ELSEVIER

A computational methodology to study coupled physical processes over partitioned domains

Scott R. Franklin [a,1], Padmanabhan Seshaiyer [b,*], Philip W. Smith [b,1]

[a] *Division of Mathematics and Sciences, Wayland Baptist University, Plainview, TX 79072-6998, USA*
[b] *Department of Mathematics and Statistics, Texas Tech University, Lubbock, TX 79409-1042, USA*

Abstract

In this paper, we describe a computational methodology to couple physical processes defined over independent subdomains, that are partitions of a global domain in three-dimensions. The methodology presented helps to compute the numerical solution on the global domain by appropriately piecing the local solutions from each subdomain. We discuss the mixed method formulation for the technique applied to a model problem and derive an error estimate for the finite element solution. We demonstrate through numerical experiments that the method is robust and reliable in higher dimensions.
© 2005 Elsevier Inc. All rights reserved.

Keywords: Finite elements; Three-field; Non-matching grids; Domain decomposition

1. Introduction

Over the last few decades, there has been a considerable amount of research into developing new techniques to solve coupled physical processes over increasingly complex domains. In practice, such complex global domains are often decomposed into several non-overlapping subdomains and the physical processes are then studied independently over each subdomain. The local solutions, hence, obtained over each subdomain are then assembled to produce a global solution to the multi-physics over the global domain.

Fig. 1 illustrates such a modeling complexity on a global domain Ω. To simplify the solution process, the global domain is partitioned into two subdomains Ω_1 and Ω_2 that may be modeled by different analysts. Moreover, to obtain faster solutions the multi-physics problem is often solved locally over each subdomain, simultaneously. During this process, each analyst may employ different finite element discretization procedures and hence the meshes may be differently constructed over each subdomain. This causes the meshes of the subdomains to not conform at the common interface.
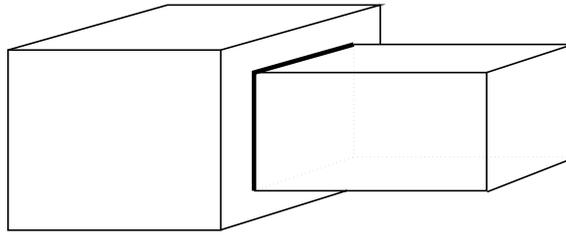

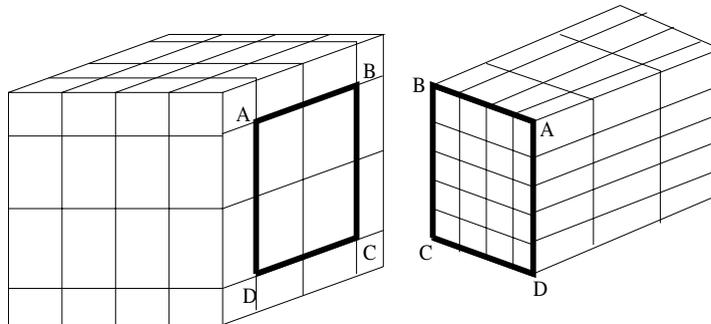Fig. 1. Global Domain $\Omega = \Omega_1 \cup \Omega_2$.

Fig. 2. Independently modeled subdomains Ω_1 and Ω_2 with non-matching grids on the interface ABCD.

Fig. 2 illustrates this, where the subdomains Ω_1 and Ω_2 are modeled using different submeshes that do not conform at the common interface ABCD. In order, to assemble the locally constructed solutions to obtain a global solution in Ω, one must employ an efficient method to *glue* the local solutions together at the interface ABCD. Such a coupling technique can be extremely useful in engineering applications when pre-meshed components need to be incorporated into an existing model.

This interdomain coupling can be enforced in a variety of ways. Let u_i be the value of the trace of the solution on the interface ABCD from subdomain Ω_i. One approach is to introduce a Lagrange multiplier on the interface ABCD to enforce the inter-domain continuity

$$u_1 = u_2 \quad \text{over ABCD} \tag{1.1}$$

An example of such a technique is the mortar finite element method ([6,15,2,5,3] and references therein). Here precise choices for the two fields (the interior solution variables and the interface Lagrange multiplier) are made to make the system stable [14]. Other choices of two-field Lagrange multiplier methods can be found in [18,16]. Note that the description of the two-field methods are in two dimensions and the extension to three dimensions (for tensor product spaces) can be found in [4,16,14].

Alternatively, one may also introduce two Lagrange multipliers (corresponding to the interior solution variable in each subdomain) on ABCD and an interface variable z (corresponding to the trace of the exact solution) on ABCD and enforce

$$u_i - z = 0 \quad \text{over ABCD} \tag{1.2}$$

for $i = 1, 2$. Once again, as in the two-field method to ensure the method is stable, one must make precise choices for the three fields: the first field being the interior solution variable, the second field being the interface Lagrange Multiplier and the third field being an artificial interface displacement. Eq. (1.2) is enforced as a weak continuity constraint to help glue the local interior solutions. This method allows more flexibility in the choice of the approximation spaces and therefore, has obvious advantages over other coupling methods. One can refer to [17] for an implementation of this technique in two dimensions. Variants of this technique in two dimensions can also be found in [1,7–9]. Let us also mention the existence of other domain decomposition algorithms and some parallel implementations in the literature (see for e.g. [12,13,19]).

This brings us to the motivation behind this paper. Our goal is to consider the suitability of three-field methods in three dimensions for *hp* implementation, to couple solutions over subdomains with non-matching grids. In this paper, we present the mixed method formulation and implementation of the three-field method for a model problem in three-dimensions. We test the accuracy and reliability of the technique by performing the *h*-version (where polynomials of fixed degree *p* are used and the mesh is refined to increase accuracy) and the *p*-version (where the method uses a fixed mesh but increases the polynomial degree *p* to increase accuracy). Finally, we test the three-field method developed for sub-domains with non-matching grids against those with matching grids. Note that the latter simply yields a conforming standard finite element formulation for which the results are well known.

2. Model problem and discretization

We consider the following second-order elliptic problem which models a steady-state convection diffusion problem given by

$$
\begin{aligned}
-\operatorname{div}(a\nabla u) + bu &= f && \text{on } \Omega \\
u &= 0 && \text{on } \partial\Omega_D \\
a\frac{\partial u}{\partial n} &= g && \text{on } \partial\Omega_N
\end{aligned}
\tag{2.3}
$$

Here a is a uniformly positive function and $b \geqslant 0$ in the bounded domain $\Omega \subset \mathbb{R}^3$ with boundary $\partial\Omega = \overline{\partial\Omega_D} \cup \overline{\partial\Omega_N}$ $(\partial\Omega_D \cap \partial\Omega_N = \emptyset, \ \partial\Omega_D \neq \emptyset)$. The form (2.3) is equivalent to the following variational form. Find $u \in E(\Omega)$ satisfying, for all $v \in E(\Omega)$,

$$
a(u, v) \overset{\text{def}}{=} \int_\Omega a\nabla u \cdot \nabla v + buv \, dx = \int_\Omega fv \, dx + \int_{\partial\Omega_N} gv \, ds \overset{\text{def}}{=} F(v)
\tag{2.4}
$$

Here, the *energy space* $E(\Omega)$ is seen to be, $E(\Omega) = \{u \in H^1(\Omega) | u = 0 \text{ on } \partial\Omega_D\} \overset{\text{def}}{=} H^1_D(\Omega)$, where we are using $H^k(\Omega)$ to denote the space of functions with k generalized derivatives on Ω. We set $L_2(\Omega) = H^0(\Omega)$ and denote by both $\|\cdot\|_{k,A}$ and $\|\cdot\|_{H^k(A)}$ the norm of $H^k(A)$. Note that the definition of these spaces can be extended to non-integer values of k by interpolation.

Let us assume Ω is the union of S non-overlapping polygonal subdomains $\{\Omega_i\}_{i=1}^S$ such that $\partial\Omega_i \cap \partial\Omega_j \,(i<j)$ is either empty, a vertex, or an edge or an entire face of Ω_i and Ω_j. The above conformity condition can be relaxed, since by using the arguments of [2], our results (with some minor changes) extend to non-conforming decompositions as well. Let us for simplicity, assume that the union of all intersections $\partial\Omega_i \cap \partial\Omega_j, \ i<j$, is the face Γ_{ij}. We set the *interface set* Γ to be the union of all Γ_{ij}. We further subdivide Ω_i into a sequence of geometrically conforming, shape regular [10] triangulations, $\{\mathcal{T}_h^i\}$. It should be noted that the triangulations over different Ω_i are independent of each other, with no compatibility enforced across interfaces. Since the meshes $\{\mathcal{T}_h^i\}$ are not assumed to conform across interfaces, two separate trace meshes can be defined on Γ_{ij}, one from Ω_i and the other form Ω_j.

2.1. Mixed formulation

Our problem of finding the continuous solution *u* satisfying (2.3) becomes solving (2.5) for u_i, the interior solution variable in each Ω^i

$$
\begin{aligned}
-\operatorname{div}(a\nabla u_i) + bu_i &= f_i && \text{on } \Omega_i \\
u_i &= 0 && \text{on } \partial\Omega_D \cap \partial\Omega_i \\
a\frac{\partial u_i}{\partial n^i} &= g_i && \text{on } \partial\Omega_N \cap \partial\Omega_i
\end{aligned}
\tag{2.5}
$$

Multiplying the partial differential equation in (2.5) by a test function $v_i \in H_D^1(\Omega_i)$, integrating by parts and applying the boundary conditions gives

$$a_S(u,v) + b_S(v,\lambda) = F(v) \tag{2.6}$$

where $a_S(u,v)$ is the bilinear form defined by

$$a_S(u,v) = \sum_{i=1}^{S} \int_{\Omega_i} a\nabla u_i.\nabla v_i + bu_i v_i \, \mathrm{d}x \tag{2.7}$$

and $b_S(v,\lambda)$ is the bilinear form defined by

$$b_S(v,\lambda) = \sum_{\Gamma_{ij} \subset \Gamma} \int_{\Gamma_{ij}} \bar{v}_i \lambda^i + \bar{v}_j \lambda^j \, \mathrm{d}s \tag{2.8}$$

Here $\bar{v}_i$ and $\bar{v}_j$ denote the traces of the function v_i and v_j on Γ_{ij}, respectively. The fluxes at the interface are $\lambda^i = -a\frac{\partial u_i}{\partial n^i} \in H^{-\frac{1}{2}}(\Gamma_{ij})$ and $\lambda^j = -a\frac{\partial u_j}{\partial n^j} \in H^{-\frac{1}{2}}(\Gamma_{ij})$, with n^i and n^j being the corresponding unit outward normals from Ω_i and Ω_j, respectively at the interface Γ_{ij}. Note that $\lambda \in \mathbf{H}^{-\frac{1}{2}}(\Gamma_{ij}) = H^{-\frac{1}{2}}(\Gamma_{ij}) \times H^{-\frac{1}{2}}(\Gamma_{ij})$.

It must also be noted that (2.6) must be solved along with the continuity condition enforced on $\bar{u}_i$, the trace of the solution u_i on Γ_{ij} given by

$$\int_{\Gamma_{ij}} \bar{u}_i \chi^i \, \mathrm{d}s = \int_{\Gamma_{ij}} t\chi^i \, \mathrm{d}s \tag{2.9}$$

where χ^i is any function in $H^{-\frac{1}{2}}(\Gamma_{ij})$ and t is a new unknown we introduce, called the *interface displacement* which belongs to $H^{\frac{1}{2}}(\Gamma_{ij})$. (Note that $H^{-\frac{1}{2}}(\Gamma_{ij})$ is the topological dual space of $H^{\frac{1}{2}}(\Gamma_{ij})$.) Using (2.8) and (2.9), we have

$$b_S(u,\chi) = \sum_{\Gamma_{ij} \subset \Gamma} \int_{\Gamma_{ij}} (\chi^i + \chi^j)t \, \mathrm{d}s \tag{2.10}$$

Also, since the solution is smooth in the interior in Ω, we have $\lambda^i + \lambda^j = 0$, which can be rewritten using a bilinear form $c_S(\lambda,\mu)$ as

$$c_S(\lambda,\mu) := -\sum_{\Gamma_{ij} \subset \Gamma} \int_{\Gamma_{ij}} (\lambda^i + \lambda^j)\mu \, \mathrm{d}s = 0 \quad \forall \mu \in \prod_{\Gamma_{ij} \subset \Gamma} H^{\frac{1}{2}}(\Gamma_{ij}) \tag{2.11}$$

The problem (2.3) can now be stated in *mixed form*: Find $(u,\lambda,t) \in \prod_{i=1}^{S} H_D^1(\Omega_i) \times \prod_{\Gamma_{ij} \subset \Gamma} \mathbf{H}^{-\frac{1}{2}}(\Gamma_{ij}) \times \prod_{\Gamma_{ij} \subset \Gamma} H^{\frac{1}{2}}(\Gamma_{ij})$ such that

$$a_S(u,v) + b_S(v,\lambda) = F(v) \tag{2.12}$$
$$b_S(u,\chi) + c_S(\chi,t) = 0 \tag{2.13}$$
$$c_S(\lambda,\mu) = 0 \tag{2.14}$$

for all $(v,\chi,\mu) \in \prod_{i=1}^{S} H_D^1(\Omega_i) \times \prod_{\Gamma_{ij} \subset \Gamma} \mathbf{H}^{-\frac{1}{2}}(\Gamma_{ij}) \times \prod_{\Gamma_{ij} \subset \Gamma} H^{\frac{1}{2}}(\Gamma_{ij})$.

2.2. Finite element discretization

Next, we discretize the problem in the mixed formulation (2.12)–(2.14) by the finite element method. Let

$$u_N \in V \subset \prod_{i=1}^{S} H_D^1(\Omega_i) \quad \lambda_N \in \Lambda \subset \prod_{\Gamma_{ij} \subset \Gamma} \mathbf{H}^{-\frac{1}{2}}(\Gamma_{ij}) \quad t_N \in T \subset \prod_{\Gamma_{ij} \subset \Gamma} H^{\frac{1}{2}}(\Gamma_{ij})$$

Our discrete problem can be stated as follows: Find $(u_N, \lambda_N, t_N) \in V \times \Lambda \times T$ such that

$$a_S(u_N, v_N) + b_S(v_N, \lambda_N) = F(v_N) \tag{2.15}$$
$$b_S(u_N, \chi_N) + c_S(\chi_N, t_N) = 0 \tag{2.16}$$
$$c_S(\lambda_N, \mu_N) = 0 \tag{2.17}$$

for all $(v_N, \chi_N, \mu_N) \in V \times \Lambda \times T$.

We now define specific finite dimensional spaces that we use to implement the three-field algorithm and generate numerical examples later. It is common to use piecewise polynomials as basis elements in such finite element formulations; however, we have chosen to use tensor product B-splines due to their simplicity in formulation and efficiency of evaluation. Additionally, they form a basis for piecewise polynomials over tensor product domains with certain controls over continuity at element boundaries.

Let $\mathbf{t} = \{t_i\}$ be a sequence of real values such that $t_i \leqslant t_{i+1} < t_{i+k}$, for which k is some positive integer. This sequence corresponds to the subintervals for the piecewise polynomials, with multiplicity lowering order of continuity at each break point. We define the ith univariate B-spline of order k for the knot sequence $\mathbf{t}$, denoted $B_{i,k,\mathbf{t}}$, by the following:

$$B_{i,k,\mathbf{t}}(x) = (t_{i+k} - t_i)[t_i, \ldots, t_{i+k}]_s(s - x)_+^{k-1} \tag{2.18}$$

where $[t_i, \ldots, t_{i+k}]_s(s - x)_+^{k-1}$ is the kth divided difference over s of the truncated power function. The following well-known properties of these functions make them a suitable choice for basis functions in finite elements (see [11]):

1. $B_{i,k,\mathbf{t}}$ has local support, i.e., $B_{i,k,\mathbf{t}}(x) = 0$ if $x \notin [t_i, t_{i+k}]$.
2. $B_{i,k,\mathbf{t}}$ is positive on its support, i.e., $B_{i,k,\mathbf{t}}(x) > 0$ if $t_i < x < t_{i+k}$.
3. The B-splines of order k sum to 1, i.e.,

$$\sum_i B_{i,k,\mathbf{t}} = \sum_{i=r+1-k}^{s-1} B_{i,k,\mathbf{t}}(x) = 1, \quad t_r < x < t_s \tag{2.19}$$

4. The B-splines of order k are linearly independent.

As mentioned, the B-splines, $\{B_{i,k,\mathbf{t}}: i = 0, \ldots, k\}$ form a basis for the piecewise polynomials of order k with break points defined by $\mathbf{t}$.

For simplicity, we restrict our domains and subdomains to products of intervals, i.e., $[x_1, x_2] \times [y_1, y_2] \times [z_1, z_2]$. We choose knot sequences $\mathbf{t}_i$, $i = 1$–3, one for each spatial dimension of the domain. Let us define the trivariate tensor product B-splines

$$B_{lmn}(x, y, z) = B_{l,k,\mathbf{t}_1}(x)B_{m,k,\mathbf{t}_2}(y)B_{n,k,\mathbf{t}_3}(z).$$

Note that we have freedom to choose the knot sequences and spline order, which may be chosen independently for each spatial dimension. However, for the results that follow, we assume they are the same.

For each subdomain, Ω_i, we let V_i be the span of such tensor product B-splines. For each interface, Γ_{ij} we use bivariate tensor product B-splines, letting $\Lambda_{ij} = \text{Span}\{B_{lm}\} \times \text{Span}\{B_{l'm'}\}$, noting that the basis elements are defined with specific knot sequences corresponding to the interface Γ_{ij}. Equivalently, we define the trace space at each interface, Γ_{ij}, by $T_{ij} = \text{Span}\{B_{lm}\}$ which may be chosen independently from Λ_{ij}. Then our finite dimensional spaces are given by

$$V = \prod_{i=1}^{S} V_i, \quad \Lambda = \prod_{\Gamma_{ij} \subset \Gamma} \Lambda_{ij} \quad \text{and} \quad T = \prod_{\Gamma_{ij} \subset \Gamma} T_{ij}$$

It should be noted that for our numerical experiments mentioned later we set the meshes on each of the spaces, Λ_{ij} and T_{ij}, as the coarsest mesh of the adjacent spaces. One notable result of these spline spaces follows from the local support mentioned above. For any two basis elements out of the spline space, V_i, we have

$$B_{lmn}B_{l'm'n'} = 0 \quad \text{if} \quad \max\{|l - l'|, |m - m'|, |n - n'|\} > k$$

which leads to sparsity in the stiffness matrix.

As the next natural step in the finite element procedure, we express the unknowns u_N, λ_N, t_N as a linear combination of the respective basis functions. Then, choosing the test functions v_N, χ_N, μ_N to be basis functions themselves then converts the above system of integral equations into a matrix system that is solved for the unknowns u_N, λ_N, t_N.

3. An error estimate

We will now prove that the error in the solution for this mixed method is bounded by an *approximation* error term $e_A(u)$ and a *consistency* error term $e_C(u)$. Let us now define the spaces

$$\mathscr{X} = \prod_{i=1}^{S} H_D^1(\Omega_i) \times \prod_{\Gamma_{ij} \subset \Gamma} \mathbf{H}^{-\frac{1}{2}}(\Gamma_{ij}) \times \prod_{\Gamma_{ij} \subset \Gamma} H^{\frac{1}{2}}(\Gamma_{ij})$$

$$X = V \times \Lambda \times T$$

Recall that the exact solution $(u, \lambda, t) \in \mathscr{X}$ satisfies (2.12)–(2.14) while the approximate solution (u_N, λ_N, t_N) satisfies (2.15)–(2.17). Let us define, $\left(\sum_{i=1}^{S} \|\nabla v^i\|_{0,\Omega_i}^2 \right)^{\frac{1}{2}} = |v|_{1,S}$.

Also, let X_N be the subspace of X satisfying the homogeneous Eq. (2.16). We then have the following theorem. Note that μ_N is specifically associated with v_N such that the homogeneous Eq. (2.16) hold for all $\chi_N \in \Lambda$.

Theorem 3.1. *If $(u, \lambda, t) \in \mathscr{X}$ satisfies (2.12)–(2.14) and $(u_N, \lambda_N, t_N) \in X$ satisfies (2.15)–(2.17) then*

$$|u - u_N|_{1,S} \leqslant C \left(\inf_{w_N \in X_N} |u - w_N|_{1,S} + \sup_{v_N \in X_N} \inf_{\xi_N \in \Lambda} \frac{b_S(v_N - \mu_N, \lambda - \xi_N)}{|v_N|_{1,S}} \right) \tag{3.20}$$

Proof. For any $v_N \in X_N$ and $w_N \in X_N$, we have from (2.12) and (2.15)

$$a_S(u_N - w_N, v_N) + b_S(v_N, \lambda_N) = a_S(u - w_N, v_N) + b_S(v_N, \lambda) \tag{3.21}$$

Using (2.16) and (2.17) we have

$$b_S(v_N, \lambda_N) = 0 \tag{3.22}$$

As mentioned above, let us choose $\mu_N \in T \subseteq \prod_{\Gamma_{ij} \subset \Gamma} H^{\frac{1}{2}}(\Gamma_{ij})$ associated with v_N such that the homogeneous Eq. (2.16) hold for all $\chi_N \in \Lambda$. Thus (2.14) yields

$$b_S(v_N, \lambda) = b_S(v_N, \lambda) + c_S(\lambda, \mu_N) \tag{3.23}$$

Now for all $\xi_N \in \Lambda$, (2.16) gives

$$b_S(v_N, \xi_N) + c_S(\xi_N, \mu_N) = 0 \tag{3.24}$$

Using (3.24), we can rewrite (3.23) as

$$\begin{aligned} b_S(v_N, \lambda) &= b_S(v_N, \lambda - \xi_N) + c_S(\lambda - \xi_N, \mu_N) \\ &= b_S(v_N - \mu_N, \lambda - \xi_N) \end{aligned} \tag{3.25}$$

Substituting (3.22) and (3.25) in (3.21) yields

$$a_S(u_N - w_N, v_N) = a_S(u - w_N, v_N) + b_S(v_N - \mu_N, \lambda - \xi_N) \tag{3.26}$$

Dividing by $|v_N|_{1,S}$ throughout and taking the supremum over all $v_N \in X_N$, Eq. (3.26) becomes

$$\sup_{v_N \in X_N} \frac{a_S(u_N - w_N, v_N)}{|v_N|_{1,S}} \leqslant \sup_{v_N \in X_N} \frac{a_S(u - w_N, v_N)}{|v_N|_{1,S}} + \sup_{v_N \in X_N} \frac{b_S(v_N - \mu_N, \lambda - \xi_N)}{|v_N|_{1,S}} \tag{3.27}$$

Since $u_N \in X_N$, we may choose $v_N = u_N - w_N$ so that, the left hand side of (3.27) becomes

$$\sup_{v_N \in X_N} \frac{a_S(u_N - w_N, v_N)}{|v_N|_{1,S}} \geqslant C_c |u_N - w_N|_{1,S} \tag{3.28}$$

where we have used the coercivity of the bilinear form $a_S(\cdot,\cdot)$. For the first term on the right hand side of (3.27), we can use the boundedness of the bilinear form $a_S(\cdot,\cdot)$ to yield

$$\sup_{v_N \in X_N} \frac{a_S(u - w_N, v_N)}{|v_N|_{1,S}} \leqslant C_b |u - w_N|_{1,S} \tag{3.29}$$

Using the triangle inequality, we have

$$|u - u_N|_{1,S} \leqslant |u - w_N|_{1,S} + |u_N - w_N|_{1,S} \tag{3.30}$$

Using (3.27)–(3.29), one can then simplify (3.30) to obtain

$$|u - u_N|_{1,S} \leqslant C \left(\inf_{w_N \in X_N} |u - w_N|_{1,S} + \sup_{v_N \in X_N} \inf_{\xi_N \in \Lambda} \frac{b_S(v_N - \mu_N, \lambda - \xi_N)}{|v_N|_{1,S}} \right) \qquad \square$$

4. Numerical results

In this section, we demonstrate the performance of the numerical technique discussed in Section 2 in three spatial dimensions. Our calculations are performed for the model problem (2.3) over the domain $[0,2] \times [0,1] \times [0,1]$ (see Fig. 3). We partition this domain into two sub-domains Ω_1 and Ω_2, by the interface plane $x = 1$. For this example, we implement using only Dirichlet boundary condition for the domain, Ω. Other boundary conditions can also be easily implemented. We utilized a direct linear solver for the solution of the large matrix systems. Specifically, we used and tested the package MUMPS (a MUltifrontal Massively
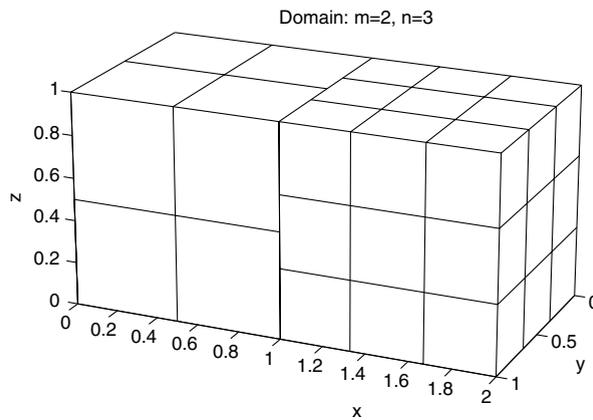
Fig. 3. Domain decomposition where $m = 2$ and $n = 3$.

Table 4.1

p_1	q_1	r	q_2	p_2	m	n	DOF	H^1 error
3	2	1	2	3	2	3	157	11.87675871
3	2	1	2	3	4	6	639	2.32783819
3	2	1	2	3	6	9	1669	0.87902316
3	2	1	2	3	8	12	3457	0.45012675
3	2	1	2	3	10	15	6213	0.27060945
3	2	1	2	3	12	18	10147	0.17972068
3	2	1	2	3	14	21	15469	0.12750407

Table 4.2

q_1	r	q_2	H^1 rate	q_1	r	q_2	H^1 rate	q_1	r	q_2	H^1 rate	q_1	r	q_2	H^1 rate
0	0	0	$O(h^2)$	3	0	3	$O(h)$	0	0	0	$O(h^2)$	2	0	2	$O(h)$
1	1	1	$O(h^3)$	3	1	3	$O(h^2)$	1	1	1	$O(h^2)$	2	1	2	$O(h^2)$
2	2	2	$O(h^3)$	3	2	3	$O(h^3)$	2	2	2	$O(h^2)$	2	2	2	$O(h^2)$
3	3	3	$O(h^3)$	3	3	3	$O(h^3)$								

The two column groups are headed "Convergence rates for $p_1 = p_2 = 3$" and "Convergence rates for $p_1 = p_2 = 2$".

For these experiments, we choose a to be the identity matrix and set $b = 0$. We manufacture the right hand side by choosing the quintic polynomial exact solution

$$u = xyz(x - 2)(y - 1)(z - 1)(x^3 + y^3 + z^3)$$

The non-conforming method is implemented as a *mixed* method discussed in previous sections. As we have described earlier we are implementing using *tensor product* meshes. Although we have the freedom to choose
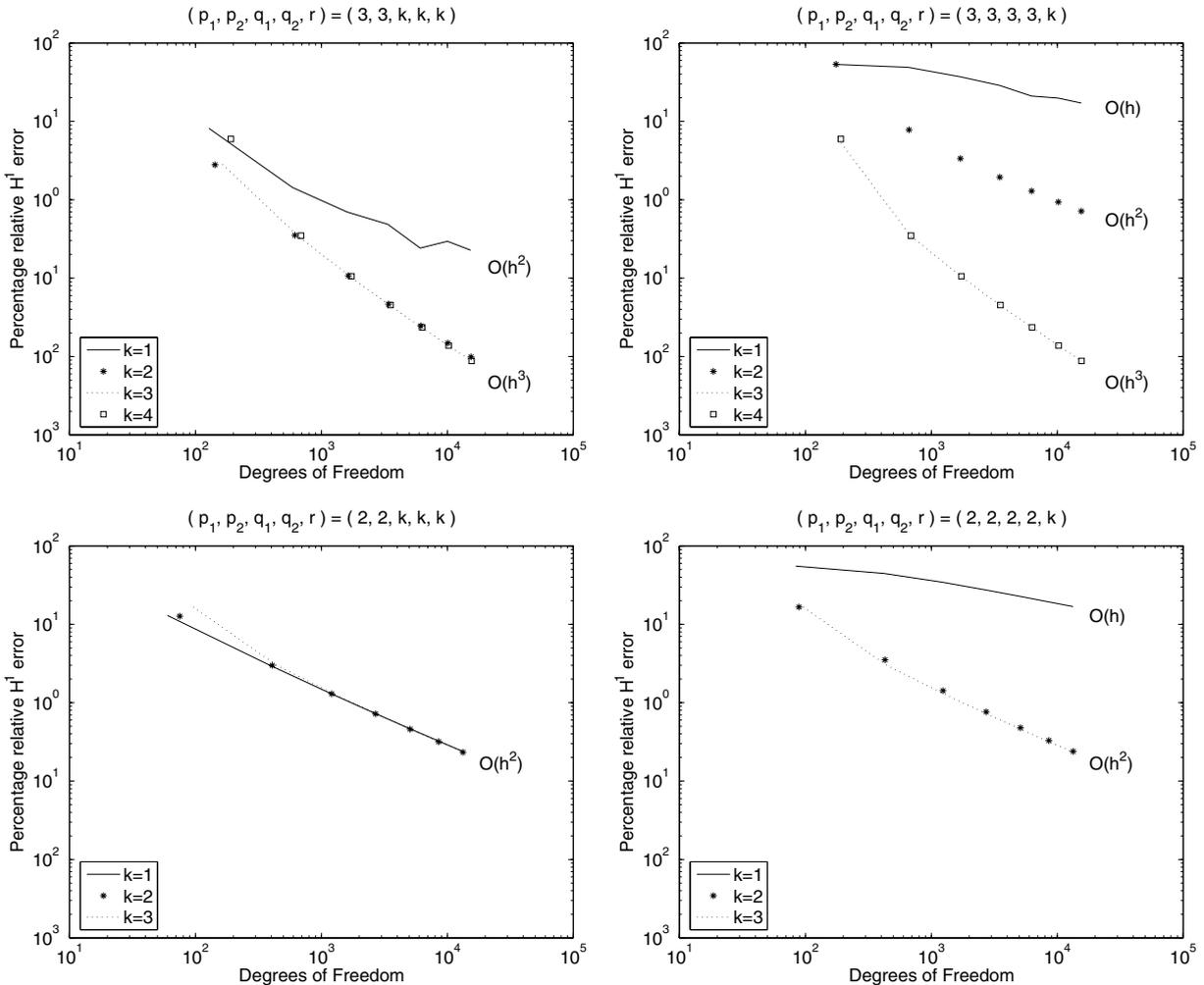

Fig. 4. Convergence rates for the *h*-version with $(m,n) = (2j, 3j)$ for $j = 1, 2, \ldots, 7$.

Table 4.3

p_1	q_1	r	q_2	p_2	m	n	DOF	H^1 error
2	2	1	2	2	4	6	429	3.525612359
3	2	1	2	3	4	6	639	2.327838189
4	2	1	2	4	4	6	925	1.504625224
5	2	1	2	5	4	6	1299	1.506151362
6	2	1	2	6	4	6	1773	1.491283643
7	2	1	2	7	4	6	2359	1.490993315

the number of subintervals for each spatial dimension in the mesh independently, we will restrict each sub-domain, Ω_1 and Ω_2, to be divided into uniformly spaced m and n grid cells, respectively, along each axis. Thus, we have m^3 elements in Ω_1 and n^3 elements in Ω_2. As mentioned earlier, we will set the mesh on the normal and interface spaces, Λ_1, Λ_2, and T, to be the coarsest mesh of the adjacent spaces, V_1 and V_2. Fig. 3 illustrates an example with $m = 2$ and $n = 3$. Having chosen a polynomial solution, we expect that for the conforming case, i.e., $m = n$, our finite element method should be able to recover the exact solution, given a high enough polynomial degree in the finite element spaces.

For $i = 1, 2$, let p_i, q_i, r be the degrees of the B-splines in the approximation spaces V_i, Λ_i, T, respectively. To begin with, we consider the h-version for the non-conforming method, i.e., we fix the degrees for the B-splines and refine the meshes on each sub-domain. For this version, we set $(p_1, p_2, q_1, q_2, r) = (3, 3, 2, 2, 1)$ and consider the mesh refinements, $(m, n) = (2k, 3k)$ where $k = 1, 2, \ldots, 7$. The results are shown in Table 4.1. We denote the number of *degrees of freedom* by DOF, which is the size of the matrix generated in the discretization of the mixed formulation, given by (2.15)–(2.17). The error is the percentage relative error that is calculated in the broken H^1 norm. It is clear from Table 4.1 that the refinement of the mesh results in an increase in dimension of the problem and the error appears to converge to zero. The convergence rate is observed to be $O(h^2)$. It is noted that in the conforming case we expect an optimal rate of convergence of $O(h^3)$ and for the non-conforming case we would not expect an improvement. Through additional experimentation it was noticed that changing the degrees of normal and trace spaces has an effect on the rate of convergence. This is illustrated by Table 4.2 and in Fig. 4 where various convergence rates are given for the h-version where $p_1 = p_2 = 3$ and $(m, n) = (2k, 3k)$ for $k = 1, 2, \ldots, 7$. Additionally, convergence rates are included for the case where $p_1 = p_2 = 2$, for which we expect the optimal convergence rate of $O(h^2)$. The rates are given when the degrees of the discretized trace and normal spaces vary. In most of the test cases, our expected optimal rate of convergence is attained. In future research, we plan to further investigate the relationships between the rate of

Table 4.4

p_1	q_1	r	q_2	p_2	m	n	DOF	H^1 error
5	0	0	0	5	4	6	1250	2.801564749
5	1	1	1	5	4	6	1277	0.162857952
5	2	2	2	5	4	6	1310	0.000096429
5	3	3	3	5	4	6	1349	0.000063664
5	4	4	4	5	4	6	1394	0.000020413
5	5	5	5	5	4	6	1445	0.000019057

Table 4.5

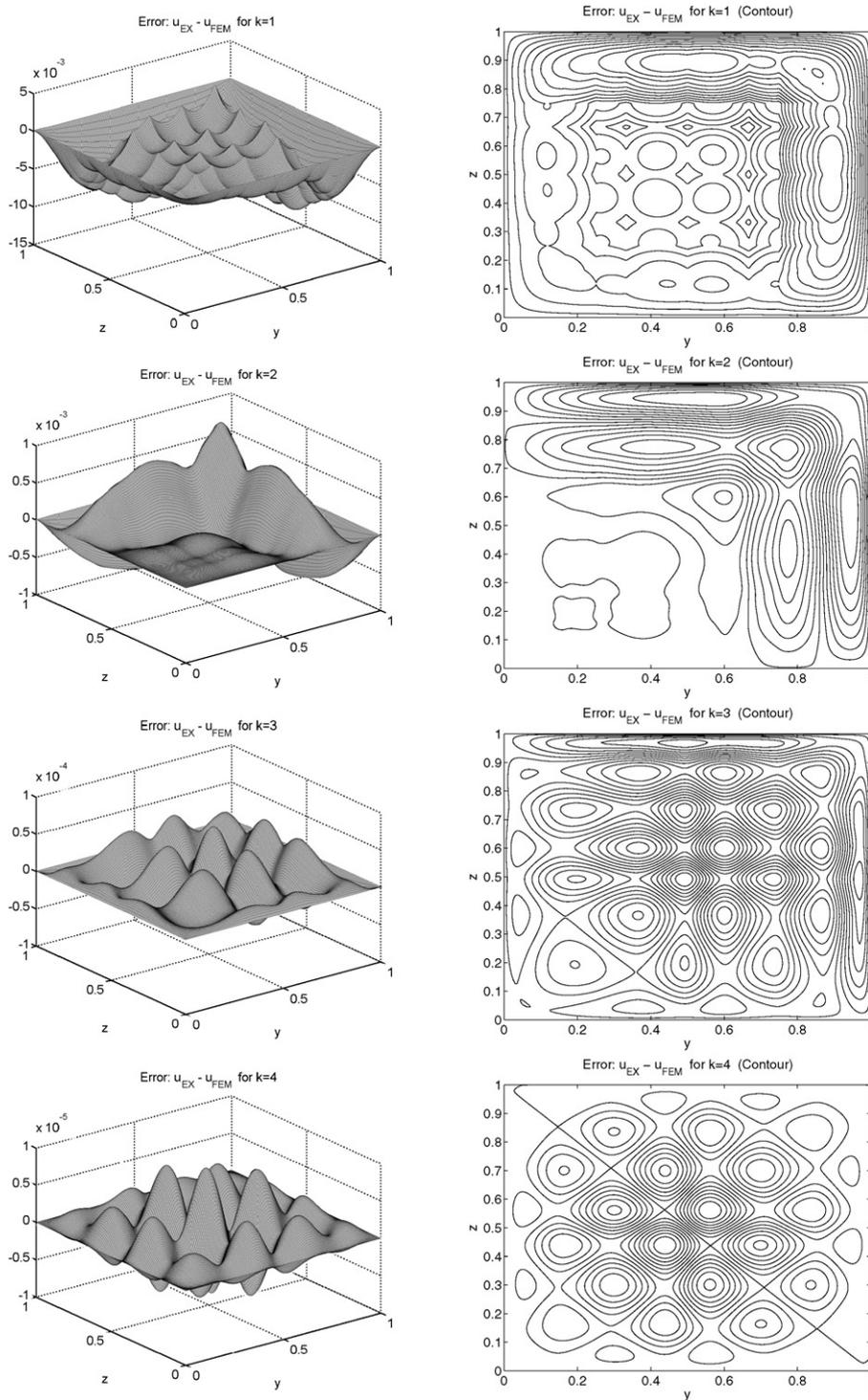| p_1	q_1	r	q_2	p_2	m	n	DOF	H^1 error
2	2	2	2	2	5	5	696	1.68009611
3	2	2	2	3	5	5	976	0.14098045
4	2	2	2	4	5	5	1344	0.01057147
5	2	2	2	5	5	5	1812	0.00000000

Fig. 5. Error on interface with $(m,n) = (4,6)$ for $p_1 = p_2 = k$ for $k = 1$–4 (p-version).

convergence and the degrees of the trace and normal spaces, Λ_1, Λ_2, T. It is clear from the results in these experiments that both terms of the error estimate (3.20) are significant.
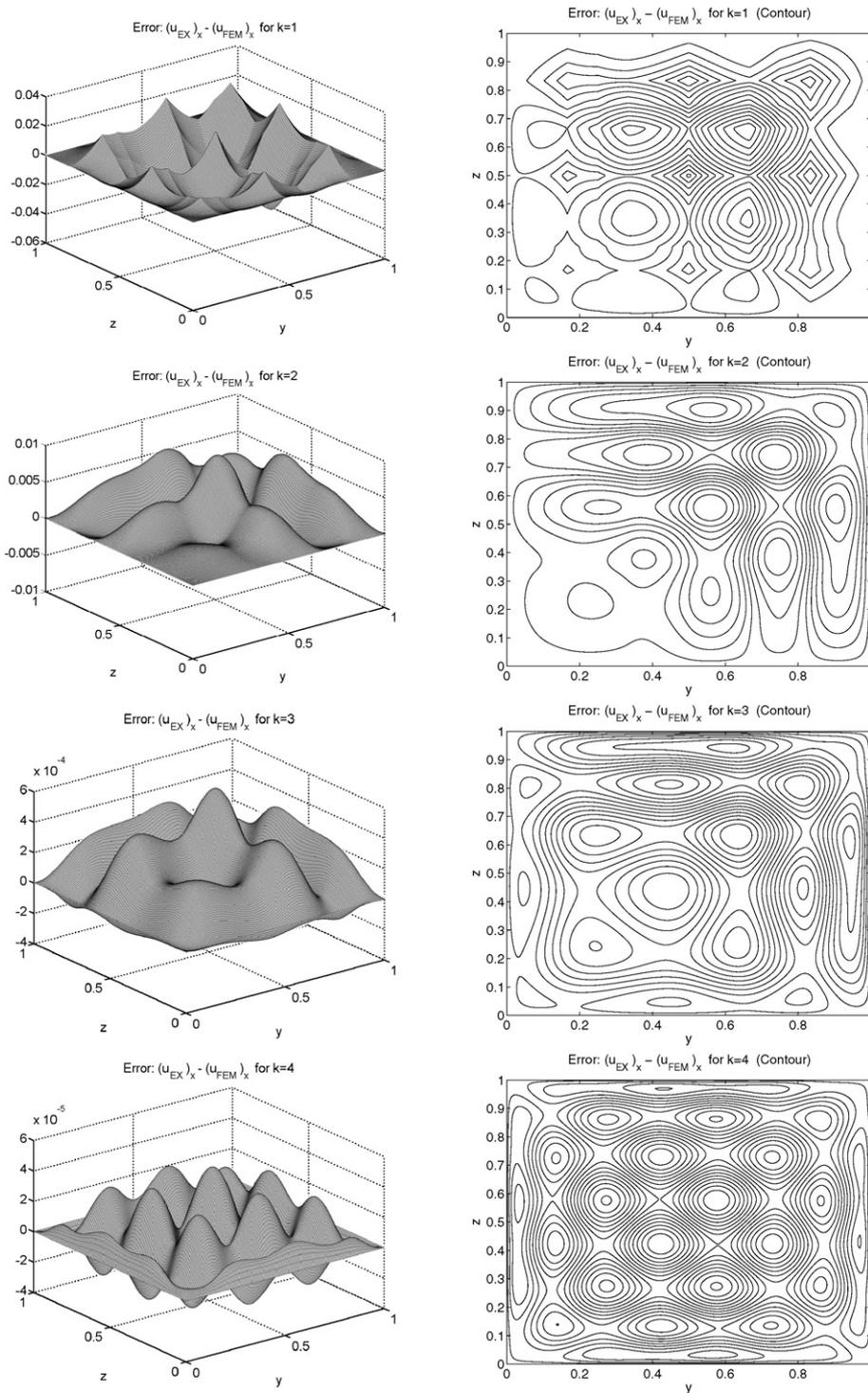
Fig. 6. Error of the first partial with respect to x on interface with $(m,n) = (4,6)$ for $p_1 = p_2 = k$ for $k = 1$–4 (p-version).

Next, we consider the p-version for the non-conforming method, i.e., we fix the mesh sizes for each domain and refine the polynomial spaces to obtain better error. At this point we set $(m,n) = (4,6)$, $(q_1, q_2, r) = (2,2,1)$ and let p_i increase from 2 to 7 for $i = 1, 2$. Thus we are increasing the order of the splines on each space V_i, for
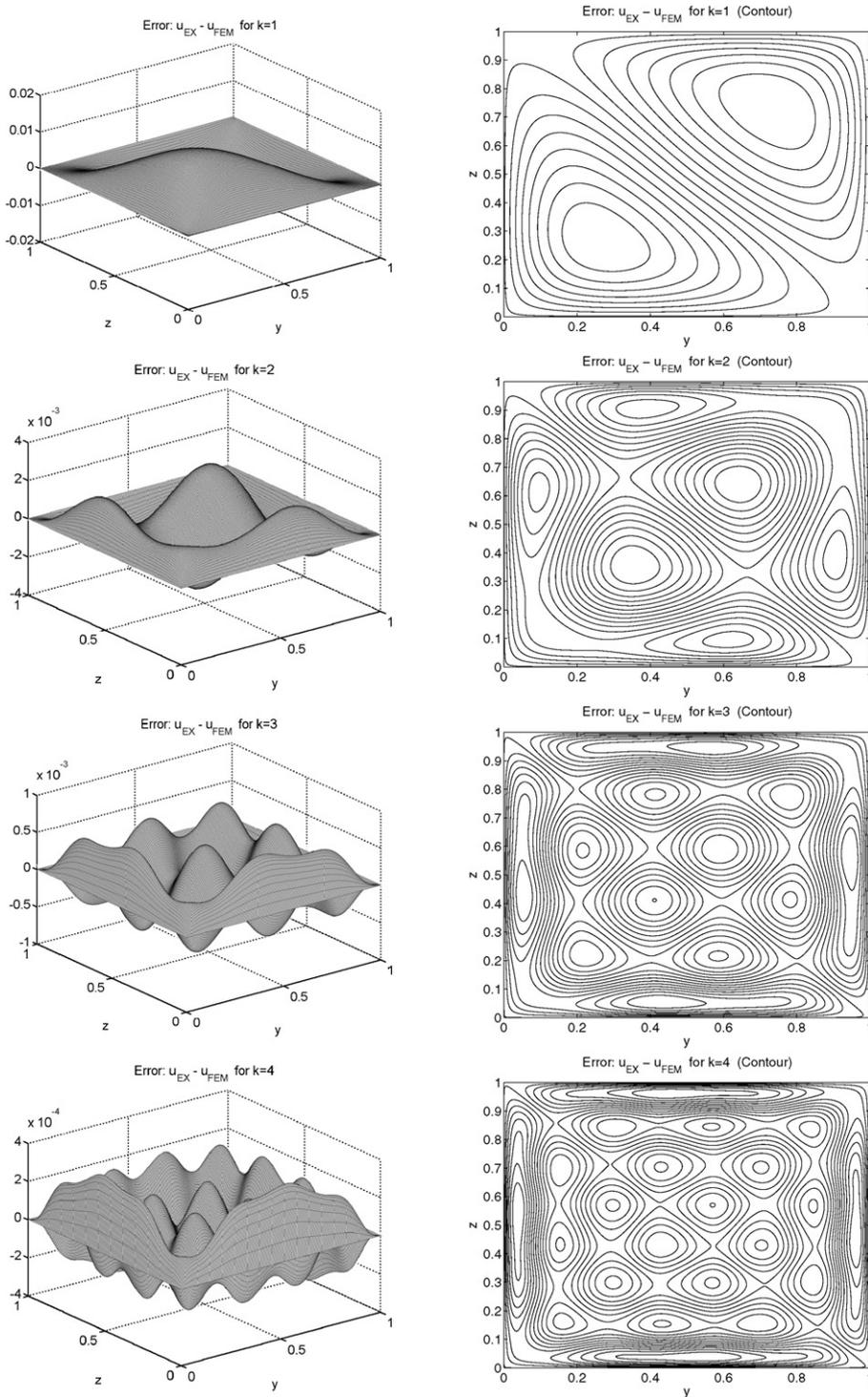
Fig. 7. Error on interface with $(m,n) = (2k, 3k)$ for $k = 1$–4 (h-version).

$i = 1, 2$. The results are shown in Table 4.3. As the order increases, the error does decrease, as expected. Also, note that because of the low order on Λ_1, Λ_2, T, we do not have a drastic decrease in the error. Additionally, monotonocity may not be expected because the spaces are not nested. Therefore, we consider the following
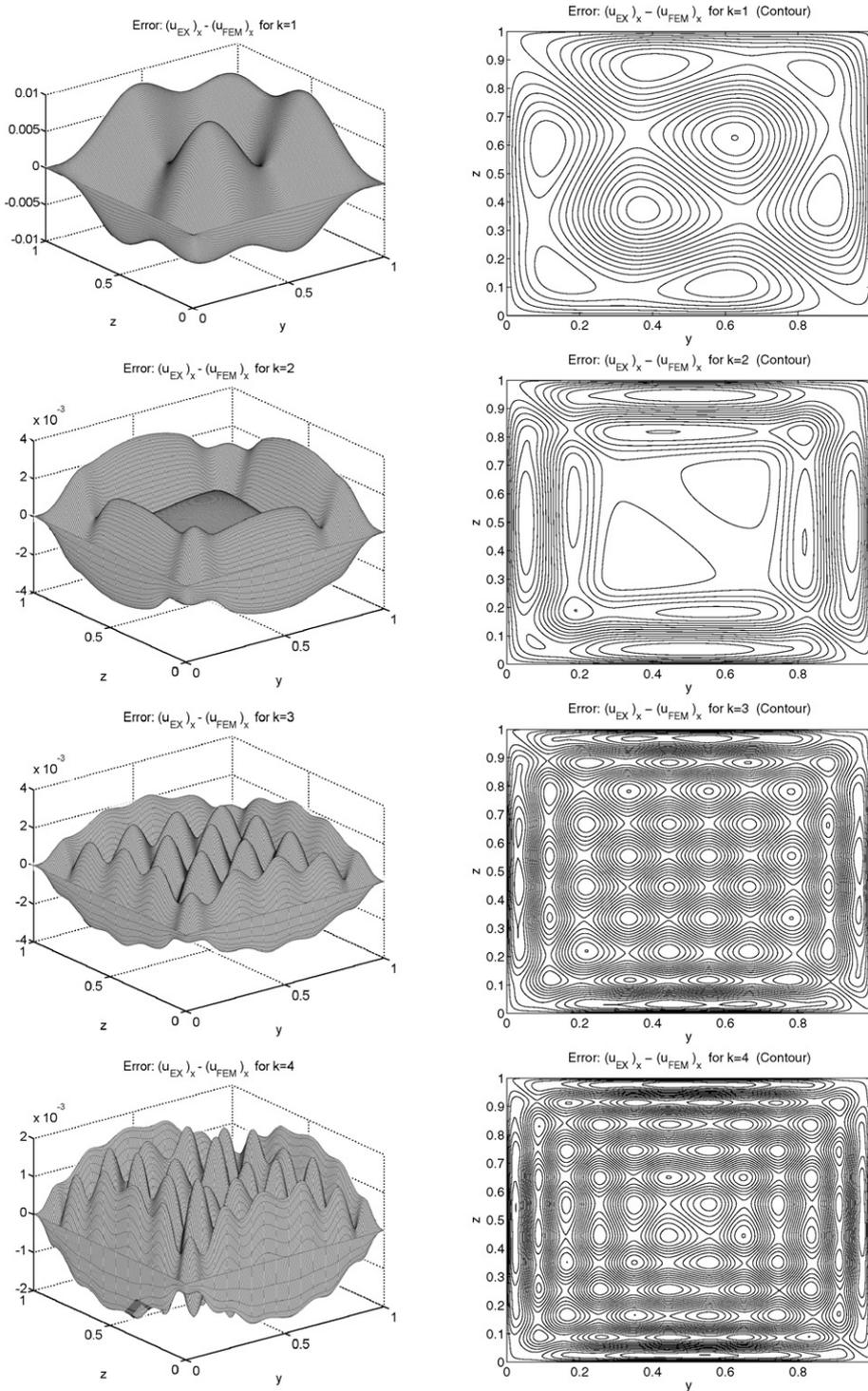
Fig. 8. Error of the first partial with respect to x on interface with $(m, n) = (2k, 3k)$ for $k = 1$–4 (h-version).

numerical experiment. As before, let $(m, n) = (4, 6)$, but this time we fix $(p_1, p_2) = (5, 5)$. Now consider the combinations $(q_1, q_2, r) \in \{(0, 0, 0), (1, 1, 1), \ldots, (5, 5, 5)\}$. The results are shown in Table 4.4. Increasing the degree of the discretized normal and trace spaces forces the error to decrease as expected.

Now we simulate the conforming method by setting $m = n = 5$. With matching meshes from both domains at the interface, we should be able to obtain nice convergence irrespective of the order on the normal and trace spaces. Let us set $(q_1, q_2, r) = (2, 2, 2)$ and let $(p_1, p_2) \in \{(2, 2), (3, 3), (4, 4), (5, 5)\}$. The results are illustrated by Table 4.5. As predicted, the method behaves as a conforming method, giving zero error when $(p_1, p_2) = (5, 5)$.

In conclusion of this section, we compare the exact point-wise function and derivative values extracted along the interface $x = 1$ with those obtained by the non-conforming finite element method. We look at a grid of points over the interface. We fix the mesh to be $(m, n) = (4, 6)$, set the polynomial degrees p_i, q_i, r to be all the same $(= k)$ and consider combinations $k = 1$–4. The finite element solution is calculated as the average value at the interface from each approximation space, Λ_i, for $i = 1$, 2. In Fig. 5, we plot the error between the exact solution and the finite element solution for each $k = 1$–4. The corresponding contour plots for each plot are also presented. For each of these figures, the error is calculated as the point-wise difference along a grid of points on the interface. Following this, in Fig. 6 we plot the difference between the first partial derivatives with respect to x of the exact solution and finite element solution. The partial derivatives with respect to y and z behave similarly in convergence and are thus omitted. Next, in Fig. 7, we consider the mesh refinement $(m, n) = (2k, 3k)$ where $k = 1$–4 and we fix $(p_1, p_2, q_1, q_2, r) = (3, 3, 2, 2, 1)$. As before, Fig. 8 has the error plotted for the first partial derivative in the x direction. Noting the range on the vertical axis, each of these figures clearly illustrate that the approximation becomes better as k increases, both in the p-version and the h-version.

5. Conclusion

A computational methodology for coupling solutions over non-matching grids has been presented via the three-field method in three dimensions. An abstract error estimate has been derived for the numerical solution. Our numerical implementation of the technique and the results clearly suggests that the method proposed herein is a reliable technique for solving multi-physics problems.

Acknowledgement

References

[1] M.A. Aminpour, S.L. McClearly, J.B. Ransom, A global/local analysis method for treating details in structural design, Proc. Third NASA Adv. Compos. Technol. Conf. (NASA CP-3178) 1 (2) (1992) 967–986.

[2] F. Ben Belgacem, The mortar finite element method with Lagrange multipliers, Numer. Math. 84 (2) (1999) 173–197.

[3] F. Ben Belgacem, L.K. Chilton, P. Seshaiyer, The *hp* mortar finite element method for the mixed elasticity and stokes problems, Comput. Math. Appl. 46 (2003) 35–55.

[4] F. Ben Belgacem, Y. Maday, The mortar element method for three dimensional finite elements, RAIRO Math. Model. Numer. Anal. 31 (1997) 289–302.

[5] F. Ben Belgacem, P. Seshaiyer, M. Suri, Optimal convergence rates of *hp* mortar finite element methods for second-order elliptic problems, RAIRO Math. Model. Numer. Anal. 34 (2000) 591–608.

[6] C. Bernardi, Y. Maday, A.T. Patera, Domain decomposition by the mortar element method, in: H.G. Kaper, M. Garbey (Eds.), Asymptotic and Numerical Methods for Partial Differential Equations with Critical Parameters, NATO ASI, Kluwer Academic Publishers, 1993, pp. 269–286.

[7] F. Brezzi, L.D. Marini, Macro hybrid elements and domain decomposition methods, in: J.A. Désideri, L. Fezoui, B. Larrouturou, B. Rousselet (Eds.), Optimization et Côntrole, 1993, pp. 89–96.

[8] F. Brezzi, L.D. Marini, A three-field domain decomposition method, in: A. Quarteroni et al. (Eds.), Domain Decomposition Methods in Science and Engineering, Series CONM 157, AMS, Providence, 1994, pp. 27–34.

[9] F. Brezzi, L.D. Marini, Implementation of the stabilized three-field formulation, in: D. Trigiante (Ed.), Recent trends in Numerical Analysis, Advances in the Theory of Computational Mathematics, vol. 3, Nova Science, 2000, pp. 59–70.

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

[11] C. de Boor, A Practical Guide to Splines, Springer, New York, 1978.

[12] C. Farhat, F.X. Roux, A method of finite element tearing and interconnecting and its parallel solution algorithm, Int. J. Numer. Meth. Eng. 32 (6) (1991) 1205–1228.

[13] C.H. Lai, A nonoverlapped domain decomposition for a class of convection–diffusion problems, Appl. Math. Model. 16 (12) (1992) 638–644.

[14] P. Seshaiyer, Stability and convergence of non-conforming *hp* finite element methods, Comput. Math. Appl. 46 (2003) 165–182.
[15] P. Seshaiyer, M. Suri, Uniform *hp* convergence results for the mortar finite element method, Math. Comput. 69 (2000) 521–546.
[16] P. Seshaiyer, M. Suri, *hp* submeshing via non-conforming finite element methods, Comput. Meth. Appl. Mech. Eng. 189 (2000) 1011–1030.
[17] P. Seshaiyer, P. Smith, A non-conforming finite element method for sub-meshing, Appl. Math. Comput. 139 (2003) 85–100.
[18] H. Swann, On the use of Lagrange multipliers in domain decomposition for solving elliptic problems, Math. Comput. 60 (1993) 49–78.
[19] M.G. Trefry, J. Ohman, G.B. Davis, A simple numerical approach for assessing coupled transport processes in partitioning systems, Appl. Math. Model. 25 (6) (2001) 479–498.