A metric-based adaptive mesh refinement criterion under constrain for solving elliptic problems on quad/octree grids

Lucas Prouvost, Anca Belme, Daniel Fuster

To cite this version:

Lucas Prouvost, Anca Belme, Daniel Fuster. A metric-based adaptive mesh refinement criterion under constrain for solving elliptic problems on quad/octree grids. Journal of Computational Physics, 2024, 506, pp.112941. 10.1016/j.jcp.2024.112941 . hal-04516638

HAL Id: hal-04516638
https://hal.science/hal-04516638v1

Submitted on 22 Mar 2024

A metric-based adaptive mesh refinement criterion under constrain for solving elliptic problems on quad/octree grids

Prouvost Lucas[a,*], Belme Anca[a], Fuster Daniel[a,**]

[a]*Sorbonne University, CNRS, Institute Jean Le Rond d'Alembert, F-75005 Paris, France*

Abstract

In this work we propose and investigate the performance of a metric-based refinement criteria for adaptive meshing used for improving the numerical solution of an elliptic problem. We show that in general, when solving elliptic equations such as the Poisson-Helmholtz equation, the minimization of the interpolation error often used as local refinement criteria, does not always guarantee the minimization of the total numerical error. Numerical and theoretical arguments are given to unveil the critical role of the mesh compression – the size aspect ratio between the finest cell size and the mean cell size of an adapted mesh – to determine whether the estimated error is purely local meaning that the interpolation error is a good enough error model for the total error or if other, non-local, sources of error needs to be accounted for. We show through particular examples that a slightly sub-optimal mesh in terms of interpolation error may significantly reduce the total error of a numerical solution, depending on the value of the compression ratio and not on the number of grid points. Based on this observation, we propose a new method to exclude the grids where non-local errors can control the accuracy of the solution. This is achieved by an automatic estimation of the optimal compression ratio, which is imposed as an additional constraint in the minimal element size during the mesh adaptation process. The method is tested on quadtree and octree grids, showing very satisfactory performances in reducing the total numerical error despite the additional constrain imposed.

Nomenclature

η	Compression ratio
η_0	Local optimal compression ratio
η_c	Characteristic compression ratio
η_{opt}	Optimal compression ratio
γ_p	Normalized $\boldsymbol{L}^p$-norm total error
γ_p^I	Normalized $\boldsymbol{L}^p$-norm interpolation error
$\overline{h}$	Mean element size
$\Pi_{\mathcal{M}}$	Continuous interpolation operator
Π_h	Discrete interpolation operator

---

[*] lucas.prouvost@dalembert.upmc.fr

[**] fuster@dalembert.upmc.fr

Preprint submitted to Elsevier *March 22, 2024*

$\tilde{s}$	Source term of a discretized PDE
$\tilde{u}$	Discrete solution of a PDE
C_n	Isotropic metric-based error prefactor
c_n	Anisotropic metric-based error prefactor
d_*	Dimensionless mesh density function
H	Hessian matrix
h	Mesh element size
h_{min}	Mesh minimal element size
l	Adaptive mesh level of refinement
L_0	Square/cube domain length
l_{max}	Adaptive mesh maximum level of refinement
N	Number of elements of the current adapted mesh
N_{fine}	Number of elements of a fine mesh
N_{goal}	Targeted number of elements for an adapted mesh
$N_{h_{min}}$	Number of elements of a uniformly refined mesh with elements of size h_{min}
s	Source term of a continuous PDE
u	Continuous solution of a PDE
u'	Implicit error term

1. Introduction

The numerical resolution of physical and mechanical problems is of primordial importance in numerous applications, from academical purposes [1] to industrial applications [2]. It is common knowledge that the analytical solution of the mathematical equations governing these complex and potentially multi-scale phenomena is out of reach while building and testing experimental prototypes is a highly costly operation. Numerical simulations render feasible the understanding and improvement of complex physical phenomena, making them a key ingredient in the decision-making processes.

The numerical resolution of partial differential equations in problems where multiple scales with potentially multiple flow phases are involved requires highly accurate numerical schemes and meshes, which leads to highly costly simulations (see for example [3] where a pseudo direct numerical simulation combined with a representative volume element is used for the simulation of water droplets dispersed in a turbulent airflow). A common methodology to increase the efficiency of computations is the use of adaptive methods, for example automatic adaptive meshes. By increasing locally or ballancing the mesh resolution in regions where a more accurate prediction is needed, it is possible to significantly reduce numerical errors without a significant increase in computational time. Mesh adaptation is therefore an interesting alternative for accurate prediction of multi-scale, multi-phase flows or for flow problems with thin boundary layers and/or singularities such as shocks. The search for efficient and fully automated mesh adaptation methods is under continuous development and has been identified as one of the major bottlenecks in CFD workflows in the CFD vision 2030 Study [4, 5].

The main purpose of mesh adaptation methods is to reach an optimal balance between the computational cost and the numerical error reduction (or targeted accuracy). The accuracy of a numerical solution is affected by errors coming from several sources [6, 7, 8, 9]. Discretization errors account for the errors due to the discrete representation (mesh, solution) of a continuous problem. In particular, interpolation errors measure the discrepancies between the exact (continuous) solution and the projection of the continuous solution into an element of size h. Interpolation errors are local measures of the solution accuracy. Other sources of numerical errors related to the numerical schemes and their implementation are not necessarily local: they can be advected or diffused throughout the computational mesh.

It has become common practice to quantify mesh discretization errors using interpolation errors since they are local by nature and can be efficiently used as error indicators for mesh refinement. H-adaptation methods are the most widespread adaptive methods for Finite Volume discretizations. They are also called Adaptive Mesh Refinement (AMR) methods and their core concept is to adapt the size of the mesh elements: fine elements are required in critical regions, whereas coarse elements are sufficient elsewhere. The generation of new elements in h-adaptation techniques may be divided in several categories: either an initial group of elements is divided in smaller elements (*cell*-based AMR [10, 11], *patch*-based [12, 13] and *block*-based AMR [14]) or the entire domain is completely re-meshed with a new elements' distribution at each adaptation step [15, 16]. One currently active research field for h-adaptation is the metric-based anisotropic adaptation [15, 17, 18, 19], which allows the generation of anisotropic elements: this is well-suited to follow the features orientation of flow fields. The metric-based method has been applied to both re-meshing techniques [9, 15, 18, 20], and cell-based AMR [19, 21]. Of crucial importance in mesh adaptation problems is the definition of an accurate refinement indicator which also accounts for the physics of the problem to be solved. Error estimators based on the *interpolation error* [15, 22, 23, 24] are often used as refinement criteria for *feature-based* methods which quantify the interpolation error, $|u - \Pi_h u|$, identified as the difference between a continuous (exact) solution u and its interpolation $\Pi_h u$ on a discrete mesh. For the special case of linear interpolation, this error is governed by the second derivatives of a physical field. Among the interpolation error estimates, some uses high-order accurate estimates of the solution [25, 26, 27, 28, 29, 30] for relatively smooth solutions. Another efficient use of the interpolation error in mesh adaptation problems combines the computation of a metric associated to a continuous interpolate in the Riemannian metric space (see for example [15, 21, 23, 24, 31]). A continuous representation of the mesh is proposed to solve the mesh optimization problem leading to the control of the interpolation error in various $\boldsymbol{L}^p$-norm. This method has shown to be efficient in capturing all the scales of the flow features including problems with shocks or discontinuities. The metric-based refinement method has been traditionnaly developed for tetrahedral and triangular elements and successfully applied in numerous problems [15, 32, 23, 33]. However, the application of this method to hexahedra and quadrangular elements is far less studied. Two main directions are under development to refine these mesh elements using the Riemannian metric theory. The most trivial path is to generate tetrahedral elements, and recombine them to form hexahedra. Recent works propose to generate quasi right-angled triangles and tetrahedras, allowing for easy recombination into hexahedra. These particular elements are obtained through a point-distribution energy minimization process done by the mesh generator [33, 34, 35]. The second direction uses directly hex-meshes [19, 21, 32, 31]. In these works, a set of initial quad/hex-elements are refined (or coarsened) based on a metric-based size criterion. This refinement (or coarsening) consists in successive element divisions (or merging).

Feature-based (including interpolation error-based) error estimates have a common property: they are only based on the local representation of the numerical solution. Therefore, they do not consider the equations being solved, nor they take into account the errors propagation. In order to account for the partial differential equation (PDE) and the associated discrete solver in the mesh adaptation problem, many error estimates have been developed that can be divided into two major classes: the *residual-based* error estimators and the *adjoint-based* error estimators. The residual-based methods [36] originate mainly from the finite-element mathematical context and rely on the estimation of the residuals: the difference between the discretized equation and its continuous counterpart. Residual-based methods are commonly used as *residual-based* errors [37, 38, 39, 40, 41] or in the *error transport equation* [42, 43, 44, 45, 46]. In the first case, the residual is directly used to bound the discretization error using usually an energy norm. The computation and accuracy of the residual depends on the continuous PDE and its discretization and is either computed analytically or approximated and reconstructed, which often necessitate to estimate quantities on different (nested) grids. In the second family of residual-based methods, a (linearized) equation governing the discretization error is written, and the residual is the source term of this equation. Considering that fact, the regions of higher residual amplitude are generally considered as the site of production of the discretization error. This second method intrinsically takes into account the transport of errors from regions with insufficient resolution [36, 42], but may lead to over-refinement in the regions where it is transported – referred as *pollution error* [47], instead of the region it is produced, resulting in sub-optimal performances [36, 47, 48]. Furthermore, this method requires the solution of an additional equation or set of equations increasing thus the computational cost.

In cases where one focuses on improving the computation and thus accuracy of some quantity of interest depending on the solution of a PDE, the solution does not need to be resolved with the same accuracy in the whole domain and the computational resources can rather be redirected in a given region of interest [49, 50]. These adaptive methods are called *adjoint-based* or *goal-oriented* [51, 52, 53] methods and require the computation of an adjoint state in the error estimation. Adjoint-based methods generally provides better error rate of convergence regarding the quantity of interest accuracy than feature-based error estimations [9] since the mesh refinement effort focuses only on capturing flow features that will affect the quantity of interest. Several adjoint-based error estimates have been developed in the literature, the most common one is expressed as a weighted residual formulation computed from primal and/or dual (adjoint) states [51, 52, 53, 54, 55, 56]. However, other forms of adjoint-based error estimates exist, some of which are combined with the interpolation error theory [15, 17, 18, 20] or more recently the norm oriented framework [57, 58, 59]. These last methods are based on the continuous framework of Riemannian metrics and propose *a priori* estimates expressed as adjoint-weighted interpolation errors.

A large extent of residual-based and goal-oriented methods have been used in the literature to improve the AMR performances of elliptic PDE solvers, which provide a large amount of error estimators. Among them, we can for example find residual-based estimators in the works [60, 61, 62, 63, 64, 65] and goal-oriented estimators in [18, 66, 67, 68]. However, the former are by nature dependent of the discretization scheme used in the solver while the latter focus only on minimizing the error of a quantity of interest. Error estimates for elliptic equations containing non-regular terms, such as Dirac source terms [69, 70, 71], have been also used as mesh adaptation criterion, as they allow to obtain graded meshes on which the order of convergence is increased with compared to uniform meshes.

In this paper, we propose to extend the metric-based mesh refinement method of [15] to improving elliptic problems resolutions on quad/octree meshes. While this adaptive mesh refinement (AMR) method focuses on minimizing interpolation errors, we show that an additional constraint on the mesh compression – the size aspect ratio between the finest cell size and the mean cell size – can be defined to ensure the estimated error remains local. Numerical and theoretical arguments are given to motivate the need for this additional constraint. We provide the open-source code as part of the Basilisk software [72, 73, 74].

In the next section, we propose a brief description of the numerical solver and grid structure used to solve the PDE of interest. Section 3 discusses some theoretical measures of error sources and their propagation, and in Section 4 we illustrate the theoretical results of the errors behavior on two analytical examples where a mesh coarsening experiment is performed. In Section 5 we introduce the improved mesh optimization problem method with the automatic computation of the mesh compression ratio and the efficiency of this methods is tested on several numerical problems in the last section. We end this paper with some conclusion and perspectives for future work.

2. Numerical solver and grid structure

In this work, we concentrate on minimizing the total error – the difference between the continuous analytical solution of a continuous problem and the numerical solution of the corresponding discretized problem – when solving a discrete Poisson–Helmholtz equation on quad/octree meshes. Quad/octree grids are an efficient way to create AMR meshes, and they are designed to facilitate cell refinement and coarsening by only allowing to divide or merge previously existing cells. We recall here some basic principles regarding the structure and particularities of the quad/octree grids implemented and tested in the solver Basilisk [11, 73, 72] .

The quadtree (*resp.* octree) grid structure [11, 75, 76] is composed with squares (*resp.* cubes). The core concept is that each square (*resp.* cube) can be divided into four squares (*resp.* eight cubes): the cell before division is referred to as a *parent* cell, and the four (*resp.* eight) new cells are its *children.* This structural link between *parent* and *child* cells defines the tree-based grids (see figure 1 (left image)). The tree grids are defined from an original element called the *root* cell of size L_0. The tree is constructed through successive cell divisions, where all the cells obtained after a common number of divisions of the root cell are by definition at the same hierarchical *level* l (with l an integer number) as illustrated in figure 1 (right image). Each cell having the same level l has the same number of parent cells, and the same size h which is directly obtained as a function of the root cell size :

$$h = \frac{L_0}{2^l} \ . \tag{1}$$

The tree grid structure contains two types of cells: the cells without children on which the solution is computed, called *leaf cells* or *leaves*, and the cells with children – called *parent cells.* The minimal cell size h_{min} of a non-uniform mesh is represented by the maximum refinement level l_{max}

$$h_{min} = \frac{L_0}{2^{l_{max}}} \ . \tag{2}$$

Three important characteristics should be highlighted. First, quad/octree grids are composed of isotropic elements, where each mesh element is defined by a unique length h (see (1)). Secondly, the set of element size $h(l)$ forms a *discrete set* due to the dependence on the associated level l, which is an integer value for each mesh element. This introduces jumps in the transition of the maximum grid size and imposes a finite size of the mesh optimization problem, which for a given number of grid elements N can be shown to scale as $N^{N/3}$ in 2D domains and the maximum admissible level of refinement is $l_{max} \approx N/3$. Finally and to ease solvers implementation, the quad/octree structure of Basilisk limits the element size change between two neighbouring elements, which can not be different than one level. This feature, called 2:1 constraint in Basilisk [11], is also known as *1-irregularity* rule [77, 78] and *graded* mesh and limit the space of possible solutions compared to methods based on *non-graded* quad/octree [79, 80, 81, 82]. The number of cells which must be added to respect this constraint is expected to be small for sufficiently smooth solutions (absence of shock waves or singularities), as it can be inferred from the results presented in [79, 80, 81, 82].

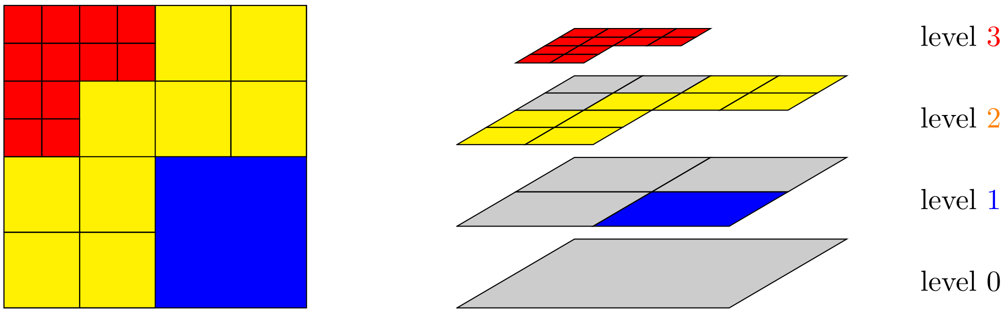


Figure 1: Example of the structure of a quadtree grid.

One important property of an adaptive grid is the **compression ratio** η defined as

$$\eta = \frac{N}{N_{h_{min}}} \ , \tag{3}$$

where N is the number of cells of an adapted mesh and $N_{h_{min}}$ is the number of cells that would contain a uniform mesh whose cell size is equal to the minimum grid size h_{min} of the adapted grid. The compression ratio represents a measure of the level of compression of a grid, and by definition it is bounded in the interval $\eta \in]0,1]$. The limit $\eta = 1$ corresponds to a uniform mesh, whereas η tends to 0 as long as the number of elements decreases while keeping a constant minimal cell size. The compression ratio may be rewritten as a function of a cell size ratio. Suppose $\overline{h}$ is the mean cell size of an adapted mesh with N elements, $\overline{h}$ and N are linked by the relation $N = \left(L_0/\overline{h}\right)^n$, with L_0 the domain size and n the problem dimension. Using the number of elements of a uniform mesh with the minimum element size, $N_{h_{min}} = \left(L_0/h_{min}\right)^n$, we deduce that the compression ratio is directly related to the ratio between the minimum and averaged grid size

$$\eta = \left(\frac{h_{min}}{\overline{h}}\right)^n \ . \tag{4}$$

3. Error sources in a numerical solution

3.1. General description

Let us consider u the exact solution of a continuous partial differential equation (PDE) expressed through it's continuous operator $\mathcal{L}$ and source term s as:

$$\mathcal{L}u = s \tag{5}$$

with appropriate boundary conditions. The discretization of the system above on a given grid is expressed as

$$\tilde{\mathcal{L}}\tilde{u} = \tilde{s} \tag{6}$$

where $\tilde{\mathcal{L}}$ represents the discretized operator applied to the numerical approximate solution $\tilde{u}$. The objective is to quantify the total error, i.e $\|u - \tilde{u}\|_{\mathbf{L^p}(\Omega)}$ commited on a computation domain Ω for a given $\mathbf{L^p}$ norm.
We introduce a linear interpolation operator Π_h and an error measure :

$$u' \equiv \tilde{u} - \Pi_h u$$

which accounts for the difference between the numerical solution $\tilde{u}$ and the interpolation of the exact solution $\Pi_h u$. In some works such as [18, 24] this error term is called implicit error. Using the triangle inequality, it readily follows that the total error is bounded by

$$\|u - \tilde{u}\|_{\mathbf{L^p}} = \|u - \Pi_h u - u'\|_{\mathbf{L^p}} \leq \|u - \Pi_h u\|_{\mathbf{L^p}} + \|u'\|_{\mathbf{L^p}} \tag{7}$$

This representation of the total error allows us to isolate the contribution of the purely local interpolation error intrinsic to the structure of the solution, $\|u - \Pi_h u\|_{\mathbf{L^p}}$, and the error introduced in the numerical solution due to the discretization of the PDE, $\|u'\|_{\mathbf{L^p}}$. Indeed, suppose a linear discretization operator $\mathcal{L}$, then the additional source of error u' verifies the relation:

$$\tilde{\mathcal{L}} u' = \tilde{s} - \tilde{\mathcal{L}}(\Pi_h u). \tag{8}$$

The right hand side in this relation is the residual of the discretized equation when the exact solution is interpolated into a given grid. This representation of the total error allows writting an explicit expression of the contribution of the purely local error into the estimation of the total error.

The interpolation error is a lower bound of the total numerical error. For situations where $\|u - \Pi_h u\|_{\mathbf{L^p}} \gg \|u'\|_{\mathbf{L^p}}$ the total numerical error is essentially controlled by local errors and the strategies to model and minimize such errors are known and local. Otherwise, the techniques to minimize u' are more involved. In this work, we will evaluate the relative contribution of both sources of errors when adapting the grid using only the interpolation error as refinement indicator.

3.2. Interpolation errors

We focus here on modeling the interpolation errors introduced by classical second order methods where the solution is bilinearly interpolated from exact nodal values. In the context of Riemannian metric-based mesh adaptation, a continuous interpolation error – equivalent to the discrete interpolation error – is defined and allows to find an optimal mesh as the solution of an interpolation error minimization problem. Expressions for the local and global optimal error for grids containing anisotropic triangular/tetrahedral elements has been derived in [83, 84]. The core of this approach is to use a continuous formulation of the mesh which is possible by equivalence relationships as shown in references bellow. A continuous mesh $\mathbf{M}(\Omega) = (\mathcal{M}(\boldsymbol{x}))_{\boldsymbol{x} \in \Omega}$ in a domain Ω is then defined using the Riemannian metric, a symetric positive defined matrix associated to each vertex in the discrete mesh $\mathcal{H}$. In this continuous framework the mesh adaptation problem is convex and the optimal mesh is obtained such that all simplicial elements are unit elements in the computed Riemannian metric.

Let u be a quadratic positive function with H its associated Hessian matrix, and $\Pi_{\mathcal{M}}$ the continuous linear interpolation operator in the continuous Riemannian metric space. Loseille and Alauzet [83, 84] show that a family of optimal grids exist for which the expressions of the local continuous interpolation error associated to $\boldsymbol{x} \in \Omega$, $|u(\boldsymbol{x}) - \Pi_{\mathcal{M}} u(\boldsymbol{x})|$, and the global interpolation error, $||u - \Pi_{\mathcal{M}} u||_{\mathbf{L^p}}$, can be written as

$$|u - \Pi_{\mathcal{M}} u|(\boldsymbol{x}) = c_n \, n \left(\int_{\Omega} (det(H(\boldsymbol{x}))^{\frac{p}{2p+n}} \ \mathrm{d}\boldsymbol{x} \right)^{\frac{2}{n}} (det(H(\boldsymbol{x})))^{\frac{1}{2p+n}} N_{goal}^{-2/n} \tag{9}$$

$$||u - \Pi_{\mathcal{M}} u||_{\mathbf{L^p}} = c_n \, n \left(\int_{\Omega} (det(H(\boldsymbol{x}))^{\frac{p}{2p+n}} \ \mathrm{d}\boldsymbol{x} \right)^{\frac{2p+n}{np}} N_{goal}^{-2/n} \tag{10}$$

where N_* is the mesh complexity, n is the dimension of the problem and c_n is a constant dependent on the problem dimension: $c_2 = 1/16$ and $c_3 = 1/20$. In the asymptotic regime, N_{goal} is proportional to the number of vertices [84]. This result has been extended to non-quadratic functions by considering the absolute Hessian, $|H|$ obtained by taking the absolute eigenvalues from the Hessian H (see [84]).

In this work, we extend this theory to quad/octree grids containing square/cubic elements. First, we extend the definition of a unit element to the case of a polyhedron as in [85]: A polyhedron $K = (\boldsymbol{e}_i)_{i \in n_{\boldsymbol{e}}}$ with $n_{\boldsymbol{e}}$ edges is unit with respect to a metric $\mathcal{M}$ if the length of all of its edges is unit in this metric. In 2D, a square element with a length h is unit for the metric $\mathcal{M}_u = h^{-2} I_2$, with I_2 the identity matrix. Following [84], we define the continuous interpolation operator $\Pi_{\mathcal{M}}$ such that the continuous error $|u - \Pi_{\mathcal{M}} u|$ is equal to the discrete interpolation error on a unit square/cubic element K represented by its metric $\mathcal{M}_u$:

$$|u - \Pi_{\mathcal{M}} u|(\boldsymbol{x}) = C_n \operatorname{tr} \left(\mathcal{M}_u^{-1/2}(\boldsymbol{x}) H(\boldsymbol{x}) \mathcal{M}_u^{-1/2}(\boldsymbol{x}) \right) \tag{11}$$

where u is a quadratic positive function with the associated Hessian H, $\operatorname{tr}(\cdot)$ is the trace of a matrix and $C_n = C_2 = C_3 = 1/12$. The details of the demonstration leading to these results are summarized in Appendix A. As in the case of simplicial elements, we extend this result to non-quadratic functions by considering their absolute Hessian $|H|$ obtained from their Hessian H by taking its absolute eigenvalues. Thus, the representation of the continuous local error obeys the form

$$|u - \Pi_{\mathcal{M}} u|(\boldsymbol{x}) = C_n \operatorname{tr}(|H|(\boldsymbol{x})) h^2. \tag{12}$$

The search for an optimal mesh that minimizes the global error in $\mathbf{L^p}$ norm writes :

$$\text{Find} \min_{\boldsymbol{M}} \left(\int_{\Omega} (|u - \Pi_{\mathcal{M}} u|(\boldsymbol{x}))^p \ \mathrm{d}\boldsymbol{x} \right)^{1/p}, \tag{13}$$

$$\text{under the constraint } \mathcal{C}(\boldsymbol{M}) = \int_{\Omega} (h^n(\boldsymbol{x}))^{-1} \ \mathrm{d}\boldsymbol{x} = N_{goal}. \tag{14}$$

Note that the mesh complexity N_{goal} used as a constraint in the error minimization problem is equivalent to the number of elements of the optimized mesh (see Appendix C).
The optimization problem is solved using Lagrange multipliers theory as detailed in Appendix C, from which follows that the local interpolation error of an N_{goal} element optimal grid is

$$|u - \Pi_{\mathcal{M}} u|(\boldsymbol{x}) = C_n \left(\int_{\Omega} (\operatorname{tr}(|H|(\boldsymbol{x})))^{\frac{np}{2p+n}} \ \mathrm{d}\boldsymbol{x} \right)^{\frac{2}{n}} (\operatorname{tr}(|H|(\boldsymbol{x})))^{\frac{n}{2p+n}} N_{goal}^{-\frac{2}{n}}. \tag{15}$$

From the local error estimate, a global formulation is readily obtained as

$$||u - \Pi_{\mathcal{M}} u||_{L^p(\Omega)} = \left(\int_{\Omega} (|u - \Pi_{\mathcal{M}} u|(\boldsymbol{x}))^p \ \mathrm{d}\boldsymbol{x} \right)^{1/p} = C_n \left(\int_{\Omega} (\operatorname{tr}(|H|(\boldsymbol{x})))^{\frac{np}{2p+n}} \ \mathrm{d}\boldsymbol{x} \right)^{\frac{2p+n}{np}} N_{goal}^{-\frac{2}{n}}. \tag{16}$$

Comparing eqs. (10) and (16), we observe that, as expected, and because any $n \times n$ symmetric matrix with positive eigenvalues matrix A verifies $n\ \det(A)^{1/n} \leq \mathrm{tr}(A)$, the minimum error for grids containing only isotropic square/cubic elements is always larger than the values reported for grids containing anisotropic simplicial elements. For both, anisotropic and isotropic cases, the local error can be written as a local function that depends on the Hessian multiplied by the square of the averaged cell size $N_{goal}^{-2/n} = \overline{h}^2 / \left(\int_\Omega d\Omega\right)^{2/n}$ and a prefactor that depends on the L^p norm chosen for mesh optimization.

It is interesting to note that the compression ratio (equation (4)) can be computed for grids that minimize the interpolation error. First, the global error is a discrete sum of local errors

$$||u - \Pi_{\mathcal{M}} u||_{L^p(\Omega)} = \left(\int_\Omega (|u - \Pi_{\mathcal{M}} u|(\boldsymbol{x}))^p \ \mathrm{d}\boldsymbol{x} \right)^{1/p} \approx \left(\sum_{i=1}^{N} (|u - \Pi_{\mathcal{M}} u|_i^p h_i^n \right)^{1/p} \tag{17}$$

Suppose a grid with constant local error, then using equation (12), the local quantity $|u - \Pi_{\mathcal{M}} u|_i^p h_i^n$ implies that the quantity:

$$C_{\text{local}} = \ \mathrm{tr}(|H|(\boldsymbol{x}))h_i^{\frac{2p+n}{p}} = \frac{||u - \Pi_{\mathcal{M}} u||_{L^p(\Omega)}}{C_n\, N_{goal}^{\frac{1}{p}}} = \left(\int_\Omega (\,\mathrm{tr}(|H|(\boldsymbol{x})))^{\frac{np}{2p+n}} \ \mathrm{d}\boldsymbol{x} \right)^{\frac{2p+n}{np}} N_{goal}^{-\frac{2p+n}{np}} \tag{18}$$

is also a constant for all cells. This defines an optimal distribution of element size

$$h(\boldsymbol{x}) = \left(\frac{||u - \Pi_{\mathcal{M}} u||_{L^p(\Omega)}}{C_n\, N_{goal}^{\frac{1}{p}}\, tr(|H|(\boldsymbol{x}))} \right)^{\frac{p}{2p+n}} = \frac{(\int_\Omega (\,\mathrm{tr}(|H|(\boldsymbol{x})))^{\frac{np}{2p+n}} \ \mathrm{d}\boldsymbol{x})^{1/n}}{tr(|H|(\boldsymbol{x}))^{\frac{p}{2p+n}}} N_{goal}^{-n} \tag{19}$$

which, using equation (4) and the definition of the averaged grid size $\overline{h} = \left(\int_\Omega d\boldsymbol{x} / \int_\Omega h_i^{-n} d\boldsymbol{x} \right)^{1/n}$, can be also writen as

$$\frac{h(\boldsymbol{x})}{\overline{h}} = \frac{(\overline{tr(|H|)^{\frac{np}{2p+n}}})^{1/n}}{tr(|H|(\boldsymbol{x}))^{\frac{p}{2p+n}}}, \tag{20}$$

where the overline is used to denote volume average. The ratio between the grid size and the averaged grid size is a function that does not depend on the number of grid points (e.g. the element size) for all grids minimizing the interpolation error. Thus, the following value of the compression ratio (equation (4)) for optimal grids

$$\eta_0 = \frac{\overline{tr(|H|)^{\frac{np}{2p+n}}}}{\max(tr(|H|)^{\frac{np}{2p+n}})} \tag{21}$$

is an intrinsic property of the solution that is independent of the number of grid points. It is imposed by the ratio between the averaged value of the Hessian of the function and its maximum value, which can take small values for cases where the interpolation error is concentrated in small regions of the domain.

3.3. Propagation errors

Unlike interpolation errors, which depend on the structure of the Hessian of the solution only and not on the equation itself, the behavior of u' depends on the nature of the equation. In this study we focus on the Poisson-Helmholtz equation :

$$\mathcal{L}u = D\nabla^2 u + \lambda u = s\ . \tag{22}$$

with D a diffusion coefficient, λ a reaction coefficient and s the source term.
Introducing nodaly exact values of the source, the propagation error is then given by the solution of the discretized equation (8) as

$$D\tilde{\nabla}^2 u_i' + \lambda u_i' = -D(\tilde{\nabla}^2(\Pi_h u))_i - \lambda(\Pi_h u)_i + s_i. \tag{23}$$

Subscript i holds here, as well as for the rest of this paper, for values applied at node coordinate $\mathbf{x}_i$.
Adding and subtracting the exact value of the Laplacian on the right-hand side and using equation (5), the previuous expression writes

$$D\tilde{\nabla}^2 u_i' + \lambda u_i' = s_i' \tag{24}$$

where

$$s_i' = D\left[(\nabla^2 u)_i - \tilde{\nabla}^2(\Pi_h u)_i\right]$$

represents the source term in the error transport equation (24) and is proportional to the errors due to the numerical discretization of the Laplacian operator applied on the projection of the exact solution on a given grid.

Although the purpose of this work is not to model these errors, it is important to understand the behavior of this source term in order to clarify the limits of a purely local adaptation criteria. Let us consider the 1D version of the discretized Laplacian operator applied on the nodal exact interpolated values ($u_i = (\Pi_h u)_i$),

$$s_i' = D(\nabla^2 u)_i - D\frac{1}{h_i^2}\left(\frac{u_{i+1} - u_i}{\frac{1}{2}\left(\frac{h_{i+1}}{h_i} + 1\right)} - \frac{u_i - u_{i-1}}{\frac{1}{2}\left(\frac{h_{i-1}}{h_i} + 1\right)}\right).$$

This discretized formula can be further developed using Taylor expansion to express the values on the neighboring cells. For the cells not touching the boundary where the level of refinement is equal to all neighboring cells we readily find

$$s_i' = -\frac{1}{12} D u_{i,xxxx} h_i^2 + \mathcal{O}(h_i^4).$$

For 1D uniformly refined grids and $\lambda = 0$ we can integrate the equation (24) for the error to obtain

$$u' = -\frac{1}{12} u_{i,xx} h_i^2 + \mathcal{O}(h_i^4).$$

Thus, u' contains local errors that, like the interpolation errors, are proportional to the Hessian of the solution, which is consistent with previous works where u' is represented as a local error [38]. Note that for $\lambda \neq 0$, u' is no longer purely local, and a local interpolation error model will most likely be insufficient to fully quantify the total error. This situation is even more problematic when the level of refinement changes. In this case the leading order contribution of the source term becomes

$$s_i' = \frac{1}{2} D u_{i,xx}\left(1 - \frac{h_{i+1} + h_{i-1}}{2h_i}\right) + \mathcal{O}(h_i^2) \tag{25}$$

implying that the error term u' is controlled by the change of the grid size and the second derivative of the function. If we consider the limit of a continuous and sufficiently fast change of the level of refinement, an approximate model for the source can be obtained using the dimensionless density function $d_* = d\overline{h} = \overline{h}/h$ that does not depend on the number of grid cells. Using the approximation

$$1 - \frac{h_{i+1}+h_{i-1}}{2h_i} \approx -\frac{1}{2}h_{,xx}h = -\frac{(d_*^{-1})_{,xx}}{2d_*}\overline{h}^2 = \frac{\overline{h}^2}{2d_*^3}\left(d_{*,xx} - \frac{d_{*,x}^2}{d_*}\right) ,$$

the source term in the continuous limit can be modeled as

$$s'(x) \approx -D\frac{u_{,xx}}{4d_*^3}\left(d_{*,xx} - \frac{d_{*,x}^2}{d_*}\right)\overline{h}^2 .$$

In a grid where the level of refinement changes continuously, we can then write the general solution using the Green function G as

$$u' = -\frac{1}{4}D\overline{h}^2 \int G(x,\xi)\frac{u_{,\xi\xi}(\xi)}{d_*^3(\xi)}\left(d_{*,\xi\xi} - \frac{d_{*,\xi}^2}{d_*}\right)d\xi$$

which reveals that u' is proportional to the square of the grid size (e.g. it is a second order error) that it is no longer proportional to the Hessian of the function. Note that the family of grids sharing the same distribution of normalized grid size distribution d_* introduce the same error that is propagated in the solution. This is indeed the case for all optimal grids, for which this ratio is given by equation (20). When the error introduced in the regions where the level of refinement change is preponderant, we cannot expect that a correlation exist between the interpolation error and the error introduced by the change of the level of refinement, invalidating the use of any local adaptation criterion.

4. Evaluation of the sources of errors in two numerical examples

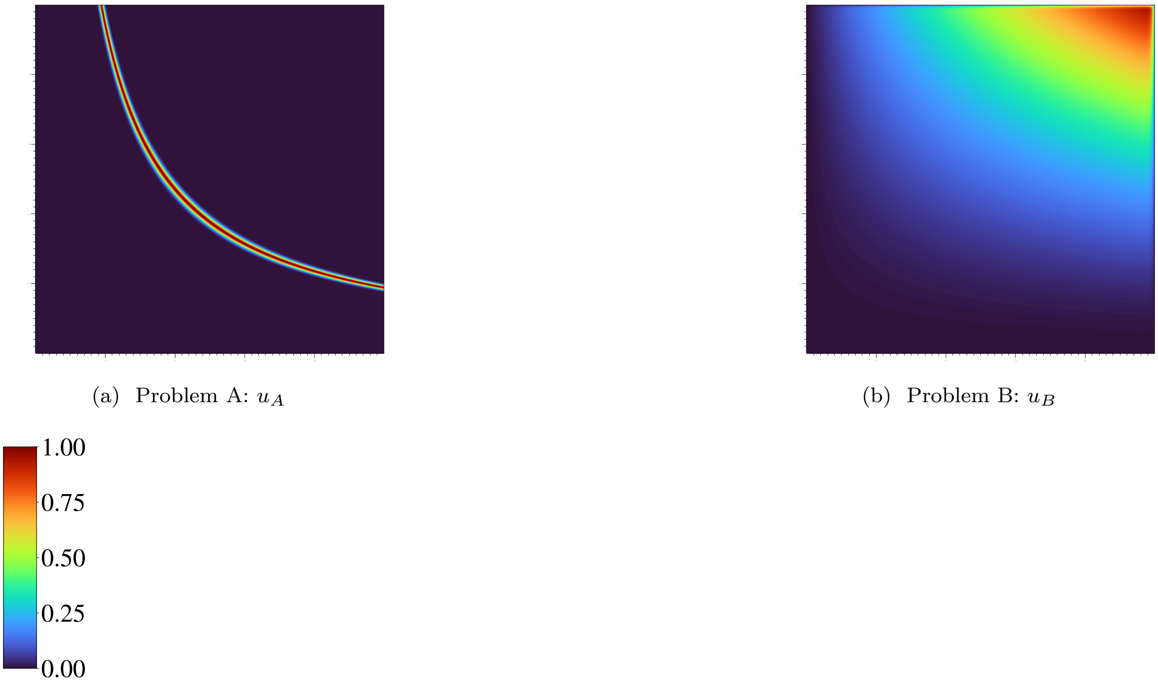
In this section, we focus on evaluating the relative importance of the two types of errors previously introduced - local and propagated - using two numerical examples. Let us first introduce the normalized error ratio

$$\gamma_p \equiv \frac{||u-\tilde{u}||_{\mathbf{L^p}(\Omega)}}{C_n\left(\int_\Omega (\,\mathrm{tr}(|H|(\boldsymbol{x})))^{\frac{np}{2p+n}}\,\mathrm{d}\boldsymbol{x}\right)^{\frac{2p+n}{np}} N^{-\frac{2}{n}}} , \tag{26}$$

as the ratio between the total error on a grid containing N points, $||u-\tilde{u}||_{\mathbf{L^p}}$, and the minimum interpolation error that can theoretically be obtained in an optimal grid with the same number of points (see equation (16)). This factor represents the relative increase of the total error obtained in a grid with respect to the minimum interpolation error achievable. This ratio is a loss factor such that by definition $\gamma_p = 1$ when the total error on the adapted mesh is equal to the theoretical minimal interpolation error for the same number of elements. Additionally, we define

$$\gamma_p^I \equiv \frac{||u-\Pi_h u||_{\mathbf{L^p}(\Omega)}}{C_n\left(\int_\Omega (\,\mathrm{tr}(|H|(\boldsymbol{x})))^{\frac{np}{2p+n}}\,\mathrm{d}\boldsymbol{x}\right)^{\frac{2p+n}{np}} N^{-\frac{2}{n}}} , \tag{27}$$

as the ratio between the interpolation error in any arbitrary grid containing N points and the minimum interpolation error that can be achieved with N grid points. This factor always verifies $\gamma_p^I \geq 1$ and it quantifies the loss of performance of a given grid with respect to the optimal grid in terms of interpolation error only. Therefore, $\gamma_p^I = 1$ for grids minimizing the interpolation error. In general, if the interpolation error is preponderant, γ_p^I is a good approximation of γ_p.



(a) Problem A: u_A

(b) Problem B: u_B

Figure 2: Representation of the two reference solutions for Problem A (left image) and respectively Problem B (right image)

We focus on two different numerical examples represented in figure 2 associated with the solution of the Poisson-Helmholtz equation with coefficients $D = 0.01$ and $\lambda = -1$ and where the source term s is computed such that the solution u is equal to a known function.
The first problem (problem A) is chosen as an illustrative example where we will clearly show below the problems associated to local adaptation criteria. The analytical solution is

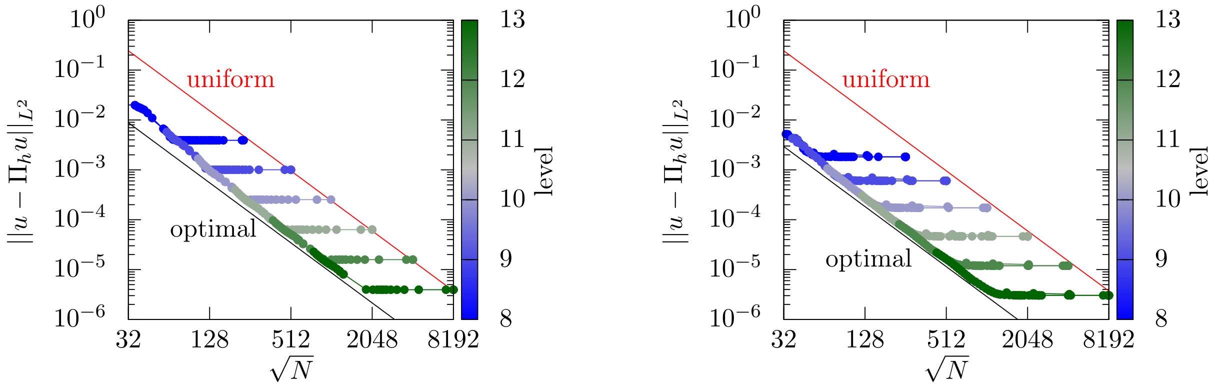
$$u_A(x, y) = \exp\left(-\left(\frac{xy - a}{\kappa^2}\right)^2\right) \tag{28}$$

with $a = 3\pi/50$ and $\kappa = \sqrt{1/(50\pi)} \approx 0.08$.
The second problem (problem B) is chosen as a known problem where local adaptation criterion has already been shown as an effective way to reduce the numerical errors [86, 87]:

$$u_B(x, y) = xy^2 - y^2 \exp\left(\frac{2(x-1)}{\kappa}\right) - x \exp\left(\frac{3(y-1)}{\kappa}\right) + \exp\left(\frac{2(x-1) + 3(y-1)}{\kappa}\right) , \tag{29}$$

with $\kappa = 10^{-2}$. The domain size for both problems is $(x, y) \in [0, 1]^2$.

Our objective in this section is not to test any grid adaptation strategy, but to evaluate the consequences of minimizing the interpolation error. To that end we perform the following coarsening study: starting with an initial uniform mesh of level l_{max} where the solution and the corresponding total and interpolation errors are calculated, a series of adapted meshes are obtained by coarsening the cells containing the smallest interpolation error. On each mesh, the numerical solution is computed by solving the discrete Poisson-Helmholtz equation, which allow us to compute the total numerical error obtained as the difference between the numerical and the analytical solution integrated in the $\mathbf{L^2}$-norm with a 5 points Gauss quadrature rule. At the same time, the analytical cell-centered solution is used to compute the exact interpolation error, which is also integrated in the domain using a 5 points Gauss quadrature rule. This



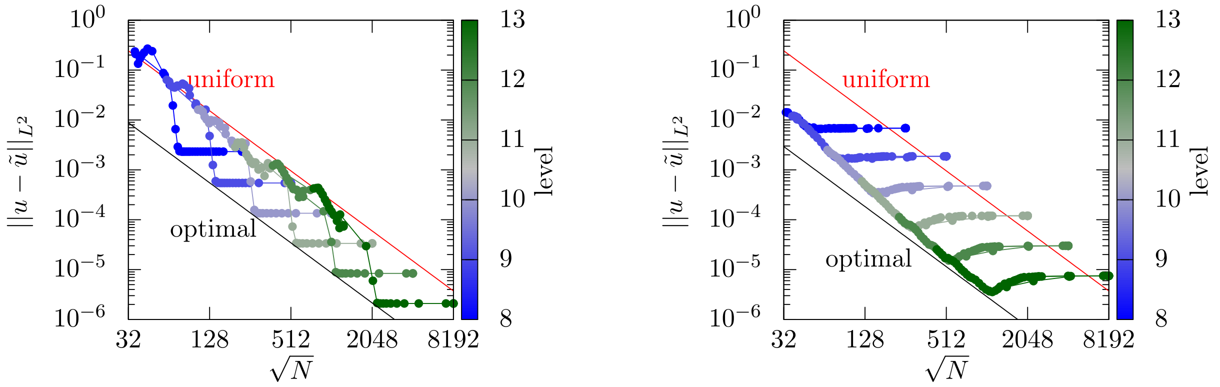
(a) Coarsening study results for problem A (u_A)

(b) Coarsening study results for problem B (u_B)

Figure 3: Interpolation error obtained for a coarsening study: an initially uniform mesh is successively coarsen based on the interpolation error. $l_{max} \in [\![8, 13]\!]$.

procedure allows us to monitor the evolution of both numerical and interpolation errors when reducing the number of grid elements.

Figure 3 illustrates the evolution of the interpolation error as a function of the number of elements on the various grids obtained for various experiments carried out with a different initial value of the maximum level of refinement imposed ($l_{max} \in [\![8, 13]\!]$). As reference, we include the theoretical convergence curves for a uniform grid (plain red curve, obtained in equation (B.2)) and the isotropic metric-based estimation of the optimal interpolation error convergence curve (plain black curve, obtained in equation (16)). We observe that for nearly uniform grids, most of the elements reach the imposed minimum grid size and the interpolation error is almost insensitive to the number of elements. This regime spans until the error is close to the minimal interpolation error curve where a smooth transition occurs tending to follow the optimal convergence curve theoretically obtained for quadtree grids. In both cases, when the grid is uniform, the interpolation error is well predicted by the theory. The theory also predicts well the minimum interpolation error for a given number of grid points N, the accuracy of the prediction being better as we increase the number of elements.

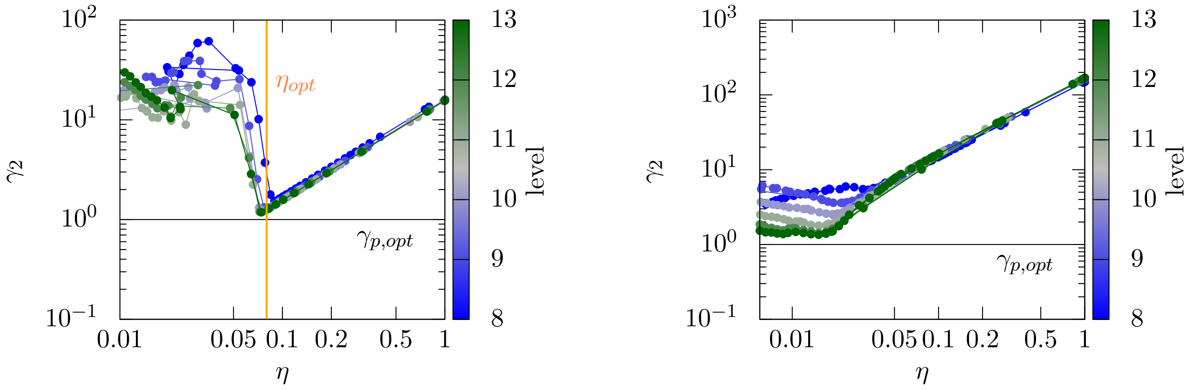


(a) Coarsening study results for problem A (u_A)

(b) Coarsening study results for problem B (u_B)

Figure 4: Total numerical error obtained for the coarsening study: an initially uniform mesh is successively coarsen based on the interpolation error. $l_{max} \in [\![8, 13]\!]$.

Figure 4 represents the $\mathbf{L^2}$-norm of the total error computed on the same grids than those where the interpolation error is analysed. For grids close to a uniform grid, the behavior of the total error is nearly identical to that of the interpolation error, showing a regime where the total error is almost insensitive to the total number of cells. Still it is worth noting that the total error obtained for the problem B is larger than the interpolation error even for cartesian grids, pointing to the existence of the additional source of error. As we approach the optimal curve, the behavior of the total error is markedly different in the two considered examples. While in problem A the total numerical error increases significantly leading to results where the total error is one order of magnitude larger than the optimal convergence curve, in the problem B the total numerical errors remain close to the optimal convergence curve. These observations, which hold irrespective of the maximum level of refinement imposed, indicate that in problem A the total error is not always correlated with the local interpolation errors preventing the direct application of a refinement criterion based on the local interpolation error. Instead, for problem B the total error seems to be always correlated to the interpolation error even for nearly optimal grids.



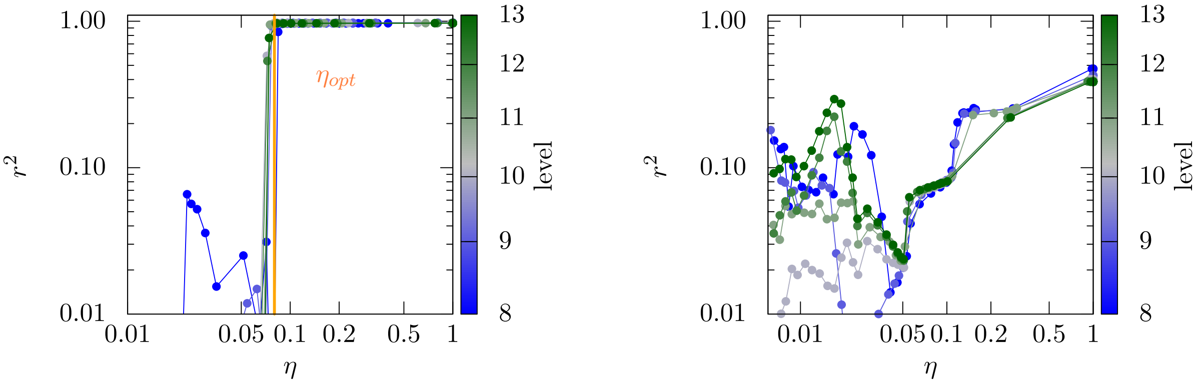
(a) Problem A: Normalized error γ_2 versus compression ratio η for all the grids represented in figure 4a

(b) Problem B: Normalized error γ_2 versus compression ratio η for all the grids represented in figure 4b

Figure 5: Normalized error obtained for coarsening studies: an initially uniform mesh is successively coarsen based on the interpolation error.

Figure 5 shows the normalized total error γ_2 obtained in the L_2 norm, as a function of the compression ratio η. An interesting remark is that all curves superpose relatively well irrespective of the minimum grid size imposed, showing how these quantities are well suited to describe the behavior of the error. The shape of the obtained function is problem dependent: while for problem A a clear minimum value of the normalized error γ_2 appears for a nearly constant value of the compression factor, for problem B the normalized error is almost a monotonically decreasing function of η. In general, when starting to coarsen a uniform grid (e.g. values of η close to one), the normalized error decreases quickly when decreasing the value of η until reaching values of γ_2 close to the optimal value. This can be identified with a group of grids where the minimization of the local interpolation error is an effective tool to reduce the total error. However, below a critical threshold, the total numerical error can significantly increase despite the reduction of the local interpolation error (figure 5a). This behavior is attributed to the drastic loss of correlation between the local interpolation error and the local total error, as we can see in figure 6. Despite the fact that u' is non-zero even for cartesian grids, both errors remain relatively well correlated in problem B for all values of η, while we observe a drastic loss of correlation below a critical value of η for problem A. This invalidates the use of any strategy based on local error adaptation to guide the optimization of grids in this region.

An example of an optimal grid minimizing the total error and respectively the interpolation error using the same number of elements is shown in figure 7. The cell size distribution is narrower for the grid minimizing the total error ($l \in [\![5, 12]\!]$) than for the grid which minimizes the interpolation error ($l \in [\![3, 13]\!]$). This leads to a significant source of errors in the regions



(a) Problem A

(b) Problem B

Figure 6: Linear regression coefficient between the total numerical error and the interpolation error for all the grids included in figure 4.

where the level of refinement changes, that are not correlated to the interpolation errors as discussed in Section 3.3. Thus, while in the case of the optimal mesh minimizing the total error, the error is concentrated in regions where the element size is minimal, in the case of the optimal mesh minimizing the interpolation error, the total error is maximal in the regions where the element size varies. One can also note that the amplitude of the local total errors is significantly larger in the last case, which is in agreement with the results observed in figure 4.

This observation is confirmed in figure 8, which shows the evolution of the interpolation error starting from a mesh minimizing the interpolation error (figure 7) and introducing a constraint on the minimum grid size. Then, a series of AMR iterations are performed until the number of elements of the final mesh is equivalent to the number of elements of the starting mesh. We clearly see in figure 8 that, by reducing the minimal cell size, the total error is reduced by around one order of magnitude at the cost of an increase of the interpolation error. These results clearly show that the mesh refinement criterion based only on the local representation of the interpolation error is not sufficient to successfully reduce the total error.

5. An adaptive refinement criterion under constrain

5.1. Refinement criterion

In order to achieve the desired number of grid points, we need to establish a criterion to divide or merge the cells. In the cells where the interpolation error is preponderant, the interpolation error is proportional to a local function depending on the Hessian of the function and the square of the element size. When the targeted number of grid points N_{goal} is imposed, we update the value of C_{local} (equation (18)) using the following iterative procedure

$$\log(C_{local}^{(j+1)}) = \log(C_{local}^{(j)}) + \frac{d\log(C_{local})}{dN}(N_{goal} - N^{(j)})$$

where $N^{(j)}$ is the number of grid points at the current iteration j and $\log(C_{local})$ is computed as

$$\log(C_{local}) = \log\left(\frac{||u - \Pi_h u||_{L_p}}{C_n N^{1/p}}\right)$$

to obtain

$$C_{local}^{(j+1)} = C_{local}^{(j)} \exp\left[-\frac{1}{p}\frac{N_{goal} - N^{(j)}}{N^{(j)}}\right]. \tag{30}$$

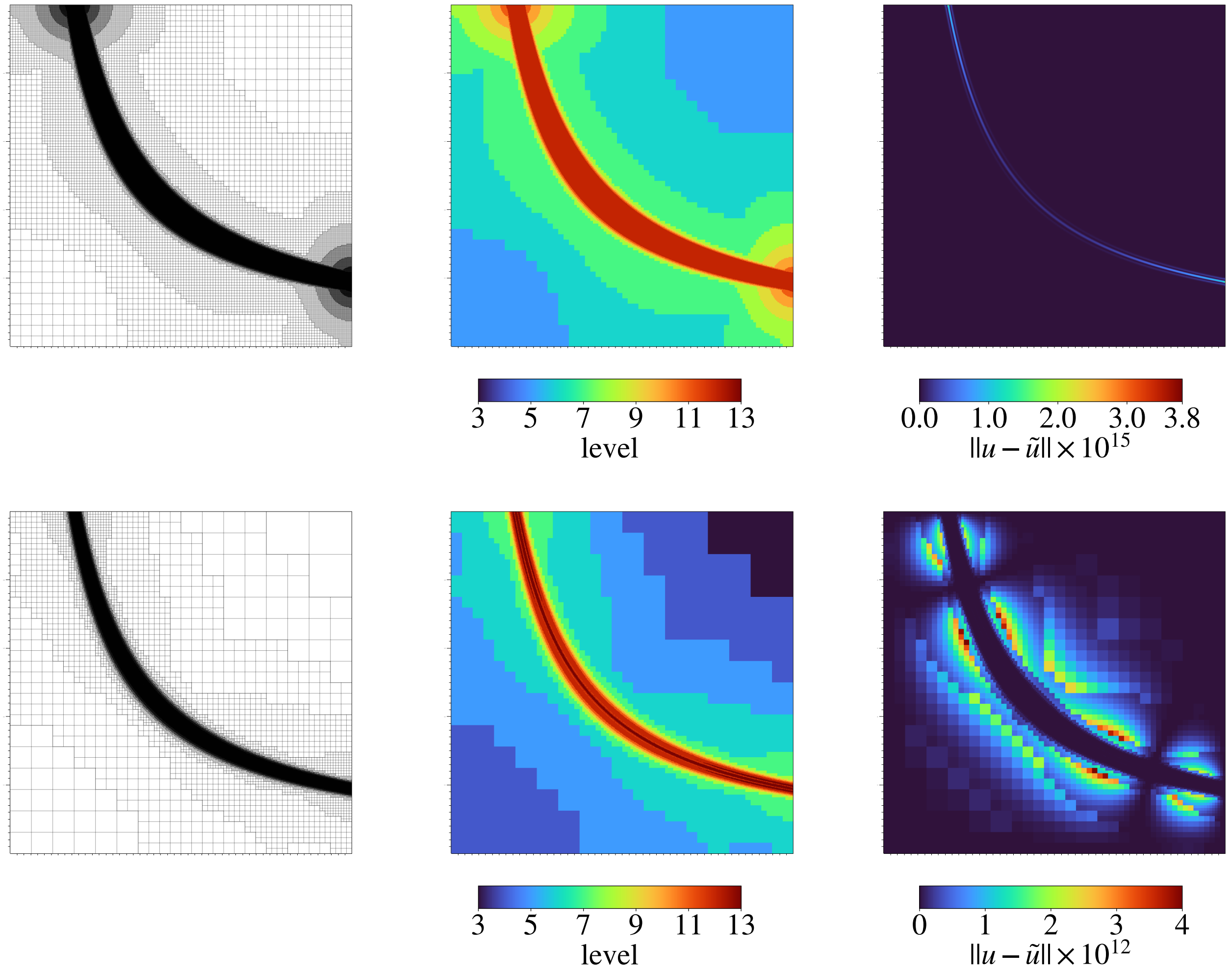


Figure 7: Illustration of grids (left), the corresponding grid levels (middle) and the local total error (right) obtained with the coarsening study. (Top) Optimal grid minimizing the total error (1 512 157 cells) and (bottom) optimal grid minimizing the interpolation error (1 578 697 cells).

These conditions are sufficient to drive the process of mesh adaptation in the absence of any additional constrain on the algorithm, leading to grids that minimize the interpolation error with a given number of grid points N_{goal}.

5.2. Mesh compression constrain

We have seen that the minimization of the total error is characterized by an optimal compression ratio η_{opt}, which is *a priori* unknown. We have concluded from an illustrative example that meshes obtained after minimizing the interpolation error can produce results with poor numerical performances as a consequence of the small values of the compression ratio associated with these grids. This issue is similar to mesh gradation problems already observed by other authors [85, 88] and shows to be directly linked to the generation of numerical errors in the regions where the level of refinement change.

In those grids where u' is preponderant with respect to the interpolation error, there is no guarantee that the minimization of the interpolation error leads to grids that minimize the total numerical error. It is then required to correct the strategy of grid adaptation, in order to restrict the use of local grid adaptation strategies only to situations where the error propagated in the solution is not significant. As we have seen, one effective strategy to discriminate among all possible grids those where the propagation error can control the total error is to use the compression ratio. Thus, the problem is reduced to finding the value of the compression ratio

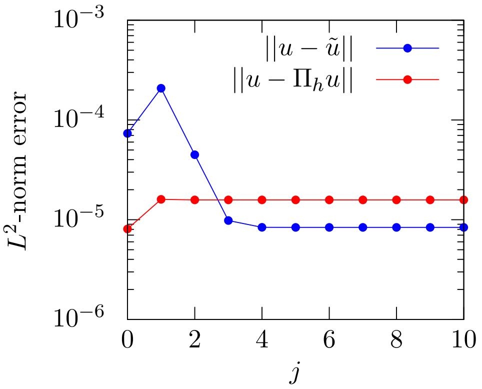


Figure 8: Evolution of the interpolation error and the total numerical error as a function of the iteration number. The starting mesh is the result of the interpolation error minimization problem for a grid with $N = 1512157$ elements (figure 7). Then, we impose the maximal cell level to $l_{max} = 12$ and we apply recursive AMR steps until converging to a mesh with the given number of elements N_{goal}

below which the total error can be eventually controlled by non-local errors. We note that imposing the minimum value of the compression ratio as defined in equation (3), is equivalent to imposing the minimum grid size h_{min} during the grid adaptation strategy for a fixed number of grid points N_{goal}:

$$h_{min} = L_0 \left(\frac{\eta}{N_{goal}} \right)^{\frac{1}{n}} , \tag{31}$$

where, for quad/octree grids, the minimum grid size is given by the maximum level of refinement

$$l_{max} = \frac{1}{n} \log_2 \left(\frac{N_{goal}}{\eta} \right) \tag{32}$$

which can only take discrete integer values for quad/octree grids .

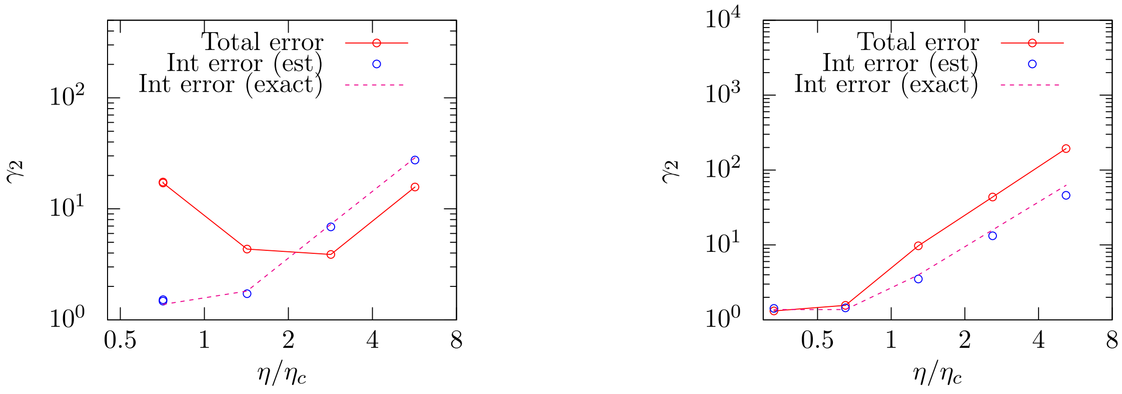
Although the optimal value of η for which the total error is minimized depends on the capability to predict u', it is possible to propose a simple constraint on the minimum grid size using the conclusions extracted from the numerical experiments reported in figure 3, where we have seen that the minimization of the interpolation error starting from a cartesian grid with $h = h_{min}$ leads to a roughly iso-error transition between a cartesian grid and an optimal grid minimizing the interpolation errors. To obtain an estimation of the optimal value of the compression ratio that ultimately defines l_{max} we introduce a characteristic compression ratio, η_c, obtained by balancing the error of an optimal grid (equation (16)) and a uniform grid (equation (B.2)):

$$C_n\, L_0^2 \left(\int_\Omega \mathrm{tr}(|H|(\boldsymbol{x}))^p \;\mathrm{d}\boldsymbol{x} \right)^{\frac{1}{p}} N_{h_{min}}^{-\frac{2}{n}} = C_n \left(\int_\Omega \left(\mathrm{tr}(|H|(\boldsymbol{x})) \right)^{\frac{np}{2p+n}} \;\mathrm{d}\boldsymbol{x} \right)^{\frac{2p+n}{np}} N_{goal}^{-\frac{2}{n}}$$

which gives

$$\eta_c = \frac{N_{goal}}{N_{h_{min}}} = \frac{\left(\int_\Omega \left(\mathrm{tr}(|H|(\boldsymbol{x})) \right)^{\frac{np}{2p+n}} \;\mathrm{d}\boldsymbol{x} \right)^{\frac{2p+n}{2p}}}{L_0^n \left(\int_\Omega \mathrm{tr}(|H|(\boldsymbol{x}))^p \;\mathrm{d}\boldsymbol{x} \right)^{\frac{n}{2p}}} . \tag{33}$$

Like η_0, the value of η_c is an intrinsic property of the sensor u that can be obtained analytically for any given function and it depends on integrated quantities based on the Hessian of the function, the dimension of the problem and the error norm.



(a) Problem A

(b) Problem B

Figure 9: Total (exact) error and interpolation error (exact and estimated) after the solution of the interpolation minimization problem without/with constrain as a function of the minimum grid size of the resulting grid. The results are obtained for the results for $N = 2^9 \times 2^9$ elements.

In general, the loss of performance in terms of interpolation error only is expected to be small for $\eta < \eta_c$, while an exponential increase in the interpolation error is expected for $\eta > \eta_c$. This is confirmed in figure 9, where, for a given number of grid points ($N_{obj} = 2^9 \times 2^9$ cells), we show the resulting normalized error for grids obtained with and without imposing the minimum grid size (e.g. minimum compression ratio) in the interpolation minimization algorithm. We remark that the smallest value of the compression ratio tested is freely selected by the interpolation error minimization algorithm without any explicit restriction on the minimum grid size. As expected, the interpolation error decreases as the minimum grid size is reduced in both problems considered. However, the total error behaves very differently in each case. While in problem A a clear minimum on the total error is observed, the total error monotonically increases with the minimum grid size in problem B. Still, close to $\eta/\eta_c = 1$, we observe a clear correlation between the interpolation error and the total error in both problems and only below a critical threshold of the compression ratio and for problem A, we observe the loss of correlation between both errors described previously.

The results of figure 9 also indicate that in the absence of any model to predict the response of the propagation errors, a first estimation of the optimal compression ratio is $\eta_{\text{opt}} \approx \eta_c$. The minimization of the local interpolation error under this constraint leads to a family of suboptimal solutions that are close to the minimum interpolation error for an imposed number of grid points N_{goal} but that significantly reduces the total error. In quad/octree grids, the problem of minimizing the total error is reduced to choose among a typically small number of grids obtained for a fixed number of elements after minimizing the local interpolation error under a constraint on the minimum grid size. This aspect will be discussed in the last section of this manuscript.

5.3. Algorithm

The algorithm in diagram 1 below summarizes the basic steps of the adaptation criteria proposed and implemented in Basilisk solver. Using as inputs the error norm p, the number of desired points N_{goal} and the maximum level of refinement l_{max}, the algorithm provides the grid that minimizes the interpolation error. Note that the value of l_{max} can be imposed by equation (34) using an estimation of η_c (equation (33)) obtained from a currently available estimation of the solution. In the absence of this restriction, the algorithm provides the grid that minimizes the interpolation error. The additional constraint is thus imposed as input (through

l_{max}) of the algorithm.

Another important ingredient of the algorithm is the imposition of the 2:1 cell level transition constrain (see section 2). This feature, inherent to the Poisson solver implemented in Basilisk for adaptive grids, has the advantage to naturally limit the source of errors introduced due to the change of level of refinement.

```
Given an initial mesh $\mathcal{M}_0$, the error norm $p$, $l_{max}$ and $N_{goal}$;
while $N \neq N_{goal} \pm 3\%$ do
    Compute the solution $S_i$ on $\mathcal{M}_i$ ;
    foreach element do
        Compute $C_{local}$;
        if $C_{local}(\boldsymbol{L}^p) > \varepsilon$ and $l < l_{max}$ then
            Refine the cell;
        else
            if $C_{local}(\boldsymbol{L}^p) < \varepsilon/4$ then
                Coarsen the cell;
            end
        end
    end
    Impose 2:1 cell level transition ;
    Count the number of elements $N$ ;
    Update $\varepsilon$;
end
```

Algorithm 1: Adaptation loop.

6. Numerical test cases

In this section, we evaluate the performance of the new algorithm presented previously. The mesh refinement criteria is based on pure local interpolation errors computation and we perform the mesh adaptation using both no constraints on the minimum grid size and a variable constraint on the minimum grid size. Unlike the numerical results presented previously, here we use an estimation of the interpolation error based on a currently available numerical solution, rather than evaluating the interpolation error using the analytical solution of the problem.

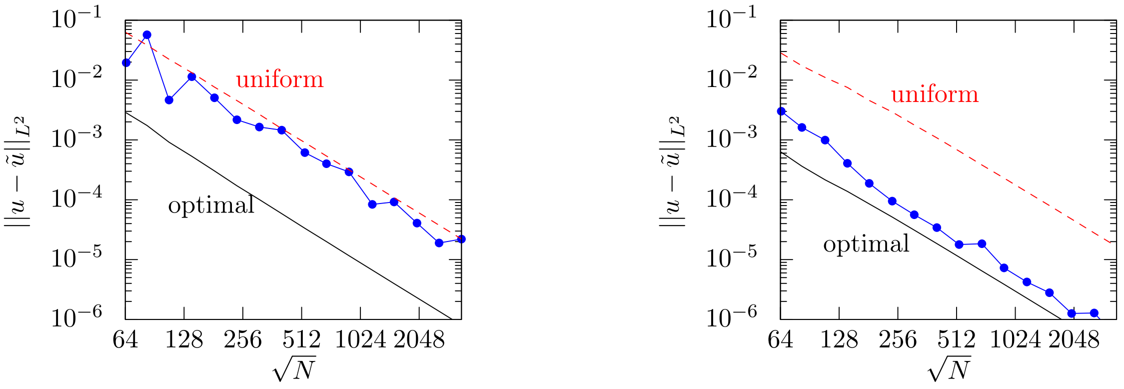
6.1. Two dimensional tests

We investigate here the performance of the mesh adaptive method applied to Problems A and B introduced in Section 4.

Figure 10 shows the total error convergence curves obtained for various grids obtained after the minimization of the interpolation error without constraint. For problem A, we observe that the grids obtained have errors that are almost one order of magnitude larger than the optimal (minimum) values of the interpolation error theoretically estimated while the algorithm provides nearly optimal grids for problem B. These results are consistent with the numerical experiments reported in the previous section, where we have seen that the minimization of the purely local error leads to grids with large numerical errors in problem A.

Figure 11 shows the results obtained after imposing on both problems the constraint on the maximum level of refinement defined as:

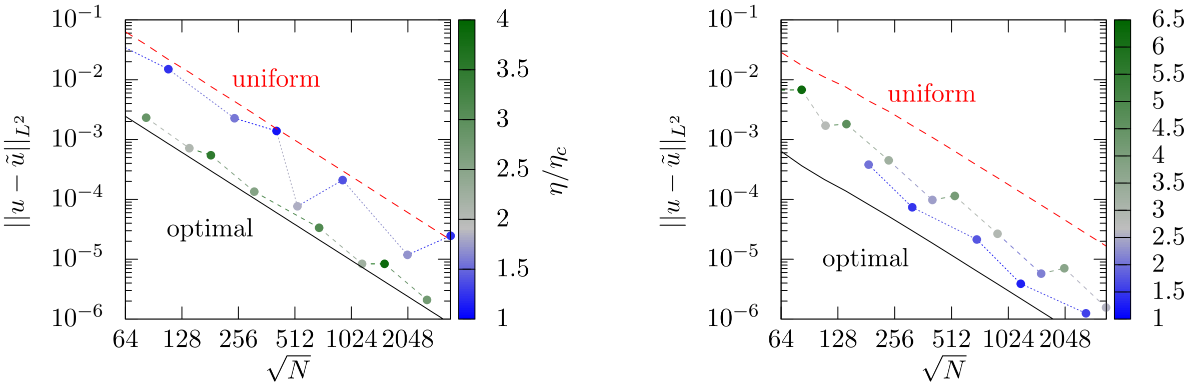
$$l_{max} = \frac{1}{n} \log_2 \left(\frac{N_{goal}}{\eta_c} \right) \tag{34}$$



(a) Problem A

(b) Problem B

Figure 10: Results of the convergence study based on the interpolation error minimization without additional constrain.



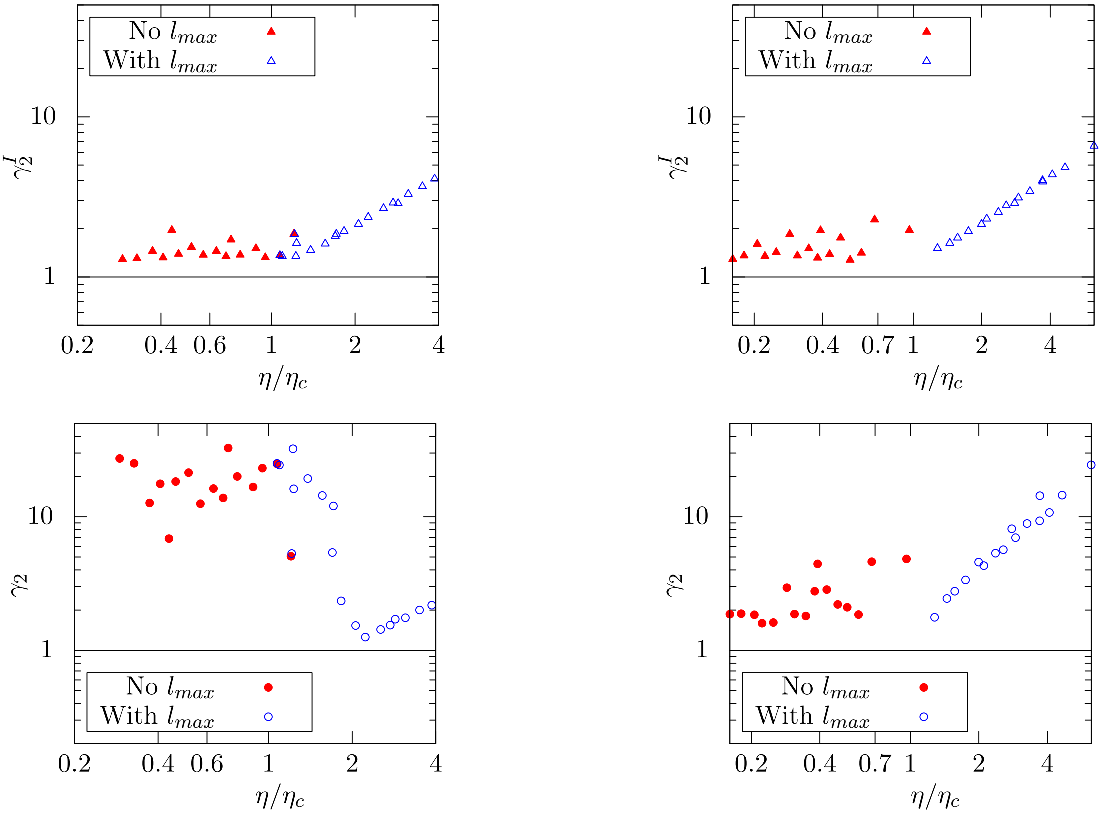
(a) Problem A

(b) Problem B

Figure 11: Results of the convergence study when imposing a restriction on the maximum refine level (minimum grid size) according to eq. (34) and a restriction on the compression ratio $\eta > \eta_c$. The dotted lines distinguish the meshes respecting $\eta > 1.5\,\eta_c$ and $1.5\,\eta_c > \eta > \eta_c$ for ease of understanding.

with η_c the value of the compression ratio numerically evaluated using equation (33) from the currently available solution. Since in quadtree grids l_{max} must be an integer value, we show the results after generating grids with the floor and also the ceil values obtained from the expression above. We add also a constrain on the compression ratio of the generated meshes: $\eta > \eta_c$. The fact that in quadtree grids the maximum level of refinement l_{max} is an integer – which prevents imposing a constant value of the compression ratio $\eta = \frac{N}{2^{nl_{max}}}$ – has important consequences in terms of the performance of quadtree/octree grids for solutions where the numerical error strongly depend on η (as illustrated in problem A). Introducing a constrain on the value of the compression ratio implies not only to impose the minimum grid size, but also limiting the subrange of grids for which the condition $\eta > \eta_c$ can be reached.

Figure 12 shows the results of the convergence analysis presented in figures 10 and 11 as a function of η/η_c for both, the total error (circles) and the interpolation error (triangles). Consistently with the results shown previously, all the grids for different N collapse into a single curve which is problem dependent where a clear minimum on the total error is observed for values of the compression ratio η that are close to η_c ($\eta_{opt}/\eta_c \approx 2.25$ for problem A and $\eta_{opt}/\eta_c \approx 1$ for problem B). In the absence of any constrain on the grid size, the algorithm provides grids with the smallest values of the compression ratio in both problems, thus minimizing the interpolation error but having dramatic consequences on the effective performance of the grid efficiency γ_2 in problem A. In this figure we can also see that the introduction of a constraint on the



(a) Problem A

(b) Problem B

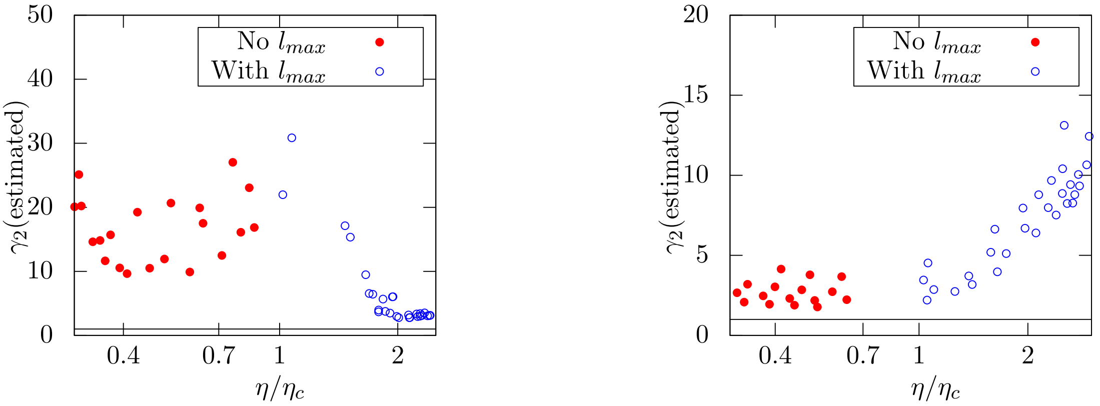
Figure 12: Results of the convergence study shown in Figures 10 -11 in a γ_2 vs. η/η_c map. In circles we show the results of the total error efficiency factor γ_2, and with triangles we show the interpolation error efficiency factor γ_2^I.

minimum grid size imposes larger values of η which penalizes the interpolation errors obtained but includes optimal grids in terms of the total error minimization.

In order to accurately impose the compression ratio restriction, a numerical estimation of the $\gamma_2 - \eta/\eta_c$ curve can be obtained as follows. Starting from any arbitrary fine grid generated either with or without the minimum cell size constrain, we can compute the solution in different coarser grids generated with and without a constraint on h_{min} as well as an estimation of the prefactor on the interpolation error. In this case, an estimation of the total error is

$$||\tilde{u} - u||_{L^p} \approx ||\tilde{u} - \tilde{u}_{\text{fine}}||_{L^p} \ ,$$

from which we can readily compute an estimation of the γ_2 factor. Because the function $\gamma_2(\eta/\eta_c)$ is shown to be independent of the number of grid points, coarse grids can be used to compute quickly the optimal value. Figure 13 represents the estimation of the $\gamma_2(\eta/\eta_c)$ function from the solution obtained in a $N_{fine} = 2^{10} \times 2^{10}$ grid resulting from the minimization of the interpolation error without constrain. From this solution we generate around 30 coarser grids with N ranging from 2^{18} to 2^{12} using the algorithm proposed without any constraints on the minimum grid size, and also with a constraint on the maximum level of refinement imposed by the floor and ceil values resulting from equation (34) and the constraint on the compression ratio value $\eta > \eta_c$. We can see that the curves are relatively well predicted in both cases. While in problem A a plateau with minimum values of γ_2 is observed for $\eta \approx 2\,\eta_c$, the smallest values of γ_2 in problem B are obtained for $\eta \leq \eta_c$. Note that for the grids with the lowest compression ratio in problem A, the estimated value of γ_2 is heavily influenced by the local mesh topology and the error generated due to the changes in the grid size that eventually propagates along the



(a) Problem A

(b) Problem B

Figure 13: Estimation of the interpolation error vs η/η_c using as reference the solution obtained in an unconstrained grid with $2^{10} \times 2^{10}$ cells.

domain. This leads to a large variance in the values of γ_2 observed in figure 13 (a).

We then conclude that in the absence of any model or method to estimate the contribution of the interpolation error, the introduction of a constraint on the minimum grid size based on $\eta \approx \eta_c$ seems to be a reasonable estimation of the optimal value of the compression ratio for which the total error is minimized. However, in problems where the total error strongly depends on η, the accurate estimation of η_{opt} is critical in order to choose the grids that minimize the total error. This optimal value strongly depends on the behavior of u' and can be estimated numerically by comparing the solution obtained on a fine grid with that obtained on several coarser grids. Otherwise, the development of an accurate model to predict u' in the absence of the exact solution is not straightforward and should be the focus of more detailed studies.

6.2. Three-dimensional test

We propose to use the estimation of the optimal compression ratio for which the total error is minimized using a constraint in the AMR procedure and to validate it on a 3D case. We solve the Poisson-Helmholtz equation in a domain $[-0.5, 0.5]^3$ with $D = 10^{-2}$ and $\lambda = -1$ and source term defined as:

$$s(x,y,z) = \frac{1}{\kappa^4} e^{-\frac{x^2y^2z^2}{\kappa^2}} \left(-2D\kappa^2 \left(x^2\left(y^2+z^2\right)+y^2z^2\right) + 4Dx^2y^2z^2\left(x^2\left(y^2+z^2\right)+y^2z^2\right) + \kappa^4 s\right) \tag{35}$$

with $\kappa = 1/(50\pi)$. The solution of this problem,

$$u_{3D}(x,y,z) = \exp\left(-\left(\frac{xy+z^2}{\kappa}\right)^2\right), \tag{36}$$

is similar to the 2D exponential function u_A previously studied, and intuitively the total error should have similar behavior.

First, we estimate the optimal compression ratio. Figure 14 represents the estimation of the $\gamma_2(\eta/\eta_c)$ function from the solution obtained in a grid with $N_{fine} = 2^{21}$ cells resulting from the minimization of the interpolation error without constrain. From this solution we generate around 30 coarser grids with N ranging from 2^{18} to 2^{15} cells using the algorithm proposed without any constrain on the minimum grid size, and also with a constrain on the maximum level of refinement imposed by the floor and ceil values resulting from equation (34). We can

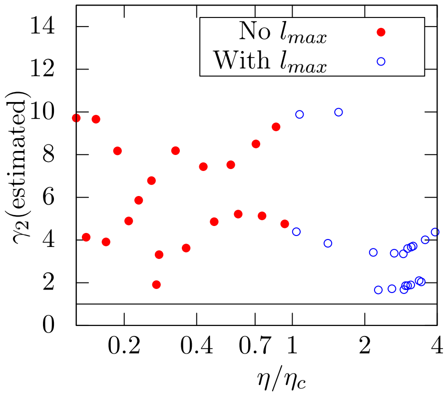


Figure 14: Estimation of the interpolation error vs η/η_c using as reference the solution obtained in an unconstrained grid with 2^{21} cells.

see that the minimum value is observed for $\eta \approx 2.5\eta_c$.

Then, we search for a series of adapted meshes respecting l_{max} from equation (34) (floor value) and imposing $2 < \eta/\eta_c < 6$. The error convergence in function of the cubic root of the number of element is presented in figure 15. We see that following this procedure, we successfully obtain adapted meshes with a nearly optimal total error. As a comparison, the error convergence curve when using mesh adaptation without imposing a maximum level of refinement is also presented in figure 15. One can immediately observe that the error obtained on these latter meshes is around ten times higher than the expected optimal error, and thus around ten times higher than the error obtained on the size-constrained meshes. An illustration of meshes obtained with and without minimial cell size constraint is shown in figure 16. It may immediately be concluded that the element size repartition is different in both cases, as finer elements may be observed in the unconstrained case. These observations validate the performance of the proposed AMR method.

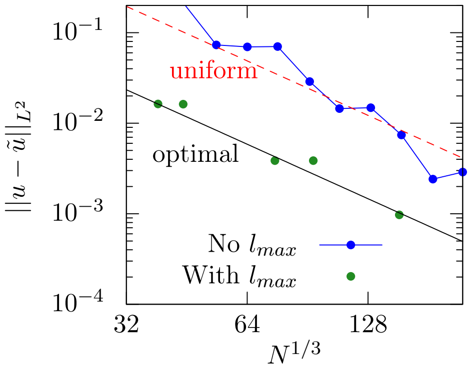


Figure 15: Results of the convergence study with or without imposed restriction on the maximum level of refinement (or minimum grid size).

7. Conclusion

We have proposed a new adaptive mesh refinement strategy to reduce the total error commited when solving numerically the Poisson-Helmholtz equation. The method is based on the Riemannian metric theory extended to quad/octree grids where an additional constraint for the minimum element size (or maximum level of refinement) is imposed. The introduction of

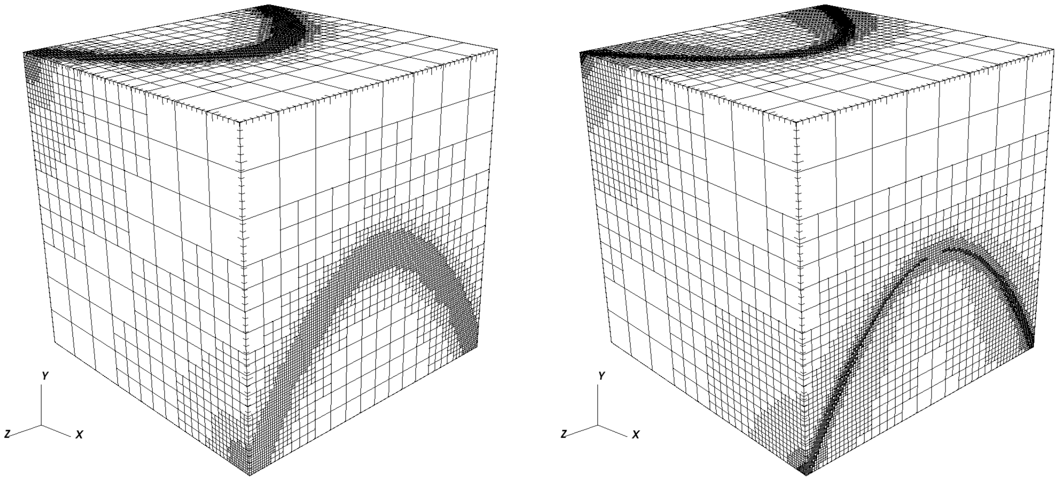


Figure 16: Adapted meshes. Left: 422 444 elements, with restriction on l_{max} (e.g. minimizing the total numerical error). Right: 442 660 elements, no restriction on l_{max} (e.g. minimizing the interpolation error).

this additional constrain in the minimization problem is motivated by the observation that for some problems, there exists a family of grids in which the error introduced in the numerical solution does not show correlation with the local interpolation error used as refinement criteria.

We show that the ratio between the total numerical error and the optimal interpolation error for a grid of N points is only a function of the structure of the solution and the compression ratio, and independent of the number of elements of the grid. For compression ratios below a given characteristic value η_c the additional constraint on the minimum element size is shown to effectively restrict the search domain to grids where the numerical error is more likely to be proportional to the interpolation error. An approximation of the value of the optimal compression ratio is theoretically obtained. This value is shown to be a function of the Hessian of the solution and the number of grid points imposed and free of any other user-defined parameter. In practice, a prefactor of order unity can be numerically evaluated to optimize the performance of the algorithm for problems where the numerical error strongly depends on the compression ratio.

Future research could consider several paths to extend this work. First, an extension to projection-based Navier-Stokes solvers [89] would increase the range of application of the present study. Preliminary results presented in [90] seem promising towards this objective. Then, the error estimate used as refinement criterion to drive the mesh adaptation procedure could be enriched with added contribution from the source term of the propagating error equation. This source term contains indeed two major ingredients: the hessian of the solution, which is similar to the interpolation error, and the element size variation, which can be responsible for the non-correlation between the numerical error and the interpolation error. As it takes into account the element size variation, it may be possible to remove the additional minimal cell size constrain. Finally, the case of problems with singular solutions to the equations should be studied carefully, as it may reduce the order of convergence of the numerical error, and thus change the behavior of the numerical error.

Appendix A. Continuous interpolation error for square elements

In this section, we derive the local continuous interpolation error for square/cubic elements, which differs slightly from the local continuous interpolation error for triangular/tetrahedral elements presented by Loseille and Alauzet [83, 84]. We first present the proof for square elements in 2D and refer to [90] for the proof for 3D cubic elements problem. This proof is based on the direct calculation of the discrete interpolation error on the considered elements, and the equivalence between continuous and discrete interpolation errors.

We start deriving the discrete $\boldsymbol{L}^1$-norm interpolation error estimate for a square element K represented by its vertices $(\boldsymbol{v}_i)_{i\in[\![1,4]\!]}$. In order to find the discrete bilinear interpolation error, we use an exact point-wise error estimate of the interpolation error within an element of reference K_{ref} and we apply a change of variables to obtain the error on K from the error on K_{ref}. The analogy between discrete and continuous interpolation error is used to provide the continuous interpolation error as in [83, 84].

The reference element is a square element with unit edge lengths. Its vertices are noted $\hat{\boldsymbol{v}}_1 = (0,0)$, $\hat{\boldsymbol{v}}_2 = (1,0)$, $\hat{\boldsymbol{v}}_3 = (0,1)$ and $\hat{\boldsymbol{v}}_4 = (1,1)$. The transformation between a coordinate $\boldsymbol{x} \in K$ and $\hat{\boldsymbol{x}} \in K_{ref}$ is given by

$$\boldsymbol{x} = \boldsymbol{v}_1 + h\hat{\boldsymbol{x}} \tag{A.1}$$

with h the edge length of K.

Let $u(\boldsymbol{x}) = \boldsymbol{x}^T H \boldsymbol{x}$ be a quadratic function exactly valued at the vertices of the cube and represented by its associated Hessian $H = \begin{pmatrix} a & d \\ d & b \end{pmatrix}$. We pose Π_h the bilinear interpolate operator. In the framework of K_{ref}, u reads

$$u(\boldsymbol{x}(\hat{\boldsymbol{x}})) = \frac{1}{2}\boldsymbol{v}_1^T H \boldsymbol{v}_1 + \frac{1}{2}\boldsymbol{v}_1^T H h\hat{\boldsymbol{x}} + \frac{1}{2}\hat{\boldsymbol{x}}^T h H \boldsymbol{v}_1 + \frac{1}{2}h^2 \hat{\boldsymbol{x}}^T H \hat{\boldsymbol{x}} \ . \tag{A.2}$$

Linear and constant terms of $u(\boldsymbol{x}(\hat{\boldsymbol{x}}))$ are exactly interpolated. Therefore, without loss of generality, we consider only the quadratic term $\hat{u}(\boldsymbol{x}) = \frac{1}{2}h^2\hat{\boldsymbol{x}}^T H \hat{\boldsymbol{x}}$, as $(u - \Pi_h u)(\boldsymbol{x}) = (\hat{u} - \Pi_h \hat{u})(\boldsymbol{x})$. To improve readability, we write u instead of $\hat{u}$ in the following.

For all $\boldsymbol{x} = (x,y,z)^T \in K_{ref}$, the trilinear interpolate $\Pi_h u$ of the function u is

$$\Pi_h u(\boldsymbol{x}) = \alpha + \lambda x + \mu y + \beta xy \ . \tag{A.3}$$

As the solution is exact-valued at the vertices, we have

$$\begin{cases} \Pi_h u(\boldsymbol{v}_1) & = \alpha = u(\boldsymbol{x}(\hat{\boldsymbol{v}}_1)) = 0 \\ \Pi_h u(\boldsymbol{v}_2) & = \lambda = u(\boldsymbol{x}(\hat{\boldsymbol{v}}_2)) = \frac{1}{2}ah^2 \\ \Pi_h u(\boldsymbol{v}_3) & = \mu = u(\boldsymbol{x}(\hat{\boldsymbol{v}}_3)) = \frac{1}{2}bh^2 \\ \Pi_h u(\boldsymbol{v}_4) & = \lambda + \mu + \beta = u(\boldsymbol{x}(\hat{\boldsymbol{v}}_4)) = \frac{1}{2}(a+b+2d)h^2 \end{cases}$$

From that, we readily obtain the point-wise interpolate

$$\Pi_h u(\boldsymbol{x}) = \frac{1}{2}ah^2 x + \frac{1}{2}bh^2 y + dh^2 xy \ . \tag{A.4}$$

and the point-wise interpolation error on the reference element K_{ref}

$$(u - \Pi_h u)(\boldsymbol{x}) = \frac{1}{2}h^2 \left[a(x^2 - x) + b(y^2 - y)\right] \ . \tag{A.5}$$

By direct integration, we obtain the interpolation error on K_{ref} in $\boldsymbol{L}^1$-norm

$$||u - \Pi_h u||_{\boldsymbol{L}^1(K_{ref})} = \int_0^1 \int_0^1 |u - \Pi_h u|(\boldsymbol{x}) dx\, dy = \frac{1}{12}\,\mathrm{tr}(H) h^2 \tag{A.6}$$

and applying the change of variables between a coordinate $\boldsymbol{x} \in K$ and $\hat{\boldsymbol{x}} \in K_{ref}$

$$\int_K f(\boldsymbol{x})\ \mathrm{d}\boldsymbol{x} = \int_{K_{ref}} f(\hat{\boldsymbol{x}}) h^2 \mathrm{d}\hat{\boldsymbol{x}} = |K| \int_{K_{ref}} f(\hat{\boldsymbol{x}}) \mathrm{d}\hat{\boldsymbol{x}} \tag{A.7}$$

with $|K| = h^2$ the volume of the element K, the $\boldsymbol{L}^1$-norm interpolation error of the element K writes

$$||u - \Pi_h u||_{\boldsymbol{L}^1(K)} = \frac{1}{12} \operatorname{tr}(H) h^2 |K| \ . \tag{A.8}$$

Finally, and similarly as in [83, 84], we define the continuous interpolate $\pi_{\mathcal{M}} u$ such that the continuous error $|u - \pi_{\mathcal{M}} u|$ is equal to the error on a square element K:

$$|u - \pi_{\mathcal{M}} u| = \frac{||u - \Pi_h u||_{\boldsymbol{L}^1(K)}}{|K|} = C_2 \operatorname{tr}\left(\mathcal{M}_u H \mathcal{M}_u\right) \ , \tag{A.9}$$

with $C_2 = \frac{1}{12}$, $\mathcal{M}_u = h^{-2} I_2$ the metric in which the cube is unit, and I_2 the identity matrix.

Appendix A.1. Continuous interpolation error for cubic elements

The idea of the proof is identical to the case of the square element. We first derive the discrete $\boldsymbol{L}^1$-norm interpolation error estimate for a cubic element K represented by its vertex $(\boldsymbol{v}_i)_{i \in [\![1,8]\!]}$. To find the discrete trilinear interpolation error, we show off an exact point-wise error estimate of the interpolation error within an element of reference K_{ref}. We apply a change of variables to obtain the error on K, from the error on K_{ref}. Then we will use the analogy between discrete and continuous interpolation error to provide the continuous interpolation error as in [83, 84].

The reference element is a cubic element with unit edge lengths. Its vertices are noted $\hat{\boldsymbol{v}}_1 = (0,0,0)$, $\hat{\boldsymbol{v}}_2 = (1,0,0)$, $\hat{\boldsymbol{v}}_3 = (0,1,0)$, $\hat{\boldsymbol{v}}_4 = (0,0,1)$, $\hat{\boldsymbol{v}}_5 = (1,1,0)$, $\hat{\boldsymbol{v}}_6 = (1,0,1)$, $\hat{\boldsymbol{v}}_7 = (0,1,1)$, $\hat{\boldsymbol{v}}_8 = (1,1,1)$. The transformation between a coordinate $\boldsymbol{x} \in K$ and $\hat{\boldsymbol{x}} \in K_{ref}$ is given by

$$\boldsymbol{x} = \boldsymbol{v}_1 + h\hat{\boldsymbol{x}} \tag{A.10}$$

with h the edge length of K. One could note here that finding the appropriate change of variable would allow to extend the proof to any hexaedron instead of the particular case of the cubic element.

Let $u(\boldsymbol{x}) = \boldsymbol{x}^T H \boldsymbol{x}$ be a quadratic function exactly valued at the vertices of the cube and represented by its associated Hessian $H = \begin{pmatrix} a & d & e \\ d & b & f \\ e & f & c \end{pmatrix}$. We pose Π_h the trilinear interpolate operator. In the framework of K_{ref}, u reads

$$u(\boldsymbol{x}(\hat{\boldsymbol{x}})) = \frac{1}{2}\boldsymbol{v}_1^T H \boldsymbol{v}_1 + \frac{1}{2}\boldsymbol{v}_1^T H h \hat{\boldsymbol{x}} + \frac{1}{2}\hat{\boldsymbol{x}}^T h H \boldsymbol{v}_1 + \frac{1}{2} h^2 \hat{\boldsymbol{x}}^T H \hat{\boldsymbol{x}} \ . \tag{A.11}$$

Linear and constant terms of $u(\boldsymbol{x}(\hat{bx}))$ are exactly interpolated, thus we consider, without loss in generality, only the quadratic term $\hat{u}(\boldsymbol{x}) = \frac{1}{2} h^2 \hat{\boldsymbol{x}}^T H \hat{\boldsymbol{x}}$, as $(u - \Pi_h u)(\boldsymbol{x}) = (\hat{u} - \Pi_h \hat{u})(\boldsymbol{x})$. To improve readability, we write u instead of $\hat{u}$ in the following.

For all $\boldsymbol{x} = (x, y, z)^T \in K_{ref}$, the trilinear interpolate $\Pi_h u$ of the function u is

$$\Pi_h u(\boldsymbol{x}) = \alpha + \lambda x + \mu y + \eta z + \beta xy + \gamma xz + \omega yz \ . \tag{A.12}$$

As the solution is exact-valued at the vertices, we have

$$\begin{cases} \Pi_h u(\boldsymbol{v}_1) & = \alpha = u(\boldsymbol{x}(\hat{\boldsymbol{v}}_1)) = 0 \\ \Pi_h u(\boldsymbol{v}_2) & = \lambda = u(\boldsymbol{x}(\hat{\boldsymbol{v}}_2)) = \frac{1}{2} a h^2 \\ \Pi_h u(\boldsymbol{v}_3) & = \mu = u(\boldsymbol{x}(\hat{\boldsymbol{v}}_3)) = \frac{1}{2} b h^2 \\ \Pi_h u(\boldsymbol{v}_4) & = \eta = u(\boldsymbol{x}(\hat{\boldsymbol{v}}_4)) = \frac{1}{2} c h^2 \\ \Pi_h u(\boldsymbol{v}_5) & = \lambda + \mu + \beta = u(\boldsymbol{x}(\hat{\boldsymbol{v}}_5)) = \frac{1}{2}(a + b + 2d) h^2 \\ \Pi_h u(\boldsymbol{v}_6) & = \lambda + \eta + \gamma = u(\boldsymbol{x}(\hat{\boldsymbol{v}}_6)) = \frac{1}{2}(a + c + 2e) h^2 \\ \Pi_h u(\boldsymbol{v}_7) & = \mu + \eta + \omega = u(\boldsymbol{x}(\hat{\boldsymbol{v}}_7)) = \frac{1}{2}(b + c + 2f) h^2 \\ \Pi_h u(\boldsymbol{v}_8) & = \lambda + \mu + \eta + \beta + \gamma + \omega = u(\boldsymbol{x}(\hat{\boldsymbol{v}}_8)) = \frac{1}{2}(a + b + c + 2d + 2e + 2f) h^2 \end{cases}$$

From that, we readily obtain the point-wise interpolate

$$\Pi_h u(\boldsymbol{x}) = \frac{1}{2} a h^2 x + \frac{1}{2} b h^2 y + \frac{1}{2} c h^2 z + d h^2 xy + e h^2 xz + f h^2 yz \ . \tag{A.13}$$

and the point-wise interpolation error on the reference element K_{ref}

$$(u - \Pi_h u)(\boldsymbol{x}) = \frac{1}{2} h^2 \left[a(x^2 - x) + b(y^2 - y) + c(z^2 - z) \right] \ . \tag{A.14}$$

By direct integration, we obtain the interpolation error on K_{ref} in $\boldsymbol{L}^1$-norm

$$||u - \Pi_h u||_{\boldsymbol{L}^1(K_{ref})} = \int_0^1 \int_0^1 \int_0^1 |u - \Pi_h u|(\boldsymbol{x}) dx\, dy\, dz = \frac{1}{12} \operatorname{tr}(H) h^2 \tag{A.15}$$

We apply the change of variables between a coordinate $\boldsymbol{x} \in K$ and $\hat{\boldsymbol{x}} \in K_{ref}$. It writes

$$\int_K f(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} = \int_{K_{ref}} f(\hat{\boldsymbol{x}}) h^3 \mathrm{d}\hat{\boldsymbol{x}} = |K| \int_{K_{ref}} f(\hat{\boldsymbol{x}}) \mathrm{d}\hat{\boldsymbol{x}} \tag{A.16}$$

with $|K| = h^3$ the volume of the element K. Thus, the $\boldsymbol{L}^1$-norm interpolation error of the element K writes

$$||u - \Pi_h u||_{\boldsymbol{L}^1(K)} = \frac{1}{12} \operatorname{tr}(H) h^2 |K| \ . \tag{A.17}$$

Finally, and similarly as in [83, 84], we define the continuous interpolate $\pi_{\mathcal{M}} u$ such that the continuous error $|u - \pi_{\mathcal{M}} u|$ is equal to the error on a square element K:

$$|u - \pi_{\mathcal{M}} u| = \frac{||u - \Pi_h u||_{\boldsymbol{L}^1(K)}}{|K|} = C_3 \operatorname{tr}\left(\mathcal{M}_u H \mathcal{M}_u \right) \ , \tag{A.18}$$

with $C_3 = \frac{1}{12}$, $\mathcal{M}_u = h^{-2} I_3$ the metric in which the cube is unit, and I_3 the identity matrix. It is interesting to note that the constant C_n is identical for the cubic and the square elements, which is not the case for triangular and tetrahedral elements.

Appendix B. Uniformly refined mesh interpolation error

The local continuous error in equation (11) is valid for any mesh containing only square or cubic element. In particular, we can use it to determine the global $\boldsymbol{L}^p$ interpolation error on a uniformly refined mesh composed with square/cubic elements:

$$||u-\pi_{\mathcal{M}}u||_{\boldsymbol{L}^p(\Omega)} = \left(\int_\Omega (|u-\pi_{\mathcal{M}}u|(\boldsymbol{x}))^p \ \mathrm{d}\boldsymbol{x}\right)^{\frac{1}{p}} = C_n\, h^2 \left(\int_\Omega \mathrm{tr}\,(|H|(\boldsymbol{x}))^p \ \mathrm{d}\boldsymbol{x}\right)^{\frac{1}{p}} \ , \tag{B.1}$$

with h the uniform cell size.

Using the number of element N of the uniform mesh $N = \left(\frac{L_0}{h}\right)^n$, the interpolation error on a uniform mesh reduces to

$$||u-\pi_{\mathcal{M}}u||_{\boldsymbol{L}^p(\Omega)} = C_n\, L_0^2 \left(\int_\Omega \mathrm{tr}(|H|(\boldsymbol{x}))^p \ \mathrm{d}\boldsymbol{x}\right)^{\frac{1}{p}} N^{-\frac{2}{n}} \ . \tag{B.2}$$

We refer the reader to [90] for the validation of this error estimation on several examples.

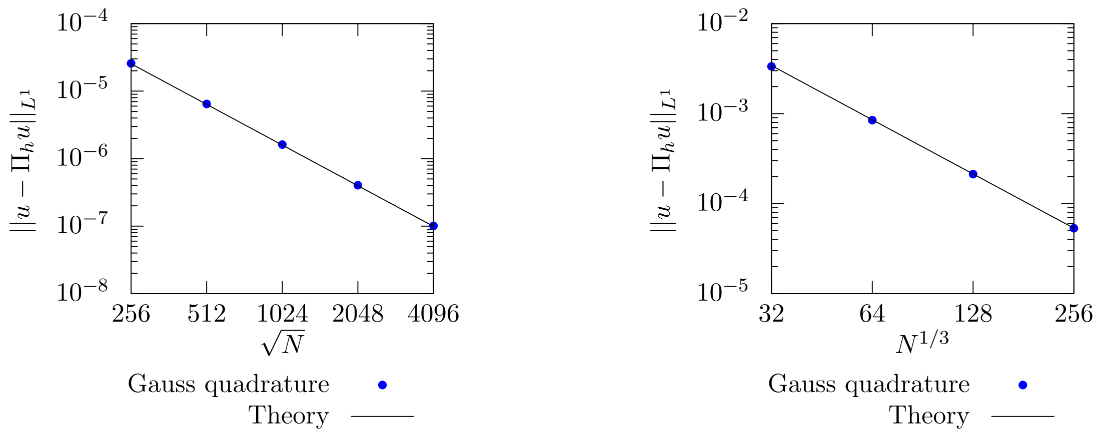
We validate this estimation using the quadratic function

$$u(x,y) = 6x^2 + 2xy + 4y^2 \tag{B.3}$$

with the associated Hessian $H = \begin{pmatrix} 12 & 2 \\ 2 & 8 \end{pmatrix}$. Imposing a unit length square domain, the global error reduces to

$$||u-\pi_{\mathcal{M}}u||_{\boldsymbol{L}^1(\Omega)} = 20\, C_2\, N^{-1} \ . \tag{B.4}$$

In figure B.17a, we can see that the theoretical prediction perfectly matches the error measured with a Gauss quadrature. This behaviour is confirmed for various functions in table B.1, where we introduce $c_{th} = C_n \int_\Omega \mathrm{tr}(|H|(\boldsymbol{x}))\ \mathrm{d}\boldsymbol{x}$ and c_{exp} the constant obtained from the linear fit of the error measured with the Gauss quadrature.



(a) Error convergence for the function u in equation (B.3)

(b) Error convergence for the function u_{3D} in equation (B.5)

Figure B.17: Comparison between the error measured on uniform meshes for a function and its theoretical prediction.

Similarly, we present in figure B.17b the validation of the octree estimation using the quadratic function

$$u_{3D}(x,y) = 6x^2 + 2xy + 4y^2 + 3xz + 4yz + 11z^2 \tag{B.5}$$

with the associated Hessian $H_{3D} = \begin{pmatrix} 12 & 2 & 3 \\ 2 & 8 & 4 \\ 3 & 4 & 22 \end{pmatrix}$. Its global interpolation error reduces to

$$||u_{3D} - \pi_{\mathcal{M}} u_{3D}||_{\boldsymbol{L}^1(\Omega)} = 42\, C_3\, N^{-2/3} \; . \tag{B.6}$$

The theoretical estimation is in close agreement with the measured error *via* a Gauss quadrature, which confirms the validity of this error estimation in 3D.

$u(x, y)$	c_{th}	c_{exp}
$6x^2 + 2xy + 4y^2$	1.67	1.67
e^{2x^2+y}	4.57	4.57
$\sin(x)$	0.038	0.038
$x^2 - e^{2(x-1)/0.1}$	1.73	1.73

Table B.1: Comparison between the theoretical error prefactor $c_{th} = C_n \int_\Omega \mathrm{tr}(|H|(\boldsymbol{x}))\; \mathrm{d}\boldsymbol{x}$ and the measured error prefactor c_{exp} (linear fit of the error measured with the Gauss quadrature) for different functions.

Appendix C. Optimal interpolation error for quad/octree grids

In this section we detail the resolution of the interpolation error minimization problem (13)-(14). This optimization problem has allready been solved in [24, 83, 84] for the case of anisotropic meshes on triangles or tethrahedras. Here we follow an analogous procedure to derive the expression for the optimal error in the case of isotropic elements (e.g. quadtree/octree grids).

For square/cubic elements, the continuous local interpolation error of a twice differentiable function u whose Hessian is H and positive Hessian $|H|$ is shown in equation (12) to be

$$|u - \Pi_{\mathcal{M}} u|(\boldsymbol{x}) = C_n\, \mathrm{tr}(|H|(\boldsymbol{x}))h^2(\boldsymbol{x}) = C_n h^2(\boldsymbol{x}) \sum_{i=1}^{n} \gamma_i(\boldsymbol{x}) \; ,$$

with $\gamma_i = |\boldsymbol{v}_i^T(\boldsymbol{x}) H(\boldsymbol{x}) \boldsymbol{v}_i(\boldsymbol{x})|$ and $(\boldsymbol{v}_i)_{i \in [\![1,n]\!]}$ the eigenvectors of the square/cubic element metric $\mathcal{M} = h^{-2} I_n$. Introducing the local density of the metric $d = h^{-n}$, the minimization problem becomes convex and writes:

$$\text{Find} \min_{\boldsymbol{M}} \left(\int_\Omega \left(d^{-\frac{2}{n}}(\boldsymbol{x}) \sum_{i=1}^{n} \boldsymbol{\gamma}_i(\boldsymbol{x}) \right)^p \mathrm{d}\boldsymbol{x} \right) \; , \tag{C.1}$$

$$\text{under the constraint } \mathcal{C}(\boldsymbol{M}) = \int_\Omega d(\boldsymbol{x})\; \mathrm{d}\boldsymbol{x} = N_{goal} \; . \tag{C.2}$$

Resolution. We solve this optimization problem using the Lagrange multipliers theory. This theory states a necessary optimality condition verified by the solution of the optimal problem: the variation of the Lagrangian associated to the quantity to minimize $E_p = \int_\Omega \left(d^{-\frac{2}{n}}(\boldsymbol{x}) \sum_{i=1}^{n} \boldsymbol{\gamma}_i(\boldsymbol{x}) \right)^p \mathrm{d}\boldsymbol{x}$ and the constraint at point $\boldsymbol{M}$ in direction $\delta\boldsymbol{M}$ vanishes. In other words, it exists a unique real α such that

$$\delta E_p(\boldsymbol{M}, \delta\boldsymbol{M}) + \alpha\, \delta\mathcal{C}(\boldsymbol{M}, \delta\boldsymbol{M}) = 0 \; . \tag{C.3}$$

The variation of E_p is approximated by:

$$\delta E_p(\boldsymbol{M}, \delta \boldsymbol{M}) \approx \int_\Omega \frac{\partial}{\partial \mathcal{M}} \left(\left(d^{-\frac{2}{n}}(\boldsymbol{x}) \sum_{i=1}^{n} \boldsymbol{\gamma}_i(\boldsymbol{x}) \right)^p \right) d\mathcal{M} \tag{C.4}$$

The constraint is constant, which implies that its variation is null:

$$\delta \mathcal{C}(\boldsymbol{M}, \delta \boldsymbol{M}) = 0 \tag{C.5}$$

The optimality condition reduces then to

$$\delta E_p(\boldsymbol{M}, \delta \boldsymbol{M}) = 0 \tag{C.6}$$

From the variation of the metric, $\delta \boldsymbol{M} = (\delta d)$, it follows

$$\frac{\partial}{\partial \mathcal{M}} \left(\left(d^{-\frac{2}{n}}(\boldsymbol{x}) \sum_{i=1}^{n} \boldsymbol{\gamma}_i(\boldsymbol{x}) \right)^p \right) = \frac{\partial}{\partial d} \left(\left(d^{-\frac{2}{n}}(\boldsymbol{x}) \sum_{i=1}^{n} \boldsymbol{\gamma}_i(\boldsymbol{x}) \right)^p \right) = -\frac{2p}{n} \left(\sum_{i=1}^{n} \boldsymbol{\gamma}_i \right)^p d^{-\frac{2p+n}{n}} . \tag{C.7}$$

Equations C.6 and C.7 leads directly to

$$d = K \left(\sum_{i=1}^{n} \boldsymbol{\gamma}_i \right)^{\frac{np}{2p+n}} , \tag{C.8}$$

with K a constant that can be obtained by applying the constraint $\mathcal{C}(\boldsymbol{M}) = N_{goal}$ to obtain

$$d = N_{goal} \left(\int_\Omega \left(\sum_{i=1}^{n} \boldsymbol{\gamma}_i \right)^{\frac{np}{2p+n}} \right)^{-1} \left(\sum_{i=1}^{n} \boldsymbol{\gamma}_i \right)^{\frac{np}{2p+n}} . \tag{C.9}$$

Inverting the change of variable, we found the optimal size

$$h_{opt}(\boldsymbol{x}) = N_{goal}^{-\frac{1}{n}} \left(\int_\Omega \left(\sum_{i=1}^{n} \boldsymbol{\gamma}_i(\boldsymbol{x}) \right)^{\frac{np}{2p+n}} \mathrm{d}\boldsymbol{x} \right)^{\frac{1}{n}} \left(\sum_{i=1}^{n} \boldsymbol{\gamma}_i(\boldsymbol{x}) \right)^{-\frac{p}{2p+n}} , \tag{C.10}$$

the point-wise local error

$$|u - \pi_{\mathcal{M}} u|(\boldsymbol{x}) = C_n N_{goal}^{-\frac{2}{n}} \left(\int_\Omega \left(\sum_{i=1}^{n} \boldsymbol{\gamma}_i(\boldsymbol{x}) \right)^{\frac{np}{2p+n}} \mathrm{d}\boldsymbol{x} \right)^{\frac{2}{n}} \left(\sum_{i=1}^{n} \boldsymbol{\gamma}_i(\boldsymbol{x}) \right)^{\frac{n}{2p+n}} , \tag{C.11}$$

and the optimal global error

$$||u - \pi_{\mathcal{M}} u||_{\Omega, L^p} = C_n N_{goal}^{-\frac{2}{n}} \left(\int_\Omega \left(\sum_{i=1}^{n} \boldsymbol{\gamma}_i(\boldsymbol{x}) \right)^{\frac{np}{2p+n}} \mathrm{d}\boldsymbol{x} \right)^{\frac{2p+n}{np}} . \tag{C.12}$$

We can further proof the optimal metric $\boldsymbol{M}_{opt}$ is also the unique solution that verifies $E_p(\boldsymbol{M}_{opt})^p \leq E_p(\boldsymbol{M})^p$ for all $\boldsymbol{M}$ having the same fixed $(\boldsymbol{\gamma}_i)_{i \in [\![1,n]\!]}$. The details of this demonstration can be found in [90].

Uniqueness. Now, we prove that this solution is the unique solution which verifies $E_p(\boldsymbol{M}_{opt})^p \leq E_p(\boldsymbol{M})^p$ for all $\boldsymbol{M}$ having the same fixed $(\boldsymbol{\gamma}_i)_{i \in [\![1,n]\!]}$, and $\boldsymbol{M}_{opt}$ being the optimal metric previously found. From equation (C.12), we have

$$E_p(\boldsymbol{M}_{opt})^p = C_n^p N_{goal}^{-\frac{2p}{n}} \left(\int_\Omega \left(\sum_{i=1}^n \boldsymbol{\gamma}_i(\boldsymbol{x}) \right)^{\frac{np}{2p+n}} \mathrm{d}\boldsymbol{x} \right)^{\frac{2p+n}{n}} . \tag{C.13}$$

The density equation (C.9) is rewritten as $d = N_* (\int_\Omega f(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x})^{-1} f$, where f is a strictly positive function. For any set of metric $\boldsymbol{M}$, writes

$$E_p(\boldsymbol{M})^p = C_n^p N_{goal}^{-\frac{2p}{n}} \left(\int_\Omega f(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} \right)^{\frac{2p}{n}} \left(\int_\Omega f(\boldsymbol{x})^{-\frac{2p}{n}} \left(\sum_{i=1}^n \boldsymbol{\gamma}_i \right)^p \mathrm{d}\boldsymbol{x} \right) . \tag{C.14}$$

To prove $E_p(\boldsymbol{M}_{opt}) < E_p(\boldsymbol{M})$, we use the inequality $(\sum_{i=1}^n \boldsymbol{\gamma}_i)^{\frac{np}{2p+n}} \leq (\sum_{i=1}^n \boldsymbol{\gamma}_i)^p$ which is equivalent to the inequality $\frac{np}{2p+n} \leq p$ as $\frac{np}{2p+n} \geq 0$, $p \geq 0$ and $\boldsymbol{\gamma}_i \geq 0$, which is equivalent to $p^2 \geq 0$ and it is always verified. We introduce $g = (\sum_{i=1}^n \boldsymbol{\gamma}_i)^{\frac{np}{2p+n}}$ and obtain

$$E_p(\boldsymbol{M}_{opt})^{\frac{np}{2p+n}} = c_n^{\frac{np}{2p+n}} N_{goal}^{-\frac{2p}{2p+n}} \int_\Omega g(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} \tag{C.15}$$

and

$$E_p(\boldsymbol{M})^{\frac{np}{2p+n}} \geq C_n^{\frac{np}{2p+n}} N_{goal}^{-\frac{2p}{2p+n}} \left(\int_\Omega f(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} \right)^{\frac{2p}{2p+n}} \left(\int_\Omega f(\boldsymbol{x})^{-\frac{2p}{n}} g(\boldsymbol{x})^{\frac{2p+n}{n}} \mathrm{d}\boldsymbol{x} \right)^{\frac{np}{2p+n}} . \tag{C.16}$$

Using Hölder inequality, it comes

$$\left(\int_\Omega f(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} \right)^{\frac{2p}{2p+n}} \left(\int_\Omega f(\boldsymbol{x})^{-\frac{2p}{n}} g(\boldsymbol{x})^{\frac{2p+n}{n}} \mathrm{d}\boldsymbol{x} \right)^{\frac{np}{2p+n}} = \left(\int_\Omega f(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} \right)^{\frac{2p}{2p+n}} \left(\int_\Omega \left(\frac{g(\boldsymbol{x})}{f(\boldsymbol{x})^{\frac{2p}{2p+n}}} \right)^{\frac{2p+n}{n}} \mathrm{d}\boldsymbol{x} \right)^{\frac{np}{2p+n}} \tag{C.17}$$

$$\geq \int_\Omega f(\boldsymbol{x})^{\frac{2p}{2p+n}} \left(\frac{g(\boldsymbol{x})}{f(\boldsymbol{x})^{\frac{2p}{2p+n}}} \right) \mathrm{d}\boldsymbol{x} \tag{C.18}$$

$$\geq \int_\Omega g(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} \ , \tag{C.19}$$

as $\frac{2p+n}{2p} \geq 1$, $\frac{2p+n}{n} \geq 1$ and $\frac{2p}{2p+n} + \frac{n}{2p+n} = 1$. Equation (C.19) implies $E_p(\boldsymbol{M}_{opt})^p \leq E_p(\boldsymbol{M})^p$ for all $\boldsymbol{M}$ having the same fixed $(\boldsymbol{\gamma}_i)_{i \in [\![1,n]\!]}$. The initial optimization problem is strictly convex, so the optimal solution $\boldsymbol{M}_{opt}$ is unique.

Finally, using the definition of the complexity N_{goal}, we get

$$N_{goal} = \int_\Omega (h(\boldsymbol{x})^{-n}) \, \mathrm{d}\boldsymbol{x} = \frac{L_0^n}{\overline{h}^n} \tag{C.20}$$

which means that the mesh complexity constrain N_{goal} is the number of element of the optimized mesh $N = L_0^n / \overline{h}^n$. From that, the continuous error writes

$$|u - \pi_{\mathcal{M}} u|(\boldsymbol{x}) = C_n \, N_{goal}^{-\frac{2}{n}} \left(\int_\Omega (\operatorname{tr}(|H|(\boldsymbol{x})))^{\frac{np}{2p+n}} \mathrm{d}\boldsymbol{x} \right)^{\frac{2}{n}} (\operatorname{tr}(|H|(\boldsymbol{x})))^{\frac{n}{2p+n}} \ , \tag{C.21}$$

with $C_2 = C_3 = \frac{1}{12}$, and

$$||u - \pi_{\mathcal{M}} u||_{\Omega, L^p} = C_n \, N_{goal}^{-\frac{2}{n}} \left(\int_\Omega (\operatorname{tr}(|H|(\boldsymbol{x})))^{\frac{np}{2p+n}} \mathrm{d}\boldsymbol{x} \right)^{\frac{2p+n}{np}} . \tag{C.22}$$

References

[1] A. Berny, Étude numérique de l'éclatement d'une bulle à la surface de différents liquides, Thèse de doctorat, Sorbonne Université, 2020. URL: `https://tel.archives-ouvertes.fr/tel-03414116`.

[2] C. Ozhan, D. Fuster, P. Da Costa, Multi-scale flow simulation of automotive catalytic converters, Chemical Engineering Science 116 (2014) 161–171.

[3] J. Gimenez, S. Idelsohn, E. Oñate, R. Löhner, A multiscale approach for the numerical simulation of turbulent flows with droplets, Archives of Computational Methods in Engineering 28 (2021) 4185–4204.

[4] J. P. Slotnick, A. Khodadoust, J. Alonso, D. Darmofal, W. Gropp, E. Lurie, D. J. Mavriplis, CFD vision 2030 study: a path to revolutionary computational aerosciences, Technical Report, 2014.

[5] M. A. Park, A. Loseille, J. Krakos, T. R. Michal, J. J. Alonso, Unstructured grid adaptation: status, potential impacts, and recommended investments towards cfd 2030, in: 46th AIAA fluid dynamics conference, 2016, p. 3323.

[6] P. J. Roache, Verification of codes and calculations, AIAA Journal 36 (1998) 696–702.

[7] W. L. Oberkampf, T. G. Trucano, Verification and validation in computational fluid dynamics, Progress in Aerospace Sciences 38 (2002) 209–272.

[8] D. Pelletier, Verification, validation, and uncertainty in computational fluids dynamicsthis article is one of a selection of papers published in this special issue on hydrotechnical engineering., Canadian Journal of Civil Engineering 37 (2010) 1003–1013.

[9] F. Alauzet, L. Frazza, Feature-based and goal-oriented anisotropic mesh adaptation for rans applications in aeronautics and aerospace, Journal of Computational Physics 439 (2021) 110340.

[10] S. Popinet, Gerris: a tree-based adaptive solver for the incompressible Euler equations in complex geometries, Journal of Computational Physics 190 (2003) 572–600.

[11] J. A. Van Hooft, S. Popinet, C. C. Van Heerwaarden, S. J. A. Van Der Linden, S. R. De Roode, B. J. H. Van de Wiel, Towards Adaptive Grids for Atmospheric Boundary-Layer Simulations, Boundary-Layer Meteorology (2018) 1–23.

[12] M. J. Berger, J. Oliger, Adaptive mesh refinement for hyperbolic partial differential equations, Journal of Computational Physics 53 (1984) 484–512.

[13] M. Berger, P. Colella, Local adaptive mesh refinement for shock hydrodynamics, Journal of Computational Physics 82 (1989) 64–84.

[14] X. Gao, C. P. Groth, A parallel solution – adaptive method for three-dimensional turbulent non-premixed combusting flows, Journal of Computational Physics 229 (2010) 3250–3275.

[15] F. Alauzet, A. Loseille, A decade of progress on anisotropic mesh adaptation for computational fluid dynamics, Computer-Aided Design 72 (2016) 13–39.

[16] X. Li, J.-F. Remacle, N. Chevaugeon, M. S. Shephard, Anisotropic mesh gradation control., in: IMR, 2004, pp. 401–412.

[17] A. Loseille, A. Dervieux, F. Alauzet, Fully anisotropic goal-oriented mesh adaptation for 3d steady euler equations, Journal of Computational Physics 229 (2010) 2866–2897.

[18] A. Belme, F. Alauzet, A. Dervieux, An a priori anisotropic goal-oriented error estimate for viscous compressible flow and application to mesh adaptation, Journal of Computational Physics 376 (2019) 1051–1088.

[19] J. Wackers, G. Deng, C. Raymond, E. Guilmineau, A. Leroyer, P. Queutey, M. Visonneau, Adaptive grid refinement for ship resistance computations, Ocean Engineering 250 (2022) 110969.

[20] A. Belme, A. Dervieux, F. Alauzet, Time accurate anisotropic goal-oriented mesh adaptation for unsteady flows, Journal of Computational Physics 231 (2012) 6323–6348.

[21] J. Wackers, G. Deng, E. Guilmineau, A. Leroyer, P. Queutey, M. Visonneau, A. Palmieri, A. Liverani, Can adaptive grid refinement produce grid-independent solutions for incompressible flows?, Journal of Computational Physics 344 (2017) 364–380.

[22] J. Peraire, M. Vahdati, K. Morgan, O. Zienkiewicz, Adaptive remeshing for compressible flow computations, Journal of Computational Physics 72 (1987) 449–466.

[23] F. Hecht, B. Mohammadi, Mesh adaption by metric control for multi-scale phenomena and turbulence, 1997. doi:`10.2514/6.1997-859`.

[24] A. Loseille, Adaptation de maillage anisotrope 3D multi-échelles et ciblée à une fonctionnelle pour la mécanique des fluides : Application à la prédiction haute-fidélité du bang sonique, Thèse de doctorat, Paris 6, 2008. URL: `http://www.theses.fr/2008PA066622`.

[25] L. F. Richardson, R. T. Glazebrook, Ix. the approximate arithmetical solution by finite differences of physical problems involving differential equations, with an application to the stresses in a masonry dam, Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character 210 (1911) 307–357.

[26] L. Richardson, A. Gaunt, The deferred approach to the limit. part i. single lattic. part ll. interpenetrationg lattices, The Philosophical Transcations of the Royal Society of London. Series A, Containing papers of a Mathematical or Physical Character 116 (1927) 405–41.

[27] P. J. Roache, P. M. Knupp, Completed richardson extrapolation, Communications in Numerical Methods in Engineering 9 (1993) 365–374.

[28] E. Fehlberg, Low-order classical Runge-Kutta formulas with stepsize control and their application to some heat transfer problems, volume 315, National aeronautics and space administration, 1969.

[29] O. C. Zienkiewicz, J. Z. Zhu, A simple error estimator and adaptive procedure for practical engineerng analysis, International Journal for Numerical Methods in Engineering 24 (1987) 337–357.

[30] V. Dolejší, P. Solin, hp-discontinuous galerkin method based on local higher order reconstruction, Applied Mathematics and Computation 279 (2016) 219–235.

[31] J. Wackers, G. Deng, A. Leroyer, P. Queutey, M. Visonneau, Adaptive grid refinement for hydrodynamic flows, Computers and Fluids 55 (2012) 85–100.

[32] J. Wackers, G. Deng, E. Guilmineau, A. Leroyer, P. Queutey, M. Visonneau, Combined refinement criteria for anisotropic grid refinement in free-surface flow simulation, Computers and Fluids 92 (2014) 209–222.

[33] K. MacLean, S. Nadarajah, Anisotropic mesh generation and adaptation for quads using the L^p-CVT method, Journal of Computational Physics (2022) 111578.

[34] T. C. Baudouin, J.-F. Remacle, E. Marchandise, J. Lambrechts, F. Henrotte, Lloyd's energy minimization in the L^p norm for quadrilateral surface mesh generation, Engineering with Computers 30 (2014) 97–110.

[35] B. Lévy, Y. Liu, L^p centroidal voronoi tessellation and its applications, ACM Transactions on Graphics (TOG) 29 (2010) 1–11.

[36] C. Roy, Review of Discretization Error Estimators in Scientific Computing, 2010. doi:`10.2514/6.2010-126`.

[37] I. Babuška, W. C. Rheinboldt, A-posteriori error estimates for the finite element method, International Journal for Numerical Methods in Engineering 12 (1978) 1597–1615.

[38] M. Ainsworth, J. Oden, A posteriori error estimation in finite element analysis, Computer Methods in Applied Mechanics and Engineering 142 (1997) 1–88.

[39] R. Verfürth, A posteriori error estimators for convection-diffusion equations, Numerische Mathematik 80 (1998) 641–663.

[40] Vohralík, Martin, Residual flux-based a posteriori error estimates for finite volume and related locally conservative methods, Numerische Mathematik 111 (2008) 121–158.

[41] Di Pietro, Daniele A., Vohralík, Martin, A review of recent advances in discretization methods, a posteriori error analysis, and adaptive algorithms for numerical modeling in geosciences, Oil Gas Sci. Technol. - Rev. IFP Energies nouvelles 69 (2014) 701–729.

[42] J. Banks, J. Hittinger, J. Connors, C. Woodward, Numerical error estimation for nonlinear hyperbolic pdes via nonlinear error transport, Computer Methods in Applied Mechanics and Engineering 213-216 (2012) 1–15.

[43] Y. Qin, T. Shih, P. Keller, R. Sun, E. Hernandez, C. Perng, N. Trigui, Z. Han, F. Shen, T. Shieh, Estimating Grid-Induced Errors in CFD by Discrete-Error-Transport Equations, 2004. doi:`10.2514/6.2004-656`.

[44] I. Celik, G. Hu, Single Grid Error Estimation Using Error Transport Equation , Journal of Fluids Engineering 126 (2004) 778–790.

[45] A. Hay, M. Visonneau, Error estimation using the error transport equation for finite-volume methods and arbitrary meshes, International Journal of Computational Fluid Dynamics 20 (2006) 463–479.

[46] J. M. Derlaga, M. A. Park, Application of Exact Error Transport Equations and Adjoint Error Estimation to AIAA Workshops, 2017. doi:`10.2514/6.2017-0076`.

[47] I. Babuška, T. Strouboulis, S. Gangaraj, C. Upadhyay, Pollution error in the h-version of the finite element method and the local quality of the recovered derivatives, Computer Methods in Applied Mechanics and Engineering 140 (1997) 1–37.

[48] O. C. Zienkiewicz, The background of error estimation and adaptivity in finite element computations, Computer Methods in Applied Mechanics and Engineering 195 (2006) 207–213. Adaptive Modeling and Simulation.

[49] R. Becker, R. Rannacher, Weighted A Posteriori Error Control in FE Methods, IWR, 1996.

[50] R. Balasubramanian, J. C. Newman III, Comparison of adjoint-based and feature-based grid adaptation for functional outputs, International Journal for Numerical Methods in Fluids 53 (2007) 1541–1569.

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

[52] D. A. Venditti, D. L. Darmofal, Grid adaptation for functional outputs: Application to two-dimensional inviscid flows, Journal of Computational Physics 176 (2002) 40–69.

[53] N. A. Pierce, M. B. Giles, Adjoint and defect error bounding and correction for functional estimates, Journal of Computational Physics 200 (2004) 769–794.

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

[55] R. Hartmann, P. Houston, Adaptive discontinuous galerkin finite element methods for the compressible euler equations, Journal of Computational Physics 183 (2002) 508–532.

[56] R. Hartmann, Multitarget error estimation and adaptivity in aerodynamic flow simulations, SIAM Journal on Scientific Computing 31 (2008) 708–731.

[57] G. Brèthes, A. Dervieux, Anisotropic norm-oriented mesh adaptation for a poisson problem, Journal of Computational Physics 322 (2016) 804–826.

[58] A. Loseille, A. Dervieux, F. Alauzet, Anisotropic Norm-Oriented Mesh Adaptation for Compressible Flows, 2015. doi:`10.2514/6.2015-2037`.

[59] L. Frazza, 3D anisotropic mesh adaptation for Reynolds Averaged Navier-Stokes simulations, Thèse de doctorat, Sorbonne université, 2018. URL: `http://www.theses.fr/2018SORUS423`.

[60] E. Aulisa, G. Ke, S.-Y. Lee, An adaptive mesh refinement strategy for finite element solution of the elliptic problem, Computers & Mathematics with Applications 76 (2018) 224–244.

[61] R. E. Bank, A. Weiser, Some a posteriori error estimators for elliptic partial differential equations, Mathematics of computation 44 (1985) 283–301.

[62] F. A. Bornemann, B. Erdmann, R. Kornhuber, A posteriori error estimates for elliptic problems in two and three space dimensions, SIAM journal on numerical analysis 33 (1996) 1188–1204.

[63] K. Eriksson, C. Johnson, An adaptive finite element method for linear elliptic problems, Mathematics of Computation 50 (1988) 361–383.

[64] M. Ainsworth, J. T. Oden, A unified approach to a posteriori error estimation using element residual methods, Numerische Mathematik 65 (1993) 23–50.

[65] L. Formaggia, S. Micheletti, S. Perotto, Anisotropic mesh adaptation in computational fluid dynamics: application to the advection–diffusion–reaction and the stokes problems, Applied Numerical Mathematics 51 (2004) 511–533.

[66] J. Carpio, J. L. Prieto, R. Bermejo, Anisotropic "goal-oriented" mesh adaptivity for elliptic problems, SIAM Journal on Scientific Computing 35 (2013) A861–A885.

[67] S. Prudhomme, J. T. Oden, On goal-oriented error estimation for elliptic problems: application to the control of pointwise errors, Computer Methods in Applied Mechanics and Engineering 176 (1999) 313–331.

[68] J. T. Oden, S. Prudhomme, Estimation of modeling error in computational mechanics, Journal of Computational Physics 182 (2002) 496–515.

[69] K. Eriksson, Improved accuracy by adapted mesh-refinements in the finite element method, Mathematics of computation 44 (1985) 321–343.

[70] R. Araya, E. Behrens, R. Rodríguez, A posteriori error estimates for elliptic problems with dirac delta source terms, Numerische Mathematik 105 (2006) 193–216.

[71] T. Apel, O. Benedix, D. Sirch, B. Vexler, A priori mesh grading for an elliptic problem with dirac right-hand side, SIAM Journal on Numerical Analysis 49 (2011) 992–1005.

[72] S. Popinet, The Basilisk software, 2013. URL: `http://basilisk.fr/`, accessed: 2022-08-20.

[73] S. Popinet, A quadtree-adaptive multigrid solver for the serre–green–naghdi equations, Journal of Computational Physics 302 (2015) 336–358.

[74] L. Prouvost, 2020. URL: `http://basilisk.fr/sandbox/prouvost/README`, accessed: 2022-08-20.

[75] S. Popinet, Quadtree-adaptive tsunami modelling, Ocean Dynamics 61 (2011) 1261–1285.

[76] S. Popinet, G. Rickard, A tree-based solver for adaptive ocean modelling, Ocean Modelling 16 (2007) 224–249.

[77] G. Bessette, E. Becker, L. Taylor, D. Littlefield, Modeling of impact problems using an h-adaptive, explicit lagrangian finite element method in three dimensions, Computer Methods in Applied Mechanics and Engineering 192 (2003) 1649–1679.

[78] D. Boffi, M. Costabel, M. Dauge, L. Demkowicz, Discrete compactness for the hp version of rectangular edge finite elements, SIAM Journal on Numerical Analysis 44 (2006) 979–1004.

[79] C. Min, F. Gibou, A second order accurate projection method for the incompressible navier–stokes equations on non-graded adaptive grids, Journal of Computational Physics 219 (2006) 912–929.

[80] M. Mirzadeh, A. Guittet, C. Burstedde, F. Gibou, Parallel level-set methods on adaptive tree-based grids, Journal of Computational Physics 322 (2016) 345–364.

[81] H. Chen, C. Min, F. Gibou, A numerical scheme for the stefan problem on adaptive cartesian grids with supralinear convergence rate, Journal of Computational Physics 228 (2009) 5803–5818.

[82] F. Gibou, C. Min, R. Fedkiw, High resolution sharp computational methods for elliptic and parabolic problems in complex geometries, Journal of Scientific Computing 54 (2013) 369–413.

[83] A. Loseille, F. Alauzet, Continuous mesh framework part i: Well-posed continuous interpolation error, SIAM Journal on Numerical Analysis 49 (2011) 38–60.

[84] A. Loseille, F. Alauzet, Continuous mesh framework part ii: Validations and applications, SIAM Journal on Numerical Analysis 49 (2011) 61–86.

[85] L.-M. Tenkes, F. Alauzet, Size gradation control for anisotropic hybrid meshes, in: V. A. Garanzha, L. Kamenski, H. Si (Eds.), Numerical Geometry, Grid Generation and Scientific Computing, Springer International Publishing, Cham, 2021, pp. 127–139.

[86] V. John, A numerical study of a posteriori error estimators for convection–diffusion equations, Computer Methods in Applied Mechanics and Engineering 190 (2000) 757–781.

[87] G. Hauke, D. Fuster, M. H. Doweidar, Variational multiscale a-posteriori error estimation for multi-dimensional transport problems, Computer Methods in Applied Mechanics and Engineering 197 (2008) 2701–2718.

[88] H. Borouchaki, F. Hecht, P. J. Frey, Mesh gradation control, International Journal for Numerical Methods in Engineering 43 (1998) 1143–1165.

[89] P.-Y. Lagrée, L. Staron, S. Popinet, The granular column collapse as a continuum: validity of a two-dimensional navier–stokes model with a mu(i)-rheology, Journal of Fluid Mechanics 686 (2011) 378–408.

[90] L. Prouvost, Mesh adaptation for elliptic equations on quadtree/octree grids, Thèse de doctorat, Sorbonne Université, 2022. URL: `https://theses.hal.science/tel-03966961`.