HAL
open science

A numerical study of some Hessian recovery techniques on isotropic and anisotropic meshes

Marco Picasso, Frédéric Alauzet, Houman Borouchaki, Paul-Louis George

To cite this version:

Marco Picasso, Frédéric Alauzet, Houman Borouchaki, Paul-Louis George. A numerical study of some Hessian recovery techniques on isotropic and anisotropic meshes. [Research Report] RR-7202, INRIA. 2010, pp.25. inria-00455799

HAL Id: inria-00455799

https://inria.hal.science/inria-00455799

Submitted on 4 Feb 2013

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

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

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

A numerical study of some Hessian recovery techniques on isotropic and anisotropic meshes

Marco Picasso, Frédéric Alauzet, Houman Borouchaki and Paul-Louis George

N° 7202

Février 2010

Domaine 1

Rapport de recherche

ISSN 0249-6399 ISRN INRIA/RR--7202--FR+ENG

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

INRIA

centre de recherche PARIS - ROCQUENCOURT

A numerical study of some Hessian recovery techniques on isotropic and anisotropic meshes

Marco Picasso*, Frédéric Alauzet, Houman Borouchaki† and Paul-Louis George

Domaine : Mathématiques appliquées, calcul et simulation
Équipe-Projet Gamma

Rapport de recherche n° 7202 — Février 2010 — 22 pages

Abstract: Continuous piecewise linear finite elements are considered to solve a Laplace problem and several numerical methods are investigated to recover second derivatives. Numerical results on 2D and 3D isotropic and anisotropic meshes indicate that the quality of the results is strongly linked to the mesh topology and that no convergence can be insured in general.

Key-words: Finite elements; hessian matrix; anisotropic meshes

* INRIA Gamma and MATHICSE, EPFL, CH-1015 Lausanne
† INRIA Gamma and UTT, 10010 Troyes

Centre de recherche INRIA Paris – Rocquencourt
Domaine de Voluceau, Rocquencourt, BP 105, 78153 Le Chesnay Cedex
Téléphone : +33 1 39 63 55 11 — Télécopie : +33 1 39 63 53 30

Une étude numérique de quelques méthodes pour le calcul des dérivées secondes sur des maillages isotropes et anisotropes

Résumé : On considére une méthode d'éléments finis de degré un par morceaux pour résoudre un problème de Laplace et on cherche à approcher les dérivées secondes de la solution en utilisant plusieurs méthodes disponibles dans la littérature. Une étude numérique en 2D et 3D sur des maillages isotropes et anisotropes montre que la précision des résultats dépend fortement de la topologie des maillages utilisés et que la convergence ne peut être assurée dans le cas général.

Mots-clés : Eléments finis; matrice hessienne; maillages anisotropes

1 Introduction

It is well known from standard finite element textbooks [7] that optimal a priori error estimates for continuous piecewise linear finite elements in the framework of elliptic problems involve the second derivatives of the exact solution:

$$\|\nabla(u - u_h)\|_{L^2(\Omega)} \leq Ch|u|_{H^2(\Omega)}. \tag{1}$$

Hereabove, u is the exact (unknown) solution of the elliptic problem in the domain Ω of $\mathbb{R}^2$ or $\mathbb{R}^3$, h denotes the mesh size, u_h is the finite element approximation, and C is a positive value independent of h and u but depending on the aspect ratio of the finite elements, triangles or tetrahedra for instance.

Recently, adaptive finite elements with large aspect ratio have been used with success for various engineering applications involving sharp boundary layers, see for instance [6, 11, 14, 1, 4, 15, 9] for CFD applications. Most of these anisotropic adaptive algorithms use numerical approximations of the second derivatives of the exact solution as refinement criterion, excepted [21, 20] where a posteriori error estimates and difference of gradients are used instead. From the theoretical point of view, the standard finite element interpolation theory was revisited to account for meshes with large aspect ratio [15, 12, 13] so that estimate (1) can be updated by using the Hessian matrix, the constant C being then independent of the mesh aspect ratio.

The goal of this paper is to study numerical methods to approach the second order derivatives of the exact solution u using the piecewise linear finite element approximation u_h. Intuitively, an approximation of second order derivatives using piecewise linear elements should not converge. Indeed, the L^2 norm of $u - u_h$ is, at most, of order two, the H^1 error of order one, the H^2 error should be of order zero. However, several techniques have been reported to converge [24], at least on special meshes. In this paper, we prove that the L^2 error for second derivatives in weak form is bounded (of order zero) for general unstructured meshes and we construct a special mesh for which no convergence occurs. We then prove that second order derivatives converge whenever superconvergence occurs for gradients, this being the case on special meshes. Numerical experiments on several type of meshes show that a convergence rate cannot be garanteed in general. However, numerical experiments on adapted meshes with large aspect ratio show that reasonable approximations of the second derivatives are obtained.

2 The model problem and numerical approximations

For the sake of clarity, the methods are presented in two space dimensions. However, the extension to three space dimensions is obvious. Numerical results are presented at the end of the paper in two and three space dimensions.

The Laplace problem is considered in a polygonal domain Ω of $\mathbb{R}^2$. Given $f \in L^2(\Omega)$ we are looking for $u : \Omega \to \mathbb{R}$ such that

$$\begin{aligned} -\Delta u &= f \qquad \text{in } \Omega, \\ u &= 0 \qquad \text{on } \partial\Omega. \end{aligned} \tag{2}$$

For any $h > 0$, let $\mathcal{T}_h$ be a conforming mesh of $\overline{\Omega}$ into triangles K with diameter h_K less than h. Let V_h be the usual finite element space of continuous, piecewise linear functions on the triangles of $\mathcal{T}_h$, $V_{h,0}$ the subset of V_h corresponding to the functions zero valued on $\partial\Omega$. The finite element approximation corresponding to (2) consists in finding $u_h \in V_{h,0}$ such that

$$\int_\Omega \nabla u_h \cdot \nabla v_h dx = \int_\Omega f v_h dx \qquad \forall v_h \in V_{h,0}. \tag{3}$$

In this paper, the theoretical results are proved only for isotropic meshes, regular in the sense of [7], but numerical results are presented for isotropic and anisotropic meshes. Our goal is to approximate

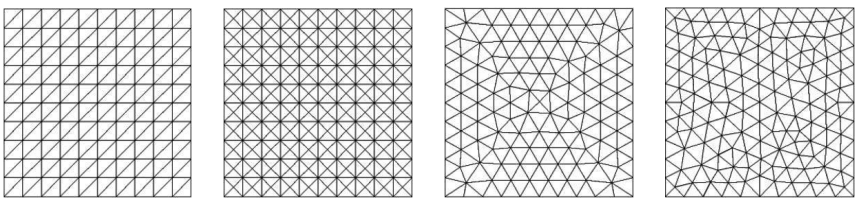
$$w = \frac{\partial^2 u}{\partial x_1^2} \tag{4}$$

using only u_h. The first method stems from the weak formulation of (4) and consist in finding an approximation $w_h \in V_h$ such that

$$\int_\Omega w_h v_h dx = -\int_\Omega \frac{\partial u_h}{\partial x_1}\frac{\partial v_h}{\partial x_1} dx + \int_{\partial\Omega} \frac{\partial u_h}{\partial x_1} v_h n_1 ds \qquad \forall v_h \in V_h, \tag{5}$$

where $n = (n_1, n_2)^T$ is the unit outer normal of $\partial\Omega$. Note that when mass lumping is used, (5) leads to a linear system with diagonal matrix. When mass lumping is not used, the condition number of the linear system is $O(1)$ so that the number of iterations required to solve the linear system by the conjugate gradient method is independent of the mesh size (typically less than ten).

Several types of structured and unstructured meshes will be used throughout the paper in order to perform numerical experiments, see Fig. 1. The structured meshes (regular and criss-cross patterns) have been produced using the `QUAC` and `Q4T` modules of the `MODULEF` software [2]. The frontal unstructured meshes are obtained using the `TRIA` module of `MODULEF` or the `BL2D-V2` software [18], while the Delaunay unstructured meshes are obtained using the `BL2D-V1` software [17]. The interested reader should note that in the frontal modules, the vertices and triangles are added in a frontal manner, starting from the boundary. In the special case when the domain is the square, it can be seen in Fig. 1 that the triangles are equilateral, excepted along the two diagonals. On the other side, in the Delaunay module, an algorithm is used to place the vertices inside the domain (proceeding for instance as in Fig. 7.10 of [15]) and then the vertices are connected using a Delaunay generator.



regular pattern | criss-cross pattern | unstructured frontal | unstructured Delaunay

Figure 1: The four types of structured and unstructured meshes used. Here $\Omega = (0,1)^2$ and the typical meshsize is $h = 0.1$.

Two important remarks can be done concerning method (5) with mass lumping.

Remark 1 *On regular pattern meshes with size h, (5) yields:*

$$w_h(P) = w_h(x_1, x_2) = \frac{u_h(x_1 - h, x_2) - 2u_h(x_1, x_2) + u_h(x_1 + h, x_2)}{h^2}$$

for any internal vertex P of the mesh. This is nothing but the $O(h^2)$ finite difference formula for $\partial^2 u/\partial x_1^2(x_1, x_2)$ provided we replace u_h by $r_h u$, the Lagrange interpolant of u.

Remark 2 *On criss-cross pattern meshes, when $u(x_1, x_2) = x_1(1 - x_1)$ (thus $w(x_1, x_2) = -2$) and when we replace u_h by $r_h u$, then (5) yields either to $w_h(P) = -3/2$ if P is an internal vertex having four neighbours or to $w_h(P) = -9/4$ if P is an internal vertex having eight neighbours.*

We therefore conclude that this method may or may not converge according to the mesh type. In fact we have the following result on general unstructured meshes, which corresponds to the worst case scenario.

Theorem 1 *Assume that Ω and f are such that the solution of (2) is in $H^3(\Omega)$. Let w be defined by (4) and w_h by (5). Then, there is a constant C such that for all $h \leq 1$, we have:*

$$\|w - w_h\|_{L^2(\Omega)} \leq C. \tag{6}$$

Proof. By definition of the L^2 norm, we have:

$$\|w - w_h\|^2_{L^2(\Omega)} = \int_\Omega (w - w_h)(w - v_h)dx + \int_\Omega (w - w_h)(v_h - w_h)dx,$$

for all $v_h \in V_h$. Using (4) and (5) we obtain

$$\begin{aligned}\|w - w_h\|^2_{L^2(\Omega)} &= \int_\Omega (w - w_h)(w - v_h)dx \\ &\quad - \int_\Omega \frac{\partial}{\partial x_1}(u - u_h)\frac{\partial}{\partial x_1}(v_h - w_h)dx + \int_{\partial\Omega} \frac{\partial}{\partial x_1}(u - u_h)(v_h - w_h)n_1 ds,\end{aligned}$$

thus using Cauchy-Schwarz inequality

$$\begin{aligned}\|w - w_h\|^2_{L^2(\Omega)} \leq \|w - w_h\|_{L^2(\Omega)}\|w - v_h\|_{L^2(\Omega)} \\ + \|\nabla(u - u_h)\|_{L^2(\Omega)}\|\nabla(v_h - w_h)\|_{L^2(\Omega)} \\ + \|\nabla(u - u_h)\|_{L^2(\partial\Omega)}\|v_h - w_h\|_{L^2(\partial\Omega)}.\end{aligned}$$

Using inverse estimates, there is a positive constant C independent of h such that

$$h^2\|\nabla(v_h - w_h)\|^2_{L^2(\Omega)} + h\|v_h - w_h\|^2_{L^2(\partial\Omega)} \leq C\|v_h - w_h\|^2_{L^2(\Omega)},$$

thus

$$\begin{aligned}\|w - w_h\|^2_{L^2(\Omega)} &\leq \|w - w_h\|_{L^2(\Omega)}\|w - v_h\|_{L^2(\Omega)} \\ &\quad + C\|v_h - w_h\|_{L^2(\Omega)}\left(\frac{1}{h}\|\nabla(u - u_h)\|_{L^2(\Omega)} + \frac{1}{h^{1/2}}\|\nabla(u - u_h)\|_{L^2(\partial\Omega)}\right).\end{aligned}$$

Finally, writing $v_h - w_h = v_h - w + w - w_h$, using the triangle inequality and Young's inequality yields:

$$\|w - w_h\|^2_{L^2(\Omega)} \leq 3\Bigg(\|w - v_h\|^2_{L^2(\Omega)} + C^2 \left(\frac{1}{h}\|\nabla(u - u_h)\|_{L^2(\Omega)} + \frac{1}{h^{1/2}}\|\nabla(u - u_h)\|_{L^2(\partial\Omega)} \right)^2 \Bigg),$$

for all $v_h \in V_h$. Since $u \in H^3(\Omega)$, $w \in H^1(\Omega)$ and we can choose $v_h = R_h w$ the Clément interpolant [8] and we have

$$\|w - v_h\|_{L^2(\Omega)} \leq Ch\|u\|_{H^3(\Omega)},$$

where C is independent of h and u. Therefore, in order to prove (6), it suffices to prove that

$$\frac{1}{h}\|\nabla(u - u_h)\|_{L^2(\Omega)} + \frac{1}{h^{1/2}}\|\nabla(u - u_h)\|_{L^2(\partial\Omega)}$$

is bounded independently of h. From standard textbooks [7], this is clearly true for the first term, therefore it remains to prove it for the second term. Indeed, by the triangle inequality, we have

$$\|\nabla(u - u_h)\|_{L^2(\partial\Omega)} \leq \|\nabla(u - r_h u)\|_{L^2(\partial\Omega)} + \|\nabla(r_h u - u_h)\|_{L^2(\partial\Omega)}.$$

Using classical interpolation results in one space dimension and a trace inequality, we have

$$\|\nabla(u - u_h)\|_{L^2(\partial\Omega)} \leq Ch|u|_{H^2(\partial\Omega)} \leq \tilde{C}h\|u\|_{H^3(\Omega)},$$

where C and $\tilde{C}$ are independent of h and u. On the other side, an inverse inequality yields

$$\|\nabla(r_h u - u_h)\|_{L^2(\partial\Omega)} \leq \frac{C}{h^{1/2}}\|\nabla(r_h u - u_h)\|_{L^2(\Omega)}.$$

It suffices to use the triangle inequality and classical interpolation results to obtain

$$\|\nabla(r_h u - u_h)\|_{L^2(\partial\Omega)} \leq \frac{C}{h^{1/2}}\Big(\|\nabla(r_h u - u)\|_{L^2(\Omega)} + \|\nabla(u - u_h)\|_{L^2(\Omega)}\Big) \leq \tilde{C}h^{1/2}|u|_{H^2(\partial\Omega)},$$

where C and $\tilde{C}$ are independent of h and u. This yields the desired result. □

Again, we would like to stress that (6) cannot be enhanced for general unstructured meshes since we already know that convergence does not occur on meshes with the criss-cross pattern, see Remark 2. However, on general anisotropic adapted meshes, (6) yields quite accurate results, see Section 3.

Looking at the proof of Theorem 1, we observe that if we could replace $\partial u_h/\partial x_1$ by a better - superconvergent - gradient, then convergence of second

order derivatives could be achieved. In order to define this better gradient, we introduce $\Pi_h : L^2(\Omega) \to V_h$ the L^2 projection operator onto V_h defined for all $g \in L^2(\Omega)$ by $\Pi_h g \in V_h$ and

$$\int_\Omega (\Pi_h g) v_h dx = \int_\Omega g v_h dx \qquad \forall v_h \in V_h. \tag{7}$$

Again, when mass lumping is used, (7) leads to a linear system with diagonal matrix and the following explicit formula holds for any vertex P of the mesh:

$$\Pi_h g(P) = \frac{\displaystyle\sum_{\substack{K \in \mathcal{T}_h \\ P \in K}} \int_K g dx}{\displaystyle\sum_{\substack{K \in \mathcal{T}_h \\ P \in K}} |K|},$$

and then we set

$$\Pi_h g(x) = \sum_{i=1}^{N_h} \Pi_h g(P_i) \varphi_i(x) \qquad \forall x \in \overline{\Omega},$$

where N_h denotes the number of vertices of $\mathcal{T}_h$ and φ_i denotes the usual shape function corresponding to vertex P_i.

We then have the following result.

Theorem 2 *Assume that Ω and f are such that the solution of* (2) *is in $H^3(\Omega)$. Let w be defined by* (4) *and set*

$$w_h = \Pi_h \frac{\partial (Gu_h)_1}{\partial x_1}, \tag{8}$$

where $(Gu_h)_1$ is the first component of a post-processed gradient $Gu_h \in V_h^d$, $d = 2, 3$. Assume that Gu_h superconverges to ∇u up to the boundary, that is there exists two constants $\tilde{C}$ and $0 < \alpha \leq 1$ independent of h such that:

$$\frac{1}{h} \|\nabla u - Gu_h\|_{L^2(\Omega)} + \frac{1}{h^{1/2}} \|\nabla u - Gu_h\|_{L^2(\partial\Omega)} \leq \tilde{C} h^\alpha.$$

Then, there is a constant C such that for all $h \leq 1$, we have:

$$\|w - w_h\|_{L^2(\Omega)} \leq C h^\alpha.$$

Proof. By definition of Π_h, (8) writes

$$\int_\Omega w_h v_h dx = \int_\Omega \frac{\partial (Gu_h)_1}{\partial x_1} v_h dx \qquad \forall v_h \in V_h.$$

Integrating by parts we obtain

$$\int_\Omega w_h v_h dx = -\int_\Omega (Gu_h)_1 \frac{\partial v_h}{\partial x_1} dx + \int_{\partial\Omega} (Gu_h)_1 v_h n_1 ds \qquad \forall v_h \in V_h.$$

We then follow the proof of Theorem 1 by replacing $\partial u_h/\partial x_1$ by $(Gu_h)_1$ and use the superconvergence property to conclude. □

In what follows, we will use two methods to obtain better gradients, namely Zienkiewicz-Zhu post-processing [29, 30] and least-squares [16, 3, 19, 27].

Zienkiewicz-Zhu post-processing corresponds to an L^2 projection of ∇u_h onto V_h:

$$(Gu_h)_1 = \Pi_h \frac{\partial u_h}{\partial x_1}. \tag{9}$$

The theoretical properties of Zienkiewicz-Zhu post-processing have been studied by various authors. In [5, 25], it is proved that superconvergence holds on parallel meshes away from the boundary. However, as remarked in [23], this is wrong on meshes with the chevron pattern. In [26], it is proved that superconvergence occurs on mildly structured meshes that is to say meshes made of near parallelogram triangle pairs. In [28], it is claimed that Delaunay mesh generators satisfy this property but the numerical results reported at the end of this paper indicate that this is not always the case.

We now turn to least squares [16, 3, 19, 27]. The goal is to find, for each vertex P_i of the mesh, a polynomial of degree two that fits in the least squares sense the values of u_h in the neighbourhood of P_i. More precisely, for each vertex P_i of the mesh, we consider the neighbouring vertices. If the number of neighbours is strictly less than five, then the neighbours of the neighbours are added to the list. Let P_i^j, $j = 1, ..., N_i$ denote the obtained list with $N_i \geq 6$ (including vertex P_i itself). We then use the local (x_1, x_2) coordinate system with P_i as origin and denote (ξ_j, η_j) the coordinates of P_i^j in this local system. The method then consists in fitting locally, in the least squares sense, the polynomial of degree two:

$$p(x_1, x_2, P_i) = a_0^i + a_1^i x_1 + a_2^i x_2 + a_3^i x_1^2 + a_4^i x_1 x_2 + a_5^i x_2^2$$

to the values $u_h(P_i^j)$ at the neighbouring vertices. This corresponds to solving the $N_i \times N_i$ linear system

$$A^T A \vec{a} = A^T \vec{b}, \tag{10}$$

with

$$A = \begin{pmatrix} 1 & \xi_1 & \eta_1 & \xi_1^2 & \xi_1\eta_1 & \eta_1^2 \\ 1 & \xi_2 & \eta_2 & \xi_2^2 & \xi_2\eta_2 & \eta_2^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \xi_{N_i} & \eta_{N_i} & \xi_{N_i}^2 & \xi_{N_i}\eta_{N_i} & \eta_{N_i}^2 \end{pmatrix} \quad \vec{a} = \begin{pmatrix} a_0^i \\ a_1^i \\ \vdots \\ a_5^i \end{pmatrix} \quad \vec{b} = \begin{pmatrix} u_h(P_i^1) \\ u_h(P_i^2) \\ \vdots \\ u_h(P_i^{N_i}) \end{pmatrix}.$$

In [19, 27] it is proved that this method preserves polynomials of degree two and that the gradient superconverges on mildly structured meshes. Therefore, an approximation of $w = \partial^2 u/\partial x_1^2$ can be obtained by setting

$$(Gu_h)_1 \in V_h, \qquad (Gu_h)_1(P_i) = \frac{\partial p}{\partial x_1}(0, 0, P_i) = a_1^i \qquad i = 1, ..., N_h, \tag{11}$$

and w_h being still defined by (8). Another approximation simply consists in setting

$$w_h \in V_h, \qquad w_h(P_i) = \frac{\partial^2 p}{\partial x_1^2}(0, 0, P_i) = 2a_3^i \qquad i = 1, ..., N_h. \tag{12}$$

3 Numerical results

3.1 2D computations

The following methods have been tested to approach $w = \partial^2 u/\partial x_1^2$ using $w_h \in V_h$ defined by:

- method 1 : (8) and (9),
- method 2 : (8) and (11),
- method 3 : (12),
- method 4 : the weak form (5).

The notations used are the following:

$$
\begin{aligned}
&De = \|\nabla(u-u_h)\|_{L^2(\Omega)}, \\
&De^{ZZ} = \left\| \frac{\partial u}{\partial x_1} - (Gu_h)_1 \right\|_{L^2(\Omega)} && \text{where } (Gu_h)_1 \text{ is defined by (9)}, \\
&D^2e^{ZZ} = \|w-w_h\|_{L^2(\Omega)} && \text{where } w_h \text{ is defined by (8) and (9)}, \\
&De^{LS} = \left\| \frac{\partial u}{\partial x_1} - (Gu_h)_1 \right\|_{L^2(\Omega)} && \text{where } (Gu_h)_1 \text{ is defined by (11)}, \\
&D^2e^{LS1} = \|w-w_h\|_{L^2(\Omega)} && \text{where } w_h \text{ is defined by (12)}, \\
&D^2e^{LS2} = \|w-w_h\|_{L^2(\Omega)} && \text{where } w_h \text{ is defined by (8) and (11)}, \\
&D^2e^{WF} = \|w-w_h\|_{L^2(\Omega)} && \text{where } w_h \text{ is defined by (5)}.
\end{aligned}
$$

All the results reported hereafter correspond to mass lumping. Numerical computations have been performed without mass lumping, the results being not reported since the trends remain the same.

We set $\Omega = (0,1)^2$ and choose f so that the exact solution u is given by $u(x_1,x_2) = sin(2\pi x_1)$. Zero Dirichlet boundary conditions apply on the vertical boundaries of Ω whereas zero Neumann boundary conditions apply on the horizontal boundaries. In Table 1 we report numerical experiments for regular pattern meshes, see Fig. 1. As expected, it is observed that $\|\nabla(u-u_h)\|_{L^2(\Omega)}$ is $O(h)$ (col 2). Concerning the convergence of gradients, Zienkiewicz-Zhu superconverges at rate $\simeq O(h^{1.5})$ (col. 3), while least squares superconverges at rate $O(h^2)$ (col. 5). Concerning second order derivatives, methods 1, 2 and 3 (col. 4, 6 and 7) converge at rate $\simeq O(h^{1.5})$ which is better than the prediction of Theorem 2. Finally, method 4 converges at rate $O(h^2)$ which was anticipated in Remark 1. We have also performed numerical expriments with a mildly structured mesh [26] obtained by adding h^2 to the coordinates of each internal vertex of the mesh. Numerical results are not reported here since they are very similar to those of Table 1.

			meth. 1		meth. 3	meth. 2	meth. 4
h	De	De^{ZZ}	D^2e^{ZZ}	De^{LS}	D^2e^{LS1}	D^2e^{LS2}	D^2e^{WF}
0.1	0.81	0.32	4.38	0.35	6.13	7.24	1.12
0.05	0.40	0.087	1.46	0.086	2.28	2.73	0.28
0.025	0.20	0.024	0.48	0.021	0.81	0.97	0.07
0.0125	0.10	0.0072	0.16	0.0050	0.29	0.34	0.018
0.00625	0.050	0.0022	0.056	0.0012	0.10	0.12	0.0044
0.003125	0.025	0.00073	0.019	0.00031	0.036	0.042	0.0011

Table 1: Regular pattern.

The criss-cross pattern of Fig. 1 is now considered, the results being displayed in Table 2. Here, $O(h^2)$ convergence of Zienkiewicz-Zhu and least squares gradients is observed, $\simeq O(h^{1.5})$ convergence is observed for methods 1 and 2, while no convergence is observed for methods 3 and 4, this being predicted for method 4 in Remark 2. The results pertaining to method 4 with $h = 0.05$ are reported in Figs. 2 and 3.

			meth. 1		meth. 3	meth. 2	meth. 4
h	De	De^{ZZ}	D^2e^{ZZ}	De^{LS}	D^2e^{LS1}	D^2e^{LS2}	D^2e^{WF}
0.1	0.66	0.19	2.31	0.28	3.84	4.02	4.85
0.05	0.33	0.048	0.71	0.077	1.85	1.36	4.90
0.025	0.16	0.011	0.22	0.020	1.31	0.44	4.92
0.0125	0.082	0.0030	0.072	0.0051	1.21	0.15	4.93
0.00625	0.041	0.00075	0.024	0.0013	1.19	0.051	4.93
0.003125	0.021	0.00019	0.0084	0.00032	1.19	0.018	4.93

Table 2: Criss-cross pattern.

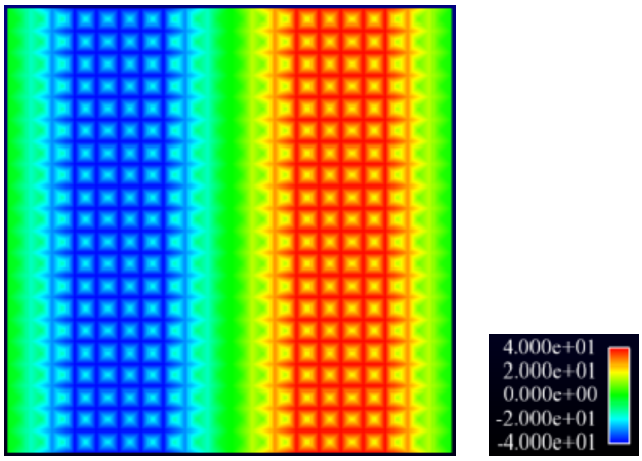


Figure 2: Criss-cross pattern $h = 0.05$. Isovalues of w_h (method 4).

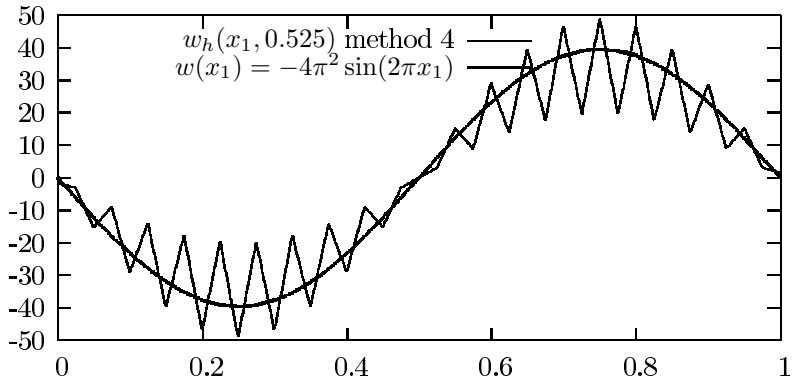


Figure 3: Criss-cross pattern $h = 0.05$. Plot of w and w_h (method 4) against x_1 at $x_2 = 0.525$.

We now turn to the unstructured frontal pattern of Fig. 1, the results being reported in Table 3. All the methods converge, but not as fast as for regular pattern meshes. On the other side, when using unstructured Delaunay meshes, see Fig. 1, the results of Table 4 are not very convincing since only method 2 seems to converge. We believe that the fact that better results are obtained when using the frontal pattern rather than the Delaunay pattern is due to the quality of the triangles. Indeed, the frontal method produces triangles which are equilateral, excepted along the two diagonals of the unit square, see Fig. 1. The results corresponding to unstructured Delaunay meshes with methods 2 and 4 with $h = 0.003125$ are reported in Fig. 4. On unstructured Delaunay meshes, there is no superconvergence for Zienkiewicz-Zhu gradients which implies, from the theoretical results of [26] that the meshes are not mildly structured (if they were, superconvergence of the gradients should occur and therefore second derivatives should converge). In [28], it is claimed that Delaunay mesh generators produce mildly structured meshes, but this is not always the case. However, it should be noted the L^2 norm of $\nabla u - Gu_h$ is always much less than $\nabla(u - u_h)$, therefore $\nabla u_h - Gu_h$ can still be used as a reliable error estimator in an adaptive algorithm. On unstructured Delaunay meshes, method 4, the weak form (5), is the worst one. This can be explained looking at Fig. 5 where w_h is plotted when $h = 0.0125$. The accuracy of w_h is poor around vertices having four neighbours, which corresponds to a local criss-cross pattern for which the method does not converge, according to Remark 2.

			meth. 1		meth. 3	meth. 2	meth. 4
h	De	De^{ZZ}	D^2e^{ZZ}	De^{LS}	D^2e^{LS1}	D^2e^{LS2}	D^2e^{WF}
0.1	0.76	0.29	4.00	0.31	5.45	6.50	1.45
0.05	0.36	0.067	1.27	0.073	1.96	2.34	0.50
0.025	0.18	0.017	0.44	0.017	0.70	0.80	0.29
0.0125	0.088	0.0049	0.20	0.0039	0.27	0.28	0.20
0.0625	0.044	0.0015	0.12	0.00096	0.12	0.097	0.14

Table 3: Unstructured frontal pattern.

			meth. 1		meth. 3	meth. 2	meth. 4
h	De	De^{ZZ}	D^2e^{ZZ}	De^{LS}	D^2e^{LS1}	D^2e^{LS2}	D^2e^{WF}
0.1	0.67	0.23	3.25	0.24	4.19	5.21	1.08
0.05	0.35	0.071	1.28	0.067	1.79	2.10	0.75
0.025	0.18	0.024	0.72	0.018	0.86	0.84	0.74
0.0125	0.089	0.01	0.63	0.0045	0.57	0.32	0.82
0.00625	0.044	0.0047	0.61	0.0014	0.51	0.16	0.79
0.003125	0.022	0.0023	0.61	0.00057	0.51	0.13	0.79

Table 4: Unstructured Delaunay pattern.

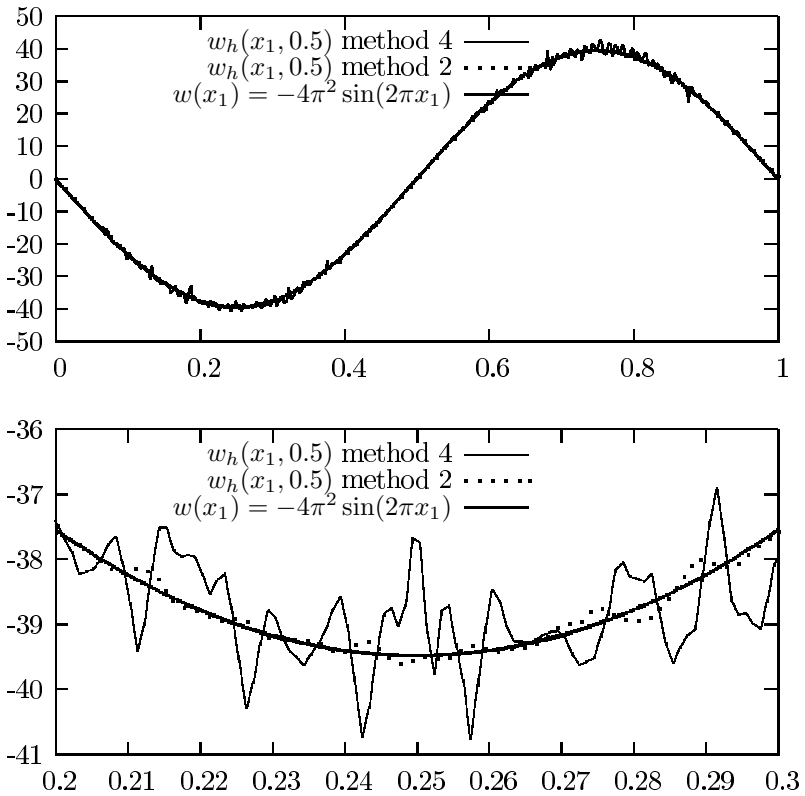


Figure 4: Unstructured Delaunay pattern $h = 0.003125$. Plot of w and w_h (methods 2 and 4) against x_1 at $x_2 = 0.5$.

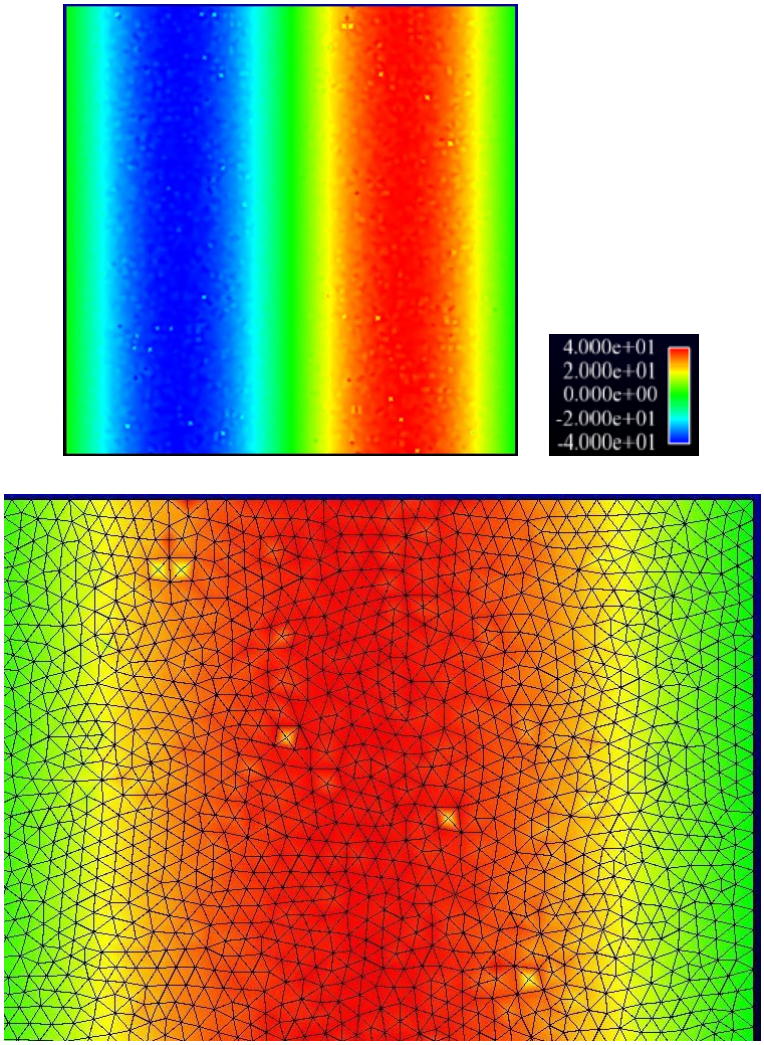


Figure 5: Unstructured Delaunay pattern $h = 0.0125$. Isovalues of w_h (method 4). Bottom: zoom with mesh.

We now study two variants of the least squares method in which (variant 1) for each vertex of the mesh, the neighbours of the neighbours are always taken into account. Another variant (variant 2) is to weight the equations corresponding to the neighbours and those corresponding to the neighbours of the neighbours differently, thus solving

$$A^T D^2 A\vec{a} = A^T D^2 \vec{b},$$

instead of (10). Hereabove, D is a diagonal matrix with entry 1 if the index corresponds to a vertex neighbour and $1/2$ if it corresponds to a neighbour of a neighbour. The results are reported in Table 5 for the unstructured Delaunay pattern and show that these two variants of method 2 lead to the best results,

namely order one convergence of the second derivatives. Such results confirm those reported in [24].

	original least squares		variant 1		variant 2	
h	D^2e^{LS1}	D^2e^{LS2}	D^2e^{LS1}	D^2e^{LS2}	D^2e^{LS1}	D^2e^{LS2}
0.1	4.19	5.21	5.91	6.55	5.68	6.04
0.05	1.79	2.10	2.57	2.68	2.45	2.47
0.025	0.86	0.84	1.06	1.05	1.01	0.97
0.0125	0.57	0.32	0.42	0.38	0.40	0.36
0.00625	0.51	0.16	0.21	0.15	0.19	0.15
0.003125	0.51	0.13	0.14	0.070	0.12	0.077

Table 5: Unstructured Delaunay pattern. Methods 2 and 3. Comparison between the original least squares method and two variants.

We have performed computations using the meshes obtained with the anisotropic adaptive algorithm presented in [21, 20, 22]. The goal of the adaptive algorithm is to generate anisotropic adapted meshes such that

$$0.75\ TOL \leq \frac{\left(\sum_{K\in\mathcal{T}_h} \eta_K^2\right)^{1/2}}{\|\nabla u_h\|_{L^2(\Omega)}} \leq 1.25\ TOL.$$

In other words the estimated relative error should be close to a preset tolerance TOL. Hereabove, η_K is a local anisotropic error estimator based on Formaggia-Perotto interpolation estimates [12, 13] and on Zienkiewicz-Zhu post-processing. In Table 6, results are reported when TOL goes to zero, ar denoting the average aspect ratio, ei^{ZZ} the effectivity index of Zienkiewicz-Zhu post-processing:

$$ei^{ZZ} = \frac{\|\nabla u_h - Gu_h\|_{L^2(\Omega)}}{\|\nabla(u-u_h)\|_{L^2(\Omega)}}.$$

The results are surprising since method 4 is again the best one as for regular pattern meshes. The approximation w_h computed with methods 2 and 4 is plotted in Figs. 6 and 7 when $TOL = 0.0625$ Eventhough there is no clear convergence rate for method 2, the pointwise error is only 2%. The error for method 4 cannot be seen in the zoom of Fig. 7, it is less than 0.3%.

			meth. 1		meth. 3	meth. 2	meth. 4
TOL	De	De^{ZZ}	D^2e^{ZZ}	De^{LS}	D^2e^{LS1}	D^2e^{LS2}	D^2e^{WF}
0.0625	0.082	0.032	1.34	0.025	2.00	1.20	0.23
0.03125	0.041	0.012	1.05	0.010	1.73	0.68	0.11
0.015625	0.020	0.0058	1.12	0.0040	1.61	0.54	0.086
0.0078125	0.011	0.0030	1.08	0.0016	1.60	0.47	0.052
0.00390625	0.0053	0.0014	0.96	0.00071	1.54	0.43	0.0350

TOL	N_h	ar	ei^{ZZ}
0.0625	241	47	0.99
0.03125	477	81	0.99
0.015625	1163	126	1.00
0.0078125	3181	169	1.00
0.00390625	9242	211	1.00

Table 6: Anisotropic adapted meshes.

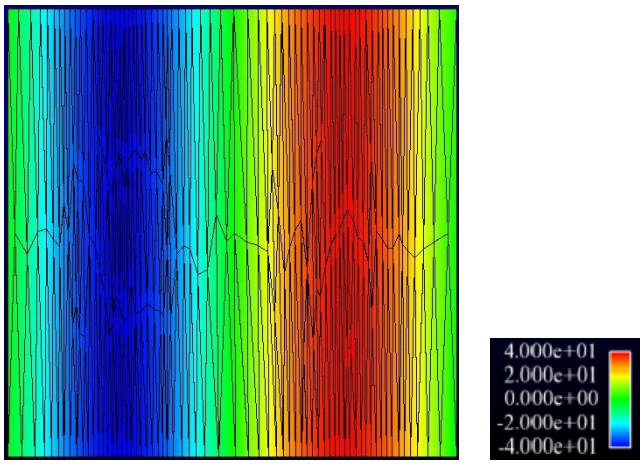


Figure 6: Anisotropic adapted mesh $TOL = 0.0625$. Isovalues of w_h (method 4).

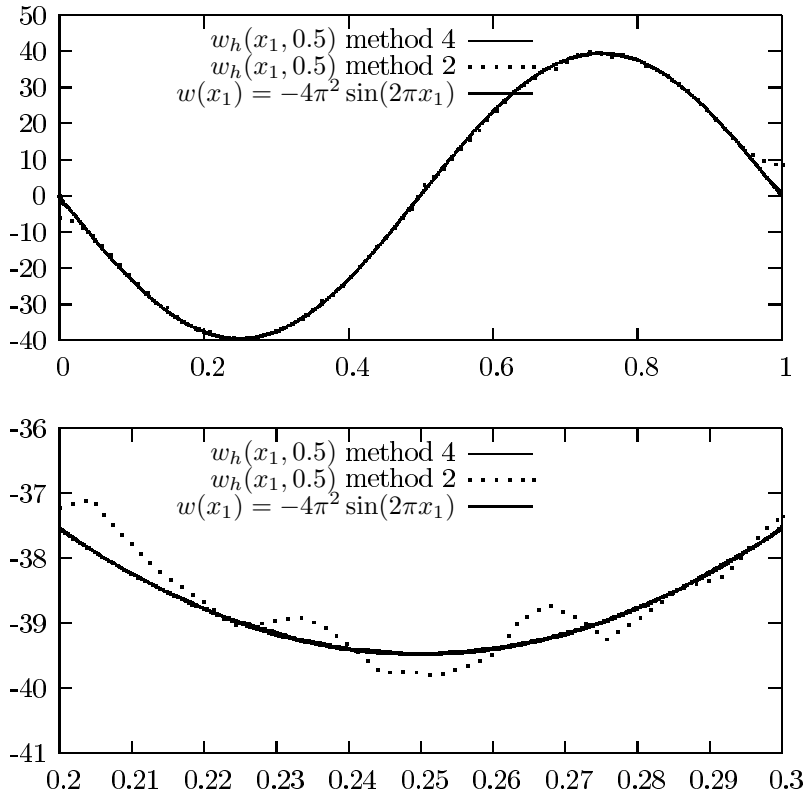


Figure 7: Anisotropic adapted mesh $TOL = 0.0625$. Plot of w and w_h (methods 2 and 4) against x_1 at $x_2 = 0.5$.

3.2 3D computations

In three space dimensions we have only implemented methods 1 and 4. We set $\Omega = (0,1)^3$ and f so that the solution of (2) is given by

$$u(x_1, x_2, x_3) = \frac{1}{2}\Big(\tanh(40r(x_1, x_2, x_3) - 0.1) - \tanh(40r(x_1, x_2, x_3) - 0.2)\Big),$$

with $r(x_1, x_2, x_3) = \sqrt{(x_1 - 0.5)^2 + (x_2 - 0.5)^2 + (x_3 - 0.5)^2}$. Thus is smooth, zero or one valued, except in a region of small width. In Table 7, we results are reported when using structured meshes using the `tetr` module of the `MODULEF` software [2]. The computational domain is cut into cubes, each cube being cut in 5 or 6 tetrahedra. Concerning method 1, as for the 2D results of table 1, superconvergence of the Zienkiewicz-Zhu gradient can be observed in both cases, so as convergence of w_h towards w, which corresponds to the results of Theorem 2. On the other side, method 4 does not seem to converge when cutting each cube in 5, whereas convergence can be observed when cutting each cube in 6. In Fig. 8, we have reported a cut along the $x_3 = 0.5$ plane of w_h computed with method 4 when $h = 0.0125$, each cube being cut into 5 tetrahedra. Clearly, oscillations can be observed, as for the criss-cross pattern in 2D, see Fig. 2.

			method 1	method 4	
h	De	De^{ZZ}	D^2e^{ZZ}	D^2e^{WF}	N_h
0.05	2.53	1.10	40.8	27.9	9261
0.025	0.88	0.25	25.1	18.5	68921
0.0125	0.41	0.062	9.98	15.5	531441
0.05	2.96	1.04	45.3	23.6	9261
0.025	1.01	0.31	26.7	7.74	68921
0.0125	0.47	0.087	11.5	2.66	531441

Table 7: 3D results. Structured meshes. Each cube is cut in 5 tetr (rows 1,2,3) or 6 tetr (rows 4,5,6).

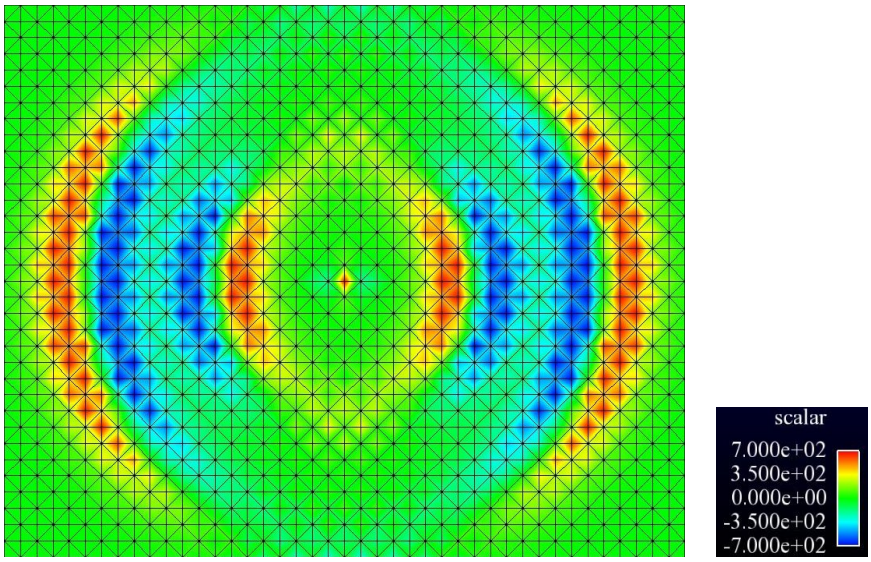


Figure 8: 3D results. Structured mesh, $h = 0.0125$, each cube is cut in 5 tetr. Zoom of w_h (method 4) at $x_3 = 0.5$.

We have also performed computations using the meshes obtained with the anisotropic adaptive algorithm presented in [22], the `mmg3d` remeshing tool [10] being used. The initial mesh was a $10 \times 10 \times 10$ mesh and 30 mesh iteration were performed for each run. The numerical approximation u_h is plotted in Fig. 9 when $TOL = 1$, the numerical results with several values of the TOL are reported in Table 8. As for 2D results with adaptive meshes (Table 6 col. 4), there is no clear convergence rate. The approximation w_h is plotted in Fig. 10 for $TOL = 0.125$ and the pointwise error remains within 10%.

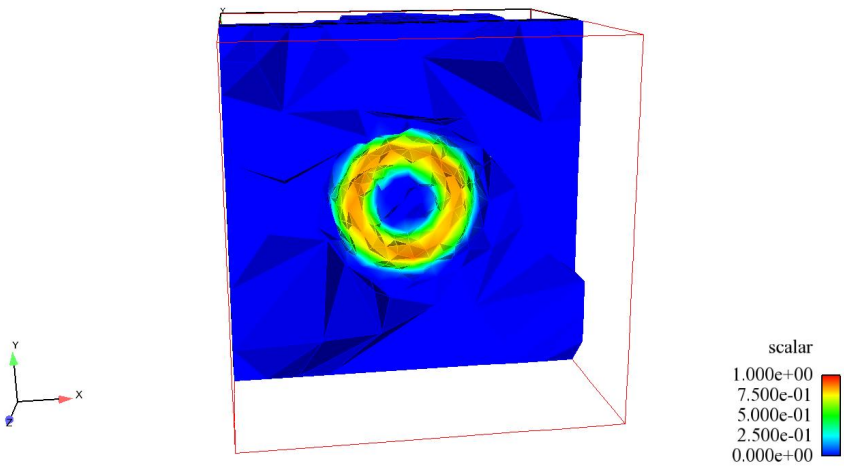


Figure 9: 3D results. Anisotropic adapted mesh $TOL = 1$. Plot of u_h at $x_3 = 0.5$.

			method 1	method 4	
TOL	De	De^{ZZ}	D^2e^{ZZ}	D^2e^{WF}	N_h
1	0.90	0.35	15.6	7.80	2264
0.5	0.46	0.14	8.24	5.65	7432
0.25	0.23	0.056	4.95	4.71	42101
0.125	0.11	0.026	4.01	4.33	340450

Table 8: 3D results. Anisotropic adapted meshes with several values of TOL.

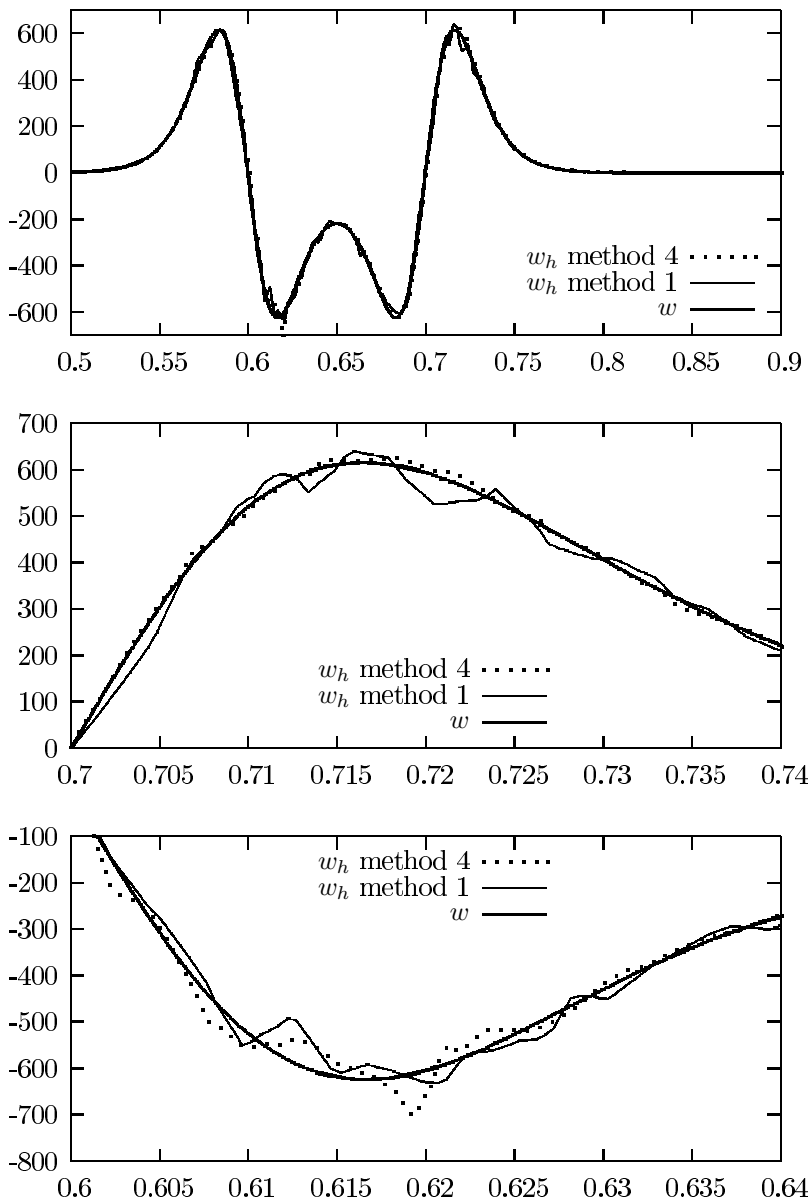


Figure 10: 3D results. Anisotropic adapted meshes $TOL = 0.125$. Plot of w and w_h (method 1 and 4) against x_1 at $x_2 = x_3 = 0.5$. Top: $0.5 \leq x_1 \leq 0.9$. Middle and bottom: zoom.

4 Conclusions

The convergence of numerical methods to approach second derivatives in the framework of the Laplace problem with continuous, piecewise linear finite elements has been investigated.

Numerical results in 2D and 3D show that the precision of all methods considered depends strongly on the mesh topology and that no convergence can be certified in general. However, there is not blow-up and the values obtained are probably accurate enough in order to be used as refinement or coarsening criteria in adaptive algorithms.

Acknowledgements

Cécile Dobrzynski and Pascal Frey are acknowledged for providing the the `mmg3d` anisotropic remeshing tool [10].

References

[1] F. Alauzet. High-order methods and mesh adaptation for Euler equations. *Internat. J. Numer. Methods Fluids*, 56(8):1069–1076, 2008.

[2] M. Bernadou, P. L. George, A. Hassim, P. Joly, P. Laug, B. Muller, A. Perronnet, E. Saltel, D. Steer, G. Vanderborck, and M. Vidrascu. *MODULEF: une bibliothèque modulaire d'éléments finis*. Institut National de Recherche en Informatique et en Automatique (INRIA), Rocquencourt, second edition, 1988.

[3] H. Borouchaki, D. Chapelle, P.-L. George, P. Laug, and P. Frey. *Maillage et Adaptation*, section 9, pages 279–309. Mécanique et Ingéniérie des Matériaux. Hermès - Lavoisier, Paris, 2001. Sous la direction de P.-L. George (in french).

[4] Y. Bourgault, M. Picasso, F. Alauzet, and A. Loseille. On the use of anisotropic a posteriori error estimators for the adaptative solution of 3D inviscid compressible flows. *Internat. J. Numer. Methods Fluids*, 59(1):47–74, 2009.

[5] J. Brandts and M. Křížek. Gradient superconvergence on uniform simplicial partitions of polytopes. *IMA J. Numer. Anal.*, 23(3):489–505, 2003.

[6] M. J. Castro-Díaz, F. Hecht, B. Mohammadi, and O. Pironneau. Anisotropic unstructured mesh adaption for flow simulations. *Internat. J. Numer. Methods Fluids*, 25(4):475–491, 1997.

[7] P. G. Ciarlet. *The finite element method for elliptic problems*, volume 40 of *Classics in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2002. Reprint of the 1978 original [North-Holland, Amsterdam; MR0520174 (58 #25001)].

[8] P. Clément. Approximation by finite element functions using local regularization. *RAIRO Anal. Numér.*, 9:77–84, 1975.

[9] G. Compère, E. Marchandise, and J.-F. Remacle. Transient adaptivity applied to two-phase incompressible flows. *J. Comput. Phys.*, 227(3):1923–1942, 2008.

[10] C. Dobrzynski. *Adaptation de maillage anisotrope 3D et application à l'aéro-thermique des bâtiments.* PhD thesis, Université Pierre et Marie Curie, Paris VI, 2005.

[11] J. Dompierre, M.-G. Vallet, Y. Bourgault, M. Fortin, and W. G. Habashi. Anisotropic mesh adaptation: towards user-independent, mesh-independent and solver-independent CFD. III. Unstructured meshes. *Internat. J. Numer. Methods Fluids*, 39(8):675–702, 2002.

[12] L. Formaggia and S. Perotto. New anisotropic a priori error estimates. *Numer. Math.*, 89(4):641–667, 2001.

[13] L. Formaggia and S. Perotto. Anisotropic error estimates for elliptic problems. *Numer. Math.*, 94(1):67–92, 2003.

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

[15] P. J. Frey and P.-L. George. *Mesh generation.* ISTE, London, second edition, 2008. Application to finite elements.

[16] B. Hamann. Curvature approximation for triangulated surfaces. In *Geometric modelling*, volume 8 of *Comput. Suppl.*, pages 139–153. Springer, Vienna, 1993.

[17] P. Laug and H. Borouchaki. The BL2D mesh generator: Beginner's guide, user's and programmer's manual. Technical Report RT-0194, Institut National de Recherche en Informatique et Automatique (INRIA), Rocquencourt, 1996.

[18] P. Laug and H. Borouchaki. BL2D-V2 mailleur bidimensionnel adaptatif. Technical Report RT-0275, Institut National de Recherche en Informatique et Automatique (INRIA), Rocquencourt, 2003.

[19] A. Naga and Z. Zhang. A posteriori error estimates based on the polynomial preserving recovery. *SIAM J. Numer. Anal.*, 42(4):1780–1800 (electronic), 2004.

[20] M. Picasso. An anisotropic error indicator based on Zienkiewicz-Zhu error estimator: application to elliptic and parabolic problems. *SIAM J. Sci. Comput.*, 24(4):1328–1355 (electronic), 2003.

[21] M. Picasso. Numerical study of the effectivity index for an anisotropic error indicator based on Zienkiewicz-Zhu error estimator. *Comm. Numer. Methods Engrg.*, 19(1):13–23, 2003.

[22] M. Picasso. Adaptive finite elements with large aspect ratio based on an anisotropic error estimator involving first order derivatives. *Comput. Methods Appl. Mech. Engrg.*, 196(1-3):14–23, 2006.

[23] R. Rodríguez. Some remarks on Zienkiewicz-Zhu estimator. *Numer. Methods Partial Differential Equations*, 10(5):625–635, 1994.

[24] M.-G. Vallet, C.-M. Manole, J. Dompierre, S. Dufour, and F. Guibault. Numerical comparison of some Hessian recovery techniques. *Internat. J. Numer. Methods Engrg.*, 72(8):987–1007, 2007.

[25] M. F. Wheeler and J. R. Whiteman. Superconvergent recovery of gradients on subdomains from piecewise linear finite-element approximations. *Numer. Methods Partial Differential Equations*, 3(1):65–82, 1987.

[26] J. Xu and Z. Zhang. Analysis of recovery type a posteriori error estimators for mildly structured grids. *Math. Comp.*, 73(247):1139–1152 (electronic), 2004.

[27] Z. Zhang and A. Naga. A new finite element gradient recovery method: superconvergence property. *SIAM J. Sci. Comput.*, 26(4):1192–1213 (electronic), 2005.

[28] Zhimin Zhang. Polynomial preserving recovery for meshes from Delaunay triangulation or with high aspect ratio. *Numer. Methods Partial Differential Equations*, 24(3):960–971, 2008.

[29] O. C. Zienkiewicz and J. Z. Zhu. A simple error estimator and adaptive procedure for practical engineering analysis. *Internat. J. Numer. Methods Engrg.*, 24(2):337–357, 1987.

[30] O. C. Zienkiewicz and J. Z. Zhu. The superconvergent patch recovery and a posteriori error estimates. I. The recovery technique. *Internat. J. Numer. Methods Engrg.*, 33(7):1331–1364, 1992.

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

Centre de recherche INRIA Bordeaux – Sud Ouest : Domaine Universitaire - 351, cours de la Libération - 33405 Talence Cedex
Centre de recherche INRIA Grenoble – Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier
Centre de recherche INRIA Lille – Nord Europe : Parc Scientifique de la Haute Borne - 40, avenue Halley - 59650 Villeneuve d'Ascq
Centre de recherche INRIA Nancy – Grand Est : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex
Centre de recherche INRIA Rennes – Bretagne Atlantique : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex
Centre de recherche INRIA Saclay – Île-de-France : Parc Orsay Université - ZAC des Vignes : 4, rue Jacques Monod - 91893 Orsay Cedex
Centre de recherche INRIA Sophia Antipolis – Méditerranée : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex

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