An Optimal Transport Based hr-Adaptive Mesh Pursuit

Sara Maad, Kenan Kergrene, Jonathan Vacher, Diane Guignard, Serge Prudhomme, Alain Rassineux

▶ **To cite this version:**

Sara Maad, Kenan Kergrene, Jonathan Vacher, Diane Guignard, Serge Prudhomme, et al.. An Optimal Transport Based hr-Adaptive Mesh Pursuit. 16ème Colloque National en Calcul de Structures (CSMA 2024), CNRS; CSMA; ENS Paris-Saclay; CentraleSupélec, May 2024, Hyères, France. hal-04610888v1

HAL Id: hal-04610888

https://hal.science/hal-04610888v1

Submitted on 17 Oct 2024 (v1), last revised 9 Dec 2024 (v2)

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

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

CSMA 2024
16ème Colloque National en Calcul des Structures
13-17 Mai 2024, Presqu'île de Giens (Var)

An Optimal Transport Based *hr*-Adaptive Mesh Pursuit

S. Maad[1], K. Kergrene[1], J. Vacher[2], D. Guignard[3], S. Prudhomme[4], A. Rassineux[1]

[1] *Laboratoire Roberval, Université de Technologie de Compiègne, {sara.maad,kenan.kergrene}@utc.fr*
[2] *Université Paris Cité, CNRS, MAP5, jonathan.vacher@u-paris.fr*
[3] *Department of Mathematics and Statistics, University of Ottawa, dguignar@uottawa.ca*
[4] *Département de Mathématiques et de Génie Industriel, Ecole Polytechnique de Montréal, serge.prudhomme@polymtl.ca*

Abstract — This research work deals with error estimation and mesh adaptation in finite element simulations. The finite element method introduces errors compared to the exact solution when solving partial differential equations, and it is crucial to identify, control and reduce them to obtain a reliable numerical approximation. We propose to use optimal transport techniques for mesh adaptation, by relocating mesh vertices based on local error indicators to their optimal position, thus yielding error-conscious mesh adjustments. Numerical experiments are provided to illustrate the performance of the proposed approach.
Keywords — numerical simulation, finite elements, mesh adaptation, optimal transport.

1 Introduction

The numerical results provided by approximation methods for partial differential equations such as the finite element method are tainted with errors compared to the exact solution of the problem. These errors can come from multiple sources e.g. discretization, numerical linear algebra, linearization, uncertainties in the input data. After carrying out a numerical simulation, it is then essential to control the error in order to certify the numerical results. We thus generally perform an estimation procedure aiming at quantifying this error, and, if necessary, a new adapted approximation space is produced in order to reduce this error and obtain more reliable results [1, 2, 3, 7]. The main objective of this work is to improve this estimation-adaptation procedure.

We propose to use techniques from the optimal transport community for optimal mesh refinement. The idea stems from the observation that during a standard estimation-adaptation procedure, the errors tend to be equally distributed among the elements. Indeed, starting from a uniform mesh exhibiting non-uniform errors, the procedure tends to produce a non-uniform mesh exhibiting uniform errors. In optimal transport, the objective is to estimate the lack of uniform density and to obtain a transformation to produce a uniform density. The optimal transport theory defines a metric over the space of finite measures allowing to continuously modify a measure (e.g. a Gaussian measure) into a target measure (e.g. a uniform measure) [5, 6]. We thus anticipate that meshes generated using optimal transport techniques could compete with those obtained using standard adaptation procedures by explicitly targeting the objective of equidistributing the errors.

To this end, after presenting the model problem and notations in Section 2, we introduce in Section 3 our method for uniformizing the error by relocating mesh vertices using an optimal transport approach. Then in Section 4 we combine this idea with the more standard h-adaptive algorithm thus yielding an hr-adaptive method. Numerical examples on selected academic problems are provided in Section 5.

2 Model Problem and Notations

Consider an abstract problem written in weak form as

$$\text{Find } u \in V \text{ such that } a(u,v) = f(v), \quad \forall v \in V, \tag{1}$$

where $(V, \|\cdot\|)$ is a Hilbert space, and bilinear form a and linear form f satisfy the usual regularity assumptions : a is continuous and coercive and f is continuous over V. This problem will be referred to

as the *primal problem* and its well-posedness is ensured by the Lax-Milgram theorem.

We now turn to the finite element formulation of the primal problem (1). Here, and in the remainder of the abstract, we consider a general conforming finite element space $V_h = \text{span}\{\varphi_i\}_{i=1}^N \subset V$, where φ_i, $i = 1,\dots,N$, are basis functions of V_h, e.g. the standard linear shape functions associated to a collection of N vertices $Y = \{y_i\}_{i=1}^N$. The classical finite element problem associated to the primal problem (1) is given by

$$\text{Find } u_h \in V_h \text{ such that } a(u_h, v_h) = f(v_h), \quad \forall v_h \in V_h.$$

The discretization error e_h associated to the finite element solution u_h is defined by

$$e_h = \|u - u_h\|. \tag{2}$$

The standard approach in error estimation and adaptivity proceeds in three main steps : estimate, mark, refine. First, local estimates of the error are computed for each element K, for instance $e_h = \sqrt{\sum_K e_{h,K}^2} = \sqrt{\sum_K \|u-u_h\|_K^2}$, where $\|\cdot\|_K$ is a local norm on element K. Among the local contributions $e_{h,K}$, the largest ones are marked for refinement and the corresponding elements are refined usually by splitting them in the middle. This defines a new approximation space where a new finite element solution can be computed.

Of course in practice the exact solution u is usually unknown, and we rely on a refined approximation $\tilde{u}_h$ defined as the solution of

$$\text{Find } \tilde{u}_h \in \tilde{V}_h \text{ such that } a(\tilde{u}_h, \tilde{v}_h) = f(\tilde{v}_h), \quad \forall \tilde{v}_h \in \tilde{V}_h,$$

where $\tilde{V}_h$ is an intermediate space between V_h and V : we have $V_h \subset \tilde{V}_h \subset V$. In our numerical experiments, we will take as V_h the space spanned by linear shape functions, while $\tilde{V}_h$ can be spanned by quadratic shape functions. The local contributions $e_{h,K}$ can then be replaced by the computable contributions $\tilde{e}_{h,K} = \|\tilde{u}_h - u_h\|_K$. We note that there are more advanced techniques for estimating the local errors, e.g. equilibrated flux reconstruction [3]. However, the focus of the present study is primarily on mesh adaptation, and thus for simplicity we use the exact error $e_h = \sqrt{\sum_K e_{h,K}^2}$ or the error estimator $\tilde{e}_h = \sqrt{\sum_K \tilde{e}_{h,K}^2}$.

3 An *r*-Adaptive Method Based on Optimal Transport

In this section, we introduce the proposed relocation method. For simplicity, we present the ideas in a 1D setting, but the method naturally extends to higher dimensions.

Starting with a given mesh Y with N vertices, ideally we would like to find the optimal position of the $N-2$ interior points that minimizes the error (2). As stated in the introduction, standard approaches in the estimation-adaptation community tend to produce meshes where the local contributions $e_{h,K}$ are more or less equal ; indeed the meshes obtained have large elements where the solution is smooth, while smaller elements are needed where the solution is more rough. We note that minimizing is not the same as equidistributing, nevertheless the two concepts often yield close results.

Consequently, our idea is to use techniques from the optimal transport community in order to equidistribute the errors.

3.1 The Vector Quantization Problem

In signal processing, the vector quantization problem (aka the optimal sampling problem) consists in finding the collection of points $Y = \{y_i\}_{i=1}^N$ that best sample a given domain Ω relative to a specific measure μ. In fact, this approach corresponds to the semi-discrete optimal transport problem [4], which involves solving the following optimization problem

$$\text{Find } Y^* \text{such that } Y^* = \underset{Y}{\text{argmin}}\, Q(Y), \tag{3}$$

where $Q(Y)$, called the quantization noise power, measures how good the points y_i are at sampling domain Ω under measure μ and is defined by

$$Q(Y) = \int_\Omega \min_i \|x - y_i\|^2 d\mu = \sum_{i=1}^{N} \int_{\mathrm{Vor}(y_i)} \|x - y_i\|^2 d\mu,$$

where $\mathrm{Vor}(y_i)$ is the Voronoi cell of y_i. The measure μ should locally reflect the error so we consider in this work $d\mu|_K = e_{h,K}^2 h_K^q dx|_K$ with h_K the size of element K and q an integer to define later.

To solve minimization problem (3), we have to compute the gradient with respect to each optimization variable $y_i,\ i = 2,\ldots,N-1$:

$$\nabla_{y_i} Q(Y) = 2m_i(y_i - g_i), \tag{4}$$

where $m_i = \mu(\mathrm{Vor}(y_i)) = \int_{\mathrm{Vor}(y_i)} d\mu$ represents the mass of the Voronoi cell $\mathrm{Vor}(y_i)$ and $g_i = \frac{1}{m_i}\int_{\mathrm{Vor}(y_i)} x d\mu$ represents the centroid of the Voronoi cell $\mathrm{Vor}(y_i)$.

Again, we emphasize that the optimization problem (3), the quantization noise power Q as well as its gradient computed in (4), the choice of measure μ, the Voronoi cells $\mathrm{Vor}(y_i)$, their masses m_i and centroids g_i, all naturally extend to higher dimensions.

3.2 Relocation Adaptivity with Fixed N

Starting from a given mesh with a fixed number of vertices N, we are first interested in solving optimization problem (3), for which we considered three approaches to iteratively update the positions of the interior mesh points. The relocation process aims at equidistributing the local errors by moving the mesh vertices to a target location depending on measure μ, itself depending on the errors. In essence the nodes are pursuing a moving target.

1. *Direct Method* : $\nabla Q = 0$ i.e. set $y_i = g_i$, $\forall i = 2,\ldots,N-1$, a direct substitution following (4).
2. *Gradient Scheme* :

$$y_{\mathrm{new},i} = y_{\mathrm{old},i} - \frac{\lambda}{2m_i}\nabla_{y_i} Q(Y_{\mathrm{old}}), \quad \forall i = 2,\ldots,N-1,$$

   where m_i is used to re-scale the step size and $\lambda \in]0,1]$ controls the step size, being a user-defined parameter. Note that if we take $\lambda = 1$, we end up with the direct case $\nabla Q = 0$.
3. *Quasi-Newton's Method* :

$$Y_{\mathrm{new}} = Y_{\mathrm{old}} - \lambda\, H(Y_{\mathrm{old}})^{-1}\nabla Q(Y_{\mathrm{old}}),$$

   with H the approximated Hessian matrix of Q and $\lambda \in]0,1]$ controls the step size, a user-defined parameter.

Unfortunately, the Direct Method often resulted in instabilities and oscillations due to the nodes moving too quickly. Thus we resorted to using the Gradient Scheme and Quasi-Newton's Method with step lengths to stabilize the behavior of the method.

Performing a one-time relocation of the nodes of a mesh with a fixed value of N using one of these methods is referred to as a single "r-step" process. In Section 5.1, we will numerically illustrate the performance of the relocation method.

4 An *hr*-Adaptive Algorithm

In the previous section, we designed a relocation procedure for a fixed number of mesh points N that can produce a mesh with equidistributed errors in the domain and overall a relatively small global error compared to the starting mesh. However this resulting error could still be large compared to a prescribed tolerance. Thus, we describe in this section an algorithm embedding the aforementioned *r*-step within a larger *h*-refinement loop, hence the name *hr*-adaptive algorithm.

Algorithm 1: *hr*-Adaptivity

Data: starting mesh Y, global error e_h, local errors $e_{h,K}$, tolerance ε, parameters $0 < \beta, \gamma < 1$
Result: new mesh Y, new global error e_h

```
while e_h > ε do
    while min_K e_{h,K} < β max_K e_{h,K} do
        Relocate the nodes using one of the relocation methods
        Compute the new finite element solution and the local errors e_{h,K}
    end
    Mark the elements K where e_{h,K} > γ max_K e_{h,K}
    Refine the mesh by splitting the marked elements K in half
    Compute the new finite element solution and the errors e_{h,K} and e_h
end
```

In Algorithm 1 the parameters β and γ control respectively the equidistribution of the local errors and the proportion of elements to be h-refined.

5 Numerical Results in 1D

These methods were implemented and tested on various academic problems including a singularity problem (5) and a boundary layer problem (7).

The singularity problem : Find u such that

$$\begin{cases} -u'' = -\alpha(\alpha-1)x^{\alpha-2} \text{ on } (0,1), \\ u(0) = 0;\ u'(1) = \alpha, \end{cases} \tag{5}$$

where $\frac{1}{2} < \alpha < \frac{3}{2}, \alpha \neq 1$. The exact solution of (5) is given by (6) and represented in Figure 1a

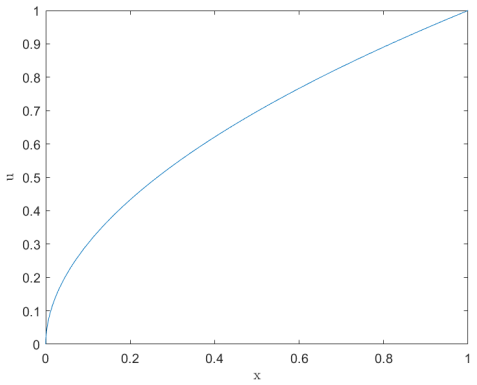
$$u_{ex}(x) = x^{\alpha}. \tag{6}$$

The boundary layer problem : Find u such that

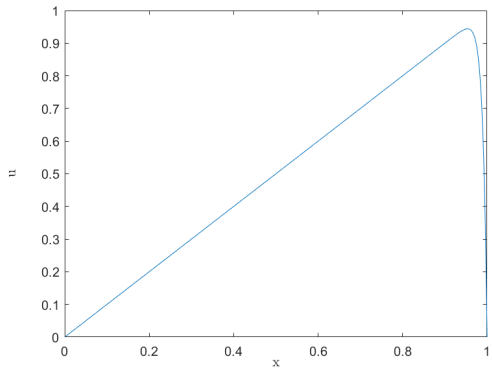
$$\begin{cases} -\varepsilon u'' + u' = 1 \text{ on } (0,1), \\ u(0) = 0;\ u(1) = 0, \end{cases} \tag{7}$$

where the parameter ε controls the size of the boundary layer on the right-hand side of the domain. The exact solution of (7) is given by (8) and represented in Figure 1b

$$u_{ex}(x) = x - \frac{e^{-1/\varepsilon} - e^{(x-1)/\varepsilon}}{e^{-1/\varepsilon} - 1}. \tag{8}$$



(a) Exact solution of the singularity problem with $\alpha = 0.52$.



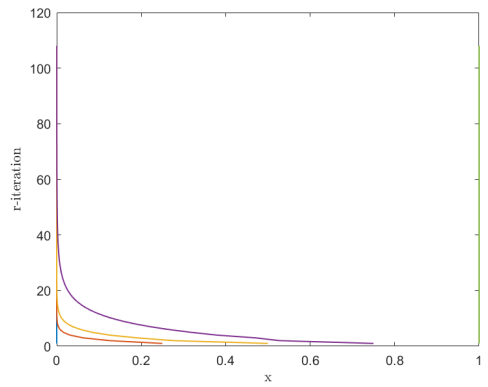
(b) Exact solution of the boundary layer problem with $\varepsilon = 0.01$.

FIGURE 1 – Exact solutions of problems (5) and (7).

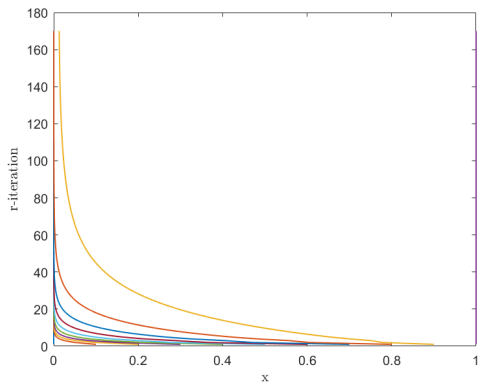
All the numerical results presented in Section 5 were obtained using the exact errors $e_{h,K}$. Note that we also conducted experiments using estimates $\tilde{e}_{h,K}$ rather than exact errors (not shown) and obtained very similar results. We also implemented and tested the relocation algorithm using entire patches instead of Voronoi cells (not shown) and obtained very similar results.

5.1 *r*-Adaptive Numerical Results

To investigate the performances of the relocation methods described in Section 3.2, we applied them to the problems (5) and (7) and obtained the best results in terms of number of iterations and stability using Quasi-Newton's Method. See Figures 2 and 3.

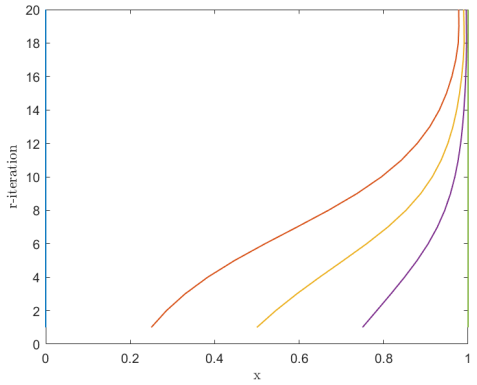


(a) Trajectories of small number of nodes $N = 5$.

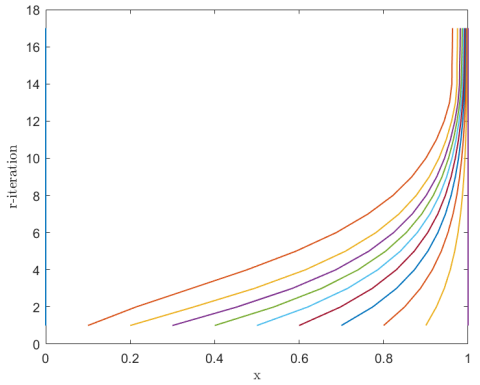


(b) Trajectories of larger number of nodes $N = 11$.

FIGURE 2 – Node trajectories (positions with respect to r-iteration numbers) for the singularity problem (5) using Quasi-Newton's Method with $q = -2$ and step length $\lambda = 1$.



(a) Trajectories of small number of nodes $N = 5$.



(b) Trajectories of larger number of nodes $N = 11$.

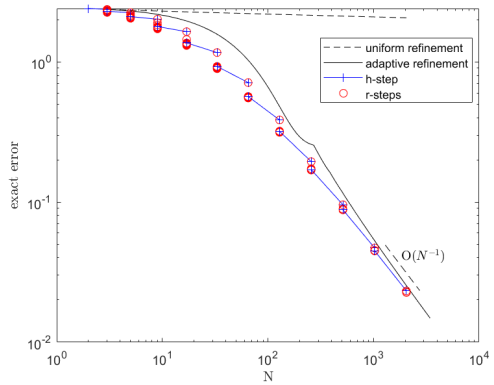
FIGURE 3 – Node trajectories (positions with respect to r-iteration numbers) for the boundary layer problem (7) using Quasi-Newton's Method with $q = -2$ and step length $\lambda = 0.5$.

As can be observed qualitatively in Figures 2 and 3, the nodes tend to move to the regions where the solution is not smooth. In quantitative terms, the relative exact errors decrease from 34.54% to 31.19% in Fig. 2a, from 33.91% to 24.64% in Fig. 2b, from 27.38% to 3.03% in Fig. 3a and from 16.8% to 1.14% in Fig. 3b.

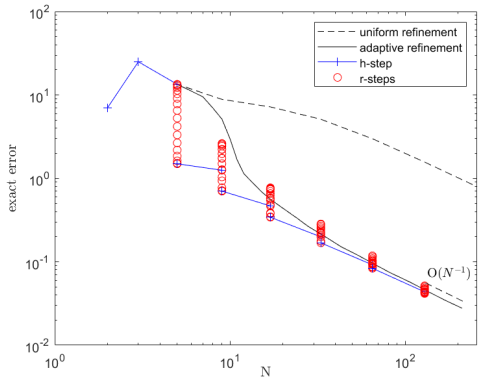
5.2 *hr*-Adaptive Numerical Results

We now apply Algorithm 1 to the two problems described above and display in Figure 4 the exact errors with respect to the number of mesh points N using Quasi-Newton's Method with $q = -2$, and $\lambda = 1$ for the singular problem and $\lambda = 0.5$ for the boundary layer problem. Also, we set $\gamma = 0.5$ and

$\beta = 0.8$ the parameters that controls respectively *h*- and *r*-refinements. For comparison we additionally include the standard uniform refinement as well as the adaptive refinement strategy also with $\gamma = 0.5$.



(a) *hr*-adaptivity and refinement methods on the singularity problem.



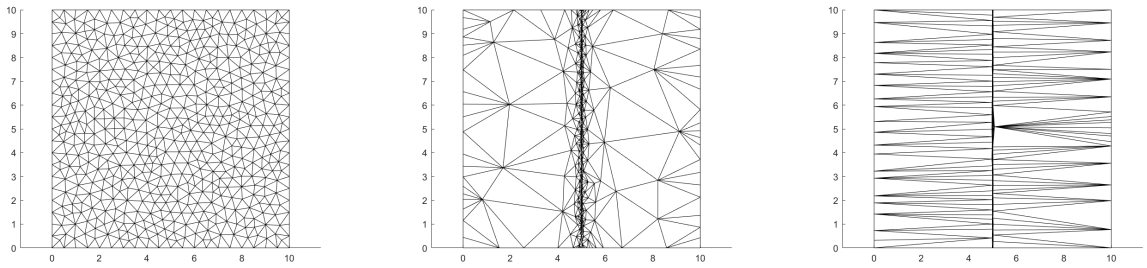
(b) *hr*-adaptivity and refinement methods on the boundary layer problem.

FIGURE 4 – Exact error with respect to the number of mesh points N when applying uniform refinement, adaptive refinement and *hr*-adaptivity using Quasi-Newton's method, $q = -2$ and step size $\lambda = 1$ for singular problem (left) or $\lambda = 0.5$ for boundary layer problem (right).

We observe in Figure 4 that the approach based on *hr*-adaptivity yields the best results, even outperforming the standard adaptive refinement procedure in terms of number of degrees of freedom. For both problems, the newly developed *hr*-adaptive method is able to recover quickly the optimal order of convergence $e_h \approx CN^{-1}$, especially on the boundary layer problem where it enters the asymptotic regime much sooner than the standard techniques.

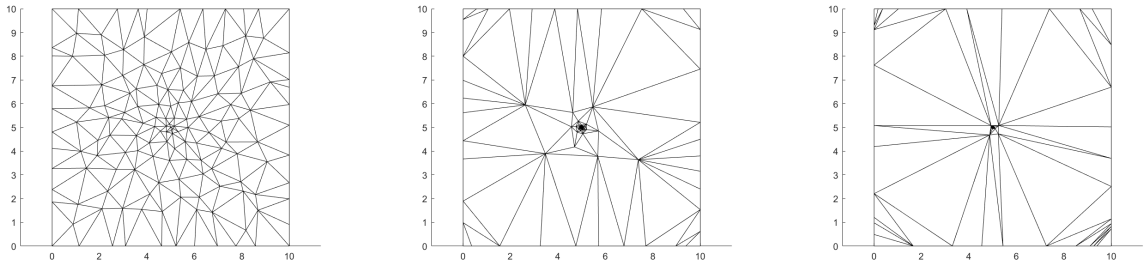
6 Numerical Results in 2D

We have started implementing the newly developed *r*-adaptive approach in the 2D case and present here preliminary results obtained on two examples. Note that these are partial results in the sense that so far we have not yet used the discretization error to guide the movement of the nodes. Instead we used an error defined with respect to a prescribed element size. In the first example, mimicking an interface problem with a vertical interface placed in the middle of domain $\Omega = [0,10] \times [0,10]$, the prescribed element size is a function of x only, first linearly decreasing for $x \in [0,5]$ and then linearly increasing for $x \in [5,10]$ symmetrically. In the second example the prescribed element size is a linearly increasing radial function centered in the middle of domain $\Omega = [0,10] \times [0,10]$. We emphasize that after each *r*-step, a Delaunay triangulation is performed based on the newly obtained positions of nodes in order to control element quality. Finally, these examples were obtained using the direct method described in Section 3.2 on entire patches rather than Voronoi cells, as we saw in the 1D cases that this did not affect the numerical results too much. The results of the *r*-adaptive method are presented in Figure 5 for the first example and in Figure 6 for the second example. We also provide in Figure 7 a close-up view of the center of domain Ω for both problems.



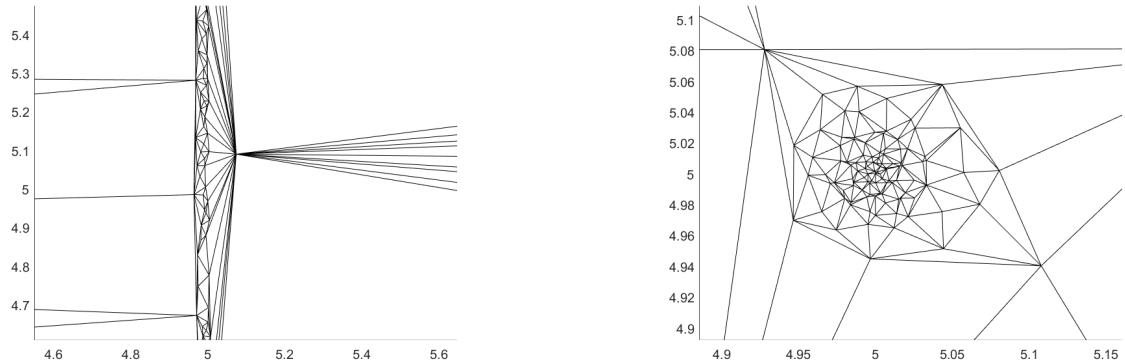
(a) (b) (c)

FIGURE 5 – Initial mesh (a), intermediate mesh (b) and final mesh (c) obtained for the first 2D problem.



(a) (b) (c)

FIGURE 6 – Initial mesh (a), intermediate mesh (b) and final mesh (c) obtained for the second 2D problem.



(a) (b)

FIGURE 7 – Zoom on the center of domain Ω for first problem (a) and second problem (b).

In all cases, we observe that the nodes are moved to the desired region of domain Ω as expected.

7 Conclusion and Future Work

We have developed a new method for relocating mesh vertices and have applied it to several problems in 1D, namely a singularity problem as well as a boundary layer problem. We obtained very satisfying results qualitatively, namely the nodes concentrated where the solution exhibited rough behavior. Then we designed an *hr*-algorithm combining this concept with mesh refinement and applied it to the aforementioned problems. Again the results were satisfying in terms of error reduction and number of degrees of freedom when compared to standard techniques. Finally we considered 2D examples and obtained encouraging results.

Future work will be dedicated to a comprehensive analysis of the methods developed and the numerical results obtained. We plan to test the proposed approach on more complex problems, such as non-linear problems, the extension to higher dimensions, to higher order elements, or the application to the estimation of quantities of interest of the solution.

References

[1] M. Ainsworth, J. T. Oden. *A posteriori error estimation in finite element analysis*, John Wiley & Sons, 2001.

[2] I. Babuška, W. C. Rheinboldt. *A posteriori error estimates for the finite element method*, International journal for numerical methods in engineering, 12, 1597-1615, 1978.

[3] P. Ladevèze, D. Leguillon. *Error Estimate Procedure in the Finite Element Method and Applications*, SIAM Journal on Numerical Analysis, 20(3), 485-509, 1983.

[4] B. Lévy. *A numerical algorithm for L_2 semi-discrete optimal transport in 3D*, ESAIM : Mathematical Modelling and Numerical Analysis, 49(6), 1693-1715, 2015.

[5] G. Peyré, M. Cuturi. *Computational Optimal Transport. With Applications to Data Science*, Found. Trends Mach. Learn. 11, No. 5-6, 1-262, 2018

[6] F. Santambrogio. *Optimal Transport for Applied Mathematicians. Calculus of Variations, PDEs, and Modeling*, Cham : Birkhäuser/Springer, Zbl 1401.49002, 2015.

[7] R. Verfürth. *A review of a posteriori error estimation techniques for elasticity problems*, Computer Methods in Applied Mechanics and Engineering , 176(1-4) :419-440, 1999.