow \frac{1}{\sin(\theta)} (x_1, \ldots, x_{n-1}).$$

Its Jacobian equals $\sin(\theta)^{n-1}$. It comes:

$$\int_{V_{n-1}(\sin(\theta))} E(\tilde{\mathbf{x}}, \tilde{\mathbf{p}})\, \mathrm{d}\tilde{\mathbf{x}} = \left(\prod_{i=1}^{n-1} (\sin(\theta))^{2p_i+1} \right) W_{n-1}(\tilde{\mathbf{p}}).$$

$W_n(\mathbf{p})$ becomes:

$$W_n(\mathbf{p}) = W_{n-1}(\tilde{\mathbf{p}}) \int_0^\pi \left(\prod_{i=1}^{n-1} (\sin(\theta))^{2p_i+1} \right) (\cos^2(\theta))^{p_n} \sin(\theta)\, \mathrm{d}\theta.$$

We finally get:

$$W_n(\mathbf{p}) = 2\, W_{n-1}(\tilde{\mathbf{p}}) \int_0^{\frac{\pi}{2}} (\sin(\theta))^{2\alpha-1} (\cos(\theta))^{2\beta-1}\, \mathrm{d}\theta,$$

with

$$\alpha = \frac{n+1}{2} + \sum_{i=1}^{n-1} p_i \quad \text{and} \quad \beta = \frac{1}{2} + p_n.$$

The last integral is the Beta function. The following tables sums up examples of integrals both in 2D and 3D. These formula are used to integrate quadratic forms on $V_n(1)$:

n	1	x_i	$x_i x_j$	x_i^2	$\lvert x_i \rvert$	$\lvert x_i x_j \rvert$
2	π	0	0	$\frac{\pi}{4}$	$\frac{4}{3}$	$\frac{1}{2}$
3	$\frac{4}{3}\pi$	0	0	$\frac{4}{15}\pi$	$\frac{\pi}{2}$	$\frac{8}{15}$

Examples. Given a quadratic form f defined on $\mathbb{R}^n$:

$$f(x) = \frac{1}{2}\, {}^t\mathbf{x}\, H\, \mathbf{x} + {}^t\mathbf{x} G + c,$$

where $H \in \mathcal{L}(\mathbb{R}^n, \mathbb{R}^n)$ is a symmetric matrix, $G = (g_i)_{i=1\ldots n} \in \mathbb{R}^n$ a vector and $c \in \mathbb{R}$ a constant, then the following integration results hold in $2D$

$$\begin{aligned}
\int_{\|\mathbf{x}\|_2^2 \le 1} f(\mathbf{x}) &= \pi \left(\frac{1}{8} \operatorname{trace}(H) + c \right), \\
\int_{\|\mathbf{x}\|_2^2 \le 1} f(\mathbf{x})\, x_1 &= \frac{1}{4}\, \pi\, g_1, \\
\int_{\|\mathbf{x}\|_2^2 \le 1} f(\mathbf{x})\, x_2 &= \frac{1}{4}\, \pi\, g_2.
\end{aligned}$$

In 3D, integration formula become:

$$\begin{aligned}
\int_{\|\mathbf{x}\|_2^2 \le 1} f(\mathbf{x}) &= \pi \left(\frac{2}{15} \operatorname{trace}(H) + \frac{4}{3}\, c \right), \\
\int_{\|\mathbf{x}\|_2^2 \le 1} f(\mathbf{x})\, x_1 &= \frac{4}{15}\, \pi\, g_1, \\
\int_{\|\mathbf{x}\|_2^2 \le 1} f(\mathbf{x})\, x_2 &= \frac{4}{15}\, \pi\, g_2, \\
\int_{\|\mathbf{x}\|_2^2 \le 1} f(\mathbf{x})\, x_3 &= \frac{4}{15}\, \pi\, g_3.
\end{aligned}$$

References

[1] V. Arsigny, P. Fillard, X. Pennec, and N. Ayache. Log-Euclidean metrics for fast and simple calculus on diffusion tensors. *Magnetic Resonance in Medicine*, 56(2):411–421, August 2006.

[2] M. Berger. *Geometry 1 & 2*. Springer Verlag, Berlin, 1987.

[3] M. Berger. *A panoramic view of Riemannian geometry*. Springer Verlag, Berlin, 2003.

[4] M. Berzins. Solution-based mesh quality for triangular and tetrahedral meshes. In *Proc. of 6th Meshing Rountable*, 1997.

[5] C. L. Bottasso. Anisotropic mesh adaption by metric-driven optimization. *Int. J. Numer. Meth. Engng*, 60:597–639, 2004.

[6] W. Cao. On the error of linear interpolation and the orientation, aspect ratio, and internal angles of a triangle. *SIAM J. Numer. Anal.*, 43:19–40, 2005.

[7] 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.

[8] P. G. Ciarlet. *The Finite Element Method for Elliptic Problems*. North-Holland, Amsterdam, 1978.

[9] P. Clément. Approximation by finite element functions using local regularization. *Revue Française d'Automatique, Informatique et Recherche Opérationnelle*, R-2:77–84, 1975.

[10] T. Coupez. Génération de maillages et adaptation de maillage par optimisation locale. *Revue Européenne des Éléments Finis*, 9:403–423, 2000.

[11] M. do Carmo. *Differential geometry of curves and surfaces*. Prentice Hall, 1976.

[12] C. Dobrzynski and P. J. Frey. Anisotropic delaunay mesh adaptation for unsteady simulations. In *Proc. of 17th Int. Meshing Rountable*, pages 177–194. Springer, 2008.

[13] J. Dompierre, M.G. Vallet, M. Fortin, Y. Bourgault, and W.G. Habashi. Anisotropic mesh adaptation: towards a solver and user independent cfd. *AIAA Paper*, 97-0861, 1997.

[14] L. Formaggia, S. Micheletti, and S. Perotto. Anisotropic mesh adaptation in computational fluid dynamics: Application to the advection-diffusion-reaction and the Stokes problems. *Appl. Numer. Math.*, 51:511–533, 2004.

[15] L. Formaggia and S. Perotto. New anisotropic a prioiri error estimate. *Numer. Math.*, 89:641–667, 2001.

[16] P. J. Frey. `Yams`, a fully automatic adaptive isotropic surface remeshing procedure. RT-0252, INRIA, 2001.

[17] P. J. Frey and F. Alauzet. Anisotropic mesh adaptation for CFD computations. *Comput. Meth. Appl. Mech. Engrg.*, 194(48-49):5068–5082, 2005.

[18] P.-L. George. `Gamhic3d`, a fully automatic adaptive mesh generation method in three dimensions. Technical Note, INRIA, 2001.

[19] M. Goldberg. Three infinite families of tetrahedral space-fillers. *J. Comb. Theory, Ser. A*, 16(3):348–354, 1974.

[20] F. Hecht. `BAMG`: bidimensional anisotropic mesh generator. Available from http://www-rocq.inria.fr/gamma/cdrom/www/bamg/eng.htm, INRIA-Rocquencourt, France, 1998.

[21] F. Hecht and B. Mohammadi. Mesh adaptation by metric control for multi-scale phenomena and turbulence. *AIAA Paper*, 97-0859, 1997.

[22] W. T. Jones, E. J. Nielsen, and M. A. Park. Validation of 3D adjoint based error estimation and mesh adaptation for sonic boom prediction. *AIAA Paper*, 2006-1150, 2006.

[23] P. Laug and H. Bourochaki. `BL2D-V2`, mailleur bidimensionnel adaptatif. RR-0275, INRIA, 2003.

[24] X. L. Li, M. S. Shephard, and M. W. Beall. 3D anisotropic mesh adaptation by mesh modification. *Comput. Meth. Appl. Mech. Engrg.*, 194(48-49):4915–4950, 2005.

[25] A. Loseille. *Adaptation de maillage 3D anisotrope multi-échelles et ciblé à une fonctionnelle. Application à la prédiction haute-fidélité du bang sonique.* PhD thesis, Université Pierre et Marie Curie, Paris VI, Paris, France, 2008.

[26] A. Loseille, F. Alauzet, A. Dervieux, and P. J. Frey. Achievement of second order mesh convergence for discontinuous flows with adapted unstructured mesh adaptation. *AIAA Paper*, 07-4186, 2007.

[27] E. J. Nadler. *Piecewise linear approximation on triangulations of a planar regions.* PhD thesis, Brown university, Providence, 1985.

[28] D. J. Naylor. Filling space with tetrahedra. *Int. J. Numer. Meth. Engrg.*, 44:1383–1395, 1999.

[29] C. C Pain, A. P. Umpleby, C. R. E. de Oliveira, and A. J. H. Goddard. Tetrahedral mesh optimisation and adaptivity for steady-state and transient finite element calculations. *Comput. Meth. Appl. Mech. Engrg.*, 190:3771–3796, 2001.

[30] E. Schall, D. Leservoisier, A. Dervieux, and B. Koobus. Mesh adaptation as a tool for certified computational aerodynamics. *Int. J. Numer. Meth. Fluids*, 45:179–196, 2004.

[31] M. Senechal. Which tetrahedra fill space? *Mathematics Magazine*, 54:227–243, 1981.

[32] D. Y. M. Sommerville. Space-filling tetrahedra in euclidean space. In *Proc. Edinburgh Math. Soc.*, volume 41, pages 49–57, 1923.

[33] S. Sykora. Quantum theory and the bayesian inference problems. *J. Statistical Phys.*, 11(1):17–27, 1974.

[34] A. Tam, D. Ait-Ali-Yahia, M. P. Robichaud, M. Moore, V. Kozel, and W. G. Habashi. Anisotropic mesh adaptation for 3D flows on structured and unstructured grids. *Comput. Meth. Appl. Mech. Engrg.*, 189:1205–1230, 2000.

[35] M.-G. Vallet. *Génération de maillages éléments finis anisotropes et adaptatifs.* PhD thesis, Université Pierre et Marie Curie, Paris VI, Paris, France, 1992.

Unité de recherche INRIA Rocquencourt
Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)

Unité de recherche INRIA Futurs : Parc Club Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 ORSAY Cedex (France)
Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)
Unité de recherche INRIA Sophia Antipolis : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399