Implementation of the Mortar method in the wavelet context

Original
Implementation of the Mortar method in the wavelet context / S., Bertoluzza; Falletta, Silvia; V., Perrier. - In: JOURNAL OF SCIENTIFIC COMPUTING. - ISSN 0885-7474. - STAMPA. - 29:2(2006), pp. 219-255.

Publisher:
Springer

Published
DOI:

Implementation of the Mortar method in the wavelet context

Silvia Bertoluzza*, Silvia Falletta†and Valérie Perrier‡

Abstract

The paper is concerned with non-conforming wavelet-type discretization of elliptic partial differential equations. In particular it analyzes some implementation issues related to the form of the constraint operator in the Mortar approach. Moreover it gives a preliminary example of the coupling of wavelets with finite elements in non-trivial geometries.

keywords: Domain decomposition, multiresolution analysis.
AMS subject classification: 65N55, 65T60.

1 Introduction

Domain decomposition methods are powerful tools for solving systems of algebraic equations arising from the discretization of partial differential equations. The computational domain is decomposed into overlapping or non-overlapping subdomains and the original problem is divided into, or assembled from, smaller subproblems corresponding to these subdomains. In this paper, we focus on a particular domain decomposition method, the Mortar method, which is of non-conforming type and therefore particularly well suited for coupling different variational approximations in different subdomains and non-matching discretizations along interior interfaces.
In the Mortar method, point-wise continuity across the interfaces is replaced by a weaker condition. More precisely, the interface of the decomposition is itself decomposed into disjoint "multiplier sides", each one being the whole edge of a given subdomain. Weak continuity is imposed by requiring that the jump on each multiplier side of a discrete function is

*I.M.A.T.I.-C.N.R., v. Ferrata 1, Pavia, Italy. *Email: aivlis@imati.cnr.it*
†Dip. Matematica e Applicazioni, Via R. Cozzi 53, Milano Bicocca. *Email: falletta@dimat.unipv.it*
‡LMC-IMAG, Domaine universitaire, BP 53 - 38 041 Grenoble cedex 9, France. *Email: Valerie.Perrier@imag.fr*

orthogonal to a suitable "multiplier space", generally defined as a subspace of the trace of the discretization in the corresponding subdomain. Such method has nowadays been applied to a wide variety of real-life problems in both two and three dimensions ([2, 29, 5, 6, 7, 3]) and it is well suited for parallel implementation ([1]).

It seems then natural to try such a method in order to overcome one of the more severe drawback of wavelet methods, namely the difficulty in the treatment of complex geometries.
Wavelet based methods are in fact nowadays widely used for the numerical solution of partial differential equations. Such bases have several attractive features, among which the existence of diagonal preconditioners for elliptic operators of any given order ([24], [19]), as well as the possibility of designing efficient adaptive approximation schemes for different types of problems ([25],[14], [18], [8], [12], [10]). But several drawbacks still limit their application to real life problems, among which the treatment of non-trivial geometries. We recall that such limitations are due to the intrinsic nature of the wavelets that were first introduced as bases of $L^2(\mathbb{R})$ (see [27]) and lately generalized to $L^2(]0,1[)$ and to $L^2(]0,1[^d)$ (see [16] and [4]): roughly speaking, wavelets for the interval are obtained by modifying the basis defined on the whole real line with suitable modifications of the basis functions that "interact" with the boundaries. Furthermore, the multidimensional case involves a tensorial basis that is derived from the mono-dimensional one, thus limiting their application to tensorial shaped domains. A possible remedy is then to resort to domain decomposition. Conforming domain decomposition methods have been proposed by several authors ([13, 17, 22]). However, the constructions involved are extremely technical and only work if the domain can be split as the union of tensor-like subdomains, which is not always practical (for a very complicated geometry the number of subdomains could be too high). By resorting to non-conforming domain decomposition one can relax the continuity requirement and obtain a lighter construction. Moreover, in principle, coupling with other methods in order to deal with complex geometries is possible ([9]). In [11] the authors studied the problem of the use of wavelet bases within the Mortar method. In particular they defined a multiplier space with all the properties needed to prove optimal error estimates for the resulting method.

The aim of the paper is to deal in detail with some implementation issues related to the Mortar method in the wavelet context. After recalling the main theoretical results, we will study the form of the constraint operator and we will show numerical results for the Laplace operator for several cases of domain decomposition of the unit square and with different choices of wavelet bases. Finally, we will demonstrate the feasibility of the non-conforming method in complex geometries showing an example of the coupling of wavelets with finite elements in such a way that the limitation of multiresolution approximation spaces to trivial (tensor-like) geometries is overcome.

2 The Mortar Wavelet Method

2.1 Approximation spaces in the wavelet context

We introduce in this section the wavelet spaces which we will need for defining the mortar wavelet method.

2.1.1 Wavelet settings

In this section we recall the main ingredients of the definition of wavelet on the interval, and focus on practical aspects. For more details we refer to the series of papers on the subject [16, 26, 28, 20].

Scaling functions on the interval We start with a couple of biorthogonal Multiresolution Analyses (MRA) of $L^2(0,1)$ ([15]), that is a couple of increasing sequences of finite dimensional approximation subspaces $V_{j_0} \subset \dots V_j \subset V_{j+1} \dots$ and $\tilde{V}_{j_0} \subset \dots \tilde{V}_j \subset \tilde{V}_{j+1} \dots$, whose respective unions are dense in $L^2(0,1)$. We assume also that:

$$\begin{aligned} V_j &= span < \varphi_{j,k}, \quad k = 0, \cdots, 2^j + 1 >, \\ \tilde{V}_j &= span < \tilde{\varphi}_{j,k}, \quad k = 0, \cdots, 2^j + 1 >, \end{aligned}$$

where the corresponding scaling function bases $\{\varphi_{j,k} \ ; \ k = 0, \dots 2^j + 1\}$ and $\{\tilde{\varphi}_{j,k} \ ; \ k = 0, \dots 2^j + 1\}$, are compactly supported and assumed to be biorthogonal , *i.e.* they verify:

$$\int_0^1 \varphi_{j,k} \tilde{\varphi}_{j,k'} = \delta_{kk'}, \quad \forall k, k'.$$

Remark 2.1 Roughly speaking, we recall that the construction of multiresolution on the interval is based on compactly supported multiresolution in $L^2(\mathbb{R})$, by suitably modifying the scaling functions that cross the boundaries, and it works under the assumption that the modifications made at boundary 0 do not interact with the ones made at boundary 1. In general, such modifications are done in such a way that $\#V_j = \#\tilde{V}_j = 2^j + 2 - \alpha$, where α is a suitable parameter depending on the support size of the scaling functions. For simplicity, we assume here that the above mentioned modifications are done in such a way that $\#V_j = \#\tilde{V}_j = 2^j + 2$.

We will assume that these are Riesz's bases , that is the two following norm equivalences hold uniformly in j :

$$\left\| \sum_{k=0}^{2^j+1} u_k \varphi_{j,k} \right\|_{0,]0,1[} \simeq \left(\sum_{k=0}^{2^j+1} |u_k|^2 \right)^{1/2} \tag{1}$$

and

$$\left\| \sum_{k=0}^{2^j+1} u_k \tilde{\varphi}_{j,k} \right\|_{0,]0,1[} \simeq \left(\sum_{k=0}^{2^j+1} |u_k|^2 \right)^{1/2} \tag{2}$$

$\|.\|_{0,]0,1[}$ being the L^2-norm on $[0,1]$.

We will make the following additional standard assumptions on V_j and $\tilde{V}_j$:
given two integers n, $\tilde{n}$, and two reals R, $\tilde{R}$ such that $n \geq R > 1$ and $\tilde{n} \geq \tilde{R} > 0$ we suppose:

$$\varphi_{j,k} \in H^R(0,1) \text{ and } \tilde{\varphi}_{j,k} \in H^{\tilde{R}}(0,1), \qquad \forall j \geq j_0, \quad k = 0, \ldots, 2^j + 1.$$

Moreover the spaces V_{j_0} and $\tilde{V}_{j_0}$ satisfy a Strang-Fix condition of order n and $\tilde{n}$ respectively, that is polynomials up to order n and $\tilde{n}$ are included in V_{j_0} and $\tilde{V}_{j_0}$ respectively (by inclusion so do V_j and $\tilde{V}_j$), which means: there exists coefficients a_k^p, b_k^p, and $\tilde{a}_k^p$, $\tilde{b}_k^p$, such that we have for $p = 0, \ldots, n-1$

$$2^{\frac{j_0}{2}} (2^{j_0} x)^p = \sum_{k=0}^{2^{j_0}+1} a_k^p \, \varphi_{j_0,k}(x), \quad \text{and} \quad 2^{\frac{j_0}{2}} \left(2^{j_0}(1-x)\right)^p = \sum_{k=0}^{2^{j_0}+1} b_k^p \, \varphi_{j_0,k}(x), \tag{3}$$

and for $p = 0, \ldots, \tilde{n} - 1$:

$$2^{\frac{j_0}{2}} (2^{j_0} x)^p = \sum_{k=0}^{2^{j_0}+1} \tilde{a}_k^p \, \tilde{\varphi}_{j_0,k}(x), \quad \text{and} \quad 2^{\frac{j_0}{2}} \left(2^{j_0}(1-x)\right)^p = \sum_{k=0}^{2_0^{j}+1} \tilde{b}_k^p \, \tilde{\varphi}_{j_0,k}(x). \tag{4}$$

Remark that the coefficients a_k^p, b_k^p, $\tilde{a}_k^p$ and $\tilde{b}_k^p$, which will play a role in the following, are available in practice and hence we can assume they are known.

From now on we will set $N = max(n, \tilde{n})$.

We recall that the scaling functions $\varphi_{j,k}$ and $\tilde{\varphi}_{j,k}$ are constructed by modification of the corresponding scaling functions on $L^2(\mathbb{R})$, which are obtained by integer translation and dyadic dilation of two "father" scaling functions φ and $\tilde{\varphi} \in L^2(\mathbb{R})$.

By construction the scaling functions have compact support

$$diam(supp\varphi_{j,k}) \simeq 2^{-j} \qquad \text{and} \qquad diam(supp\tilde{\varphi}_{j,k}) \simeq 2^{-j},$$

and are scale invariant, *i.e.* $\forall j \geq j_0$, $\forall x \in [0,1]$, and $\forall k = 0, \ldots, N-1$ on the left boundary it holds

$$\varphi_{j,k}(x) = 2^{\frac{j-j_0}{2}} \varphi_{j_0,k}(2^{j-j_0}x), \qquad \tilde{\varphi}_{j,k}(x) = 2^{\frac{j-j_0}{2}} \tilde{\varphi}_{j_0,k}(2^{j-j_0}x), \tag{5}$$

while on the right boundary we have $\forall k = 0, \ldots, N-1$

$$\varphi_{j,2^j-k+1}(1-x) = 2^{\frac{j-j_0}{2}} \varphi_{j_0,2^{j_0}-k+1}(2^{j-j_0}(1-x)), \tag{6}$$

$$\tilde{\varphi}_{j,2^j-k+1}(1-x) = 2^{\frac{j-j_0}{2}} \tilde{\varphi}_{j_0,2^{j_0}-k+1}(2^{j-j_0}(1-x)). \tag{7}$$

Moreover, the *interior* scaling functions (*i.e.* scaling functions whose support is included into [0,1]) coincide with the original scaling functions on the real line and $\forall j \geq j_0$, $\forall k = N, \ldots, 2^j - N + 1$ they take the form:

$$\varphi_{j,k}(x) = \varphi_{j,N}(x - 2^{-j}(k-N)) = 2^{j/2}\varphi(2^j x - k),$$

where $\varphi(x) = 2^{-j_0/2}\varphi_{j_0,N}(2^{-j_0}(x+N))$ is the scaling function of the corresponding multi-scale analysis for $L^2(\mathbb{R})$. An analogous relation holds for the duals $\tilde{\varphi}_{j,k}$, for $k = N, \ldots, 2^j - N + 1$.

Finally we can also suppose that all scaling functions of V_j vanish at the edges 0 and 1, except one function at each edge. To fix the ideas we will assume:

$$\begin{aligned} &\varphi_{j,0}(0) \neq 0 \quad \text{and} \quad \varphi_{j,2^j+1}(1) \neq 0\ , \quad \forall j \geq j_0, \\ &\varphi_{j,k}(0) = 0 \quad \text{and} \quad \varphi_{j,k}(1) = 0\ , \quad \forall k = 1, \ldots, 2^j. \end{aligned} \tag{8}$$

It is well known that the above assumptions imply that the projectors $P_j : L^2(0,1) \to V_j$ and $\tilde{P}_j : L^2(0,1) \to \tilde{V}_j$ defined by

$$P_j f = \sum_{k=0}^{2^j+1} < f, \tilde{\varphi}_{j,k} > \varphi_{j,k}, \qquad \tilde{P}_j f = \sum_{k=0}^{2^j+1} < f, \varphi_{j,k} > \tilde{\varphi}_{j,k}, \tag{9}$$

satisfy the following direct estimates for all $u \in H^t(0,1)$:

$$\begin{aligned} \|u - P_j u\|_{s,]0,1[} &\lesssim 2^{-j(t-s)}\|u\|_{t,]0,1[} \qquad \text{if } s \leq R, \qquad s < t \leq n+1 \\ \|u - \tilde{P}_j u\|_{s,]0,1[} &\lesssim 2^{-j(t-s)}\|u\|_{t,]0,1[} \qquad \text{if } s \leq \tilde{R}, \qquad s < t \leq \tilde{n}+1. \end{aligned}$$

Wavelets on the interval Biorthogonal wavelet bases $\psi_{j,k}$ and $\tilde{\psi}_{j,k}$ are constructed as bases of the spaces $W_j = V_{j+1} \cap (\tilde{V}_j)^{\perp}$ and $\tilde{W}_j = \tilde{V}_{j+1} \cap (V_j)^{\perp}$. They verify the same translation and dilation invariance properties as for the scaling functions, that is:

- For $k = N-1, \ldots, 2^j - N$ the *interior* wavelets of W_j coincide with the original wavelets on the real line and they take the form:

$$\psi_{j,k}(x) = \psi_{j,N}(x - 2^{-j}(k-N)) = 2^{j/2}\psi(2^j x - k),$$

where $\psi(x) = 2^{-j_0/2}\psi_{j_0,N}(2^{-j_0}(x+N))$ is the wavelet of the corresponding multi-scale analysis for $L^2(\mathbb{R})$. An analogous relation holds for the duals $\tilde{\psi}_{j,k}$, for $k = N-1, \ldots, 2^j - N$, which span the space $\tilde{W}_j$.

- On the left boundary, the left edge wavelets of W_j: $\psi_{j,k}$, $k = 0, \ldots, N-2$ verify a relation of the form (5), in which $\varphi_{j,k}$ is replaced by $\psi_{j,k}$ (and similarly for $\tilde{\psi}_{j,k}$, $k = 0, \ldots, N-2$).

- On the right boundary, the right edge wavelets of W_j: $\psi_{j,2^j-k-1}$, $k = 0, \ldots, N-2$ verify a relation of the form (6), in which $\varphi_{j,2^j-k+1}$ is replaced by $\psi_{j,2^j-k-1}$ (and similarly for $\tilde{\psi}_{j,2^j-k-1}$, $k = 0, \ldots, N-2$).

Consequently, $L^2(0,1)$ will be spanned by wavelet bases, and we will use the decompositions:

$$L^2(0,1) = V_{j_0} \bigoplus_{j=j_0}^{+\infty} W_j = \tilde{V}_{j_0} \bigoplus_{j=j_0}^{+\infty} \tilde{W}_j.$$

Moreover for all $J > j_0$, the projectors P_J and $\tilde{P}_J$ of (9) rewrite:

$$P_J f = P_{j_0} f + \sum_{j=j_0}^{J-1} \sum_{k=0}^{2^j-1} < f, \tilde{\psi}_{j,k} > \psi_{j,k}, \tag{10}$$

$$\tilde{P}_J f = \tilde{P}_{j_0} f + \sum_{j=j_0}^{J-1} \sum_{k=0}^{2^j-1} < f, \psi_{j,k} > \tilde{\psi}_{j,k}.$$

Fast wavelet transform The wavelet coefficients $< f, \tilde{\psi}_{j,k} >$ and $< f, \psi_{j,k} >$ of a given function f, involved in the wavelet decompositions (10) are computed in the same way as on the real line, except for a specific treatment of the (left and right) boundary coefficients, which necessitates (small size-)matrix vector products. For the sake of completeness, we remind here the practical way to compute filters and to deal with the Fast Wavelet Algorithm. We will focus on the computation of the $< f, \psi_{j,k} >$, since the one of the $< f, \tilde{\psi}_{j,k} >$ carries on the same manner.

Two-scale equations :

For each scale $j > j_0$, the families $\{\varphi_{j,k}\}$ and $\{\psi_{j,k}\}$ satisfy a two scale equation. In particular, for the internal functions it holds that (recall that they coincide with the functions defined on the whole real line)

$$\frac{1}{\sqrt{2}}\varphi_{j,k}(\frac{x}{2}) = \sum_p h_{p-2k}\varphi_{j,p}(x),$$

and

$$\frac{1}{\sqrt{2}}\psi_{j,k}(\frac{x}{2}) = \sum_p g_{p-2k}\varphi_{j,p}(x).$$

The family of left-edge scaling functions $\{\varphi_{j,k}\}_{k=0,N-1}$ satisfies the two-scale equation:

$$\frac{1}{\sqrt{2}}\begin{pmatrix}\varphi_{j,0}\\ \vdots\\ \varphi_{j,N-1}\end{pmatrix}(\frac{x}{2}) = H_0\begin{pmatrix}\varphi_{j,0}\\ \vdots\\ \varphi_{j,N-1}\end{pmatrix}(x) + h_0\begin{pmatrix}\varphi_{j,N}\\ \vdots\\ \varphi_{j,3N-3}\end{pmatrix}(x) \quad (11)$$

where H_0 and h_0 are $N \times N$ and $N \times (2N-2)$ matrices.

We also have a two-scale equation for the left-edge wavelets:

$$\frac{1}{\sqrt{2}}\begin{pmatrix}\psi_{j,0}\\ \vdots\\ \psi_{j,N-2}\end{pmatrix}(\frac{x}{2}) = G_0\begin{pmatrix}\varphi_{j,0}\\ \vdots\\ \varphi_{j,N-1}\end{pmatrix}(x) + g_0\begin{pmatrix}\varphi_{j,N}\\ \vdots\\ \varphi_{j,3N-3}\end{pmatrix}(x). \quad (12)$$

where G_0 and g_0 are $(N-1) \times N$ and $(N-1) \times (2N-2)$ matrices.

For the edge 1 we use in fact the following notation:

$$\begin{aligned}\frac{1}{\sqrt{2}}\begin{pmatrix}\varphi_{j,2^j-N+2}\\ \vdots\\ \varphi_{j,2^j+1}\end{pmatrix}(2^j(1-x)) = \quad & H_1\begin{pmatrix}\varphi_{j,2^{j+1}-N+2}\\ \vdots\\ \varphi_{j,2^{j+1}+1}\end{pmatrix}(2^{j+1}(1-x))\\ + \quad & h_1\begin{pmatrix}\varphi_{j,2^{j+1}-3N+4}\\ \vdots\\ \varphi_{j,2^{j+1}-N+1}\end{pmatrix}(2^{j+1}x) \quad (13)\end{aligned}$$

and the analogous formula for G_1 and g_1:

$$\begin{aligned}\frac{1}{\sqrt{2}}\begin{pmatrix}\psi_{j,2^j-N+1}\\ \vdots\\ \psi_{j,2^j-1}\end{pmatrix}(2^j(1-x)) = \quad & G_1\begin{pmatrix}\varphi_{j,2^{j+1}-N+2}\\ \vdots\\ \varphi_{j,2^{j+1}+1}\end{pmatrix}(2^{j+1}(1-x))\\ + \quad & g_1\begin{pmatrix}\varphi_{j,2^{j+1}-3N+4}\\ \vdots\\ \varphi_{j,2^{j+1}-N+1}\end{pmatrix}(2^{j+1}x). \quad (14)\end{aligned}$$

Without going into a deep description, we recall that H_0, h_0, G_0, g_0, H_1, h_1, G_1, g_1 are suitable matrices depending on the standard filters on the line $\{h_k\}$ and $\{g_k\}$ and on the modifications made to build the boundary function bases. For more details see [28] and [26].

Numerically, all was done to find the above two-scale relations that give the matrices used in the fast algorithms. The practical computation of the filters for the specific Daubechies wavelets is detailed in annex.

Fast algorithms

We will now describe the fast algorithm for analyzing and synthesizing vectors in the spaces of the multiresolution analysis. As in the case of the real line, the algorithm is based on the elementary step:

$$\tilde{V}_j = \tilde{V}_{j-1} \oplus \tilde{W}_{j-1}$$

Suppose that we know the scalar products of a given function f in $\tilde{V}_j$ with the scaling functions φ_{jk}, and that we want to compute scalar products $\langle f, \varphi_{j-1,k}\rangle$ with the scaling functions of V_{j-1} and $\langle f, \psi_{j-1,k}\rangle$, with the wavelets of W_{j-1}. We start then from a vector c^j composed of:

- $\langle\ f \mid \varphi_{j,k}\ \rangle$ for $k = 0, N-1$, left-edge scaling coefficients
- $\langle\ f \mid \varphi_{j,k}\ \rangle$ for $N \leq k \leq 2^j - N + 1$, interior scaling coefficients
- $\langle\ f \mid \varphi_{j,2^j+1-k}\ \rangle$ right-edge scaling coefficients for $k = 0, N-1$.

To obtain the projection of the function f on $\tilde{V}_{j-1}$, we have to multiply the vector c^j by the matrix

$$\begin{pmatrix} H_0 & h_0 & 0 \\ 0 & \mathcal{H}^j & 0 \\ 0 & h_1 & H_1 \end{pmatrix}$$

where:

- $\mathcal{H}^j$ is the $(2^{j-1} - 2N + 2) \times (2^j - 2N + 2)$ matrix whose general term is $\mathcal{H}^j_{k,l} = h_{-N+2+l-2k}$,
- h_0 (defined, as H_0, in (11) for the edge 0) is completed with columns of 0 at the right to fit the width of $\mathcal{H}^j$,
- h_1 (defined, as H_1, in (13) for the edge 1) is completed with columns of 0 at the left.

To obtain the projection on W_{j-1}, we multiply c^j by the matrix

$$\begin{pmatrix} G_0 & g_0 & 0 \\ 0 & \mathcal{G}^j & 0 \\ 0 & g_1 & G_1 \end{pmatrix}$$

where

- $\mathcal{G}^j$ is the $(2^{j-1}-2N+2)\times(2^j-2N+2)$ matrix whose general term is $\mathcal{G}^j_{k,l}=g_{-N+2+l-2k}$,

- g_0 (defined, as G_0, in (12) for the edge 0) is completed with columns of 0 at the right to fit the width of $\mathcal{G}^j$,

- g_1 (defined, as G_1, in (14) for the edge 1) is completed with columns of 0 at the left.

The result lies in the vectors c^{j-1} and d^{j-1}.

Reversely, to find c^j starting from c^{j-1} and d^{j-1}, we have to multiply the column vector

$$\begin{pmatrix} c^{j-1} \\ d^{j-1} \end{pmatrix}$$

by the matrix

$$\begin{pmatrix} \tilde{H}_0 & 0 & 0 & \tilde{G}_0 & 0 & 0 \\ \tilde{h}_0 & \tilde{\mathcal{H}}^j & \tilde{h}_1 & \tilde{g}_0 & \tilde{\mathcal{G}}^j & \tilde{g}_1 \\ 0 & 0 & \tilde{H}_1 & 0 & 0 & \tilde{G}_1 \end{pmatrix}$$

where the matrices $\tilde{\mathcal{H}}^j$ and $\tilde{\mathcal{G}}^j$ (which have the same size as $\mathcal{H}^j$ and $\mathcal{G}^j$) are built from the standard filters on the line $\tilde{h}_k$ and $\tilde{g}_k$ and $\tilde{h}_0,\tilde{g}_0,\tilde{h}_1,\tilde{g}_1$ are completed with columns of zeros, as above.
Notice that this algorithm is not applied as multiplications of matrices. It is worth for the edges, but for the interior, it just relies on discrete convolutions, as on the real line.

2D Multiresolution Approximation As usual in the unit square $]0,1[^2$ we will define the approximation spaces by tensor product:

$$\mathcal{V}_j = V_j \otimes V_j = span < \varphi_{j,k}\otimes\varphi_{j,k'}\ ,\ k,k'=0,\ldots,2^j+1 > . \qquad (15)$$

The family $(\mathcal{V}_j)_{j\geq j_0}$ constitutes a MRA of $L^2(]0,1[^2)$. The two-dimensional biorthogonal projection on $\mathcal{V}_j$ will be denoted by $\mathcal{P}_j$. Two-dimensional wavelets are constructed (as usual) by tensor products of one-dimensional bases. The direct inequalities are still valid in dimension 2 ([15]). In particular, for all $u\in H^s(]0,1[^2)$, $1<s\leq n+1$ it holds:

$$\|u-\mathcal{P}_j(u)\|_{1,]0,1[^2} \lesssim 2^{-j(s-1)}\|u\|_{s,]0,1[^2} \qquad (16)$$

where $\|.\|_{s,\Omega}$ denotes the H^s-norm on Ω.

2.1.2 The biorthogonal wavelet multiplier space

Let now $V_j^0 \subset H_0^1(0,1)$ be the space of functions of V_j vanishing at the boundaries of the interval $]0,1[$. Following (8), we get:

$$V_j^0 = V_j \cap H_0^1(0,1) = span < \varphi_{j,k} \ , \ k = 1, \ldots, 2^j >,$$

$$V_j = V_j^0 \oplus span < \varphi_{j,0} \ , \ \varphi_{j,2^j+1} > .$$

In the Mortar Wavelet Method, this space V_j^0 plays the role of the space of traces on a "multiplier side" of discrete functions, vanishing at the extrema.

In order to build a suitable space M_δ to be used as a multiplier space for imposing weak continuity (see the definition of $\mathcal{X}_\delta$ (30) in the following), we construct a second multi-scale analysis $\tilde{M}_j$, biorthogonal to V_j^0. It is well known that the multiplier space M_δ needs to approximate well functions that take non zero values at the extrema of the subdomain edges (namely it needs to approximate the outer normal derivative of the solution) and, following the Mortar point of view, it needs to be biorthogonal to V_j^0 in order to treat each mortar independently. Moreover, we recall that the natural choice of the multiplier space on an edge is not necessarily a subset of the space of traces of the interior functions, as it is in the classical Mortar method; we will see that, under suitable assumptions which are easily verified in the wavelet case, this does not yield any major modification in the results that can be obtained, with no substantial difference with respect to the "classical" Mortar method (in this respect see also [30]). More precisely set

$$\begin{array}{rclr}
\tilde{\phi}_{j,k} & = & \tilde{\varphi}_{j,k} + c_k \ \tilde{\varphi}_{j,0} \ , & for \ k = 1, N \\
\tilde{\phi}_{j,k} & = & \tilde{\varphi}_{j,k} \ , & for \ k = N+1, 2^j - N \\
\tilde{\phi}_{j,k} & = & \tilde{\varphi}_{j,k} + d_k \ \tilde{\varphi}_{j,2^j+1} \ , & for \ k = 2^j - N + 1, 2^j
\end{array} \tag{17}$$

with

$$\begin{array}{rclcrcl}
c_k & = & -\frac{\alpha_k}{\alpha_0}, & k = 1, N-1 & c_N & = & \frac{1}{\alpha_0} \\
d_k & = & -\frac{\beta_k}{\beta_{2^j+1}}, & k = 2^j - N + 2, 2^j & d_{2^j-N+1} & = & \frac{1}{\beta_{2^j+1}}
\end{array} \tag{18}$$

where $(\alpha_k)_{k=0,\cdots,N-1}$ and $(\beta_k)_{k=2^j-N+2,\cdots,2^j+1}$ are respectively the solutions of the two following linear systems

$$\sum_{k=0}^{N-1} \tilde{a}_k^p \alpha_k = \tilde{a}_N^p, \quad \text{and} \qquad \sum_{k=2^j-N+2}^{2^j+1} \tilde{b}_k^p \beta_k = \tilde{b}_{2^j-N+1}^p, \quad \forall p = 0, N-1. \tag{19}$$

where $\tilde{a}^p_k$ and $\tilde{b}^p_k$ have been introduced in (4). Then we set

$$\tilde{M}_j = span < \tilde{\phi}_{j,k}, \quad k = 1, \ldots, 2^j > .$$

Remark again that, as requested, $\tilde{M}_j$ does not verify homogeneous boundary conditions; moreover this construction is, in fact, a particular case of the construction presented in [21].
The following theorem was proven in [11]:

Theorem 2.1 For all $j \geq j_0$ the set $\{\tilde{\phi}_{j,k}, k = 1, \cdots, 2^j\}$ is biorthogonal to the set $\{\varphi_{j,k}, k = 1, \cdots, 2^j\}$, that is it holds

$$\int_{]0,1[} \varphi_{j,k}(x)\tilde{\phi}_{j,k'}(x)\, dx = \delta_{kk'}. \tag{20}$$

Moreover the projector $\pi_j : L^2(0,1) \to V^0_j$ and its adjoint $\tilde{\pi}_j : L^2(0,1) \to \tilde{M}_j$ defined by

$$\pi_j f = \sum_{k=1}^{2^j} < f, \tilde{\phi}_{j,k} > \varphi_{j,k}, \tag{21}$$

and

$$\tilde{\pi}_j f = \sum_{k=1}^{2^j} < f, \varphi_{j,k} > \tilde{\phi}_{j,k}, \tag{22}$$

verify the following direct estimates: for all $u \in H^t_0(0,1)$,

$$\|u - \pi_j u\|_{H^s_0(]0,1[)} \lesssim 2^{-j(t-s)} \|u\|_{H^t_0(]0,1[)} \qquad \text{if } s \leq R, \qquad s < t \leq n+1$$

and for all $u \in H^t(0,1)$,

$$\|u - \tilde{\pi}_j u\|_{s,]0,1[} \lesssim 2^{-j(t-s)} \|u\|_{t,]0,1[} \qquad \text{if } s \leq \tilde{R}, \qquad s < t \leq \tilde{n}+1.$$

2.2 Non conforming domain decomposition

Let $\Omega \subset \mathbb{R}^2$ be a polygonal domain. We will consider the following model problem. Given $f \in L^2(\Omega)$, find $u : \Omega \longrightarrow \mathbb{R}$ such that

$$\begin{cases} -\Delta u = f, & \text{in } \Omega, \\ u = 0, & \text{on } \partial\Omega. \end{cases} \tag{23}$$

In order to solve such a problem by the mortar wavelet method, we will consider a decomposition of Ω as the union of L subdomains Ω_ℓ,

$$\Omega = \bigcup_{\ell=1,L} \Omega_\ell, \tag{24}$$

which we will assume to be rectangular and regular in shape, that is we will assume that there exist constants C_0, C_1 and C_2 such that, for all ℓ, Ω_ℓ contains a ball of diameter $C_0 H_\ell$, it is contained in a ball of diameter H_ℓ, and the length of each side is bounded from below by $C_0 H_\ell$. Moreover any interior angle ω satisfies $0 < C_1 < \omega < C_2 < \pi$.

We set

$$\Gamma_{\ell,\ell'} = \partial\Omega_\ell \cap \partial\Omega_{\ell'}, \tag{25}$$
$$S = \cup\Gamma_{\ell,\ell'}, \tag{26}$$

and we denote by $\gamma_\ell^{(i)}$ $(i = 1, \dots, 4)$ the i-th side of the ℓ-th domain:

$$\partial\Omega_\ell = \bigcup_{i=1}^{4} \gamma_\ell^{(i)}. \tag{27}$$

For simplicity we will assume here that the decomposition is *geometrically conforming*, that is each edge $\gamma_\ell^{(i)}$ coincides with $\Gamma_{\ell,\ell'} (= \partial\Omega_\ell \cap \partial\Omega_{\ell'})$ for some ℓ', $1 \leq \ell' \leq L$.

The space approximation is performed by introducing for each ℓ the affine and boundedly invertible transformation $F^\ell : [0,1]^2 \to \Omega_\ell$ mapping the reference square onto the subdomain Ω_ℓ:

$$\bar{\Omega}_\ell = F^\ell([0,1]^2).$$

Moreover we will denote by $F_i^\ell : [0,1] \to \gamma_\ell^{(i)}$ the restriction to the $i-$th side (which is identified with the interval $[0,1]$) of F^ℓ.

We then choose a discretization level $j_\ell \geq j_0$ and set

$$\mathcal{V}_\delta^\ell = F^\ell(\mathcal{V}_{j_\ell}) \cap H_0^1(\Omega)|_{\Omega_\ell}.$$

According to the mortar approach, in order to impose weak continuity, we start by splitting the skeleton as the disjoint union of a certain number of subdomain sides $\gamma_\ell^{(i)}$, usually called “non mortars” or “slave sides”. More precisely, we choose an index set $I \subset \{1, \dots, L\} \times \{1, \dots, 4\}$ such that,

$$S = \bigcup_{(\ell,i)\in I} \gamma_\ell^{(i)}, \qquad \begin{array}{c}(\ell_1, i_1), (\ell_2, i_2) \in I,\\ (\ell_1, i_1) \neq (\ell_2, i_2)\end{array} \Rightarrow \gamma_{\ell_1}^{(i_1)} \cap \gamma_{\ell_2}^{(i_2)} = \emptyset. \tag{28}$$

On each slave side γ_m (with the compact index notation $m = (\ell, i) \in I$) we define a multiplier space

$$M_\delta^m = F_i^\ell(\tilde{M}_{j_\ell}), \qquad (\ell, i) \in I,$$

and we look for a solution of our problem among the functions which satisfy across each "slave" side the jump constraint

$$\int_{\gamma_m} [u]\lambda_\delta = 0, \qquad \forall \lambda_\delta \in M_\delta^m, \tag{29}$$

where we recall that, for $\gamma_m = \partial\Omega_\ell \cap \partial\Omega_k$, $[u] = u^\ell - u^k$ denotes the jump across γ_m.

More precisely we set

$$\mathcal{X}_\delta = \left\{ v_\delta \in \prod_{\ell=1}^{L} \mathcal{V}_\delta^\ell, \forall m \in I, \forall \lambda_\delta \in M_\delta^m \int_{\gamma_m} [v_\delta]\lambda_\delta = 0 \right\}. \tag{30}$$

We remark that unlike the classical Mortar method, the spaces M_δ^m used for defining on each mortar side the space M_δ are not included in the trace spaces $\mathcal{V}_\delta^\ell|_{\gamma_\ell^{(i)}}$, but in the corresponding dual spaces.

We can now introduce the following discrete problem:

Problem 2.1 (PD) Find $u_\delta \in \mathcal{X}_\delta$, such that for all $v_\delta \in \mathcal{X}_\delta$,

$$\sum_{\ell=1}^{L} \int_{\Omega_\ell} \nabla u_\delta \nabla v_\delta = \int_\Omega f v_\delta. \tag{31}$$

It is possible to prove the following theorem [11]:

Theorem 2.2 There exists a unique solution of problem (PD), satisfying the following error estimate: if the solution u of problem (23) satisfies $u|_{\Omega_\ell} \in H^s(\Omega_\ell)$, $s > 1$ $\forall \ell = 1, \cdots, L$, then

$$\begin{aligned} \|u - u_\delta\|_{1,*} \lesssim \quad & \left(\textstyle\sum_{\ell=1}^{L} 2^{-2j_\ell(s-1)} \|u\|_{s,\Omega_\ell}^2 \right)^{1/2} \\ & + \left(\textstyle\sum_{\ell=1}^{L} 2^{-2j_\ell(s-1)} \left\| \frac{\partial u}{\partial \nu_\ell} \right\|_{s-3/2,\partial\Omega_\ell}^2 \right)^{1/2}, \end{aligned} \tag{32}$$

where $\|\cdot\|_{1,*}$ denotes the broken H^1 norm defined as

$$\|u\|_{1,*} = \left(\sum_{\ell=1}^{L} \|u\|_{1,\Omega_\ell}^2 \right)^{1/2},$$

and ν_ℓ is the outer unit normal to the subdomain Ω_ℓ.

3 Implementation

3.1 Numerical solution in the constrained space

It is well known that in the implementation of the mortar method, the multiplication by the stiffness matrix can be performed by applying subdomainwise the local stiffness matrix (which does not take into account the constraints), after multiplication by a "transfer matrix", which gives the values of the constrained degrees of freedom (the ones living on the interior of multiplier sides) in terms of the remaining (free) degrees of freedom, and which is a discrete realization of the projector π (see Section 2.1.2).

For each subdomain Ω_ℓ, we will denote by R^ℓ the local stiffness matrix relative to the discretization of the Laplace operator in $\mathcal{V}_\delta^\ell$: more precisely, setting for each $\mathbf{k} = (k_1, k_2) \in \{0, \cdots, 2^{j_\ell} + 1\}^2$

$$\begin{aligned}
\Phi_{j_\ell,\mathbf{k}}(x,y) &= \varphi_{j_\ell,k_1}(x)\varphi_{j_\ell,k_2}(y), \qquad (x,y) \in]0,1[^2, && (33)\\
\Phi^\ell_{j_\ell,\mathbf{k}}(x,y) &= \Phi_{j_\ell,k}((F^\ell)^{-1}(x,y)), \qquad (x,y) \in \Omega_\ell, && (34)
\end{aligned}$$

we write

$$R^\ell_{\mathbf{n},\mathbf{k}} = \int_{\Omega_\ell} \nabla\Phi^\ell_{j_\ell,\mathbf{k}} \nabla\Phi^\ell_{j_\ell,\mathbf{n}}.$$

An element u_δ of $\mathcal{X}_\delta$ has the form

$$u_\delta = (u_\delta^\ell)_{\ell=1,L}, \qquad \text{with} \qquad u_\delta^\ell = \sum_{\mathbf{k}} u^\ell_{\mathbf{k}}\, \Phi^\ell_{j_\ell,\mathbf{k}},$$

where the coefficients $(u^\ell_{\mathbf{k}})$ must satisfy the discrete equivalent of the jump constraint on the interface S.

The actual degrees of freedom are all the coefficients $u^\ell_{\mathbf{k}}$, $\mathbf{k} = (k_1, k_2) \in \{1, \cdots, 2^{j_\ell}\}^2$, corresponding to basis functions vanishing on $\partial\Omega_\ell$, and those coefficients $u^\ell_{\mathbf{k}}$, k_1 and/or k_2 being either 0 or $2^{j_\ell}+1$, corresponding to either a vertex of Ω_ℓ or to a "non mortar" side ($\gamma_\ell^{(i)}$, $(\ell, i) \notin I$, see (28)).The value of those coefficients $u^\ell_{\mathbf{k}}$ (k_1 or k_2 being either 0 or $2^{j_\ell} + 1$) corresponding to basis functions vanishing at the vertices of Ω_ℓ and "living" on mortar sides ($\gamma_\ell^{(i)}$, $(\ell, i) \in I$) is uniquely determined by the remaining coefficients through the jump condition. For the sake of simplicity, we will introduce the following notations: let

$$\mathbf{u} = \begin{pmatrix} \mathbf{u}^1 \\ \vdots \\ \mathbf{u}^L \end{pmatrix},$$

be a generic global coefficient vector, whose components $\mathbf{u}^\ell$, with $\ell = 1, \cdots, L$ ($\mathbf{u}^\ell = (u^\ell_{\mathbf{k}})_{\kappa \in \{0,\ldots,2^{j_\ell}+1\}^2}$), represent the vector of the local coordinates of u_δ relative to the ℓ-th subdomain. We will denote by $\mathbf{u}^M$ and

$\mathbf{u}^S$ the coefficient vectors corresponding to the degrees of freedom and to the constrained ones respectively. Therefore, by introducing P as a suitable permutation matrix allowing us to write $\mathbf{u}$ in the compact form

$$\mathbf{u} = P \begin{pmatrix} \mathbf{u}^M \\ \mathbf{u}^S \end{pmatrix}, \tag{35}$$

and denoting by C the matrix expressing the global constraint operator, such that

$$\mathbf{u}^S = C\mathbf{u}^M, \tag{36}$$

we will write

$$\mathbf{u} = P \begin{bmatrix} I \\ C \end{bmatrix} \mathbf{u}^M. \tag{37}$$

For the implementation, each function in $\mathcal{X}_\delta$ can be expressed through two different coefficient vectors. The first one is the vector $\mathbf{u}^M$ of coefficients corresponding to the actual degrees of freedom, the second is the redundant vector $\mathbf{u}$: in fact, the global stiffness matrix R corresponding to problem (31), acting on the vector $\mathbf{u}^M$ of degrees of freedom is, by definition, the matrix R such that, denoting by $\mathbf{v}^M$ and $\mathbf{u}^M$ the vectors of coefficients of two functions u_δ and v_δ in $\mathcal{X}_\delta$, it holds

$$(\mathbf{v}^M)^T R\ \mathbf{u}^M = a_X(u_\delta, v_\delta).$$

Therefore, since

$$a_X(u_\delta, v_\delta) = \sum_{\ell=1}^{L} \int_{\Omega_\ell} \nabla u_\delta^\ell \nabla v_\delta^\ell = \mathbf{v}^T \begin{pmatrix} R^1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & R^L \end{pmatrix} \mathbf{u},$$

using relation (37) we easily obtain

$$R = \begin{bmatrix} I & C^T \end{bmatrix} P^T \begin{pmatrix} R^1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & R^L \end{pmatrix} P \begin{bmatrix} I \\ C \end{bmatrix}.$$

Therefore, the application of the global stiffness matrix consists on three steps:

1. apply the constraint operator to the global coefficient vector $\mathbf{u}$ in order to obtain the constrained coefficients in function of the degrees of freedom

2. apply the local stiffness operator subdomain-wise

3. apply the transpose of the constraint operator.

Remark that steps **1** and **3** are the two operations that allows the different subdomains to interact with each other, while step **2** depends only on the corresponding subdomain.

3.2 The constraint operator in the wavelet context

Let us then give a closer look at the form that the constraint takes in the wavelet context .

We first split the function u as the sum of several contributions corresponding to the interior part, the vertices and the four edges respectively:

$$u_\delta^\ell = \sum_{\mathbf{k}\in I_0^\ell} u_\mathbf{k}^\ell \,\Phi_{j_\ell,\mathbf{k}}^\ell + \sum_{\mathbf{k}\in I_V^\ell} u_\mathbf{k}^\ell \,\Phi_{j_\ell,\mathbf{k}}^\ell + \sum_{i=1}^{4}\sum_{\mathbf{k}\in I_i^\ell} u_\mathbf{k}^\ell \,\Phi_{j_\ell,\mathbf{k}}^\ell, \tag{38}$$

with

$$\begin{aligned}
I_0^\ell &= \{\mathbf{k}=(k_1,k_2), \quad k_1 = 1,\dots,2^{j_\ell}, \quad k_2 = 1,\dots,2^{j_\ell}\},\\
I_V^\ell &= \{(0,0),(0,2^{j_\ell}+1),(2^{j_\ell}+1,0),(2^{j_\ell}+1,2^{j_\ell}+1)\},\\
I_1^\ell &= \{(k,0), \quad k=1,\dots,2^{j_\ell}\},\\
I_2^\ell &= \{(2^{j_\ell}+1,k), \quad k=1,\dots,2^{j_\ell}\},\\
I_3^\ell &= \{(k,2^{j_\ell}+1), \quad k=1,\dots,2^{j_\ell}\},\\
I_4^\ell &= \{(0,k), \quad k=1,\dots,2^{j_\ell}\}.
\end{aligned}$$

The last sum in (38) can be further split as the contribution of the master sides plus the contribution of the slave sides, yielding

$$u_\delta^\ell = \sum_{\mathbf{k}\in I_0^\ell} u_\mathbf{k}^\ell \,\Phi_{j_\ell,\mathbf{k}}^\ell + \sum_{\mathbf{k}\in I_V^\ell} u_\mathbf{k}^\ell \,\Phi_{j_\ell,\mathbf{k}}^\ell + \sum_{i:(\ell,i)\in I}\sum_{\mathbf{k}\in I_i^\ell} u_\mathbf{k}^\ell \,\Phi_{j_\ell,\mathbf{k}}^\ell + \sum_{i:(\ell,i)\notin I}\sum_{\mathbf{k}\in I_i^\ell} u_\mathbf{k}^\ell \,\Phi_{j_\ell,\mathbf{k}}^\ell.$$

It is clear that there is a natural correspondence between I_i^ℓ and the set $\{1,\dots,2^{j_\ell}\}$ and for a given $\mathbf{k}\in I_i^\ell$ we will denote by k the corresponding element of such a set, given by

$$u_k^{\ell,i} = u_{\mathbf{k}(k,i)}^\ell \alpha^{k,i} \qquad \text{with} \qquad \mathbf{k}(k,i) = \begin{cases} (k,0) & i=1\\ (2^{j_\ell}+1,k) & i=2\\ (k,2^{j_\ell}+1) & i=3\\ (0,k) & i=4 \end{cases}$$

$$\text{and} \quad \alpha^{k,i} = \begin{cases} \varphi_{j_\ell,0}(0) & i=1,4\\ \varphi_{j_\ell,2^{j_\ell}+1}(1) & i=2,3 \end{cases}$$

The actual degrees of freedom are the coefficients $u^\ell_{\mathbf{k}}$ with $k \in I^\ell_0 \cup I^\ell_V \cup_{i:(\ell,i)\notin I} I^\ell_i$, while the coefficients $u^\ell_{\mathbf{k}}$ with $k \in \cup_{i:(\ell,i)\in I} I^\ell_i$ are uniquely determined by the constraint. Let us then assume that $(\ell, i) \in I$ and that the corresponding master is (ℓ', i'), that is

$$\gamma^{(i)}_\ell = \Gamma_{\ell,\ell'} = \gamma^{(i')}_{\ell'}.$$

Observing that the trace operator reads

$$u^\ell_\delta|_{\Gamma_{\ell,\ell'}} = \sum_{k=0}^{2^{j_\ell}+1} u^{\ell,i}_k \ \varphi_{j_\ell,k}, \qquad u^{\ell'}_\delta|_{\Gamma_{\ell,\ell'}} = \sum_{k=0}^{2^{j_{\ell'}}+1} u^{\ell',i'}_k \ \varphi_{j_{\ell'},k},$$

the constraint becomes

$$\int_{\Gamma_{\ell,\ell'}} \left(\sum_{k=0}^{2^{j_\ell}+1} u^{\ell,i}_k \ \varphi_{j_\ell,k} - \sum_{k=0}^{2^{j_{\ell'}}+1} u^{\ell',i'}_k \ \varphi_{j_{\ell'},k} \right) \tilde{\phi}_{j_\ell,k'} = 0, \qquad k' = 1, \dots, 2^{j_\ell}.$$

Using the biorthogonality relation (20), and according to the definition (17) of $\tilde{\phi}_{j_\ell,k'}$, this can be rewritten as

$$\begin{aligned}
u^{\ell,i}_{k'} \quad &= \quad \int_{\Gamma_{\ell,\ell'}} \left(\sum_{k=0}^{2^{j_{\ell'}}+1} u^{\ell',i'}_k \varphi_{j_{\ell'},k} \right) \tilde{\phi}_{j_\ell,k'} - u^{\ell,i}_0 \int_{\Gamma_{\ell,\ell'}} \varphi_{j_\ell,0} \tilde{\phi}_{j_\ell,k'} \\
&- \quad u^{\ell,i}_{2^{j_\ell}+1} \int_{\Gamma_{\ell,\ell'}} \varphi_{j_\ell,2^{j_\ell}+1} \tilde{\phi}_{j_\ell,k'} \\
&= \quad \int_{\Gamma_{\ell,\ell'}} \left(\sum_{k=0}^{2^{j_{\ell'}}+1} u^{\ell',i'}_k \varphi_{j_{\ell'},k} \right) \tilde{\phi}_{j_\ell,k'} - u^{\ell,i}_0 c_{k'} - u^{\ell,i}_{2^{j_\ell}+1} d_{k'},
\end{aligned}$$

with $k' = 1, \dots, 2^{j_\ell}$. In the case that a geometrically conforming decomposition is considered and that all subdomains are discretized by wavelets, the application of the constraint (*i.e* the multiplication by the matrix C) will reduce to performing either a *Fast Wavelet Transform* or an *Inverse Fast Wavelet Transform* depending on which of the two discretizations on the "trace" and on the "multiplier" side is finer.

Indeed on one mortar side $\gamma^{(i)}_\ell = \Gamma_{\ell,\ell'}$, which for instance we can assume to be the lower side of Ω_ℓ and the upper side of $\Omega_{\ell'}$, the jump condition would imply

$$\begin{aligned}
\sum_{k=1}^{2^{j_\ell}+1} u^\ell_{(k,0)} \varphi_{j_\ell,k}(x) \quad &= \quad \pi_{j_\ell} \Big(\sum_{k=0}^{2^{j_{\ell'}+1}} u^{\ell'}_{(k,2^{j_{\ell'}}+1)} \varphi_{j_{\ell'},k}(x) \\
&\qquad - u^\ell_{(0,0)} \varphi_{j_\ell,0}(x) - u^\ell_{(2^{j_\ell}+1,0)} \varphi_{j_\ell,2^{j_\ell}+1}(x) \Big),
\end{aligned}$$

where $\pi_{j_\ell} = F_i^\ell \circ \pi_j \circ (F_i^\ell)^{-1}$ is the projection operator $\pi_{j_\ell} : L^2(\gamma_\ell^{(i)}) \longrightarrow V_{j_\ell}^0(\gamma_\ell^{(i)})$ obtained by (21).

The coefficients of the left hand side can be retrieved by means of a Fast Wavelet Transform (FWT) in $O(2^{\max\{j_\ell, j_{\ell'}\}})$ operations. In other words, multiplying $\mathbf{u}^M$ by C reduces to applying a sequence of FWTs on the mortar sides.

4 Numerical Results

We test the Mortar Wavelet method on a very simple model problem, namely the Poisson equation

$$-\Delta u = f \tag{39}$$

on a square domain, with homogeneous boundary conditions. The tests we show are referred to different choices of the function f and of the wavelet discretization spaces.

In particular we use biorthogonal B-spline compactly supported scaling functions, showing that, in agreement with the theory, the solution is correctly calculated by the method proposed though different levels of discretization are used in the different subdomains, and though no strong continuity is imposed at the interfaces. We study the behavior of the error with respect to the following norms :

- the relative L^2 norm $\frac{\|u-u_\delta\|_0}{\|u\|_0}$;
- the relative H^1 broken norm $\frac{\|u-u_\delta\|_{1,*}}{\|u\|_{1,*}}$;
- the relative L^∞ norm $\frac{\|u-u_\delta\|_\infty}{\|u\|_\infty}$.

Moreover, we evaluate the jump $[u_\delta]$ of the numerical solution.

We consider three different functions f, which are described in the following cases together with the exact analytical solution u:

<u>Case 1</u>:

$$\begin{cases} f(x,y) = -2\pi^2 \sin(\pi x)\sin(\pi y) \\ \\ u(x,y) = \sin(\pi x)\sin(\pi y). \end{cases} \tag{40}$$

Case 2: Given $a \in \mathbb{R}$,

$$
\begin{cases}
f(x,y) = -(4a^2x^2 - 4a^2x + a^2 - 2a)e^{-a(x-1/2)^2}\left[e^{-a(y-1/2)^2} - e^{-a/4}\right] \\
\qquad -(4a^2y^2 - 4a^2y + a^2 - 2a)e^{-a(y-1/2)^2}\left[e^{-a(x-1/2)^2} - e^{-a/4}\right] \\
\\
u(x,y) = \left[e^{-a(x-1/2)^2} - e^{-a/4}\right]\left[e^{-a(y-1/2)^2} - e^{-a/4}\right].
\end{cases}
\tag{41}
$$

Case 3: Given $a \in \mathbb{R}$,

$$
\begin{cases}
f(x,y) = \begin{cases}
\frac{1}{1-e^{-a/4}}\left[e^{-a(x-1/2)^2}\left(4a^2(x-1/2)^2 - 2a\right)y(y-1)\right. \\
\qquad \left. +2\left(e^{-a(x-1/2)^2} - e^{-a/4}\right)\right] & \text{if } x \leq 1/2, \\
-\pi^2\sin(\pi x)y(y-1) + 2\sin(\pi x) & \text{if } x > 1/2
\end{cases} \\
\\
u(x,y) = \begin{cases}
\frac{1}{1-e^{-a/4}}\left[e^{-a(x-1/2)^2} - e^{-a/4}\right]y(1-y) & \text{if } x \leq 1/2, \\
\sin(\pi x)y(1-y) & \text{if } x > 1/2
\end{cases}
\end{cases}
\tag{42}
$$

Remark 4.1 **Case 3** refers to a solution u which belongs to $C^\infty([0,.5) \times [0,1]) \times C^\infty(.5,1] \times [0,1])$, but whose second derivative with respect to the variable x is discontinuous. Figure 1 show the behavior of the exact solution, its gradient along the x axis and its Laplacian.

4.1 Wavelet basis: B-spline 2.2

As first test, we choose a B-spline 2.2 basis, that is a discretization space spanned by tent functions, for which the lower level parameter $j_0 = 3$. The following sections deal with different choices of the function f and several cases of conforming domain decomposition, the square unit domain being decomposed into rectangular subdomains, some of such cases treating also the presence of cross-points.

4.1.1 Case 1

We start by considering the right hand side of equation (39) given by (40). Tables 1 – 6 show the error for several choices of resolution in the master and in the slave edges, some referring to a decomposition of the original domain into two subdomains, and others to a domain decomposition with the presence of cross-points.
We remark that, whenever there is no presence of cross-points and the scale

parameter j_M of the master subdomain is lower or equal than the scale parameter j_S in the slave one, the solution is continuous along the interface, since the discretization on the mortar side is included in the discretization of the multiplier one (the resolution in the multiplier side is finer and an Inverse Fast Wavelet Transform has to be performed in order to compute the constraint operator). In Figure 2 we plot the L^2 norm and H^1 broken norm of the error in logarithmic scales together with the asymptotic rates $O(h)$ and $O(h^2)$. Remark that the error behaves as previewed by the theoretical estimate (Theorem 2.1). In particular the Mortar Wavelet method displays the same optimal convergence behavior as the Wavelet element method, without the need of imposing strong continuity of the details of the multilevel decomposition (whose implementation is, as already observed, extremely technical (see [13, 17, 22])).

4.1.2 Case 2

The numerical tests refer to the choice of the function f corresponding to Case 2. We choose the subdivision of the unit square into six rectangular subdomains (a 3×2 decomposition); the corresponding numerical behavior of the errors is shown in Table 7 for the value of the parameter a appearing in the exponential function (see the definition of f (41)) equals to 20. The level parameters j in each subdomain, given in anti-clockwise order, are

case (a1): $j = 3$ in all the subdomains

case (a2): $j = 4$ in all the subdomains

case (a3): $j = 5$ in all the subdomains

case (a4): $j_1 = 3$, $j_2 = 4$, $j_3 = 3$, $j_4 = 4$, $j_5 = 3$, $j_6 = 4$

case (a5): $j_1 = 4$, $j_2 = 3$, $j_3 = 5$, $j_4 = 3$, $j_5 = 4$, $j_6 = 5$

case (a6): $j_1 = 4$, $j_2 = 5$, $j_3 = 6$, $j_4 = 4$, $j_5 = 5$, $j_6 = 5$.

4.2 Wavelet basis: B-spline 3.3

In this section we show the numerical results for a different choice of the discretization spaces, considering a B-spline 3.3 basis; we recall that in such a case the minimum value of the level parameter is $j_0 = 4$ and that both the discretization and the multiplier spaces reproduce polynomials of degree 2. As we did in the previous section, we distinguish the different cases referred to the different choices of f.

4.2.1 Case 1

Tables 8 – 12 refer to the choice of the right hand side given by 40 and treat several cases of geometrical decompostion. Also in this case remark that the jump of the solution across the interfaces (see the column corresponding to the evaluation of the L^∞ norm of the jump $[u_\delta]$) is visible whenever cross-points are present in the decomposition or when the master discretization is finer (and therefore not included) than the slave one.

4.2.2 Case 2

The parameter of the exponential function is $a = 20$ and, as in the case where the B-spline 2.2 were considered, we test the method for a 3×2 D.D. We treat the following cases corresponding to different choices of level resolution in each subdomain:

case (a1): $j = 4$ in all the subdomains

case (a2): $j = 5$ in all the subdomains

case (a3): $j_1 = 4$, $j_2 = 5$, $j_3 = 4$, $j_4 = 5$, $j_5 = 4$, $j_6 = 5$

case (a4): $j_1 = 4$, $j_2 = 5$, $j_3 = 6$, $j_4 = 5$, $j_5 = 4$, $j_6 = 4$.

Table 13 shows the behavior of the errors.

4.2.3 Case 3

The function f considered in this examples corresponds to case 3 (see (42) for its analytical expression); the true solution has a jump in the Laplacian along the x axis (see Figure 1)). The tests we perform refer to Tables 14, 15 and 16; in the first case we consider the solution without decomposing the original domain and for increasing values of the level parameter. We compare the results with other two cases, the second being the case in which the domain is split into two subdomains, with an horizontal edge orthogonal to the discontinuity (a 1×2 D.D) and the third with a vertical edge along the discontinuity of u_{xx} respectively (a 2×1 D.D); in both cases the level j of the two subdomain is equal. As we can notice Tables 14 and 15 show the same behavior of the error, while Table 16 shows that the choice of decomposing the original domain into two subdomains with an interface along the irregularity of the solution gives better results. Table 17 refers to a 2×2 D.D with uniform resolution for each subdomain, and finally Table 18 shows the behavior of the numerical solution for a nine subdomains decomposition with alternating values of levels.

4.2.4 Coupling of wavelets and finite elements

As already briefly observed in the introduction, the wavelet domain decomposition method, be it conforming or non conforming, does not allow to (easily) treat general complex geometries. This is because not all geometries can be easily split as the union of a small number of (reasonably big) tensor product like domains. In the non conforming approach one can however resort to the coupling with finite elements. The idea is here to split the geometry in two regions: one which can be further split as the union of tensor product like sub-domains, to be treated via wavelets, and the other, containing perhaps complex geometrical details, to be discretized with the finite element method.

In order to demonstrate the feasibility of such an approach, we considered the following model problem. Denoting be $\check{\Omega}$

$$\check{\Omega} = \{(x, y) : \ 64(x - .25)^2 + 16(y - .5)^2 \leq 1\}$$

the ellipse centered in $(.25, .5)$ and with semi-axes $a = 1/8$ and $b = 1/4$, we set

$$\Omega =]0, 1[^2 \backslash \check{\Omega}.$$

On such a domain we consider Problem (39) with homogeneous Dirichlet boundary conditions and the right hand side f constructed in such a way that the function

$$u(x, y) = \sin \pi x \sin \pi y \left[4(x - .25)^2 + (y - .5)^2 - .0625\right]$$

is the true solution.

In order to apply the Mortar domain decomposition method we split Ω as $\Omega = \Omega_1 \cap \Omega_2$ with

$$\Omega_1 =]0, 1/2[\times]0, 1[\backslash\check{\Omega}, \qquad \Omega_2 =]1/2, 1[\times]0, 1[.$$

We choose to discretize the solution in Ω_1 by the finite element method, and in Ω_2 (which is rectangular) by means of wavelet.

The main difficulty of the Wavelet/FEM coupling is the computation of the integrals appearing in the jump constraint (29). In particular, there is in general no good classical quadrature rule allowing to compute the integral of the product of a wavelet type function times a piecewise polynomial on a generic grid. Such a problem has been faced and it has been given a satisfactory solution in [9]. It is however far from the scope of this paper to give further details on such an issue and we address the reader to such a paper. We only want to give here an example of the feasibility of the Wavelet/FEM coupling approach in order to treat complex domains, so we will consider a very particular case in which the constraint can be computed exactly and we will choose finite element grids whose trace on the interface is dyadic (that is uniform with mesh-size $h = 2^{-j}$ for some positive integer j).

In principle any of the two subdomains can be chosen as slave side. Optimal error estimates are available independently of such a choice (see [11] for an abstract error estimate which includes both wavelets and finite elements as particular cases). From an implementation point choosing the wavelet side as slave side has some small advantage over choosing the finite element side and using the classical multipliers for the finite element space, as proposed by Bernardi, Maday and Patera. In fact, in the wavelet case, the matrix to be inverted in order to compute the mortar projection is diagonal (while it is tridiagonal in the finite element case). We remark however that in the recent approach introduced by B. Wohlmuth in [30] suitable dual Lagrange multipliers are introduced in such a way that such a matrix is diagonal also in the finite element case, and the two choice become equivalent. In what follows we choose Ω_1 to be the master side and Ω_2 as slave side. For all tests we used 2.2 B-splines wavelets and $\mathbb{P}_1$ finite elements. As already observed it is far from the aim of this paper to fully explore the behavior of the Wavelet/FEM coupling (of which we only want to illustrate the feasibility) so we only consider a bunch of cases. In particular in Table 19 we show the error behavior with respect to the usual norms and for some choices of the approximation spaces. The parameter j indicates the level of the wavelet decomposition in the slave subdomain and the parameter h indicates the uniform meshsize on the interface of the geometrical decomposition corresponding to the trace of an unstructured finite element decomposition in the master subdomain. In Figures 3 and 4 we show the behavior of the numerical solution corresponding to other cases of approximations.

Acknowledgements

Work partially supported by the EC-IHP Network HPRN CT2002 00286 *"Breaking complexity"*, and by the *Région Rhône Alpes* .

5 Annex: Practical computations

Practical Construction of filters in the case of orthonormal basis.

In this section we describe how to construct practically the wavelet basis on the interval, from a classical MRA of $L^2(R)$. We restrict here our explanation to the orthonormal case and Daubechies wavelets.

5.1 Starting from the construction of Cohen-Daubechies-Vial

We start from a usual orthonormal MRA of $L^2(R)$ and its associated scaling function φ and wavelet ψ, compactly supported, with N vanishing moments and minimal support $[-N+1, N]$ [23]. Let (h_k) and (g_k) be the filter coefficients such that:

$$\varphi(x) = \sqrt{2} \sum_{k=-N+1}^{N} h_k \varphi(2x-k), \psi(x) = \sqrt{2} \sum_{k=-N+1}^{N} g_k \varphi(2x-k). \qquad (43)$$

and a particular choice of g_k is: $g_k = (-1)^k h_{1-k}$.

5.1.1 Computation of scaling function moments

Polynomials up to degree $N-1$, are locally reconstructed in the MRA and we can write:

$$\text{For } \ell = 0, \ldots, N-1, \qquad \frac{x^\ell}{\ell!} = \sum_{k=-\infty}^{+\infty} P_\ell(k)\ \varphi(x-k) \qquad (44)$$

By orthogonality,

$$P_\ell(k) =< \frac{x^\ell}{\ell!} \mid \varphi(x-k) > \qquad (45)$$

is the ℓ^{th} moment of the scaling function $\varphi(x-k)$. Following [16], P_ℓ is a polynomials of degree ℓ defined by:

$$P_\ell(X) = \sum_{n=0}^{\ell} \frac{C_{\ell-n}}{n!} X^n, \qquad (46)$$

where:

$$C_m = \int_{-\infty}^{+\infty} \frac{x^m}{m!} \varphi(x) dx = P_m(0). \qquad (47)$$

and the coefficients C_ℓ can be computed recursively using:

$$\begin{cases} C_0 & = & 1 \\ C_\ell & = & \frac{1}{2^\ell - 1} \sum_{r=1}^{\ell} M_r C_{\ell-r} \end{cases}$$

with

$$M_r = \frac{1}{\sqrt{2}} \sum_{m=-N+1}^{N} h_m \frac{m^r}{r!}.$$

5.1.2 The edge scaling functions

The left-edge scaling functions are defined to remain polynomials up to degree $N-1$ to remain in the MRA V_j of $L^2(0,1)$. For instance we can define [28]:

Definition. *For $\ell = 0, \ldots, N-1$, the left-edge scaling functions are defined by:*

$$\varphi_\ell^{left}(x) = \sum_{k=-N+1}^{N-2} P_\ell(k)\ \varphi(x-k)\ \chi_{[0,+\infty[}(x)\ . \tag{48}$$

Functions φ_ℓ^{left} are such that for all x in $[0,+\infty[$:

$$\frac{x^\ell}{\ell!} = \varphi_\ell^{left}(x) + \sum_{k=N-1}^{+\infty} P_\ell(k)\ \varphi(x-k)$$

where the functions $\varphi(.-k)$ for $k \geq N-1$ are all supported on $[0,+\infty[$.

The left-edge scaling functions verify a two-scale equation:

Proposition 5.1 There exists a matrix b of size $N \times (2N-2)$ such that, writing $D = (d_{ij})_{1\leq i,j\leq N}$ the diagonal matrix $d_{ij} = \frac{\delta_{i-j}}{2^{i-1}}$,

$$\begin{pmatrix} \varphi_0^{left} \\ \vdots \\ \varphi_{N-1}^{left} \end{pmatrix} (\frac{x}{2}) = D \begin{pmatrix} \varphi_0^{left} \\ \vdots \\ \varphi_{N-1}^{left} \end{pmatrix} (x) + b \begin{pmatrix} \varphi_{N-1} \\ \vdots \\ \varphi_{3N-4} \end{pmatrix} (x). \tag{49}$$

with the notation $\varphi_k(x) = \varphi(x-k)$. Moreover, the general term of the matrix b is:

$$b_{i+1,j-N+2} = \frac{P_i(j)}{2^i} - \sqrt{2} \sum_{m=N-1}^{\lfloor \frac{j+N-1}{2} \rfloor} P_i(m) h_{j-2m} \tag{50}$$

for $i = 0, \ldots, N-1$ and $j = N-1, \ldots, 3N-4$, $\lfloor x \rfloor$ being the integer part of x.

The right-edge scaling functions on $[0,1]$ are constructed by considering the operator T, defined by:

$$\forall f \in L^2(R), \quad Tf(x) = f(1-x). \tag{51}$$

We see that *support* $T\varphi = [-N+1, N]$, the same as φ. $T\varphi$ satisfies the two-scale relationship:

$$T\varphi(x) = \sqrt{2} \sum_{k=-N+1}^{N} \check{h}_k \; T\varphi(2x-k), \tag{52}$$

where $\check{h}_k = h_{1-k}$ for all k in $\mathbb{Z}$.

The right-edge scaling functions are then defined by, for all $\ell = 0, N-1$:

$$\varphi_\ell^{right} = T(T\varphi)_\ell^{left}.$$

Functions φ_ℓ^{right} are such that for all x in $]-\infty, 1]$:

$$\frac{(1-x)^\ell}{\ell!} = \sum_{k=-\infty}^{1-N} \check{P}_\ell(-k) \; \varphi(x-k) + \varphi_\ell^{right}(x)$$

where the functions $\varphi(.-k)$ for $k \leq 1-N$ are all supported on $]-\infty, 1]$.

By orthogonality,

$$\check{P}_\ell(k) = \langle \; \frac{(1-x)^\ell}{\ell!} \mid \varphi(x+k) \; \rangle = \langle \; \frac{x^\ell}{\ell!} \mid T\varphi(x-k) \; \rangle \tag{53}$$

is the ℓ^{th} moment of the scaling function $T\varphi(x-k)$, obtained in the same way as P_ℓ, but using coefficients $\check{h}_k$ instead of h_k.

The right-edge wavelets are defined in the same way and we have, for $\ell = 0, N-2$:

$$\psi_\ell^{right} = T(T\psi)_\ell^{left}.$$

5.2 The orthonormalization procedure

In each space V_j or W_j we dispose of a Riesz basis of scaling or wavelet function that we have to orthogonalize or biorthogonalize. As interior functions are already orthogonal and as edge functions are orthogonal to the interior ones, we just have to (bi-)orthogonalize the (left and right-) edge functions. In any case, the orthonormalization, or more generally the biorthogonalization of basis functions requires the computation of the Gram matrix of the family:

$$G^{left} = [< \varphi_k^{left}; \varphi_\ell^{left} >]_{0 \leq k, \ell \leq N-1}$$

Proposition 5.2 The Gram matrix G^{left} of the left-edge scaling functions φ_ℓ^{left} for $\ell = 0, \ldots, N-1$ is computed by:

$$2 \; G^{left} = DG^{left}D + bb^t, \tag{54}$$

where the matrices D and b have been introduced in (49). It goes similarly for the Gram matrix G^{right}.

Our orthonormalization procedure uses a Gram-Schmidt algorithm, in a reverse order, as it preserves the number of vanishing derivatives at the edges.

Proposition 5.3 Let us define

$$\begin{pmatrix} \Phi_0 \\ \vdots \\ \Phi_{N-1} \end{pmatrix} = \mathcal{B} \begin{pmatrix} \varphi_0^{left} \\ \vdots \\ \varphi_{N-1}^{left} \end{pmatrix} \tag{55}$$

where $(G^{left})^{-1} = \mathcal{B}^T\mathcal{B}$ is some Cholesky decomposition of the inverse of the Gram matrix, imposing that $\mathcal{B}$ is triangular superior.
The family of scaling functions $(\Phi_\ell)_{\ell=0,N-1}$ is orthonormal and satisfies the two-scale equation:

$$\frac{1}{\sqrt{2}} \begin{pmatrix} \Phi_0 \\ \vdots \\ \Phi_{N-1} \end{pmatrix} (x/2) = H_0 \begin{pmatrix} \Phi_0 \\ \vdots \\ \Phi_{N-1} \end{pmatrix} (x) + h_0 \begin{pmatrix} \varphi_{N-1} \\ \vdots \\ \varphi_{3N-4} \end{pmatrix} (x) \tag{56}$$

where H_0 and h_0 are the $N \times N$ and $N \times (2N-1)$ matrices:

$$H_0 = \frac{1}{\sqrt{2}}\ \mathcal{B}\ D\ \mathcal{B}^{-1} \quad \text{and} \quad h_0 = \frac{1}{\sqrt{2}}\ \mathcal{B}\ b.$$

It goes similarly for the computation of the filters associated to edge wavelets [28].
Remark: An alternative choice for the orthonormalization procedure could be to take $\mathcal{B} = (G^{left})^{-\frac{1}{2}}$ and corresponds to the Gram procedure which is used in [28], and preserves the translation invariance of functions (whenever it exists).

Another possibility is to biorthogonalize the edge functions and to introduce the dual functions:

$$\tilde{\varphi}_k^{left} = \sum_{\ell=0}^{N-1} (G^{left})^{-T}_{k,\ell}\ \varphi_\ell^{left}$$

References

[1] G. Abdoulaev, Y. Achdou, Y. Kutznetsov, and C. Prud'homme. On the parallel inplementation of the mortar element method. *M2AN Math. Model. Numer. Anal.*, 33(2):245–259, 1999.

[2] Y. Achdou and O. Pironneau. A fast solver for Navier-Stokes equations in the laminar regime using mortar finite element and boundary element method. *SIAM J. Numer. Anal.*, 32(4):985–1016, 1995.

[3] G. Anagnostou, Y. Maday, and A.T. Patera. A sliding mesh method for partial differential equations in nonstationary geometries: application to the incompressible Navier–Stokes equations. Technical Report 91024, Laboratoire d'Analyse Numérique, 1991.

[4] L. Anderson, N. Hall, B. Jawerth, and G. Peters. Wavelets on closed subsets on the real line. In L.L. Schumaker and G. Webb eds, editors, *Topics in the Theory and Applications of Wavelets*, pages 1–61, Boston, MA, 1993. Academic Press.

[5] F. Ben Belgacem. The mortar element method with Lagrange multiplier. *Numer. Math.*, 84(2):173–197, 1999.

[6] F. Ben Belgacem, A. Buffa, and Y. Maday. The mortar element method for 3d Maxwell's equations: First results. *SIAM J. Numer. Anal.*, 39(3):880–901, 2001.

[7] Z. Belhachmi and C. Bernardi. The mortar spectral element method for fourth-order problems. *Comp. Methods in Applied Mech. and Eng.*, 116, 1994.

[8] S. Bertoluzza. An adaptive wavelet collocation method based on interpolating wavelets. In *Multiscale Wavelet Methods for Partial Differential Equations*, volume 6 of *Wavelets Analysis and its Applications*, pages 109–135. Academic Press, 1997.

[9] S. Bertoluzza, S. Falletta, and V. Perrier. Wavelet/fem coupling by the mortar method. In *Lecture Notes in Computational Science and Engineering*, pages 119 – 132, 2002.

[10] S. Bertoluzza, S. Mazet, and M. Verani. A nonlinear Richardson algorithm for the solution of elliptic PDE's. *Math. Models Methods Appl. Sci.*, 2:143–158, 2003.

[11] S. Bertoluzza and V. Perrier. The mortar method in the wavelet context. *ESAIM:M2AN*, 4:647–673, 2001.

[12] S. Bertoluzza and P. Pietra. Space frequency adaptive approximation for quantum hydrodynamic models. *Transport Theory and Stat. Phys.*, 28:375–395, 2000.

[13] C. Canuto, A. Tabacco, and K. Urban. The wavelet element method. II. Realization and additional features in 2D and 3D. *Appl. Comput. Harmon. Anal.*, 8(2):123–165, 2000.

[14] P. Charton and V. Perrier. A pseudo-wavelet scheme for the two-dimensional Navier-Stokes equation. *Comp. Appl. Math.*, 15:139–160, 1996.

[15] A. Cohen. Numerical analysis of wavelet methods. In P.G. Ciarlet and J.L. Lions, editors, *Handbook in Numerical Analysis*, volume VII. Elsevier Science Publishers, North Holland, 2000.

[16] A. Cohen, I. Daubechies, and P. Vial. Wavelets on the interval and fast wavelet transforms. *ACHA*, 1:54–81, 1993.

[17] A. Cohen and R. Masson. Wavelet adaptive method for second order elliptic problems: boundary conditions and domain decomposition. *Numer. Math.*, 86(2):193–238, 2000.

[18] S. Dahlke, W. Dahmen ans R. Hochmut, and R. Schneider. Stable multiscale bases and local error estimation for elliptic problems. *Appl. Numer. Math.*, 23:21–48, 1997.

[19] W. Dahmen and A. Kunoth. Multilevel preconditioning. *Numer. Math.*, 63:315–344, 1992.

[20] W. Dahmen, A. Kunoth, and K. Urban. Biorthogonal spline-wavelets on the interval – stability and moment condition. *ACHA*, 6:132–196, 1999.

[21] W. Dahmen and R. Schneider. Wavelets with complementary boundary conditions – Function spaces on the cube. *Results in Mathematics*, 34:255–293, 1998.

[22] W. Dahmen and R. Schneider. Wavelets on manifolds. I. Construction and domain decomposition. *SIAM J. Math. Anal.*, 31(1):184–230, 1999.

[23] I. Daubechies. Ten lectures on wavelets. In *CBMS Lecture Notes*, volume 61. SIAM, Philadelphia, 1992.

[24] S. Jaffard. Wavelet methods for fast resolution of elliptic problems. *SIAM J. Numer. Anal*, 29:965–986, 1992.

[25] Y. Maday, V. Perrier, and J.C. Ravel. Adaptivité dynamique sur bases d'ondelettes pour l'approximation d'équations aux derivées partielles. *C. R. Acad. Sci Paris*, 312, (Série I):405–410, 1991.

[26] R. Masson. Biorthogonal spline wavelets on the interval for the resolution of boundary problems. M^3AS, 6(6), 1996.

[27] Y. Meyer. *Ondelettes et Opérateurs*. Hermann, 1990.

[28] P. Monasse and V. Perrier. Orthonormal wavelet bases adapted for partial differential equations with boundary conditions. *SIAM J. Math. Anal.*, 29:1040–1065, 1998.

[29] C. Prud'homme. A strategy for the resolution of the tridimensionnal incompressible Navier-Stokes equations. In *Méthodes itératives de décomposition de domaines et communications en calcul parallèle*, volume 10 of *Calculateurs Parallèles Réséaux et Systèmes répartis*, pages 371–380. Hermes, October 1998.

[30] B. Wohlmuth. A mortar finite element method using dual spaces for the Lagrange multiplier. *SIAM J. Numer. Anal.*, 38:989–1012, 2000.

Table 1: Two subdomains domain decomposition. The value of the level j in the first column represents the resolution parameter of both subdomains.

level	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_0}{\|\mathbf{u}\|_0}$	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_{1,*}}{\|\mathbf{u}\|_{1,*}}$	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_\infty}{\|\mathbf{u}\|_\infty}$	$\|\,[\mathbf{u}_\delta]\,\|_\infty$
$j=3$	$1.002e-02$	$8.764e-02$	$8.043e-03$	$4.441e-16$
$j=4$	$2.501e-03$	$4.274e-02$	$2.006e-03$	$4.441e-16$
$j=5$	$6.251e-04$	$2.031e-02$	$5.021e-04$	$4.441e-16$
$j=6$	$1.571e-04$	$9.081e-03$	$1.255e-04$	$2.221e-16$
$j=7$	$3.944e-05$	$3.432e-03$	$3.221e-05$	$4.441e-16$

Table 2: Two subdomains domain decomposition. j represents the value of the level parameter j_M in the master subdomain, while in the slave one the level $j_S = j_M + 1$.

level	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_0}{\|\mathbf{u}\|_0}$	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_{1,*}}{\|\mathbf{u}\|_{1,*}}$	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_\infty}{\|\mathbf{u}\|_\infty}$	$\| [\mathbf{u}_\delta] \|_\infty$
$j = 3$	$7.088e-03$	$7.035e-02$	$9.593e-03$	$6.661e-16$
$j = 4$	$1.742e-03$	$3.381e-02$	$2.427e-03$	$4.441e-16$
$j = 5$	$4.291e-04$	$1.582e-02$	$6.159e-04$	$4.441e-16$
$j = 6$	$1.083e-04$	$6.892e-03$	$1.536e-04$	$3.331e-16$

Table 3: Two subdomains domain decomposition. j represents the value of the level parameter j_M in the master subdomain, while in the slave one the level $j_S = j_M - 1$.

level	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_0}{\|\mathbf{u}\|_0}$	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_{1,*}}{\|\mathbf{u}\|_{1,*}}$	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_\infty}{\|\mathbf{u}\|_\infty}$	$\| [\mathbf{u}_\delta] \|_\infty$
$j = 4$	$6.896e-03$	$6.897e-02$	$9.746e-03$	$9.797e-03$
$j = 5$	$1.718e-03$	$3.347e-02$	$3.347e-02$	$2.419e-03$
$j = 6$	$4.296e-04$	$1.573e-02$	$6.147e-04$	$6.031e-04$
$j = 7$	$1.062e-04$	$6.867e-03$	$1.567e-04$	$1.506e-04$

Table 4: 2×2 domain decomposition. The level j chosen equal in all the subdomains.

level	$\frac{\Vert\mathbf{u}-\mathbf{u}_\delta\Vert_0}{\Vert\mathbf{u}\Vert_0}$	$\frac{\Vert\mathbf{u}-\mathbf{u}_\delta\Vert_{1,*}}{\Vert\mathbf{u}\Vert_{1,*}}$	$\frac{\Vert\mathbf{u}-\mathbf{u}_\delta\Vert_\infty}{\Vert\mathbf{u}\Vert_\infty}$	$\Vert\,[\mathbf{u}_\delta]\,\Vert_\infty$
$j=3$	$3.801e-03$	$5.551e-02$	$3.218e-03$	$6.349e-05$
$j=4$	$9.503e-04$	$2.709e-02$	$8.035e-04$	$2.431e-06$
$j=5$	$2.376e-04$	$1.288e-02$	$2.007e-04$	$3.743e-07$
$j=6$	$5.798e-05$	$5.765e-03$	$5.195e-05$	$1.305e-08$
$j=7$	$1.474e-05$	$2.186e-03$	$1.279e-05$	$8.047e-09$

Table 5: 3×3 domain decomposition. The level j in chosen equal in all the subdomains.

level	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_0}{\|\mathbf{u}\|_0}$	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_{1,*}}{\|\mathbf{u}\|_{1,*}}$	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_\infty}{\|\mathbf{u}\|_\infty}$	$\|\,[\mathbf{u}_\delta]\,\|_\infty$
$j=3$	$1.69e-03$	$3.725e-02$	$3.016e-03$	$4.287e-03$
$j=4$	$4.226e-04$	$1.809e-02$	$7.537e-04$	$1.023e-03$
$j=5$	$1.056e-04$	$8.591e-03$	$1.885e-04$	$2.494e-04$
$j=6$	$2.572e-05$	$3.844e-03$	$4.769e-05$	$6.153e-05$

Table 6: 3×3 domain decomposition. The first column represents the minimum value of the level parameter, which is alternated with $j+1$ in the other subdomains.

level	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_0}{\|\mathbf{u}\|_0}$	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_{1,*}}{\|\mathbf{u}\|_{1,*}}$	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_\infty}{\|\mathbf{u}\|_\infty}$	$\|\,[\mathbf{u}_\delta]\,\|_\infty$
$j=3$	$1.253e-03$	$2.992e-02$	$2.726e-03$	$3.013e-03$
$j=4$	$3.142e-04$	$1.443e-02$	$6.835e-04$	$6.924e-04$
$j=5$	$7.802e-05$	$6.773e-03$	$1.718e-04$	$1.687e-04$

Table 7: A 3×2 D.D. with different values of the level parameter j in each subdomain.

case	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_0}{\|\mathbf{u}\|_0}$	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_{1,*}}{\|\mathbf{u}\|_{1,*}}$	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_\infty}{\|\mathbf{u}\|_\infty}$	$\|\,[\mathbf{u}_\delta]\,\|_\infty$
$(a1)$	$1.216e-02$	$1.169e-01$	$9.819e-03$	$1.679e-02$
$(a2)$	$3.048e-03$	$5.687e-02$	$1.957e-03$	$2.256e-03$
$(a3)$	$7.625e-04$	$2.704e-02$	$4.889e-04$	$2.873e-04$
$(a4)$	$8.739e-03$	$9.299e-02$	$7.556e-03$	$1.542e-02$
$(a5)$	$8.236e-03$	$8.856e-02$	$9.142e-03$	$1.548e-02$
$(a6)$	$1.071e-03$	$3.069e-02$	$1.406e-03$	$2.815e-03$

Table 8: Two subdomains domain decomposition. The value of the level j in the first column represents the resolution parameter of both subdomains.

level	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_0}{\|\mathbf{u}\|_0}$	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_{1,*}}{\|\mathbf{u}\|_{1,*}}$	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_\infty}{\|\mathbf{u}\|_\infty}$	$\|\,[\mathbf{u}_\delta]\,\|_\infty$
$j=4$	$1.208e-03$	$8.697e-03$	$1.875e-03$	$2.942e-15$
$j=5$	$1.118e-04$	$1.595e-03$	$2.412e-04$	$2.276e-15$
$j=6$	$1.001e-05$	$2.877e-04$	$3.025e-05$	$1.776e-15$
$j=7$	$8.891e-07$	$6.821e-05$	$3.759e-06$	$8.466e-16$

Table 9: Two subdomains domain decomposition. j represents the value of the level parameter j_M in the master subdomain, while in the slave one the level $j_S = j_M + 1$.

level	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_0}{\|\mathbf{u}\|_0}$	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_{1,*}}{\|\mathbf{u}\|_{1,*}}$	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_\infty}{\|\mathbf{u}\|_\infty}$	$\| [\mathbf{u}_\delta] \|_\infty$
$j = 4$	$8.776e-04$	$6.589e-03$	$1.859e-03$	$2.665e-15$
$j = 5$	$8.017e-05$	$1.178e-03$	$2.422e-04$	$3.053e-15$
$j = 6$	$7.141e-06$	$2.118e-04$	$3.049e-05$	$1.749e-15$

Table 10: Two subdomains domain decomposition. j represents the value of the level parameter j_M in the master subdomain, while in the slave one the level $j_S = j_M - 1$.

level	$\frac{\Vert \mathbf{u}-\mathbf{u}_\delta \Vert_0}{\Vert \mathbf{u} \Vert_0}$	$\frac{\Vert \mathbf{u}-\mathbf{u}_\delta \Vert_{1,*}}{\Vert \mathbf{u} \Vert_{1,*}}$	$\frac{\Vert \mathbf{u}-\mathbf{u}_\delta \Vert_\infty}{\Vert \mathbf{u} \Vert_\infty}$	$\Vert [\mathbf{u}_\delta] \Vert_\infty$
$j = 5$	$8.559e-04$	$6.282e-03$	$1.781e-03$	$1.939e-03$
$j = 6$	$7.917e-05$	$1.149e-03$	$2.346e-04$	$2.528e-04$
$j = 7$	$7.096e-06$	$2.093e-04$	$2.992e-05$	$3.193e-05$

Table 11: 2×2 domain decomposition. The level j chosen equal in all the subdomains.

level	$\frac{\Vert \mathbf{u}-\mathbf{u}_\delta \Vert_0}{\Vert \mathbf{u} \Vert_0}$	$\frac{\Vert \mathbf{u}-\mathbf{u}_\delta \Vert_{1,*}}{\Vert \mathbf{u} \Vert_{1,*}}$	$\frac{\Vert \mathbf{u}-\mathbf{u}_\delta \Vert_\infty}{\Vert \mathbf{u} \Vert_\infty}$	$\Vert\, [\mathbf{u}_\delta]\, \Vert_\infty$
$j = 4$	$1.594e-04$	$2.253e-03$	$2.413e-04$	$2.548e-10$
$j = 5$	$1.415e-05$	$4.078e-04$	$3.035e-05$	$4.277e-10$
$j = 6$	$1.254e-06$	$9.883e-05$	$3.787e-06$	$1.377e-10$
$j = 7$	$1.113e-07$	$2.396e-05$	$4.701e-07$	$2.357e-11$

Table 12: 3×3 domain decomposition. The first column represents the minimum value of the level parameter, which is alternated with $j+1$ in the other subdomains.

level	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_0}{\|\mathbf{u}\|_0}$	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_{1,*}}{\|\mathbf{u}\|_{1,*}}$	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_\infty}{\|\mathbf{u}\|_\infty}$	$\| [\mathbf{u}_\delta] \|_\infty$
$j=4$	$3.132e-05$	$6.988e-04$	$1.653e-04$	$2.337e-04$
$j=5$	$2.764e-06$	$1.293e-04$	$2.132e-05$	$2.861e-05$

Table 13: A 3×2 D.D. Behavior of the errors for the choice of the level parameters corresponding to cases (a1) – (a4).

Fig.	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_0}{\|\mathbf{u}\|_0}$	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_{1,*}}{\|\mathbf{u}\|_{1,*}}$	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_\infty}{\|\mathbf{u}\|_\infty}$	$\|\, [\mathbf{u}_\delta] \,\|_\infty$
$(a1)$	$1.2566e-03$	$1.371e-02$	$3.089e-03$	$6.089e-03$
$(a2)$	$7.686e-05$	$1.905e-03$	$4.671e-04$	$9.216e-04$
$(a3)$	$8.788e-04$	$9.849e-03$	$2.643e-03$	$4.236e-03$
$(a4)$	$3.622e-04$	$4.804e-03$	$1.769e-03$	$3.321e-03$

Table 14: Behavior of the errors on the whole domain. No decomposition of the unit square has been considered.

level	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_0}{\|\mathbf{u}\|_0}$	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_{1,*}}{\|\mathbf{u}\|_{1,*}}$	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_\infty}{\|\mathbf{u}\|_\infty}$	$\|\,[\mathbf{u}_\delta]\,\|_\infty$
$j=4$	$2.953e-03$	$1.993e-02$	$5.497e-03$	$-$
$j=5$	$2.211e-04$	$3.037e-03$	$5.338e-04$	$-$
$j=6$	$7.438e-05$	$5.687e-04$	$8.991e-05$	$-$

Table 15: Decomposition of the unit square into two subdomains. In particular it is a 1×2 D.D, where the interface is orthogonal to the discontinuity of the Laplacian of the exact solution. The level j is chosen equal in both the subdomains.

level	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_0}{\|\mathbf{u}\|_0}$	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_{1,*}}{\|\mathbf{u}\|_{1,*}}$	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_\infty}{\|\mathbf{u}\|_\infty}$	$\| [\mathbf{u}_\delta] \|_\infty$
$j=4$	$2.953e-03$	$1.993e-02$	$5.495e-03$	$7.355e-16$
$j=5$	$2.215e-04$	$3.035e-03$	$5.356e-04$	$5.689e-16$
$j=6$	$7.513e-05$	$5.628e-04$	$9.144e-05$	$2.914e-16$

Table 16: Decomposition of the unit square into two subdomains. It is a 2×1 D.D, with an interface along the discontinuity of the Laplacian of the exact solution. The level j is chosen equal in both the subdomains.

level	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_0}{\|\mathbf{u}\|_0}$	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_{1,*}}{\|\mathbf{u}\|_{1,*}}$	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_\infty}{\|\mathbf{u}\|_\infty}$	$\| [\mathbf{u}_\delta] \|_\infty$
$j=4$	$6.486e-04$	$8.393e-03$	$1.823e-03$	$8.466e-16$
$j=5$	$4.951e-05$	$9.566e-04$	$1.671e-04$	$6.801e-16$
$j=6$	$3.673e-05$	$2.298e-04$	$5.193e-05$	$3.539e-16$

Table 17: 2×2 D.D. The level j in equal in all the subdomains.

level	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_0}{\|\mathbf{u}\|_0}$	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_{1,*}}{\|\mathbf{u}\|_{1,*}}$	$\frac{\|\mathbf{u}-\mathbf{u}_\delta\|_\infty}{\|\mathbf{u}\|_\infty}$	$\|\,[\mathbf{u}_\delta]\,\|_\infty$
$j=4$	$6.489e-04$	$8.392e-03$	$1.831e-03$	$1.682e-08$
$j=5$	$4.906e-05$	$9.527e-04$	$1.626e-04$	$1.779e-09$
$j=6$	$3.638e-05$	$2.141e-04$	$5.086e-05$	$2.216e-10$

Table 18: 3×3 D.D - level j and $j+1$ alternated.

level	$\frac{\lVert \mathbf{u}-\mathbf{u}_\delta \rVert_0}{\lVert \mathbf{u} \rVert_0}$	$\frac{\lVert \mathbf{u}-\mathbf{u}_\delta \rVert_{1,*}}{\lVert \mathbf{u} \rVert_{1,*}}$	$\frac{\lVert \mathbf{u}-\mathbf{u}_\delta \rVert_\infty}{\lVert \mathbf{u} \rVert_\infty}$	$\lVert [\mathbf{u}_\delta] \rVert_\infty$
$j=4$	$2.084e-04$	$4.112e-03$	$8.221e-04$	$1.683e-04$
$j=5$	$3.044e-05$	$7.332e-04$	$1.104e-04$	$2.005e-05$

Table 19: 2×1 D.D: Wavelet/FEM coupling. The parameter j indicates the level of the wavelet decomposition and the parameter h indicates the meshsize of the unstructured finite element decomposition whose trace on the interface is uniform.

decompositions	$\frac{\Vert \mathbf{u}-\mathbf{u}_\delta \Vert_0}{\Vert \mathbf{u} \Vert_0}$	$\frac{\Vert \mathbf{u}-\mathbf{u}_\delta \Vert_{1,*}}{\Vert \mathbf{u} \Vert_{1,*}}$	$\frac{\Vert \mathbf{u}-\mathbf{u}_\delta \Vert_\infty}{\Vert \mathbf{u} \Vert_\infty}$	$\Vert\, [\mathbf{u}_\delta] \,\Vert_\infty$
$j = 3$, $h = .0625$	$1.162e-02$	$9.124e-02$	$1.183e-02$	$1.457e-02$
$j = 3$, $h = .03125$	$1.309e-02$	$9.42e-02$	$2.085e-02$	$2.053e-02$
$j = 4$, $h = .03125$	$2.88e-03$	$3.348e-02$	$3.36e-03$	$4.39e-03$
$j = 5$, $h = .015625$	$2.92e-03$	$2.166e-02$	$1.487e-02$	$1.11e-03$

FIGURE CAPTIONS

Figure 1: Behavior of the exact solution of Case 3, together with its gradient u_x along the x axis and its Laplacian.

Figure 2: Logarithmic plot of the L^2 norm and H^1 broken norm of the error in the cases of the 2×1 and 2×2 decompositions of Tables 1 and 4.

Figure 3: WAV/FEM coupling. The master(left) subdomain discretized by finite elements on unstructured mesh and the slave (right) one by wavelets. The mesh defined on the master side of the interface corresponds to the trace of the unstructured triangular mesh of the associated subdomain, it is uniform and dyadic and is finer and therefore strictly included in the mesh defined on the slave side. In particular the parameters correspond to the second case of Table 19.

Figure 4: WAV/FEM coupling. An analogous situation as in Figure 3 is presented. The values of the parameters of the mesh-size defined on the interface corresponds to the third case of Table 19.

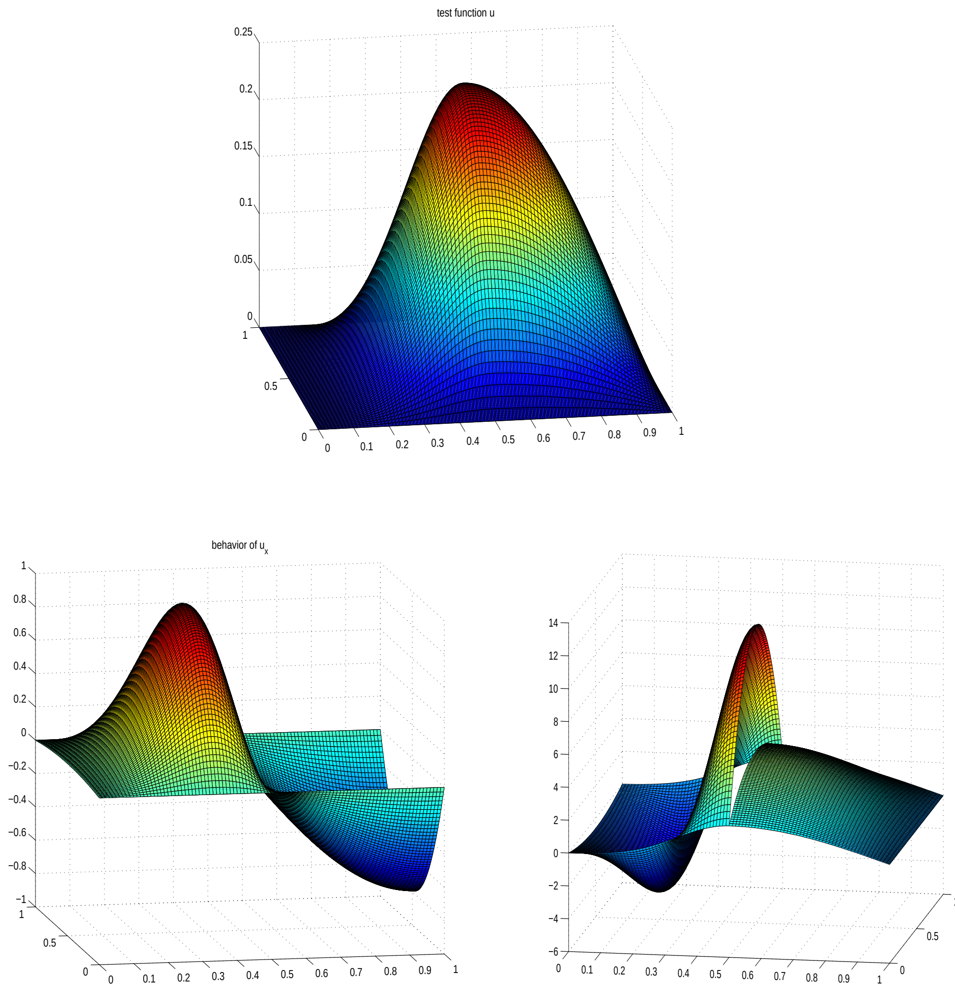
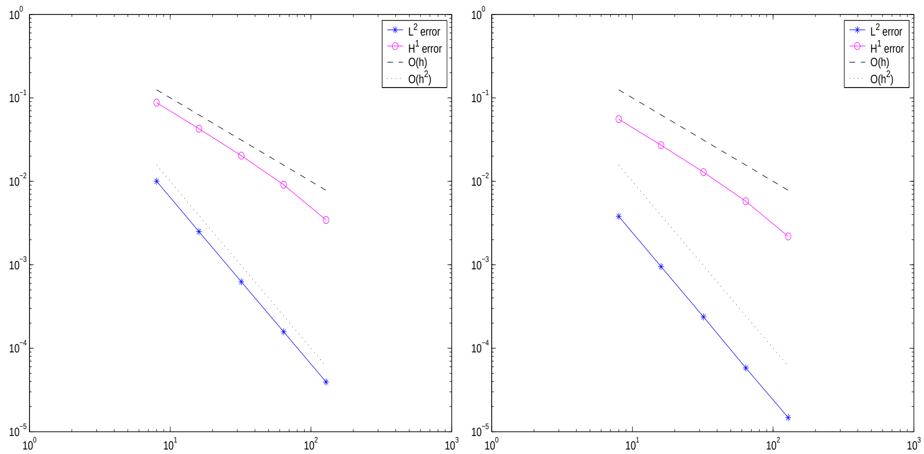


Figure 1:



10^0
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
10^0
10^1
10^2
10^3
L^2 error
H^1 error
O(h)
O(h^2)
10^0
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
10^0
10^1
10^2
10^3
L^2 error
H^1 error
O(h)
O(h^2)

Figure 2:

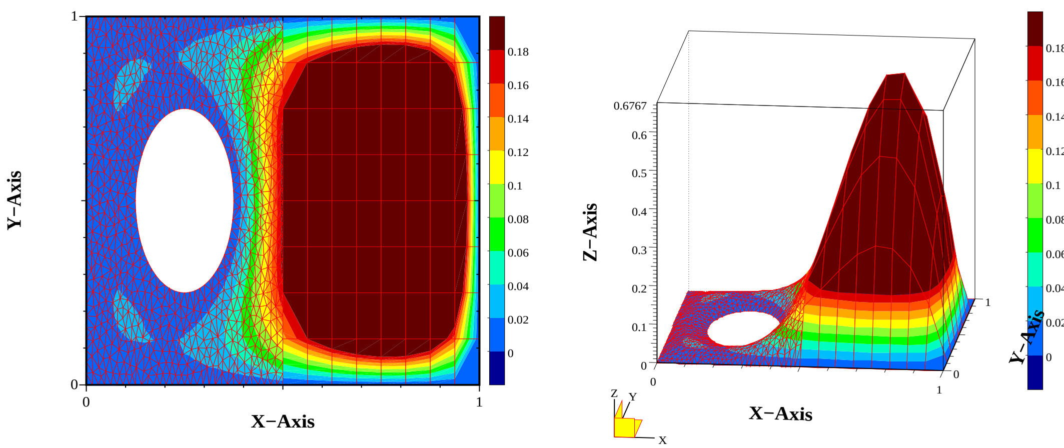
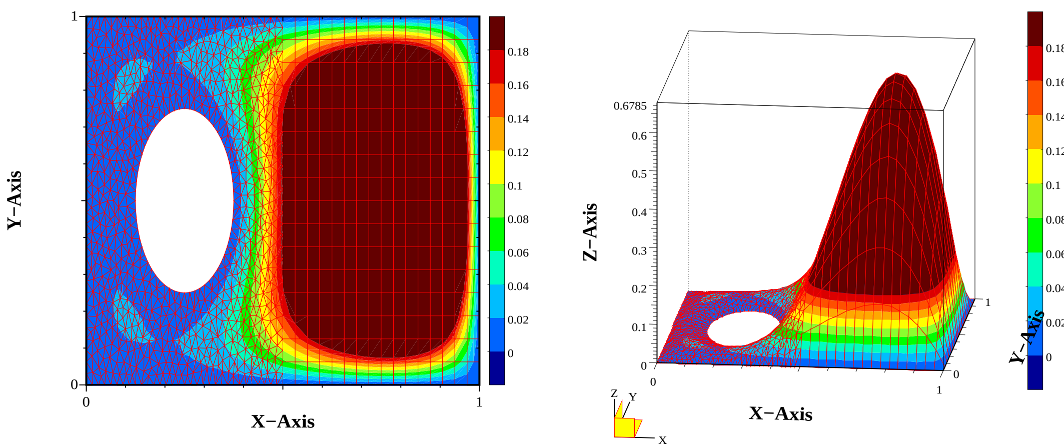

Y−Axis
X−Axis
0
1
0.18
0.16
0.14
0.12
0.1
0.08
0.06
0.04
0.02
0
Z−Axis
0.6767
0.6
0.5
0.4
0.3
0.2
0.1
Y−Axis
Z
Y
X

Figure 3:



Y−Axis
X−Axis
0
1
0.18
0.16
0.14
0.12
0.1
0.08
0.06
0.04
0.02
0
Z−Axis
0.6785
0.6
0.5
0.4
0.3
0.2
0.1
Y−Axis

Figure 4: