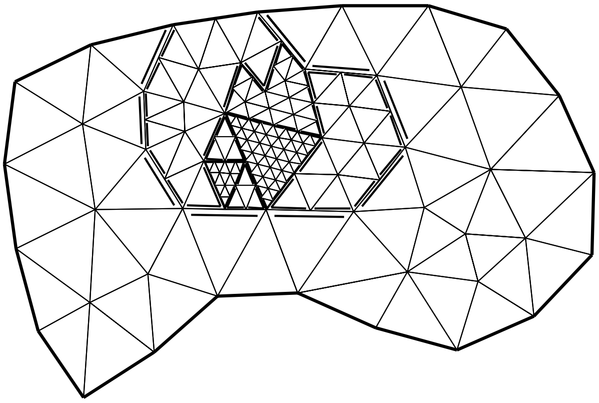
Domain Decomposition Methods in Sciences and Engineering



12th International Conference
on Domain Decomposition Methods

Chiba, JAPAN

Edited by: Tony Chan,
Takashi Kako,
Hideo Kawarada,
Olivier Pironneau,

Published by ddm.org

Domain Decomposition Methods in Sciences and Engineering

12th International Conference
on Domain Decomposition Methods

Chiba, JAPAN

Domain Decomposition Methods in Sciences and Engineering

12th International Conference on Domain Decomposition Methods, Chiba, JAPAN

Edited by

Tony Chan,
UCLA, U.S.A.

Takashi Kako,
The University of Electro-Communications, Japan

Hideo Kawarada,
Chiba University, Japan

Olivier Pironneau,
Université Paris 6, France

Published by ddm.org

Visit our Home Page at http://www.ddm.org

Produced from LATEXmanuscripts submitted by the authors.

Printed and bound in Japan.

ISBN 4-901404-00-8

Preface

This volume represents the Proceedings of the 12th International Conference on Domain Decomposition Methods held at Chiba University, Chiba, Japan, October 25th – 29th, 1999.

Domain Decomposition (DD) has served as an organizing principle for many concepts and methodologies in mathematics, computer science, and computational science and engineering. And also DD will contribute to originate new concepts and methodologies in related fields mentioned above, which will give a clue to understand and solve complex problems existing in our real world. The objective of DD12 was to promote understanding and use of DD for the solution of problems arising in various fields of science and engineering and to promote interaction between researchers throughout the above-mentioned disciplines. These proceedings include invited plenary talks by leading experts in the field from academia, research institutions, and industry, as well as mini-symposia and contributed papers. The papers included are divided into three parts, theory, algorithm and application.

The conference was organized by I. Hagiwara (TIT), T. Ikeda (Ryukoku Univ.), H. Imai (Tokushima Univ.), T. Kako (UEC), H. Kawarada (Chairperson, Chiba Univ.), H. Koshigoe (Chiba Univ.), M. Mori (Kyoto Univ.), M. Nakamura (Nihon Univ.), H. Okamoto (Kyoto Univ.), H. Suito (Chiba Univ.), M. Tabata (Kyushu Univ.), T. Takeda (UEC) and G. Yagawa (Univ. of Tokyo). The conference received support from Chiba Convention Bureau, Chiba University, GAMNI, Inoue Foundation for Science, Iwaki city and the Japanese Ministry of Education.

We wish to thank the conference secretary, Mrs. A. Tonomura, for her hard work towards the success of the conference. We are also grateful to the organizers of mini-symposia for attracting high quality presentations.

Finally, we wish to express sincere gratitude to Dr. Hiroshi Suito (Chiba University, Japan) for technical editing of this proceedings and to Dr. Martin Gander (McGill University, Canada) for preparing LaTeX2e environment.

Tony Chan,
UCLA, U.S.A.

Takashi Kako,
The University of Electro-Communications, Japan

Hideo Kawarada,
Chiba University, Japan

Olivier Pironneau,
Université Paris 6, France

Feburary 2001

Contents

Part I

Theory

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau, ©2001 DDM.org

1. Analysis of a Multigrid Algorithm for the Mortar Finite Element Method

Dietrich Braess[1]

Introduction

The *mortar method* has attracted much interest as a special *domain decomposition* method. It has been analysed in a series of papers (see e. g. [BM97, BMP94, BDW00, Woh99a]) in particular for second order elliptic boundary value problems

$$\begin{aligned} -\operatorname{div} a(x) \operatorname{grad} u(x) &= f(x) && \text{in } \Omega, \\ a(x)\frac{\partial u}{\partial n} &= g(x) && \text{on } \Gamma_N \subset \partial\Omega, \\ u &= 0 && \text{on } \Gamma_D := \partial\Omega \setminus \Gamma_N. \end{aligned} \tag{1}$$

Here $a(x)$ is a (sufficiently smooth) uniformly positive definite matrix in the bounded domain $\Omega \subset \mathbb{R}^d$, Γ_D is a subset of the boundary Γ of Ω, and $\Gamma_N := \Gamma \setminus \Gamma_D$.

Let Ω be decomposed into non-overlapping subdomains $\Omega_k, k = 1, \dots, K$,

$$\bar{\Omega} \;=\; \bigcup_{k=1}^{K} \bar{\Omega}_k, \quad \Omega_k \cap \Omega_l = \emptyset \;\text{ for } k \neq l. \tag{2}$$

Let $H^s(\Omega)$ denote the usual Sobolev spaces endowed with the Sobolev norms $\| \cdot \|_{s,\Omega}$, and $H^1_{0,D}(\Omega)$ be the closure in H^1 of all C^∞-functions vanishing on Γ_D. The natural space associated to the domain decomposition (2) is the product space

$$X_\delta := \{v \in L_2(\Omega) : v|_{\Omega_k} \in H^1(\Omega_k),\, k = 1, \dots, K,\; v|_{\Gamma_D} = 0\}, \tag{3}$$

endowed with the (broken) norm

$$\|v\|_{1,\delta} := \left(\sum_{k=1}^{K} \|v\|^2_{1,\Omega_k} \right)^{1/2}. \tag{4}$$

The space $H^1_{0,D}(\Omega)$ is determined as a subspace of X_δ by appropriate linear constraints. Corresponding discretizations lead to *saddle point problems.* In this paper we present a multigrid method for the *efficient* solution of such *indefinite* systems of equations. According to standard multigrid convergence theory the main tasks are to establish appropriate approximation properties in terms of *direct estimates* as well as to design suitable *smoothing procedures* which give rise to corresponding *inverse estimates.*

The discretization error of the mortar finite elements can be analyzed either by the theory of nonconforming elements and the lemma of Berger, Scott, and Strang

[1] Faculty of Mathematics, Ruhr-University, 44780 Bochum, Germany, braess@num.ruhr-uni-bochum.de

(Strang's second lemma), or by the theory of saddle point problems. Up to now most investigations have used the first approach. It has the advantage that the analysis can be performed with standard Sobolev spaces.

On the other hand, the framework of mixed methods is more appropriate when the computions are performed for the saddle point formulation and fast solvers are to be developed. It seems to be necessary to use mesh-dependent norms if Brezzi's theory is applied. Ellipticity of the variational form, boundedness of the functional in the definition of the constraints, and the inf-sup condition have to be guaranted. We will follow this scheme.

We note that there is also an alternative which can be found in [BB99] and [Woh99a]. The finite element spaces for the direct variables and the Lagrange multipliers need not be balanced so strictly if the error estimates are derived in a two-stage process. First, the direct variables are treated as nonconforming elements. Having an error estimate for them, only the inf-sup condition and no ellipticity assumption is required when the error of the Lagrange multipliers are treated; cf. Remark 1.

There is a correspondence between all the approaches. Roughly speaking, the terms in the formula of the lemma of Berger, Scott, and Strang are obtained by arguments which are refound in the analysis of the mixed method and vice versa; there are, however, some tiny but very sophisticated differences. *Although we admit that the finite element functions are not continuous at the cross points, the subset of the functions without jumps at cross points is responsible for the stability of the mortar elements.*

A suitable smoothing procedure for the multigrid algorithm that is consistent with the approximation properties above is obtained by a method known from the Stokes problem. The paper concludes with a numerical example.

The Continuous Problem

For convenience, we assume that the domain $\Omega \subset \mathbb{R}^d$ and the subdomains Ω_k in (2) are polygonal. If Ω_k and Ω_l share a common interface, we set $\bar{\Gamma}_{kl} := \bar{\Omega}_k \cap \bar{\Omega}_l$. The interior faces form the *skeleton*

$$\mathcal{S} := \bigcup_{k,l} \Gamma_{kl}. \tag{5}$$

Γ_{kl}, Γ_N, and Γ_D will always be assumed to be the union of polygonal subsets of the boundaries of the Ω_k. Often such a decomposition is called *geometrically conforming.*

In order to characterize $H^1_{0,D}(\Omega)$ as a subspace of X_δ, recall that for any (sufficiently regular) manifold Γ the Sobolev spaces $H^s(\Gamma)$ can be defined by their intrinsic norms (see [LM72, Section 7.3]), or alternatively, when Γ is part of a boundary, as a *trace space.* In fact, whenever $s - 1/2$ is *not* an integer,

$$\|v\|_{s-1/2,\Gamma} := \inf_{w \in H^s(\Omega), w|_\Gamma = v} \|w\|_{s,\Omega}$$

is an equivalent norm for $H^{s-1/2}(\partial\Omega)$. Moreover, if Γ' is a smooth subset of Γ, $H^s_{00}(\Gamma')$ consists of those elements $v \in H^s(\Gamma')$ whose extension $\tilde{v}$ of v by zero to all of Γ belongs to $H^s(\Gamma)$, cf. [LM72, p. 66], in particular,

$$\begin{array}{rcl} H_{00}^{1/2}(\Gamma_{kl}) & = & \{v \in H^{1/2}(\Gamma_{kl}) : \tilde{v} \in H^{1/2}(\partial\Omega_k),\ \tilde{v}|_{\Gamma_{kl}} = v;\ \tilde{v} = 0 \text{ on } \partial\Omega_k\backslash\Gamma_{kl}\}, \\ \|v\|_{H_{00}^{1/2}(\Gamma_{kl})} & := & \|\tilde{v}\|_{1/2,\partial\Omega_k}. \end{array} \qquad (6)$$

We note that $H_{00}^{1/2}(\Gamma_{kl})$ is an *interpolation space* between $L_2(\Gamma_{kl})$ and $H_0^1(\Gamma_{kl})$

$$H_{00}^{1/2}(\Gamma_{kl}) = [H_0^1(\Gamma_{kl}), L_2(\Gamma_{kl})]_{1/2},$$

while

$$H^{1/2}(\Gamma_{kl}) = [H^1(\Gamma_{kl}), L_2(\Gamma_{kl})]_{1/2}.$$

This can be realized, e. g., by the K-method [LM72, pp. 64–66, pp. 98–99].

It is appropriate to characterize $H_{0,D}^1(\Omega)$ as a subspace of

$$X_{00} := \{v \in X_\delta : [v]\ |_{\Gamma_{kl}} \in H_{00}^{1/2}(\Gamma_{kl})\ \forall\,\Gamma_{kl} \subset \mathcal{S}\}, \qquad (7)$$

endowed with the norm

$$\|v\|_X^2 := \sum_k \|v\|_{1,\Omega_k}^2 + \sum_{\Gamma_{kl}\subset\mathcal{S}} \|[v]\|_{H_{00}^{1/2}(\Gamma_{kl})}^2. \qquad (8)$$

The trace terms in (8) arise from the fact that X_{00} is a proper subspace of X_δ, and they motivate our later treatment of the finite element discretization. Specifically we have

$$H_{0,D}^1(\Omega) = \{v \in X_{00} : (\mu, [v])_{0,\Gamma_{kl}} = 0\ \ \forall\mu \in H_{00}^{-1/2}(\Gamma_{kl}),\ \Gamma_{kl} \subset \mathcal{S}\}. \qquad (9)$$

Here and in the sequel we write $H_{00}^{-1/2}$ and $H^{-1/2}$ for the dual of $H_{00}^{1/2}$ and $H^{1/2}$, respectively.

We now turn the problem (1) into a weak form based on the above characterization of $H_{0,D}^1(\Omega)$. Let

$$a(u,v) := \sum_k \int_{\Omega_k} (a(x)\nabla u(x)) \cdot \nabla v(x)dx, \qquad (10)$$

$$b(v,\mu) := \sum_{\Gamma_{kl}\subset\mathcal{S}} (\mu, [v])_{0,\Gamma_{kl}}. \qquad (11)$$

Setting

$$M := \prod_{\Gamma_{kl}\subset\mathcal{S}} H_{00}^{-1/2}(\Gamma_{kl}),$$

we consider the variational problem: *find* $(u,\lambda) \in X_{00} \times M$ *such that*

$$\begin{array}{rcll} a(u,v) + b(v,\lambda) & = & (f,v)_{0,\Omega} + (g,v)_{0,\Gamma_N}, & v \in X_{00}, \\ b(u,\mu) & = & 0, & \mu \in M. \end{array} \qquad (12)$$

From the definition of the trace spaces it follows that the operator $B : X_{00} \to \prod_{\Gamma_{kl}} H_{00}^{1/2}(\Gamma_{kl})$, $v \mapsto Bv$ defined by $(Bv,\mu)_{0,\mathcal{S}} = \sum_{\Gamma_{kl}\subset\mathcal{S}}(\mu,[v])_{0,\Gamma_{kl}}$ for any $\mu \in M$, is bounded.

Moreover, the saddle point problem (12) satisfies the *inf-sup condition.* A straight forward proof can be found in [BDW00]. The crucial point is that the jump on Γ_{kl} belongs to $H_{00}^{1/2}(\Gamma_{kl})$, and it can be extended without interference to other parts of the skeleton.

Furthermore, we know from (9) that $H^1_{0,D}(\Omega) = V := \ker B$. Since $\|v\|_X = \|v\|_{1,\Omega}$ for $v \in H^1_{0,D}$, the bilinear form $a(\cdot,\cdot)$ is V-elliptic, i.e., elliptic on the kernel of B.

The discrete problem

In the discussion of the finite element discretization of (12), we will restrict ourselves to the bivariate case, $d = 2$. For each subdomain Ω_k we choose a family of (conforming) triangulations $\mathcal{T}_{k,h}$ independently of the neighboring subdomains; i.e., the nodes in $\mathcal{T}_{k,h}$ that belong to Γ_{kl} need not match the nodes of $\mathcal{T}_{l,h}$. The corresponding spaces of piecewise linear finite elements on $\mathcal{T}_{k,h}$ are denoted by $S_h(\mathcal{T}_{k,h})$. Following [BB99, BM97, BMP94] we set

$$X_h := X_\delta \cap \prod_{k=1}^{K} S_h(\mathcal{T}_{k,h}), \tag{13}$$

i.e., the functions in X_h are not required to be continuous at the cross-points of the polygonal subdomains Ω_k and $X_h \not\subset X_{00}$. We associate with each interface Γ_{kl} the *nonmortar side* which, by the usual convention, is Ω_k while Ω_l is the *mortar side.* Let $M_{kl,h}$ be the space of all continuous piecewise linear functions on Γ_{kl} on that partition induced by the triangulation $\mathcal{T}_{k,h}$ on the nonmortar side, under the additional constraint that the elements in $M_{kl,h}$ are *constant* on the two intervals containing the end points of Γ_{kl}. Thus the dimension of $M_{kl,h}$ agrees with the dimension of $\tilde{T}_{kl,h} := S_h(\mathcal{T}_{k,h}) \cap H_0^1(\Gamma_{kl}) \subseteq H_{00}^{1/2}(\Gamma_{kl})$. The space of discrete multipliers is defined as

$$M_h := \prod_{\Gamma_{kl}\subset\mathcal{S}} M_{kl,h}. \tag{14}$$

The kernel of the restriction operator is

$$V_h := \{v_h \in X_h : b(v_h,\mu_h) = 0 \ \text{ for } \ \mu_h \in M_h\}. \tag{15}$$

As already anounced in the introduction we will employ mesh-dependent norms as in [AT95, Woh99b]. Setting

$$\|w\|_{1/2,h,\Gamma_{kl}} := h^{-1/2}\|w\|_{0,\Gamma_{kl}},$$

let

$$\begin{aligned} \|v_h\|^2_{1,h} \quad &:= \quad \|v_h\|^2_{1,\delta} + \sum_{\Gamma_{kl}\subset\mathcal{S}} \|[v_h]\|^2_{1/2,h,\Gamma_{kl}} \\ &= \quad \|v_h\|^2_{1,\delta} + \sum_{\Gamma_{kl}\subset\mathcal{S}} h^{-1}\|[v_h]\|^2_{0,\Gamma_{kl}}\,, \end{aligned} \tag{16}$$

$$\|\mu\|^2_{-1/2,h} \quad := \quad \sum_{\Gamma_{kl}\subset\mathcal{S}} \|\mu\|^2_{-1/2,h,\Gamma_{kl}} = \sum_{\Gamma_{kl}\subset\mathcal{S}} h\|\mu\|^2_{0,\Gamma_{kl}}. \tag{17}$$

Obviously, (16) corresponds to (8). Whenever a distinction of local mesh sizes matters, the global h in (16)–(17) has to be replaced by the mesh size h_k of the non-mortar side in the summands for Γ_{kl}. In this framework,

$$\begin{array}{lcll} a(u_h, v_h) + b(v_h, \lambda_h) & = & (f, v_h)_{0,\Omega} + (g, v_h)_{0,\Gamma_N}, & v_h \in X_h, \\ b(u_h, \mu_h) & = & 0, & \mu_h \in M_h, \end{array} \tag{18}$$

is a stable discretization of (12).

When verifying this, one crucial point of the analysis is the proof of the inf-sup condition. This is well-known for the saddle point formulation, but the reader may wonder that we find the arguments for the inf-sup condition (often very concealed) also in the analysis by the theory of nonconforming elements. It is done for the following reason. Given $u \in H^2(\Omega_k)$, by the classical theory there is a finite element function $v_h \in X_\delta$ such that $\|u - v_h\|_{1,\delta}$ can be easily estimated. The lemma of Berger, Scott, and Strang, however, requires a good approximation by an element that satisfies the mortaring condition. Now Fortin's theory (see [BF91] or [Bra97, p. 130]) yields this property whenever the inf-sup condition holds.

There is one more point that is found in all treatments of mortar elements which we know. Although the analysis in the papers aim at different norms (the usual Sobolev norms or mesh-dependent norms), they start with an inf-sup condition for the L_2 inner product on the skeleton. We will exemplify a simple proof. Here the inf-sup condition is stated in terms of a projection operator.

To this end we consider the trace space on an interface Γ_{kl} and let

$$\xi_0 < \xi_1 < \ldots < \xi_{p-1} < \xi_p$$

be a partition of the interval $[\xi_0, \xi_p]$ which represents Γ_{kl}. Motivated by the setting (14) of $\tilde{T}_{kl,h}$ and $M_{kl,h}$ we consider two subspaces of the space of continuous piecewise linear functions on $[\xi_0, \xi_p]$. Let $\tilde{T}_{kl,h}$ be the subspace of those functions that vanish at the endpoints ξ_0 and ξ_p, and let $M_{kl,h}$ be the subspace of those functions that are constant on the first and on the last interval. So $\tilde{T}_{kl,h}$ and $M_{kl,h}$ have the same dimension $p-1$.

Lemma 1 *The projectors $Q_h : L_2[\xi_0, \xi_p] \to \tilde{T}_{kl,h}$ defined by*

$$(Q_h f, v)_0 = (f, v)_0 \quad \textit{for } v \in M_{kl,h}, \tag{19}$$

are uniformly bounded in L_2, specifically

$$\|Q_h f\|_0 \le \frac{4}{3}\|f\|_0 \quad \textit{for } f \in L_2[\xi_0, \xi_p]. \tag{20}$$

Proof: For $u_h := Q_h f \in \tilde{T}_{kl,h}$ let $v_h \in M_{kl,h}$ be defined by $v_h(\xi_i) = u_h(\xi_i)$, $i = 1, \ldots, \xi_{p-1}$. The two functions are determined by these $p-1$ values. Thus u_h and v_h agree on $[\xi_1, \xi_{p-1}]$, and $\int_{\xi_1}^{\xi_{p-1}} u_h v_h dx = \frac{1}{2}\int_{\xi_1}^{\xi_{p-1}} (u_h^2 + v_h^2) dx$. On the other hand, one obtains for the first (and last) interval

$$\int_{\xi_0}^{\xi_1} u_h v_h dx = \frac{1}{2} D, \qquad \int_{\xi_0}^{\xi_1} u_h^2 dx = \frac{1}{3} D, \qquad \int_{\xi_0}^{\xi_1} v_h^2 dx = D,$$

where $D := (\xi_1 - \xi_0)u_h(\xi_1)^2$. Hence,

$$\int_{\xi_0}^{\xi_1} u_h v_h dx = \frac{3}{8} \int_{\xi_0}^{\xi_1} (u_h^2 + v_h^2) dx.$$

Summing over all intervals and using Young's inequality yields

$$\|f\|_0 \|v_h\|_0 \geq (f, v_h)_0 = (u_h, v_h)_0 \geq \frac{3}{8} \left(\|u_h\|_0^2 + \|v_h\|_0^2 \right) \geq \frac{3}{4} \|u_h\|_0 \|v_h\|_0, \tag{21}$$

which proves (20). ■

Let $\mu_h \in M_h$ and Γ_{kl} be an interface. It follows from the lemma that $(\mu_h, w_{kl})_{0,\Gamma_{kl}}$ is large if $w_{kl} := Q_{h,kl}\, \mu_h$. Specifically we conclude that

$$\inf_{\mu_h \in M_h} \sup_{v_h \in X_h} \frac{b(v_h, \mu_h)}{\|v_h\|_{0,\mathcal{S}}\, \|\mu_h\|_{0,\mathcal{S}}} \geq \frac{3}{4}.$$

The proof of the Brezzi condition for the correct norms usually proceeds in a standard way. Interpolation theory yields an inverse estimate

$$\|w_{kl}\|_{H_{00}^{1/2}} \leq c h^{-1/2} \|w_{kl}\|_{0,\Gamma_{kl}} = c \|w_{kl}\|_{1/2,h,\Gamma_{kl}}. \tag{22}$$

There is an extension v such that

$$[v] = w_{kl} \quad \text{on } \Gamma_{kl}\,,$$

and the $\|\cdot\|_1$-norm of the extension is bounded by the $H_{00}^{1/2}$ norm above. Thus the same construction is good for the proof of the Brezzi condition for the mesh-dependent norms or for the Sobolev norms.

Theorem 1 *Assume that the triangulation in each subdomain Ω_k is quasiuniform. The discretizations (18) based on the spaces X_h, M_h defined by (13) and (14), respectively, satisfy the LBB-condition, i.e., there exists some $\beta > 0$ such that*

$$\inf_{\mu_h \in M_h} \sup_{v_h \in X_h} \frac{b(v_h, \mu_h)}{\|v_h\|_{1,h} \|\mu_h\|_{-1/2,h}} \geq \beta, \tag{23}$$

and

$$\inf_{\mu_h \in M_h} \sup_{v_h \in X_h} \frac{b(v_h, \mu_h)}{\|v_h\|_X\, \|\mu_h\|_{H_{00}^{-1/2}}} \geq \beta \tag{24}$$

holds uniformly in h.

We want to stress one point. Since we admit that the finite element functions in X_h can be discontinuous at the cross points, their jumps on the interface Γ_{kl} are only in $H^{1/2}(\Gamma_{kl})$. Nevertheless, the construction for the proof of the inf-sup condition yielded finite element functions with jumps in the subspaces $H_{00}^{1/2}(\Gamma_{kl})$, and we conclude that the subspace of finite elements with this property is thick enough. Therefore it is natural that the Lagrange multipliers are equipped with the norms $\|\mu_h\|_{H_{00}^{-1/2}(\Gamma_{kl})}$ in those investigations in which norms of the classical Sobolev spaces rather than mesh-dependent norms are preferred.

After the inf-sup condition has been established, only approximation properties are required for the proof of the error estimate. Assume that the problem is H^2-regular, i.e., $u \in H^2$. Let $w_h \in X_h$ be finite element function with $\|u - w_h\|_{1,\delta} \le ch\|u\|_2$ that need not satisfy the mortar condition. Similarly, we have $\|\frac{\partial u}{\partial n} - \mu_h\|_{0,\Gamma_{kl}} \le ch^{1/2}\|u\|_2$, for some $\mu_h \in M_h$ and all Γ_{kl}, (and this term appear also in the usual bounds of the consistency error of the second Strang lemma). The regularity assumption and a density argument assert that the first equation in (12) holds for all $v \in X_{00} + X_h$. Hence,

$$\begin{array}{lcll} a(u_h - w_h, v_h) + b(v_h, \lambda_h - \mu_h) & = & \langle l, v\rangle & \forall v_h \in X_h, \\ b(u_h - w_h, \mu) & = & 0, & \forall \mu \in M_h, \end{array} \tag{25}$$

where $\langle l, v\rangle := a(u - w_h, v) + b(v, \lambda - \mu_h)$. By construction $|\langle l, v\rangle| \le ch\,\|u\|_2\,\|v\|_{1,h}$. From this bound and the stability of (25) we obtain the error estimate

$$\|u - u_h\|_{1,h} + \|\lambda - \lambda_h\|_{-1/2,h} \le ch\|u\|_2 \tag{26}$$

and by a duality argument

$$\|u - u_h\|_0 + h\|\lambda - \lambda_h\|_{-1/2,h} \le ch^2\|u\|_2. \tag{27}$$

For details the reader is refered to [BDW00].

Remark 1 *We have provided the well-known arguments in the derivation of the error estimates since we want to be more specific about the remark at the end of the introduction.*

In establishing (25) we have used ellipticity of $a(\cdot,\cdot)$, boundedness of $b(\cdot,\cdot)$, and the inf-sup condition. On the other hand, if a bound for $\|u - u_h\|_{1,\delta}$ has been determined elsewhere, following [BB99, Woh99a] the first equation in (25) may be rewritten

$$b(v_h, \lambda_h - \mu_h) = b(v_h, \lambda - \mu_h) - a(u_h - u, v_h) \quad \forall v_h \in X_h. \tag{28}$$

From (24) we know that we obtain $\|\lambda_h - \mu_h\|_{H_{00}^{1/2}} \le \beta^{-1} b(v_h, \lambda_h - \mu_h)/\|v_h\|_X$ with an appropriate test function v_h. Specifically, the right test function has its jumps on the interfaces in $H_{00}^{1/2}$, and it is not an obstacle that $b(\cdot,\cdot)$ is not bounded on $H^{1/2} \times H_{00}^{-1/2}$. After applying the triangle inequality the error of the Lagrange multipliers is established in the $H_{00}^{-1/2}$ norm. This technique circumvents the fact that the bilinear form b is not bounded on $H^{1/2} \times H_{00}^{-1/2}$. – The unboundedness is an obstacle for the direct application of Brezzi's theory.

Finally, we note that recently other finite elements for the Lagrange multipliers have been suggested. Computations are easier if they are obtained from a dual basis [Woh99c].

Multigrid Convergence Analysis

The saddle point problem (18) gives rise to a linear system of the form

$$\begin{pmatrix} A & B^T \\ B & \end{pmatrix} \begin{pmatrix} u_h \\ \lambda_h \end{pmatrix} = \begin{pmatrix} f \\ 0 \end{pmatrix}, \tag{29}$$

where the dimension of the vectors coincides with the dimension of the finite element spaces X_h and M_h, respectively. For convenience, the same symbol is taken for the finite element functions and their vector representations, and the index h is suppressed whenever no confusion is possible.

The finite element basis functions are assumed to be normalized such that the Euclidean norm of the vectors $\|\cdot\|_{\ell_2}$ is equivalent to the L_2-norm of the functions, i.e.

$$\|v_h\|_{\ell_2} \approx \|v_h\|_{0,\Omega} \quad \text{for } v_h \in X_h. \tag{30}$$

When the equations (29) are solved by a multigrid algorithm, the design of the smoothing procedure is the crucial point. Motivated by [BS97] our smoothing procedure will be based on the following concept. Let C be a preconditioner for A which, in particular, is normalized so that

$$v^T A v \le v^T C v, \quad v \in X_h, \tag{31}$$

and for which the linear system

$$\begin{pmatrix} C & B^T \\ B & \end{pmatrix} \begin{pmatrix} v \\ \mu \end{pmatrix} = \begin{pmatrix} d \\ e \end{pmatrix} \tag{32}$$

is easily solvable. In actual computations the vectors v, μ are obtained by implementing $S\mu = BC^{-1}d - e, v = C^{-1}(d - B^T\mu)$, where $S := BC^{-1}B^T$ is the *Schur complement* of C in (32).

Then the iteration that will serve as a smoother in our multigrid scheme has the form

$$\begin{aligned} \begin{pmatrix} u_h^{j+1} \\ \lambda_h^{j+1} \end{pmatrix} &:= \begin{pmatrix} u_h^j \\ \lambda_h^j \end{pmatrix} - \begin{pmatrix} C & B^T \\ B & \end{pmatrix}^{-1} \left\{ \begin{pmatrix} A & B^T \\ B & \end{pmatrix} \begin{pmatrix} u_h^j \\ \lambda_h^j \end{pmatrix} - \begin{pmatrix} f \\ 0 \end{pmatrix} \right\} && (33) \\ &= \begin{pmatrix} u_h^j \\ 0 \end{pmatrix} - \begin{pmatrix} C & B^T \\ B & \end{pmatrix}^{-1} \begin{pmatrix} Au_h^j - f \\ Bu_h^j \end{pmatrix}, && (34) \end{aligned}$$

where superscripts will denote iteration indices. It is important to note that u_h^{j+1} always satisfies the constraint, i.e.,

$$Bu_h^{j+1} = 0, \tag{35}$$

see [BS97]. Specifically the implementations are based on (33) in order to have auxiliary problems with small (correction) vectors, while the representation (34) shows that the next iterate is independent of the old Lagrange multiplier λ_h^j.

Now we assume that the reader is familiar with the general concept of multigrid algorithms [Hac85] and knows some simple application. This is sufficient since the finite element spaces $X_h \subset X_\delta$ and $M_h \subset M$ are nested and the coarse grid correction of the multigrid scheme can be performed in the standard manner, see e.g. [BS97] or [Hac85, p. 235].

As usually, the analysis of the multigrid method will be based on two different norms. The fine topology will be defined by the norm

$$|||v_h, \mu_h|||_2 := \|Av_h + B^T \mu_h\|_{\ell_2}, \tag{36}$$

i.e., by a discrete analogon of the H^2-norm, and the coarse one by the L_2-norm

$$\|v_h\|_{0,\Omega}.$$

The latter expression is independent of the Lagrange multiplier since the iteration (34) is independent of the multiplier in the previous step. We recall that $\lambda_{\max}(A) = O(h^{-2})$.

Smoothing property: Assume that $\lambda_{\max}(A) \leq \alpha \leq c\lambda_{\max}(A)$. If m smoothing steps of the relaxation (34) with $C := \alpha I$ are performed, then

$$|||u_h^m - u_h, \lambda_h^m - \lambda_h|||_2 \leq \frac{ch^{-2}}{m} \|u_h^0 - u_h\|_{0,\Omega}. \tag{37}$$

Approximation property: For the coarse grid correction u_{2h} one has

$$\|u_h - u_{2h}\|_{0,\Omega} \leq ch^2 |||u_h, \lambda_h|||_2. \tag{38}$$

The proof of the two properties are now quite standard. The verification of the smoothing property is performed by purely algebraic manipulations [BDW00, BS97]. The approximation property looks very much like the L_2-error estimate (27). Indeed, it is derived from the latter by a duality argument; cf. [BDW00] or [Bra97, Lemma V.2.8].

Recently, a version was implemented as a *cascadic multigrid* algorithm; see [BDL99]. In that context it is shown that the Lagrange multipliers must be treated in a different way than the u-variables if the iteration (33) is built into a conjugate gradient method.

Numerical Example

We report on one of the examples in [BDW00] with big jumps of coefficients and several cross points. The equation (1.1) is considered with scalar diffusion coefficients that are constant on each subdomain. In Figure 1 large bricks are separated by thin channels. Fixing the diffusion constant for the bricks to $a_0 = 1$, we test the cases where the channels have higher or lower permeability ($a_1 = 10^6$ or $a_1 = 10^{-6}$, resp.). We perform the cg-method with V(1,1)-cycle and two inner iterations. The convergence rates are stable if the mortar side is on the side with the smaller diffusion constant and large step size, resp.; otherwise the method may fail. The results in Figure 1 for the case $a_1 = 10^6$ show clearly that the diffusion is faster in the small channels.

level	elements	$a_0 = 1$, $a_1 = 1$	$a_0 = 1$, $a_1 = 10^6$	$a_0 = 1$, $a_1 = 10^{-6}$
3	6784	0.21	0.12	0.08
4	27136	0.21	0.14	0.07
5	108544	0.21	0.14	0.08
inner iterations		1–2	1–2	1–2

Table 1: Convergence for the example with several cross-points for MG with V(1,1)-cycle

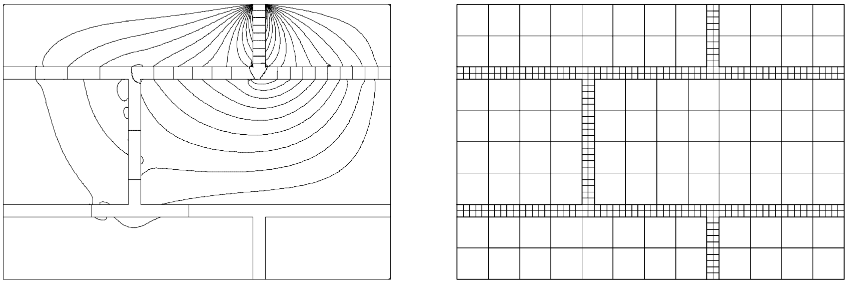


Figure 1: Example with several cross-points

References

[AT95]A. Agouzal and J.-P. Thomas. Une méthode d'élement finis hybridesen décomposition de domaines. *RAIRO M^2AN*, 29:749–764, 1995.

[BB99]Faker Ben Belgacem. The mortar finite element method with Lagrange multipliers. *Numer. Math.*, 84(2):173–197, 1999.

[BDL99]D. Braess, P. Deuflhard, and K. Lipnikov. A cascadic multigrid algorithm for mortar elements. Technical report, Konrad-Zuse-Zentrum, Berlin, 1999.

[BDW00]D. Braess, W. Dahmen, and C. Wieners. A multigrid algorithm for the mortar finite element method. *SIAM J. Numer. Anal.*, 37:48–69, 2000.

[BF91]F. Brezzi and M. Fortin. *Mixed and Hybrid Finite Element Methods.* Springer-Verlag, New York – Berlin – Heidelberg, 1991.

[BM97]Faker Ben Belgacem and Yvon Maday. The mortar element method for three dimensional finite elements. *RAIRO Mathematical Modelling and Numerical Analysis*, 31(2):289–302, 1997.

[BMP94]Christine Bernardi, Yvon Maday, and Anthony T. Patera. A new non conforming approach to domain decomposition: The mortar element method. In Haim Brezis and Jacques-Louis Lions, editors, *Collège de France Seminar.* Pitman, 1994. This paper appeared as a technical report about five years earlier.

[Bra97]D. Braess. *Finite Elements: Theory, Fast Solvers, and Applications in Solid Mechanics.* Cambridge University Press, Cambridge, 1997.

[BS97]D. Braess and R. Sarazin. An efficient smoother for the Stokes problem. *Applied Numerical Mathematics*, 23:3–19, 1997.

[Hac85]W. Hackbusch. *Multi-Grid Methods and Applications.* Springer-Verlag, Berlin

– Heidelberg – New York, 1985.

[LM72]J.L. Lions and E. Magenes. *Non-Homogeneous Boundary Value Problems.* Springer-Verlag, New York – Berlin – Heidelberg, 1972.

[Woh99a]B. Wohlmuth. Discretization methods and iterative solvers based on domain decomposition. Technical report, Habilitation, Department of Mathematics, Augsburg, 1999.

[Woh99b]B. Wohlmuth. Hierarchical a posteriori error estimators for mortar finite element methods with lagrange multipliers. *SIAM J. Numer. Anal.*, 36:1636–1658, 1999.

[Woh99c]B. Wohlmuth. A mortar finite element method using dual spaces for the lagrange multipliers. Technical report, Department of Mathematics, Augsburg, 1999.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau, ©2001 DDM.org

2. Optimized Schwarz Methods

Martin J. Gander [1], Laurence Halpern [2], Frederic Nataf [3]

Introduction

Schwarz methods lead to parallel preconditioners for large linear systems of equations arising in the solution process of partial differential equations [SBG96]. Optimal convergence results for the Schwarz method are known in the sense that the condition number of the preconditioned system is independent of (or only weakly dependent on) the mesh parameter and the number of subdomains. Thus asymptotically Schwarz methods have optimal scalability.

This optimality result contains however constants which remain unknown in the analysis. Thus it does not imply that the current Schwarz methods have optimal performance. It does not guarantee either that Schwarz methods are competitive to other parallel methods. Thus the word ”optimal” can be misleading.

We analyze the performance of the classical Schwarz method for two model problems, Laplace's equation and the Helmholtz equation. Our analysis is performed at the continuous level which seems natural for the Schwarz method since the method itself is defined at the continuous level. Our investigation reveals that the convergence rate of the Schwarz methods depends intrinsicly on the transmission conditions employed between subdomains. The classical transmission conditions used by Schwarz are Dirichlet transmission conditions [Sch70]. These transmission conditions lead to convergence rates which are not uniform with respect to frequency: high frequency components converge rapidly whereas low frequency components converge only slowly. Motivated by the analysis of Overlapping Schwarz Waveform Relaxation in [GHN99] we construct optimal transmission conditions for the Laplace and Helmholtz equation in two dimensions. These conditions are global in nature and thus not ideal for implementations. We therefore introduce local approximations of the optimal conditions and optimize them for performance, which leads to the optimized Schwarz methods.

Other people have looked at different transmission conditions before. Generalized Schwarz splittings with Robin transmission conditions have been analyzed by Tang [Tan92] and led to an over-determined Schwarz algorithm in [ST96]. The main difficulty remaining in this approach is the determination of the relaxation parameter in the Robin conditions, like for SOR methods. For Helmholtz problems radiation conditions for overlapping Schwarz have been proposed by [CCEW98]. For non-overlapping versions of the Schwarz algorithms Dirichlet transmission conditions are not effective and Lions proposed to use Robin conditions to obtain a convergent algorithm in [Lio90]. Through the work by Charton, Nataf and Rogier [CNR91], Nataf and Rogier [NR95] and Japhet [Jap98] new types of transmission conditions for convection diffu-

[1] Department of Mathematics and Statistics, McGill University, Montreal, Canada. mgander@math.mcgill.ca

[2] Département de Mathématiques, Université Paris XIII, 93430 Villetaneuse and CMAP, Ecole Polytechnique, 91128 Palaiseau, France. halpern@math.univ-paris13.fr

[3] CMAP, Ecole Polytechnique, 91128 Palaiseau, France. nataf@cmap.polytechnique.fr

sion problems have been introduced which are optimal in a physical sense and contain the Robin conditions as a first order approximation. A similar approach was developed for the Helmholtz equation in [DJR92] and [CN98]. An overlapping version for Laplace's equation was analyzed in [EZ98]. The same type of analysis was applied to overlapping Schwarz waveform relaxation algorithms in [GHN99] and led to optimized Schwarz algorithms for evolution problems where one can easily visualize that the optimal transmission conditions are absorbing boundary conditions. The key is that a simple optimization procedure leads to local transmission conditions with optimized performance for the Schwarz algorithm. We derive optimized Schwarz methods for elliptic definite and indefinite problems in this note.

Optimized Schwarz Method for Laplace's Equation

We consider Laplace's equation in the domain $\Omega = \mathbb{R}^2$,

$$\Delta u = f(x,y), \quad x,y \in \Omega, \quad u \text{ bounded at infinity.} \tag{1}$$

We decompose the domain Ω into two overlapping half planes $\Omega_1 = (-\infty, L] \times \mathbb{R}$ and $\Omega_2 = [0,\infty) \times \mathbb{R}$ where $L > 0$ is the overlap parameter. The classical Schwarz method to solve (1) solves iteratively Laplace's equation on Ω_1 and Ω_2 and exchanges Dirichlet values on the interfaces at 0 and L,

$$\begin{array}{rcll} \Delta v^{n+1} & = & f(x,y), & x,y \in \Omega_1, \\ v^{n+1}(L,y) & = & w^n(L,y), & \\ \Delta w^{n+1} & = & f(x,y), & x,y \in \Omega_2, \\ w^{n+1}(L,y) & = & v^n(L,y). & \end{array} \tag{2}$$

To analyze the convergence of the classical Schwarz method, it suffices by linearity to consider the homogeneous problem, $f(x,y) = 0$ in (2), and to analyze convergence to zero.

Fourier Analysis of the Classical Schwarz Method

Our results are based on Fourier analysis. We denote the Fourier transform $\hat{f}(k)$ of $f(x) : \mathbb{R} \longrightarrow \mathbb{R}$ by

$$\hat{f}(k) = \mathcal{F}_x(f)(k) := \int_{-\infty}^{\infty} e^{-ikx} f(x)dx$$

and the inverse Fourier transform of $\hat{f}(k)$ by

$$f(x) = \mathcal{F}_x^{-1}(\hat{f})(x) := \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{ikx} \hat{f}(k)dk.$$

Taking a Fourier transform in y of (2) for $f(x,y) = 0$ we obtain

$$\begin{array}{rcll} \hat{v}_{xx}^{n+1}(x,k) - k^2\hat{v}^{n+1}(x,k) & = & 0, \quad x \in (-\infty, L), k \in \mathbb{R}, & (3) \\ \hat{v}^{n+1}(L,k) & = & \hat{w}^n(L,k), & \\ \hat{w}_{xx}^{n+1}(x,k) - k^2\hat{w}^{n+1}(x,k) & = & 0, \quad x \in (0,\infty), k \in \mathbb{R}, & (4) \\ \hat{w}^{n+1}(0,k) & = & \hat{v}^n(0,k) & \end{array}$$

where a subscript x denotes a partial derivative with respect to x. Solving the ordinary differential equation (4) using the boundedness condition at infinity and inserting the result into the boundary condition of (3) we find the solution of (3) at $x = 0$ to be

$$\hat{v}^{n+1}(0,k) = e^{-2|k|L}\hat{v}^{n-1}(0,k).$$

Similarly we obtain for the solution of (4) at $x = L$

$$\hat{w}^{n+1}(L,k) = e^{-2|k|L}\hat{w}^{n-1}(L,k).$$

Defining the convergence rate

$$\rho(k,L) := e^{-2|k|L} \tag{5}$$

we see that the classical Schwarz method converges for all $k \neq 0$ if there is overlap, $L > 0$. The convergence rate is linear and depends on the size of the overlap L as well as the frequency k. High frequency components converge fast, whereas low frequency components converge only slowly. Note that for $|k| \to 0$ the convergence rate ρ tends to 1.

Optimal Transmission Conditions

The preceding analysis shows that the Schwarz method is slowed down by the low frequency components. They are dictating the convergence rate and thus the performance of the Schwarz method. For better performance, one would like to improve the convergence rate for the low frequency components. This can be achieved by changing the transmission conditions to become more transparent for low frequency components. Following the approach in [GHN99] for evolution problems, we introduce new transmission conditions into the classical Schwarz method (2). Instead of using Dirichlet transmission conditions, we impose at the artificial boundaries

$$\begin{array}{rcl} v_x^{n+1}(L,y) + \Lambda_v(v^{n+1}(L,y)) &=& w_x^n(L,y) + \Lambda_v(w^n(L,y)) \\ w_x^{n+1}(0,y) + \Lambda_w(w^{n+1}(0,y)) &=& v_x^n(0,y) + \Lambda_w(v^n(0,y)), \end{array} \tag{6}$$

where the linear operators Λ_v and Λ_w are degrees of freedom we can use to optimize the performance of the algorithm. Note that the Schwarz method itself remains the same, only the transmission conditions have been changed. We have the following

Theorem 1 (Optimal Convergence) *Choosing Λ_v to have the symbol $\lambda_v(k) := |k|$ and Λ_w to have the symbol $\lambda_w(k) := -|k|$ the Schwarz method with transmission conditions (6) converges in two iterations independently of the overlap $L \geq 0$.*

Proof Applying a Fourier transform in y to (3), (4) with transmission conditions (6) we obtain

$$\begin{array}{rcll} \hat{v}_{xx}^{n+1}(x,k) - k^2\hat{v}^{n+1}(x,k) &=& 0, \quad x \in (-\infty, L), k \in \mathbb{R}, & (7) \\ \hat{v}_x^{n+1}(L,k) + \lambda_v(k)\hat{v}^{n+1}(L,k)) &=& \hat{w}_x^n(L,k) + \lambda_v(k)\hat{w}^n(L,k), & \\ \hat{\hat{w}}_{xx}^{n+1}(x,k) - k^2\hat{\hat{w}}^{n+1}(x,k) &=& 0, \quad x \in (0,\infty), k \in \mathbb{R}, & (8) \\ \hat{w}_x^{n+1}(0,k) + \lambda_w(k)\hat{w}^{n+1}(0,k) &=& \hat{v}_x^n(0,k) + \lambda_w(k)\hat{v}^n(0,k). & \end{array}$$

Solving (8) at iteration step n for $\hat{w}^n$ and inserting the result into the transmission conditions of (7) we find for $\hat{v}^{n+1}$ at $x = 0$

$$\hat{v}^{n+1}(0,k) = \rho_l \hat{v}^{n-1}(0,k)$$

and by a similar computation for $\hat{w}^{n+1}$ at $x = L$

$$\hat{w}^{n+1}(L,k) = \rho_l \hat{w}^{n-1}(L,k)$$

where the convergence rate ρ_l is given by

$$\rho_l(k,L) := \frac{-|k| + \lambda_v(k)}{|k| + \lambda_v(k)} \cdot \frac{|k| + \lambda_w(k)}{-|k| + \lambda_w(k)} e^{-2|k|L}. \qquad (9)$$

Hence choosing $\lambda_v(k) := |k|$ and $\lambda_w(k) := -|k|$ the convergence rate vanishes, $\rho_l \equiv 0$ and thus, independently of the initial guess, after two steps of the Schwarz iteration the iterates are zero on $x = 0$ and $x = L$ respectively. To see that they vanish identically, it suffices to note that by the boundedness condition at infinity, $\hat{v}^2(x,k) = Ae^{|k|x}$ and $\hat{w}^2(x,k) = Be^{-|k|x}$ for some constants A and B. But $\hat{v}^2(0,k) = 0$ then implies $A = 0$ and $\hat{w}^2(L,k) = 0$ implies $B = 0$ and the result follows. ■

Note that the new convergence rate (9) still contains the exponential factor like the classical one (5), but the new transmission conditions (6) introduced an additional factor with the degrees of freedom $\lambda_v(k)$ and $\lambda_v(k)$. Theorem 1 shows what the optimal choice is for the transmission conditions in theory. One can show that with this choice and N subdomains in strips the Schwarz algorithm converges in N steps, see [NRdS94]. This is an optimal result since the solution of Laplace's equation in one subdomain depends on the source term f in every other subdomain and when only a local mechanism of communication is employed one has to communicate at least N steps to get the information from the left most subdomain across all the other subdomains to the rightmost subdomain.

However to use the algorithm in practice, one either needs to work in Fourier space or one has to back-transform the optimal transmission conditions to the real space. The inverse Fourier transform of $\lambda_{vw} = \pm|k|$ leads to the optimal transmission operators Λ_{vw} which are non local in y and thus harder to implement. Note that the optimal transmission operators correspond to the Dirichlet to Neumann map at the artificial interfaces and thus the optimal transmission conditions are the absorbing boundary conditions as in the case of the evolution problems [GHN99].

Optimized Local Transmission Conditions

For a real implementation of the Schwarz algorithm, it is desirable to have local transmission conditions. We therefore approximate the nonlocal optimal transmission conditions found in the previous subsection by local ones. Local operators are represented by polynomials in Fourier space and we analyze in the sequel the performance of the zeroth and second order approximation of the optimal transmission conditions,

$$\lambda_{vw} = \pm p \quad \text{or} \quad \lambda_{vw} = \pm(p + qk^2). \qquad (10)$$

The parameters $p, q > 0$ are free parameters and they can be used to optimize the performance of the new Schwarz method which leads to the optimized Schwarz method.

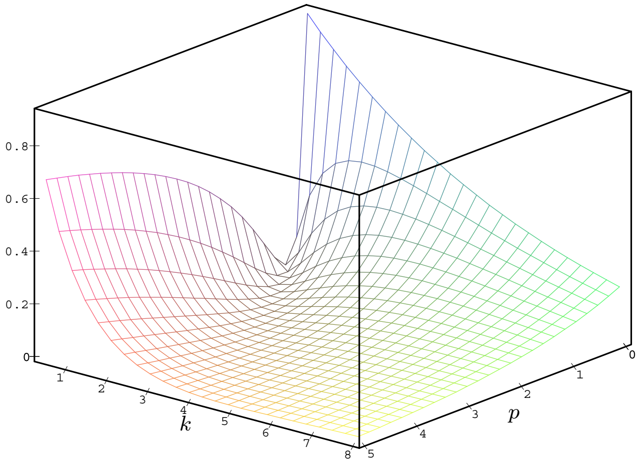


Figure 1: Dependence of the convergence rate on the frequency k and the optimization parameter p for Laplace's equation.

Since real computations are performed on bounded domains and discretized operators, the range of the frequency parameter k is not arbitrary. It is bounded from below by a lowest frequency dependent on the size of the domain in y direction and the boundary conditions imposed, $k^2 > k_{\text{min}}^2$ and from above, k is bounded by the mesh size h in y direction, $k^2 < k_{max}^2 := (\pi/h)^2$. Thus to obtain optimal performance of the Schwarz method, we have to solve the min-max problem

$$\min_{p>0}\left(\max_{k_{\min}<k<k_{\max}} \frac{(|k|-p)^2}{(|k|+p)^2}e^{-2|k|L}\right)$$

in the case of the zeroth order approximation. Figure 1 shows the dependence of the convergence rate on the frequency k and the free parameter p. Note that the convergence rate is symmetric in k and only the part for $k_{\min} < k < k_{\max}$ is shown in the figure. One can clearly identify that for a certain parameter value p the convergence rate will become small for all values of k, $k_{\min} < k < k_{\max}$. For large p however the low frequencies will dominate again the convergence rate and in the limit as p goes to infinity, we recover the classical Schwarz method.

For the second order approximation of the optimal transmission conditions, we find the min-max problem

$$\min_{p,q>0}\left(\max_{k_{\min}<k<k_{\max}} \frac{(|k|-p-qk^2)^2}{(|k|+p+qk^2)^2}e^{-2|k|L}\right).$$

Both min-max problems can be solved analytically and we show in Figure 2 the convergence rates obtained for the classical Schwarz method and the two optimized Schwarz

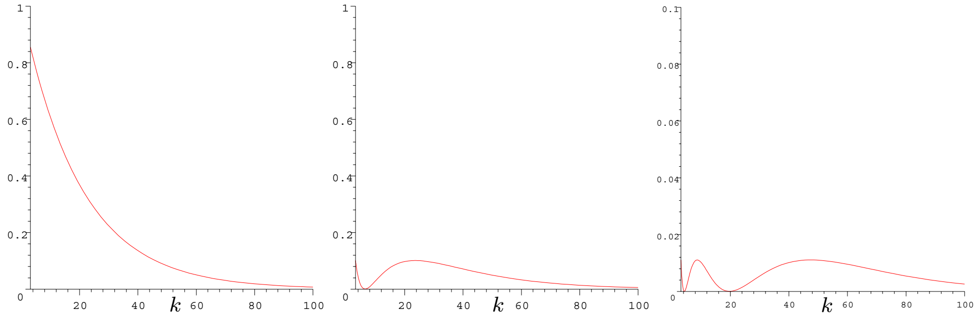


Figure 2: Convergence rates in Fourier space for Laplace's equation. The classical Schwarz method on the left, zeroth order optimized Schwarz method in the middle and second order optimized Schwarz method on the right. Note the *scaling factor of 10* in the right most figure.

methods for the model problem (11) with mesh parameter $h = 1/80$. Note how the zeroth order approximation, which leads to a Robin condition instead of a Dirichlet one in the Schwarz algorithm, reduces the convergence rate already from 0.82 to 0.05 and the second order approximation reduces it further to 0.006. The numerical experiments in the following subsections confirm the enormous improvement of the optimized Schwarz algorithm over the classical one.

Numerical Experiments for Laplace's Equation

We solve Laplace's equation on the rectangular domain $\Omega = [0, 2] \times [0, 1]$,

$$\Delta u = 0, \quad x, y \in \Omega \tag{11}$$

with given Dirichlet boundary conditions. We decompose Ω into two subdomains $\Omega_1 = [0, 1 + \delta] \times [0, 1]$ and $\Omega_2 = [1 - \delta, 2] \times [0, 1]$ and apply the Schwarz algorithm as an iterative solver. Figure 3 shows the performance of the classical Schwarz method compared to the zeroth order optimized one and the second order optimized one for an overlap of $2\delta = 1/40$. Clearly the optimized Schwarz method perform much better than the classical one. The convergence rate improvement due to the new transmission conditions manifests itself in the numerical experiments. While the classical Schwarz method only reduces the error by a few percent in 8 iterations, the zeroth order optimized Schwarz method reduces the error by a factor of 10^5 and the second order optimized Schwarz method reduces the error by a factor of 10^{13}. Note that these contraction rates are comparable to multi-grid, and we have not used a Krylov method yet, just classical Schwarz as an iterative solver.

To accelerate convergence, one usually uses the Schwarz method as a preconditioner, which greatly improves the performance of the classical Schwarz method. Figure 4 shows the decay of the error in the same experiment as above, but now the Schwarz methods are used as preconditioners. Clearly the classical Schwarz method is improved a great deal by the Krylov method, but the optimized Schwarz methods are accelerated as well and still converge much faster than the classical Schwarz method.

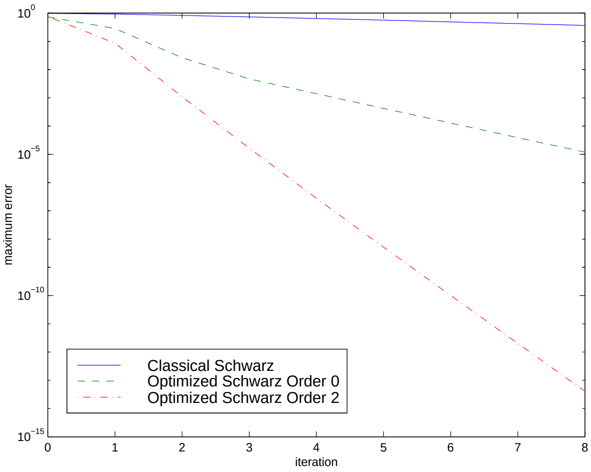


Figure 3: The performance of the optimized Schwarz methods for Laplace's equation compared to the classical Schwarz method as an iterative solver.

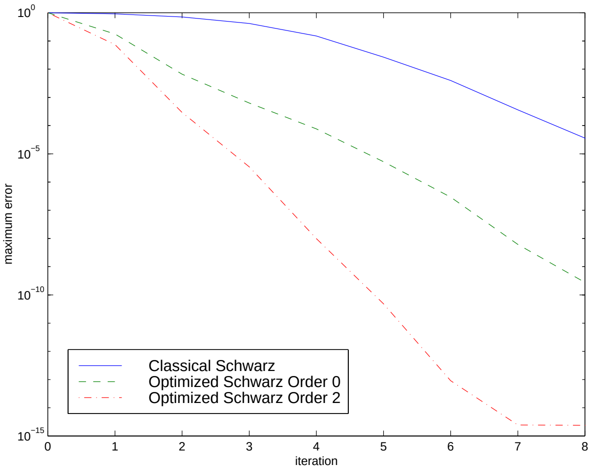


Figure 4: Optimized Schwarz methods used as preconditioners for Laplace's equation.

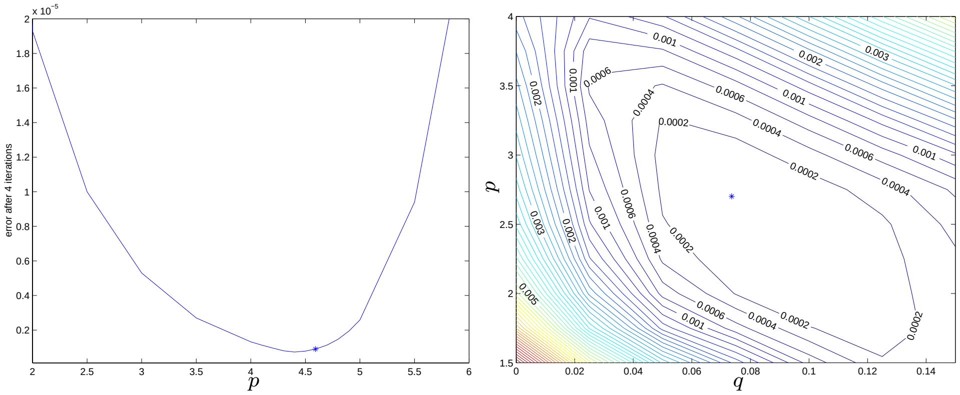


Figure 5: Comparison of the optimal parameters found by Fourier analysis and the best parameters in numerical experiments for Laplace's equation.

Note again that with the second order optimized Schwarz method, we observe a similar phenomenon like with multi grid: the acceleration with the Krylov method is not really necessary, it only brings a small improvement, since the basic iterative solver is already converging at an extremely fast rate.

Finally we investigate how close the optimal parameters obtained by Fourier analysis are to the the really optimal parameters we obtained from numerical experiments. Note that the optimal discrete parameters could also be obtained for regular rectangular meshes by a discrete Fourier analysis, but such an analysis would have to be redone for every mesh, whereas our continuous analysis is valid independently of the mesh. It is more important to have results at the continuous level for a method defined at the continuous level, since then these results remain relevant once the problem is solved on a mesh which resolves the continuous properties, independently of the particular mesh. Figure 5 shows on the left the error reduction obtained after 4 iterations of the zeroth order optimized Schwarz method for various parameters p and also indicated by a star the optimal parameter obtained by Fourier analysis. Clearly the Fourier analysis indicates where the discrete optimum lies. On the right we show a level set plot of the error after four iterations for the second order optimized Schwarz method. Again the star indicates the optimum found by the Fourier analysis. This shows that Fourier analysis is a viable tool to compute optimized Schwarz methods and the figures also show that optimized Schwarz methods are rather robust with respect to the optimization parameters.

Optimized Schwarz Method for the Helmholtz Equation

We consider the Helmholtz equation in the domain $\Omega = \mathbb{R}^2$,

$$(\Delta + \omega^2)(u) = f(x, y), \quad x, y \in \Omega \tag{12}$$

with Sommerfeld radiation conditions at infinity. We decompose Ω into two overlapping half planes $\Omega_1 = (-\infty, L] \times \mathbb{R}$ and $\Omega_2 = [0, \infty) \times \mathbb{R}$ where $L > 0$ is the overlap parameter. The classical Schwarz method to for (12) is given by

$$\begin{array}{rcll} (\Delta + \omega^2)(v^{n+1}) &=& f(x,y) & x, y \in \Omega_1, \\ v^{n+1}(L,y) &=& w^n(L,y), & \\ (\Delta + \omega^2)(w^{n+1}) &=& f(x,y) & x, y \in \Omega_2, \\ w^{n+1}(L,y) &=& v^n(L,y). & \end{array} \tag{13}$$

To analyze if the classical Schwarz method converges for the Helmholtz equation, it suffices by linearity to consider again the homogeneous problem, $f(x,y) = 0$ in (13) and to analyze convergence to zero.

Fourier Analysis of the Classical Schwarz Method

Taking a Fourier transform in y of (13) for $f(x,y) = 0$ we obtain

$$\begin{array}{rcll} \hat{v}_{xx}^{n+1}(x,k) + (\omega^2 - k^2)\hat{v}^{n+1}(x,k) &=& 0, & x \in (-\infty, L), k \in \mathbb{R}, \qquad (14) \\ \hat{v}^{n+1}(L,k) &=& \hat{w}^n(L,k), & \\ \hat{w}_{xx}^{n+1}(x,k) + (\omega^2 - k^2)\hat{w}^{n+1}(x,k) &=& 0, & x \in (0, \infty), k \in \mathbb{R}, \qquad (15) \\ \hat{w}^{n+1}(0,k) &=& \hat{v}^n(0,k). & \end{array}$$

Solving the ordinary differential equation (15) using the radiation condition at infinity and inserting the result into the boundary condition of (14) we find the solution of (14) at $x = 0$ to be

$$\hat{v}^{n+1}(0,k) = e^{-2\sqrt{k^2-\omega^2}L}\hat{v}^{n-1}(0,k)$$

and similarly for (15)

$$\hat{w}^{n+1}(L,k) = e^{-2\sqrt{k^2-\omega^2}L}\hat{w}^{n-1}(L,k).$$

Defining the convergence rate

$$\rho(k,\omega,L) := e^{-2\sqrt{k^2-\omega^2}L} \tag{16}$$

we have now two cases to distinguish: if $k^2 > \omega^2$ then $|\rho(k,\omega,L)| < 1$ and the algorithm converges as in the case of Laplace's equation. If however $k^2 < \omega^2$ then

$$|\rho(k,\omega,L)| = \left| e^{-2i\sqrt{\omega^2-k^2}L} \right| = 1$$

and convergence is lost. Therefore the classical Schwarz algorithm for the Helmholtz equation does not converge in general, the low frequencies in the error are not damped. Often it is precisely the low frequencies which are important in Helmholtz problems, since they correspond to the propagating frequencies. Thus for Helmholtz problems one is obliged to modify the Schwarz algorithm to make it work. In [CW92] a coarse mesh is introduced, fine enough to carry all the propagating modes, and in [CCEW98] the classical radiation conditions of Robin type are employed at the interfaces to obtain damping of the propagating modes. In [DJR92] and [CN98] non-overlapping variants of the Schwarz algorithm are analyzed with approximately absorbing transmission conditions. Following our analysis for Laplace's equation, we first compute the optimal transmission conditions for the Helmholtz case.

Optimal Transmission Conditions

Imposing the new transmission conditions (6) in the Schwarz algorithm for the Helmholtz equation we obtain the analog to Theorem 1 in the case of Laplace's equation:

Theorem 2 (Optimal Convergence) *Choosing Λ_v to have the symbol $\lambda_v(k) := \sqrt{k^2-\omega^2}$ and Λ_w to have the symbol $\lambda_w(k) := -\sqrt{k^2-\omega^2}$ the Schwarz method with transmission conditions (6) for the Helmholtz equation converges in two iterations independently of the overlap $L \geq 0$ and the frequency parameter k.*

Proof A Fourier transform in y and a similar calculation as in the case of Laplace's equation leads to

$$\hat{v}^{n+1}(0,k) = \rho_h \hat{v}^{n-1}(0,k)$$

and similarly for $\hat{w}^{n+1}$

$$\hat{w}^{n+1}(L,k) = \rho_h \hat{w}^{n-1}(L,k)$$

where the convergence rate ρ_h is given by

$$\rho_h(k,L) := \frac{-\sqrt{k^2-\omega^2}+\lambda_v(k)}{\sqrt{k^2-\omega^2}+\lambda_v(k)} \cdot \frac{\sqrt{k^2-\omega^2}+\lambda_w(k)}{-\sqrt{k^2-\omega^2}+\lambda_w(k)} e^{-2\sqrt{k^2-\omega^2}L}. \tag{17}$$

Hence for $\lambda_v = \sqrt{k^2-\omega^2}$ and $\lambda_w = -\sqrt{k^2-\omega^2}$ the convergence rate (17) vanishes, $\rho_h \equiv 0$ and thus, independently of the initial guess, after two steps of the Schwarz iteration the iterates are zero. ■ Again the optimal transmission conditions involve the Dirichlet to Neumann map, as in the case of Laplace's equation, and to avoid a nonlocal implementation, we propose local approximations of the optimal transmission conditions.

Optimized Local Transmission Conditions

Using a zeroth and second order approximation as given in (10), we are led to the optimization problems

$$\min_{p>0}\left(\max_{k_{\min}<k<k_{\max}}\left|\frac{(\sqrt{k^2-\omega^2}-p)^2}{(\sqrt{k^2-\omega^2}+p)^2}e^{-2\sqrt{k^2-\omega^2}L}\right|\right) \tag{18}$$

in the zeroth order approximation case and to

$$\min_{p,q>0}\left(\max_{k_{\min}<k<k_{\max}}\left|\frac{(\sqrt{k^2-\omega^2}-p-qk^2)^2}{(\sqrt{k^2-\omega^2}+p+qk^2)^2}e^{-2\sqrt{k^2-\omega^2}L}\right|\right) \tag{19}$$

in the second order approximation case. But these optimization problems have an intrinsic difficulty in the Helmholtz case: for $k^2 = \omega^2$ we obtain 1, independently of the choice of the parameter p in (18) and the parameters p and q in (19). Thus there is no hope to minimize the convergence rate uniformly in k and even the optimized Schwarz method might not converge when applied in an iterative way to the Helmholtz problem. When used as a preconditioner however, the Krylov method can easily cope with outliers in the spectrum and thus we optimize the convergence rates for all k relevant to the discrete spectrum except $k = \omega$. This leads to the convergence rates shown in Figure 6 for the model problem (20).

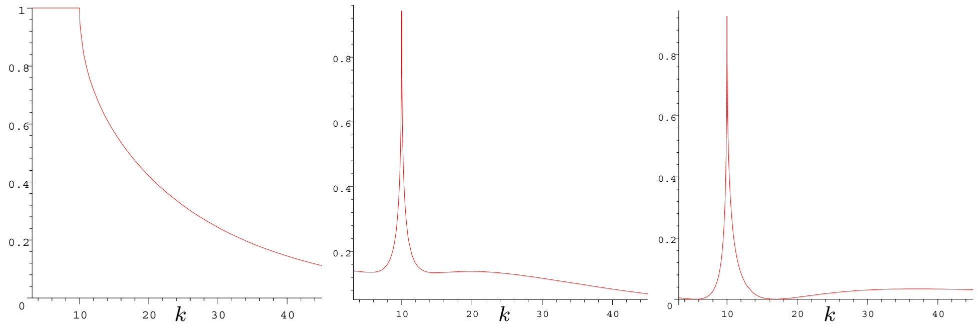


Figure 6: Convergence rates in Fourier space for a Helmholtz problem. The classical Schwarz method on the left, zeroth order optimized Schwarz method in the middle and second order optimized Schwarz method on the right.

Numerical Experiments for the Helmholtz Equation

We solve the Helmholtz equation on a rectangular domain $\Omega = [0, 2] \times [0, 1]$

$$(\Delta + \omega^2)(u) = 0, \quad x, y \in \Omega, \tag{20}$$

Robin conditions on the left and the right and homogeneous Dirichlet conditions on top and bottom. We decompose Ω into two subdomains $\Omega_1 = [0, 1 + \delta] \times [0, 1]$ and $\Omega_2 = [1 - \delta, 2] \times [0, 1]$ and apply the Schwarz algorithm as preconditioner for GMRES. Figure 7 shows the performance of the classical Schwarz method compared to the zeroth order optimized one and the second order optimized one for an overlap of $2\delta = 1/10$ with mesh parameter $h = 1/80$ and $\omega = 10$. Clearly the optimized Schwarz method shows a much better performance than the classical one.

Conclusions

We have introduced a small modification to the classical Schwarz method with a big impact. Exchanging the classical transmission conditions of Dirichlet type with transmission conditions involving local approximations of the Dirichlet to Neumann operator, the Schwarz algorithm converges orders of magnitudes faster, both when used as an iterative solver and as a preconditioner for symmetric definite and indefinite model problems.

References

[CCEW98]X.C. Cai, M. A. Casarin, F. W. Jr. Elliott, and O. B. Widlund. Overlapping Schwarz algorithms for solving Helmholtz's equation. In *Domain decomposition methods, 10 (Boulder, CO, 1997)*, pages 391–399. Amer. Math. Soc., Providence, RI, 1998.

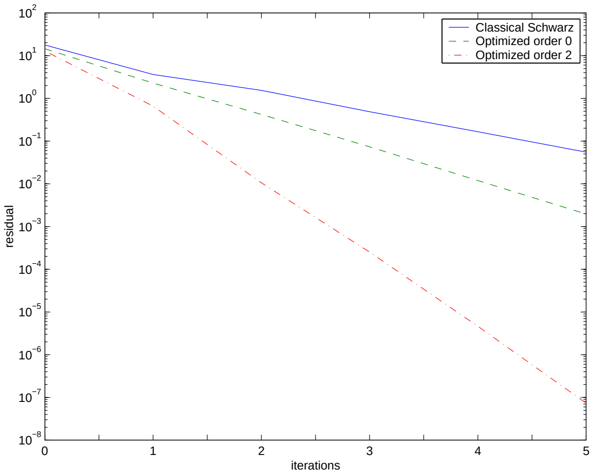


Figure 7: Performance of the classical Schwarz preconditioner compared to the optimized Schwarz preconditioners for a Helmholtz problem.

[CN98]Philippe Chevalier and Frédéric Nataf. Symmetrized method with optimized second-order conditions for the Helmholtz equation. In *Domain decomposition methods, 10 (Boulder, CO, 1997)*, pages 400–407. Amer. Math. Soc., Providence, RI, 1998.

[CNR91]P. Charton, F. Nataf, and F. Rogier. Méthode de décomposition de domaine pour l'equation d'advection-diffusion. *C. R. Acad. Sci.*, 313(9):623–626, 1991.

[CW92]Xiao-Chuan Cai and Olof Widlund. Domain decomposition algorithms for indefinite elliptic problems. *SIAM J. Sci. Statist. Comput.*, 13(1):243–258, January 1992.

[DJR92]Bruno Després, Patrick Joly, and Jean E. Roberts. A domain decomposition method for the harmonic Maxwell equations. In *Iterative methods in linear algebra (Brussels, 1991)*, pages 475–484. North-Holland, Amsterdam, 1992.

[EZ98]Bjorn Engquist and Hong-Kai Zhao. Absorbing boundary conditions for domain decomposition. *Appl. Numer. Math.*, 27(4):341–365, 1998.

[GHN99]M. J. Gander, L. Halpern, and F. Nataf. Optimal convergence for overlapping and non-overlapping Schwarz waveform relaxation. In C-H. Lai, P. Bjørstad, M. Cross, and O. Widlund, editors, *Eleventh international Conference of Domain Decomposition Methods*. ddm.org, 1999.

[Jap98]Caroline Japhet. Optimized Krylov-Ventcell method. Application to convection-diffusion problems. In Petter E. Bjørstad, Magne S. Espedal, and David E. Keyes, editors, *Proceedings of the 9th international conference on domain decomposition methods*, pages 382–389. ddm.org, 1998.

[Lio90]Pierre Louis Lions. On the Schwarz alternating method. III: a variant for nonoverlapping subdomains. In Tony F. Chan, Roland Glowinski, Jacques Périaux, and Olof Widlund, editors, *Third International Symposium on Domain Decomposition Methods for Partial Differential Equations , held in Houston, Texas, March 20-22, 1989*, Philadelphia, PA, 1990. SIAM.

[NR95]F. Nataf and F. Rogier. Factorization of the convection-diffusion operator and the Schwarz algorithm. *M^3AS*, 5(1):67–93, 1995.

[NRdS94]F. Nataf, F. Rogier, and E. de Sturler. Optimal interface conditions for domain decomposition methods. Technical report, CMAP (Ecole Polytechnique), 1994.

[SBG96]Barry F. Smith, Petter E. Bjørstad, and William Gropp. *Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations*. Cambridge University Press, 1996.

[Sch70]H. A. Schwarz. Ueber einen Grenzübergang durch alternierendes Verfahren. *Vierteljahrsschrift der Naturforschenden Gesellschaft in Zürich*, 15:272–286, May 1870.

[ST96]H. Sun and W.-P. Tang. An overdetermined Schwarz alternating method. *SIAM Journal on Scientific Computing*, 17(4):884–905, Jul. 1996.

[Tan92]Wei Pai Tang. Generalized Schwarz splittings. *SIAM J. Sci. Stat. Comp.*, 13(2):573–595, 1992.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau,

3. Dual and Dual-Primal FETI Methods for Elliptic Problems with Discontinuous Coefficients in Three Dimensions

Axel Klawonn[1], Olof B. Widlund[2]

Introduction

The Finite Element Tearing and Interconnecting (FETI) methods were first introduced by Farhat and Roux [FMR94]. An important advance, making the rate of convergence of the iteration less sensitive to the number of unknowns of the local problems, was made by Farhat, Mandel, and Roux a few years later [FMR94]. For a detailed introduction, see [FR94] and we also refer to our own papers for many additional references. Our own work, cf. [KW01, KW00b], owes a great deal to the pioneering theoretical work by Mandel and Tezaur [MT96, MT00].

The principal purpose of this paper is to survey some recent results developed by the authors. We introduce new one-parameter families of one-level FETI as well as of dual–primal FETI preconditioners which have a rate of convergence which is bounded independently of possible jumps of the coefficients of an elliptic model problem often considered in the theory of Neumann–Neumann and other iterative substructuring algorithms; see, e.g., [DW95, DSW94, MB96] and the references therein. Our new results become possible because of special scalings. One of them, for the preconditioner, is closely related to an important algorithmic idea used in the best of the Neumann–Neumann methods. The other scaling affects the choice of the projection which is used in each step of the one–level FETI iteration, whether preconditioned or not. For a certain choice of the two scalings, our preconditioner for the one–level FETI methods results in a method that is identical to one recently tested successfully for very difficult and large problems by Bhardwaj et al. [BDF$^+$00]. The scaling used in the preconditioner was originally introduced by Rixen and Farhat; see [RF99]. We note that, by now, many variants of the FETI algorithms have been designed and that a number of them have been tested extensively; see in particular [RFTM99]. Some of our results have also already been extended to Maxwell's equation in two dimensions by Toselli and Klawonn [TK99].

Recently, Farhat et al. [FLLT$^+$99] introduced a dual–primal FETI algorithm suitable for second order elliptic problems in the plane and for plate problems. A convergence analysis in the case of benign coefficients is given by Mandel and Tezaur [MT00]. Numerical experiments show a poor performance for this algorithm in three

[1] SCAI - Institute for Algorithms and Scientific Computing, GMD – German National Research Center for Information Technology, Schloss Birlinghoven, D–53754 Sankt Augustin, Germany. E-mail: `klawonn@gmd.de`, URL: `http://www.gmd.de/SCAI/people/klawonn`. This work was supported in part by the National Science Foundation under Grants NSF-CCR-9732208.

[2] Courant Institute of Mathematical Sciences, New York University, 251 Mercer Street, New York, NY 10012, USA. E-mail: `widlund@cs.nyu.edu`, URL: `http://www.cs.nyu.edu/cs/faculty/widlund`. This work was supported in part by the National Science Foundation under Grants NSF-CCR-9732208 and in part by the US Department of Energy under Contract DE-FG02-92ER25127.

dimensions; cf. [FLLT$^+$99]. Recent experiments with alternative algorithms are reported in [FLP00, Pie00]. We give a brief description of our own recent work in the final section; see [KW00b] for many more details.

The remainder of this paper is organized as follows. In the next, the second section, we introduce our elliptic problems and the basic geometry of the decomposition. In the following section, we give a short introduction to one–level FETI methods. In the fourth section, we introduce our family of preconditioners and formulate one of our main results; our results could also be extended to certain other elliptic problems as in [KW00a]. Finally, we present results on a new dual–primal FETI method for problems with discontinuous coefficient in three dimensions; see [KW00b].

A model problem, finite elements, and geometry

Let $\Omega \subset \mathbb{R}^3$, be a bounded, polyhedral region, let $\partial\Omega_D \subset \partial\Omega$ be a closed set of positive measure, and let $\partial\Omega_N := \partial\Omega \setminus \partial\Omega_D$ be its complement. We impose homogeneous Dirichlet and general Neumann boundary conditions, respectively, on these two subsets and introduce the Sobolev space $H_0^1(\Omega, \partial\Omega_D) := \{v \in H^1(\Omega) : v = 0 \text{ on } \partial\Omega_D\}$.

For simplicity, we will only consider a piecewise linear, conforming finite element approximation of the following scalar, second order model problem:

Find $u \in H_0^1(\Omega, \partial\Omega_D)$, such that

$$a(u, v) = f(v) \quad \forall v \in H_0^1(\Omega, \partial\Omega_D), \tag{1}$$

where

$$a(u, v) := \int_\Omega \rho(x)\nabla u \cdot \nabla v dx, \;\; f(v) := \int_\Omega f v dx \; + \; \int_{\partial\Omega_N} g_N v ds, \tag{2}$$

where g_N is the Neumann boundary data defined on $\partial\Omega_N$; it provides a contribution to the load vector of the finite element problem. The coefficient $\rho(x) > 0$ for $x \in \Omega$.

We decompose Ω into non-overlapping subdomains $\Omega_i, i = 1, \dots, N$, also known as substructures, and each of which is the union of shape-regular elements with the finite element nodes on the boundaries of neighboring subdomains matching across the interface $\Gamma := \overline{\left(\bigcup_{i=1}^N \partial\Omega_i\right)} \setminus \partial\Omega$. The interface Γ is decomposed into subdomain faces, regarded as open sets, which are shared by two subregions, edges which are shared by more than two subregions and the vertices which form the endpoints of edges. We denote faces of Ω_i by $\mathcal{F}^{ij}$, edges by $\mathcal{E}^{ik}$, and vertices by $\mathcal{V}^{i\ell}$.

We denote the standard finite element space of continuous, piecewise linear functions on Ω_i by $W^h(\Omega_i)$. For simplicity, we assume that the triangulation of each subdomain is quasi uniform. The diameter of Ω_i is H_i, or generically, H. We denote the corresponding finite element trace spaces by $W_i := W^h(\partial\Omega_i), i = 1, \dots, N$, and by $W := \prod_{i=1}^N W_i$ the associated product space. We will often consider elements of W which are discontinuous across the interface.

The finite element approximation of the elliptic problem is continuous across Γ and we denote the corresponding subspace of W by $\widehat{W}$. We note that while the stiffness matrix K and Schur complement S which correspond to the product space W generally are singular those of $\widehat{W}$ are not.

For the dual–primal FETI methods, we will also use an additional, intermediate subspace $\widetilde{W}$ of W for which a relatively small number of continuity constraints are enforced across the interface throughout the iteration. In our dual–primal FETI methods, the selection of these constraints will be closely related to the coarse spaces of certain primal iterative substructuring methods. One of the benefits of working in $\widetilde{W}$, rather than in W, is that certain related Schur complements $\widetilde{S}$ and S_Δ are positive definite.

We assume that possible jumps of $\rho(x)$ are aligned with the subdomain boundaries and, for simplicity, that on each subregion Ω_i, $\rho(x)$ has the constant value $\rho_i > 0$. Our bilinear form and load vector can then be written, in terms of contributions from individual subregions, as

$$a(u,v) = \sum_{i=1}^{N} \rho_i \int_{\Omega_i} \nabla u \cdot \nabla v dx, \quad f(v) = \sum_{i=1}^{N} \Big(\int_{\Omega_i} f v dx \; + \; \int_{\partial\Omega_i \cap \partial\Omega_N} g_N v ds \Big). \tag{3}$$

In our theoretical analysis, we assume that the subregions Ω_i are tetrahedra or hexahedra and that they are shape regular, i.e., not very thin. We also make a number of technical assumptions on the intersection of the boundary of the substructures and $\partial\Omega_D$; see [KW01]. We assume that H_i and H_j are comparable if the subdomains Ω_i and Ω_j are neighbors. The sets of nodes in Ω_i, on $\partial\Omega_i$, and on Γ are denoted by $\Omega_{i,h}, \partial\Omega_{i,h}$, and Γ_h, respectively.

As in previous work on Neumann–Neumann algorithms, a crucial role is played by *the weighted counting functions* $\mu_i \in \widehat{W}$, which are associated with the individual subdomain boundaries $\partial\Omega_i$; cf., e.g., [DSW96, DW95]. In this paper they will be used primarily in the definition of certain diagonal scaling matrices. These functions are defined, for $\gamma \in [1/2, \infty)$, and for $x \in \Gamma_h \cup \partial\Omega_h$, by a sum of contributions from Ω_i, and its relevant next neighbors

$$\mu_i(x) = \begin{cases} \displaystyle\sum_{j \in \mathcal{N}_x} \rho_j^\gamma(x) & x \in \partial\Omega_{i,h} \cap \partial\Omega_{j,h}, \\ \rho_i^\gamma(x) & x \in \partial\Omega_{i,h} \cap (\partial\Omega_h \setminus \Gamma_h), \\ 0 & x \in (\Gamma_h \cup \partial\Omega_h) \setminus \partial\Omega_{i,h}. \end{cases} \tag{4}$$

Here, $\mathcal{N}_x$ is the set of indices of the subregions which have x on its boundary. We note that any node of Γ_h belongs either to two faces, more than two edges, or to the vertices of several substructures.

The pseudo inverses $\mu_i^\dagger$ are defined, for $x \in \Gamma_h \cup \partial\Omega_h$, by

$$\mu_i^\dagger(x) = \begin{cases} \mu_i^{-1}(x) & \text{if } \mu_i(x) \neq 0, \\ 0 & \text{if } \mu_i(x) = 0. \end{cases}$$

A review of one–level FETI methods

In this section, we give a brief review of the original FETI method of Farhat and Roux [FMR94, FR94] and the variant with a Dirichlet preconditioner introduced in Farhat, Mandel, and Roux [FMR94]. The more general projection operators, described in this section, were first introduced for heterogeneous problems in [FR94] and they have been tested in very large scale numerical experiments; see [BDF$^+$00].

For a chosen finite element method and for each subdomain Ω_i, we assemble the local stiffness matrix $K^{(i)}$ and the local load vector corresponding to a single, appropriate term in the sums of (3). Any nodal variable, not associated with Γ_h, is called interior and it only belongs to one substructure. The interior variables of any subdomain can be eliminated by a step of block Gaussian elimination; this work can clearly be parallelized across the subdomains. The resulting matrices are the Schur complements

$$S^{(i)} = K^{(i)}_{\Gamma\Gamma} - K^{(i)}_{\Gamma I}(K^{(i)}_{II})^{-1}K^{(i)}_{I\Gamma}, \quad i = 1, \dots, N.$$

Here, Γ and I represent the interface and interior, respectively. We note that the $S^{(i)}$ are only needed in terms of matrix-vector products and that therefore the elements of these matrices need not be explicitly computed.

The values of the right hand vectors also change when the interior variables are eliminated. We denote the resulting vectors, representing the modified load originating in Ω_i, by f_i and the local vectors of interface nodal values by u_i.

We can now reformulate the finite element problem, reduced to the interface Γ, as a minimization problem with constraints given by the requirement of continuity across Γ :

Find $u \in W$, such that

$$\left.\begin{array}{c} J(u) := \frac{1}{2}\langle Su, u\rangle - \langle f, u\rangle \rightarrow \ \min \\ Bu = 0 \end{array}\right\} \tag{5}$$

where $u = [u_1 \dots u_N]^t, f = [f_1 \dots f_N]^t$, and $S = diag_{i=1}^N(S^{(i)})$ is block–diagonal. The matrix $B = [B^{(1)}, \dots, B^{(N)}]$ is constructed from $\{0, 1, -1\}$ such that the values of the solution u, associated with more than one subdomain, coincide when $Bu = 0$. We note that the choice of B is far from unique. The local Schur complements $S^{(i)}$ are positive semidefinite and they are singular for any subregion with a boundary which does not intersect $\partial\Omega_D$. The problem (5) is uniquely solvable if and only if $ker\,(S) \cap ker\,(B) = \{0\}$, i.e., if and only if S is invertible on $ker\,(B)$.

By introducing a vector of Lagrange multipliers λ, to enforce the constraints $Bu = 0$, we obtain a saddle point formulation of (5):

Find $(u, \lambda) \in W \times U$, such that

$$\left.\begin{array}{ccccc} Su & + & B^t\lambda & = & f \\ Bu & & & = & 0 \end{array}\right\}. \tag{6}$$

We note that the solution λ of (6) is unique only up to an additive vector of $ker\,(B^t)$. The space of Lagrange multipliers U is therefore chosen as $range\,(B)$.

We will also use a full column rank matrix built from all of the null space elements of S; these elements are associated with individual subdomains (the rigid body motions in the case of elasticity),

$$R = [R^{(1)} \dots R^{(N)}].$$

Thus, $range\,(R) = ker\,(S)$. We note that no subdomain with a boundary which intersects $\partial\Omega_D$ contributes to R.

The solution of the first equation in (6) exists if and only if $f - B^t\lambda \in range\,(S)$; this constraint will lead to the introduction of a projection P. We obtain,

$$u = S^\dagger(f - B^t\lambda) + R\alpha \text{ if } f - B^t\lambda \perp ker\,(S),$$

where $S^\dagger$ is a pseudoinverse of S. The value of α can be determined easily once λ has been found.

Substituting u into the second equation of (6) gives

$$BS^\dagger B^t\lambda = BS^\dagger f + BR\alpha. \tag{7}$$

We now introduce a symmetric, positive definite matrix Q which induces an inner product on U; it is defined by $\langle\lambda,\mu\rangle_Q := \langle\lambda, Q\mu\rangle$. By considering the component which is Q^{-1}−orthogonal to $G := BR$, we find that

$$\left.\begin{array}{rcl} P^tF\lambda & = & P^td \\ G^t\lambda & = & e \end{array}\right\} \tag{8}$$

with $F := BS^\dagger B^t, d := BS^\dagger f, P := I - QG(G^tQG)^{-1}G^t$, and $e := R^tf$. We note that P is an orthogonal projection, from U onto $\mathit{ker}\,(G^t)$, in the Q^{-1}−inner product, i.e., the inner product defined by $\langle\lambda, Q^{-1}\mu\rangle$.

There are different good choices for Q. In the case of homogeneous coefficients, it is sufficient to use $Q = I$, while for problems with jumps in the coefficients, we have to make a more elaborate choice to make our proofs work satisfactorily. In our analysis, Q will be a diagonal scaling matrix or we will use the preconditioner; other alternatives are discussed in [BDF$^+$00, FR94].

By multiplying (7) by $(G^tQG)^{-1}G^tQ$, we find that $\alpha := (G^tQG)^{-1}G^tQ(F\lambda - d)$ which then fully determines the primal variables in terms of λ.

We introduce the space

$$V := \{\mu \in U \;:\; \langle\mu, Bz\rangle = 0 \quad \forall z \in \mathit{ker}\,(S)\} = \mathit{ker}\,(G^t) = \mathit{range}\,(P),$$

and a space that is isomorphic to its dual,

$$V' := \{\lambda \in U : \langle\lambda, Bz\rangle_Q = 0 \quad \forall z \in \mathit{ker}\,(S)\} = \mathit{range}\,(P^t).$$

As is usual in the literature on FETI methods, we can call V the space of admissible increments. The original FETI method is a conjugate gradient method in the space V applied to

$$P^tF\lambda = P^td, \qquad \lambda \in \lambda_0 + V, \tag{9}$$

with an initial approximation λ_0 chosen such that $G^t\lambda_0 = e$. The most basic FETI preconditioner, as introduced in Farhat, Mandel, and Roux [FMR94], is of the form

$$M^{-1} := BSB^t.$$

To apply M^{-1} to a vector, N independent Dirichlet problems have to be solved, one on each subregion; it is therefore called the Dirichlet preconditioner.

To keep the search directions of the resulting preconditioned conjugate gradient method in the space V, the application of the preconditioner M^{-1} is followed by an application of the projection P. Hence, the Dirichlet variant of the FETI method is the conjugate gradient algorithm applied to the equation

$$PM^{-1}P^tF\,\lambda = PM^{-1}P^t\,d, \qquad \lambda \in \lambda_0 + V. \tag{10}$$

We note that for $\lambda \in V$, $PM^{-1}P^tF\lambda = PM^{-1}P^tP^tFP\lambda$, and that we can therefore view the operator on left hand side of (10) as the product of two symmetric matrices.

It is well known that an appropriate norm of the iteration error of the conjugate gradient method will decrease at least by a factor

$$2(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1})^k,$$

in k steps. Here κ is the ratio of the largest and smallest eigenvalues of the iteration operator. The main task in the theory is therefore always to obtain a good bound for the condition number κ.

We note that several different possibilities of improving the FETI preconditioner M^{-1} have already been explored. Some interesting variants are discussed by Rixen and Farhat [RF99], in a framework of mechanically consistent preconditioners, in the case of redundant Lagrange multipliers; see also Klawonn and Widlund [KW01, section 5] for an analysis.

New one-level FETI preconditioners with non-redundant Lagrange multipliers

In this section, we outline some of our results on a family of new FETI preconditioners with an improved condition number estimate compared to that of Mandel and Tezaur [MT96]. Most importantly, we obtain a uniform bound for arbitrary positive values of the ρ_i if the scaling matrix Q, which enters the definition of P, is chosen carefully. In our proofs, we use several arguments developed in [MT96], but our presentation also differs considerably in several respects.

We now assume that B has full row rank, i.e., the constraints are linearly independent and there are no redundant Lagrange multipliers.

Our new preconditioner is defined, for any diagonal matrix D with positive elements, as

$$\widehat{M}^{-1} := (BD^{-1}B^t)^{-1}BD^{-1}SD^{-1}B^t(BD^{-1}B^t)^{-1}. \tag{11}$$

To obtain a method, which converges at a rate which is independent of the coefficient jumps, we now choose a special family of matrices D; a careful choice of the scaling Q, introduced in the definition of the operator P, will also be required. As in previous work on Neumann–Neumann algorithms, a crucial role is played by the weighted counting functions μ_i, associated with the individual $\partial\Omega_i$, and already introduced in (4). The diagonal matrix $D^{(i)}$ has the diagonal entry $\rho_i^\gamma(x)\mu_i^\dagger(x)$ corresponding to the point $x \in \partial\Omega_{i,h}$. Finally, we set $D := diag_{i=1}^N(D^{(i)})$. We note that this matrix is a block–diagonal matrix which operates on elements in the product space W.

We now give a condition number estimate for the preconditioned FETI operator $P\widehat{M}^{-1}P^tF$; cf. [KW01]. The result holds for $Q = \widehat{M}^{-1}$ and also for a special choice of B and a special diagonal Q; in the case of continuous coefficients, it is sufficient to choose Q as a multiple of the identity matrix for the next theorem to be valid.

Theorem 1 *The condition number of the FETI method, with the new preconditioner $\widehat{M}$, satisfies*

$$\kappa(P\widehat{M}^{-1}P^tF) \leq C\,(1+\log(H/h))^2.$$

Here, $\kappa(P\widehat{M}^{-1}P^tF)$ is the spectral condition number of $P\widehat{M}^{-1}P^tF$, and C is independent of h, H, γ, and the values of the ρ_i.

A New Dual–Primal FETI method

In previous studies of dual–primal FETI methods for problems in two dimensions, see Farhat, Lesoinne, Le Tallec, Pierson, and Rixen [FLLT$^+$99] and Mandel and Tezaur [MT00], the constraints on the degrees of freedom associated with the vertices of the substructures are enforced, i.e., the corresponding degrees of freedom have been added to the global set of variables, while all the constraints associated with the edge nodes are enforced only at the convergence of the iterative method. In each step of the iteration a fully assembled linear subsystem is solved. In a simple two–dimensional case, this subsystem corresponds to all the interior and cross point variables; these variables can be eliminated at a modest expense since we can first eliminate all the interior variables, in parallel across the subdomains, resulting in a Schur complement for the cross point variables which can be shown to be sparse. It has a dimension which equals the number of subdomain vertices which do not belong to $\partial\Omega_D$.

In their recent paper, Mandel and Tezaur [MT00] established a condition number bound of the form $C(1+\log(H/h))^2$ for the resulting FETI method equipped with a Dirichlet preconditioner which is very similar to those used for the older FETI methods and which is built from local solvers on the subregions with zero Dirichlet conditions at the vertices of the subregions. They also established a corresponding result for a fourth-order elliptic problem in the plane. Their elegant proof in [MT00] relies, for the second order equation, on linear algebra arguments and a lemma from a classical paper by Bramble, Pasciak, and Schatz [BPS86, Lemma 3.5].

The same algorithm is also defined for three dimensions but it does not perform well. This is undoubtedly related to the poor performance of many *vertex-based* iterative substructuring methods; see [DSW94, Section 6.1] and [KW00b]. Recently, Farhat et al. added constraints to this basic algorithm, see [FLP00], and improved the performance.

In our approach, we first carry out a change of variables prior to dividing the variables into a primal and a dual subspace. The number of constraints enforced in each iteration will now be larger, but we will still be able to work with a number of constraints which is uniformly bounded for each substructure.

One of our new algorithms is given in terms of a space $\widetilde{W} \subset W$ for which we have continuity at the subdomain vertices, and also common values of the averages over all edges and all faces of the interface. This space can naturally be written as a direct sum of two subspaces, corresponding to a primal and a dual part of the problem, i.e.,

$$\widetilde{W} = \widehat{W}_\Pi \oplus \widetilde{W}_\Delta.$$

The first subspace, $\widehat{W}_\Pi \subset \widehat{W}$, which together with the interior subspaces, defines the subsystem which is fully assembled, factored, and solved in each iteration step.

It is defined as the range of the following interpolation operator I_B^h defined, for any $u_h \in \widetilde{W}$, by

$$I_B^h u_h(x) = \sum_{\mathcal{V}^{i\ell} \in \Gamma} u^h(\mathcal{V}^{i\ell})\varphi_{\mathcal{V}^{i\ell}}(x) + \sum_{\mathcal{E}^{ik} \subset \Gamma} \bar{u}^h_{\mathcal{E}^{ik}} \theta_{\mathcal{E}^{ik}}(x) + \sum_{\mathcal{F}^{ij} \subset \Gamma} \bar{u}^h_{\mathcal{F}^{ij}} \theta_{\mathcal{F}^{ij}}(x). \qquad (12)$$

Here,

$$\bar{u}^h_{\mathcal{E}^{ik}} = \frac{\int_{\mathcal{E}^{ik}} I^h(\theta_{\mathcal{E}^{ik}} u_h) ds}{\int_{\mathcal{E}^{ik}} \theta_{\mathcal{E}^{ik}} ds} \quad and \quad \bar{u}^h_{\mathcal{F}^{ij}} = \frac{\int_{\mathcal{F}^{ij}} I^h(\theta_{\mathcal{F}^{ij}} u_h) dx}{\int_{\mathcal{F}^{ij}} \theta_{\mathcal{F}^{ik}} dx},$$

$\varphi_{\mathcal{V}^{i\ell}}$ are the standard nodal basis function, and $\theta_{\mathcal{E}^{ik}}$ and $\theta_{\mathcal{F}^{ij}}$ the discrete harmonic functions which equal 1 on $\mathcal{E}_h^{ik}$ and $\mathcal{F}_h^{ij}$, respectively, and vanish elsewhere on Γ_h. The operator I_B^h, introduced in [DSW94, p. 1690], has almost optimal stability properties. Let us note that several cheaper algorithm, based on different interpolation operators, are also discussed in [KW00b].

The subspace $\widehat{W}_\Pi$ is thus given in terms of the vertex variables, the averages of the values over the individual edges of the set of interface nodes Γ_h, and the averages over the individual faces of substructures.

We note that the dimension of this first subspace is relatively small; in the case of hexahedral substructures there are seven global variables for each interior substructure since there are eight vertices, each shared by eight hexahedra, twelve edges, each shared by four, and six faces each shared by a pair of substructures. We note that the count is smaller, relative to the number of substructures, in the case of tetrahedral subregions. We can demonstrate that the resulting system can be assembled and solved at an acceptable cost which only exceeds that for the more primitive algorithm in which we enforce only the vertex constraints in each step, by a constant factor. We note that we have also developed a second method with only four global variables per subdomain; our theoretical results for that method involves a third power of the logarithm. We have no doubts that a number of other promising alternatives could be developed given the rich choice of coarse spaces for the primal iterative substructuring methods.

The second subspace, denoted by $\widetilde{W}_\Delta$, is associated with the nodal points on the edges and faces of the interface Γ. It is the direct sum of local subspaces of $\widetilde{W}$. For each subdomain Ω_i, the local subspace consists of functions that vanish at the subdomain vertices and have zero average on each individual edge and face. They are extended by zero on all of the $\partial\Omega_j, j \neq i$; it is easy to see that these functions satisfies the continuity requirements associated with $\widetilde{W}$.

The linear systems solved in the preconditioning step of our FETI–DP algorithm, which is directly related to $\widetilde{W}_\Delta$, have zero Dirichlet boundary conditions at the vertices and also satisfy the constraints that the averages over individual edges and faces vanish. The nodal values represent the original nodal values minus the average over the edge or face to which the node belongs. This construction makes the local solvers well defined and the resulting set of variables represent a subspace complementary to the first subspace; together with the interior spaces they represent the variables of the entire linear space of the partially subassembled system.

We can now formulate one of our FETI–DP algorithms; for details on its implementation, we refer to Klawonn and Widlund [KW00b].

We first eliminate, after a partial change of variables, all unknowns of the first subspace as well as the interior variables, and obtain a Schur complement $\widetilde{S}$.

Analogously, we get from the load vectors associated with each subdomain a reduced right hand side $\tilde{f}_\Delta$. We can now reformulate the original finite element problem, reduced to the degrees of freedom of the second subspace $\widetilde{W}_\Delta$, as a minimization problem with constraints given by the requirement of continuity across Γ_h:

Find $u_\Delta \in \widetilde{W}_\Delta$, such that

$$\left.\begin{array}{c} J(u_\Delta) := \frac{1}{2}\langle \widetilde{S}u_\Delta, u_\Delta\rangle - \langle \tilde{f}_\Delta, u_d\rangle \to \min \\ B_\Delta u_\Delta = 0 \end{array}\right\}. \tag{13}$$

The matrix B_Δ is constructed from $\{0, 1, -1\}$ in the same fashion as B. Since we already have imposed a constraint on the averages over each edge and each face, we may drop one of the point constraints for each edge and each face when constructing the matrix B_Δ. By introducing a set of Lagrange multipliers $\lambda \in V := \mathit{range}\,(B_\Delta)$, to enforce the constraints $B_\Delta u_\Delta = 0$, we obtain a saddle point formulation of (13), which is similar to (6). We use that $\widetilde{S}$ is invertible and eliminate the subvector u_Δ, and obtain the following system for the dual variable:

$$F_\Delta \lambda = d_\Delta, \tag{14}$$

where

$$F_\Delta := B_\Delta \widetilde{S}^{-1} B_\Delta^t$$

and the right hand side

$$d_\Delta := B_\Delta \widetilde{S}^{-1} \tilde{f}_\Delta.$$

To define the FETI–DP Dirichlet preconditioner, we need to introduce an additional, third set of Schur complement matrices,

$$S_{\Delta\Delta}^{(i)} := K_{\Delta\Delta}^{(i)} - K_{\Delta I}^{(i)}(K_{II}^{(i)})^{-1}K_{I\Delta}^{(i)}, \quad i = 1, \ldots, N,$$

which can also be obtained from $S^{(i)}$ by removing the rows and columns that correspond to the vertices and the edge and face averages, i.e., all the variables of the first subspace $\widehat{W}_\Pi$. Here, $K_{\Delta\Delta}^{(i)}$ is the principal minor of the stiffness matrix after the change of variables and it is related to the variables of $\widetilde{W}_\Delta$. The associated block–diagonal matrix is denoted by

$$S_{\Delta\Delta} := \mathit{diag}_{i=1}^N(S_{\Delta\Delta}^{(i)}).$$

We can compute the action of $S_{\Delta\Delta}$ on a vector from the second subspace $\widetilde{W}_\Delta$ by solving local problems with solutions that are constrained to vanish at the cross points and to have zero edge and face averages; these constraints can be enforced by using Lagrange multipliers or a partial change of basis.

As in the fourth section, cf. (11), we solve the dual system (14) using the preconditioned conjugate gradient algorithm with the preconditioner

$$M_B^{-1} := (B_\Delta D_\Delta^{-1} B_\Delta^t)^{-1} B_\Delta D_\Delta^{-1} S_{\Delta\Delta} D_\Delta^{-1} B_\Delta^t (B_\Delta D_\Delta^{-1} B_\Delta^t)^{-1}. \tag{15}$$

Here, D_Δ is a diagonal matrix with positive elements on the diagonal. It can be easily seen that $B_\Delta D_\Delta^{-1} B_\Delta^t$ is a block-diagonal matrix and thus its inverse can be computed at essentially no extra cost; the block sizes are n_x, where n_x is the number of Lagrange multipliers employed to enforce continuity at the point x. In order to obtain a method that converges at a rate which is independent of the coefficient jumps, we now choose a special family of matrices D_Δ, cf. also Klawonn and Widlund [KW01, sect. 4]. We first define the contributions of each subdomain boundary $\partial\Omega_i$ in terms of a diagonal matrix $D_\Delta^{(i)}$. For any point x on an edge or a face of Ω_i, there is an entry on the diagonal of $D_\Delta^{(i)}$ equal to $\rho_i^\gamma(x)\mu_i^\dagger(x)$. We now set

$$D_\Delta := diag_{j=1}^N(D_\Delta^{(j)}).$$

The dual–primal FETI method is now the standard preconditioned conjugate gradient algorithm for solving the preconditioned system

$$M_B^{-1}F_\Delta\lambda = M_B^{-1}d_\Delta.$$

A proof of the following theorem can be found in Klawonn and Widlund [KW00b].

Theorem 2 *The condition number of the FETI–DP method with the preconditioner* M_B *satisfies*

$$\kappa(M_B^{-1}F_\Delta) \le C\,(1+\log(H/h))^2.$$

Here, C *is independent of* h, H, γ, *and the values of the* ρ_i.

References

[BDF$^+$00]Manoj Bhardwaj, David Day, Charbel Farhat, Michel Lesoinne, Kendall Pierson, and Daniel Rixen. Application of the FETI method to ASCI problems - scalability results on one thousand processors and discussion of highly heterogeneous problems. *Int. J. Numer. Meth. Engng.*, 47:513–535, 2000.

[BPS86]James H. Bramble, Joseph E. Pasciak, and Alfred H. Schatz. The construction of preconditioners for elliptic problems by substructuring, I. *Math. Comp.*, 47(175):103–134, 1986.

[DSW94]Maksymilian Dryja, Barry F. Smith, and Olof B. Widlund. Schwarz analysis of iterative substructuring algorithms for elliptic problems in three dimensions. *SIAM J. Numer. Anal.*, 31(6):1662–1694, December 1994.

[DSW96]Maksymilian Dryja, Marcus V. Sarkis, and Olof B. Widlund. Multilevel Schwarz methods for elliptic problems with discontinuous coefficients in three dimensions. *Numer. Math.*, 72(3):313–348, 1996.

[DW95]Maksymilian Dryja and Olof B. Widlund. Schwarz methods of Neumann-Neumann type for three-dimensional elliptic finite element problems. *Comm. Pure Appl. Math.*, 48(2):121–155, February 1995.

[FLLT$^+$99]Charbel Farhat, Michel Lesoinne, Patrick Le Tallec, Kendall Pierson, and Daniel Rixen. FETI-DP: A dual-primal unified FETI method – part i: A faster alternative to the two-level FETI method. Technical Report U–CAS–99–15, University of Colorado at Boulder, Center for Aerospace Structures, August 1999.

[FLP00]Charbel Farhat, Michel Lesoinne, and Kendall Pierson. A scalable dual-primal domain decomposition method. *Numer. Lin. Alg. Appl.*, 2000. To appear.

[FMR94]Charbel Farhat, Jan Mandel, and Francois-Xavier Roux. Optimal convergence properties of the FETI domain decomposition method. *Comput. Methods Appl. Mech. Engrg.*, 115:367–388, 1994.

[FR94]Charbel Farhat and François-Xavier Roux. Implicit parallel processing in structural mechanics. In J. Tinsley Oden, editor, *Computational Mechanics Advances*, volume 2 (1), pages 1–124. North-Holland, 1994.

[KW00a]Axel Klawonn and Olof B. Widlund. A domain decomposition method with Lagrange multipliers and inexact solvers for linear elasticity. *SIAM J. Sci. Comput.*, 22(4):1199–1219, 2000.

[KW00b]Axel Klawonn and Olof B. Widlund. FETI-DP Methods for Three-Dimensional Elliptic Problems with Heterogeneous Coefficients. Technical report, Courant Institute of Mathematical Sciences, 2000. In Preparation.

[KW01]Axel Klawonn and Olof B. Widlund. FETI and Neumann–Neumann Iterative Substructuring Methods: Connections and New Results. *Comm. Pure Appl. Math.*, 54:57–90, January 2001.

[MB96]Jan Mandel and Marian Brezina. Balancing Domain Decomposition for Problems with Large Jumps in Coefficients. *Math. Comp.*, 65:1387–1401, 1996.

[MT96]Jan Mandel and Radek Tezaur. Convergence of a Substructuring Method with Lagrange Multipliers. *Numer. Math.*, 73:473–487, 1996.

[MT00]Jan Mandel and Radek Tezaur. On the convergence of a dual-primal substructuring method. Technical report, University of Colorado at Denver, Department of Mathematics, January 2000. To appear in Numer. Math. `URL: http://www-math.cudenver.edu/ jmandel/papers/dp.ps.gz`.

[Pie00]Kendall H. Pierson. *A family of domain decomposition methods for the massively parallel solution of computational mechanics problems.* PhD thesis, University of Colorado at Boulder, Aerospace Engineering, 2000.

[RF99]Daniel Rixen and Charbel Farhat. A simple and efficient extension of a class of substructure based preconditioners to heterogeneous structural mechanics problems. *Int. J. Numer. Meth. Engng.*, 44:489–516, 1999.

[RFTM99]Daniel Rixen, Charbel Farhat, Radek Tezaur, and Jan Mandel. Theoretical Comparison of the FETI and Algebraically Partitioned FETI Methods, and Performance Comparisons with a Direct Sparse Solver. *Int. J. Numer. Meth. Engng.*, 46:501–534, 1999.

[TK99]Andrea Toselli and Axel Klawonn. A FETI domain decomposition method for Maxwell's equations with discontinuous coefficients in two dimensions. Technical Report 788, Department of Computer Science, Courant Institute, 1999. Submitted to SIAM J. Numer. Anal.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau, ©2001 DDM.org

4. Decomposition of Energy Space and Virtual Control for Parabolic Systems

J.L. Lions [1]

Introduction

Methods of choice for attempting the control of distributed systems (i.e. systems modelled by Partial Differential Equations, PDE's in short) are **decomposition methods**.

Given the state equation -i.e. PDE's containing control (should it be distributed or on the boundary)- one can decompose (i) **the operator**, or (ii) **the geometrical domain**, or (iii) **the spaces describing the domain of the operator**.

Method (i), based on **splitting up** ideas has been used in a paper by A. Bensoussan, J.L. Lions and R. Temam[BLT94]. At the end of this paper, some remarks were made concerning domain decomposition. New methods (also based on virtual control) are given in a note of J.L. Lions and O. Pironneau[LP99a] and in the paper of J.L. Lions[Lio00].

DDM (Domain Decomposition Methods) are now absolutely essential for the **Analysis** of problems (i.e. PDE's without control). As observed by J.E. Lagnese and G. Leugering[LL00] while there is an extensive literature on DDM for direct simulation, the literature is much more scarse concerning DDM and optimal control.

The first contributions were due to B. Despres[Des91], J.D. Benamou and B. Despres[JD97], J.D. Benamou[Ben97, Ben98], and the paper just quoted by J. Lagnese and G. Leugering.

Another set of ideas has been introduced by J.L. Lions and O. Pironneau in 3 notes[LP98a, LP98b, LP99b] where one introduces for **all** problems (i.e. problems with or **without** control functions) so called **virtual controls** with the goal to have **all** problems entering in **one** model.

First numerical results are reported in these notes.

We want here to study the possibility (iii), namely the decomposition of **spaces** describing the domain of the operator. In a (slightly) more precise manner, if A is the main symmetric part of the stationary operator contained in the model, then we consider the "**energy space**" $D(A^{1/2})$ (the domain of $A^{1/2}$) - a space that we denote by V. It is this space that we decompose in the present paper.

For stationary problems without control, this technique has been introduced in R. Glowinski, J.L. Lions and O. Pironneau[GLP99] .

We show here how it can be applied for the **control of parabolic systems**.

[1] Collège de France, jacques-louis.lions@college-de-france.fr

Elliptic regularization of parabolic equations

Let V and H be two real Hilbert spaces, such that

$$V \subset H \text{ },V \text{ dense in } H,\ V \rightarrow H \text{ continuous.} \tag{2.1}$$

We shall identify H with its dual, so that

$$V \subset H \subset V' \tag{2.2}$$

where V' denotes the dual of V.

Let $a(\varphi, \hat{\varphi})$ be a continuous bilinear form on V, such that

$$a(\varphi,\varphi) \geq \alpha \|\varphi\|^2 \qquad \forall\, \varphi \in V\ ,\ \alpha > 0\ , \tag{2.3}$$

where $\|\varphi\|$ denotes the norm of φ in V.

Let f be given such that

$$f \in L^2(0,T;V'). \tag{2.4}$$

We are looking for a function u such that

$$\left| \begin{array}{l} u \in L^2(0,T;V),\ \dfrac{\partial u}{\partial t} \in L^2(0,T;V'), \\ (\dfrac{\partial u}{\partial t}, \hat{u}) + a(u,\hat{u}) = (f,\hat{u}) \quad \forall\, \hat{u} \in V\ , \\ u|_{t=0} = 0 \end{array} \right. \tag{2.5}$$

(in (2.5) (f, φ) denotes the duality between V' and V). It follows from $(2.5)_1$ that (after possible change on a set of measure 0) the function $t \rightarrow u(t)$ is continuous from $[0,T] \rightarrow H$.

It is known that problem (2.5) **admits a unique solution** (cf. J.L. Lions[Lio61] where considerably more general situations are considered), the mapping $f \rightarrow u$ being continuous from $L^2(0,T;V')$ into the space of functions u satisfying to $(2.5)_1$. □

For reasons that will appear later on, we are going to use an **elliptic regularization** (J.L. Lions[Lio63]) **of problem** (2.5).

We define

$$W = \{u|\ u \in L^2(0,T;V), \frac{\partial u}{\partial t} \in L^2(0,T;H), u(0) = 0\}\ . \tag{2.6}$$

For $u, \hat{u} \in W$, we define

$$\mathcal{A}_\gamma(u,\hat{u}) = \gamma \int_0^T\ (\frac{\partial u}{\partial t}\ ,\ \frac{\partial \hat{u}}{\partial t})dt + \int_0^T\ [(\frac{\partial u}{\partial t}, \hat{u}) + a(u,\hat{u})]dt\ , \tag{2.7}$$

where γ is given > 0.

The bilinear form $u, \hat{u} \rightarrow \mathcal{A}_\gamma(u,\hat{u})$ is continuous on W. Moreover

$$\mathcal{A}_\gamma(u,u) = \gamma \int_0^T\ \|u(t)\|_H^2 dt + \frac{1}{2}\|u(T)\|_H^2 + \int_0^T\ a(u)\ dt\ , \tag{2.8}$$

where $\|u\|_H = (u,u)^{1/2}$, $a(u) = a(u,u)$.

By virtue of (2.3) it follows that

$$\mathcal{A}_\gamma(u,u) \geq \gamma \int_0^T \|\frac{\partial u}{\partial t}(t)\|_H^2 dt + \alpha \int_0^T \|u\|^2 \, dt \ , \tag{2.9}$$

so that in particular

$$\left| \begin{array}{l} \mathcal{A}_\gamma(u,u) \geq \inf(\gamma,\alpha) \parallel u \parallel_W^2 \qquad \text{where} \\ \parallel u \parallel_W^2 = \displaystyle\int_0^T (\parallel u(t) \parallel^2 + \parallel \frac{\partial u}{\partial t}(t) \parallel_H^2) dt \ . \end{array} \right. \tag{2.10}$$

It then immediately follows (LAX-MILGRAM's Lemma) that there exists **a unique element** u_γ solution of

$$\left| \begin{array}{l} \mathcal{A}_\gamma(u_\gamma,\hat{u}) = \int_0^T (f,\hat{u})dt \quad \forall \ \hat{u} \in W \ , \\ u_\gamma \in W \ . \end{array} \right. \tag{2.11}$$

Equation (2.11) **is called an elliptic regularization of** (2.5).

One has the following property (J.L. Lions[Lio63]) :

(2.12) | as $\gamma \to 0$, **the solution** u_γ of (2.11) **converges toward the solution** u **of** (2.5) **in the sense that**
$u_\gamma \to u$ in $L^2(0,T;V)$ weakly,
$\dfrac{\partial u_\gamma}{\partial t} \to \dfrac{\partial u}{\partial t}$ in $L^2(0,T;V')$ weakly .

Before briefly recalling the (simple) proof of (2.12), a few remarks are in order.

Remark 2.1.

Let's give an interpretation of the above equations in **non**-variational terms. We define $A \in \mathcal{L}(V;V')$ by

$$(A\varphi,\psi) = a(\varphi,\psi) \quad \forall \ \varphi,\psi \in V \ .$$

Then (2.5) reads

$$\frac{\partial u}{\partial t} + Au = f \ , \quad u|_{t=0} = 0 \ , \quad u \in L^2(0,T;V) \ , \tag{2.13}$$

and (2.11) is equivalent to

$$\left| \begin{array}{l} -\gamma \, \dfrac{\partial^2 u_\gamma}{\partial t^2} + \dfrac{\partial u_\gamma}{\partial t} + Au_\gamma = f \ , \\ u_\gamma|_{t=0} = 0 \ , \quad \dfrac{\partial u_\gamma}{\partial t}(T) = 0 \ , \\ u_\gamma \in L^2(0,T;V), \ \dfrac{\partial u_\gamma}{\partial t} \in L^2(0,T;H). \end{array} \right. \tag{2.14}$$

□

Remark 2.2.

If A is a second order elliptic operator, then the operator

$$-\gamma \frac{\partial^2}{\partial t^2} + \frac{\partial}{\partial t} + A \tag{2.15}$$

is indeed **an elliptic operator**. We are dealing with elliptic regularization. But if A is, say, a 4th order elliptic operator, then the operator (2.15) is **quasi** elliptic. We nevertheless keep the term of elliptic regularization. □

Let us now sketch the proof of (2.12). It follows from (2.9) that as $\gamma \to 0$, $u_\gamma(\text{resp.}\sqrt{\gamma}\,\frac{\partial u_\gamma}{\partial t})$ remains in a bounded set of $L^2(0,T;V)$ (resp. $L^2(0,T;H)$). We can therefore extract a subsequence still denoted by u_γ such that

$$u_\gamma \to w \text{ in } L^2(0,T;V) \text{ weakly}$$

and $\sqrt{\gamma}\,\frac{\partial u_\gamma}{\partial t} \to \xi$ in $L^2(0,T;H)$ weakly. But $\sqrt{\gamma}\,\frac{\partial u_\gamma}{\partial t} \to 0$ in the space of distributions in t with values in V, so that $\xi = 0$.

We rewrite (2.11) as

$$\gamma \int_0^T (\frac{\partial u_\gamma}{\partial t}, \frac{\partial \hat{u}}{\partial t})_H dt - \int_0^T (u_\gamma, \frac{\partial \hat{u}}{\partial t})_H dt + \int_0^T a(u,\hat{u})dt = \int_0^T (f,\hat{u})dt \tag{2.16}$$

where we have taken $\hat{u} \in W$ such that

$$\hat{u}(T) = 0\ . \tag{2.17}$$

We can pass now to the limit in (2.16). We obtain

$$-\int_0^T (w, \frac{\partial \hat{u}}{\partial t})_H dt + \int_0^T a(w,\hat{u})dt = \int_0^T (f,\hat{u})dt$$

$\forall \hat{u} \in W$ such that (2.17) is satisfied.

Hence $w = u$. □

Remark 2.3.

We could also use a different elliptic regularization, namely

$$\int_0^T (\gamma, (\frac{\partial u}{\partial t}, \frac{\partial \hat{u}}{\partial t})_{V'} dt + \int_0^T [(\frac{\partial u}{\partial t}, \hat{u}) + a(v,\hat{u})]dt \tag{2.18}$$

defined on the space of functions u such that $u \in L^2(0,T;V)$ and $\frac{\partial u}{\partial t} \in L^2(0,T;V')$ (instead of $L^2(0,T;H)$). In a sense (2.18) is more natural but (2.7) avoids the use of V'. □

Remark 2.4.

One has also (cf. J.L. Lions[Lio63])

$$\frac{\partial u_\gamma}{\partial t} \to \frac{\partial u}{\partial t} \qquad \text{in } L^2(0,T;V') \quad \text{weakly}\ . \tag{2.19}$$

□

A Control problem and its elliptic regularization

We introduce now the **space of controls** v

$$\left| \begin{array}{l} v \in L^2(0,T;\mathcal{U}), \\ \mathcal{U} = \text{ real Hilbert space .} \end{array} \right. \tag{3.1}$$

If B is an operator such that

$$B \in \mathcal{L}(\mathcal{U};V'), \tag{3.2}$$

the state equation is given by

$$\left| \begin{array}{l} (\dfrac{\partial y}{\partial t}, \hat{y}) + a(y,\hat{y}) = (Bv,\hat{y}) \quad \forall\, \hat{y} \in V \ , \\ y \in L^2(0,T;V),\ \dfrac{\partial y}{\partial t} \in L^2(0,T;V'), \\ y|_{t=0} = 0 \ . \end{array} \right. \tag{3.3}$$

The cost function is given by

$$J(v) = \frac{1}{2}\int_0^T \|v\|^2_{\mathcal{U}} dt + \frac{\beta}{2}\|y(T;v) - y^T\|^2_{H} \tag{3.4}$$

where β is given > 0 and where y^T is a given element of H.

The problem of control is now to find

$$\inf_{v \in L^2(0,T;\mathcal{U})} J(v) \tag{3.5}$$

This problem admits a unique solution v_{opt}, i.e. there exists a unique v_{opt} such that

$$J(v_{\text{opt}}) = \inf_{v \in L^2(0,T;\mathcal{U})} J(v) \ . \tag{3.6}$$

□

We consider now the "elliptic regularization" of problem (3.3) (3.6).

With the notations of previous section , we define the state $y_\gamma \in W$ by

$$\mathcal{A}_\gamma(y_\gamma, \hat{y}) = \int_0^T (Bv,\hat{y})dt \quad \forall \hat{y} \in W \ . \tag{3.7}$$

This problem admits a unique solution $y_\gamma = y_\gamma(v)$, and we can introduce

$$J_\gamma(v) = \frac{1}{2}\int_0^T \|v\|^2_{\mathcal{U}} dt + \frac{\beta}{2}\|y_\gamma(T;v) - y^T\|^2_{H} \tag{3.8}$$

Of course, **there exists a unique element** v_γ **in** $L^2(0,T;\mathcal{U})$ such that

$$J_\gamma(v_\gamma) = \inf .J_\gamma(v), \quad v \in L^2(0,T;\mathcal{U}) \ . \tag{3.9}$$

Let us briefly sketch the (easy) proof of

$$(3.10)\qquad \left| \begin{array}{l} \text{as } \gamma \to 0\ ,\ J_\gamma(v_\gamma) \to J(v_{\text{opt}}),\ v_\gamma \to v_{\text{opt}} \ \text{ in} \\ L^2(0,T;\mathcal{U}) \text{ weakly and } y_\gamma(v_\gamma) \to y(v_{\text{opt}}) \text{ in } L^2(0,T;V) \text{ weakly .} \end{array} \right.$$

For v fixed in $L^2(0,T;\mathcal{U})$, one knows that $y_\gamma(v) \to y(v)$ in $L^2(0,T;V)$ weakly, and (cf. J.L. Lions[Lio63]) $y_\gamma(T;v) \to y(T;v)$ in H strongly. Therefore $J_\gamma(v) \to J(v)$ so that

$$(3.11)\qquad \lim.\sup.J_\gamma(v_\gamma) \le \inf J(v)\ ,\ v \in L^2(0,T;\mathcal{U}).$$

It follows from (3.11) that v_γ remains in a bounded subset of $L^2(0,T;\mathcal{U}$. By extracting a subsequence, we can assume that

$$(3.12)\qquad v_\gamma \to w \quad \text{in } L^2(0,T;\mathcal{U}) \quad \text{faible}$$

and one verifies that $y_\gamma(T;v_\gamma) \to y(T;w)$ in H weakly.

Therefore

$$(3.13)\qquad \lim\inf.J_\gamma(v_\gamma) \ge J(w).$$

Comparing (3.11) (3.13) and using (3.12) gives (3.10). □

Remark 3.1.

Everything which has been said above readily extends to similar problems with constraints on v :

$$(3.14)\qquad \left| \begin{array}{l} v \in L^2(0,T;\mathcal{U}_{ad}) \\ \mathcal{U}_{ad} \quad \text{closed convex subset of } \mathcal{U}\ . \end{array} \right.$$

□

Remark 3.2.

One can write, for all the problems considered, the necessary and sufficient conditions (the so called "**Optimality System**") for v to be optimal. Cf. J.L. Lions[Lio68].

□

Orientation.

We want now to "decompose" problem (3.5) based on

(i) a decomposition of the energy space V ;

(ii) the elliptic-regularized problem (3.9). □

Decomposition of the energy space

We assume that

$$(4.1)\qquad V = V_1 + V_2$$

where

$$(4.2)\qquad V_i = \text{ closed subspace of } V\ ,$$

$$V_1 \cap V_2 = \{0\} \quad \text{or not.} \tag{4.3}$$

In other words, every φ in V admits at least a decomposition

$$\varphi = \varphi_1 + \varphi_2$$

and actually an infinite number of them if $V_1 \cap V_2 \neq \{0\}$.

Remark 4.1.

Everything we are going to say readily extends to the case when

$$V = V_1 + \cdots + V_m \ , m > 2 \ . \tag{4.4}$$

□

Remark 4.2.

Examples (for a stationnary situation without control) are given in R. Glowinski, J.L. Lions and O. Pironneau[GLP99]. □

We introduce now the natural decomposition of W (defined in (2.6) attached to (4.1), namely

$$\left| \begin{array}{l} W = W_1 + W_2 \ , \\ W_i = \{\varphi | \varphi \in L^2(0,T;V_i), \dfrac{\partial \varphi}{\partial t} \in L^2(0,T;H), \varphi(0) = 0\} \ . \end{array} \right. \tag{4.5}$$

Let s_i $(i = 1,2)$ be a continuous bilinear form on V (or on V_i) such that

$$\left| \begin{array}{l} s_i \textbf{ is symmetric} \text{ and } s_i(\varphi_i, \varphi_i) \geq s_{0i} \|\varphi_i\|^2 \quad \forall \ \varphi_i \in V_i \ , \\ s_{0i} > 0. \end{array} \right. \tag{4.6}$$

We then define $\forall \ \varphi, \hat{\varphi} \in W_i$,

$$\sigma_i(\varphi, \hat{\varphi}) = \gamma \int_0^T (\frac{\partial \varphi}{\partial t}, \frac{\partial \hat{\varphi}}{\partial t}) dt + \int_0^T \ s_i(\varphi, \hat{\varphi}) dt \ . \tag{4.7}$$

Given the **virtual controls** $\lambda_1, \lambda_2 \in W_1 \times W_2$, we define $y_1, y_2 \in W_1 \times W_2$ as the solution of

$$\left| \begin{array}{l} \sigma_1(y_1 - \lambda_1, \hat{y_1}) + \mathcal{A}_\gamma(\lambda_1 + \lambda_2, \hat{y_1}) = \displaystyle\int_0^T (Bv, \hat{y_1}) dt \quad \forall \ \hat{y_1} \in W_1 \ , \\ \sigma_2(y_2 - \lambda_2, \hat{y_2}) + \mathcal{A}_\gamma(\lambda_1 + \lambda_2, \hat{y_1}) = \displaystyle\int_0^T (Bv, \hat{y_2}) dt \quad \forall \ \hat{y_2} \in W_2 \ . \end{array} \right. \tag{4.8}$$

Remark 4.3.

It is obvious that, given v and λ_1, λ_2, the system (4.8) admits a unique solution. For instance y_1 is given by the solution of

$$\sigma_1(y_1, \hat{y_1}) = \sigma_1(\lambda_1, \hat{y_1}) - \mathcal{A}_\gamma(\lambda_1 + \lambda_2, \hat{y_1}) + \int_0^T (Bv, \hat{y_1}) dt \quad \forall \ \hat{y_1} \in W_1 \ .$$

□

Remark 4.4.

The equations (4.8) can be solved in parallel. □

Remark 4.5.

If one can choose λ_1, λ_2 such that

$$y_i = \lambda_i \tag{4.9}$$

then (4.8) is equivalent to

$$\mathcal{A}_\gamma(y_1 + y_2, \hat{y_1}) = \int_0^T (Bv, \hat{y_1})dt$$
$$\mathcal{A}_\gamma(y_1 + y_2, \hat{y_2}) = \int_0^T (Bv, \hat{y_2})dt$$

so that $y_1 + y_2 = y(= y_\gamma)$ the solution of (3.7). □

We now define the new cost function

$$\left| \begin{array}{l} \mathcal{J}(v, \lambda) = \frac{1}{2} \int_0^T \ \|v\|^2_{\mathcal{U}}\ dt + \frac{\beta}{2}\ \|y_1(T) + y_2(T) - y^T\ \|^2_H\ , \\ y_i \text{ solution of (4.8)}, \lambda = \{\lambda_1, \lambda_2\}\ . \end{array} \right. \tag{4.10}$$

According to Remark 4.5., we have

$$\inf_{y_i = \lambda_i} .\mathcal{J}(v, \lambda) = \inf J_\gamma(v)\ . \tag{4.11}$$

It is therefore natural to introduce a **penalty term** (in order to take care of "$y_i = \lambda_i$") as follows :

$$\left| \begin{array}{l} \mathcal{J}_\varepsilon(v, \lambda) = \frac{1}{2} \int_0^T \ \|v\|^2_{\mathcal{U}}\ dt + \frac{\beta}{2}\ \|y_1(T) + y_2(T) - y^T\ \|^2_H + \\ \qquad + \frac{1}{2\varepsilon}[\sigma_1(y_1 - \lambda_1) + \sigma_2(y_2 - \lambda_2)]\ ,\ \lambda = \{\lambda_1, \lambda_2\} \end{array} \right. \tag{4.12}$$

and to consider the problem

$$\inf .\mathcal{J}_\varepsilon(v, \lambda), \tag{4.13}$$

$$v \in L^2(0, T; \mathcal{U}),\ \lambda = \{\lambda_1, \lambda_2\} \in W_1 \times W_2\ .$$

We study now (4.13) and **we show it gives an approximation of** (3.9), **which is itself an approximation** (as $\gamma \to 0$) **of** (3.6).

Approximation results

We are going to show

Theorem 5.1. - *We assume that* (2.3), (4.6), (4.1), (4.2), (4.3) *hold true. The elliptic regularization parameter γ is fixed (arbitrarily small).*

(i) *For $\varepsilon > 0$ fixed, problem* (4.13) *admits a unique solution $v_\varepsilon, y_{i\varepsilon} - \lambda_{i\varepsilon}, i = 1, 2$.*

(ii) *As $\varepsilon \to 0$, one has*

$$\inf \mathcal{J}_\varepsilon(v, \lambda) \to \inf J_\gamma(v) = J_\gamma(v_\gamma)$$

$$v_\varepsilon \to v_\gamma \qquad \text{in} \quad L^2(0,T;\mathcal{U}) \quad \text{weakly}$$

(in fact $v_\varepsilon = v_{\varepsilon,\gamma}$). □

We prove Theorem 5.1. in several steps.

Step 1. - The existence of a solution of (4.13) is straightforward, provided we notice that we have informations on $y_i - \lambda_i$ rather that on y_i.

Step 2. - Given v, we compute $y(v) = y_\gamma(v)$ and we decompose $y(v)$ in, say, $y(v) = z_1 + z_2$, $z_i \in L^2(0,T;V_i)$.

Choosing $\lambda_i = z_i$, it follows that $y_i = z_i = \lambda_i$
so that

$$\inf \mathcal{J}_\varepsilon(v,\lambda) \le J_\gamma(v) \qquad \forall\, v, i.e.$$

$$\inf \mathcal{J}_\varepsilon(v,\lambda) \le \inf . J_\gamma(v) \qquad v \in L^2(0,T;\mathcal{U}). \tag{5.1}$$

□

Step 3. - It follows from (5.1) that, as $\varepsilon \to 0$,

$$v_\varepsilon \text{ remains in a bounded subset of } L^2(0,T;\mathcal{U}), \tag{5.2}$$

$$\sigma_i(y_{i\varepsilon} - \lambda_{i\varepsilon}) \le c\sqrt{\varepsilon}. \tag{5.3}$$

Step 4. - We use (4.8) with $y_{i\varepsilon}, \lambda_{i\varepsilon}$ but we do not write for a moment the indices "ε". Let $\eta_1 + \eta_2$ be an arbitrary decomposition of $y_1 + y_2$

$$y_1 + y_2 = \eta_1 + \eta_2, \quad \eta_i \in W_i \tag{5.4}$$

(of course $\eta_i = y_i$ if $V_1 \cap V_2 = \{0\}$).

We choose $\hat{y}_i = \eta_i$ in (4.8) and we add up the results.

We obtain

$$\sigma_1(y_1 - \lambda_1, \eta_1) + \sigma_2(y_2 - \lambda_2, \eta_2) + \mathcal{A}(\lambda_1 + \lambda_2, y_2 + y_2) = \int_0^T (Bv, y_1 + y_2)dt \tag{5.5}$$

We observe that (writing $\mathcal{A}_\gamma(\varphi,\varphi) = \mathcal{A}_\gamma(\varphi)$)

$$\left| \begin{aligned} &\mathcal{A}_\gamma(\lambda_1 + \lambda_2, y_1 + y_2) = \frac{1}{2}[\mathcal{A}_\gamma(\lambda_1 - y_1 + \lambda_2 - y_2, y_1 + y_2) + \mathcal{A}_\gamma(y_1 + y_2) + \\ &\qquad +(\mathcal{A}_\gamma(\lambda_1 + \lambda_2, y_1 - \lambda_1 + y_2 - \lambda_2) + \mathcal{A}_\gamma(\lambda_1 + \lambda_2)] \ge \\ &\qquad \ge c[\, \|y_1 + y_2\|_W^2 + \|\lambda_1 + \lambda_2\|_W^2] - c\sqrt{\varepsilon}\, [\, \|y_1 + y_2\|_W + \|\lambda_1 + \lambda_2\|_W]\ , \end{aligned} \right. \tag{5.6}$$

(where the c's denote various constants).

We also observe that

$$|\, \sigma_1(y_1 - \lambda_1, \eta_1) + \sigma_2(y_2 - \lambda_2, \eta_2)\, | \le c\sqrt{\varepsilon}\, (\|\eta_1\|_{W_1} + \|\eta_2\|_{W_2}). \tag{5.7}$$

We can choose η_1, η_2 in such a way that

$$\|\eta_1\|_{W\,1} + \|\eta_2\|_{W_2} \le c\|\eta_1 + \eta_2\|_W$$

so that (5.7) implies

$$|\ \sigma_1(y_1-\lambda_1,\eta_1)+\sigma_2(y_2-\lambda_2,\eta_2)\ | \le c\sqrt{\varepsilon}\ \|y_1+y_2\|_W\,. \tag{5.8}$$

It follows from (5.5) (5.6) and (5.8) that, as $\varepsilon \to 0$,

$$\|y_{1\varepsilon}+y_{2\varepsilon}\|_W + \|\lambda_{1\varepsilon}+\lambda_{2\varepsilon}\|_W \le c\,. \tag{5.9}$$

Step 5. - One verifies that one can then pass to the limit in ε in the equations (4.8) (where $\lambda_1=\lambda_{i\varepsilon}, y_i=y_{i\varepsilon}$). One extracts a subsequence $v_\varepsilon, \lambda_{i\varepsilon}, y_{i\varepsilon}$ such that

$$\begin{array}{lll} v_\varepsilon \to w & \text{in} \quad L^2(0,T;\mathcal{U}) & \text{weakly}\,, \\ y_{1\varepsilon}+y_{2\varepsilon} \to z & \text{in} \quad W & \text{weakly}\,, \\ y_{1\varepsilon}-\lambda_{i\varepsilon} \to 0 & \text{in} \quad W_i & (\text{like } \sqrt{\varepsilon}), \end{array}$$

and one obtains

$$\mathcal{A}_\gamma(z,\hat{y}_1)=\int_0^T (Bw,\hat{y}_1)dt \qquad \forall \hat{y}_1 \in W_1,$$

$$\mathcal{A}_\gamma(z,\hat{y}_2)=\int_0^T (Bw,\hat{y}_2)dt \qquad \forall \hat{y}_2 \in W_2,$$

so that $z=y(w)=y_\gamma(w)$.

One can also verify that

$$y_{1\varepsilon}(T)+y_{2\varepsilon}(T) \to z(T) \quad \text{in} \quad H \quad \text{weakly}\,.$$

Then

$$\mathcal{J}_\varepsilon(v_\varepsilon,\lambda_\varepsilon) \ge \frac{1}{2}\int_0^T \|v_\varepsilon\|^2_{\mathcal{U}}\,dt+\frac{\beta}{2}\ \|y_{1_\varepsilon}(T)+y_{2_\varepsilon}(T)-y^T\ \|^2_H$$

implies

$$\liminf \mathcal{J}_\varepsilon(v_\varepsilon,\lambda_\varepsilon) \ge J_\gamma(w)\,. \tag{5.10}$$

Comparing with (5.1), Theorem 5.1 follows. □

Algorithms

We proceed with the computation of the 1st variation of $\mathcal{J}_\varepsilon(v,\lambda)$, in fact of $\varepsilon\mathcal{J}_\varepsilon(v,\lambda)$. We have :

(6.1)

$$\left|\begin{aligned} &\delta(\varepsilon\mathcal{J}_\varepsilon(v,\lambda))=\varepsilon\textstyle\int_0^T(v,\delta v)_{\mathcal{U}}\,dt+\varepsilon\,\beta(y_1(T)+y_2(T)-y^T,\delta y_1(T)+\delta y_2(T))_H+\\ &\qquad\qquad +\sigma_1(y_1-\lambda_1,\delta y_1-\delta\lambda_1)+\sigma_2(y_2-\lambda_2,\delta y_2-\delta\lambda_2)\,. \end{aligned}\right.$$

It follows from (4.8) that

$$\sigma_1(\delta y_1-\delta\lambda_1,\hat{y}_1)+\mathcal{A}_\gamma(\delta\lambda_1+\delta\lambda_2,\hat{y}_1)=\int_0^T(B\delta v,\hat{y}_1)dt \tag{6.2}$$

and the analogous equation for $\sigma_2(\delta y_2 - \delta\lambda_2, \hat{y}_2)$. If we take $\hat{y}_1 = y_1 - \lambda_1$ in (6.2), and the analogous choice with the index "2", we obtain that

(6.3)
$$\left|\begin{array}{l} \sigma_1(y_1-\lambda_1,\delta y_1-\delta\lambda_1)+\sigma_2(y_2-\lambda_2,\delta y_2-\delta\lambda_2)= \\ \qquad = \int_0^T(B\delta v, y_1-\lambda_1+y_2-\lambda_2)dt-\mathcal{A}_\gamma(\delta\lambda_1+\delta\lambda_2,y_1-\lambda_1+y_2-\lambda_2)\ . \end{array}\right.$$

Let us introduce the adjoint $\mathcal{A}_\gamma^*$ of $\mathcal{A}_\gamma$:

$$\mathcal{A}_\gamma^*(\varphi,\hat{\varphi})=\mathcal{A}_\gamma(\hat{\varphi},\varphi)$$

and let us define $p_1, p_2 \in W_1 \times W_2$ by

$$\text{(6.4)}\qquad \left|\begin{array}{l} \sigma_1(p_1,\hat{p}_1)=\mathcal{A}_\gamma^*(y_1-\lambda_1+y_2-\lambda_2,\hat{p}_1)-\varepsilon\,\beta(y_1(T)+y_2(T)-y^T,p_1(T)) \\ \qquad\qquad\qquad\qquad\qquad\qquad\forall\,\hat{p}_1\in W_1 \\ \sigma_2(p_2,\hat{p}_2)=\mathcal{A}_\gamma^*(y_1-\lambda_1+y_2-\lambda_2,\hat{p}_2)-\varepsilon\,\beta(y_1(T)+y_2(T)-y^T,p_2(T)) \\ \qquad\qquad\qquad\qquad\qquad\qquad\forall\,\hat{p}_2\in W_2\ . \end{array}\right.$$

Then using (6.3) and (6.4) one obtains

$$\text{(6.5)}\qquad \left|\begin{array}{l} \delta(\varepsilon\mathcal{J}_\varepsilon(v,\lambda))=\displaystyle\int_0^T(\varepsilon v+B^*(y_1-\lambda_1+y_2-\lambda_2),\delta v)_{\mathcal{U}}\,dt- \\ \qquad\qquad -\sigma_1(p_1,\delta\lambda_1)-\sigma_2(p_2,\delta\lambda_2)\ , \end{array}\right.$$

where B^* is the adjoint of B defined by

$$\text{(6.6)}\qquad (B^*\,f,v)_{\mathcal{U}}=(f,Bv)\qquad\forall\,f\in V'\ ,\quad\forall v\in\mathcal{U}\ .$$

The simplest (if not the most efficient) algorithm one can deduce from (6.5) is then the following. Assuming that $v^n, \lambda_1^n, \lambda_2^n, y_1^n, y_2^n$ have been computed, define

$$\text{(6.7)}\qquad \left|\begin{array}{l} v^{n+1}=v^n-\rho(\varepsilon v^n+B^*(y_1^n-\lambda_1^n+y_2^n-\lambda_2^n)), \\ \lambda_1^{n+1}=\lambda_1^n+\rho\,p_1^n\ , \\ \lambda_2^{n+1}=\lambda_2^n+\rho\,p_2^n\ , \end{array}\right.$$

where $\rho > 0$ is chosen small enough.

Compute y_1^{n+1}, y_2^{n+1} (in parallel) by (4.8), where one uses v^{n+1}, λ_i^{n+1}. Then compute p_1^{n+1}, p_2^{n+1} (in parallel by (6.4) and proceed. □

Remark 6.1.

More powerful algorithms (conjugate gradients) are given, for similar situations, without control, in R. Glowinski, J.L. Lions and O. Pironneau[GLP99]. □

Remark 6.2.

As we already said, everything extends to the situation when

$$V = V_1 + \cdots + V_m, m > 2.$$

□

Remark 6.3.

The "elliptic regularization parameter" γ is **fixed** ("small"). What happens to the above algorithms when $\gamma \to 0$ is an open question. □

Remarks and extensions

Remark 7.1.

In principle all the methods introduced here apply to problems **without control**.

But one is led to 2-points Boundary Value Problems (BVP) **in time**, not a wise thing to do. Of course the situation is different when effective control is present, since there 2 points BVP are needed anyway (one way or the other). □

Remark 7.2.

In case there are constraints on v then, of course, the algorithms in previous section should be modified accordingly. □

Remark 7.3.

All the methods presented here can apply, with suitable modifications, for systems modelled by

non linear PDE

or

hyperbolic (or Petrowsky type) models

or

Schroedinger models

or coupled models. We shall return to these questions on other occasions. □

Remark 7.4.

A systematic presentation of other decomposition methods for the control of distributed systems is given in the paper of J.L. Lions[Lio00]. □

References

[Ben97]J.D. Benamou. Décomposition de domaine pour le contrôle de systèmes gouvernés par des équations d'évolution. *C. R. Acad. Sci. Paris, Série I*, 324:1065–1070, 1997.

[Ben98]J.D. Benamou. Domain decomposition, optimal control of systems governed by partial differential equations and synthesis of feedback laws. *J. Opt. Theory Appl.*, 99, 1998.

[BLT94]A. Bensoussan, J.L. Lions, and R. Temam. Sur les méthodes de déomposition, de décentralisation et de coordination et applications. In J.L. Lions and G.I. Marchuk, editors, *Méthodes Mathématiques de l'Informatique*, pages 133–257. Dunod, Paris, 1994.

[Des91]B. Despres. *Méthodes de décomposition de domaine pour les problèmes de propagation d'ondes en régimes harmoniques.* PhD thesis, Paris IX, 1991.

[GLP99]R. Glowinski, J.L. Lions, and O. Pironneau. Decomposition of energy spaces and applications. C.R.A.S., Paris, 1999.

[JD97]J.D.Benamou and B. Deprés. A domain decomposition method for the helmholtz equation and related optimal control problems. *J. of Comp. Physics*, 136:68–82, 1997.

[Lio61]J.L. Lions. *Equations Différentielles opérationnelles.* Springer, 1961.

[Lio63]J.L. Lions. Equations différentielles opérationnelles dans les espaces de hilbert. Cours CIME, Varenna, 30 Mai-8 Juin, 1963.

[Lio68]J.L. Lions. *Contrôle optimal des systèmes gouvernés par des équations aux dérivées partielles.* Paris. Dunod, Gauthier Villars, 1968.

[Lio00]J.L. Lions. Virtual and effective control for distributed systems and decomposition of everything. J. d'Analyse Math. Jerusalem, 2000.

[LL00]J.E. Lagnese and G. Leugering. Dynamic domain decomposition in approximate and exact boundary control in problems of transmission for wave equations. to appear, 2000.

[LP98a]J.L. Lions and O. Pironneau. Algorithmes parallèles pour la solution des problémes aux limites. *C.R.A.S., Paris*, 327:947–952, 1998.

[LP98b]J.L. Lions and O. Pironneau. Sur le contrôle parallèle des systèmes distribués. *C.R.A.S., Paris*, 327:993–998, 1998.

[LP99a]J.L. Lions and O. Pironneau. Contrôle virtuel, répliques et décomposition d'opérateurs. *C.R.A.S.*, 1999.

[LP99b]J.L. Lions and O. Pironneau. Domain decomposition method for CAD. *C.R.A.S., Paris*, 328:73–80, 1999.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau,

5. On Schwarz Methods for Monotone Elliptic PDEs

S. H. LUI[1]

Introduction

The Schwarz Alternating Method was devised by H. A. Schwarz more than one hundred years ago to solve linear boundary value problems. It has garnered interest recently because of its potential as an efficient algorithm for parallel computers. See [Lio88], and [Lio89], the recent reviews [CM94], [LT94], and [XZ98], and the books [SBG96] and [QV99]. The literature for nonlinear problems is rather sparse. Besides Lions' works, see also [Bad91], [ZH92], [CD94], [Tai94], [TE98], [TX01], [Pao95], [Xu96], [DH97], [Lui00], [Lui01], and references therein. The effectiveness of Schwarz methods for nonlinear problems (especially those in fluid mechanics) has been demonstrated in many papers. See proceedings of the annual domain decomposition conferences.

This paper is a continuation of previous works by this author attempting to survey various classes of nonlinear elliptic PDEs for which Schwarz methods are applicable. We consider elliptic PDEs amenable to analysis by the monotone method (also known as the method of subsolutions and supersolutions).

The paper [KC67] was among the first to employ the monotone method to solve boundary value problems. Subsequent works by these two authors as well as by [Sat72], [Ama76], and many others have made this method into one of the important tools in nonlinear analysis. See [Pao92] for a very complete reference with many applications as well as a good bibliography. [Lio89] shows the convergence of a multiplicative Schwarz method for the Poisson's equation using the monotone method. Here, we prove convergence for an additive Schwarz method on finitely many subdomains for scalar as well as coupled systems of nonlinear elliptic PDEs. Our results on coupled systems can be applied to the three types of Lotka-Volterra models in population biology: competition, cooperation and predator-prey.

In the following section, we indicate convergence of two Schwarz methods for a class of scalar nonlinear elliptic PDEs. This is followed by a treatment of the so-called quasi-monotone non-increasing case of a coupled system of PDEs on finitely many subdomains. In the remaining part of this introduction, we set some notations.

Let Ω be a bounded, connected domain in $\mathbb{R}^N$ with a smooth boundary. Suppose Ω is composed of $m \geq 2$ subdomains, that is, $\Omega = \Omega_1 \cup \cdots \cup \Omega_m$. The boundary of each subdomain is also assumed to be smooth. Let $X = C^\alpha(\overline{\Omega}) \cap C^2(\Omega)$ for some $0 < \alpha < 1$. We shall look for solutions of PDEs lying in this space.

[1] Department of Mathematics, HKUST, Clear Water Bay, Kowloon, Hong Kong (email: shlui@ust.hk). This work was supported in part by a grant from the RGC of HKSAR, China (HKUST6171/99P).

Scalar Equations

Consider the PDE

$$-\triangle u = f(x,u) \text{ on } \Omega, \qquad u = h \text{ on } \partial\Omega. \tag{1}$$

A smooth function $\underline{u} \in X$ is a *subsolution* of the above PDE if

$$-\triangle \underline{u} - f(x,\underline{u}) \leq 0 \text{ on } \Omega \quad \text{and} \quad \underline{u} \leq h \text{ on } \partial\Omega.$$

Similarly, a *supersolution* is one which satisfies the above with both inequalities reversed.

Let us now record the assumptions for the above PDE. Suppose that it has a subsolution $\underline{u}$ and a supersolution $\overline{u}$ which satisfy $\underline{u} \leq \overline{u}$ on Ω. Define the sector of smooth functions

$$\mathcal{A} \equiv \{u \in X,\ \underline{u} \leq u \leq \overline{u} \text{ on } \overline{\Omega}\}.$$

Assume f is a smooth (Holder continuous) function defined on $\overline{\Omega} \times \mathcal{A}$ and h is a smooth function defined on the boundary. In addition, suppose there exists some bounded non-negative function c defined on Ω so that

$$-c(x)(u-v) \leq f(x,u) - f(x,v), \qquad x \in \Omega, \quad v \leq u \in \mathcal{A}.$$

With these assumptions, it is known (section 3.2 in [Pao92]) that the PDE has a (not necessarily unique) solution in the sector $\mathcal{A}$.

We begin with a comparison lemma.

Lemma 1 *Suppose S is an open set. Let $w \in H^1(S) \cap C(\overline{S})$ satisfy*

$$\int_S (\nabla w \cdot \nabla \phi + cw\phi) \geq 0, \qquad \forall \text{ non-negative } \phi \in H_0^1(S) \tag{2}$$

and $w \geq 0$ on ∂S. Then $w \geq 0$ on $\overline{S}$.

We now show convergence of a (multiplicative) Schwarz sequence for the PDE (1) for the two subdomain case. For convenience, we suppress the dependence of f on $x \in \Omega$. Note that each subdomain problem is a linear one. Despite the possibility of multiple solutions, the Schwarz iteration always converges to a specific solution.

Theorem 1 *Let $u^{(0)} = u^{(-\frac{1}{2})} = \underline{u}$ on $\overline{\Omega}$ with $\underline{u} = h$ on $\partial\Omega$. Define the Schwarz sequence by ($n \geq 0$)*

$$-\triangle u^{(n+\frac{1}{2})} + cu^{(n+\frac{1}{2})} = f(u^{(n-\frac{1}{2})}) + cu^{(n-\frac{1}{2})} \text{ on } \Omega_1, \qquad u^{(n+\frac{1}{2})} = u^{(n)} \text{ on } \partial\Omega_1,$$

$$-\triangle u^{(n+1)} + cu^{(n+1)} = f(u^{(n)}) + cu^{(n)} \text{ on } \Omega_2, \qquad u^{(n+1)} = u^{(n+\frac{1}{2})} \text{ on } \partial\Omega_2.$$

Here, $u^{(n+\frac{1}{2})}$ is defined as $u^{(n)}$ on $\overline{\Omega} \setminus \overline{\Omega}_1$ and $u^{(n+1)}$ is defined as $u^{(n+\frac{1}{2})}$ on $\overline{\Omega} \setminus \overline{\Omega}_2$. Then $u^{(n+\frac{i}{2})} \to u$ in $C^2(\overline{\Omega}_i)$, $i = 1, 2$, where u is a solution of (1) in $\mathcal{A}$. If v is any solution in $\mathcal{A}$, then $u \leq v$ on $\overline{\Omega}$.

If $u^{(0)} = u^{(-\frac{1}{2})} = \overline{u}$ on $\overline{\Omega}$ with $\overline{u} = h$ on $\partial\Omega$ instead, then the same conclusion holds except that $u \geq v$ on $\overline{\Omega}$.

Sketch of Proof: We only consider the case $u^{(0)} = \underline{u}$ with $\underline{u} = h$ on $\partial\Omega$. The proof can be divided into four steps. First, we demonstrate that the sequence is monotone:

$$\underline{u} \leq u^{(n-\frac{1}{2})} \leq u^{(n)} \leq u^{(n+\frac{1}{2})} \leq \overline{u} \text{ on } \overline{\Omega}, \qquad n \geq 0. \tag{3}$$

Since the sequences are bounded above, the following limits are well defined on $\overline{\Omega}$

$$\lim_{n\to\infty} u^{(n+\frac{1}{2})} = u_1, \qquad \lim_{n\to\infty} u^{(n)} = u_2.$$

In the second step, we prove that the function u_i satisfies the same PDE on Ω_i using an elliptic regularity argument (see p. 102 in [Pao92]). We can also infer that the convergence to u_i is in the sense of $C^2(\overline{\Omega}_i)$. In the third step, we prove that $u_1 = u_2$ on $\overline{\Omega}$ which follows directly from (3). Define $u = u_1$. Then u is a solution of (1). Finally, if v is any other solution in $\mathcal{A}$, replace $\overline{u}$ by v in the above steps to obtain $u \leq v$ on $\overline{\Omega}$. This completes the sketch of the proof.

The above Schwarz iteration is an adaptation of the classical Schwarz iteration for the Poisson's equation. The next Schwarz method is called an additive Schwarz method. It generalizes the additive method for linear PDEs first introduced in [DW87]. It is sometimes preferable to the (multiplicative) Schwarz method above because the subdomain PDEs are independent and hence can be solved in parallel. We consider the general m-subdomain case.

Theorem 2 *Let $u^{(0)} = u_i^{(0)} = \underline{u}$ on $\overline{\Omega}$, $i = 1, \cdots, m$ with $\underline{u} = h$ on $\partial\Omega$. Define the additive Schwarz sequence by $(n \geq 1)$*

$$-\triangle u_i^{(n)} + cu_i^{(n)} = f(u_i^{(n-1)}) + cu_i^{(n-1)} \text{ on } \Omega_i, \qquad u_i^{(n)} = u^{(n-1)} \text{ on } \partial\Omega_i, \quad i = 1, \cdots, m.$$

Here, $u_i^{(n)}$ is defined as $u^{(n-1)}$ on $\overline{\Omega} \setminus \overline{\Omega}_i$ and

$$u^{(n)}(x) = \max_{1\leq i\leq m} u_i^{(n)}(x), \qquad x \in \overline{\Omega}.$$

Then $u_i^{(n)} \to u$ in $C^2(\overline{\Omega}_i)$, $i = 1, \cdots, m$ where u is a solution of (1) in $\mathcal{A}$. If v is any solution in $\mathcal{A}$, then $u \leq v$ on $\overline{\Omega}$.

If $u^{(0)} = u_i^{(0)} = \overline{u}$ on $\overline{\Omega}$ with $\overline{u} = h$ on $\partial\Omega$ instead, then the same conclusion holds except that $u \geq v$ on $\overline{\Omega}$.

Sketch of Proof: The details of this proof are quite similar to those of the last proof. Assume $u^{(0)} = \underline{u}$. The following monotone properties hold:

$$\underline{u} \leq u_i^{(n)} \leq u_i^{(n+1)} \leq \overline{u} \text{ on } \overline{\Omega}_i, \qquad \underline{u} \leq u^{(n)} \leq u^{(n+1)} \leq \overline{u} \text{ on } \overline{\Omega}, \tag{4}$$

$$u^{(n)} \leq u_i^{(n+1)} \text{ on } \overline{\Omega}, \qquad i = 1, \cdots, m. \tag{5}$$

The inequalities in (4) can be shown in a straightforward manner by induction using the maximum principle. To show the second set of inequalities in (4), take a fixed n and $x \in \Omega$. Then there is some integer i in between 1 and m inclusive so that $u^{(n)}(x) = u_i^{(n)}(x) \leq u_i^{(n+1)}(x) \leq u^{(n+1)}(x)$.

The inequality (5) can also be shown by induction. This can be done using the following (nontrivial) inequality

$$\int_{\Omega_i} (\nabla u^{(n)} \cdot \nabla \phi + c u^{(n)} \phi) \leq \int_{\Omega_i} \left(f(u^{(n-1)}) + c u^{(n-1)} \right) \phi, \qquad \forall \text{ non-negative } \phi \in H_0^1(\Omega_i).$$

which says that $u^{(n)}$ is a subsolution in some weak sense.

Next, we define on $\overline{\Omega}$, for $i = 1, \cdots, m$,

$$\lim_{n\to\infty} u_i^{(n)} = u_i, \qquad \lim_{n\to\infty} u^{(n)} = u_0$$

and show using elliptic regularity theory that the limit u_i satisfies the same PDE on $\Omega_i, \ i = 1, \cdots, m$ and that the convergence to u_i is in the sense of $C^2(\Omega_i)$. We have $u_i \leq u_0$ on $\overline{\Omega}, \ i = 1, \cdots, m$. By (5), we have for any $j, \ u_0 \leq u_j \leq u_0 \leq u_i$. From these inequalities, we conclude that $u_i = u_j = u_0, \ 1 \leq i, j \leq m$. Define u to be this common function which must be a solution of (1) in $\mathcal{A}$. The proof of $u \leq v$ for any solution of (1) in $\mathcal{A}$ is the same as before.

Quasi-monotone Non-increasing Coupled Systems

Consider the system

$$-\triangle u = f(u, v), \qquad -\triangle v = g(u, v) \qquad \text{on } \Omega, \tag{6}$$

$$u = r, \qquad v = s \qquad \text{on } \partial\Omega.$$

The pairs of smooth functions $(\underline{u}, \underline{v})$ and $(\overline{u}, \overline{v})$ are called *subsolution and supersolution pairs* if they satisfy

$$-\triangle \underline{u} - f(\underline{u}, \overline{v}) \leq 0 \leq -\triangle \overline{u} - f(\overline{u}, \underline{v}) \text{ on } \Omega,$$

$$-\triangle \underline{v} - g(\overline{u}, \underline{v}) \leq 0 \leq -\triangle \overline{v} - g(\underline{u}, \overline{v}) \text{ on } \Omega, \text{ and}$$

$$\underline{u} \leq r \leq \overline{u}, \qquad \underline{v} \leq s \leq \overline{v} \quad \text{on } \partial\Omega.$$

Furthermore, they are said to be *ordered* if

$$\underline{u} \leq \overline{u}, \qquad \underline{v} \leq \overline{v} \quad \text{on } \overline{\Omega}.$$

Define the sector

$$\mathcal{A} \equiv \left\{ \begin{bmatrix} u \\ v \end{bmatrix}, \ u, v \in X, \ \underline{u} \leq u \leq \overline{u}, \ \underline{v} \leq v \leq \overline{v} \text{ on } \overline{\Omega} \right\}.$$

Suppose $f, g \in C^1(\mathcal{A})$. Our system of PDEs is called *quasi-monotone non-increasing* if

$$\frac{\partial f}{\partial v}, \ \frac{\partial g}{\partial u} \leq 0 \text{ on } \mathcal{A}. \tag{7}$$

Suppose our system of PDEs is quasi-monotone non-increasing. Then it can be shown (section 8.4 in [Pao92]) that it has a solution (u, v) in $\mathcal{A}$. Without further assumptions, it may have more than one solution. Despite this, the following additive Schwarz sequence converges for an appropriately chosen initial guess. Note that the subdomain problems at each iteration are independent and are decoupled.

Theorem 3 *Suppose the system (6) is quasi-monotone non-increasing and let $(\underline{u}, \underline{v})$ and $(\overline{u}, \overline{v})$ be ordered subsolution and supersolution pairs. Consider any non-negative functions $c, d \in C^\alpha(\overline{\Omega})$ so that*

$$\frac{\partial f(u,v)}{\partial u} \geq -c, \qquad \frac{\partial g(u,v)}{\partial v} \geq -d, \qquad (u,v) \in \mathcal{A}. \tag{8}$$

For $i = 1, \cdots, m$, let

$$u^{(0)} = u_i^{(0)} = \underline{u} \text{ and } v^{(0)} = v_i^{(0)} = \overline{v} \text{ on } \overline{\Omega} \text{ with } \underline{u} = r \text{ and } \overline{v} = s \text{ on } \partial\Omega. \tag{9}$$

Define the Schwarz sequence for $i = 1, \cdots, m$ and $n \geq 1$

$$-\triangle u_i^{(n)} + c u_i^{(n)} = f(u_i^{(n-1)}, v_i^{(n-1)}) + c u_i^{(n-1)} \text{ on } \Omega_i, \qquad u_i^{(n)} = u^{(n-1)} \text{ on } \partial\Omega_i$$

$$-\triangle v_i^{(n)} + d v_i^{(n)} = g(u_i^{(n-1)}, v_i^{(n-1)}) + d v_i^{(n-1)} \text{ on } \Omega_i, \qquad v_i^{(n)} = v^{(n-1)} \text{ on } \partial\Omega_i.$$

Here, $u_i^{(n)}$ and $v_i^{(n)}$ are defined as $u^{(n-1)}$ and $v^{(n-1)}$, respectively, on $\overline{\Omega} \setminus \overline{\Omega}_i$ while

$$u^{(n)}(x) = \max_{1 \leq i \leq m} u_i^{(n)}(x), \quad v^{(n)}(x) = \min_{1 \leq i \leq m} v_i^{(n)}(x) \qquad \text{on } \overline{\Omega}.$$

Then $u_i^{(n)} \to \underline{u}_0$ and $v_i^{(n)} \to \overline{v}_0$ in $C^2(\Omega_i)$, $i = 1, \cdots, m$, where $(\underline{u}_0, \overline{v}_0)$ is a solution of (6) in $\mathcal{A}$. If (u, v) is any solution in $\mathcal{A}$, then $\underline{u}_0 \leq u$ and $v \leq \overline{v}_0$.

If $u^{(0)} = u_i^{(0)} = \overline{u}$ and $v^{(0)} = v_i^{(0)} = \underline{v}$ on $\overline{\Omega}$ with $\overline{u} = r$ and $\underline{v} = s$ on $\partial\Omega$ replace the assumption (9), then the above Schwarz sequence satisfies $u_i^{(n)} \to \overline{u}_0$ and $v_i^{(n)} \to \underline{v}_0$ in $C^2(\Omega_i)$, $i = 1, \cdots, m$, where $(\overline{u}_0, \underline{v}_0)$ is also a solution of (6) in $\mathcal{A}$. If (u, v) is any solution in $\mathcal{A}$, then $u \leq \overline{u}_0$ and $v \geq \underline{v}_0$.

Sketch of Proof: We only consider the case where $u^{(0)} = \underline{u}$ and $v^{(0)} = \overline{v}$. The proof can be divided into four steps. We first show that the following monotone properties hold on $\overline{\Omega}$ for $i = 1, \cdots, m$,

$$\underline{u} \leq u_i^{(n)} \leq u_i^{(n+1)} \leq \overline{u}, \qquad u^{(n)} \leq u^{(n+1)}, \qquad u^{(n)} \leq u_i^{(n+1)} \tag{10}$$

and

$$\underline{v} \leq v_i^{(n+1)} \leq v_i^{(n)} \leq \overline{v}, \qquad v^{(n+1)} \leq v^{(n)}, \qquad v_i^{(n+1)} \leq v^{(n)}. \tag{11}$$

Since the sequences are bounded, the following limits on $\overline{\Omega}$ are well defined

$$\lim_{n\to\infty} u_i^{(n)} = \underline{u}_i, \qquad \lim_{n\to\infty} v_i^{(n)} = \overline{v}_i \qquad i = 1, \cdots, m,$$

and

$$\lim_{n\to\infty} u^{(n)} = \underline{u}_0, \qquad \lim_{n\to\infty} v^{(n)} = \overline{v}_0.$$

In the second step, we prove, using a similar elliptic regularity argument as before, that the limit functions satisfy the following PDEs on Ω_i:

$$-\triangle \underline{u}_i = f(\underline{u}_i, \overline{v}_i), \qquad -\triangle \overline{v}_i = g(\underline{u}_i, \overline{v}_i), \qquad i = 1, \cdots, m, \tag{12}$$

and that convergence to $\underline{u}_i$ and to $\overline{v}_i$ is in the sense of $C^2(\Omega_i)$. Third, we demonstrate that the functions $\underline{u}_i$ are identical. This follows because from (10) and the definition of $u^{(n)}$,

$$u_i^{(n)} \le u^{(n)} \le u_j^{(n+1)} \le u^{(n+1)} \le u_i^{(n+2)}, \qquad 1 \le i, j \le m.$$

Take the limit to obtain $\underline{u}_i = \underline{u}_j = \underline{u}_0$ on $\overline{\Omega}$. Similarly, we use (11) to show $\overline{v}_i = \overline{v}_j = \overline{v}_0$ on $\overline{\Omega}$ for $1 \le i, j \le m$. From (12), it follows that $(\underline{u}_0, \overline{v}_0)$ is a solution of (6).

Fourth, we prove that any solution (u, v) of (6) in $\mathcal{A}$ must satisfy

$$\underline{u}_0 \le u \quad \text{and} \quad v \le \overline{v}_0 \text{ on } \Omega_i. \tag{13}$$

This follows from the observation that $(\underline{u}, v)$ and $(u, \overline{v})$ form subsolution and supersolution pairs. Apply the above result to establish (13).

One example where a quasi-monotone non-increasing system occurs is the Lotka-Volterra competition model

$$-\triangle u = u(a_1 - b_1 u - c_1 v), \qquad -\triangle v = v(a_2 - b_2 u - c_2 v).$$

Here u, v stand for the population of two species competing for the same food sources and/or territories and all other variables are positive constants.

Similarly, it can be shown that the additive Schwarz method converges for other types of coupled systems. For instance, a quasi-monotone non-decreasing system is one where $f_v, g_u \ge 0$ on $\mathcal{A}$ in place of (7). (The definition of subsolution and supersolution pairs is slightly different though.) One example where a quasi-monotone non-decreasing system occurs is the Lotka-Volterra cooperating model

$$-\triangle u = u(a_1 - b_1 u + c_1 v), \qquad -\triangle v = v(a_2 + b_2 u - c_2 v).$$

Here u, v stand for the population of two species which have a symbiotic relationship and all other variables are positive constants.

A third class of coupled systems, known as mixed quasi-monotone is one where $f_v, -g_u \le 0$ on $\mathcal{A}$ in place of (7). Using essentially the same technique, one can show that the additive Schwarz method also works for this class of problems as well. One example where a mixed quasi-monotone system occurs is the Lotka-Volterra predator-prey model

$$-\triangle u = u(a_1 - b_1 u - c_1 v), \qquad -\triangle v = v(a_2 + b_2 u - c_2 v).$$

Here u stands for the population of a prey while v denotes the population of a predator and all other variables are positive constants.

References

[Ama76]H. Amann. Fixed point equations and nonlinear eigenvalue problems in ordered banach spaces. *SIAM Rev.*, 18:620–709, 1976.

[Bad91]L. Badea. On the schwarz alternating method with more than two subdomains for nonlinear monotone problems. *SIAM J. N. A.*, 28:179–204, 1991.

[CD94]X. C. Cai and M. Dryja. Domain decomposition methods for monotone nonlinear elliptic problems. In D. Keyes and J. Xu, editors, *Domain decomposition methods in scientific and engineering computing*, pages 335–360. AMS, 1994.

[CM94]Tony F. Chan and Tarek P. Mathew. Domain decomposition algorithms. In *Acta Numerica 1994*, pages 61–143. Cambridge University Press, 1994.

[DH97]M. Dryja and W. Hackbusch. On the nonlinear domain decomposition method. *BIT*, pages 296–311, 1997.

[DW87]Maksymilian Dryja and Olof B. Widlund. An additive variant of the Schwarz alternating method for the case of many subregions. Technical Report 339, also Ultracomputer Note 131, Department of Computer Science, Courant Institute, 1987.

[KC67]H. B. Keller and D. S. Cohen. Some positone problems suggested by nonlinear heat generation. *J. Math. Mech.*, 16:1361–1376, 1967.

[Lio88]Pierre-Louis Lions. On the Schwarz alternating method. I. In Roland Glowinski, Gene H. Golub, Gérard A. Meurant, and Jacques Périaux, editors, *First International Symposium on Domain Decomposition Methods for Partial Differential Equations*, pages 1–42, Philadelphia, PA, 1988. SIAM.

[Lio89]Pierre Louis Lions. On the Schwarz alternating method. II. In Tony Chan, Roland Glowinski, Jacques Périaux, and Olof Widlund, editors, *Domain Decomposition Methods*, pages 47–70, Philadelphia, PA, 1989. SIAM.

[LT94]Patrick Le Tallec. Domain decomposition methods in computational mechanics. In J. Tinsley Oden, editor, *Computational Mechanics Advances*, volume 1 (2), pages 121–220. North-Holland, 1994.

[Lui00]S. H. Lui. On schwarz alternating methods for nonlinear elliptic pdes. *SISC*, 2000.

[Lui01]S. H. Lui. On schwarz alternating methods for the incompressible Navier-Stokes equations. *SISC*, 2001.

[Pao92]C. V. Pao. *Nonlinear parabolic and elliptic equations.* Plenum Press, 1992.

[Pao95]C. V. Pao. Block monotone iterative methods for numerical solutions of nonlinear elliptic equations. *Numer. Math.*, 72:239–262, 1995.

[QV99]A. Quarteroni and A. Valli. *Domain Decomposition Methods for Partial Differential Equations.* Oxford Science Publications, 1999.

[Sat72]D. H. Sattinger. Monotone methods in nonlinear elliptic and parabolic boundary balue problems. *Indianna Univ. Math. J.*, 21:979–1000, 1972.

[SBG96]Barry F. Smith, Petter E. Bjørstad, and William Gropp. *Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations.* Cambridge University Press, 1996.

[Tai94]X. C. Tai. Domain decomposition for linear and nonlinear elliptic problems via function or space decomposition. In D. Keyes and J. Xu, editors, *Domain decomposition methods in scientific and engineering computing*, pages 335–360. AMS, 1994.

[TE98]X.-C. Tai and M. S. Espedal. Rate of convergence of some space decomposition method for linear and non-linear elliptic problems. *SIAM J. Numer. Anal.*, 35:1558–1570, 1998.

[TX01]X. C. Tai and J. Xu. Global convergence of subspace correction methods for convex optimization problems. *Math. Comput.*, 2001.

[Xu96]J. Xu. Two-grid discretization techniques for linear and nonlinear pdes. *SIAM J. N. A.*, 33:1759–1777, 1996.

[XZ98]J. Xu and J. Zou. Some nonoverlapping domain decomposition methods. *SIAM Rev*, 40:857–914, 1998.

[ZH92]J. Zou and H. C. Huang. Algebraic subproblem decomposition methods and parallel algorithms with monotone convergence. *J. Comput. Math.*, 10:47–59, 1992.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau, ©2001 DDM.org

6. Operator Theoretical Analysis to Domain Decomposition Methods

Norikazu SAITO[1], Hiroshi FUJITA[2]

Introduction

The purpose of the present paper is to give a brief summary of our recent study on the domain decomposition method from an operator theoretical point of view. There are a large number of works devoted to the mathematical analysis of the domain decomposition methods. Most of these works carry out the convergence analysis without any assumptions of general nature on the geometry of the decomposition. However, from the viewpoint of mathematical theory as well as from that of applications in science and engineering, we are seriously interested in the effect of relationships between the rate of convergence of iterations and the geometric shape of decomposed domains. Moreover, the choice of relaxation parameters is of importance. Our method enables us to get explicit convergence factors under some assumptions on geometric shapes of decomposed domains. Furthermore, our convergence theorems give information on the choice of relaxation parameters which guarantees a fast convergence.

The problem considered in this paper is well discussed in the monograph by A. Quarteroni and A. Valli (Domain Decomposition Methods for Partial Differential Equations, Oxford, 1999), and the results described here may be said to be particular cases of theorems presented in their monograph. However, the advantage of employing our method is already described above.

We shall present a rough sketch of the method of analysis and theorems without the proofs; for the complete proofs, we refer to [Fuj97], [FKKN96], [FFS98] and [FS97].

Model Problem

In order to fix the idea, let $\Omega \subset \mathbf{R}^2$ be a bounded domain with a Lipschitz boundary Γ, and consider the Poisson equation:

$$-\triangle u = f \text{ in } \Omega, \qquad u = \beta \text{ on } \Gamma. \tag{1}$$

We assume that $f \in L^2(\Omega)$ and $\beta \in H^{1/2}(\Gamma)$. The exact solution of (1) is denoted by $\tilde{u}$.

We divide the target domain Ω into two disjoint subdomains Ω_1 and Ω_2 by a smooth simple curve γ;

$$\overline{\Omega} = \overline{\Omega_1 \cup \Omega_2}, \quad \Omega_1 \cap \Omega_2 = \emptyset.$$

We assume that γ connects transversally two points on Γ. The outer unit normal vector to the boundary of a domain in consideration is denoted by n. If necessary, by

[1] International Institute for Advanced Studies. nsaito@kurims.kyoto-u.ac.jp.

[2] The Research Institute of Educational Development, Tokai University. hfujita@yoyogi.ycc.u-tokai.ac.jp

ν we indicate the one to γ outgoing from Ω_1. Put $\Gamma_1 = \partial\Omega_1 \backslash \gamma$ and $\Gamma_2 = \partial\Omega_2 \backslash \gamma$. The curve γ is called the artificial boundary.

We consider the following domain decomposition algorithm which is well-known as the Dirichlet-Neumann (DN) method. Take a function $\mu^{(0)}$ defined on γ as the initial guess to $\tilde{u}|_\gamma$. Then, we successively generate $u_1^{(k)}$, $u_2^{(k)}$ and $\mu^{(k+1)}$, for $k = 0, 1, 2, \cdots$, by solving

$$\begin{cases} -\triangle u_1^{(k)} &= f & \text{in } \Omega_1, \\ u_1^{(k)} &= \beta & \text{on } \Gamma_1, \\ u_1^{(k)} &= \mu^{(k)} & \text{on } \gamma, \end{cases}$$

$$\begin{cases} -\triangle u_2^{(k)} &= f & \text{in } \Omega_2, \\ u_2^{(k)} &= \beta & \text{on } \Gamma_2, \\ \dfrac{\partial u_2^{(k)}}{\partial n} &= -\dfrac{\partial u_1^{(k)}}{\partial \nu} & \text{on } \gamma, \end{cases}$$

The value of $\mu^{(k)}$ is adapted by

$$\mu^{(k+1)} = (1-\theta)\mu^{(k)} + \theta u_2^{(k)}|_\gamma,$$

where θ is the relaxation parameter subject to $0 < \theta < 1$.

Notation. The basic Hilbert space in our consideration is $X = L^2(\gamma)$. The usual $L^2(\gamma)$ inner product and norm are denoted by $(\cdot,\cdot)_X$ and $\|\cdot\|_X$, respectively. The space $V = H_{00}^{1/2}(\gamma)$ is familiar (See, for example, [LM72]), and the norm of V is denoted by $\|\cdot\|_V$. For any $\xi \in V$, a solution in $H^1(\Omega_1)$ of the harmonic problem

$$\triangle w = 0 \text{ in } \Omega_1, \quad w = 0 \text{ on } \Gamma_1, \quad w = \xi \text{ on } \gamma$$

is called the harmonic extension of ξ into Ω_1 and is denoted by $w = H_1\xi$. The harmonic extension $H_2\xi$ of ξ into Ω_2 is defined in the similar manner. As a consequence of the trace theorem (Theorem 1.5.2.3, [Gri85]) and the elliptic estimates, we have

$$C\|\xi\|_V \le \|\nabla H_i \xi\|_{L^2(\Omega_i)} \le C'\|\xi\|_V \quad (\forall \xi \in V) \tag{2}$$

with domain constants $C > 0$ and $C' > 0$, for $i = 1, 2$.

Amplification Operator for the Error

It is easy to derive that the error $\xi^{(k)} = \mu^{(k)} - \tilde{u}|_\gamma$ can be expressed as

$$\xi^{(k+1)} = (1-\theta)\xi^{(k)} - \theta S_2^{-1} S_1 \xi^{(k)}. \tag{3}$$

Here S_1 and S_2 stand for the Steklov-Poincaré (SP) operators corresponding to Ω_1 and Ω_2, respectively. The formal definition of S_1 is

$$S_1\xi = \frac{\partial(H_1\xi)}{\partial\nu}\Big|_\gamma$$

for $\xi \in V$. S_2 is also defined in the similar way. Actually, thought Kato's representation theorems concerning unbounded quadratic forms in a Hilbert space ([Kat76]), we have:

1. S_i is a positive and self-adjoint operator, and $S_i^{1/2}$ is so too.
2. The domain $D(S_i^{1/2})$ of $S_i^{1/2}$ coincides with V.
3. The identity $\|S_i^{1/2}\xi\|_X = \|\nabla H_i \xi\|_{L^2(\Omega_i)}$ holds for any $\xi \in V$.
4. $S_2^{-1}S_1$ with its domain $D(S_1)$ admits of a bounded extension H into V. In fact, H is given by
$$H = S_2^{-1/2}(S_1^{1/2}S_2^{-1/2})^* S_1^{1/2},$$
where * means the adjoint in X.

Therefore, the precise meaning of (3) is understood as:

$$\xi^{(k+1)} = A_\theta \xi^{(k)}, (k = 0, 1, 2, \cdots), \qquad \xi^{(0)} \in V,$$

where A_θ is the amplification operator for the error defined by

$$A_\theta = (1-\theta)I - \theta H, \quad (I \text{ is the identity}).$$

To treat A_θ as a self-adjoint operator, we employ the following device. Thus, we introduce a special inner product in V in terms of the SP operator:

$$((\xi, \eta)) = (S_2^{1/2}\xi, S_2^{1/2}\eta)_X, \qquad (\forall \xi, \eta \in V). \tag{4}$$

Then V again forms a Hilbert space with the new inner product (4). Moreover, by virtue of (2), we deduce that the corresponding norm $|||\cdot||| = ((\cdot,\cdot))^{1/2}$ is equivalent to $\|\cdot\|_V$ in V. Furthermore, under (4), H and therefore A_θ are self-adjoint in V.

Concerning the spectral radius $r_\sigma(A_\theta)$ of A_θ, as a direct consequence of the spectral mapping theorem (See, for example, [Yos80]), we have

$$r_\sigma(A_\theta) = \sup_{s \in \sigma(H)} |1 - \theta - \theta s|,$$

where $\sigma(H)$ denotes the spectrum of H.

Shape Conditions and Convergence Results

Throughout this section, we assume that γ is a line segment on the x_2-axis. In order to evaluate $r_\sigma(A_\theta)$, we introduce shape conditions of subdomains under the notation:

- R denotes reflection with respect to the x_2-axis defined by
$$R: \ (x_1, x_2) \mapsto (-x_1, x_2).$$
- T_m, m being a positive constant, denotes the contraction mapping along the x_1-axis defined by
$$T_m: \ (x_1, x_2) \mapsto \Big(\frac{x_1}{m}, x_2\Big).$$

Conditions (I_m) and (I^l). Let $1 \leq m < \infty$. We say that Condition (I_m) is satisfied if

$$RT_m\Omega_2 \subseteq \Omega_1.$$

On the other hand, for $1 \leq l \leq \infty$, we say that Condition (I^l) is satisfied if

$$RT_l\Omega_1 \subseteq \Omega_2.$$

In the above definition, we understand that Condition (I^l) is not satisfied if $l = \infty$. The following lemma is a consequence of Conditions (I_m) and (I^l).

Lemma 1 *Let $1 \leq m < \infty$ and $1 \leq l \leq \infty$, and suppose that both Conditions (I_m) and (I^l) are satisfied. Then we have*

$$\frac{1}{l} \leq H \leq m. \tag{5}$$

That is, $1/l \leq s \leq m$ holds for any $s \in \sigma(H)$.

Therefore, concerning a convergence of the DN method, we obtain the following theorems:

Theorem 1 *Let $1 \leq m < \infty$ and $1 \leq l \leq \infty$, and suppose that both Conditions (I_m) and (I^l) are satisfied. For $0 < \theta < 1$, we define*

$$\tilde{r}(\theta) = \begin{cases} 1 - (1 + \frac{1}{l})\theta & \text{for } 0 < \theta \leq \frac{2}{m+l^{-1}+2}, \\ (m+1)\theta - 1 & \text{for } \frac{2}{m+l^{-1}+2} \leq \theta < 1. \end{cases}$$

Furthermore, assume that $0 < \theta < \dfrac{2}{m + l^{-1} + 1}$. Then $0 < \tilde{r} < 1$ and there exists a positive constant c_0 depending only on Ω_2 such that

$$\|\xi^{(k)}\|_V \leq c_0 \tilde{r}(\theta)^k \|\xi^{(0)}\|_V, \qquad (k = 1, 2, 3, \cdots).$$

Theorem 2 *Under the same assumptions of Theorem 1, we have*

$$\begin{aligned} \|u_1^{(k)} - \tilde{u}|_{\Omega_1}\|_{H^1(\Omega_1)} &\leq c_1 \tilde{r}(\theta)^k \|u_1^{(0)} - \tilde{u}|_{\Omega_1}\|_{H^1(\Omega_1)}, \\ \|u_2^{(k)} - \tilde{u}|_{\Omega_2}\|_{H^1(\Omega_2)} &\leq c_2 \tilde{r}(\theta)^k \|u_2^{(0)} - \tilde{u}|_{\Omega_2}\|_{H^1(\Omega_2)}, \end{aligned}$$

where c_1 and c_2 are domain constants.

Theorem 3 *Under the same assumptions of Theorem 1, by choosing $\theta = \dfrac{2}{m + l^{-1} + 2}$, we get $\tilde{r}_{opt} = \dfrac{m + l^{-1}}{m + l^{-1} + 2}$ as the optimal value of $\tilde{r}$.*

Optimality of (5)

The estimate (5) in Lemma 1 is really optimal in a certain sense. We below explain this fact with the aid of a simple example. We consider the case where Ω is a rectangle and γ is a line segment parallel to the lateral sides of the rectangle. Specifically, we assume that, for $0 < a_1 \le a_2$ and $b > 0$,

$$\begin{cases} \Omega_1 &= \{(x_1, x_2);\ -a_1 < x_1 < 0,\ 0 < x_2 < b\}, \\ \Omega_2 &= \{(x_1, x_2);\ 0 < x_1 < a_2,\ 0 < x_2 < b\}, \\ \gamma &= \{(0, x_2);\ 0 < x_2 < b\}. \end{cases}$$

Let $\xi \in V$ and write

$$\xi = \sum_{n=1}^{\infty} c_n \phi_n, \quad c_n = c_n(\xi) = (\xi, \phi_n)_X,$$

where ϕ_n are the eigenfunctions of (7) which will appear in Appendix. In this case, the harmonic extensions $w_1 = H_1\xi$ and $w_2 = H_2\xi$ can be expressed explicitly. In particular, we have

$$\frac{\partial w_1}{\partial x_1}\Big|_{x_1=0} = \sum_{n=1}^{\infty} \sqrt{\lambda_n} c_n \coth(\sqrt{\lambda_n} a_1)\phi_n(x_2),$$

where $\sqrt{\lambda_n} = n\pi/b$ and $\coth s = (e^s + e^{-s})(e^s - e^{-s})^{-1}$. Hence

$$S_1\phi_n = \sqrt{\lambda_n} \coth(\sqrt{\lambda_n} a_1)\phi_n,$$

since $c_j = (\phi_n, \phi_j)_X = \delta_{n,j}$ (Kronecker's delta). This means that $\zeta_n^{(1)} = \sqrt{\lambda_n} \coth(\sqrt{\lambda_n} a_1)$ are the eigenvalues of S_1. In the similar way, $\zeta_n^{(2)} = \sqrt{\lambda_n} \coth(\sqrt{\lambda_n} a_2)$ are the eigenvalues of S_2. ϕ_n is the eigenfunction of S_1 and S_2 corresponding to $\zeta_n^{(1)}$ and $\zeta_n^{(2)}$, respectively. The operator H is a compact operator in X, since, from Lemma 2 in Appendix, the imbedding operator from V into X is compact. Therefore, the spectrum of H consists of only the set of the eigenvalues $\{\zeta_n^{(1)}/\zeta_n^{(2)}\}_{n=1}^{\infty}$. As a result, by Rayleigh's principle,

$$\sup_{\xi \in V} \frac{((H\xi, \xi))}{|||\xi|||^2} = \frac{\zeta_1^{(1)}}{\zeta_1^{(2)}} = \frac{\tanh(\pi a_2/b)}{\tanh(\pi a_1/b)} \equiv \tau(a_1, a_2, b).$$

In addition, we have

$$\inf_{\xi \in V} \frac{((H\xi, \xi))}{|||\xi|||^2} = \inf_{n \ge 1} \frac{\zeta_n^{(1)}}{\zeta_n^{(2)}} = 1,$$

since $\zeta_n^{(1)}/\zeta_n^{(2)}$ is a non-increasing sequence in n and is greater than 1. Therefore we obtain

$$1 \le H \le \tau(a_1, a_2, b).$$

On the other hand, both Conditions (I_m) and (I^l) are satisfied with $l = 1$ and $m = a_2/a_1$;

$$1 \le H \le \frac{a_2}{a_1}. \tag{6}$$

We now note that

$$1 < \tau(a_1, a_2, b) < \frac{a_2}{a_1},\ (b > 0) \quad \text{and} \quad \tau(a_1, a_2, b) \to \frac{a_2}{a_1},\ (b \to \infty).$$

This means that the estimate (6) by Conditions (I_m) and (I^l) is really optimal when b is sufficiently large for fixed a_1 and a_2.

Remarks

1. Numerical results to exemplify our theoretical results are presented in [Fuj97], [FKKN96], [FFS98] and [FS97].
2. Our method of analysis works for some other domain decomposition algorithms, for instance, the Neumann-Neumann method proposed in [BGLTV89].
3. The similar problem for the Stokes equations is discussed in [Sai00]. There a new important role of the inf-sup constant is revealed.

Appendix. An Equivalent Norm to $\|\cdot\|_V$

We assume that γ is a line segment on the x_2-axis. Let $\{\lambda_n\}_{n=1}^{\infty}$ be the set of the eigenvalues of the eigenvalue problem:

$$-\frac{d^2}{dx_2^2}\phi = \lambda\phi \text{ in } \gamma, \quad \phi = 0 \text{ on } \partial\gamma \tag{7}$$

and, let $\phi_n = \phi_n(x_2)$ be the eigenfunction corresponding to λ_n which is normalized as $\|\phi_n\|_X = 1$. Then each element $\xi \in X$ can be expanded by the Fourier series as follows:

$$\xi = \sum_{n=1}^{\infty} c_n\phi_n \quad \text{with } c_n = (\xi, \phi_n)_X.$$

We introduce

$$U^{1/4} = \Big\{\xi = \sum_{n=1}^{\infty} c_n\phi_n \in X;\ \sum_{n=1}^{\infty} c_n^2\lambda_n^{1/2} < \infty\Big\}$$

with the norm

$$\|\xi\|_{U^{1/4}} = \Big(\sum_{n=1}^{\infty} c_n^2 + \sum_{n=1}^{\infty} \lambda_n^{1/2}c_n^2\Big)^{1/2} \quad \text{for } \xi = \sum_{n=1}^{\infty} c_n\phi_n \in U^{1/4}. \tag{8}$$

The space $U^{1/4}$ forms a Hilbert space equipped with the norm $\|\cdot\|_{U^{1/4}}$. Moreover, $U^{1/4}$ coincides with the domain $D(L^{1/4})$ of the fractional power of L, where L means a minus Laplacian on γ with the zero boundary condition. Then we have, by virtue of [Fuj67], $V = U^{1/4}$ with the equivalent norm (8). This implies, in view of the closed graph theorem, that

$$C\|\xi\|_V \le \|\xi\|_{U^{1/4}} \le C'\|\xi\|_V, \quad (\forall\xi \in V).$$

Namely, we can employ $\|\cdot\|_{U^{1/4}}$ as the norm of V. This sometimes gives a better viewpoint of our discussion. For instance, the following proposition is an easy consequence of (8).

Lemma 2 *$U^{1/4} = D(L^{1/4})$ is compactly imbedded in X, if γ is a line segment.*

Proof We set

$$i_N\xi = \sum_{n=1}^{N} c_n\phi_n \quad \text{for } \xi = \sum_{n=1}^{\infty} c_n\phi_n \in X.$$

The operator i_N is a degenerate operator from $U^{1/4}$ into X. Let i be the imbedding operator from $U^{1/4}$ into X. Then we can calculate as

$$\|(i - i_N)\xi\|_X^2 = \sum_{n=N+1}^{\infty} c_n^2 \le \lambda_{N+1}^{-1/2} \sum_{n=1}^{\infty} c_n^2 \lambda_n^{1/2} \le \lambda_{N+1}^{-1/2} \|\xi\|_{U^{1/4}}^2.$$

Thus we have $\|i - i_N\|_{U^{1/4},X} \le \lambda_{N+1}^{-1/4} \to 0$ as $N \to \infty$. Since degenerate operators i_N are compact, i is also compact. ■

References

[BGLTV89]Jean-François Bourgat, Roland Glowinski, Patrick Le Tallec, and Marina Vidrascu. Variational formulation and algorithm for trace operator in domain decomposition calculations. In Tony Chan, Roland Glowinski, Jacques Périaux, and Olof Widlund, editors, *Domain Decomposition Methods*, pages 3–16, Philadelphia, PA, 1989. SIAM.

[FFS98]Hiroshi Fujita, Makoto Fukuhara, and Norikazu Saito. On the rate of convergence of iterations in the domain decomposition methods. In Zhong-Ci Shi and Masatake Mori, editors, *Proceedings of the third China-Japan seminar on numerical mathematics*, pages 30–43, Beijing and New-York, 1998. Science Press.

[FKKN96]Hiroshi Fujita, Masashi Katsurada, Asako Kobari, and Yoshiaki Nagasaka. Analytical and numerical study of convergence of the domain decomposition method, I. *Mem. Inst. Sci. Tech. Meiji Univ.*, 35(8):103–135, 1996. in Japanese.

[FS97]Hiroshi Fujita and Norikazu Saito. An analytical study of optimal speed of convergence of iterations in ddm under certain shape assumptions of domains. In *Computational Science for the 21st Century*, pages 139–148, New-York, 1997. John Wiley & Sons.

[Fuj67]Daisuke Fujiwara. Concrete characterization of the domains of fractional powers of some elliptic differential operators of the second order. *Proc. Japan Acad.*, 43:82–86, 1967.

[Fuj97]Hiroshi Fujita. Remarks on the domain-dependence of convergence rate of iterations in a certain domain decomposition method. In *Collection of Papers on Geometry, Analysis and Mathematical Physics*, pages 71–84, River Edge, NJ, 1997. World Science Publishing.

[Gri85]P. Grisvard. *Elliptic problems in nonsmooth domains.* Pitman Publishing, Boston, 1985.

[Kat76]Tosio Kato. *Perturbation Theory for Linear Operators*. Springer-Verlag, second edition, 1976.

[LM72]Jacques-Louis Lions and Enrico Magenes. *Nonhomogeneous Boundary Value Problems and Applications*, volume I. Springer, New York, Heidelberg, Berlin, 1972.

[Sai00]N. Saito. On the domain-dependence of convergence rate in a domain decomposition method for the Stokes equations. *East-West J. of Numer. Math.*, 8(1):37–55, 2000.

[Yos80]Kôsaku Yosida. *Functional Analysis*. Springer-Verlag, sixth edition, 1980.

Part II

Algorithms

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau,

7. The mortar element method with overlapping subdomains

Yves Achdou[1] , Yvon Maday [2]

Introduction

The mortar element methods were introduced in [BMP94] for non overlapping domain decompositions in order to couple different variational approximations in different subdomains. In the finite element context, one important advantage of the mortar element methods is that it allows for using structured grids in subdomains thus fast solvers [AAH$^+$98]. The resulting methods are nonconforming but still yield optimal approximations. The literature on the mortar element methods is growing numerous see [AMW99] and reference therein.

In this paper, we shall discuss the case of overlapping subdomains, with meshes constructed in an independent manner in each subdomain. As pointed by F. Hecht, J.L. Lions, and O. Pironneau, [LP99, HLP99] such a situation can occur if the domain of computation is a scene constructed by Constructive Solid Geometry as usual in Image Synthesis and Virtual Reality : each object of the scene is described by set operations on primitive shapes like cubes, cylinders, spheres and cones. With VRML (the language of VR), the objects may be described as unions of more elementary objects with primitive shapes, which are never intersected, so it is not possible to construct a global mesh. Each simple object must have its individual mesh. In [LP99, HLP99], many algorithms (including algorithms from control theory) for this situation are proposed, and cover more general cases than overlapping subdomains (domain with holes for example).

We also note that independent of the development of the mortar methods, overlapping domain decomposition with non matching grids has been used for finite difference discretizations in the engineering community : these methods are often referred to as the *chimera methods* see [CH90, SB87],

To our knowledge, mortar methods with overlapping subdomains have been proposed first by Y. Kuznetsov [Kuz97] who focused on iterative solvers with Lagrange multipliers. For two overlapping subdomains, the mortar method has been analyzed by X.C. Cai and M. Dryja and M. Sarkis [CDS99] in two dimensions. They have considered two subdomains, with non matching grids and piecewise linear Lagrange finite elements. In particular, they have considered the case when the overlapping parameter is 0, (two rectangular subdomains for a L shaped domain). They have also proposed iterative solvers and preconditioners for the linear systems arising from the mortar discretization.

In this paper, we generalize their method in two dimensions, with more than two subdomains. We shall see that technical difficulties arise when the boundary of two

[1] laboratoire ASCI, Université Paris Sud and INSA Rennes, 20 Av des Buttes de Coesmes, 35043 Rennes, France

[2] laboratoire ASCI, Université Paris Sud and Laboratoire d'Analyse numérique, Université Paris 6

subdomains cross each other. For simplicity, we consider the Laplace equation and we rule out the case when the overlap may vanish. For such situations, one should mix the method described in [CDS99] and the one below.

This paper contains the results of a more detailed work, see [YM00] where the proofs of the results below are given, and where iterative preconditioned solvers are discussed too.

Description of the method and numerical analysis

First definitions

In all what follows, c or C will stand for various constants, independent from the geometric parameters.

We consider a polygonal domain Ω of $\mathbb{R}^2$ and the model boundary value problem in Ω

$$\begin{array}{rcll} -\Delta u & = & f & \text{in } \Omega, \\ u & = & 0 & \text{on } \partial\Omega. \end{array} \tag{1}$$

We consider first a family of overlapping subdomains $(\Omega_k)_{k\in\{1,\dots,K\}}$ with polygonal shapes covering Ω:

$$\Omega = \overset{K}{\underset{k=1}{\cup}} \Omega_k. \tag{2}$$

We denote by H_k the diameter of Ω_k and H the maximal diameter $H = \max_{1\le k\le K} H_k$. We assume that there exists a constant c such that for any k, $1 \le k \le K$, $cH \le H_k \le H$. We also suppose that there exists a constant τ such that any subdomain Ω_k contains a ball of diameter greater than τH.

For any subdomain Ω_k, we denote by δ_k the minimum distance of overlap between Ω_k and $\cup_{i\neq k}\Omega_i$:

$$\delta_k = \inf_{x\in\Omega_k\setminus\cup_{i\neq k}\Omega_i} \; \inf_{y\in\cup_{i\neq k}\Omega_i\setminus\Omega_k} |x-y|.$$

We also define $\delta \equiv \min_k \delta_k$.

Assumption 1 *We assume that the intersection of two subdomains' boundaries can only be isolated points, called crosspoints. We assume that there exists a constant α, $0 < \alpha \le \frac{\pi}{2}$ such that the angles (taken not greater than $\frac{\pi}{2}$) between two subdomains boundaries crossing each other are all greater than α. For simplicity, we assume also that a given crosspoint is neither the intersection of more than two subdomains' boundaries, nor the vertex of a subdomain.*

Assumption 2 *We assume that there exists a constant number N_1 such that, for any ball B of diameter H, $B \cap \Omega$ is covered by at most N_1 subdomains.*

This assumption yields two important consequences:

Property 1 *We denote by ω_k the union of the subdomains intersecting Ω_k, and by $\mathcal{I}_k$ the set of the integers i such that $\Omega_i \subset \omega_k$. There exists a constant $n_1(N_1)$ such that, for any k, $1 \le k \le K$, cardinal$(\mathcal{I}_k) \le n_1(N_1)$.*

Figure 1: The spaces Z_k^l and $\widetilde{W_k^l}$: nodal bases

Property 2 *There exists a constant $n_2(N_1)$ such that, the number of subdomains containing a given point in Ω is bounded by $n_2(N_1)$.*

We also make the assumption

Assumption 3 *The number of crosspoints lying on $\partial\Omega_k$ is bounded by a constant N_2.*

On each subdomain Ω_k, we have a family of triangular meshes $\mathcal{T}_{k,h_k}$ whose triangles have maximal diameters h_k. The meshes are constructed in an independent manner. The mesh points on $\partial\Omega_k$ need not match with the mesh points in the overlapping subdomains. We assume that the families $(\mathcal{T}_{k,h_k})_{h_k}$ are shape regular and quasi uniform, see [Cia78]. We agree to simplify the notations by replacing $\mathcal{T}_{k,h_k}$ by $\mathcal{T}_k$.

Assumption 4 *We call $h = \max_k h_k$ and we assume that, for a given positive constant C,*

$$Ch < \delta. \tag{3}$$

Associated with the mesh $\mathcal{T}_k$, we consider the spaces Z_k and X_k of piecewise linear Lagrange finite elements

$$Z_k = \left\{ \begin{array}{l} u_k \text{ is continuous in } \overline{\Omega}_k, \\ \forall t \in \mathcal{T}_k, u_k|_t \text{ is linear} \end{array} \right\}, \quad X_k \equiv \{u_k \in Z_k, u_k = 0 \text{ on } \partial\Omega \cap \partial\Omega_k\}.$$

Each space X_k and Z_k is supplied with its usual nodal basis functions. We define $X = \Pi_{k=0}^K X_k$. The vectors $u = (u_k)_{k\in\{1\ldots K\}}$ of X are collections of functions defined in the subdomains, but no continuity constraints are imposed at the subdomains boundaries. The nodal basis of X can be found by taking the product of the nodal bases of the spaces X_k.

We denote by $(\Gamma_k^l)_{l\in\{1\ldots E_k\}}$ the edges of $\partial\Omega_k$. For an edge Γ_k^l of $\partial\Omega_k$, we denote by Z_k^l the space of functions obtained by taking the trace on Γ_k^l of the functions of Z_k, and by $\mathcal{T}_k^l$ the trace of the mesh $\mathcal{T}_k$ on Γ_k^l. The space Z_k^l is the space of piecewise linear Lagrange finite elements on $\mathcal{T}_k^l$.

The matching condition

In order to discretize (1), we need to define a subspace Y of X by imposing weak continuity constraints at the subdomains boundaries $\partial\Omega_k$, $1 \le k \le K$.

For an edge Γ_k^l of $\partial\Omega_k \backslash \partial\Omega$ we denote by $\widetilde{W_k^l}$ the subspace of Z_k^l of the functions whose restrictions to the extreme elements of $\mathcal{T}_k^l$ are constant. Such spaces are used as mortar spaces for the non overlapping case (see [BMP94]). Here, we will have to additionally modify them locally, near the crosspoints.

We consider an edge Γ_k^l of $\partial\Omega_k\backslash\partial\Omega$. Let $(j_i)_{i\in\{1\cdots n_k^l\}}$ be the family of the indices such that $|\Gamma_k^l \cap \Omega_{j_i}| > 0$ and $j_i \neq k$. Note that, from assumption 3 , n_k^l is bounded by a constant C. For $i \in \{1\cdots n_k^l\}$, we define $\Gamma_k^{l,i} = \Gamma_k^l \cap \Omega_{j_i}$. From (2),

$$\Gamma_k^l = \overset{n_k^l}{\underset{i=1}{\cup}} \Gamma_k^{l,i}.$$

Call $p_k^l(x)$ the piecewise constant function defined on Γ_k^l by

$$p_k^l(x) = \sum_{i=1}^{n_k^l} 1_{\Gamma_k^{l,i}}(x). \tag{4}$$

From (2) and property 2, there exists a constant C such that $1 \le p_k^l \le C$.

Given W_k^l a space of test functions defined on Γ_k^l, the first possible matching condition on Γ_k^l will be of the form

$$\forall w \in W_k^l, \quad \int_{\Gamma_k^l} \frac{1}{p_k^l(x)+1}\left(u_k(x) - \frac{1}{p_k^l(x)}\sum_{i=1}^{n_k^l} 1_{\Gamma_k^{l,i}}(x) u_{j_i}(x)\right) w(x)dx = 0. \tag{5}$$

Basically, the space W_k^l will be a subspace of $\widetilde{W_k^l}$, and the spaces will differ essentially due to the presence of crosspoints.

There remains now to define the space W_k^l. Suppose that for $i \in \{1,\dots,n_k^l\}$, $\Gamma_k^l \cap \partial\Omega_{j_i} \neq \emptyset$. Then, from assumption 1, we know that the intersections do not take place at a vertex of $\partial\Omega_{j_i}$ and let $\Gamma_{j_i}^{l'}$ be the edge of $\partial\Omega_{j_i}$ such that $\Gamma_k^l \cap \Gamma_{j_i}^{l'}$ is a point denoted by x . If no special attention is taken for the choices of W_k^l and $W_{j_i}^{l'}$, then the matching condition on Γ_k^l and $\Gamma_{j_i}^{l'}$ will strongly couple the degrees of freedom of u_k and u_{j_i} near the crosspoint x, and there might be cases when these conditions are too restrictive *i.e.* the functions u_k and u_{j_i} must be constant even zero near x. To avoid such a situation, and also in order to get a solver with good parallel properties, we have to relax the weak continuity condition near x.

We call $(x_m)_{m\in\{1,\dots M_k^l\}}$ the nodes of $\mathcal{T}_k^l$ different from the endpoints of Γ_k^l, and $(\phi_m)_{m\in\{1,\dots M_k^l\}}$ (resp. $(\psi_m)_{m\in\{0,\dots M_k^l+1\}}$) the nodal basis functions of $\widetilde{W_k^l}$ (resp. of Z_k^l). Note that $\phi_m = \psi_m$ for $2 \le m \le M_k^l - 1$.

We select the nodes for which the support of the corresponding basis function of X_k does not intersect $\Gamma_{j_i}^{l'}$: we obtain the set of nodes $(x_m)_{m\in\{1,\dots m_1\}\cup\{m_2,\dots M_k^l\}}$. We call $\widetilde{\phi}_{m_1}$ the continuous function vanishing outside (x_{m_1-1}, x_{m_2}), linear on (x_{m_1-1}, x_{m_1}) and on (x_{m_1}, x_{m_2}), and such that $\widetilde{\phi}_{m_1}(x_{m_1}) = 1$. Likewise, $\widetilde{\phi}_{m_2}$ is the continuous function vanishing outside (x_{m_1}, x_{m_2+1}), linear on (x_{m_1}, x_{m_2}) and on (x_{m_2}, x_{m_2+1}), and such that $\widetilde{\phi}_{m_2}(x_{m_2}) = 1$. The space $W_k^{l,x}$ is defined by

$$W_k^{l,x} \equiv span(\phi_1,\dots,\phi_{m_1-1},\widetilde{\phi}_{m_1},\widetilde{\phi}_{m_2},\phi_{m_2+1},\dots\phi_{M_k^l}). \tag{6}$$

The space $W_k^{l,x}$ is displayed on Figure 2. For what follows, we also define the space

$$\begin{array}{rcl} X_k^{l,x} & \equiv & \{u \in X_k^l, u = 0 \text{ at the endpoints of } \Gamma_k^l \text{ and } x_{m_1+1},\dots,x_{m_2-1}\} \\ & = & span(\psi_1,\dots,\psi_{m_1},\psi_{m_2},\dots\psi_{M_k^l}). \end{array} \tag{7}$$

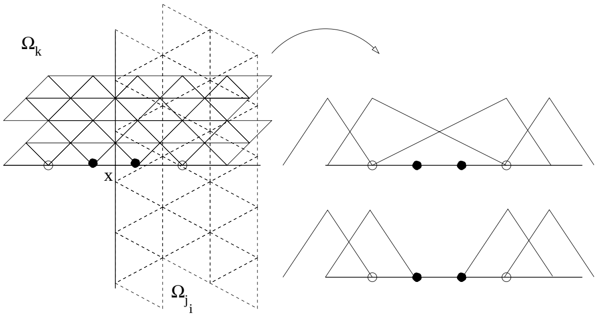


Figure 2: The spaces $W_k^{l,x}$ and $X_k^{l,x}$ (only two subdomains have been represented)

Definition 1 *For the crosspoint x, we define the zone of influence of x on Γ_k^l as the interval (x_{m_1-1}, x_{m_2+1}). We also define the zone of influence of a vertex x of Ω_k on Γ_k^l as the union of the two elements of $\mathcal{T}_k^l$ next to x. From Assumption 1, the zone of influence of a crosspoint has a size smaller than Ch.*

Assumption 5 *The zones of influence of two crosspoints on Γ_k^l are disjoint. Moreover, the zones of influence on Γ_k^l of a crosspoint and a vertex of Ω_k are disjoint.*

Finally, we define $\mathcal{X}_k^l$ the set of crosspoints on Γ_k^l and we set

$$W_k^l \equiv \bigcap_{x \in \mathcal{X}_k^l} W_k^{l,x} \tag{8}$$

and, likewise

$$X_k^l \equiv \bigcap_{x \in \mathcal{X}_k^l} X_k^{l,x}, \tag{9}$$

and Y is the subspace of X defined by

$$Y \equiv \{u \in X; \forall k \in \{1 \dots K\}, \forall l \in \{1 \dots E_k\}, u \text{ satisfies (5) } \}. \tag{10}$$

for W_k^l defined by (8) and (6).

Remark 1 *The functions in W_k^l will resemble those of $\widetilde{W_k^l}$ except at a few nodes near crosspoints. Furthermore, from assumption 5, these exceptional regions around crosspoints are disjoint.*

Remark 2 *The spaces W_k^l and X_k^l have the same dimension.*

Let $\mathcal{V}_k$ be the set of the nodes containing

1. the vertices of $\partial\Omega_k$.
2. all the other nodes of $\mathcal{T}_k$ on $\partial\Omega_k$ such that the support of the corresponding nodal basis function of X_k intersects another subdomain's boundary.

Lemma 1 *For a given crosspoint x on Γ_k^l, let $(x_m)_{m\in\{1,\dots m_1\}\cup\{m_2,\dots M_k^l\}}$ be the nodes of $\mathcal{T}_k^l$ involved in the above construction of $W_k^{l,x}$. Let δ^{--}, δ^-, δ^+ and δ^{++} be defined by $\delta^{--}=\frac{x_{m_1}-x_{m_1-1}}{x_{m_2}-x_{m_1}}$, $\delta^-=\frac{x_{m_1+1}-x_{m_1}}{x_{m_2}-x_{m_1}}<1$, $\delta^+=\frac{x_{m_2}-x_{m_2-1}}{x_{m_2}-x_{m_1}}<1$ and $\delta^{++}=\frac{x_{m_2+1}-x_{m_2}}{x_{m_2}-x_{m_1}}$. Assume that there exists a constant c such that for all crosspoint x,*

$$\begin{array}{l}\frac{3}{2}\delta^- + \delta^{--} - (\delta^+)^2 \geq c,\\ \frac{3}{2}\delta^+ + \delta^{++} - (\delta^-)^2 \geq c,\end{array} \tag{11}$$

then there exists a constant C independent of h such that

$$\inf_{u\in W_k^l}\ \sup_{0\neq w\in X_k^l} \frac{\int_{\Gamma_k^l}\frac{1}{p_k^l(x)+1}u(x)w(x)dx}{\|w\|_{L^2(\Gamma_k^l)}} \geq C\|u\|_{L^2(\Gamma_k^l)}. \tag{12}$$

Let u be a function in $L^2(\Gamma_k^l)$. As a consequence of lemma 1, and if (11) is satisfied, the problem : find $u_k^l \in Z_k^l$ such that :

$$\begin{array}{l} u_k^l \text{ is given at the nodes of } \Gamma_k^l \cap \mathcal{V}_k,\\ \forall w_k^l \in W_k^l, \quad \displaystyle\int_{\Gamma_k^l}\frac{1}{p_k^l(x)+1}u_k^l(x)w_k^l(x)dx = \int_{\Gamma_k^l}\frac{1}{p_k^l(x)+1}u(x)w_k^l(x)dx \end{array} \tag{13}$$

is well posed. Furthermore, if we impose that $u_k^l = 0$ at the nodes in $\Gamma_k^l\cap\mathcal{V}_k$, then we have

$$\|u_k^l\|_{L^2(\Gamma_k^l)} \leq C\|u\|_{L^2(\Gamma_k^l)}. \tag{14}$$

Likewise, let x_i be a given node in $\Gamma_k^l\cap\mathcal{V}_k$. Under the same technical assumptions, the solution of the problem: find $\widetilde{\psi}_i \in Z_k^l$ such that

$$\begin{array}{l} \widetilde{\psi}_i(x_i) = 1,\\ \widetilde{\psi}_i = 0 \text{ at the other nodes of } \Gamma_k^l\cap\mathcal{V}_k,\\ \forall w_k^l \in W_k^l, \quad \displaystyle\int_{\Gamma_k^l}\frac{1}{p_k^l(x)+1}\widetilde{\psi}_i(x)w_k^l(x)dx = 0, \end{array} \tag{15}$$

satisfies

$$\|\widetilde{\psi}_i\|_{L^2(\Gamma_k^l)} \leq Ch^{\frac{1}{2}}. \tag{16}$$

The discrete problem

From now on, we shall assume that the conditions (11) are satisfied.

Let $\sigma(x) = \sum_{k=1}^K 1_{\Omega_k}(x)$. From Property 2, σ is bounded from above by a constant, and $\sigma \geq 1$. Consider the discrete problem : find $u\in Y$ such that for all $v \in Y$,

$$\sum_{k=1}^K \int_{\Omega_k}\frac{1}{\sigma}\nabla u_k\cdot\nabla v_k = \sum_{k=1}^K\int_{\Omega_k}\frac{1}{\sigma}fv_k. \tag{17}$$

Call a the symmetric bilinear form on Y :

$$a(u,v) = \sum_{k=1}^{K} \int_{\Omega_k} \frac{1}{\sigma} \nabla u_k \cdot \nabla v_k. \tag{18}$$

Now, we wish to obtain an estimate on the ellipticity constant, under typical but not necessarily optimal assumptions.

Assumption 6 *Let Ω_k be a subdomain. We assume that for a positive constant C, for each $i \neq k \in \mathcal{I}_k$, there exists an edge Γ_i^e and a sub-interval γ_i of Γ_i^e such that*

- $\gamma_i \subset \Omega_k$.
- $|\gamma_i| > CH$.
- γ_i *is the union of elements of* $\mathcal{T}_i^e$.

Lemma 2 *Under the assumptions 1 to 6 and (11), there exists a constant C_e independent on the mesh parameters such that*

$$\forall u \in Y, \quad a(u,u) \geq C_e \sum_{k=1}^{K} \int_{\Omega_k} \left(|\nabla u_k|^2 + u_k^2 \right). \tag{19}$$

If only assumptions 1 to 5 and (11) are satisfied, we have (19), but we only know that there exists a constant C independent on the mesh parameters such that

$$C_e \leq C \frac{1}{\max_l (1 + \log \frac{H}{h_l})}. \tag{20}$$

Error analysis

By the Berger-Scott-Strang lemma, see [BSS72, SF73], we know that the error of the method is the sum of a consistency error plus a best approximation error: calling u^* be the weak solution of (1),

$$\|u - u^*\|_* \leq \frac{1}{C_e} \left(\inf_{v \in Y} |u^* - v|_* + \sup_{0 \neq v \in Y} \frac{|a(u^*, v) - \sum_{k=1}^{K} \int_{\Omega_k} \frac{1}{\sigma} f v_k|}{|v|_*} \right). \tag{21}$$

where C_e is the ellipticity constant. The first term in the right hand side of (21) is a best approximation error while the second one is a consistency error due to non conformity.

Lemma 3 Consistency error. *Let u^* be the weak solution of (1). Assume that $u^*|_{\Omega_k}$ belongs to $H^{\sigma_k}(\Omega_k)$, with $\sigma_k > \frac{3}{2}$. Then the consistency error is bounded by*

$$C(1 + \max_k \log \frac{H}{h_k}) \left(\sum_{k=1}^{K} \max_{i \in \mathcal{I}_k} \left(1 + \sqrt{\frac{h_i}{h_k}} \right)^2 h_k^{2(\sigma_k - 1)} |u^*|^2_{H^{\sigma_k}(\Omega_k)} \right)^{\frac{1}{2}}.$$

Lemma 4 Best approximation error. *Let $v^* \in H^1(\Omega)$ be such that for $1 \le k \le K$, $v^*|_{\Omega_k} \in H^{\sigma_k}(\Omega_k)$ with $2 \ge \sigma_k > 1$. Then there exists $v \in Y$ such that*

$$\sum_{k=1}^{K} \frac{1}{h_k} \|v_k^* - v_k\|_{L^2(\Omega_k)} + |v_k^* - v_k|_{H^1(\Omega_k)} \le C \sum_{k=1}^{K} h_k^{\sigma_k - 1} |v_k^*|_{H^{\sigma_k}(\Omega_k)}. \tag{22}$$

Then the error estimate is given by the following theorem :

Theorem 1 *Assume that the solution u^* of (1) is such that for $1 \le k \le K$, $u^*|_{\Omega_k} \in H^{\sigma_k}(\Omega_k)$ with $2 \ge \sigma_k > \frac{3}{2}$. Then there exists a constant C such that, if $u \in Y$ is the solution of (17)*

$$\begin{aligned} &\textstyle\sum_{k=1}^{K} \|u_k^* - u_k\|_{H^1(\Omega_k)} \\ &\le \frac{C}{C_e}(1 + \max_k \log \frac{H}{h_k}) \left(\sum_{k=1}^{K} \max_{i \in \mathcal{I}_k} \left(1 + \sqrt{\frac{h_i}{h_k}}\right)^2 h_k^{2(\sigma_k - 1)} |u^*|^2_{H^{\sigma_k}(\Omega_k)} \right)^{\frac{1}{2}}, \end{aligned} \tag{23}$$

where C_e is the ellipticity constant.

Remark 3 *It seems possible but not easy to improve the consistency error estimate and get rid of some logarithmic factors. It will be the topic of a future research.*

A strengthened matching condition

We give below an example of stronger matching conditions in the neighborhood of crosspoints.

With the notations introduced in § 7, it is possible to strengthen the previous matching condition by supplementing the previous test function space $W_k^{l,0} \equiv W_k^l$ with Q supplementary spaces $(W_k^{l,q})_{1 \le q \le Q}$ (to be defined below) such that $dim(W_k^l) + \sum_{q=1}^{Q} dim(W_k^{l,q}) \le dim(\widetilde{W_k^l})$. Typically, each new space will correspond to a crosspoint on Γ_k^l. We define the direct sum : $\overline{W_k^l} = \bigoplus_{q=0}^{Q} W_k^{l,q}$, and we introduce a family of coefficients $\lambda_{0i} = 1$ for $1 \le i \le n_k^l$ and $\lambda_{qi} \in \{0,1\}$ for $1 \le q \le Q$ and $1 \le i \le n_k^l$ (these coefficients will be defined below) and we call p_q the function defined on Γ_k^l by

$$p_q(x) = \sum_{i=1}^{n_k^l} \lambda_{qi} 1_{\Gamma_k^{l,i}}(x). \tag{24}$$

Then the strengthened matching condition reads

$$\forall w \in W_k^{l,0}, \quad \int_{\Gamma_k^l} \frac{1}{p_k^l(x) + 1} \left(u_k(x) - \frac{1}{p_0(x)} \sum_{i=1}^{n_k^l} 1_{\Gamma_k^{l,i}}(x) u_{j_i}(x) \right) w(x) dx = 0, \tag{25}$$

$$\forall q \in \{1, \dots, Q\}, \; \forall w \in W_k^{l,q}, \quad \int_{\Gamma_k^l} \left(u_k(x) - \frac{1}{p_q(x)} \sum_{i=1}^{n_k^l} \lambda_{qi} 1_{\Gamma_k^{l,i}}(x) u_{j_i}(x) \right) w(x) dx = 0. \tag{26}$$

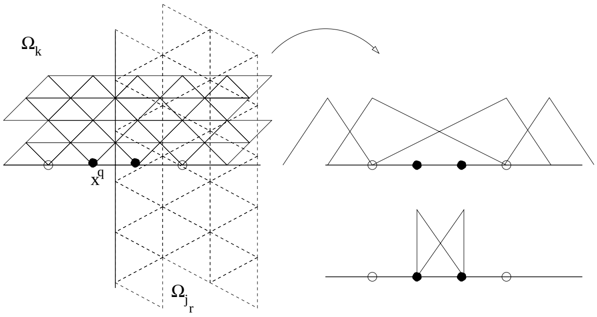


Figure 3: The spaces $W_k^{l,0}$ and $W_k^{l,q}$ (only two subdomains have been represented). In the case presented here, the dimension of $W_k^{l,q}$ is two.

Remark 4 *Conditions (25, 26) are stronger than (5), since* $W_k^{l,0} = W_k^l$.

We have to specify the spaces $W_k^{l,q}$, for $q \geq 1$. Call $(x^q)_{1\leq q\leq Q}$ the crosspoints $x^q \in \mathcal{X}_k^l$. For a crosspoint x^q, (assume that $\{x^q\} = \Gamma_k^l \cap \Gamma_{j_r}^{l'}$), see Figure 3, we call $\{x_{m_1+1}, \dots, x_{m_2-1}\}$ the nodes of $\mathcal{T}_k^l$ for which the support of the corresponding basis function of X_k intersects the edge $\Gamma_{j_r}^{l'}$.

We call $\widetilde{\phi}_{m_1+1}$ the piecewise linear and continuous (except at x_{m_1+1}) function, vanishing outside $[x_{m_1+1}, x_{m_1+2})$, linear on $[x_{m_1+1}, x_{m_1+2})$, and equal to 1 at x_{m_1+1} and 0 at x_{m_1+2}. Likewise, we call $\widetilde{\phi}_{m_2-1}$ the piecewise linear and continuous except at x_{m_2-1} function, vanishing outside $(x_{m_2-2}, x_{m_2-1}]$, linear on $(x_{m_2-2}, x_{m_2-1}]$, and equal to 1 at x_{m_2-1} and 0 at x_{m_2-2}.

We define $W_k^{l,q} \equiv span(\widetilde{\phi}_{m_1+1}, \phi_{m_1+2} \dots, \phi_{m_2-2}, \widetilde{\phi}_{m_2-1})$. The spaces $W_k^{l,0}$ and $W_k^{l,q}$ are displayed on Figure 3. Note that with this choice of $W_k^{l,q}$, the supports of the functions in $W_k^{l,q}$ do not intersect the supports of the functions of X_k^{l,x_q}.

We have obviously

$$dim(\widetilde{W_k^l}) = dim(\bigoplus_{q=0}^{Q} W_k^{l,q}).$$

Now we need to define the coefficients λ_{qi}. We set $\lambda_{0i} = 1$, for all $1 \leq i \leq n_k^l$. For $k \geq 1$, assume that $\{x_q\} = \Gamma_k^l \cap \Gamma_{j_r}^{l'}$. Then we set $\lambda_{qr} = 0$ and $\lambda_{qi} = 1$, for all $1 \leq i \leq n_k^l$, $i \neq r$.

Then Y is the subspace of X defined by

$$Y \equiv \{u \in X; \forall k \in \{1 \dots K\}, \forall l \in \{1 \dots E_k\}, u \text{ satisfies (25), (26). } \}. \qquad (27)$$

Remark 5 *Let $u = (u_k) \in Y$. Then it is very clear from (25), (26) that all the nodal values of u_k located on $\partial\Omega_k$ except at the vertices of $\partial\Omega_k$ can be found from the d.o.f. in the adjacent subdomains and from the d.o.f. located at the vertices of $\partial\Omega_k$. With this matching condition, all the nodal values located on $\partial\Omega_k$ except at the vertices of $\partial\Omega_k$ are slave nodal values.*

Remark 6 *Finding the slave nodal values can be achieved in two steps :*

1. *find the unknown located at the black nodes on Figure 3, by taking the test functions in the spaces $W_k^{l,q}, q > 0$. This corresponds to solving a small linear system with a mass matrix for each crosspoint on Γ_k^l.*
2. *find the remaining nodal values (located on $\Gamma_k^l \backslash \mathcal{V}_k$) by solving a problem of the type (13). We have seen above that this problem is well posed under conditions (11).*

It can be proved that Theorem 1 also holds for these strengthened matching conditions.

References

[AAH$^+$98]Y. Achdou, G. Abdulaiev, J.C. Hontand, Y.Kuznetsov, O. Pironneau, and C.Prud'homme. Non matching grids for fluids. In J. Mandel, C Farhat, and X.C. Cai, editors, *Domain Decomposition Methods 10*, pages 377–383. AMS, 1998. Proceedings from the Tenth International Conference, June 1997, Boulder, Colorado.

[AMW99]Y. Achdou, Y. Maday, and O. Widlund. Substructuring preconditioners for the mortar method in dimension two. *SIAM J. of Numerical Analysis.*, 32(2):551–580, 1999.

[BMP94]Christine Bernardi, Yvon Maday, and Anthony T. Patera. A new non conforming approach to domain decomposition: The mortar element method. In Haim Brezis and Jacques-Louis Lions, editors, *Collège de France Seminar.* Pitman, 1994. This paper appeared as a technical report about five years earlier.

[BSS72]M. Berger, R. Scott, and G. Strang. Approximate boundary conditions in the finite element method. In *Symposia Mathematica.* x, 1972.

[CDS99]X.-C. Cai, M. Dryja, and M. V. Sarkis. Overlapping nonmatching grid mortar element methods for elliptic problems. *SIAM J. Numer. Anal.*, 36:581–606, 1999.

[CH90]G. Chesshire and W. Henshaw. Composite overlapping meshes for the solution of partial differential equations. *J. Comp. Phy*, 90:1–64, 1990.

[Cia78]Philippe G. Ciarlet. *The Finite Element Method for Elliptic Problems.* North-Holland, Amsterdam, 1978.

[HLP99]F. Hecht, J.L. Lions, and O. Pironneau. Domain decomposition algorithm for computed aided design. In A. Sequeira et al, editor, *Applied Nonlinear Analysis*, pages 185–198. Kluwer Academic-Plenum Publishers, New York, 1999.

[Kuz97]Yuri. A. Kuznetsov. Overlapping domain decomposition with non matching grids. In Petter E. Bjørstad, Magne Espedal, and David Keyes, editors, *Domain Decomposition Methods in Sciences and Engineering.* J. Wiley, 1997. Proceedings from the Ninth International Conference, June 1996, Bergen, Norway.

[LP99]J.L. Lions and O. Pironneau. Domain decomposition methods for cad. *C.R. Acad. Sci. Paris*, 328:73–80, 1999.

[SB87]J. Steger and J. Benek. On the use of composite grid schemes in computational aerodynamics. *Comp. Meth. Appl. Mech. Eng.*, 64:301–320, 1987.

[SF73]Gilbert Strang and George J. Fix. *An Analysis of the Finite Element Method.* Prentice-Hall, Englewood Cliffs, N.J., 1973.

[YM00]Y.Achdou and Y. Maday. The mortar element method with overlapping subdomains. Technical report, Universite Pierre et Marie Curie, 2000. submitted.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau,

8. Mesh adaptivity in the mortar finite element method

by Christine BERNARDI [1], Frédéric HECHT [2]

Introduction

The mortar element method [BMP94, BMP93] becomes an important tool for mesh adaptivity in finite elements. Indeed, completely independent finite element discretizations can be used on the subdomains of a nonconforming partition of the initial domain without overlapping. This solves the contradiction between conformity and regularity and allows for working on a fully adapted mesh with a much smaller number of degrees of freedom.

In the case of the Laplace equation in a polygon Ω

$$\begin{cases} -\Delta u = f & \text{in}\,\Omega, \\ \\ \;\; u = 0 & \text{on}\,\partial\Omega. \end{cases} \tag{1}$$

the *a priori* analysis of the discrete problem obtained at each step of adaptivity is performed in [BM00], and optimal estimates are proved. The first results of *a posteriori* analysis [BOV99, Woh99] required saturation assumptions. However, as explained in [BH00], these assumptions can be avoided for some appropriate residual type error indicators, and fully optimal estimates are derived for the Laplace equation, in the sense that the error in the energy norm is equivalent to the Hilbertian sum of error indicators, up to some negligible terms related to the data. We recall these estimates and prove the efficiency of these error indicators thanks to some numerical experiments.

An outline of this paper is as follows. In the second section, we describe the discrete problem. Next we introduce the error indicators and recall from [BH00] the results of *a posteriori* analysis, which consist of a global upper bound for the error and a local upper bound for each indicator. In the third section, we describe our adaptivity algorithm and present some numerical experiments that seem in good coherency with the previous estimates.

The discrete problem and its error indicators

Let $(\mathcal{T}_h^0)_{h^0}$ be a family of "coarse" triangulations of the domain Ω, in the usual sense: each $\mathcal{T}_h^0$ is a finite set of triangles such that $\overline{\Omega}$ is the union of these triangles and the intersection of two different elements of $\mathcal{T}_h^0$, if not empty, is a vertex or a whole edge of both of them. As usual, h^0 denotes the maximal diameter of the elements of $\mathcal{T}_h^0$. We make the further assumption that this family is regular, *i.e.* there exists a positive

[1] C.N.R.S. et Université Pierre et Marie Curie, bernardi@ann.jussieu.fr

[2] Université Pierre et Marie Curie, hecht@ann.jussieu.fr

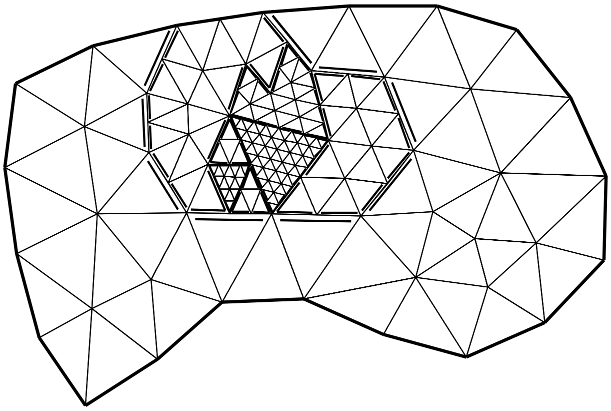


Figure 1: Example of mesh

constant σ such that, for all h^0 and for all K in $\mathcal{T}_h^0$, the ratio of the diameter of K to the diameter of its inscribed circle is smaller than σ.

Starting from this family $(\mathcal{T}_h^0)_{h^0}$, we build iteratively new families of refined triangulations as follows. Assuming that the family $(\mathcal{T}_h^{n-1})_{h^{n-1}}$ is known, for each value of the parameter h^{n-1},
• for arbitrary positive integers ℓ, we cut some elements of $\mathcal{T}_h^{n-1}$ into $2^{2\ell}$ subtriangles by iteratively joining the midpoints of the edges of these elements,
• we denote by $\mathcal{T}_h^{n,k}$ the set of these triangles which have area equal to 2^{-2k} the area of the triangle K of $\mathcal{T}_h^0$ in which they are contained, and by K^n the largest value of k such that $\mathcal{T}_h^{n,k}$ is not empty,
• we denote by $\Omega^{n,k}$ the open subdomain of Ω such that $\overline{\Omega}^{n,k}$ is the union of the triangles of $\mathcal{T}_h^{n,k}$,
• and we call $\mathcal{T}_h^n$ the union of the $\mathcal{T}_h^{n,k}$.
Figure 1 illustrates a triangulation $\mathcal{T}_h^n$ (with $K^n = 3$). The discretization parameter δ is now the pair (n, h^n), where h^n denotes the maximal diameter of the elements of $\mathcal{T}_h^n$.

Next, at each step n, we define the skeleton

$$\mathcal{S}^n = \bigcup_{k=0}^{K^n} \partial\Omega^{n,k} \setminus \partial\Omega, \tag{2}$$

and, as standard in the mortar method [BMP94], we fix a decomposition of it into disjoint (open) mortars

$$\overline{\mathcal{S}}^n = \bigcup_{m=1}^{M^n} \overline{\gamma}^m \quad \text{and} \qquad \gamma^m \cap \gamma^{m'} = \emptyset, \quad 1 \le m < m' \le M^n. \tag{3}$$

We make the final assumption that each $\overline{\gamma}^m$, $1 \le m \le M^n$, is a whole edge of a triangle of one of the triangulations $\mathcal{T}_h^{n,k}$, located on one side of γ^m, and that, on the other side, it is the union of edges of triangles in $\mathcal{T}_h^{n,k_1} \cup \dots \cup \mathcal{T}_h^{n,k_p}$, where all k_i are $> k$. We agree to denote by $k(m)$, $k_1(m), \dots, k_p(m)$, the corresponding exponents k, $k_1, \dots, k_p$, and by $p(m)$ the number p. We call $\mathcal{E}^m$ the set of connected components of $\overline{\gamma}^m \cap \partial\Omega^{n,k_i(m)}$, $1 \le i \le p(m)$. For simplicity, we assume from now on that there exists a constant λ independent of δ such that

$$\forall n, \quad \sup_{1 \le m \le M^n} \ \sup_{1 \le i \le p(m)} k_i(m) - k(m) \le \lambda. \tag{4}$$

We fix an integer $\ell \ge 2$ and, with each value of δ, we associate the local discrete spaces, for $0 \le k \le K^n$,

$$X^{n,k} = \left\{ v_k \in \mathcal{C}^0(\overline{\Omega}^{n,k});\ \forall K \in \mathcal{T}_h^{n,k},\ v_{k\,|K} \in \mathcal{P}_\ell(K) \right\}, \tag{5}$$

where $\mathcal{P}_\ell(K)$ stands for the space of restrictions to K of polynomials with total degree $\le \ell$.

Remark 1 *We refer to* [BH00] *for the analysis of the case of piecewise affine functions ($\ell = 1$), where some restrictions on the decomposition are needed.*

Let now γ^m, $1 \le m \le M^n$, be one of the mortars. With each e in $\mathcal{E}^m$, we associate the space $\widetilde{W}^m(e)$ of continuous functions on e such that their restrictions to each edge $e' = \overline{\gamma}^m \cap \partial K$ for all K in $\mathcal{T}_h^{n,k_i(m)}$ belongs to $\mathcal{P}_{\ell-1}(e')$ if e' contains an endpoint of e, to $\mathcal{P}_\ell(e')$ if not.

The discrete space $\mathbb{X}_\delta$ is now defined in the usual way, see [BMP93]. It is the space of functions v_δ such that

- their restrictions to each $\Omega^{n,k}$, $0 \le k \le K^n$, belong to $X^{n,k}$,
- they vanish on $\partial\Omega$,
- the following matching condition holds on any γ^m, $1 \le m \le M^n$,

$$\forall e \in \mathcal{E}^m,\ \forall \chi \in \widetilde{W}^m(e), \quad \int_e [v_\delta](\tau)\chi(\tau)\,d\tau = 0, \tag{6}$$

where $[v_\delta]$ denotes the jump of v_δ through e.

Remark 2 *As proposed in the first version of the mortar method* [BMP94]*, some further matching conditions can be added, more precisely the functions in $\mathbb{X}_\delta$ can be enforced to be continuous at the endpoints of all γ^m. These conditions are satisfied in the numerical experiments of this paper, but they are not necessary for the analysis.*

For fixed data f in $L^2(\Omega)$, the discrete problem now reads:

Find u_δ in $\mathbb{X}_\delta$ such that

$$\forall v_\delta \in \mathbb{X}_\delta, \quad a_\delta(u_\delta, v_\delta) = \int_\Omega f(\mathbf{x}) v_\delta(\mathbf{x})\,d\mathbf{x}, \tag{7}$$

where the bilinear form $a_\delta(\cdot,\cdot)$ is defined by

$$a_\delta(u_\delta, v_\delta) = \sum_{k=0}^{K^n} \int_{\Omega^{n,k}} \mathbf{grad}\, u_\delta \cdot \mathbf{grad}\, v_\delta\,d\mathbf{x}. \tag{8}$$

Thanks to the matching conditions (6), it is readily checked that this problem has a unique solution. Moreover, the following *a priori* error estimate can be proved as an extension of [BM00, Thm 2.8]: if the solution u of problem (1) is such that each $u_{|\Omega^{n,k}}$, $0 \leq k \leq K^n$, belongs to $H^{s_k}(\Omega^{n,k})$, $s_k > 1$,

$$\|u - u_\delta\|_{H^1_\delta(\Omega)} \leq c\,\Big(\sum_{k=0}^{K^n} (2^{-k}\,h^0)^{2(s_k-1)}\,\|u\|^2_{H^{s_k}(\Omega^{n,k})}\Big)^{\frac{1}{2}}, \tag{9}$$

where the mesh-dependent norm $\|\cdot\|_{H^1_\delta(\Omega)}$ is defined by

$$\|v\|_{H^1_\delta(\Omega)} = \Big(\sum_{k=0}^{K^n} \|v\|^2_{H^1(\Omega^{n,k})}\Big)^{\frac{1}{2}}. \tag{10}$$

Remark 3 *As explained in* [BB99]*, the matching conditions* (6) *can be enforced thanks to to the introduction of a Lagrange multiplier. In this case, problem* (7) *is equivalent to a saddle-point problem. The corresponding global matrix is symmetric, so that solving it is not expensive.*

However, we are more specifically interested with *a posteriori* estimates. As usual, we fix an approximation f_δ of the function f in the space

$$\mathbb{Z}_\delta = \big\{g_\delta \in L^2(\Omega);\ \forall K \in \mathcal{T}_h^n,\ g_{\delta\,|K} \in \mathcal{P}_{\ell^*}(K)\big\}. \tag{11}$$

where ℓ^* is a nonnegative integer. We consider two types of indicators.

- Error indicators linked to the finite elements

For each K in $\mathcal{T}_h^n$, we denote by $\mathcal{E}_K$ the set of edges of K which are not contained in $\partial\Omega$. In what follows, h_K stands for the diameter of K and h_e for the length of any e in $\mathcal{E}_K$ (or in any $\mathcal{E}^m$).

The residual error indicator η_K associated with any triangle in $\mathcal{T}_h^n$ is now defined in a completely standard way, see [Ver96, (1.18)]:

$$\eta_K = h_K\,\|f_\delta + \Delta u_\delta\|_{L^2(K)} + \frac{1}{2}\sum_{e\in\mathcal{E}_K} h_e^{\frac{1}{2}}\,\|\,[\partial_n u_\delta]\,\|_{L^2(e)}, \tag{12}$$

where ∂_n denotes the normal derivative on e and $[\,\cdot\,]$ the jump through e. Note that the term "residual" here means that, when suppressing all the δ in the previous line, the quantity in the right-hand side is zero.

- Error indicators linked to the edges of the skeleton

Like in [BV96, (3.3)], for $1 \leq m \leq M^n$, we associate with each e in $\mathcal{E}^m$ the indicator η_e defined as

$$\eta_e = h_e^{-\frac{1}{2}}\,\|\,[u_\delta]\,\|_{L^2(e)}. \tag{13}$$

There also, this quantity vanishes when suppressing the δ.

Remark 4 *It is readily checked that, for all m, $1 \leq m \leq M^n$, and for all e in $\mathcal{E}^m$, the quantity η_e is equivalent to the norm $\|\,[u_\delta]\,\|_{H^{\frac{1}{2}}(e)}$. However it is much easier to compute.*

We sum up in the two following propositions the estimates concerning these indicators, and we refer to [BM00, §3] for their detailed proofs. The first one relies on the formula, obtained by local integration by parts,

$$\begin{aligned}\|u-u_\delta\|_{H^1_\delta(\Omega)} \leq \;& c\Big(\sum_{K\in\mathcal{T}_h^n}\Big(\|f_\delta+\Delta u_\delta\|_{L^2(K)}\,\frac{\|v-v_\delta\|_{L^2(K)}}{\|v\|_{H^1_\delta(\Omega)}}\\ &+\|f-f_\delta\|_{L^2(K)}\,\frac{\|v-v_\delta\|_{L^2(K)}}{\|v\|_{H^1_\delta(\Omega)}}\\ &-\frac{1}{2}\sum_{e\in\mathcal{E}_K}\|[\partial_n u_\delta]\|_{L^2(e)}\,\frac{\|v-v_\delta\|_{L^2(e)}}{\|v\|_{H^1_\delta(\Omega)}}\Big)\\ &+|\sum_{m=1}^{M^n}\sum_{e\in\mathcal{E}^m}\int_e \partial_n(u-u_\delta)\,[u_\delta]\,d\tau|^{\frac{1}{2}}\Big),\end{aligned} \tag{14}$$

where v is equal to $u-u_\delta$ and v_δ is any "conforming" approximation of v, *i.e.* which belongs to $\mathbb{X}_\delta\cap H^1_0(\Omega)$. The first three terms are evaluated by constructing an extension of Clément's operator to the present situation, while estimating the last one relies on conditions (6). The arguments for proving the second proposition are standard for residual indicators, see [Ver96, Chap.3].

Proposition 1 *There exists a constant c independent of δ such the following error estimate holds between the solutions u of problem* (1) *and u_δ of problem* (7)*:*

$$\|u-u_\delta\|_{H^1_\delta(\Omega)} \leq c\Big(\sum_{K\in\mathcal{T}_h^n}\big(\eta_K^2+h_K^2\,\|f-f_\delta\|^2_{L^2(K)}\big)+\sum_{m=1}^{M^n}\sum_{e\in\mathcal{E}^m}\eta_e^2\Big)^{\frac{1}{2}}. \tag{15}$$

Proposition 2 *There exists a constant c' independent of δ such that the following estimate holds for all K in $\mathcal{T}_h^n$:*

$$\eta_K \leq c'\,\big(\|u-u_\delta\|_{H^1_\delta(\Xi_K)}+\big(\sum_{K'\subset\Xi_K}h_{K'}^2\,\|f-f_\delta\|^2_{L^2(K')}\big)^{\frac{1}{2}}\big), \tag{16}$$

where Ξ_K is the union of at most four triangles such that at least an edge of K is contained in an edge of such triangles. There exists a constant c'' independent of δ such that the following estimate holds for all m, $1\leq m\leq M^n$, and for all e in $\mathcal{E}^m$:

$$\eta_e \leq c''\,\|u-u_\delta\|_{H^1_\delta(\Xi_e)}, \tag{17}$$

where Ξ_e is the union of the triangle of $\mathcal{T}_h^{n,k(m)}$ that intersects γ^m and a triangle K contained in an $\overline{\Omega}^{n,k_i(m)}$ such that e is an edge of K.

Remark 5 *The constants c, c' and c'' in the previous propositions only depend on the regularity parameter σ of the initial family of triangulations $(\mathcal{T}_h^0)_{h^0}$ and on the constant λ introduced in* (4)*. Their dependency with respect to λ is explicitly written in* [BH00].

Proposition 2 states a local version of the upper bounds for the error indicators. However a global version can be proven by similar arguments. When compared with (15), this global estimate proves that the error $\|u - u_\delta\|_{H^1_\delta(\Omega)}$ is equivalent to the Hilbertian sum of the indicators, up to some terms concerning the approximation of the data f. These terms are most often negligible, so that the combined two types of indicators lead to an optimal evaluation of the error.

The adaptivity algorithm and numerical experiments

Thanks to the previous choice of the discrete problem, the algorithm for mesh adaptivity is now straightforward. Assume that we are given a triangulation $\mathcal{T}_h^n$ and the corresponding skeleton $\mathcal{S}^n$. We solve the associated problem (7) and compute all the error indicators η_K and η_e, next the meanvalue $\overline{\eta}^n$ of the η_K, $K \in \mathcal{T}_h^n$, and the mean value $\overline{\eta}^n_*$ of the η_e, $e \in \mathcal{E}^m$, $1 \leq m \leq M^n$. The next triangulation $\mathcal{T}_h^{n+1,k}$ and skeleton $\mathcal{S}^{n+1}$ are then built in two steps.

- Step 1. For all K in $\mathcal{T}_h^n$, there exists an integer k such that

$$2^k\,\overline{\eta}^n \leq \eta_K \leq 2^{k+1}\,\overline{\eta}^n. \tag{18}$$

If k is positive, we cut the triangle K into 2^{2k} equal subtriangles by iteratively joining the middle of the edges. This allows for defining an intermediary skeleton $\mathcal{S}^n_*$.

- Step 2. We only consider the e in $\mathcal{E}^m$, $1 \leq m \leq M^n$, such that

$$\eta^e \geq 2\,\overline{\eta}^n_*. \tag{19}$$

If this edge remains in $\mathcal{S}^n_*$ after Step 1, we cut the triangles on both sides of e, such that e becomes "conforming", *i.e.* it is no longer contained in the next skeleton $\mathcal{S}^{n+1}$.

We stop the algorithm either after a finite number of iterations or when the following condition is satisfied for a given tolerance ε:

$$\sum_{K\in\mathcal{T}_h^n} \eta_K^2 + \sum_{m=1}^{M^n} \sum_{e\in\mathcal{E}^m} \eta_e^2 \leq \varepsilon^2. \tag{20}$$

We now present some numerical results in the case where Ω is the L-shaped domain $]-1,1[^2\backslash[0,1]^2$ and the data f is equal to 1. We work with piecewise quadratic functions ($\ell = 2$) and an initial mesh $\mathcal{T}_h^0$ made of 64 triangles. The dimension of the corresponding space $\mathbb{X}_\delta$ is 105.

Figure 2 presents the initial mesh $\mathcal{T}_h^0$ and the first five refined meshes $\mathcal{T}_h^n$, $n = 1$, 2, 3, 4, 5, according to the previous algorithm.

Figure 3 presents the isovalue curves of the error indicators η_K, $K \in \mathcal{T}_h^n$, for each of the previous $\mathcal{T}_h^n$.

Table 1 presents for each triangulation $\mathcal{T}_h^n$ the number of triangles N_T^n, the total number of mortars M^n, the dimension $\dim \mathbb{X}_\delta$ of the corresponding space $\mathbb{X}_\delta$, the maximal value K^n of the k and finally the Hilbertian sum η^n_{norm} of all indicators η_K and η_e. It can be observed that, for a fixed initial mesh, this sum decreases at each iteration n.

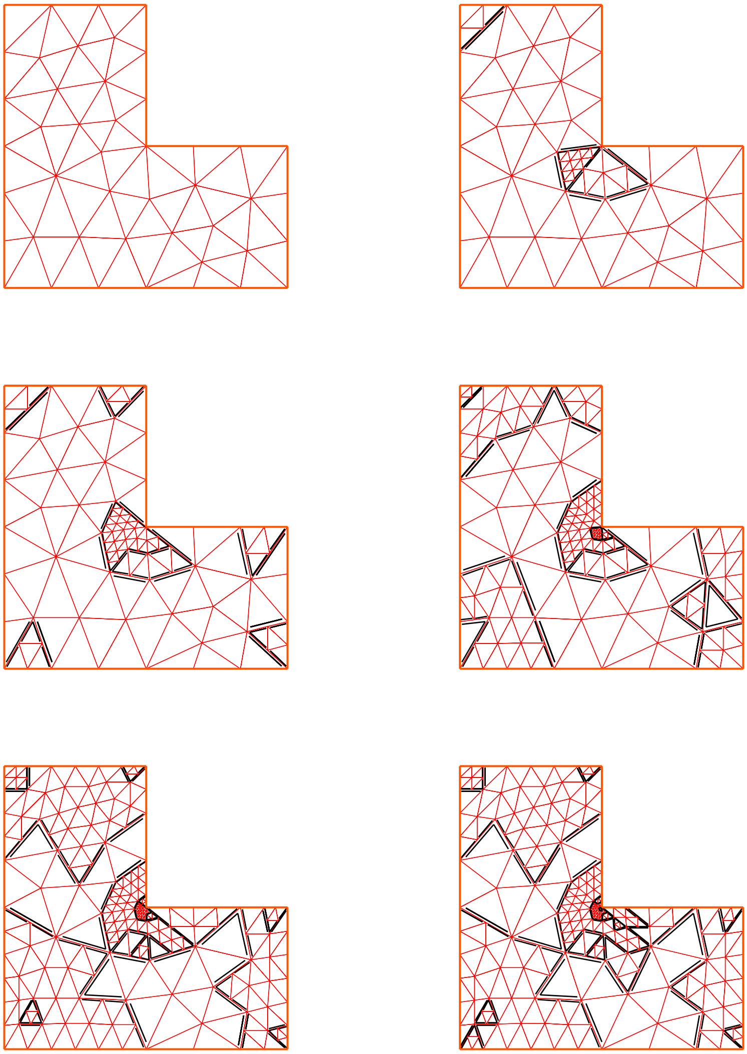


Figure 2: The sequence of adapted meshes

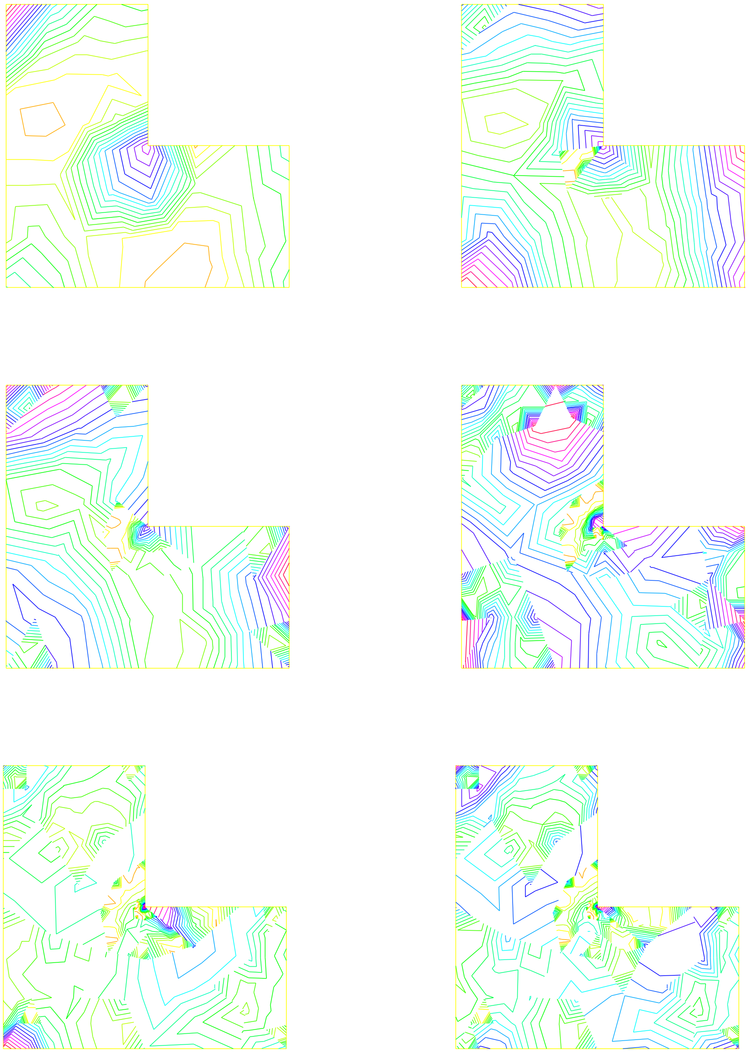


Figure 3: Isovalue curves of the indicators

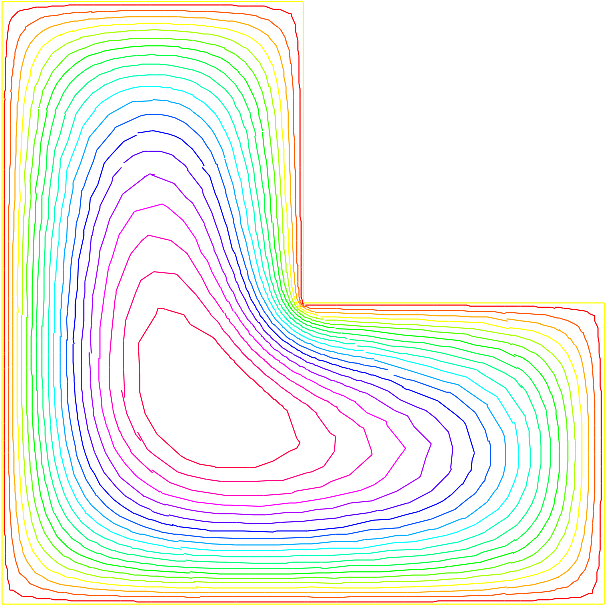


Figure 4: Isovalue curves of the solution

n	0	1	2	3	4	5
N_T^n	64	88	121	214	301	325
M^n	0	8	18	32	47	59
$\dim \mathbb{X}_\delta$	105	171	256	466	661	730
K^n	0	2	2	4	4	5
η_{norm}^n	0.3438	0.2187	0.1587	0.0927	0.0677	0.0562

Table 1

The isovalue curves of the discrete solution obtained on the mesh $\mathcal{T}_h^5$ are presented in Figure 4 .

References

[BB99]Faker Ben Belgacem. The mortar finite element method with Lagrange multipliers. *Numer. Math.*, 84(2):173–197, 1999.

[BH00]Christine Bernardi and Frédéric Hecht. Error indicators for the mortar finite element discretization of the Laplace equation. *Math. Comput.*, 2000. (submitted).

[BM00]Christine Bernardi and Yvon Maday. Mesh adaptivity in finite elements using the mortar method. *Revue Européeenne des Éléments Finis*, (9):451–465, 2000.

[BMP93]Christine Bernardi, Yvon Maday, and Anthony T. Patera. Domain decomposition by the mortar element method. In H.G. Kaper ans M. Garbey, editor, *Asymptotic and Numerical Methods for Partial Differential Equations with Critical Parameters*, pages 269–286. N.A.T.O. ASI, Kluwer Academic Publishers, 1993.

[BMP94]Christine Bernardi, Yvon Maday, and Anthony T. Patera. A new non conforming approach to domain decomposition: The mortar element method. In Haim Brezis and Jacques-Louis Lions, editors, *Collège de France Seminar*. Pitman, 1994. This paper appeared as a technical report about five years earlier.

[BOV99]Christine Bernardi, Robert G. Owens, and José Valenciano. An error indicator for mortar element solutions to the Stokes problem. Technical Report 99030, Laboratoire d'Analyse Numérique, Université Pierre et Marie Curie, Paris, 1999.

[BV96]D. Braess and R. Verfürth. A posteriori error estimators for the Raviart–Thomas element. *SIAM J. Numer. Anal.*, 33:2431–2444, 1996.

[Ver96]Rüdiger Verfürth. *A Review of A Posteriori Error Estimation and Adaptive Mesh-Refinement Techniques*. Wiley and Teubner, 1996.

[Woh99]Barbara I. Wohlmuth. A residual based error estimator for mortar finite element discretizations. *Numer. Math.*, 84:143–171, 1999.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau,

9. Adaptive ENO-Wavelet Transforms for Discontinuous Functions

T.F. Chan[1] H.M. Zhou[2]

Introduction

We have desiged an adaptive ENO-wavelet transform for approximating discontinuous functions without oscillations near the discontinuities. Our approach is to apply the one-side information idea from Essentially Non-Oscillatory (ENO) schemes for numerical shock capturing to standard wavelet transforms. This transform retains the essential properties and advantages of standard wavelet transforms such as concentrating the energy to the low frequencies and having a multiresolution framework and fast algorithms, all without any edge artifacts. Furthermore, we have obtained a rigorous uniform approximation error bound regardless of the presence of discontinuities. We will show some numerical examples and some applications to image compression.

It is well known that wavelet linear approximation (i.e. truncating the high frequencies) can approximate smooth functions very efficiently but cannot achieve similar results for piecewise continuous functions, especially functions with large jumps. Several problems arise near jumps, primarily caused by the well-known Gibb's phenomenon. The jumps generate large high frequency wavelet coefficients and thus linear approximations cannot get the same high accuracy near discontinuties as in the smooth region.

To overcome these problems within the standard wavelet transform framework, non-linear data-dependent approximations, which selectively retain certain high frequency coefficients, are often used, e.g. hard and soft thresholding techniques, see [Don95],[Mal98]. Another way is to construct orthonormal basis to represent the discontinuities, such as Donoho's wedgelets [Don97], rigdelets [CD99b], and curvelets [CD99a], and Mallat's bandelets [Mal00].

A different aproach is to modify the wavelet transform to not generate large wavelet coefficients near jumps. Claypoole, Davis, Sweldens and Baraniuk [PCB99] proposed an adaptive lifting scheme which lowers the order of approximation near jumps, thus minimizing the Gibbs' effect. We use a different approach in developing our ENO-wavelet transforms by borrowing the well developed Essentially Non-Oscillatory (ENO) technique for shock capturing in computational fluid dynamics (e.g. see [AHC87]) to modify the standard wavelet transform near discontinuities so that the Gibbs' phenomenon can be completely removed. ENO schemes are systematic ways of adaptively defining piecewise polynomial approximations of the given functions according to their smoothness. A crucial point in designing ENO schemes is to use one-sided information near jumps, and never differencing across the discontinuities.

Combining the ENO idea with the multiresolution data representation is a natural

[1] Department of Mathematics, UCLA, chan@ipam.ulca.edu. Research supported in part by grants ONR-N00017-96-1-0277 and NSF DMS-9973341. The material in this report is patent pending.

[2] Department of Applied Mathematics, Caltech, hmzhou@acm.caltech.edu

way to avoid oscillations in the approximations. In fact, it has been explored by Harten in his general framework of multiresolution [Har94], which is similar to the lifting scheme of Sweldens [Swe97]. However, his method was not developed to be directly applied to the widely used pyramidal filtering algorithms which the standard wavelet transforms are usually implemented in.

The way we accomplish this is to not change the wavelet transforms or the filter coefficients, which most data dependent multiresolution algorithms do, but instead locally change the function near the discontinuities in such a way that the standard filters are only applied to smooth data, and therefore no large high frequency coefficients are generated. By recording how the changes are make, the original discontinuous function can be exactly recovered by using the original inverse filters. We show that the resulting wavelet transform retains all the desirable properties of the standard transform. The extra cost (in floating point operations) required is insignificant, which, in fact, is of order $O(dl)$ where d is the number of discontinuities and l the filter length.

The arrangement of the paper is as follows. In the next section, we give a general algorithm to implement the ENO-wavelet transform discretely. And we also state the rigorous uniform error estimate in this section. In the last section, we give some numerical examples to illustrate the main advantage of the ENO-wavelet transforms, including some examples in image compression.

In this short proceeding paper, we are forced to leave out many mathematical details, and we aim only to give a general idea of the algorithms and the numerical results. For more details, see [CZ99].

ENO-wavelet Transforms

First, we briefly review the standard discrete wavelet transforms, e.g. see [Dau92], [Mal98] and [SN96]. In practice, discrete wavelet transforms are often used by starting with a set of discrete numbers which are the low frequency coefficients of the L^2 function $f(x)$ at the finest level. In many applications, this set of numbers are sample values of the function $f(x)$ on a fine grid (although in [SN96], this is called a "wavelet crime"). Let $\alpha_{j,k}$ ($\beta_{j,k}$) denote the low (high) frequency coefficients at level j. The wavelet transform coefficients at a coarser level $j-1$ can be computed by:

$$\alpha_{j-1,k} = \sum_{s=0}^{l} c_s \alpha_{j,2k+s}; \qquad \beta_{j-1,k} = \sum_{s=0}^{l} h_s \alpha_{j,2k+s}, \tag{1}$$

where c_s (h_s) are called low (high) pass filters. It is well known that the inverse transforms can be easily formed by using orthogonality of the wavelet transforms. The linear approximation refers to reconstructing $\alpha_{j,k}$ by setting the high frequencies $\beta_{j-1,k}$ to zero.

In Fig 1, the left picture is a piecewise continuous function (dotted) and its linear approximation (solid). The middle one is a zoom-in at a discontinuity. We clearly see oscillations near discontinuities. The right one is its DB-4 wavelet coefficients. We see that most of the high frequency coefficients are zero, except for a few large coefficients which these coefficients are computed near jumps. In this figure, we clearly see oscillations near discontinuities.

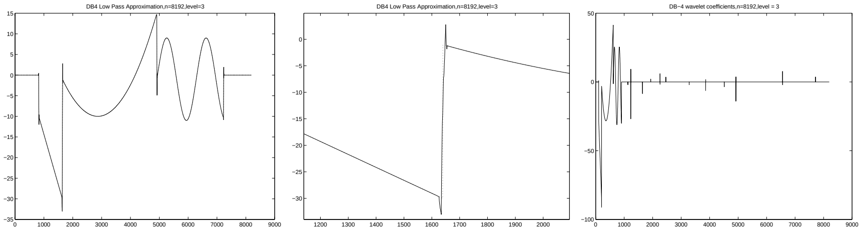


Figure 1: Left: The initial discontinuous function (dotted) and its linear approximation (solid). Middle: A zoom-in at a discontinuity. Oscillations are generated near the discontinuity. Right: its DB4 coefficients. Most of the high frequency coefficients (right part) are zero except for a few large coefficients computed near the jumps.

To simplify the presentation, we shall assume that the discontinuities in the functions are well-separated in the following sense:

Definition 1 *For a given wavelet filter with stencil length l, we say the j-th level approximation of the function $f(x)$ with spatial step $\Delta x = 2^{-j}$ satisfies the* **Discontinuity Separation Property** *(DSP) if $(l+2)\Delta x < t$, where $f(x)$ has discontinuous set D and t is the closest distance between any two discontinuous points.*

For any piecewise discontinuous function and a fixed stencil length l, an approximation will satisfy this DSP if j is sufficiently large, i.e. if the discretization is fine enough. On the other hand, at the place where the DSP is invalid, we will see that the approximations produced by the ENO-wavelet transforms are comparable to that by the standard wavelet transforms.

Now, we are ready to introduce the ENO-wavelet transforms. In addition to the standard wavelet transform computation, ENO-wavelet transforms have two more phases: locating the jumps and forming the approximations at the discontinuities. First, assuming knowledge of the location of the jumps, we give the ENO-wavelet approximations at the discontinuities by using one-sided information to avoid oscillations. Then, we give the methods to detect the location of the discontinuities. We also give the approximation error bound at the end of this section. In this short paper, we only consider Daubechies' orthonormal wavelets. The idea can be similarly extended to other wavelets.

The main idea of the ENO schemes for shock capturing is to use one-sided polynomial interpolations for data with large discontinuities. For ENO wavelets, we borrow this idea of using one-sided information to form the approximation and avoid applying the wavelet filters crossing the discontinuities.

The first way is to directly extend the function values, or in general the low frequencies on the finer level, at the discontinuity by p-th order extrapolation from both sides. For example, a straightforward way is to use p-point polynomial extrapolation. Least square can be used too [WA95]. Then one can apply the standard wavelet transforms on the extended functions by using (1) to compute the coarser level wavelet coefficients.

There is a storage problem for this direct function extrapolation. Indeed, it doubles the number of the wavelet coefficients near every discontinuity. To retain the perfect

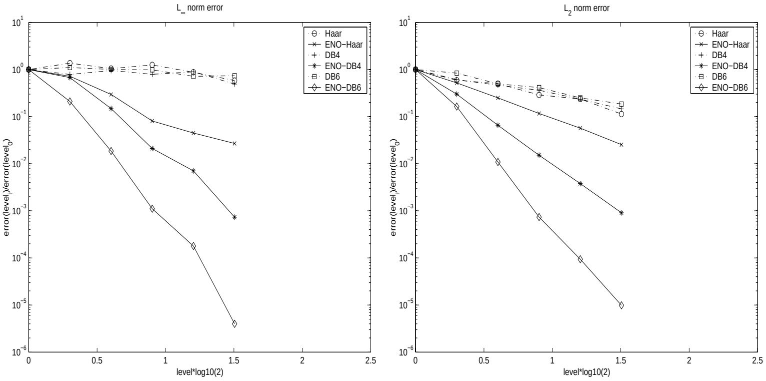


Figure 2: The approximation accuracy comparison of ENO-wavelet and standard wavelet transforms. Both L_∞ (left) and L_2 (right) order of accuracy show that ENO transforms maintain the order 1, 2 and 3 for ENO-Haar, ENO-DB4 and ENO-DB6 respectively and they agree with the theoretical results. In contrast, standard transforms do not retain the order.

invertible property, we need to store the ENO-wavelet coefficients from both sides. Thus, the output sequences are no longer the same size as the input sequences. In many applications, such as image compression, this extra storage requirement definitely needs to be avoided.

To keep the size of the output sequences the same as that of the input sequences without significant extra computation, we introduce the coarse level extrapolation schemes. The idea is to extrapolate the coarser level wavelet coefficients near the discontinuities instead of the function values or the finer level wavelet coefficients. Let us consider the extension from the left side first.

We have two choices: (1) We can extrapolate the low frequency coefficients $\alpha_{j-1,k}$ first, then determine the corresponding high frequency coefficients $\beta_{j-1,k}$. (2) Or we can first extend high frequency coefficients $\beta_{j-1,k}$, for example to zero, then determine the corresponding low frequency coefficients $\alpha_{j-1,k}$. By symmetry, we have two analogous choices for the right side of the jump.

The storage problem can be easily solved in both options. For example, we can store the high frequency coefficients for choice (1) and the low frequency coefficients for choice (2). The corresponding low frequency and high frequency coefficients can be easily recovered.

For each stencil crossing a jump, an extra cost (in floating point operation) is required in extrapolating low frequency coefficients, and in computing the corresponding high or low frequency coefficients. Overall, the extra cost over the standard wavelet transform is of order $O(dl)$. Compared to the cost of the standard transform, which is of order $O(nl)$ where n is the size of data, the ratio of the extra cost over that of the standard transform is $O(\frac{d}{n})$, which is independent of l and negligible when n is large.

Next, we introduce the methods to detect the location of the discontinuities for noisy and noise free functions. First we consider noise free data.

It is well known that for the smooth functions, we have $|\beta_{j,i}| = |f^{(p)}|O(\Delta x^p)$. In contrast, $|\beta_{j,i}|$ is at least one order lower than that if it involves a discontinuity So, an obvious way, also the cheapest way, to identify the discontinuities is to compare the magnitudes of the high frequency coefficients on the current standard stencils $|\beta_{j,i}|$ with that on the previous standard stencils $|\beta_{j,i-1}|$. Thus, we can design a method to detect the discontinuities as follows: If we have $|\beta_{j,i}| \leq a|\beta_{j,i-1}|$, where $a > 1$ is a given thresholding constant, then we treat the current stencil as a smooth stencil. Otherwise, we conclude that there are discontinuities contained in it.

The above described detection method may not be reliable if the function is polluted by noise, especially when the noise is "large". In this situation, we need to use heuristics to locate the exact position of the essential discontinuities. In many applications such as in image processing, large discontinuities in function value are the most significant features. A simple way to detect this kind of discontinuities is to look for these large magnitude high frequency coefficients and then compare the data values in the corresponding stencils to locate the exact jump positions.

Finally, we present the following uniform error estimate; the proof can be found in [CZ99]

Theorem 1 *Suppose the wavelets have finite support in* $[0, l]$*, and* p *vanishing moments,* $f(x)$ *is a piecewise continuous function in* $[a, b]$*, and* $f_j(x)$ *is its* j*-th level ENO-wavelet approximation. If the approximation* $f_{j+1}(x)$ *satisfies the DSP, then*

$$\|f(x) - f_j(x)\| \leq C(\Delta x)^p \|f^{(p)}(x)\|_{(a,b)\setminus D}, \tag{2}$$

where $\Delta x = 2^{-j}$ *and* D *is the jump set. The norm* $\|\cdot\|$ *can be either* L^2 *or* L^∞*.*

This theorem shows that the error in the ENO-wavelet approximation depends only on the size of the derivative of the function *away* from the discontinuities. In contrast, the error estimate for standard wavelet transforms depends on $\|f^{(p)}(x)\|_{(a,b)}$ which is unbounded at discontinuities. From an approximation point of view, the error bound for ENO-wavelets is as if the discontinuities were not there, and this is the best possible for discontinuous functions.

Numerical Examples

In this section, we give some 1-D and 2-D numerical examples by using the ENO-wavelet transforms. In particular, we show results for the ENO-Haar, ENO-DB4 and ENO-DB6 wavelet transforms.

To illustrate the performance of ENO-wavelet transforms, we show picture comparisons of the standard wavelet approximations (dash dotted in all figures) and corresponding ENO-wavelet approximations (solid). In addition, we compare their L_∞ and L_2 errors at level i: $E_{\infty,i}$ and $E_{2,i}$. Also, we compute the order of accuracy defined by: $Order_\infty = \log_2 \frac{E_{\infty,i}}{E_{\infty,i-1}}$, $Order_2 = \log_2 \frac{E_{2,i}}{E_{2,i-1}}$.

We apply Haar and ENO-Haar, DB4 and ENO-DB4, and DB6 and ENO-DB6 to this function and compare the approximation errors. Fig 2 shows the comparison of the order in L_∞ and in L_2 norm. It is clear that both the L_∞ and L_2 order of accuracy for ENO-wavelet transforms are of the order 1, 2 and 3 for ENO-Haar, ENO-DB4 and

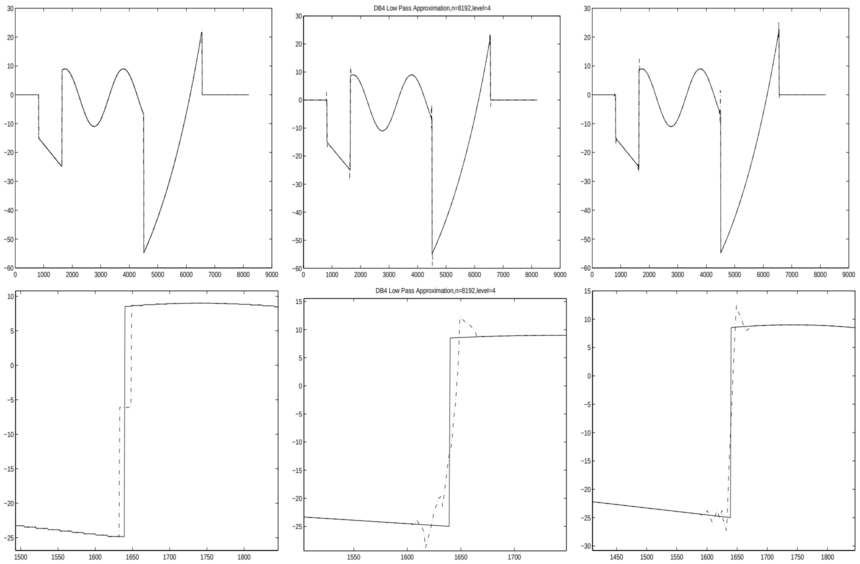


Figure 3: The 4-level ENO-Haar and Haar (left), ENO-DB4 and DB4 (middle), and ENO-DB6 and DB6 (right) approximation. The second row are corresponding zoom-ins near a discontinuity. We see the Gibbs' phenomenon in the standard approximation but not in the ENO approximation.

ENO-DB6 respectively, agreeing with the results of Theorem 1. In contrast, standard wavelet transforms do not retain the corresponding order.

To see the Gibbs' oscillations, we display the 4-level ENO-wavelet and standard wavelet approximations in the first row of Fig 3, for ENO-Haar (left), ENO-DB4 (middle) and ENO-DB6 (right) respectively. The second row are corresponding zoom-in at a same discontinuity. We clearly see the Gibbs' oscillations in the standard approximations. In contrast, the ENO-wavelet approximations preserve the jumps accurately.

In Fig 4, we also present the standard DB4 (dotted) and the ENO-DB4 (solid) wavelet coefficients respectively. There are some large standard high frequency coefficients (right part) related to the discontinuities. On the other hand, no large high frequency coefficients present in the ENO-wavelet coefficients.

The next 1-D example shows the ENO-DB6 wavelet transform applied to noisy data (see Fig. 5). Despite the presence of noise in the initial data (circles), the level-3 ENO-DB6 approximation (solid line) still retains the sharp edges (see zoom-in in the right picture) compared to the standard DB6 approximation (dash-dotted line) which not only has oscillations at the discontinuities but also smears them.

Finally, we give a 2-D image compression example to compare the standard Haar and the ENO-Haar approximations. Here we use tensor products of 1-D transforms. The original picture (left), the 3-level standard Haar (middle) and ENO-Haar (right) approximation are shown in Fig 6. Both approximations use the same number of low frequencies ($\frac{1}{64}$ of the original data). It is clear that in the standard Haar case, the

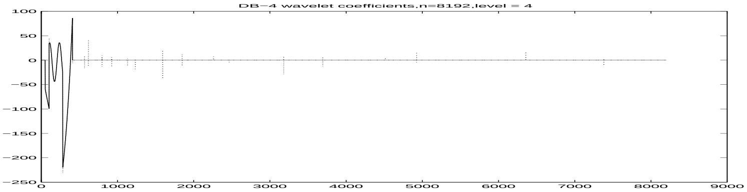


Figure 4: The 4-level ENO-DB4 (solid) and the standard DB4 (dotted) coefficients. There are large high frequency coefficients (right part) near the discontinuities in the standard transform but not in the ENO-DB4 transform.

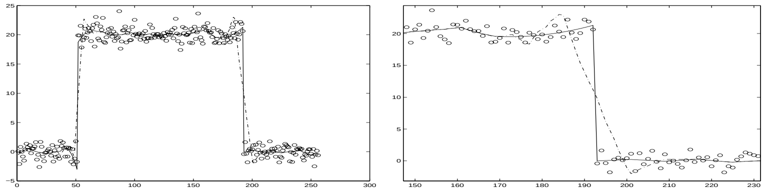


Figure 5: Left: The comparison of the 3-level ENO-DB6 (solid line) with the standard DB6 (dash-dotted line) approximation for noisy initial data (circles). Right: A zoom-in of the left example at a discontinuities. The ENO-DB6 approximation retains the sharp jumps but the standard DB6 approximation does not.

image becomes fuzzier than the ENO-Haar case. This illustrates that the ENO-Haar approximation can reduce the edge oscillations for 2-D images.

References

[AHC87]S. Osher A. Harten, B. Engquist and S. Chakravarthy. Uniformly high order essentially non-oscillatory schemes, iii. *Journal of Computational Physics*, 71:231–303, 1987.

[CD99a]E. Candes and D. Donoho. Curvelets: A surprisingly effective nonadaptive

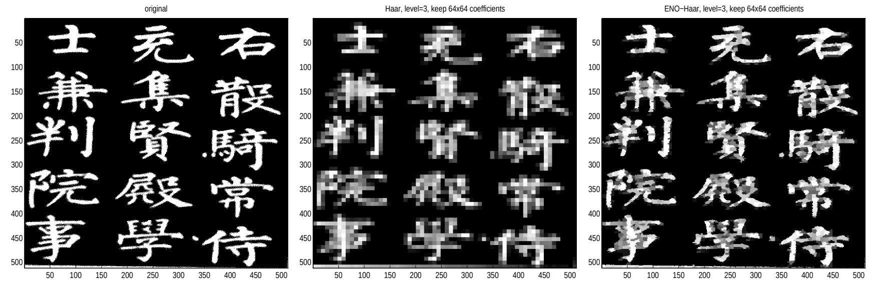

original
Haar, level=3, keep 64x64 coefficients
ENO−Haar, level=3, keep 64x64 coefficients
50
100
150
200
250
300
350
400
450
500

Figure 6: Original 2-d Function (left), The 3-level standard Haar (middle), and the 3-level ENO-Haar (right).

representation of objects with edges. Technical report, Department of Statistics, Stanford University, 1999.

[CD99b]E. Candes and D. Donoho. Ridgelets: a key to higher-dimensional intermittency? *Phil. Trans. R. Soc. Lond.*, A, 1999.

[CZ99]T. F. Chan and H. M. Zhou. Adaptive eno-wavelet transforms for discontinuous functions. Technical Report 99-21, Dept. of Math, UCLA, CAM Report,, June 1999.

[Dau92]I. Daubechies. *Ten Lectures on Wavelets.* SIAM, 1992.

[Don95]D. Donoho. De-noising by soft thresholding. *IEEE Trans. Inf. Th.*, 41:613–627, 1995.

[Don97]D. Donoho. Wedgelets: Nearly-minimax estimation of edges. Technical report, Tech Report, Dept. of Stat., Stanford Univ., 1997.

[Har94]A. Harten. Multiresolution representation of data, ii. general framework. Technical Report 94-10, Dept. of Math., UCLA., CAM Report, April 1994.

[Mal98]S. Mallat. *A Wavelet Tour of Signal Processing.* Academic Press, 1998.

[Mal00]S. Mallat. Geometrical image representations with bandelets. The talk at the IMA Workshop on Image Processing and Low Level Vision, held at Minneapolis, Minnesota, Oct 16-20, 2000, Oct. 2000.

[PCB99]W. Sweldens P. Claypoole, G. Davis and R. Baraniuk. Nonlinear wavelet transforms for image coding. Submitt to IEEE Tran. on Image Proc., 1999.

[SN96]G. Strang and T. Nguyen. *Wavelets and Filter Banks.* Wellesley-Cambridge Press, 1996.

[Swe97]W. Sweldens. The lifting scheme: A construction of second generation wavelets. *SIAM J. Math. Anal.*, 29(2):511–546, 1997.

[WA95]J. R. Williams and K. Amaratunga. A discrete wavelet transform without edge effects. Technical report, MIT, IESL Tech. Report, 1995.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau,

10. A Two-Stage Multi-Splitting Method for Non-Overlapping Domain Decomposition for Parabolic Equations

Daoud S. Daoud[1], Bruce A. Wade[2]

Domain Decomposition

In domain decomposition for parabolic partial differential equations (PDE) several approaches have been developed— breaking the domain into multiple subdomains of either overlapping or non-overlapping type, or using algebraic type splittings— *cf.* [CM94] for an overview. An important aspect is how to present the boundary conditions across interfaces or across common unknown points of subdomains, *cf.* [GS98, HT96, Tan92]. Towards parallelism, we divide the domain into subdomains with one grid point in common, adding an extra unknown at the interface to have effectively a non-overlapping decomposition.

In the present numerical method we have designed a one gridpoint overlap together with an extra equation in order to arrive at an effective multi-splitting approach. The transmission of data at the interface is through a discrete parametrized Robin boundary condition across interior interface points. A significant part of this report is the design and experimental study of optimizing boundary parameter coupled with particular choices of inner and outer splittings. We are interested here in extending some work of San and Tang [HT96] and Tang [Tan92] to parabolic problems. There is a parameter γ that acts like a feedback gain across the artificial interfaces. The primary aspect of this article is to construct and demonstrate effective multi-splitting methods as depending on the interface boundary condition.

Consider the numerical solution of parabolic problems of form:

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + f(x,t), \quad x \in (0,L), t > 0, \tag{1}$$

subject to initial condition $u(x,0) = g(x)$, $x \in \Omega := (0,L)$ and boundary conditions $u(0,t) = u(1,t) = 0$.

With the usual $(\Delta x, \Delta t)$ mesh in (x,t) $(\Delta x = 1/M, x_i = i\Delta x, i{=}0,\cdots,M, t_j{=} j\Delta t, j = 0,\cdots,N)$, let $\bar{x}$ be a point of interface by which the domain Ω is decomposed into two subdomains $\Omega_1 = \{x \in \Omega : x < \bar{x} + \Delta x\}$ and $\Omega_2 = \{x \in \Omega : x > \bar{x} - \Delta x\}$, where $\bar{x} = m\Delta x$, for some m with $1 < m < M-1$. We are utilizing just two subdomains to address the essential issues. Solutions u_1 and u_2 are restricted versions of u over the domains.

[1] Department of Mathematics, Eastern Mediterranean University, Famagusta, North Cyprus, Via Mersin 10, Turkey. daoud.daoud@emu.edu.tr.

[2] Department of Mathematical Sciences, University of Wisconsin– Milwaukee, Milwaukee, Wisconsin 53201-0413. wade@uwm.edu.

We introduce a boundary condition at $\bar{x}$ as follows

$$\alpha u_1 + (1-\alpha)\frac{\partial u_1}{\partial n}\bigg|_{\bar{x}-} = \alpha u_2 + (1-\alpha)\frac{\partial u_2}{\partial n}\bigg|_{\bar{x}+} \qquad \alpha \in [0,1]. \tag{2}$$

Then the global problem (1) is split over Ω_1 and Ω_2 in a natural fashion.

One can employ the method of lines with (1) through a second order central difference approximation [Smi85]: $\frac{\partial^2 u}{\partial x^2}\Big|_{(x_i,t_j)} = \Delta x^{-2}(u_{i-1,j} - 2u_{i,j} + u_{i+1,j}) + O(\Delta x^2)$. The normal derivative in the boundary condition (2) at $(\bar{x}, t) = (x_m, t)$ is approximated by forward or backward differences for Ω_1, and Ω_2, respectively, as follows:

$$\begin{aligned} \alpha u_1|_{m,j} &+ (1-\alpha)\frac{1}{\Delta x}\left(u_1|_{m+1,j} - u_1|_{m,j}\right) = \\ \alpha u_2|_{m,j} &+ (1-\alpha)\frac{1}{\Delta x}\left(u_2|_{m+1,j} - u_2|_{m,j}\right), \qquad \bar{x} = x_m \in \Omega_1, \end{aligned} \tag{3}$$

$$\begin{aligned} \alpha u_2|_{m,j} &+ (1-\alpha)\frac{1}{\Delta x}\left(u_2|_{m,j} - u_2|_{m-1,j}\right) = \\ \alpha u_1|_{m,j} &+ (1-\alpha)\frac{1}{\Delta x}\left(u_1|_{m,j} - u_1|_{m-1,j}\right), \qquad \bar{x} = x_m \in \Omega_1. \end{aligned} \tag{4}$$

In equation (3) u_2 is considered as given (known), $u_1|_{m,j}$ is an unknown, and $u_1|_{m+1,j}$ is a fictitious value for which this equation provides substitution. The situation is similar for equation (4). Re-write these equations by introducing a convenient parameter $\gamma := (1 - \alpha(1+\Delta x))/(1-\alpha)$ as follows:

$$u_1|_{m+1,j} - u_2|_{m+1,j} = \gamma\left(u_1|_{m,j} - u_2|_{m,j}\right), \tag{5}$$

$$u_2|_{m-1,j} - u_1|_{m-1,j} = \gamma\left(u_2|_{m,j} - u_1|_{m,j}\right). \tag{6}$$

In equations (5) and (6) the parametrized discrete Robin boundary condition at the matrix level (below) amounts to a kind of error feedback where γ is the gain. If γ is first chosen, then α in (3) and (4) becomes $\alpha = (\gamma-1)/(\gamma-(1+\Delta x))$. We note the particular choice of $\gamma = 0$ ($\alpha = 1/(1+\Delta x)$), giving a variation on the SAM Dirichlet condition in which an extra column has been inserted to slide the entries over; we call this the Sliding Dirichlet Condition.

By substituting (3) and (4) in (2), a couple of first order systems of differential for u_1 and u_2 arise:

$$\frac{du_1}{dt} = B_1 u_1 + \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \Delta x^{-2}(u_{2,m+1} - \gamma u_{2,m}) \end{bmatrix}, \tag{7}$$

$$\frac{du_2}{dt} = B_2 u_2 + \begin{bmatrix} \Delta x^{-2}(u_{1,m-1} - \gamma u_{1,m}) \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \tag{8}$$

where $B_1 \in \mathbf{R}^{m\times m}$, and $B_2 \in \mathbf{R}^{M-m\times M-m}$.

The i-th row of B_1 has form $\Delta x^{-2}[0,\cdots,0,1,-2,1,0,\cdots,0]$, $i = 1,\cdots,m-1$, while the m-th row of B_1 is $\Delta x^{-2}[0,\cdots,1,-2+\gamma]$. For B_2 we retain the numbering from the spatial grid, so the i-th row is $\Delta x^{-2}[0,\cdots,0,1,-2,1,0,\cdots], i = m+1,\cdots,M-1$, and the m-th row (the first row in actuality) is $\Delta x^{-2}[-2+\gamma,1,0,...,0]$. Then we assemble the matrices B_1 and B_2 into a block matrix of coefficients A to represent the global system of ordinary differential equations. There are now M unknowns due to the extra unknown $u_{2,m}$ (formerly, there were $M-1$ unknowns). Setting $u = [u_1, u_2]^T$, we arrive at:

$$\frac{du}{dt} = Bu. \tag{9}$$

Here, $B \in \mathbf{R}^{M\times M}$ is given by:

$$B = \Delta x^{-2}\begin{bmatrix} -2 & 1 & & & & & & & \\ & \ddots & \ddots & \ddots & & & & & \\ & 1 & -2 & 1 & & & & & \\ & & 1 & -2+\gamma & -\gamma & 1 & & & \\ & & 1 & -\gamma & -2+\gamma & 1 & & & \\ & & & & 1 & -2 & 1 & & \\ & & & & & \ddots & \ddots & \ddots & \\ & & & & & & 1 & -2 \end{bmatrix}. \tag{10}$$

The exact solution of the semi-discrete system (9) satisfies the following two-term recurrence relation, [Smi85, Var62]:

$$u(t+\Delta t) = e^{\Delta tB}u(t). \tag{11}$$

Several algorithms for the numerical solution of (11) can be generated through an approximation to the exponential $e^{\Delta tB}$; in particular, we shall use rational functions via implicit Padé approximations, focussing on (1,0) and (1,1) Padé schemes, *cf.* [Smi85].

The (1,0) and (1,1) Padé approximations are given (respectively) by

$$e^{\Delta tB} = (I-\Delta tB)^{-1} + O(\Delta t) \;\&\; = (I-0.5\Delta tB)^{-1}(I+0.5\Delta tB) + O(\Delta t^2). \tag{12}$$

For each scheme of (12) the recurrence relation for $u(t+\Delta t)$ in (11) gives the following linear system to solve:

$$(I-\sigma B)u(t+\Delta t) = R_m(\Delta tB)u(t), \tag{13}$$

where (for (1,0) and (1,1), respectively)

$$R_1(\Delta tB) = I,\ \sigma = \Delta t,\ \&\ R_2(\Delta tB) = I+\sigma B, \quad \sigma = 0.5\Delta t.$$

Let $A = I - \sigma B$ and $\tau = \sigma\Delta x^{-2}$. Then A is illustrated as follows:

$$A = I - \sigma B = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, \tag{14}$$

where

$$A_{11} = \begin{bmatrix} 1+2\tau & -\tau & 0 & \cdots & 0 \\ -\tau & 1+2\tau & -\tau & & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & & -\tau & 1+2\tau & -\tau \\ 0 & \cdots & 0 & -\tau & 1+(2-\gamma)\tau \end{bmatrix},$$

$$A_{12} = \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 \\ \vdots & & & & \vdots \\ 0 & 0 & \cdots & 0 & 0 \\ \gamma\tau & \tau & 0 & 0 & 0 \end{bmatrix},$$

$$A_{21} = \begin{bmatrix} 0 & 0 & \cdots & \tau & \gamma\tau \\ 0 & 0 & \cdots & 0 & 0 \\ \vdots & & & & \vdots \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$A_{22} = \begin{bmatrix} 1+(2-\gamma)\tau & -\tau & 0 & \cdots & 0 \\ -\tau & 1+2\tau & -\tau & & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & & -\tau & 1+2\tau & -\tau \\ 0 & \cdots & 0 & -\tau & 1+2\tau \end{bmatrix}.$$

Due to the implicit nature of the finite difference scheme, we desire to avoid undue restriction on the size of Δt. Thus, we assume only that $\Delta t/\Delta x \leq 1$, so that τ is at most $O(\Delta x^{-1})$ and is expected to be of order Δx^{-1}. We desire to choose α (or γ) to assure that A is an M-matrix or an H-matrix. It is easy to check that one must have $\gamma \leq 1$ for $\alpha \in [0,1]$; if $0 < \gamma \leq 1$ and $2\gamma\tau \leq 1$, then A is an H-matrix; and if $\gamma \leq 0$ then A is an M-matrix. Since τ depends on the mesh size, we want to reject the condition $2\gamma\tau \leq 1$. Therefore, we are now concentrating on the situation where $\gamma \leq 0$, in which case A is an M-matrix. This is easy to check upon consulting [Var62, p. 85].

The Two-Stage Multi-splitting Algorithm

The main advantage in the proposed boundary condition (3, 4) at interfaces is that it leads to a partitioning of the global matrix A in (14) into particularly beneficial submatrix blocks, which are amenable to solution of multi-splitting iterative type. The parallel multi-splitting method of O'leary and White [OW85] to solve a linear system $Au = b$ is defined by considering multi-splittings from the decomposition $A = M_k - N_k, k = 1, \cdots, K$, such that each M_k is invertible; we can form an iterative method, as follows:

$$u^{i+1} = M_k^{-1} N_k u^i + M_k^{-1} b. \tag{15}$$

The multi-splitting iterates in (15) can be computed concurrently by introducing the non-negative diagonal matrices E_k, $\sum_{k=1}^{K} E_k = I$, and using them as follows:

$$u_{i+1} = Hu_i + c, \tag{16}$$

where $H = \sum E_k M_k^{-1} N_k$ and $c = \sum E_k M_k^{-1} b$.

For the linear system of concern here the first stage of splitting is according to the subdomains Ω_k (*cp.* [FS97]):

$$M_i = \begin{bmatrix} D_{11} & & & & \\ & \ddots & & & \\ & & Aii & & \\ & & & \ddots & \\ & & & & D_{KK} \end{bmatrix}, \tag{17}$$

where the matrices D_{kk} are the diagonals of A_{kk}, respectively, or any dummy diagonal matrices to make $M_k, k = 1, \cdots, K$ invertible. We take $N_k = M_k - A$.

To calculate u^{n+1} from u^n (where $u = [u_1, u_2, \cdots, u_K]^T$), use:

$$\begin{aligned} M_k w_k^{i+1} &= N_k u^i + b, \quad k = 1, \cdots, K, \\ u^{n+1} &= \sum_{k=1}^{K} E_k w_k^{i+1}. \end{aligned} \tag{18}$$

This is rewritten as a single splitting with M =diag (A_{ii}) .

With the splitting above one could implement a nice parallel algorithm. There are a wide variety of iterative schemes to consider. As in [FS94], it is useful to examine the possibility of introducing a second state of multi-splitting methods, this time splitting the matrices M_k according to their convenient algebraic struture [OW85].

The second stage of splitting will be considered for the matrices M_k, $k = 1, \cdots, K$. For this article, we deal with three rather standard approaches for inner iteration: the Jacobi, Gauss-Seidel, and SOR. Each subdomain block has been constructed to be of form only slightly different from a standard type matrix, and it is natural to test these schemes to gain insight into the proposed boundary treatment.

For brevity, we shall only describe the procedure, assuming the standard version of each algorithm is well known. The outer iteration (also called the first stage) consists of splitting the matrix A as diagrammed earlier. Then we split the submatrices A_{ij} to the right hand side whenever $i \neq j$ and on processor i we split all other A_{jj} submatrices to the right, leaving only their diagonals.

The Parallel Jacobi (or PJacobi) version of the inner iteration (or second stage) is to iterate s times on processor i for unknowns corresponding to the i^{th} subdomain, splitting the upper and lower diagonal parts to the right hand side as well.

The Parallel Gauss-Seidel (or PGauss-Seidel) inner iteration is to iterate s times on processor i for unknowns corresponding to the i^{th} subdomain, splitting the upper diagonal parts within that block to the right hand side.

Parallel Successive Over-Relaxation (or PSOR) does s iterations for i^{th} subdomain unknowns of standard SOR (using ω as the parameter).

Numerical Experiments

In this section we demonstrate the performance of the algorithms described earlier and give empirical support to claim of convergence.

Let's work with the following prototype parabolic (heat) equation:

$$u_t = u_{xx} + x(1-x)\pi \sin(\pi t) - 2\cos(\pi t), \quad x \in (0,1), t > 0$$

$$u(x,0) = x(x-1), \quad x \in (0,1),$$

with boundary conditions $u(0,t) = u(1,t) = 0$. The exact solution is $x(x-1)\cos(\pi t)$. We will compute approximations to the solution at target time $t_* = 1.0$. All iterations have a halting signal of $||u^{n+1} - u^n|| \leq 0.5\Delta t^2 \Delta x ||u^n||$, which means smaller step sizes require more iterations.

The first numerical experiment to report, see Figure 1, is with inner splittings of Jacobi type, different values of K (the number of subdomains), two particular choices of γ, and $s = 1$. The computations were done in FORTRAN 90 on a Penitum III (not in parallel). The case $\gamma = 0.0$ is called the Sliding Dirichlet Condition because it looks like a Dirichlet condition, shifted over by one column. It arises from $\alpha = 1/(1 + \Delta x)$ in the discrete parametrized Robin condition, so it is not the ordinary Schwarz Alternating Method Dirichlet condition. It is observed in Figure 1 that the Sliding Dirichlet Condition allows the ordinary Jacobi algorithm to be parallelized in an efficient manner, in that the spectral radius appears to be nearly identical over any number of domain splittings (the numbers shown are rounded). Thus, our Parallel Jacobi method (PJacobi) would run a factor of K times faster on a parallel machine with K processors. (Here, communication is neglected because it is relatively insignificant.)

Moreover, there are better choices of γ. As shown in Figure 1, $\gamma = -1.0$ speeds up the PJacobi multi-splitting scheme quite significantly. (The value $\gamma = -1.0$ is not the absolute optimal from experiment, but is nearly so, and this is chosen for convenience.) This corresponds to a discrete parametrized Robin boundary condition with $\alpha = 2/(2 + \Delta x)$.

Using old values from neighboring domains and a single inner iteration ($s = 1$), we want to have a look at the Gauss-Seidel scheme on each subdomain. The Parallel Gauss-Seidel scheme performs best when γ is zero (or slightly positive); see Figure 2.

References

[CM94]Tony F. Chan and Tarek P. Mathew. Domain decomposition algorithms. In *Acta Numerica 1994*, pages 61–143. Cambridge University Press, 1994.

[FS94]A. Frommer and D.B. Szyld. Asynchronous two stage iterative methods. *Numer. Math.*, 69:141–153, 1994.

[FS97]A. Frommer and H. Schwandt. A unified representation and theory of algebraic additive schwarz and multisplitting methods. *SIAM J. Matrix Anal. Appl.*, 18:893–912, 1997.

[GS98]M.J. Gander and A.M. Stuart. Space time continuous analysis of waveform relaxation for the heat equation. *SIAM J.*, 19:2014–2031, 1998.

[HT96]H.San and W.P. Tang. An overdetermined schwarz alternating method. *SIAM J. Sci. Comput.*, 7:884–905, 1996.

K	γ	Avg. No. Iter. Per Step $\Delta x = 0.003125$	Avg. No. Iter. Per Step $\Delta x = 0.0015625$
1	N/A	4407	9856
	N/A		
2	0.0	4407	9856
	-1.0	4058	9436
4	0.0	4407	9856
	-1.0	3097	8070
8	0.0	4407	9856
	-1.0	1632	5164
16	0.0	4407	9856
	-1.0	1468	2935
32	0.0	4407	9856
	-1.0	1486	2953

Figure 1: Parallel Jacobi showing the effect of γ and K ($s = 1$).

K	Avg. No. Iter. Per Step $\Delta x = 0.003125$	Avg. No. Iter. Per Step $\Delta x = 0.0015625$
1	941	1895
2	948	1905
4	948	1905
8	950	1906
16	957	1912
32	971	1927

Figure 2: Parallel Gauss-Seidel showing the effect of K with $\gamma = 0.0$ and $s = 1$.

[OW85]D. P. O'Leary and R.E. White. Multi-splittings of matrices and parallel solution of linear systems. *SIAM J. Alg. Disc. Meth.*, 6:630–640, 1985.

[Smi85]G.D. Smith. *Numerical Solution of Partial Differential Equations: Finite Difference Methods*. Oxford Univ. Press, second edition, 1985.

[Tan92]W.P. Tang. Generalized schwarz splittings. *SIAM J. Sci. Stat. Comput.*, 13:573–595, 1992.

[Var62]R. Varga. *Matrix Iterative Analysis*. Prentice Hall, first edition, 1962.

K	Avg. No. Iter. Per Step (ω) $\Delta x = 0.003125$	Avg. No. Iter. Per Step (ω) $\Delta x = 0.0015625$
1	108 (1.90)	167 (1.92)
2	118 (1.90)	214 (1.90)
4	142 (1.87)	217 (1.90)
8	170 (1.84)	243 (1.89)
16	231 (1.78)	321 (1.85)
32	365 (1.64)	510 (1.75)

Figure 3: Parallel SOR (at near optimal ω) showing the effect of K with $\gamma = 0.0$ and $s = 1$.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau,

11. Domain Decomposition and Splitting Methods for Mortar Mixed Approximations to Parabolic Problems

I. Faille[1], S. Gaiffe[1], R. Glowinski[2], P. Lemonnier[1], R. Masson[1]

Introduction

Mixed Finite Element (MFE) methods have become popular for the numerical simulation of single phase flow in porous media due to their good approximation of the flux variable and their local and global mass conservation properties. In many situations such as flow around wells or through conductive faults, the complexity of the geometry, the heterogeneities of the media, or the singularities of the data may require the use of flexible meshes including hybrid meshes or local refinements to capture the spatial behavior of the solution. In that case, non-overlapping domain decomposition techniques with Mortar elements at the interfaces of the decomposition have proven to be efficient since they enable to define the grids independently in the subdomains regions (see [GW88], [Yot96]), [ACWY96]).

On the other hand, the transient behavior of the solution may also warrant the use of different time steps in the different subdomains.

The idea of the domain decomposition method introduced in this paper is to combine Mortar Mixed Finite Element methods for the space discretization with operator splitting techniques for the time discretization in order to obtain (1) a fully parallel algorithm and (2) the possibility to use flexible meshes and local time steppings in the subdomains.

We consider a domain $\Omega \subset \mathbb{R}^d$ of boundary Γ and the parabolic equation

$$\left\{ \begin{array}{l} \partial_t p + \nabla \cdot u = f, \; u = -K\nabla p \text{ in } \Omega, \\ p = g \text{ on } \Gamma, \; p|_{t=0} = p_0, \end{array} \right. \tag{1}$$

where K is a symmetric matrix, positive definite uniformly in $\overline{\Omega}$.

Most domain decomposition algorithms for such parabolic problems involve, at each time step, the solution of an elliptic problem, using classical domain decomposition iterative algorithms for elliptic equations. The present domain decomposition approach takes advantage of the parabolic structure of the problem to obtain, through operator splitting, a non-iterative method in the sense that the subdomains problems are solved only once at each time step. Other related non-iterative domain decomposition and splitting methods for parabolic problems can be found in [MPW98], [CL96], and [Dry91], and the references therein. The main originality of our method is to allow

[1] Division Informatique Scientifique et Mathématiques Appliquées, Institut Français du Pétrole, 92852 Rueil Malmaison Cedex, France, e–mail:isabelle.faille@ifp.fr, stephanie.gaiffe@ifp.fr, patrick.lemonnier@ifp.fr, roland.masson@ifp.fr

[2] Dept. of Mathematics, University of Houston, 4800 Calhoun Rd, Houston, TX 77204-3476, USA, e–mail: roland@math.uh.edu

by construction non-matching grids at the interfaces of the domain decomposition.

Notation: for two positive functions $A(v)$ and $B(v)$, the notation $A \lesssim B$ means that there exists a constant C, independent of the various parameters, such that for all v one has $A(v) \leq CB(v)$.

Mixed Finite Element Domain Decomposition Method

Let us consider a domain decomposition of Ω into N non-overlapping subdomains $\Omega_i, i = 1, \ldots, N$ such that $\Omega_i \cap \Omega_j = \emptyset$ for all $i \neq j$, and $\overline{\Omega} = \bigcup_{i=1}^N \overline{\Omega}_i$. We set $\Gamma_i := \partial\Omega_i / \Gamma$. For $i, j \in I := \{i, j \text{ s.t. } i \neq j \text{ and } \text{mes}_{d-1} \partial\Omega_i \cap \partial\Omega_j \neq 0\}$, we denote by $\Gamma_{i,j} := \partial\Omega_i \cap \partial\Omega_j$ the interface between two subdomains, and by $\gamma := \bigcup_{i,j \in I} \Gamma_{i,j}$, the skeleton of the domain decomposition.

On each subdomain Ω_i, we introduce the function spaces $M_i := L^2(\Omega_i)$ and $V_i = H(\Omega_i; \text{div}) := \{v \in L^2(\Omega_i)^d \text{ s.t. } \nabla \cdot v \in M_i\}$, endowed with their usual norms denoted by $\|q_i\|_{0,i}$ and $\|v_i\|_{V_i} := \left(\|v_i\|_{0,i}^2 + \|\nabla \cdot v_i\|_{0,i}^2 \right)^{1/2}$. On the domain Ω, we define the product spaces $M := \bigoplus_{i=1}^N M_i$ and $V := \bigoplus_{i=1}^N V_i$ endowed with their Hilbertian product norms $\|q\|_0$ and $\|v\|_V$.

In the non-overlapping domain decomposition framework, the smoothness assumptions on the solution will be as usual measured in the broken norms $\| \cdot \|_{\mathcal{H}^r(\Omega)}$ related to the product spaces $\mathcal{H}^r(\Omega) := \bigoplus_{i=1}^N H^r(\Omega_i),\ r \geq 0$. On the skeleton γ, we define the norm $\|\mu\|_{\frac{1}{2},\gamma} := \sup_{v \in V} \frac{\sum_{i=1}^N \int_{\Gamma_i} (v \cdot n_i) \mu d\gamma}{\|v\|_V}$, and we shall denote by $H^{\frac{1}{2}}(\gamma)$, the subspace of $L^2(\gamma)$ of functions μ such that $\|\mu\|_{\frac{1}{2},\gamma} < \infty$.

We consider, on the domain decomposition $(\Omega_i)_{i=1,\ldots,N}$, a Mortar Mixed Finite Element (MMFE) discretization of (1), introduced in [GW88] for matching grids, and extended in [Yot96], [ACWY96] to the case of non-matching grids at the interfaces between the subdomains Ω_i. In that case, a so called Mortar space $\Lambda_h \subset L^2(\gamma)$ is introduced on the skeleton γ. Then, equation (1) is discretized on each subdomain by a Mixed Finite Element Method, and the matching at the interfaces is written in the weak sense through the continuity of the orthogonal projection on Λ_h of the normal fluxes defined on each sides of $\Gamma_{i,j}$.

Let $\mathcal{T}_{i,h}$ be a quasi-uniform mesh of Ω_i. We consider, on these grids, MFE approximation spaces $V_{i,h} \subset V_i$, $M_{i,h} \subset M_i$ of order $k+1$, that can be either the RT_k or BDF_k or BDFM_k MFE discretizations (see [RT91] or [BF91]). In addition we shall assume in the sequel that $\nabla \cdot V_{i,h} = M_{i,h}$.

On the domain Ω, we define the product spaces $M_h := \bigoplus_{i=1}^N M_{i,h} \subset M$ and $V_h := \bigoplus_{i=1}^N V_{i,h} \subset V$. The dual space of V_h (resp. M_h) is denoted by V_h' (resp. M_h') endowed with the dual norm $\| \cdot \|_{V_h'}$ (resp. $\| \cdot \|_{M_h'}$). We shall denote by $\langle \cdot, \cdot \rangle$ the duality pairing.

We reproduce the choice of the Mortar space Λ_h as described in [Yot96]. Let $\mathcal{T}_{i,j,h}$,

$i, j \in I$ be a quasi-uniform mesh of $\Gamma_{i,j}$ and $\Lambda_{i,j.h}$ a finite element space on $\mathcal{T}_{i,j,h}$, either continuous or discontinuous, and of order $k+2$. The Mortar space on the skeleton γ is the product space $\Lambda_h := \bigoplus_{i,j\in I} \Lambda_{i,j,h} \subset L^2(\gamma)$.

In order to write the MMFE variational formulation of (1), we define the operators $S_h, A_h : V_h \to V_h'$, $B_h^t : \Lambda ambda_h \to V_h'$, $\mathrm{div}_h : V_h \to M_h'$, $T_h^t : H^{1/2}(\Gamma) \to V_h'$ such that for all $v_h = (v_{i,h})_{i=1,\dots,N}, w_h = (w_{i,h})_{i=1,\dots,N} \in V_h$, $q_h = (q_{i,h})_{i=1,\dots,N} \in M_h$, $\mu_h \in \Lambda ambda_h$, $\varphi \in H^{1/2}(\Gamma)$:

$$\begin{array}{l} \langle S_h v_h, w_h\rangle := \sum_{i=1}^N \int_{\Omega_i} K^{-1} v_{i,h} \cdot w_{i,h} dx, \\ \langle A_h v_h, w_h\rangle := \sum_{i=1}^N \int_{\Omega_i} (\nabla \cdot v_{i,h})(\nabla \cdot w_{i,h}) dx, \\ \langle \mathrm{div}_h v_h, q_h\rangle := \sum_{i=1}^N \int_{\Omega_i} (\nabla \cdot v_{i,h}) q_{i,h} dx, \\ \langle B_h^t \mu_h, v_h\rangle := \sum_{i=1}^N \int_{\Gamma_i} \mu_h (v_{i,h} \cdot n_i) d\gamma,\ \langle T_h^t \varphi, v_h\rangle := \int_\Gamma \varphi(v_h \cdot n) d\sigma. \end{array} \tag{2}$$

Then, the MMFE spatial discretization of (1) looks for $(p_h, u_h, p_{\gamma,h}) \in M_h \times V_h \times \Lambda_h$ such that

$$\left\{ \begin{array}{l} \partial_t p_h + \mathrm{div}_h u_h = i_{M_h}^t f, \\ S_h u_h = \mathrm{div}_h^t p_h - B_h^t p_{\gamma,h} - T_h^t g, \\ B_h u_h = 0, \\ p_h|_{t=0} = p_{0,h}. \end{array} \right. \tag{3}$$

The stationnary MMFE approximation (3) is analysed in [Yot96] and [ACWY96]. In order to obtain a well posed problem, one has to assume that the Mortar space Λ_h verifies a compatibility condition with the normal traces on γ of V_h. In particular this condition ensures that the operator B_h^t is into as well as the property

$$\{q_h, \text{ t. q. } \langle \mathrm{div}_h v_h, q_h\rangle = 0, \text{ for all } v_h \in W_h := \mathrm{Ker} B_h\} = \{0\},$$

which all together guarantees existence and uniqueness of the solution. We refer to [Yot96] for the proof, under this assumption, of optimal error estimates for the solutions $u_h, p_h, p_{\gamma,h}$ of the stationnary problem.

An equivalent flux formulation

As a preliminary step towards the time discretization by an operator splitting technique, it is useful to introduce an equivalent flux formulation of (3) obtained by elimination of the discrete pressure unknown in (3). This formulation will also be crucial to analyse the stability and the error estimates of our method.

Proposition 1 *Let us define $\lambda_h := \partial_t p_{\gamma,h}$ and $g_0 := g|_{t=0}$. Then problem (3) is equivalent to the following flux formulation:*

$$\left\{ \begin{array}{l} S_h \partial_t u_h + A_h u_h + B_h^t \lambda_h + T_h^t \partial_t g = \mathrm{div}_h^t f, \\ B_h u_h = 0, \\ u_h|_{t=0} = u_h^0, \end{array} \right. \tag{4}$$

given the initialization

$$\left\{ \begin{array}{l} S_h u_h^0 = \mathrm{div}_h^t p_{0,h} - B_h^t p_{\gamma,h}^0 - T_h^t g_0, \\ B_h u_h^0 = 0, \end{array} \right. \tag{5}$$

and the pressure equation

$$\left\{\begin{array}{l} \partial_t p_h + \mathrm{div}_h u_h = i^t_{M_h} f, \\ \partial_t p_{\gamma,h} = \lambda_h, \\ p_h|_{t=0} = p_{0,h},\ p_{\gamma,h}|_{t=0} = p^0_{\gamma,h}. \end{array}\right. \tag{6}$$

proof: the proof relies on elementary algebra using the assumption on the MFE spaces that $\nabla \cdot V_h = M_h$, and assuming enough regularity on the trace g.

Time discretization by operator splitting

The flux formulation (4) is a mixed problem formally equivalent to the Stokes equation. The idea of the time discretization by operator splitting is then to apply to the flux formulation (4) a projection scheme introduced by Chorin [Cho68] and analysed in [Ran92] in the framework of the Navier-Stokes equations.

In the framework of the MMFE method, the projection scheme splits the system (4) into two successive steps: (i) advance in time with a fixed λ_h given by the previous time step, (ii) orthogonal projection (with respect to the scalar product $\langle S_h \cdot, \cdot\rangle$) of the flux onto W_h, and updating of λ_h. The initialization of the flux is still given by equation (5). This scheme requires to be given an approximation $\lambda^0_h \in \Lambda_h$ of $\lambda|_{t=0}$. At first order accuracy in time, we shall see that it is sufficient to set $\lambda^0_h = 0$. However, in order to expect second order accuracy, a first order accurate approximation of λ^0_h has to be obtained by one time step calculation of the fully coupled system.

$$\begin{array}{ll} (i) & S_h \frac{\tilde{u}^{n+1}_h - u^n_h}{\Delta t} + A_h \tilde{u}^{n+1}_h + B^t_h \lambda^n_h + T^t_h \frac{g^{n+1} - g^n}{\Delta t} = \mathrm{div}^t_h f^{n+1}, \\ (ii) & \left\{\begin{array}{l} S_h \frac{u^{n+1}_h - \tilde{u}^{n+1}_h}{\Delta t} + B^t_h(\lambda^{n+1}_h - \lambda^n_h) = 0, \\ B_h u^{n+1}_h = 0, \end{array}\right. \end{array} \tag{7}$$

The pressures p^n_h and $p^n_{\gamma,h}$ are obtained by discrete integration in time.

$$\left\{\begin{array}{l} \frac{p^{n+1}_h - p^n_h}{\Delta t} + \mathrm{div}_h \tilde{u}^{n+1}_h = i^t_{M_h} f^{n+1},\ p^0_h = p_{0,h}, \\ \frac{p^{n+1}_{\gamma,h} - p^n_{\gamma,h}}{\Delta t} = \lambda^{n+1}_h,\ p^0_{\gamma,h} \text{ given by (5).} \end{array}\right. \tag{8}$$

As for the semi-discrete formulation, the space-time discretization (7)-(8) admits an equivalent mixed pressure-flux formulation which, from elementary algebra, writes:

$$\begin{array}{ll} (i) & \left\{\begin{array}{l} \frac{p^{n+1}_h - p^n_h}{\Delta t} + \mathrm{div}_h \tilde{u}^{n+1}_h = i^t_{M_h} f^{n+1}, \\ S_h \tilde{u}^{n+1}_h = \mathrm{div}^t_h p^{n+1}_h - B^t_h(2p^n_{\gamma,h} - p^{n-1}_{\gamma,h}) - T^t_h g^{n+1}, \end{array}\right. \\ (ii) & \left\{\begin{array}{l} S_h u^{n+1}_h = \mathrm{div}^t_h p^{n+1}_h - B^t_h p^{n+1}_{\gamma,h} - T^t_h g^{n+1}, \\ B_h u^{n+1}_h = 0, \end{array}\right. \end{array} \tag{9}$$

with $p^0_h := p_{0,h}$ and $p^{-1}_{\gamma,h} := p^0_{\gamma,h} - \Delta t \lambda^0_h$. We note that (9) corresponds, at step (i), to a second order linear extrapolation in time of the interface pressure $p^{n+1}_{\gamma,h} \simeq 2p^n_{\gamma,h} - p^{n-1}_{\gamma,h}$.

The main advantage of the projection scheme is that the prediction step (i) can be solved in a fully parallel way on each subdomain independently, while the projection step (ii) reduces to inverse the interface problem related to the operator $B_h S^{-1}_h B^t_h$.

Let us restrict ourselves to the assumption that only RT_0 mixed finite elements are used in the neighborhood of the skeleton γ. Then, a mass condensation of the matrix representing the operator S_h in the nodal basis can be locally performed, preserving the order of approximation of the discretization. It results that the interface operator matrix in the nodal basis of Λ_h is diagonal and can be readily inverted in $\mathcal{O}(N_{\Lambda_h})$ operations where N_{Λ_h} is the dimension of Λ_h.

More generally, the interface problem can be efficiently solved by a conjugate gradient iterative algorithm preconditioned by the approximate interface matrix obtained by mass condensation of S_h in the neighborhood of γ.

Stability analysis

Let $Z_h := B_h S_h^{-1} B_h^t$ denote the interface operator related to the projection step (ii). For any $\mu \in L^2(\gamma)$, we set $\|\mu\|_{Z_h} := \langle Z_h \mu, \mu \rangle^{\frac{1}{2}}$, which defines a semi-norm on $L^2(\gamma)$ and a norm on Λ_h. On the other hand, we define $\|B_h^t \mu\|_{V_h'} := \sup_{v_h \in V_h} \frac{\sum_{i=1}^N \int_{\Gamma_i} (v_h \cdot n_i) \mu d\gamma}{\|v_h\|_V}$, semi-norm on $L^2(\gamma)$ (and norm on Λ_h).

The stability analysis of the incremental scheme is done in its equivalent flux formulation (7)-(8) in order to avoid to deal with the three steps equations (9). It is then formally similar to the analysis performed for Navier Stokes equations (see [She92], [GQ98]) with necessary adaptations to the framework of domain decomposition and MMFE.

Theorem 1 *Let $t_n := n\Delta t$, and assume $\partial_t g \in L^2(0, t_m; H^{\frac{1}{2}}(\Gamma))$, $\sum_{n=0}^{m-1} \Delta t \|f^{n+1}\|_0^2 \lesssim 1$, then the incremental projection scheme (7)-(8) or (9) is unconditionally stable in the sense that for all Δt*

$$\left\{ \begin{array}{l} \|u_h^m\|_0^2 + \Delta t^2 \|\lambda_h^m\|_{Z_h}^2 + \sum_{n=0}^{m-1} \Delta t \|\nabla \cdot \tilde{u}_h^{n+1}\|_0^2 \\ \qquad \lesssim \ \|u_h^0\|_0^2 + \Delta t^2 \|\lambda_h^0\|_{Z_h}^2 + \Delta t \sum_{n=0}^{m-1} \|f^{n+1}\|_0^2 + \int_0^{t_m} \|\partial_t g(s)\|_{H^{\frac{1}{2}}(\Gamma)}^2 ds, \\ \|p_h^m\|_0^2 \ \lesssim \ \|p_{0,h}\|_0^2 + \sum_{n=0}^{m-1} \Delta t \|\nabla \cdot \tilde{u}_h^{n+1}\|_0^2 + \Delta t \sum_{n=0}^{m-1} \|f^{n+1}\|_0^2, \\ \|B_h^t p_{\gamma,h}^m\|_{V_h'} \ \lesssim \ \|u_h^m\|_0 + \|p_h^m\|_0 + \|g^m\|_{H^{\frac{1}{2}}(\Gamma)}, \end{array} \right. \tag{10}$$

with constants independent of h, Δt, N and depending only on t_m and K.

Error estimates

We denote by $(u, p) \in C^0(0, t_m; H(\Omega; \mathrm{div})) \times C^0(0, t_m; M)$ the weak solution of (1) on the interval $[0, t_m]$. We shall assume that the pressure p and its derivative $\partial_t p$ are globally in $H^1(\Omega)$ in order to define the interface pressure $p_\gamma := p|_\gamma$ and its derivative $\lambda := \partial_t p|_\gamma = \partial_t p_\gamma$ in $H^{1/2}(\gamma)$. We set $t_n = n\Delta t$ and $u^n := u(t_n)$, $p^n := p(t_n)$, $\lambda^n := \lambda(t_n)$, $p_\gamma^n := p_\gamma(t_n)$.

The dependence of the semi-norm $\|\cdot\|_{Z_h}$ on the mesh size h, as given by the estimate $\|\mu\|_{Z_h} \lesssim h^{-\frac{1}{2}} \|\mu\|_{L^2(\gamma)}$ $\forall \mu \in L^2(\gamma)$, deteriorates the convergence of the method. We can prove the following theorem.

Theorem 2 *Let $(u, p) \in C^0(0, t_m; H(\Omega; \mathrm{div})) \times C^0(0, t_m; M)$, be the weak solution of (1) such that $p \in C^1(0, t_m; H^1(\Omega))$. Pour $1 \leq r \leq k+1$ et $u \in H^1(0, t_m; \mathcal{H}^r(\Omega)^d)$,*

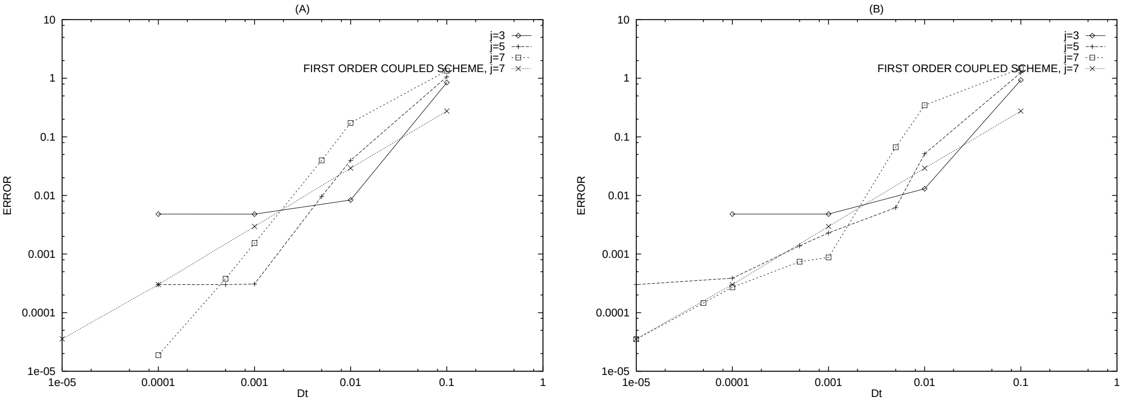


Figure 1: Convergence history of the pressure error in $l^\infty(L^2(\Omega))$ norm: (A) first order incremental and coupled schemes, (B) second order incremental and first order coupled schemes.

$\partial_{t^2} u \in L^2(0,t_m;V')$, $\partial_t \lambda \in L^2(0,t_m;L^2(\gamma))$, $\partial_{t^2} g \in L^2(0,t_m;H^{\frac{1}{2}}(\Gamma))$, $\partial_{t^2} p \in L^2(0,t_m;$ $L^2(\Omega))$, $p \in W^{1,\infty}(0,t_m;\mathcal{H}^{r+1}(\Omega))$, $\sum_{n=0}^{m-1} \Delta t \|\nabla \cdot u^{n+1}\|^2_{\mathcal{H}^r(\Omega)} \lesssim 1$, *we have*

$$\begin{aligned} &\|u^m - u_h^m\|_0 + \|p^m - p_h^m\|_0 + \|B_h^t(p_\gamma^m - p_{\gamma,h}^m)\|_{V_h'} \\ &\qquad + \Big(\textstyle\sum_{n=0}^{m-1} \Delta t \|\nabla \cdot (u^{n+1} - \tilde{u}_h^{n+1})\|_0^2\Big)^{\frac{1}{2}} \lesssim \Delta t(1 + h^{-\frac{1}{2}}) + h^r, \end{aligned} \tag{11}$$

with a constant depending only on t_m, K. To obtain these estimates, it suffices to choose for $p_{0,h}$ the orthogonal projection of p_0 onto M_h and $\lambda_h^0 = 0$.

Numerical example

Let us consider in dimension $d = 1$, the interval $\Omega =]-1,1[$ splitted into two subdomains $\Omega_1 =]-1,0[$ and $\Omega_2 =]0,1[$, and equation (1) for $g = 0$ and $K = 1$ with exact solution $p(x,t) = \cos\frac{\pi x}{2}(\cos 6t + 2)$. This problem is discretized on a uniform mesh of size $h = 2^{-j}, j \in \mathbb{N}$ using RT_0 MFE with mass condensation. Figure 1 reports the convergence history of the error $p_h^n - p^n$ in $l^\infty(L^2(\Omega))$ norm for 3 different time discretizations: (a) the incremental scheme (9), (b) the incremental scheme with second order Crank-Nicholson time discretization in the subdomains at step (i), (d) the first order Euler backward fully coupled discretization.

From the numerical results displayed Figures 1, we see that the error behaves like $\min(\frac{\Delta t}{h^{1/2}}, \frac{\Delta t^2}{h})$ for the incremental projection scheme (a), like $\min(\frac{\Delta t}{h^{1/2}}, \frac{\Delta t^2}{h})$ $+\Delta t$ for the incremental projection scheme (b). The same results can be observed for the flux u and the interface pressure p_γ.

These results suggest that the error is the sum of the error produced by the coupled scheme and the splitting error (between the coupled scheme and the projection scheme) of order $\min(\frac{\Delta t}{h^{1/2}}, \frac{\Delta t^2}{h})$.

Conclusion

The method introduced in this paper combines Mortar Mixed Finite Element domain decomposition spatial discretization with projection schemes, in order to obtain a fully parallel algorithm for parabolic equations. In addition this method enables the use of hybrid meshes and local time steppings.

Although the scheme is shown to be unconditionally stable, the convergence is obtained only if the condition $\Delta t \lesssim h^{\frac{1}{2}}$ is verified. This is the price to pay to decouple the interface problem from the computation of the subdomain solutions.

This strategy has proven to be efficient to solve single phase Darcy flows around 2D wells and faults with high physical heterogeneities and complex geometries, and we refer to [Gai00] where such numerical tests are reported.

References

[ACWY96]T. Arbogast, L.C. Cowsar, M.F. Wheeler, and I. Yotov. Mixed finite element methods on non-matching multiblock grids. *SIAM J. Numer. Anal.*, 1996. submitted.

[BF91]F. Brezzi and M. Fortin. *Mixed and Hybrid Finite Element Methods.* Springer-Verlag, New-York, 1991.

[Cho68]A.J. Chorin. Numerical solutions of Navier-Stokes equations. *Math. Comp.*, 22:49–73, 1968.

[CL96]H. Chen and R.D. Lazarov. Domain splitting algorithms for mixed finite element. *East-West J. Numer. Math.*, 4:121–135, 1996.

[Dry91]Maksymilian Dryja. Substructuring methods for parabolic problems. In Roland Glowinski, Yuri A. Kuznetsov, Gérard A. Meurant, Jacques Périaux, and Olof Widlund, editors, *Fourth International Symposium on Domain Decomposition Methods for Partial Differential Equations*, Philadelphia, PA, 1991. SIAM.

[Gai00]S. Gaiffe. *Maillages hybrides et décomposition de domaine pour la modélisation des réservoirs pétrolier.* PhD thesis, Université Pierre et Marie Curie - Paris VI, 2000. to appear.

[GQ98]J. L. Guermond and L. Quartapelle. On the approximation of the unsteady Navier-Stokes equations by finite element projection methods. *Numer. Math.*, 80:207–238, 1998.

[GW88]Roland Glowinski and Mary F. Wheeler. Domain decomposition and mixed finite element methods for elliptic problems. In Roland Glowinski, Gene H. Golub, Gérard A. Meurant, and Jacques Périaux, editors, *First International Symposium on Domain Decomposition Methods for Partial Differential Equations*, Philadelphia, PA, 1988. SIAM.

[MPW98]T.P. Mathew, P.L. Polyakov, and G. Russo J. Wang. Domain decomposition operator splittings for the. *SIAM J. Sci. Comput.*, pages 912–932, 1998.

[Ran92]R. Rannacher. *On Chorin's projection methods for Navier-Stokes equations.* Lecture Notes in Mathematics. Springer, Berlin, 1992.

[RT91]J.E. Roberts and J.M Thomas. *Handbook of Numerical Analysis, Mixed and hybrid methods*, volume II. Elsevier Science Publishers B.V., North-Holland, Amsterdam, 1991.

[She92]J. Shen. On error estimates of projection methods for Navier-Stokes equations: first order schemes. *SIAM J. Numer. Anal.*, 1:49–73, 1992.

[Yot96]I Yotov. *Mixed Finite Element Methods for Flow in Porous Media.* PhD thesis, TICAM, University of Texas at Austin, 1996.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau,

12. Mortar Projection in Overlapping Composite Mesh Difference Methods

Serge Goossens[1], Xiao-Chuan Cai[2]

Introduction

We study experimentally the effect of the mortar projection in an overlapping composite mesh difference method for two-dimensional elliptic problems. In [CDS99], an overlapping mortar element method was proposed. This method has several desirable properties. For example, the discretisation is consistent, the accuracy is of optimal order and the error is independent of the size of the overlap, as well as the ratio of the mesh sizes. However, a major disadvantage of the method is that it needs weights in the bilinear form. The artificially introduced piecewise constant weights make the scheme consistent, but at the same time make it impossible to use fast solvers for the subdomain problems. On the other hand, the composite mesh difference method (CMDM) [Sta77, CMS00, GC99] does not need any weights, and its accuracy is also of optimal order if used with higher order interface interpolations. For example, the 2D bicubic or modified 1D cubic interface interpolation [GC99] is needed if one uses P1 or Q1 finite elements for the interior of the subdomains. But if the computationally more efficient low order interpolation is used on the interfaces, it may lead to a local inconsistent discretisation, resulting in an error that depends on the size of the overlap. The goal of this paper is to take the mortar approach, drop the weights and compare its results to the non-mortar methods. Of course, in an ideal scheme, which is yet to be discovered, the accuracy should be of optimal order and the error be independent of the size of the overlap and the ratio of mesh sizes. In order to be able to use fast solvers for the subdomain problems, it is also desirable not to have weights in the discretisation on the overlapping parts of subdomains.

Overlapping Nonmatching Grids Mortar Element Method

In this section we briefly describe the overlapping nonmatching grid mortar method. A two-subdomain version was given in [CDS99] and a many subdomain version was given by Maday et al[3]. Let $\Omega = \Omega_1' \cup \Omega_2'$ be the union of two overlapping, polygonal

[1] Department of Computer Science, K. U. Leuven, Celestijnenlaan 200A, B3001 Leuven, Belgium (Serge.Goossens@cs.kuleuven.ac.be, http://www.cs.kuleuven.ac.be/∼serge/). Part of this work was carried out during the visit of the first author to the University of Colorado at Boulder. The financial support for this visit by the FWO-Vlaanderen is gratefully acknowledged. This research is financed by a specialisation scholarship of the Flemish Institute for the Promotion of Scientific and Technological Research in Industry (IWT).

[2] Department of Computer Science, University of Colorado at Boulder, Boulder, CO 80309-0430, USA (cai@cs.colorado.edu, http://www.cs.colorado.edu/∼cai/).

[3] Presentation at the 12th International Conference on Domain Decomposition Methods, October 25–29, 1999, Chiba University, Chiba, Japan

subdomains. On each Ω'_i $(i = 1, 2)$, we define a function space for P1 or Q1 finite elements on a uniform grid with mesh size h_i and denote this function space by V_{h_i}. We denote $h = \min_i\{h_i\}$. We define the interface by $\gamma_i = \partial\Omega'_i \setminus \partial\Omega$ and the trace space $V_{h_i}(\gamma_j)$ as the restriction of V_{h_i} on γ_j. The mortar projection π_1 maps the space $V_{h_2}(\gamma_1)$ into $V_{h_1}(\gamma_1)$:

$$\int_{\gamma_1} (\varphi - \pi_1\varphi)\psi \; ds = 0 \;\; \forall \psi \in \tilde{\mathcal{W}}_{h_1}(\gamma_1). \tag{1}$$

The interface test function space $\tilde{\mathcal{W}}_{h_1}(\gamma_1)$ denotes the space of continuous piecewise linear functions that are constants in the first and last intervals, see [BMP94, CDS99]. Similarly we can define π_2. This projection is used in the definition of the solution space

$$V_h = \{(u_1, u_2) | u_1 \in V_{h_1}, u_2 \in V_{h_2}, u_{1|\gamma_1} = \pi_1(u_{2|\gamma_1}), u_{2|\gamma_2} = \pi_2(u_{1|\gamma_2})\}. \tag{2}$$

With the space V_h the variational form can be defined as:

$$\text{Find } u = (u_1, u_2) \in V_h \text{ such that } a_h(u, v) = f_h(v) \;\; \forall v = (v_1, v_2) \in V_h, \tag{3}$$

where the weighted bilinear form is defined as

$$\begin{aligned} a_h(u, v) \;\; &= \;\; \int_{\Omega'_1\setminus\Omega'_2} \nabla u_1.\nabla v_1 \; dx \; + \; \frac{1}{2}\int_{\Omega'_1\cap\Omega'_2} \nabla u_1.\nabla v_1 \; dx \\ &+ \;\; \frac{1}{2}\int_{\Omega'_1\cap\Omega'_2} \nabla u_2.\nabla v_2 \; dx \; + \int_{\Omega'_2\setminus\Omega'_1} \nabla u_2.\nabla v_2 \; dx \end{aligned} \tag{4}$$

and the right-hand side is given by

$$f_h(v) = \int_{\Omega'_1\setminus\Omega'_2} f v_1 \; dx \; + \; \frac{1}{2}\int_{\Omega'_1\cap\Omega'_2} f v_1 \; dx + \; \frac{1}{2}\int_{\Omega'_1\cap\Omega'_2} f v_2 \; dx \; + \int_{\Omega'_2\setminus\Omega'_1} f v_2 \; dx. \tag{5}$$

Here $f \in L^2(\Omega)$ is given. The theory by Cai et al. [CDS99] shows that the H^1 norm of the error is of order h. Their numerical results confirm this and show further that the L^∞ norm and the L^2 norm of the error are both of order h^2.

Composite Mesh Difference Method

A CMDM on two subdomains was described by Starius [Sta77], while Cai et al. [CMS00] studied the case of many subdomains. In [GC99] we outlined a CMDM for solving the second-order elliptic partial differential equation $\mathcal{L}u = f$ in Ω with a Dirichlet boundary condition $u = g$ on $\partial\Omega$.

Given a domain Ω consisting of p nonoverlapping subdomains Ω_i such that $\bar{\Omega} = \cup_{i=1}^{p}\bar{\Omega}_i$, we independently construct a grid of size h_i on each extended subdomain Ω'_i of Ω_i. Due to the extension of the subdomains these grids overlap. We denote by $\Gamma_i = \partial\Omega'_i \cap \partial\Omega$ the intersection of the boundaries $\partial\Omega'_i$ and $\partial\Omega$. The global discretisation $u_h = (u_{h_1}, u_{h_2}, \cdots, u_{h_p})$ on the composite grid is obtained by coupling the local discretisations through the requirement that the solution matches the interpolation of the discrete solutions from adjacent grids. The system of equations consists of p subproblems, each having the following form:

$$\begin{cases} \mathcal{L}_{h_i} u_{h_i} = f_{h_i} & \text{in } \Omega_i', \\ u_{h_i} = g_{h_i} & \text{on } \Gamma_i, \\ u_{h_i} = z_{h_i} = \mathcal{I}_i u_h & \text{on } \partial\Omega_i' \setminus \Gamma_i. \end{cases} \tag{6}$$

Here $\mathcal{I}_i$ is an interface interpolation operator. As shown in [CMS00], the error in the discrete solution satisfies

$$\sum_{i=1}^{p} \|e_{h_i}\|_\infty \leq \left(1 + \frac{\sigma}{1-\tau}\right) \left(\sum_{i=1}^{p} K_i \|\alpha_i\|_\infty + \sum_{i=1}^{p} \|\beta_i\|_\infty \right). \tag{7}$$

In this bound the truncation error $\alpha_i(x) = (\mathcal{L}_{h_i} - \mathcal{L})\, u(x)$ is of order p_i:

$$\|\alpha_i\|_\infty \leq C_{\alpha_i} h_i^{p_i} \tag{8}$$

and the interpolation error $\beta_i(x) = (u - \mathcal{I}_i u)(x)$ is of order q_i:

$$\|\beta_i\|_\infty \leq C_{\beta_i} h_i^{q_i}. \tag{9}$$

The constants C_{α_i}, C_{β_i} and K_i are independent of the mesh size h_i. The interpolation constant $\sigma = \max_i \sigma_i$ is the maximum of the norms $\sigma_i = \|\mathcal{I}_i\|_\infty$ of the interpolation matrices. Let u_{h_i} be the solution of (6) with $f_{h_i} = 0$ and $g_{h_i} = 0$ restricted to the nonoverlapping domain $\bar{\Omega}_i$. Then, in terms of the data z_{h_i} on the interface $\partial\Omega_i' \setminus \Gamma_i$, it can be proved that

$$\|u_{h_i}\|_{\infty, \bar{\Omega}_i} \leq \rho_i \|z_{h_i}\|_{\infty, \partial\Omega_i' \setminus \Gamma_i}. \tag{10}$$

The convergence theory requires the contraction factor of the mapping to be smaller than 1, i.e. $\tau = \max_i\, (\rho_i \sigma) < 1$. Since ρ_i generally depends on the size of the overlap, τ may also depend on the size of the overlap.

Standard P1 Stencil & Bilinear Interpolation

Since both the standard P1 stencil and bilinear interpolation are second order, the error bound (7) shows that the resulting CMDM is also second order. However this scheme does not satisfy the consistent interpolation condition, see [GCR98, GC99], i.e.,

$$\frac{S}{h^2} - (u_{xx} + u_{yy}) = \frac{\gamma_k^2}{2} \left(\xi(1-\xi) u_{xx} + \eta(1-\eta) u_{yy} \right) + \mathcal{O}(h), \tag{11}$$

where S is the stencil, $\gamma_k = k/h$ is the ratio of the mesh sizes. The scaled local coordinates (ξ, η) used in the interpolation and the mesh sizes h and k are shown in Fig. 1. The scheme is consistent only if ξ and η are either 0 or 1, which implies that the two meshes match each other on the interface.

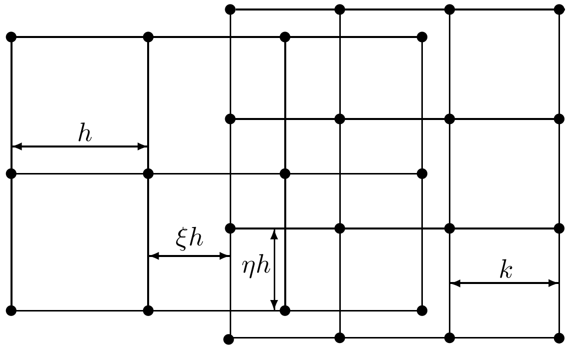


Figure 1: The scaled local coordinates (ξ, η) used in the interpolation.

Mortar Projection in CMDM

We now study a new scheme which takes the mortar approach and drops the weights in the bilinear form (4). In every subdomain we set up a finite element discretisation with the classic bilinear form

$$a_{h_i}(u_i, v_i) = \int_{\Omega_i} \nabla u_i . \nabla v_i \, dx \tag{12}$$

and use the mortar projection (1) to compute the Dirichlet conditions along the interfaces $\gamma_i = \partial\Omega_i' \setminus \Gamma_i$. Hence we have p local problems of the form (6). The mortar projection is a second order accurate interpolation and can be used in a CMDM. The interpolation constant σ can be larger than 1 in the bound $\|\pi\varphi\|_\infty \leq \sigma\|\varphi\|_\infty$ and we may need a large overlap to make the contraction factor ρ small enough in order to have $\tau < 1$.

In Fig. 2 we illustrate that the mortar projection does not, in general, satisfy the maximum principle, i.e. there exists a function φ that satisfies:

$$\|\pi\varphi\|_\infty > \|\varphi\|_\infty. \tag{13}$$

In this special example, the master function is obtained by sampling the function $\sin(\pi x)$ at the grid points $x_i^{(m)} = ih_m$ for $i = 0, 1, \ldots, 5$ where $h_m^{-1} = 5$. The slave nodes are $x_i^{(s)} = ih_s$ for $i = 0, 1, \ldots, 4$ where $h_s^{-1} = 4$. The slave function is set to 0 at the grid points $x_0^{(s)}$ and $x_4^{(s)}$ and the values at $x_i^{(s)}$ for $i = 1, 2, 3$ are determined from (1). We see that the slave function is larger than the master function at $x_2^{(s)} = 0.5$.

The P1 and Q1 finite element discretisations on a uniform mesh can be considered as finite difference stencils for which the local truncation error is second order. All the assumptions for a CMDM are satisfied and the error bound (7) shows that the resulting scheme is second order.

Due to the fact that the values for the Dirichlet boundary conditions on the interior subdomain boundaries, obtained by the mortar projection, are only $\mathcal{O}(h^2)$ accurate, the discretisations which use these values will be inconsistent, since the discretisation error contains the interpolation error divided by h^2. This leaves a constant term in

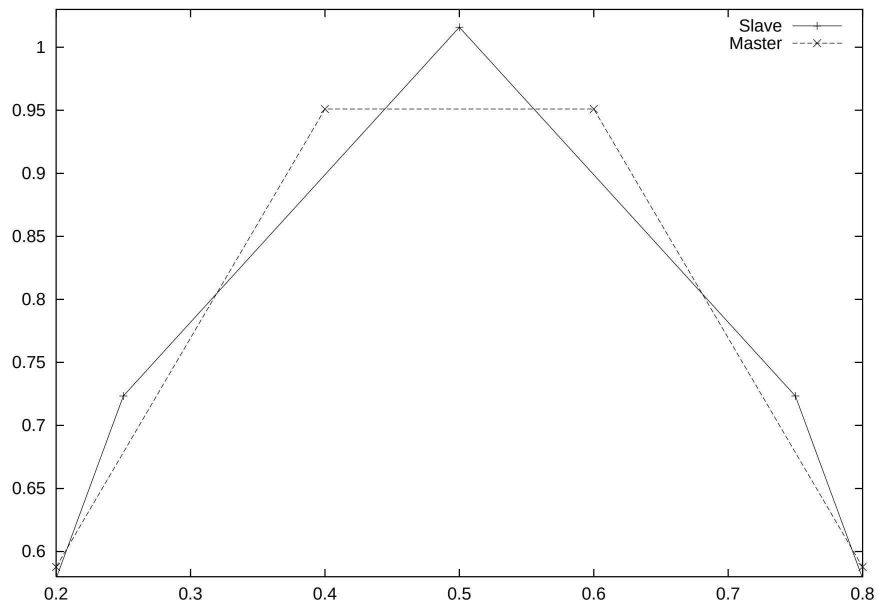


Figure 2: The mortar projection does not satisfy the maximum principle.

the error expansion of the combined discretisation interpolation pair, which does not tend to zero as the mesh size h tends to zero. Consequently this scheme does not satisfy the consistent interpolation condition defined in [GCR98] and we expect the global accuracy to depend on the size of the overlap.

The interpolation from the master to the slave side of the mortar on the interface is only one part of the interpolation issue. In the case of overlapping nonmatching grids we also need to compute the master side of the mortar, which requires evaluating the P1 or Q1 finite element function. This boils down to linear interpolation. As a result for P1 and Q1 finite elements a linear interpolation is done in the direction normal on interface.

Based on our experience with bilinear interpolation we can estimate the effect of doing linear interpolation in the direction normal on interface. Suppose the interface is at $x = x_\Gamma$ between the grid lines at x_i and x_{i+1}. The coefficients for the linear interpolation in the direction normal on the interface are $\xi = (x_\Gamma - x_i)/(x_{i+1} - x_i)$ and $(1 - \xi)$. We expect this interpolation to give rise to a term $\xi(1 - \xi)u_{xx}$ in the bound on the error in the extended subdomain just as in the case of the standard P1 stencil with bilinear interpolation. The numerical results in Table 1 clearly show the influence of the term $\xi(1 - \xi)u_{xx}$ in the error bound.

A final point is the dependency on the overlap. We have already pointed out that a large overlap may be required since the mortar projection does not satisfy the maximum principle. However this does not imply that the error on the nonoverlapping subdomain depends on the size of the overlap. The standard stencils with bicubic interpolation and our modified stencil with 1D cubic interpolation also require some

Table 1: Effect of inconsistent discretisation: results for P1 stencil with bilinear interpolation (columns 3–6) and with mortar projection (columns 7–10).

		bilinear interpolation				mortar projection			
l	ξ_1	$\Vert e_{\Omega_1'}\Vert_\infty$	γ_e	$\Vert e_{\Omega_2'}\Vert_\infty$	γ_e	$\Vert e_{\Omega_1'}\Vert_\infty$	γ_e	$\Vert e_{\Omega_2'}\Vert_\infty$	γ_e
0	0.6	1.65e-2		1.02e-2		2.95e-2		1.64e-2	
1	0.2	2.97e-3	5.57	2.98e-3	3.42	5.02e-3	5.88	5.02e-3	3.26
2	0.4	9.58e-4	3.10	1.55e-4	19.1	1.85e-3	2.71	1.57e-4	32.0
3	0.8	1.60e-4	6.00	2.59e-5	6.00	3.11e-4	5.96	2.59e-5	6.04
4	0.6	5.98e-5	2.67	9.70e-6	2.67	1.17e-4	2.66	9.71e-6	2.67
5	0.2	9.97e-6	6.00	1.62e-6	6.00	1.95e-5	5.99	1.62e-6	6.00
6	0.4	3.74e-6	2.67	6.06e-7	2.67	7.32e-6	2.66	6.06e-7	2.67

overlap in order to make sure that $\tau < 1$ because the interpolation constants are larger than 1. But the numerical results show that there is no dependency on the amount of overlap since these schemes are fully consistent.

In this case the error depends on the size of the overlap and this is due to the inconsistency mentioned above. In Table 2 we show numerical results illustrating the effect of the size of the overlap. These results also confirm the well known fact that increasing the size of the overlap results in faster convergence of the additive Schwarz method.

Numerical results

Our testcase concerns the solution of $-\nabla^2 u = f$ on $\Omega = \Omega_1 \cup \Omega_2$, where $\Omega_1 = [0,1]\times[0,1]$ and $\Omega_2 = [1,2]\times[0,1]$. The r.h.s. f and the Dirichlet boundary conditions g are chosen so that the exact solution is $u(x,y) = x^2$. The overlapping subdomains are $\Omega_1' = [0,1.4]\times[0,1]$ with $h_1 = 0.2\times 2^{-l}$ and $\Omega_2' = [0.75,2]\times[0,1]$ with $h_2 = 0.25\times 2^{-l}$.

In Table 1 we list the L^∞ norm of the error $\Vert e_{\Omega_1'}\Vert_\infty$ and $\Vert e_{\Omega_2'}\Vert_\infty$ on the overlapping extended domains Ω_1' and Ω_2' for the standard P1 stencil with bilinear interpolation and with mortar projection. Both these combinations satisfy all the assumptions for a CMDM so the error bound (7) shows that these methods are second order. For a second order scheme, the ratio between two successive error norms should be 4 when the mesh sizes are halved.

The discussion here is based on the bound on the error in every extended subdomain Ω_i' for the standard P1 stencil with bilinear interpolation. The presence of the inconsistency results in a dependency of the error on $\xi(1-\xi)$, i.e. the relative position of the interface in the other mesh. For this testcase the dominant term in the error bound is $e \approx (\xi(1-\xi)c_1 + c_2)\,h^2$, where c_1 and c_2 are constants independent of ξ and h. With this expression, we can estimate the ratio γ_e between two successive error norms. When the mesh is refined by halving the mesh size, i.e. $h_{i+1} = h_i/2$, we have

$$\gamma_e = \frac{\Vert e_{\Omega'_{h_i}}\Vert_\infty}{\Vert e_{\Omega'_{h_{i+1}}}\Vert_\infty} = \frac{c_1\,(\xi_i(1-\xi_i)+\gamma_c)\,h_i^2}{c_1\,(\xi_{i+1}(1-\xi_{i+1})+\gamma_c)\,h_{i+1}^2} = \frac{\xi_i(1-\xi_i)+\gamma_c}{\xi_{i+1}(1-\xi_{i+1})+\gamma_c}\,4 \tag{14}$$

where $\gamma_c = c_2/c_1$. The worst case scenario is $\gamma_c = 0$ which results in values of 6.00

Table 2: Effect of overlap on the convergence rate of the Schwarz method and on the accuracy for the standard P1 stencil with mortar projection. The same results are obtained with bilinear interpolation.

m	$n_{\mathtt{solver}}$	$n_{\mathtt{prec}}$	$\|e_{\Omega_1}\|_\infty$	$\|e_{\Omega_2}\|_\infty$
0	587	35	1.00e-4	9.99e-5
1	305	26	4.42e-5	4.47e-5
2	159	19	1.74e-5	1.59e-5
3	83	14	6.01e-6	5.07e-6
4	44	10	4.81e-6	3.41e-6
5	24	8	1.77e-6	8.49e-7
6	13	6	1.31e-6	2.77e-7
7	8	5	2.51e-7	1.04e-8

and 2.67 for γ_e since in this testcase the term $\xi(1-\xi)$ alternates between 0.24 and 0.16. For the function $u(x, y) = x^2$ we have $\gamma_c \approx 0$. The numerical results in Table 1 show ratios γ_e equal to 6.00 and 2.67, illustrating the effect of the inconsistency due to linear interpolation in the x-direction.

Apart from this phenomenon both schemes are second order, since fitting a power of the mesh size $\|e_{\Omega_1'}\|_\infty \approx \kappa h^\lambda$ yields $\lambda \approx 2$. The second order accuracy can also be seen when the mesh is refined twice, i.e. the mesh size is divided by 4, in this case $\xi(1-\xi)$ does not change and we get ratios between two successive error norms, which are very close to the theoretical value of 16.

A fully consistent scheme such as the standard P1 stencil with bicubic interpolation or the modified stencil with 1D cubic interpolation by Goossens and Cai [GC99], computes the exact solution up to machine precision for this testcase on any grid.

In order to see the effect of the overlap, we fix the mesh sizes to be $h_1^{-1} = 320$ and $h_2^{-1} = 256$ and vary the overlap according to $\delta_1 = 2 \times 2^m h_1$ and $\delta_2 = 2^m h_2$ for the values of m listed in Table 2. This table shows the number of additive Schwarz iterations required to satisfy the convergence criterion of $\|r_n\|_2 \leq 10^{-10}\|r_0\|_2$ and the L^∞ norm of the error in the nonoverlapping subdomains Ω_1 and Ω_2. First we list the number of iterations ($n_{\mathtt{solver}}$) the method needs when it is used a solver, i.e. in a Richardson iteration, in this case the convergence rate is bounded by τ. We also list the number of iterations ($n_{\mathtt{prec}}$) the method needs when it is used as a right preconditioner for GMRES. As expected the number of additive Schwarz iterations decreases in both cases, as the overlap increases. These results clearly show the advantage of using a Krylov subspace method to accelerate the convergence of the iterative solver. From the results it is clear that the global accuracy of these two methods increases as the overlap increases, thus necessitating substantial overlap. The sensitivity to the size of the overlap is quite high since the error decreases 3 orders of magnitude when the overlap is increased from $m = 0$ to $m = 7$. This is highly undesirable. With a consistent scheme, this error would be independent of the size of the overlap.

Concluding remarks

We studied the effect of using a mortar projection as the interface interpolation in a composite mesh difference method for overlapping nonmatching grids problems. In this case the results are comparable to using bilinear interpolation for the Dirichlet boundary conditions on the interfaces. This is due to the fact that a linear interpolation in the direction that is normal to the interface is used to define the values on the master side of the interface. This results in a dependency of the error on the relative position of the interface nodes in the other mesh. Also due to the inconsistency, the global accuracy depends on the size of the overlap.

References

[BMP94]Christine Bernardi, Yvon Maday, and Anthony T. Patera. A new non conforming approach to domain decomposition: The mortar element method. In Haim Brezis and Jacques-Louis Lions, editors, *Collège de France Seminar*. Pitman, 1994. This paper appeared as a technical report about five years earlier.

[CDS99]X.-C. Cai, M. Dryja, and M. V. Sarkis. Overlapping nonmatching grid mortar element methods for elliptic problems. *SIAM J. Numer. Anal.*, 36:581–606, 1999.

[CMS00]X.-C. Cai, T. P. Mathew, and M. V. Sarkis. Maximum norm analysis of overlapping nonmatching grid discretizations of elliptic equations. *SIAM J. Numer. Anal.*, 37:1709–1728, 2000.

[GC99]S. Goossens and X.-C. Cai. Lower dimensional interpolation in overlapping composite mesh difference methods. In C-H. Lai, P. Bjørstad, M. Cross, and O. Widlund, editors, *Eleventh International Conference on Domain Decomposition Methods*, pages 248–255, Bergen, Norway, 1999. Domain Decomposition Press.

[GCR98]S. Goossens, X.-C. Cai, and D. Roose. Overlapping nonmatching grids method: Some preliminary studies. In J. Mandel, C. Farhat, and X.-C. Cai, editors, *Domain Decomposition Methods 10*, pages 254–261, Providence, RI, 1998. AMS.

[Sta77]G. Starius. Composite mesh difference methods for elliptic boundary value problems. *Numer. Math.*, 28:243–258, 1977.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau,

13. The direct method of lines for incompressible material problems on polygon domains

Houde Han[1], Zhongyi Huang[2]

Introduction

In this paper, we discuss the numerical solutions of the incompressible material problems on a polygon using a semi-discrete method [HH99]. After a suitable transformation of the coordinates, the original boundary value problem (BVP) is reduced to a discontinuous coefficients problem on a rectangle, which is semi-discreted to a BVP of a system of ordinary differential equations (O.D.E's). After solving the BVP of the system by a direct method, the semi-discrete approximation of the original problem is obtained. It's worth to point out that the semi-discrete approximation in form of separable variables naturally possesses the singularity of the original problem. Finally, the numerical examples show that our method is feasible and very effective for solving the incompressible material problems with singularities numerically.

The use of nearly incompressible materials is common in many engineering applications, such as tires, building and bridge bearings, engine mounts, gaskets etc. The natural rubber is the nearly incompressible material, typically the bulk modulus of rubber is several thousand times of the shear modulus. As the material is undergoing plastic deformations, it is nearly incompressible too. We can use the Stokes equations as a model to deal with the incompressible materials. It is also a model for the incompressible fluids. The stress analysis of incompressible materials becomes very significant.

The difficulties for solving the incompressible material problems numerically are: the stress singularity existing at the joint of the interface, the crack-tip or the corner; the incompressibility and the large deformations. To overcome the above difficulties, a great deal of research effort by engineers and mathematicians has been devoted to the development of the FEM(finite element method) for the numerical approximation of incompressible problems. Herrmann [Her65] presented a mixed variational formulation for incompressible isotropic materials. Babuska and Brezzi [Bab73, Bre74] derived the inf-sup condition for the mixed FEM for incompressible problems. Oden et al [OK82, JTOS82] presented general criteria for stability and convergence of mixed and penalty methods(with reduced integration) and applied these criteria to the analysis of elasticity and Stokesian flow problems. Recently, many researchers developed other methods for incompressible problems [AWS95]. For more references, we refer to the paper by Gadala [Gad86]. The singularities of incompressible materials have also been paid attention by researchers [NAHM96]. We know that the singularities at singular points in the incompressible composite material problems are very complex. On each singular point, the singularity is different. Therefore, the standard finite element method and finite difference method can not give satisfied results for incom-

[1] Tsinghua University, Email: hanwu@sun.ihep.ac.cn
[2] Tsinghua University, Email: zhuang@math.tsinghua.edu.cn

pressible material problems. Special consideration is usually needed for the numerical approaches to improve the results.

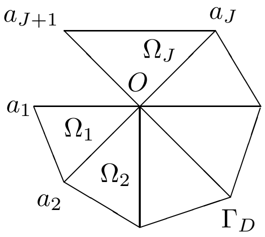


Fig. (1)

In this paper, we deal with the more general incompressible material problems on a polygon. Suppose that $\overline{\Omega} = \bigcup_{i=1}^{J} \overline{\Omega}_i \subset R^2$ is a J-material wedge with a boundary $\Gamma = \overline{Oa_1} \bigcup \overline{Oa_{J+1}} \bigcup \Gamma_D$ (see Fig.(1)), where O is the origin of the coordinate system, $\overline{Oa_1}$ is parallel to the x_1-axis, the ith material occupies Ω_i, $\Gamma_D = \bigcup_{i=1}^{J} \Gamma_i$ with $\Gamma_i = \overline{a_i a_{i+1}}$, and $\{a_i = (x_1^i, x_2^i), i = 1, 2, \cdots, J+1\}$ denote the vertexes of the polygon Ω, $x_1^i = R_i \cos\theta_i$, $x_2^i = R_i \sin\theta_i$ satisfying

$$-\pi = \theta_1 < \theta_2 < \cdots < \theta_{J+1} \leq \pi.$$

We consider the following problem of Stokes equations on the J-material wedge Ω:

$$\begin{aligned} -\mu^i \triangle u^i + \operatorname{grad} p^i &= 0, \qquad \text{in } \Omega_i, \quad 1 \leq i \leq J, &(1)\\ \operatorname{div} u^i &= 0, \qquad \text{in } \Omega_i, \quad 1 \leq i \leq J, &(2)\\ u^i\big|_{\Gamma_i} &= f^i, \qquad 1 \leq i \leq J, &(3)\\ u^{i-1}\big|_{\theta=\theta_i^-} = u^i\big|_{\theta=\theta_i^+}, &\qquad \sigma_n^{i-1}\big|_{\theta=\theta_i^-} = \sigma_n^i\big|_{\theta=\theta_i^+}, \quad 2 \leq i \leq J, &(4)\\ \sigma_n^1\big|_{\theta=\theta_1} = 0, &\qquad \sigma_n^J\big|_{\theta=\theta_{J+1}} = 0, &(5) \end{aligned}$$

where (r, θ) denotes the polar coordinate in the plane; $u^i = (u_1^i, u_2^i)^T$ denotes the displacement in Ω_i; $\mu^i > 0$ is the Lame constant; $f^i = (f_1^i, f_2^i)^T$ is a given vector valued function on the polygonal line Γ_i; $\sigma_n^{i-1}\big|_{\theta=\theta_i^-} = (\sin\theta_i\ \sigma_{11}^{i-1} - \cos\theta_i\ \sigma_{12}^{i-1}, \sin\theta_i\ \sigma_{21}^{i-1} - \cos\theta_i\ \sigma_{22}^{i-1})^T$, $\sigma_n^i\big|_{\theta=\theta_i^+} = (\sin\theta_i\ \sigma_{11}^i - \cos\theta_i\ \sigma_{12}^i, \sin\theta_i\ \sigma_{21}^i - \cos\theta_i\ \sigma_{22}^i)^T$ denote the normal stress on $\overline{Oa_i}$, and $\sigma^i = (\sigma_{kl}^i)_{2\times 2}$ denote the stress tensor in Ω_i with entries

$$\sigma_{kl}^i = -\delta_{kl} p^i + \mu^i \left(\frac{\partial u_k^i}{\partial x_l} + \frac{\partial u_l^i}{\partial x_k} \right), \quad 1 \leq k, l \leq 2, \quad 1 \leq i \leq J.$$

The equivalent variational-differential formulation of problem (1)-(5)

We introduce the transformation of variables on each triangle Ω_i:

$$\left. \begin{aligned} x_1 &= \frac{\rho_i \rho \cos\phi}{\sin(\phi - \alpha_i)} \\ x_2 &= \frac{\rho_i \rho \sin\phi}{\sin(\phi - \alpha_i)} \end{aligned} \right\} \quad \text{for } \theta_i \leq \phi \leq \theta_{i+1}, \quad 0 \leq \rho \leq 1; \qquad (6)$$

with

$$\left.\begin{array}{l}\sin\alpha_i = \dfrac{x_2^{i+1}-x_2^i}{|\overline{a_i a_{i+1}}|}, \qquad \cos\alpha_i = \dfrac{x_1^{i+1}-x_1^i}{|\overline{a_i a_{i+1}}|},\\ |\overline{a_i a_{i+1}}| = \sqrt{(x_1^{i+1}-x_1^i)^2+(x_2^{i+1}-x_2^i)^2},\\ \rho_i = x_2^i\cos\alpha_i - x_1^i\sin\alpha_i.\end{array}\right\}\quad \text{for} \quad 1\le i\le J. \tag{7}$$

We can show that $\rho_i < 0$ and $\sin(\phi-\alpha_i)\neq 0$ for $\theta_i \le \phi \le \theta_{i+1}$. The transformation (6) maps Ω_i onto the rectangle $\widetilde{\Omega}_i = \{(\rho,\phi)|\,\theta_i<\phi<\theta_{i+1},\ 0<\rho<1\}$ and maps segment $\overline{a_i a_{i+1}}$ onto the segment $\{\rho=0,\ \theta_i\le\phi\le\theta_{i+1}\}$ as shown in Fig.(2). Hence the domain Ω is mapped onto $\widetilde{\Omega} = \{(\rho,\phi)|\ -\pi<\phi<\theta_{J+1},\ 0<\rho<1\}$. In the new coordinate (ρ,ϕ), the BVP (1)-(5) is reduced to a discontinuous coefficients problem on the rectangle $\widetilde{\Omega}$. Furthermore, we introduce the following spaces:

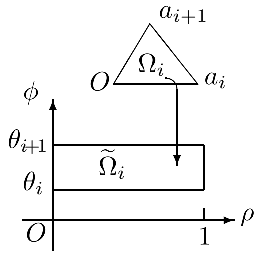


Fig. (2)

$$\begin{aligned}&V_1 = \left\{v_1(\phi)\big|\,v_1\in H^1(\theta_1,\theta_{J+1}),\ \text{namely}\ \ v_1, v_1'\in L^2(\theta_1,\theta_{J+1})\right\},\\ &U_1 = \left\{u_1(\rho,\phi)\Big|\,\text{for fixed}\ \ 0<\rho\le 1, u_1(\rho,\cdot), \frac{\partial u_1}{\partial\rho}(\rho,\cdot), \frac{\partial^2 u_1}{\partial\rho^2}(\rho,\cdot)\in V_1\right\},\\ &V = V_1\times V_1, \qquad U = U_1\times U_1,\\ &Q = \left\{q(\phi)\big|\,q\in L^2(\theta_1,\theta_{J+1})\right\},\\ &S = \left\{p(\rho,\phi)\big|\,\text{for fixed}\ \ 0<\rho\le 1, p(\rho,\cdot)\in Q\right\}.\end{aligned}$$

Then the BVP (1)–(5) is equivalent to the following variational-differential problem:

$$\left.\begin{array}{l}\text{Find}\ \ (u,p)\in U\times S,\ \text{such that}\\ -\left(\dfrac{d}{d\rho}\rho\dfrac{d}{d\rho}\right)a_2(u,v) + \dfrac{d}{d\rho}a_1(u,v) + \dfrac{1}{\rho}a_0(u,v)\\ \qquad\qquad -\left(\dfrac{d}{d\rho}\rho\right)b_1(p,v) + b_0(p,v) = 0, \quad \forall v\in V,\ 0<\rho<1;\\ \rho\dfrac{d}{d\rho}b_1(q,u) + b_0(q,u) = 0, \qquad\qquad\qquad \forall q\in Q,\ 0<\rho<1;\\ u\Big|_{\rho=1} = \widetilde{f}, \qquad u\ \text{is bounded , when}\ \rho\to 0;\end{array}\right\} \tag{8}$$

where

$$\begin{aligned}
a_2(u,v) &= \sum_{i=1}^{n}\int_{\theta_i}^{\theta_{i+1}} \frac{\mu^i}{\sin^2(\phi-\alpha_i)}(u^i)^T\,\mathcal{K}_1^i(\alpha_i)\,v^i\,d\phi,\\
a_1(u,v) &= \sum_{i=1}^{n}\int_{\theta_i}^{\theta_{i+1}} \frac{\mu^i}{\sin(\phi-\alpha_i)}\left[\left(\frac{\partial u^i}{\partial\phi}\right)^T\mathcal{K}_2^i\,v^i-(u^i)^T(\mathcal{K}_2^i)^T\,\frac{dv^i}{d\phi}\right]d\phi,\\
a_0(u,v) &= -\sum_{i=1}^{n}\int_{\theta_i}^{\theta_{i+1}} \mu^i\left(\frac{\partial u^i}{\partial\phi}\right)^T\,\mathcal{K}_1^i(\phi)\,\frac{dv^i}{d\phi}\,d\phi;\\
b_1(q,v) &= \sum_{i=1}^{n}\int_{\theta_i}^{\theta_{i+1}} \frac{\rho_i q^i}{\sin^2(\phi-\alpha_i)}(\sin\alpha_i v_1^i-\cos\alpha_i v_2^i)\,d\phi,\\
b_0(q,v) &= \sum_{i=1}^{n}\int_{\theta_i}^{\theta_{i+1}} \frac{\rho_i q^i}{\sin(\phi-\alpha_i)}\left(\sin\alpha_i\frac{\partial v_1^i}{\partial\phi}-\cos\alpha_i\frac{\partial v_2^i}{\partial\phi}\right)\,d\phi,
\end{aligned}$$

with

$$\mathcal{K}_1^i(\psi)=\begin{pmatrix}1+\sin^2\psi & -\dfrac{\sin 2\psi}{2}\\ -\dfrac{\sin 2\psi}{2} & 1+\cos^2\psi\end{pmatrix},$$

$$\mathcal{K}_2^i=\begin{pmatrix}\cos(\phi-\alpha_i)+\sin\phi\sin\alpha_i & -\sin\phi\cos\alpha_i\\ -\cos\phi\sin\alpha_i & \cos(\phi-\alpha_i)+\cos\phi\cos\alpha_i\end{pmatrix};$$

and v^i denotes the restriction of v on $[\,\theta_i,\,\theta_{i+1}]$.

The numerical solution of the variational-differential problem (8)

Suppose that

$$-\pi=\phi_1<\phi_2<\cdots<\phi_{M+1}=\theta_{J+1} \tag{9}$$

is a partition of the interval $I\equiv[-\pi,\theta_{J+1}]$, such that each of $\{\theta_i\}_{i=1}^n$ is a node of this partition, namely for each θ_i there is a $\phi_j=\theta_i$. Let $h=\max\limits_{1\le j\le M}(\phi_{j+1}-\phi_j)$,

$$Q_h=\left\{q^h(\phi)\,\middle|\,q^h\in C^0(I),\ q^h\big|_{[\phi_j,\phi_{j+1}]}\in P_1([\phi_j,\phi_{j+1}]),\ 1\le j\le M\right\},$$

$$S_h=\left\{p^h(\rho,\phi)\mid \text{for the fixed } 0<\rho\le 1,\quad p^h(\rho,.)\in Q_h\right\}.$$

Assume that $\{\Phi_j(\phi),\ j=1,2,\cdots,M+1\}$ is a basis of Q_h such that $\Phi_j(\phi_i)=\delta_{ij}$, $1\le i,j\le M+1$. Furthermore, we refine the partition (9)

$$-\pi=\phi_1<\phi_{3/2}<\phi_2<\cdots<\phi_{M+1/2}<\phi_{M+1}=\theta_{J+1}. \tag{10}$$

We use quadratic elements to construct the space

$$V_1^h=\left\{v_1^h(\phi)\,\middle|\,v_1^h\in C^0(I),\ v_1^h\big|_{[\phi_j,\phi_{j+1}]}\in P_2([\phi_j,\phi_{j+1}]),\ 1\le j\le M\right\}.$$

Let $\{\psi_1(\phi),\ \psi_{3/2}(\phi),\ \psi_2(\phi),\ \cdots,\ \psi_M(\phi),\ \psi_{M+1/2}(\phi),\ \psi_{M+1}(\phi)\}$ is a basis of the finite dimensional space V_1^h such that

$$\begin{array}{lll}
\psi_j(\phi_i) = \delta_{ij}, & 1 \le i \le M+1, & 1 \le j \le M+1; \\
\psi_j(\phi_{i+1/2}) = 0, & 1 \le i \le M, & 1 \le j \le M+1; \\
\psi_{j+1/2}(\phi_i) = 0, & 1 \le i \le M+1, & 1 \le j \le M; \\
\psi_{j+1/2}(\phi_{i+1/2}) = \delta_{ij}, & 1 \le i \le M, & 1 \le j \le M.
\end{array}$$

In addition, we introduce:

$$\begin{aligned}
U_1^h &= \left\{u_1^h(\rho,\phi) \mid \text{for the fixed } 0<\rho\le 1, \quad u_1^h(\rho,.) \in V_1^h\right\}, \\
V_h &= V_1^h \times V_1^h, \qquad U_h \ = \ U_1^h \times U_1^h. \\
\Phi(\phi) &= \begin{pmatrix} \Phi_1(\phi) & \Phi_2(\phi) & \cdots & \Phi_{M+1}(\phi) \end{pmatrix}^T,
\end{aligned}$$

$$\Psi(\phi) = \begin{pmatrix} \psi_1(\phi) & 0 & \psi_{3/2}(\phi) & 0 & \cdots & \cdots & \psi_{M+1}(\phi) & 0 \\ 0 & \psi_1(\phi) & 0 & \psi_{3/2}(\phi) & \cdots & \cdots & 0 & \psi_{M+1}(\phi) \end{pmatrix}^T.$$

For $p^h(\rho,\phi) \in S_h$, $u^h(\rho,\phi) \in U_h$, and $\widetilde{f}^h(\phi) \in V_h$ is the interpolating function of $\widetilde{f}$ in space V_h, we have

$$\left.\begin{aligned}
p^h(\rho,\phi) &= \Phi^T(\phi) \overset{\wedge}{P_h} (\rho), \\
u^h(\rho,\phi) &= \Psi^T(\phi) \overset{\wedge}{U_h} (\rho), \\
\widetilde{f}^h(\phi) &= \Psi^T(\phi) F \ ,
\end{aligned}\right\} \tag{11}$$

where

$$\overset{\wedge}{P_h} (\rho) = \left(p^h(\rho,\phi_1), p^h(\rho,\phi_2), \cdots, p^h(\rho,\phi_{M+1})\right)^T \ ,$$

$$\overset{\wedge}{U_h}(\rho) = \left(u_1^h(\rho,\phi_1), u_2^h(\rho,\phi_1), u_1^h(\rho,\phi_{\frac{3}{2}}), u_2^h(\rho,\phi_{\frac{3}{2}}), \cdots, u_1^h(\rho,\phi_{M+1}), u_2^h(\rho,\phi_{M+1})\right)^T,$$

$$F = \left(\widetilde{f}_1(\phi_1), \widetilde{f}_2(\phi_1), \widetilde{f}_1(\phi_{\frac{3}{2}}), \widetilde{f}_2(\phi_{\frac{3}{2}}), \cdots, \widetilde{f}_1(\phi_{M+1}), \widetilde{f}_2(\phi_{M+1})\right)^T.$$

Then we have the numerical approximation of the problem (8):

$$\left.\begin{array}{l}
\text{Find } \ (u^h,p^h) \in U_h \times S_h, \ \text{ such that} \\
-\left(\dfrac{d}{d\rho}\rho\dfrac{d}{d\rho}\right) a_2(u^h,v^h) + \dfrac{d}{d\rho} a_1(u^h,v^h) + \dfrac{1}{\rho} a_0(u^h,v^h) \\
\qquad -\left(\dfrac{d}{d\rho}\rho\right) b_1(p^h,v^h) + b_0(p^h,v^h) = 0, \quad \forall v^h \in V_h, \ 0<\rho<1; \\
\rho\dfrac{d}{d\rho} b_1(q^h,u^h) + b_0(q^h,u^h) = 0, \qquad\qquad\qquad \forall q^h \in Q_h, \ 0<\rho<1; \\
u^h\Big|_{\rho=1} = \widetilde{f}^h, \qquad u^h \text{ is bounded }, \text{ when } \rho \to 0.
\end{array}\right\} \tag{12}$$

Using (11), the discrete variational-differential problem (12) is equivalent to a BVP of a system of O.D.E's. We can reduce the BVP of the system of O.D.E's to an eigenvalue problem. After solving the eigenvalue problem numerically, we obtain neither more nor less than $4M+2$ eigenvalues λ_j^h $(j = 1, 2, \cdots, 4M+2)$ with non-negative real part corresponding to the eigenvectors $(\zeta_j, \xi_j, \eta_j)^T$, $j = 1, 2, \cdots, 4M+2$, where $\lambda_1^h = \lambda_2^h = 0$, $\xi_1 = (1, 0, \cdots, 1, 0)^T \in R^{4M+2}$, $\zeta_1 = 0$, $\xi_2 = (0, 1, \cdots, 0, 1)^T \in R^{4M+2}$, $\zeta_2 = 0$. Particularly we assume λ_j^h $(1 \leq j \leq 2m)$ are real eigenvalues and λ_j^h $(2m+1 \leq j \leq 4M+2)$ are complex eigenvalues with nonzero imaginary parts such that $\lambda_{2l}^h = \overline{\lambda}_{2l-1}^h$ $(m+1 \leq l \leq 2M+1)$. Introduce matrices

$$\begin{aligned}
D(\rho) &= \big[\rho^{\lambda_1^h}\xi_1, \ \cdots, \ \rho^{\lambda_{2m}^h}\xi_{2m}, \ \mathrm{Re}(\rho^{\lambda_{2m+2}^h}\xi_{2m+2}), \ \mathrm{Im}(\rho^{\lambda_{2m+2}^h}\xi_{2m+2}), \\
&\quad \cdots, \mathrm{Re}(\rho^{\lambda_{4M+2}^h}\xi_{4M+2}), \ \mathrm{Im}(\rho^{\lambda_{4M+2}^h}\xi_{4M+2})\big], \\
E(\rho) &= \big[\rho^{\lambda_1^h-1}\eta_1, \ \cdots, \ \rho^{\lambda_{2m}^h-1}\eta_{2m}, \ \mathrm{Re}(\rho^{\lambda_{2m+2}^h-1}\eta_{2m+2}), \ \mathrm{Im}(\rho^{\lambda_{2m+2}^h-1}\eta_{2m+2}), \\
&\quad \cdots, \mathrm{Re}(\rho^{\lambda_{4M+2}^h-1}\eta_{4M+2}), \ \mathrm{Im}(\rho^{\lambda_{4M+2}^h-1}\eta_{4M+2})\big].
\end{aligned}$$

Finally, we get the semi-discrete approximate solution of problem (12):

$$u^h(\rho, \phi) = \Psi(\phi)^T D(\rho) D(1)^{-1} F, \tag{13}$$

$$p^h(\rho, \phi) = \Phi(\phi)^T E(\rho) D(1)^{-1} F. \tag{14}$$

Remark: *We can deal with the Neumann boundary value problem based on the expression of the semi-discrete solution of the Dirichlet BVP given in (13)–(14). In addition, We can similarly define the stress intensity factors (SIFs) K_I and K_{II} at the crack-tip in the incompressible materials.*

Numerical examples

In order to demonstrate the effectiveness of the direct method of lines given in this paper, two numerical examples are discussed. First, we consider the following problem with a corner.

Example 1. We consider the problem

$$-\mu^i \triangle u^i + \mathrm{grad}\, p^i = 0, \qquad \text{in } \Omega_i, \quad 1 \leq i \leq J, \tag{15}$$

$$\mathrm{div}\, u^i = 0, \qquad \text{in } \Omega_i, \quad 1 \leq i \leq J, \tag{16}$$

$$u^i\big|_{\Gamma_i} = f^i, \qquad 1 \leq i \leq J, \tag{17}$$

$$u^{i-1}\big|_{\theta=\theta_i^-} = u^i\big|_{\theta=\theta_i^+}, \qquad \sigma_n^{i-1}\big|_{\theta=\theta_i^-} = \sigma_n^i\big|_{\theta=\theta_i^+}, \quad 2 \leq i \leq J, \tag{18}$$

$$\sigma_n^1\big|_{\theta=\theta_1} = 0, \qquad \sigma_n^J\big|_{\theta=\theta_{J+1}} = 0, \tag{19}$$

where $J = 4$ and Ω_i $(i = 1, 2, \cdots, 4)$ is given in Fig. (3),

$$\begin{aligned}
\Gamma_D &= \partial\Omega \setminus \left\{ x \in R^2 \,\middle|\, -1 \leq x_1 \leq 0, \, x_2 = 0^- \text{ or } 0 \leq x_2 \leq 1, \, x_1 = 0^+ \right\}, \\
f &= \begin{pmatrix} f_1 \\ f_2 \end{pmatrix} = \begin{pmatrix} x_2 \\ -x_1 \end{pmatrix}
\end{aligned}$$

We assume that $\mu^i = 2^{i-1}\mu$, for $1 \leq i \leq J$. In all examples we let $\mu = 300$. We know the exact solution is $u = (x_2, \, -x_1)^T, \ p = 0$.

Let M be an even positive integer, the partition of $[-\pi, \pi/2]$ is given by (10) with $\theta_J = \pi/2$, $h = \frac{3\pi}{4M}$,

$$\begin{aligned} &\phi_j = -\pi + 2(j-1)h, && j = 1, 2, \cdots, M+1, \\ &\phi_{j+1/2} = \phi_j + h, && j = 1, 2, \cdots, M. \end{aligned} \tag{20}$$

Denoting the numerical solution of (15)–(19) by (u^h, p^h), the results of the first three eigenvalues for the one material case of Example 1 are given in Table 1 for different M, which have been compared with the exact results.

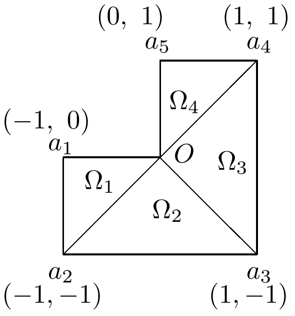


Fig.(3)

Table 1: The results of Example 1.

M	λ_3^h	λ_4^h	λ_5^h	Error
12	0.550231	0.939503	0.997947	2.9338e-2
24	0.549332	0.937753	0.999883	4.3508e-3
48	0.548838	0.937581	0.999986	1.1345e-3

where Error $= \|u - u^h\|_{1,\Omega} + \|p - p^h\|_{0,\Omega}$.

Example 2. We now consider a interface crack problem with Neumann boundary condition (see Fig. (4)):

$$-\mu^i \triangle u^i + \operatorname{grad} p^i = 0, \quad \text{in } \Omega_i, \quad 1 \le i \le J, \tag{21}$$

$$\operatorname{div} u^i = 0, \quad \text{in } \Omega_i, \quad 1 \le i \le J, \tag{22}$$

$$\sigma_n^i\big|_{\Gamma_i} = g^i, \quad 1 \le i \le J, \tag{23}$$

$$u^{i-1}\big|_{\theta=\theta_i^-} = u^i\big|_{\theta=\theta_i^+}, \qquad \sigma_n^{i-1}\big|_{\theta=\theta_i^-} = \sigma_n^i\big|_{\theta=\theta_i^+}, \quad 2 \le i \le J, \tag{24}$$

$$\sigma_n^1\big|_{\theta=\theta_1} = 0, \qquad \sigma_n^J\big|_{\theta=\theta_{J+1}} = 0, \tag{25}$$

where

$$g(x) = \begin{pmatrix} g_1(x) \\ g_2(x) \end{pmatrix}$$

$$g_1(x) = 0, \ g_2(x) = \begin{cases} -1, & x_2 = -1, \\ 1, & x_2 = 1, \\ 0, & \text{otherwise;} \end{cases}$$

$$\mu^i = 2^{i-1}\mu, \quad 1 \le i \le J.$$

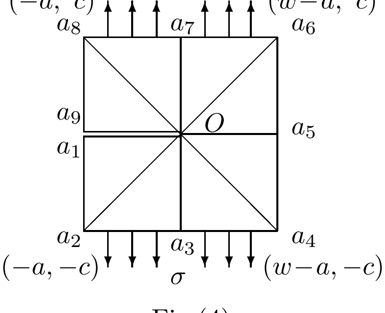


Fig.(4)

Here $J = 8$, $c/w = a/w = 0.5$. Let M be an even positive integer, the partition of $[-\pi, \pi]$ is given by (10) with $\theta_J = \pi$, $h = 2\pi/M$ and

$$\begin{aligned} &\phi_j = -\pi + (j-1)h, && j = 1, 2, \cdots, M+1, \\ &\phi_{j+1/2} = \phi_j + h/2, && j = 1, 2, \cdots, M. \end{aligned} \tag{26}$$

The results for Example 2 are given in Table 2 for different M, where $K^{h*} = K^h \cdot a^{\lambda_3^h - 1}/\sigma\sqrt{\pi} = K_I^{h*} + iK_{II}^{h*}$. We can see that our method is effective for solving the incompressible problems and calculating SIFs.

Table 2: The results of Example 2.

M	λ_3^h	λ_4^h	λ_5^h	λ_6^h	K_I^{h*}	K_{II}^{h*}
16	0.576618	0.600351	0.964218	0.997352	0.481622	0.0336196
32	0.574633	0.598260	0.956448	0.999875	0.479231	0.0320021
64	0.573838	0.597324	0.953956	0.999993	0.477376	0.0313716

References

[AWS95]B. Fischer A. Wathen and D. Silvester. The convergence rate of the minimal residual method for the Stokes problem. *Numer. Math.*, 71:121–134, 1995.

[Bab73]I. Babuska. The finite element method with lagrangian multipliers. *Numer. Math.*, 20:179–192, 1973.

[Bre74]F. Brezzi. On the existence, uniqueness and approximation of saddle-point problems arising from lagrangian multipliers. *R.A.I.R.O.*, 8:129–151, 1974.

[Gad86]M. S. Gadala. Numerical solutions of nonlinear problems of continua — ii. survey of incompressibility constraints and software aspects. *Computers & Structures*, 22:841–855, 1986.

[Her65]L. R. Herrmann. Elasticity equations for incompressible and nearly incompressible. *AIAA Journal*, 3:1896–1900, 1965.

[HH99]H. Han and Z. Huang. The direct method of lines for the numerical solutions of interface problem. *Comput. Meth. Appl. Mech. Engrg.*, 171(1-2):61–75, March 1999.

[JTOS82]N. Kikuchi J. T. Oden and Y. J. Song. Penalty finite element for the analysis of Stokesian flows. *Comput. Meth. Appl. Mech. Engrg.*, 31:297–329, 1982.

[NAHM96]H. Ghfiri N. A. Hocine, M. N. Abdelaziz and G. Mesmacque. Evaluation of the energy parameter j on rubber-like materials: Comparison between experimental and numerical results. *Engrg. Frac. Mech.*, 55:919–933, 1996.

[OK82]J. T. Oden and N. Kikuchi. Finite element methods for constrained problems in elasticity. *Int. J. Numer. Mech. Engrg.*, 18:701–725, 1982.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau,

14. The Coupling of Natural BEM and Composite Grid FEM

Q.Y. Hu[1], D.H. Yu[2]

Introduction

The coupling of boundary elements and finite elements is of great importance for the numerical treatment of boundary value problems posed on unbounded domains. It permits us to combine the advantages of boundary elements for treating domains extended to infinity with those of finite elements in treating the complicated bounded domains.

The standard procedure of coupling the boundary element and finite element methods is described as follows. First, the (unbounded) domain is divided into two subregions, a bounded inner region and an unbounded outer one, by introducing an auxiliary common boundary. Next, the problem is reduced to an equivalent one in the bounded region. There are many ways to accomplish this reduction (refer to [Cos87], [FY83], [GHW94], [HZ94], [JN80], [Med98] and [ESH79]). The FEM-BEM coupled method can be viewed as a domain decomposition method to solve unbounded domain problems.

The natural boundary reduction method proposed by [FY83] has obvious advantages over the usual boundary reduction methods: the coupled bilinear form preserve automatically the symmetry and coerciveness of the original bilinear form,so not only the analysis of the discrete problem is simplified, but also the optimal error estimates and the numerical stability are restored (see [FY83] and [Yu93]).

It is well known that the analytic solution of the Dirichlet exterior problem is in general singular at the corner points. The fast adaptive composite grid (iteration) method advanced by McCormick (refer to [BPWX91], [MT86] and [McC89]) is very effective in dealing with this kind of local singularity. However, it can not be applied directly to the case of unbounded domain.

In the present paper we combine the composite grid method with the coupling method of natural boundary element and finite element to handle the corner singularity of the Dirichlet exterior problems. Under suitable assumptions we obtain the optimal error estimates of the corresponding approximate solutions. The underlying linear system is expensive to solve directly due to the complicated structure (which is neither sparse nor band). Instead, we introduce two iterative methods to solve this coupled system: (1) a combination algorithm between the inexact two-level multiplicative Schwarz method and the steepest descent method; (2) the preconditioning

[1] Institute of Computational Mathematics and Scientific/ Engineering Computing, Chinese Academy of Sciences, Beijing 100080, China. The work of the author was supported by the National Natural Science Foundation and the National Postdoctor Foundation of China (email: hqy@lsec.cc.ac.cn)

[2] Institute of Computational Mathematics and Scientific/ Engineering Computing, Chinese Academy of Sciences, Beijing 100080, China. The work of the author was supported by the State Major Key Project for Basic Researches of China (email: ydh@lsec.cc.ac.cn)

conjugate gradient (PCG) method by constructing a kind of simple preconditioner for the coupled "stiffness" matrix. Both the two algorithms have the fast convergence speed independent of the (coarse and fine) mesh sizes, which has been proved in [HY99b] and [HY99a]. We give numerical examples to illustrate our theoretical results.

The FEM-BEM coupling

We consider the following model exterior Dirichlet problem in two dimensions:

$$-\Delta u = f \quad in \;\; \Omega^c = \mathbf{R}^2 \backslash (\Omega \cup \Gamma), \tag{1}$$

$$u = g \qquad on \;\; \partial\Omega \tag{2}$$

with the asymptotic condition:

$$u(x, y) \;\; is \;\; bounded \;\; as \;\; r = \sqrt{x^2 + y^2} \to \infty.$$

Where Ω is a Lipschitz bounded domain, f and g are given functions satisfying $f \in L^2(\Omega^c)$ and $g \in H^{\frac{1}{2}}(\partial\Omega)$.

The variational form of the boundary value problem (1) is: to find $u \in \bar{H}^1(\Omega^c)$, such that

$$D(u, v) = (f, v), \quad \forall \; v \in \bar{H}^1_0(\Omega^c), \tag{3}$$

where

$$\bar{H}^1(\Omega^c) = \{v : \frac{v}{\sqrt{(r^2+1)} \cdot \ln(r^2+2)}, \frac{\partial v}{\partial x}, \frac{\partial v}{\partial y} \in L^2(\Omega^c)\},$$

$$\bar{H}^1_0(\Omega^c) = \{v : v \in H^1(\Omega^c), v|_{\partial\Omega} = 0\},$$

$$D(u, v) = (\nabla u, \nabla v), \;\; \forall \; u, v \in \bar{H}^1(\Omega^c),$$

with $(\cdot, \cdot)$ be the L^2 innerproduct on Ω^c.

Let Ω_0 is a circle disc (with the radius R) containing Ω and having a boundary Γ. Set $\Omega_1 = \Omega^c \cap \Omega_0$ and $\Omega_2 = \Omega_0^c = \mathbf{R}^2 \backslash \Omega_0$. We assume that the ratio of the area of Ω_1 over the area of Ω is not small.

Let $G(p, p')$ denote the Green function of the Laplace operator on the domain Ω_2. Set

$$\frac{\partial}{\partial n} G(p, p') = G_n^{(2)}(p, p'), \;\; p, p' \in \Gamma,$$

and

$$-\int_\Gamma \frac{\partial^2}{\partial n \partial n'} G(p, p') \cdot u(p') dp' = K_2 u(p), \;\; p \in \Gamma.$$

where n and n' denote respectively the exterior normal vectors of Γ (which is regarded as the boundary of Ω_2) at the points p and p'.

Define the bilinear form

$$D_1(u, v) = \int_{\Omega_1} \nabla u \cdot \nabla v ds, \quad u, v \in H^1(\Omega_1)$$

and the Sobolev spaces

$$H_g^1(\Omega_1) = \{v : v \in H^1(\Omega_1), v|_{\partial\Omega} = g\}$$

and

$$H_0^1(\Omega_1) = \{v : v \in H^1(\Omega_1), v|_{\partial\Omega} = 0\}.$$

Let $< \cdot, \cdot >_\Gamma$ denote the L^2 innerproduct on Γ. Then, it can be verified by the Green formular that (3) is equivalent to the coupling variational problem (see [Yu93]): to find $u \in H_g^1(\Omega_1)$ such that

$$D_1(u, v) + < K_2 u, v >_\Gamma = \int\int_{\Omega_1} f v dx dy - < w_f, v >_\Gamma, \ \forall \ v \in H_0^1(\Omega_1), \tag{4}$$

where

$$w_f(p) = \int\int_{\Omega_2} f(p') G_n^{(2)}(p, p') dp', \ p \in \Gamma.$$

The coupling bilinear form

$$A(u, v) = D_1(u, v) + < K_2 u, v >_\Gamma$$

is symmetric, bounded and coercive in $H_0^1(\Omega_1)$, so (4) has unique solution $u \in H_g^1(\Omega_1)$.

Composite grid discretization

Without loss of generality, we assume that: (i) the domain Ω is a polygon; (ii) $g \equiv 0$. Let the auxiliary boundary Γ be divided into m circular arcs with the same length. Moreover, let the domain Ω_1 be divided into some quasi-uniform triangular or quarilateral elements with the diameter H ($\approx 2\pi R/m$), such that the finite element nodes on Γ coincide with the m dividing points on Γ. The corresponding piecewise linear finite element space is denoted by $S_H(\Omega_1) \subset H_g^1(\Omega_1) = H_0^1(\Omega_1)$. Because the analytic solution u is in general singular nearby the concave angle points of Ω_1, even if the given functions f and g are smooth enough on their definition domains Ω^c and $\partial\Omega$, the finite-dimensional subspace $S_H(\Omega_1)$ can not provide a "good" approximation of u unless the mesh size H is very small. Let Ω_3 is a subdomain of Ω_1, such that $\overline{\Omega_3}$ containes the concave angle points of Ω_1. We assume that Ω_3 is just the union set of some elements of Ω_1. Set

$$H_0^1(\Omega_3) = \{v : v \in H^1(\Omega_1), supp\ v \subset \Omega_3\}.$$

We make a refining division to Ω_3, such that the diameter of finer elements is $h < H$. Let $S_h^0(\Omega_3) \subset H_0^1(\Omega_3)$ be the corresponding piecewise linear finite element space. We define the composite grid space $S_{h,H} \subset H_g^1(\Omega_1) = H_0^1(\Omega_1)$ by $S_{h,H} = S_H(\Omega_1) + S_h^0(\Omega_3)$.

The discrete variational problem of (4) is: to find $u_{h,H} \in S_{h,H}$ such that

$$A(u_{h,H}, v) = \int\int_{\Omega_1} f v dx dy - < w_f, v >_\Gamma, \ \forall \ v \in S_{h,H} \cap H_0^1(\Omega_1). \tag{5}$$

For this approximation, we have the following error estimates (which have been proved in [HY99b] or [HY99a]).

Theorem 1 *Assume that $f \in L^2(\Omega^c)$ and $g \in H^{\frac{1}{2}}(\partial\Omega)$. Then, there is a decomposition $u = \hat{u} + \tilde{u}$, such that $\hat{u} \in H^2(\Omega_1) \cap H_0^1(\Omega_1)$ and $\tilde{u} \in H_0^1(\Omega_3) \cap H^{1+\alpha}(\Omega_3)$ with $0 < \alpha < 1$. Moreover, we have*

$$(\|u_{h,H} - u\|^2_{1,\Omega_1} + \|u_{h,H} - u\|^2_{\frac{1}{2},\Gamma})^{\frac{1}{2}} \le C(h^\alpha \|\tilde{u}\|_{1+\alpha,\Omega_3} + H\|\hat{u}\|_{2,\Omega_1}) \tag{6}$$

and

$$\|u_{h,H} - u\|_{0,\Omega_1} \le C(h^{2\alpha} \|\tilde{u}\|_{1+\alpha,\Omega_3} + H^2\|\hat{u}\|_{2,\Omega_1}). \tag{7}$$

Remark 1 *The above theorem indicates that the fine mesh size h and the coarse mesh size H should satisfy $h^\alpha \approx H$.*

It is clear that the stiffness matrix of the bilinear form $A(\cdot,\cdot)$ is neither sparse nor band. Thus, it is expensive to solve the discrete problem (5) in the direct way.

A iteration algorithm of the discrete problem

In this section, we introduce an iteration algorithm to solve (5).

For ease of notation, we set

$$V = S_{h,H},\ V_1 = S_h^0(\Omega_3) \text{ and } V_2 = S_H(\Omega_1).$$

At first, we describe a version of the composite grid iteration algorithm (refer to [MT86] and [McC89]), which is applied to solving (5).

The standard algorithm Let $u_0 \in V$ be a initial approximation. When we have gotten $u_n \in V$, we look for $u_{n+1} \in V$ as follows:

1^o Solving $u^1 \in V_1$ by

$$A(u^1, v_1) = \Phi(v_1) - A(u_n, v_1),\ \forall v_1 \in V_1,$$

namely,

$$D_1(u^1, v_1) = (f, v_1) - D_1(u_n, v_1),\ \forall v_1 \in V_1.$$

Set

$$u_{n+\frac{1}{2}} = u_n + u^1;$$

2^o Solving $u^2 \in V_2$ by

$$D_1(u^2, v_2) = \Phi(v_2) - A(u_{n+\frac{1}{2}}, v_2),\ \forall v_2 \in V_2.$$

Set

$$u_{n+1} = u_{n+\frac{1}{2}} + \theta u^2,$$

where $\theta > 0$ is a relaxation parameter (remaining to be determined).

We define the projection-like operator $Q_H : V \to V_2$

$$D_1(Q_H\varphi, \psi) = A(\varphi, \psi),\ \varphi \in V,\ \forall \psi \in V_2.$$

Here, we have used the fact that $D_1(\cdot,\cdot)$ is symmetric and positive definite in V_2. Let $e_n = u_{h,H} - u_n$ denote the error function. It can be verified directly that the error propagation relation is

$$e_{n+1} = (I - \theta Q_H)(I - P_h)e_n.$$

It can be shown (refer to [HY99b]) that there is a constant $\tilde{C} > 1$, such that

$$D_1(\varphi,\varphi) \leq A(\varphi,\varphi) \leq \tilde{C} D_1(\varphi,\varphi), \ \forall \varphi \in V.$$

Thus, from the convergence theory of the multiplicative Schwarz iteration (see [SBG96] and [Xu92]), we know that the above iteration algorithm is convergent, provided the relaxation parameter θ is chosen as $0 < \theta < 2/\tilde{C}$. However, there is no simple way to estimate the value of the constant $\tilde{C}$.

We discuss how to choose the relaxation parameter θ when we do not know the value of the constant $\tilde{C}$.

Set $e_0(\theta_{-1}^0) = u_{h,H} - u_0$. If we have determined value of positive number θ_{n-1}^0, then we set

$$e_{n+1}(\theta_n) = (I - \theta_n Q_H)(I - P_h)e_n(\theta_{n-1}^0), \ n = 0, 1, \cdots .$$

Let $\|\cdot\|$ denote the norm generated by the innerproduct $[\cdot,\cdot] = A(\cdot,\cdot)$. We define the function of θ_n by

$$F(\theta_n) = \|e_{n+1}(\theta_n)\|^2, \ n = 0, 1, \cdots .$$

Our idea is to select properly a positive number θ_n^0, such that

$$F(\theta_n^0) = \min_{\theta_n} F(\theta_n), \ n = 0, 1, \cdots . \tag{8}$$

Without loss of generality, we assume that $g_n = (I - P_h)e_n(\theta_{n-1}^0) \neq 0$ (otherwise, $u_{n+\frac{1}{2}} = u_{h,H}$). Since there is a decomposition $g_n = v_n^1 + v_n^2$, with $v_n^1 \in V_1$ and $v_n^2 \in V_2$, we have

$$\begin{aligned} \|g_n\|^2 &= [g_n, v_n^1 + v_n^2] && (9)\\ &= D_1(g_n, v_n^1) + [g_n, v_n^2] && (10)\\ &= D_1(Q_H g_n, v_n^2). && (11) \end{aligned}$$

Hence $Q_H g_n \neq 0$. Therefore, it follows from (4.1) that

$$F'(\theta_n) = 0.$$

Thus, we obtain

$$\theta_n^0 = \frac{[g_n, Q_H g_n]}{\|Q_H g_n\|^2}.$$

We must illustrate how to calculate these positive numbers θ_n^0. In fact, $Q_H g_n$ can be obtained directly by the step 1 and step 2 in the above standard algorithm, namely, $Q_H g_n = u^2$. Furthermore, we have

$$[g_n, Q_H g_n] = D_1(Q_H g_n, Q_H g_n) = |u_2|_{1,\Omega_1}^2.$$

Now, we can describe an new algorithm.

Schwarz-steepest descent algorithm Let $u_0 \in V$ be a initial approximation. When we have gotten $u_n \in V$, we look for $u_{n+1} \in V$ as follows:

1^o Solving $u^1 \in V_1$ by

$$D_1(u^1, v_1) = (f, v_1) - D_1(u_n, v_1), \ \forall v_1 \in V_1,$$

and set

$$u_{n+\frac{1}{2}} = u_n + u^1;$$

2^o Solving $u^2 \in V_2$ by

$$D_1(u^2, v_2) = \Phi(v_2) - A(u_{n+\frac{1}{2}}, v_2), \ \forall v_2 \in V_2.$$

3^o Computing norms $|u^2|^2_{1,\Omega_1}$ and $\|u^2\|^2$, and set

$$u_{n+1} = u_{n+\frac{1}{2}} + \theta^0_n u^2,$$

with $\theta^0_n = \frac{|u^2|^2_{1,\Omega_1}}{\|u^2\|^2}$.

For the above algorithm, we have the following convergence result (see [HY99b]).

Theorem 2 *There is a constant C independent of h and H, such that*

$$\|e_{n+1}(\theta^0_n)\|^2 \leq (1 - \frac{1}{C})\|e_n(\theta^0_{n-1})\|^2, \ n \geq 1. \tag{12}$$

Remark 2 *If we set $\theta^0_n = 1$, which corresponds to the standard two-level multiplicative Schwarz algorithm, this algorithm may be divergent.*

A preconditioner for the discrete system

Because the stiffness matrix associated with the discrete problem (5) is symmetric and positive definite, this linear system can also be solved by the PCG method.

Now we construct a kind of preconditioner for this bilinear form.

For convenience' sake, we define the operators $A, \bar{A} : V \to V$ by

$$(A\varphi, \psi) = D_1(\varphi, \psi) + < K_2\varphi, \psi >_\Gamma, \ \forall \varphi, \ \psi \in V$$

and

$$(\bar{A}\varphi, \psi) = D_1(\varphi, \psi), \varphi \in S_{h,H}, \ \forall \psi \in V.$$

Let $A_1 : V_1 \to V_1$ and $A_2 : V_2 \to V_2$ denote the restrictions of the operator $\bar{A}$, which satisfy

$$(A_1\varphi_1, \psi_1) = (\bar{A}\varphi_1, \psi_1), \ \varphi_1 \in V_1, \ \forall \psi_1 \in V_1$$

and

$$(A_2\varphi_2, \psi_2) = (\bar{A}\varphi_2, \psi_2), \ \varphi_2 \in V_2, \ \forall \psi_2 \in V_2.$$

It is clear that the operators A_1 and A_2 are symmetric and positive definite with respect to the L^2 innerproduct.

We define the preconditioner of the operator A as

$$B = A_1^{-1}Q_1 + A_2^{-1}Q_2, \tag{13}$$

where $Q_1 : V \rightarrow V_1$ and $Q_2 : V \rightarrow V_2$ are the L^2 orthogonal projection operators.

The following result has been proved in [HY99a].

Theorem 3 *There exists a constant C independent of h and H, such that*

$$cond(BA) \leq C. \tag{14}$$

Remark 3 *Since the operator K_2 in the second section can be expressed explicitly, we need not solve any (singular) integral equation. Instead, we need only to calculate some singular integrations (refer to [HY99b], [HY99a] and [Yu93]). Besides, only two subproblems with two standard bases are needed to be solved. These are the main merits of the algorithm introduced in this paper.*

Remark 4 *The preconditioning algorithm introduced in this section has faster convergence speed than the Schwarz algorithm introduced in the last section (see the next section). Moreover, it is additive, so the result can be extended directly to the case of inexact local solver. On the other hand, the stiffness matrix of (5) can not be obtained directly (refer to [MT86], [McC89] and [SBG96]), because $V_1 \cap V_2 \neq \emptyset$. For the Schwarz algorithm given in the last section, the global stiffness matrix of the bilinear form $A(\cdot,\cdot)$ need not be generated (therefore, no need to care about basis for V). Besides, this algorithm has minimal memory requirement. These are the merits of the Schwarz algorithm.*

Numerical examples

To illustrate the theoretical results stated in this paper, we consider

$$-\Delta u = f, \quad \in \ \Omega^c, \tag{15}$$

$$u = g, \qquad on \ \partial\Omega, \tag{16}$$

where $\Omega = [-1, 0] \times [-1, 0]$; f and g are given functions such that its exact sulution is $u(x, y) = \frac{(x^2+y^2)^{\frac{1}{3}}}{(x+\frac{1}{2})^2+(y+\frac{1}{2})^2}$.

$$(f(x,y) = -u(x,y)\{\frac{2/3}{(x^2+y^2)[(x+\frac{1}{2})^2+(y+\frac{1}{2})^2]} + \frac{8/3}{(x+\frac{1}{2})^2+(y+\frac{1}{2})^2} - \frac{8/9}{x^2+y^2}\})$$

It is clear that the analytic solution u is singular at the corner point (0,0) ($\alpha = \frac{2}{3}$). This problem is solved by the method introduced in the second section. Here, radius of the auxiliary circle Γ is $R = 2$. Moreover, the subdomain Ω_3 is chosen as the sector with radius 1. We use quasi-uniform triangular elements. The resulting linear system is solved by the Schwarz-steepest descent algorithm (or the PCG method with the preconditioner defined in the last section).

The error estimates (6) and (7) are confirmed by Table 1 (with the equivalent discrete norms).

Table 1
error estimates ($H = 4\pi/m, h = H/4$)

m	$\|u_H - u\|_{1,\Omega_1}$	$\|u_{h,H} - u\|_{1,\Omega_1}$	$\|u_H - u\|_{0,\Omega_1}$	$\|u_{h,H} - u\|_{0,\Omega_1}$
20	9.87D-1	7.25D-1	9.31D-1	4.66D-1
40	6.37D-1	3.64D-1	3.75D-1	1.20D-1
80	4.12D-1	1.83D-1	1.53D-1	3.14D-2
160	2.65D-1	9.24D-2	6.14D-2	8.07D-3 (or 8.09D-3)

The numbers of iteration are given in Table 2 (or Table 3), which can confirm Theorem 8 (or Theorem 13). Here, the domination error with the discrete l^2 norm is 5.0×10^{-5}.

Table 2
numbers of iteration

m	20	40	80	160
iter	21	22	21	22

Table 3
numbers of iteration

m	20	40	80	160
PCG	14	14	15	14

References

[BPWX91]James H. Bramble, Joseph E. Pasciak, Junping Wang, and Jinchao Xu. Convergence estimates for multigrid algorithms without regularity assumptions. *Math. Comp.*, 57(195):23–45, 1991.

[Cos87]M Costabel. Symmetric methods for the coupling of finite elements and boundary elements. In *Boundary Elements IV, Vol.1, Comput. Mech.*, pages 441–420. Brebier, Southampton, 1987.

[ESH79]W. Wendland E. Stephan and G. Hsiao. On the integral equation method for the plane mixed boundary value problem of the Laplacian. *Math. Methods Appl. Sci.*, 1:265–321, 1979.

[FY83]K. Feng and D. Yu. Canonical integral equation of elliptic boundary value problems and their numerical solution. In *Proc. of China-France Symp. On FEM*, pages 211–252, Beijing, 1983. Science Press.

[GHW94]B. Khoromskij G. Hsiao and W. Wendland. Boundary integral operators and domain decomposition. Technical report, Mathematishes Institute, Universitat Stuttgart, 1994.

[HY99a]Q. Hu and D. Yu. A preconditioner for coupled system of natural bem and composite grid fem. Technical report, Institute of Computational Mathematics and Scientific/Engineering Computing, Chinese Academy of Science, Beijing, 1999.

[HY99b]Q. Hu and D. Yu. Solving singularity problems in unbounded domain by coupling of natural bem and composite grid fem. Technical report, Institute of Computational Mathematics and Scientific/Engineering Computing, Chinese Academy of Science, Beijing, 1999. to appear in App. Numer. Math.

[HZ94]G. Hsiao and S. Zhang. Optimal order multigrid methods for solving exterior boundary value problems. *Siam J. Numer. Anal.*, 31(2):680–694, 1994.

[JN80]C. Johnson and J. Nedelec. On the coupling of boundary integral and finite element methods. *Math. Comp.*, 35:1063–1079, 1980.

[McC89]Stephen F. McCormick. *Multilevel Adaptive Methods for Partial Differential Equations*. SIAM, Philadelphia, PA, 1989.

[Med98]S. Meddahi. An optimal iterative process for the johnson-nedelec method of coupling boundary and finite elements. *Siam J. Numer. Anal.*, 35:1393–1415, 1998.

[MT86]Steve McCormick and Jim Thomas. The fast adaptive composite grid (FAC) method for elliptic equations. *Math. Comp.*, 46(174):439–456, 1986.

[SBG96]Barry F. Smith, Petter E. Bjørstad, and William Gropp. *Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations*. Cambridge University Press, 1996.

[Xu92]Jinchao Xu. Iterative methods by space decomposition and subspace correction. *SIAM Review*, 34(4):581–613, December 1992.

[Yu93]D. Yu. *The mathematical theory of the natural boundary element methods.* Science Press, Beijing, 1993.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau, ©2001 DDM.org

15. Direct Mehtod of Lines for Solving an Elliptic Transmission Problem

Kiyoshi Kitahara [1], Hideyuki Koshigoe [2]

Introduction

The object of this paper is to present the numerical algorithm to obtain a finite difference solution for an elliptic transmission problem by use of the direct method of lines ([Nak65],[KK98], [KK99]). Let Π be a rectangular domain in $\mathbb{R}^2$, Ω_1 be an open subset of Π and $\Omega_2 = \Pi \setminus \overline{\Omega_1}$, $\Gamma = \partial\Omega_1$ (see Figure 1). Then the elliptic transmission problem is formulated as follows. And it is well known that (3) and (4) are called the conditions of transmission (cf. [DL90], [Lio71]).

Problem I Find $(u_1, u_2) \in H^1(\Omega_1) \times H^1(\Omega_2)$ such that

$$-\epsilon_1 \triangle u_1 = f_1 \quad \text{in} \quad \Omega_1 \ , \tag{1}$$

$$-\epsilon_2 \triangle u_2 = f_2 \quad \text{in} \quad \Omega_2 \ , \tag{2}$$

$$u_1 = u_2 \quad \text{on} \quad \Gamma \ , \tag{3}$$

$$\epsilon_1 \frac{\partial u_1}{\partial \nu} = \epsilon_2 \frac{\partial u_2}{\partial \nu} \quad \text{on} \quad \Gamma \ , \tag{4}$$

$$u_2 = g \quad \text{on} \quad \partial\Pi \ . \tag{5}$$

Hhere ϵ_1 and ϵ_2 are positive constants, $\{f_1, f_2\} \in L^2(\Omega_1) \times L^2(\Omega_2)$, $g \in H^{1/2}(\partial\Pi)$ and ν is the unit normal vector on Γ directed from Ω_1 to Ω_2 .

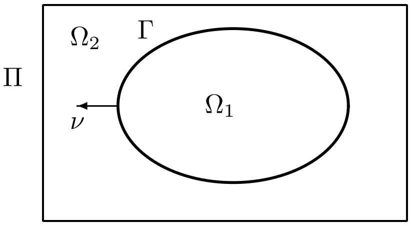


Figure 1: Interface Γ and unit normal ν

Equations (1)-(5) of this type are arisen in various contexts. One of such examples can be found in the context of electricity. In fact, let $\{\epsilon_1, \epsilon_2\}$ denote dielectric constant, $\{u_1, u_2\}$ be potential of the electric field and $\{f_1, f_2\}$ be charge density in the dielectric material $\{\Omega_1, \Omega_2\}$ respectively. Then the conditions (3) and (4) mean that the tangential component of the electric field and the normal component of electric

[1] Department of General Education, Kogakuin University, Tokyo, Japan. kitahara@cc.kogakuin.ac.jp

[2] Institute of Applied Mathematics, Chiba University, Chiba, Japan. koshigoe@applmath.tg.chiba-u.ac.jp

flux density are continuous across Γ respectively. Moreover if g=0, (5) represents that $\mathbb{R}^2 \setminus \Pi$ is occupied by a perfect conductor.

The problem of transmission type has been studied from the viewpoint of both theoretical and numerical researchs. And the method of the auxilliary domain plays the important role in the field of numerical analysis. In this paper we present another point of view to solve it numerically. That is to use the method of the successive eliminations of lines and to solve directly the kernel of the Steklov-Poincaré operator T , which is defined as the linear operator from the Dirichlet data on Γ to the Neumann data on Γ

$$T : H^{1/2}(\Gamma) \ni w \rightarrow \epsilon_1 \frac{\partial u_1}{\partial \nu} - \epsilon_2 \frac{\partial u_2}{\partial \nu} \in H^{-1/2}(\Gamma),$$

in the sense of the finite difference. We remark here that the discretized equations of Problem I is reduced to solve the linear system of equations defined on Γ (i.e., the kernel of the Steklov-Poincaré operator T) and another parts of unknowns are automatically decided by the algebraic computation using the explicit formula of the approximate solutions stated in the section 4.

Now considering the kernel of the Steklov-Poincaré operator T, Problem I is rewritten by Problem II. In fact, two formulations are equivalent by use of the distribution theoretical approach and Green's formula. Hence from now on, we consider the construction of the solution for Problem II in the sense of the finite difference.

Problem II Find $u \in H^1(\Pi)$ such that

$$\begin{cases} -\mathrm{div}\,(a(x)\,\nabla u) = f & \text{in} \quad D'(\Pi)\ , \\ u = g & \text{on} \quad \partial\Pi\ . \end{cases} \tag{6}$$

Here $a(x) = \epsilon_1\,\chi_{\Omega_1}(x) + \epsilon_2\,\chi_{\Omega_2}(x)$, $f(x) = f_1(x)\,\chi_{\Omega_1}(x) + f_2(x)\,\chi_{\Omega_2}(x)$ and $\chi_\Omega(x)$ is defined by

$$\chi_\Omega(x) = \begin{cases} 1 & if\ x \in \Omega \\ 0 & if\ x \notin \Omega \end{cases}$$

for any subset Ω of Π.

The contents of this paper are as follows. In the second section, we introduce a small perturbation on Γ for the numerical computation, which is defined by $\frac{1}{2}(\ f_1(x) + f_2(x)\ )\ \delta(x - \Gamma)$, in the discretized formulation of Problem II. Roughly speaking, it implies that $T(u_h) = O(h)$ for any mesh size h . In the third section, we prepare the representation formula of the solution for a system of linear equations. This is the background in the numerical algorithm we propose here. In the fourth section, the kernel of the Steklov-Poincaré operator T and the explicit formula of the approximate solutions will be presented using the results in the third section. In the fifth section, two numerical results will be shown by use of the explicit formula in the fourth section.

Finite difference approximation for Problem II

We partition the region Π into rectangles by vertical $m-1$ lines and horizontal $n-1$ lines. We denote mesh size for x direction as Δx and for y direction as Δy. Moreover by Γ_Δ we denote the set of all mesh points (which are interior of Π) such that from each point the horizontal distance to Γ is less than $\Delta x/2$ or the vertical distance to Γ is less then $\Delta y/2$ (see Figure 2). We designate the point in Γ_Δ as artifitial interface mesh point. By Π_Δ we denote the set of all interior mesh points which do not belong to Γ_Δ.

In order to denote a discretized model for Problem II we prepare some notations. We assume that the boundary data g is continuous on $\partial\Pi$ and the charge densities f_1, f_2 are coutinuous on $\overline{\Omega}_1$, $\overline{\Omega}_2$ respectively. Let denote u_{ij} as approximate value of the solution u at mesh point P_{ij}. We denote $P_{i+1/2,j}$ as the center of the points P_{ij} and $P_{i+1,j}$ and denote $P_{i,j+1/2}$ as the center of the points P_{ij} and $P_{i,j+1}$. For every mesh piont $P_{ij} \in \Gamma_\Delta$ we denote P^Γ_{ij} as the nearest point form P_{ij} among the points which are on the intersection of mesh lines and Γ (see Figure 3).

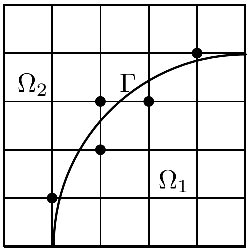


Figure 2: Mesh point near Γ

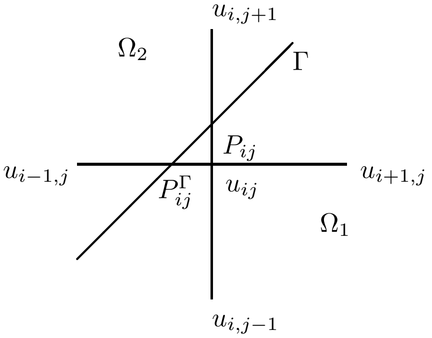


Figure 3: The point P^Γ_{ij}

Let us define the function $\epsilon(P)$ and the elements f_{ij} as the following :

$$\epsilon(P) = \begin{cases} \epsilon_1 & \text{if } P \in \Omega_1, \\ (\epsilon_1+\epsilon_2)/2 & \text{if } P \in \Gamma, \\ \epsilon_2 & \text{if } P \in \Omega_2, \end{cases} \qquad f_{ij} = \begin{cases} f(P_{ij}) & \text{if } P_{ij} \in \Pi_\Delta, \\ (f^1(P^\Gamma_{ij}) + f^2(P^\Gamma_{ij}))/2 & \text{if } P_{ij} \in \Gamma_\Delta. \end{cases}$$

By use of above notations we define a discretized model for Problem II at a mesh piont P_{ij} by the following form :

Discretized formula for Problem II at P_{ij}

$$\begin{aligned} -\frac{1}{\Delta x}&\left[\epsilon(P_{i+1/2,j})\frac{u_{i+1,j}-u_{ij}}{\Delta x} - \epsilon(P_{i-1/2,j})\frac{u_{ij}-u_{i-1,j}}{\Delta x}\right] \\ &- \frac{1}{\Delta y}\left[\epsilon(P_{i,j+1/2})\frac{u_{i,j+1}-u_{ij}}{\Delta y} - \epsilon(P_{i,j-1/2})\frac{u_{ij}-u_{i,j-1}}{\Delta y}\right] = f_{ij}. \end{aligned} \tag{7}$$

Now we put that

$$b^W_{ij} = \epsilon(P_{i-1/2,j}),\ b^E_{ij} = \epsilon(P_{i+1/2,j}),\ c^S_{ij} = \epsilon(P_{i,j-1/2}),\ c^N_{ij} = \epsilon(P_{i,j+1/2}), \tag{8}$$

then we can rewrite the equation (7) to the following form :

$$\begin{aligned}-tc^S_{ij}u_{i,j-1} + \big(b^W_{ij} + b^E_{ij} + t(c^S_{ij} + c^N_{ij})\big)u_{ij} - tc^N_{ij}u_{i,j+1}& \\ = b^W_{ij}\,u_{i-1,j} + b^E_{ij}u_{i+1,j} + (\Delta x)^2 f_{ij},&\end{aligned} \tag{9}$$

where $t = (\Delta x)^2/(\Delta y)^2$. The coefficients $b^W_{ij}, \dots, c^E_{ij}$ have the following properties.

$$b^E_{ij} = b^W_{i+1,j}, \quad c^N_{ij} = c^S_{i,j+1} \qquad \text{for all mesh points ,} \tag{10}$$

$$b^W_{ij} = b^E_{ij} = \epsilon(P_{ij}) = c^S_{ij} = c^N_{ij} \qquad \text{if } P_{ij} \in \Pi_\Delta. \tag{11}$$

These properties are obvious from definitions (8).

Now for $i = 1, 2, \dots, m-1$, we denote U_i as unknown columun vector $[u_{ij}]_{1\le j\le n-1}$ and define coefficient matrices A^ϵ_i, B^W_i and B^E_i by the following forms.

$$A^\epsilon_i = \begin{bmatrix} a^\epsilon_{i,1} & -tc^N_{i,1} & 0 & \cdots & \cdots \\ -tc^S_{i,2} & a^\epsilon_{i,2} & -tc^N_{i,2} & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \cdots & \cdots & 0 & -tc^S_{i,n-1} & a^\epsilon_{i,n-1} \end{bmatrix}, \tag{12}$$

$$B^W_i = \mathrm{diag}[b^W_{ij}]_{1\le j\le n-1}, \quad B^E_i = \mathrm{diag}[b^E_{ij}]_{1\le j\le n-1}, \tag{13}$$

where $a^\epsilon_{ij} = b^W_{ij} + b^E_{ij} + t(c^S_{ij} + c^N_{ij})$ and A^ϵ_i is a tridiagonal matrix.

By use of these notations we can rewrite equations (9) to the following system of equations which is a discretized model for Problem II .

Problem III Find vectors U_i $(1 \le i \le m-1)$ such that

$$A^\epsilon_i\, U_i = B^W_i\, U_{i-1} + B^E_i\, U_{i+1} + F_i \quad (1 \le i \le m-1), \tag{14}$$

where $U_0 = 0$, $U_m = 0$ and F_i $(1 \le i \le m-1)$ are known vectors constructed form the functions f and g.

From the equations (10) we know that A^ϵ_i $(1 \le i \le m-1)$ are symmetric matricies and $B^E_i = B^W_{i+1}$ $(1 \le i \le m-2)$.

Construction of the solution for linear equations based on the direct method of lines

Before proceeding to solve Problem III, we shall state our result about linear equations for the unkown matrix $\{X_i\}$ as follows:

$$A\, X_i = X_{i-1} + X_{i+1} + Y_i \quad (1 \le i \le m-1). \tag{15}$$

Here we make the following assumptions:

(H1) A is a square matrix of order N.

(H2) X_i $(0 \le i \le m)$ and Y_i $(1 \le i \le m-1)$ are $N \times M$ matrices which satisfy the system of equations (15).

Then we have the following representation for any one X_k.

Theorem 1 *We assume (H1),(H2). Then we have*

$$\begin{aligned} A_m\, X_k = A_{m-k}\, X_0 + A_{m-k} \sum_{i=1}^{k-1} A_i\, Y_i \\ + A_k \sum_{i=k}^{m-1} A_{m-i}\, Y_i + A_k\, X_m \qquad (1 \leq k \leq m-1) \end{aligned} \tag{16}$$

where the sequence of matrices $\{A_i\}$ *is defined by*

$$A_1 = I,\ A_2 = A,\ A_{i+1} = A\,A_i - A_{i-1} \quad (i = 2, 3, \dots). \tag{17}$$

If $A = [a_{ij}]$ is the $(n-1)-$ symmetrix tridiagonal matrix as follows:

$$a_{jj} = 2\,s \quad a_{j,j+1} = a_{j+1,j} = -t \quad \text{where} \quad s = 1 + t \tag{18}$$

then the representation (16) turns out to be a very simple form. In fact we can reduce A to a diagonal form by means of the orthgonal transfomation : $P = {}^t[P_1\,, P_2\,, \dots\,, P_{n-1}]$ where

$$p_{l,j} = \sqrt{\frac{2}{n}}\,\sin\left(\frac{l\,j\,\pi}{n}\right) \quad (1 \leq l, j \leq n-1). \tag{19}$$

and P has the following properties,

$${}^tP = P,\ P^2 = I. \tag{20}$$

Multiplying $P\,A_m^{-1}$ on the both sides of (16), we obtain the following result.

Proposition 1 *Assume* $X_0 = X_m = O$*, then we have*

$$P\,X_k = \sum_{i=1}^{k-1} D_{m-k,i}\,P\,Y_i + \sum_{i=k}^{m-1} D_{k,m-i}\,P\,Y_i \qquad (1 \leq k \leq m-1) \tag{21}$$

where for l *and* i *(*$1 \leq l, i \leq m-1$*)*

$$D_{l,i} = P\,A_m^{-1}\,A_l\,A_i\,P = \text{diag}\left[\frac{\sinh(l\,a_j)\,\sinh(i\,a_j)}{\sinh(m\,a_j)\,\sinh a_j}\right]_{1\leq j\leq n-1}. \tag{22}$$

Explicit formula of the solution for Problem III

We return to Problem III. We can rewrite the system of equations (14) to the following new system of equations, which is more useful forms for the Method of Lines, by use of splitting unknown vectors. Then our last problem is reduced to the following.

Problem IV Find $\{V_i,\ W_i\}$ $(1 \leq i \leq m-1)$ such that

$$A\,V_i = V_{i-1} + V_{i+1} + F_i + B_i^W\,W_{i-1} - A_i^\epsilon\,W_i + B_i^E\,W_{i+1} \qquad (1 \leq i \leq m-1) \tag{23}$$

where $V_0 = V_m = W_0 = W_m = 0$ and the matrix A is given by the equation (18).

Now we will derive the system of equations (23) from the system (9). At first, for any column vector $V(= [v_j])$, we define a set of indices as that $\mathrm{supp}(V) = \{j \mid v_j \neq 0\}$.

Let us divide each unknown vector U_i into two parts.

$$U_i = U_i' + W_i, \qquad (1 \leq i \leq m-1) \tag{24}$$

where

$$\mathrm{supp}(U_i') \subseteq \{j \mid P_{ij} \in \Pi_\Delta\} \quad \text{and} \quad \mathrm{supp}(W_i) \subseteq \{j \mid P_{ij} \in \Gamma_\Delta\}. \tag{25}$$

If $j \in \mathrm{supp}(U_i')$ then by use of the above definition (25) and the relation (11), we obtain the eqaution that $B_i^W U_i' = B_i^E U_i'$. Then we define the new unknown vectors V_i:

$$V_i = B_i^W U_i' = B_i^E U_i' \qquad (1 \leq i \leq m-1). \tag{26}$$

From the definition of V_i and W_i we have the following relations.

$$\mathrm{supp}(V_i) \cap \mathrm{supp}(W_i) = \emptyset \qquad (1 \leq i \leq m-1). \tag{27}$$

Moreover it follows from (10) that

$$V_i = B_{i-1}^E U_i' = B_{i+1}^W U_i' \qquad \text{and} \qquad A_i^\epsilon U_i' = A\, V_i \qquad (1 \leq i \leq m-1), \tag{28}$$

where the matrix A is given by the equation (18). By use of the relations (28) we can rewrite the system of equations (14) to the new system of equations (23).

Applying Proposition 1 to our difference equations (23) we obtain the following expressions.

Proposition 2 *For each number k ($1 \leq k \leq m-1$),*

$$\begin{aligned} P\,V_k = &\sum_{i=1}^{k-1} D_{m-k,i}\, P\left[B_i^W\, W_{i-1} - A_i^\epsilon\, W_i + B_i^E\, W_{i+1}\right] \\ &+ \sum_{i=k}^{m-1} D_{k,m-i}\, P\left[B_i^W\, W_{i-1} - A_i^\epsilon\, W_i + B_i^E\, W_{i+1}\right] \\ &+ \left(\sum_{i=1}^{k-1} D_{m-k,i}\, P\, F_i + \sum_{i=k}^{m-1} D_{k,m-i}\, P\, F_i\right) \end{aligned} \tag{29}$$

From the equations (29), we can elimimate V_k and get the linear equations for $\{W_i\}$ by use of the support property (27) and the orthogonarity of $\{P_l\}$.

Theorem 2 (Equations for W_i) *For $l \in supp(W_k)$,*

$$\begin{aligned} &\sum_{i=1}^{k-1} {}^tP_l\, D_{m-k,i}\, P\left[-B_i^W\, W_{i-1} + A_i^\epsilon\, W_i - B_i^E\, W_{i+1}\right] \\ &\quad + \sum_{i=k}^{m-1} {}^tP_l\, D_{k,m-i}\, P\left[-B_i^W\, W_{i-1} + A_i^\epsilon\, W_i - B_i^E\, W_{i+1}\right] \\ &\quad = {}^tP_l\left(\sum_{i=1}^{k-1} D_{m-k,i}\, P\, F_i + \sum_{i=k}^{m-1} D_{k,m-i}\, P\, F_i\right). \end{aligned} \tag{30}$$

By use of the orthogonarity of $\{P_l\}$, we get the following Theorem.

Theorem 3 (Expressions for $v_{i,j}$) *For $l \in supp(V_k)$,*

$$
\begin{aligned}
v_{k,l} = & \sum_{i=1}^{k-1} {}^tP_l\, D_{m-k,i}\, P \left[B_i^W\, W_{i-1} - A_i^\epsilon\, W_i + B_i^E\, W_{i+1}\right] \\
& + \sum_{i=k}^{m-1} {}^tP_l\, D_{k,m-i}\, P \left[B_i^W\, W_{i-1} - A_i^\epsilon\, W_i + B_i^E\, W_{i+1}\right] \\
& + {}^tP_l \left(\sum_{i=1}^{k-1} D_{m-k,i}\, P\, F_i + \sum_{i=k}^{m-1} D_{k,m-i}\, P\, F_i \right).
\end{aligned}
\tag{31}
$$

Examples

Using Theorem 2 and 3, we show the numerical results for the elliptic transmission problem (1) under the following geometry.

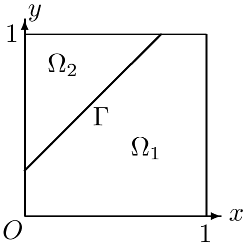


Figure 4: Example 1

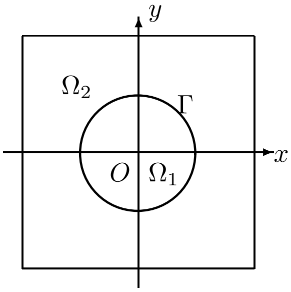


Figure 5: Example 2

Example 1:

Let $\Pi = (0,1) \times (0,1) = \Omega_1 \cup \Gamma \cup \Omega_2$, $\Gamma : x - y + 1/4 = 0$ as Figure 4. Set $\epsilon_1 = 1, \quad \epsilon_2 = 3$ in Problem I and $\Delta x = \Delta y = h$ in (7).

We then use test functions:

$$
u = \begin{cases} \sin(x - y + 1/4) + x + 1 & \text{in } \Omega_1, \\ (x - y + 1/4)^2 + x + 1 & \text{in } \Omega_2, \end{cases} \qquad f = \begin{cases} 2\,\epsilon_1 \sin(x - y + 1/4) & \text{in } \Omega_1, \\ -4\,\epsilon_2 & \text{in } \Omega_2. \end{cases}
$$

and get the Table 1 below.

Example 2:

Let $\Pi = (-0.5, -0.5) \times (0.5, 0.5) = \Omega_1 \cup \Gamma \cup \Omega_2$, $\Gamma : x^2 + y^2 = R^2$, $R = 1/4$ as Figure 5. Set $\epsilon_1 = 5, \quad \epsilon_2 = 3$ in Problem I and $\Delta x = \Delta y = h$ in (7).

We then use test functions:

$$u = \begin{cases} x^3 - y^3 & \text{in } \Omega_1, \\ (x^3 - y^3)(x^2 + y^2)/R^2 & \text{in } \Omega_2, \end{cases}$$

$$f = \begin{cases} -6\,\epsilon_1\,(x - y) & \text{in } \Omega_1, \\ -2\,\epsilon_2\,(11\,x^3 - 3\,x^2\,y + 3\,x\,y^2 - 11\,y^3)/R^2 & \text{in } \Omega_2. \end{cases}$$

and get the Table 2 below.

In the table 1 and 2, we use the following notations. The 'maximum point' (i, j) means the mesh point where the muximum error occurs and $\|u - u_h\|_\infty = \max_{i,j} |u(i\,h, j\,h) - u_{i,j}|$. Moreover 'ratio' means the percentage of the number of unknowns $\{w_{ij}\}$ for the system of linear equations in Theorem 2 to the total number of unknowns $\{u_{ij}\}$ in (7).

mesh size	ratio	maximum point	$\|u - u_h\|_\infty$
1/16	4.89%	(6, 10)	6.232701×10^{-5}
1/32	2.39%	(12, 20)	1.557993×10^{-5}
1/64	1.18%	(24, 40)	3.894824×10^{-6}
1/128	0.59%	(48, 80)	9.736942×10^{-7}

Table 1: Numerical result of example 1

mesh size	ratio	maximum point	$\|u - u_h\|_\infty$
1/16	10.67%	(4, 12)	7.426605×10^{-3}
1/32	4.58%	(9, 19)	2.119219×10^{-3}
1/64	2.32%	(28, 47)	8.459878×10^{-4}
1/128	1.12%	(58, 95)	4.221186×10^{-4}

Table 2: Numerical result of example 2

Remark

Both examples show that the 'ratio' is decreasing in proportion to mesh size. Hence our method seems to be advantageous in the situation where the mesh size is very small.

References

[DL90]R. Dautray and J. L. Lions. *Mathematical analysis and numerical methods for science and technology.* Springer-Verlag, 1990.

[KK98]K. Kitahara and H. Koshigoe. Method of lines coupled with domain decompositions and its application. *The institute of statistical mathematics-Cooperative research report*, 110:132–139, 1998.

[KK99]H. Koshigoe and K. Kitahara. Method of lines coupled with fictitious domain for solving Poisson's equation. *Gakuto international series, Mathematical Sciences and Applications*, 12:233–242, 1999.

[Lio71]J. L. Lions. *Optimal control systems governed by partial differential equations.* Springer-Verlarg, 1971.

[Nak65]K. Nakashima. Numerical computation of elliptic partial differential equations I, method of lines. *Memoirs of the school of science and engineering, Waseda Univ.*, 1965.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau, ©2001 DDM.org

16. Finite Difference Method with Fictitious Domain Applied to a Dirichlet Problem

Hideyuki Koshigoe [1] Kiyoshi Kitahara [2],

Introduction

In this paper we shall consider the construction of the solution by the method of lines coupled with a fictitious domain for the following Dirichlet problem (1) in a bounded domain Ω of $\mathbb{R}^2$.

Problem I. For given functions f and g, find u in $H^2(\Omega)$ such that

$$\begin{cases} -\Delta u = f & \text{in } \Omega\ , \\ \quad\ \ u = g & \text{on } \partial\Omega\ . \end{cases} \tag{1}$$

Here $f \in L^2(\Omega)$, $g \in H^{3/2}(\partial\Omega)$ and Ω is a bounded domain in $\mathbb{R}^2$ with the smooth boundary $\partial\Omega$ (see Figure 1).

The method of lines for solving Problem I works well if Ω is a rectangular domain since the finite difference solution is expressed explicitly by use of eigenvalues and eigenvectors for the finite difference scheme([BGN70], [Nak65]). But one says that this method seems difficult to be applied to the case where Ω is not a rectangular domain. However the solution algorithm using the fictitious domain and the domain decomposition has been developed recently ([AKP95], [GPP94], [HH99], [FKK95], [KK99], [MKM86]). Hence from this point of view we shall propose a numerical algorithm by the method of lines coupled with a fictitious domain in this paper.

First of all, we embed Ω in a rectangular domain Π whose boundary $\partial\Pi$ consists of straight lines parallel to axes and set $\Omega_1 = \Pi \setminus (\Omega \cup \partial\Omega)$ (see Figure 2). Then Π is called a fictitious domain.

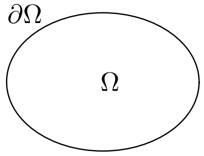


Figure 1: Figure 1

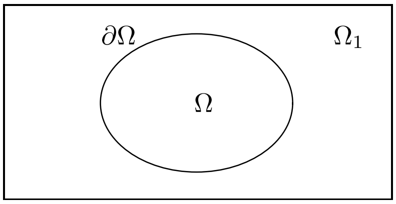


Figure 2: ($\Pi = \overline{\Omega} \cup \Omega_1$)

Hereafter we shall construct the numerical algorithm for solving Poisson's equation (1) in the fictitious domain Π. In §2, Problem I is reduced to a fictitious domain

[1] Institute of Applied Mathematics, Chiba University, Chiba, Japan. koshigoe@applmath.tg.chiba-u.ac.jp

[2] Department of General Education, Kogakuin University, Tokyo, Japan. kitahara@cc.kogakuin.ac.jp

formulation by use of the distribution theoretical approach. In §3, we shall discuss characterizations of the solution for the fictitious domain formulation. In §4, a numerical algorithm of the direct method of lines will be proposed and the results of numerical computations will be shown.

A fictitious domain formulation of Problem I.

Using the trace operator γ in Sobolev space and distribution theoretical argument, we deduce a fictitious domain formulation from Problem I. It is well-known that there exists a function $G \in H^2(\Omega)$ such that $\gamma G = g$ on $\partial\Omega$ because of $g \in H^{3/2}(\partial\Omega)$. Then putting $u = v + G$, Problem I is reduced to

Problem II. Find $v \in H^2(\Omega)$ such that

$$\begin{cases} -\Delta v &= f + \Delta G \quad \text{in } \Omega, \\ \quad v &= 0 \quad \text{on } \partial\Omega. \end{cases} \tag{2}$$

Remark 1 *Set $u = v + G$ where v is a solution of Problem II. Then u is a unique solution of Problem I. In this case, it is important to be independent of a choice of G in Problem II (see p.232 in [Miz73]). And this fact will be used in §4.*

We now define a function $\widetilde{v}$ as follows: for any function $v \in L^2(\Omega)$,

$$\begin{cases} \widetilde{v}(x) &= v(x) \quad (x \in \Omega) \\ \widetilde{v}(x) &= 0 \quad (x \in \mathbb{R}^2 \backslash \Omega). \end{cases} \tag{3}$$

Then for $v \in H_0^1(\Omega)$, $\widetilde{v}$ belongs to $H^1(\mathbb{R}^2)$ and the equality

$$\frac{\partial}{\partial x_i}\widetilde{v} = \widetilde{\frac{\partial v}{\partial x_i}} \tag{4}$$

holds(see p.187-189 in [Miz73]). Moreover operating Δ to $\widetilde{v}$, we have the following lemma which was shown by Kawarada([Kaw89]).

Lemma 1 *Let $v \in H^2(\Omega) \cap H_0^1(\Omega)$. Then*

$$\widetilde{\Delta v} = \Delta\widetilde{v} + \frac{\partial v}{\partial n} \cdot \delta(\partial\Omega) \quad \textit{in the sense of distribution in } \mathbb{R}^2, \tag{5}$$

holds where n is the unit normal vector at $\partial\Omega$, directed towards the outer of Ω and $\delta(\partial\Omega)$ means the delta measure supported on $\partial\Omega$.

By use of (3)-(5) and the definition of the weak derivative in the sense of the distribution, we have

Theorem 1 *Problem II is equivalent to the following Problem III. i.e.,*

Problem III. *Find $\widetilde{v} \in H_0^1(\Pi)$ and $w \in L^2(\partial\Omega)$ such that*

$$-\Delta\widetilde{v} = \widetilde{f + \Delta G} + w\ \delta(\partial\Omega) \textit{ in } D'(\Pi) \tag{6}$$

Remark 2 $\widetilde{v} \in H_0^1(\Pi)$ *means that* $v \in H_0^1(\Omega)$ *and* $v \equiv 0$ *in* Ω_1.

Corollary 1 *The solution* $\{\widetilde{v}, w\}$ *of (6) has the following relation:*

$$w = \frac{\partial v}{\partial n} \quad on \quad \partial\Omega.$$

Remark 3 *We call (6) a fictitious domain formulation of Problem II and this formulation is essential for our discussions. It will be used to construct the finite difference solution of Problem I in* §4.

Characterization of the solution of the fictitious domain formulation (6)

Before proceeding to the construction of the finite difference scheme under the fictitious domain formulation, we state the relation between (6) and the auxiliary domain method ([Lio73]).

Proposition 1 *The following statements are equivalent to each other.*

(i) There exists a unique solution $v \in H^2(\Omega)$ *satisfying Problem II. And set* $w = \frac{\partial v}{\partial n}$.

(ii) There exists a solution $\{\widetilde{v}, w\} \in H_0^1(\Pi) \times L^2(\partial\Omega)$ *in Problem III which satisfies*

$$-\Delta\widetilde{v} \;=\; \widetilde{f + \Delta G} + w\ \delta(\partial\Omega) \ in \ D'(\Pi).$$

(iii) There exists a solution $\{v_0, v_1, w\} \in H_0^1(\Omega) \times H_0^1(\Omega_1) \times L^2(\partial\Omega)$ *such that*

$$\begin{cases} -\Delta v_0 = \ f + \Delta G \qquad in \ \ \Omega \ , \\ -\Delta v_1 = \ 0 \qquad in \ \ \Omega_1 \ , \\ \ v_0 = v_1 = 0 \qquad on \ \ \partial\Omega \ , \\ \ \dfrac{\partial v_0}{\partial n} = w \qquad on \ \ \partial\Omega \ , \\ \ v_1 \ = \ 0 \qquad on \ \ \partial\Pi. \end{cases}$$

(iv) There exists a solution $\{v, w\} \in V \times L^2(\partial\Omega)$ *satisfying*

$$\int_\Pi \nabla v \cdot \nabla\varphi \ dx = \int_\Omega (f + \Delta G)\varphi + \int_{\partial\Omega} \omega\varphi \ d\Gamma \quad for\ any\ \varphi \in V \tag{7}$$

where $V = H^1(\Pi) \cap H_0^1(\Omega_1)$.

Remark 4 ω *in the form (7) is usually called a Lagrange multiplier .*

Proposition 2 *The solution* $\widetilde{v}$ *of Problem III with* $g = 0$ *is the limit function of the approximate solutions* $\{v_0^\epsilon, v_1^\epsilon\}$ *as* $\epsilon \to 0$*:*

$$\begin{cases} -\Delta v_0^\epsilon = \ f \qquad in \ \ \Omega \ , \\ -\epsilon^{2\alpha}\Delta v_1^\epsilon + \epsilon^{-2\beta} v_1^\epsilon = \ 0 \qquad in \ \ \Omega_1 \ , \\ \ v_0^\epsilon = v_1^\epsilon \qquad on \ \ \partial\Omega \ , \\ \ \dfrac{\partial v_0^\epsilon}{\partial n} = \epsilon^{2\alpha}\dfrac{\partial v_1^\epsilon}{\partial n} \qquad on \ \ \partial\Omega \ , \\ \ v_1^\epsilon \ = \ 0 \qquad on \ \ \partial\Pi \end{cases}$$

for any α, β *satisfying* $0 < \alpha < \beta$.

Proof In fact, it is known that for α and β $(0 < \alpha < \beta)$,
$v_0^\epsilon \to v_0$ in $H^1(\Omega)$, $v_1^\epsilon \to 0$ in $L^2(\Omega_1)$, and v_0 is the solution of Problem II (see Theorem 10.1, pp. 78-82 in [Lio73]). Hence setting $w = \frac{\partial v_0}{\partial n}$, $\widetilde{v_0}$ is exactly the solution of Problem III by (iii) of Propostion 1. ■

Numerical Algorithm of the fictitious domain formulation (6) by use of the direct method of lines

In this section, we shall propose a numerical algorithm of the direct method of lines by use of the fictitious domain formulation (6).

Discretization of the fictitious domain formulation (6)

We first assume the fictitious domain Π given by

$$\Pi = \{ \ (x, y) \mid 0 < x < 1, \ 0 < y < 1 \ \}, \tag{8}$$

which consists of $\overline{\Omega}$ and Ω_1 where $\Omega_1 = \Pi \setminus \overline{\Omega}$ (see Figure 2).

While the set of grid points, $\overline{\Pi_h}$, is of the form

$$\overline{\Pi_h} = \{ \ (x_i, y_j) \mid 0 \le i \le m, \ 0 \le j \le m \ \},$$

here $x_i = ih, y_j = jh$ for a suitable spacing $h = 1/m$ and $P(i,j) = (x_i, y_j)$.

With each grid point (x_i, y_j) of Π_h, we associate the cross line with center (x_i, y_j):

$$M\Big((x_i, y_j)\Big) = \{ \ (x_i + s, y_j), s \in (-\frac{h}{2}, \frac{h}{2})\} \cup \{ \ (x_i, y_j + s), s \in (-\frac{h}{2}, \frac{h}{2})\}.$$

We then define

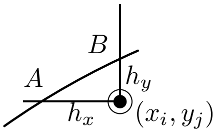
$$\begin{aligned}
\Omega_h^0 &= \{ \ (x_i, y_j) \ : \ (x_i, y_j) \in \overline{\Pi_h}, \ M((x_i, y_j)) \subset \Omega \ \},\\
\Omega_h^1 &= \{ \ (x_i, y_j) \ : \ (x_i, y_j) \in \overline{\Pi_h}, \ M((x_i, y_j)) \subset \Omega_1 \ \},\\
\partial\Omega_h^0 &= \{ \ (x_i, y_j) \ : \ (x_i, y_j) \in \overline{\Pi_h}, M((x_i, y_j)) \cap \partial\Omega \neq \phi \ \},\\
\partial\Omega_h^1 &= \{ \ (x_i, y_j) \ : \ (x_i, y_j) \in \overline{\Pi_h}, \ M((x_i, y_j)) \cap \partial\Omega_1 \neq \phi \ \}.
\end{aligned}$$

We then define for each i, j $(0 \le i, j \le m)$,

$$F_{i,j} = F(x_i, y_j) = \begin{cases} f(x_i, y_j) & \text{for } P(i,j) \in \Omega_h^0 \ , \\ 0 & \text{otherwise} \ . \end{cases}$$

$$G_{i,j} = G(x_i, y_j) = \begin{cases} \overline{g}(x_i, y_j) & \text{for } P(i,j) \in \partial\Omega_h^0 \\ 0 & \text{otherwise} \ . \end{cases}$$

Here $\overline{g}(x_i, y_j)$ is defined as follows:



$$\overline{g}(x_i, y_j) = \begin{cases} g(x_i, y_j) & \text{if } h_x = 0 \text{ or } h_y = 0, \\ g(A) & \text{if } 0 < h_x \le h/2 \text{ and } h_y > h/2, \\ g(B) & \text{if } h_x > h/2 \text{ and } 0 < h_y \le h/2, \\ \dfrac{g(A) \times h_y + g(B) \times h_x}{h_x + h_y} & \text{if } 0 < h_x, h_y < h/2. \end{cases}$$

Then the finite difference approximation of (6) can be formulated as follows.

Find $v(= \{v_{i,j}\})$ and $w(= \{w_{i,j}\})$ such that

$$-(\Delta_h v)(x_i, y_j) = F_{i,j} + (\Delta_h G)(x_i, y_j) + \frac{\sqrt{2}}{h} w_{i,j}\ \delta(P_{i,j}) \text{ for all } (x_i, y_j) \in \Pi_h \qquad (9)$$

where $\delta(P_{i,j}) = 1$ if $P_{i,j} \in \partial\Omega_h^0, \delta(P_{i,j}) = 0$ if $P_{i,j} \notin \partial\Omega_h^0$ and the finite difference operator $-\Delta_h$, approximating the Laplace operator $-\Delta$ is of the form

$$-(\Delta_h v)(x_i, y_j) = \frac{1}{h^2}[v_{i+1,j} + v_{i-1,j} + v_{i,j+1} + v_{i,j-1} - 4v_{i,j}]$$

and $v_{i,j} = v(x_i, y_j)$ as usual.

Theorem 2 *There exists a unique solution* $\{v_{i,j}\}$ *and* $\{w_{i,j}\}$ *of (9).*

Proof In fact, (9) is rewritten as follows. Find $v(= \{v_{i,j}\})$ and $w(= \{w_{i,j}\})$ such that

$$-(\Delta_h v)(x_i, y_j) = F_{i,j} + (\Delta_h G)(x_i, y_j) \text{ for all } (x_i, y_j) \in \Omega_h^0 \qquad (10)$$

$$v(x_i, y_j) = 0 \quad \text{for all } (x_i, y_j) \in \partial\Omega_h^0, \qquad (11)$$

$$-(\Delta_h v)(x_i, y_j) = 0 \text{ for all } (x_i, y_j) \in \Omega_h^1 \qquad (12)$$

$$v(x_i, y_j) = 0 \quad \text{for all } (x_i, y_j) \in \partial\Omega_h^1, \qquad (13)$$

$$-(\Delta_h v)(x_i, y_j) = \frac{\sqrt{2}}{h} w_{i,j}\ \delta(P_{i,m+j}) \text{ for all } (x_i, y_j) \in \partial\Omega_h^0. \qquad (14)$$

Then it follwos from these forms and the standard arguments in the finite difference method ([Joh67]) that the Dirichlet problems (10)- (11) and (12)-(13) have unique solutions $v_{i,j}$ on $\Pi_h \backslash \partial\Omega_h^0$, from which and (14) , $v_{i,j}$ on $\partial\Omega_h^0$ are uniquely determined. ■

Now in order to give the matrix expression of (9), we shall introduce the following notations. For each $i(1 \leq i \leq m-1)$,

$$V_i = (v_{i,1}, \cdots, v_{i,m-1})^T, \tag{15}$$

$$W_i = (\xi_{i,1}, \cdots, \xi_{i,m-1})^T, \tag{16}$$

$$Z_i = h^2 \ (F_{i,1} + (\Delta_h)G(x_i, y_1), \cdots, F_{i,m-1} + (\Delta_h)G(x_i, y_{m-1}) \ )^T \tag{17}$$

Here we set $\xi_{i,j} = \sqrt{2} \ h \ w_{i,j} \quad (1 \leq i, j \leq m-1)$.

Remark 5 *If $P(i,j) \in \partial\Omega_h^0$, then $v_{i,j} = 0$ and $w_{i,j} \neq 0$. If $P(i,j) \notin \partial\Omega_h^0$, then $w_{i,j} = 0$.*

We introduce the concept of the support of vectors which is used in our numerical algorithm.

Definition 1. *The support for an (m-1)-vector $V_i(= \{v_{i,j}\})$ is defined by*

$$supp(V_i) = \{j \mid v_{i,j} \neq 0\}.$$

Then Remark 5 shows that $supp(V_i) \cap supp(W_i) = \phi$ and $\xi_{i,j} = 0$ if $j \notin supp(W_i)$.

Using the above notations, the discrete equation for (6) is to

find $\{V_i \ , \ W_i\} \ (1 \leq i \leq m-1)$ such that

$$A \ V_i = \ V_{i-1} + V_{i+1} + W_i + Z_i \quad (1 \leq i \leq m-1) \tag{18}$$

where $V_0 = 0$, $V_m = 0$ and A is $(m-1) \times (m-1)$ matrix as follows;

$$A = \begin{pmatrix} 4 & -1 & & & \\ -1 & 4 & \cdot & & \\ & \cdot & \cdot & \cdot & \\ & & \cdot & \cdot & -1 \\ & & & -1 & 4 \end{pmatrix}. \tag{19}$$

Numerical algorithm by use of the direct method of lines

From now on we shall construct a numerical algorithm for (18). Using successive eliminations by lines, we have

Theorem 3 *For each k $(1 \leq k \leq m-1)$,*

$$PV_k = \sum_{i=1}^{k-1} D_m^{-1} D_{m-k} D_i P W_i + \sum_{i=k}^{m-1} D_m^{-1} D_k D_{m-i} P W_i + G_k \tag{20}$$

holds where $G_k = \sum_{i=1}^{k-1} \ D_m^{-1} D_{m-k} D_i P Z_i + \ \sum_{i=k}^{m-1} \ D_m^{-1} D_k D_{m-i} P Z_i$

and the diagonal matrix D_k and the orthogonal matrix P are determined by

$$D_k = \begin{pmatrix} a_1^k & & & \mathbf{0} \\ & a_2^k & & \\ & & \ddots & \\ \mathbf{0} & & & a_{m-1}^k \end{pmatrix}$$

where the elements a_j^k $(1 \le j \le m-1)$ are determined exactly by

$$a_j^k = \frac{\sinh(ka_j)}{\sinh(a_j)}, \quad a_j = \cosh^{-1}(\frac{\lambda_j}{2}), \quad \lambda_j = 2\Big(2 - \cos(\frac{j}{m}\pi)\Big). \tag{21}$$

and the orthogonal matrix $P = [p_1, p_2, \cdots, p_{m-1}]$ consists of

$$p_j = \sqrt{\frac{2}{m}} \begin{pmatrix} \sin(\frac{j}{m}\pi) \\ \sin(\frac{2j}{m}\pi) \\ \cdot \\ \cdot \\ \cdot \\ \sin(\frac{(m-1)j}{m}\pi) \end{pmatrix} \qquad (1 \le j \le m-1). \tag{22}$$

Remark 6 *By use of the property of the orthogonal matrix P, PV_i and PW_i are expressed as follows:*

$$PV_i = \sum_{j \in supp(V_i)} v_{i,j}\ p_j, \quad PW_i = \sum_{j \in supp(W_i)} w_{i,j}\ p_j. \tag{23}$$

Finally we propose a numerical algorithm which is deduced from Theorem 3 and (23).

Numerical algorithm:

(1st step) Calculate $\{\xi_{i,j}\}(j \in supp(W_i),\ 1 \le i \le m-1)$ such that

$$\sum_{i=1}^{k-1} \sum_{j \in supp(W_i)} \Big(p_l \bullet D_m^{-1} D_{m-k} D_i\ p_j\Big)\ \xi_{i,j}\ +\ \sum_{i=k}^{m-1} \sum_{j \in supp(W_i)} \Big(p_l \bullet D_m^{-1} D_k D_{m-i}\ p_j\Big)\ \xi_{i,j}$$
$$=\ -p_l \bullet G_k \text{ for all } l\ \in\ supp(W_k).$$

(2nd step) Compute $\{v_{k,l}\}$ $(l \in supp(V_k), 1 \le k \le m-1)$ by

$$v_{k,l} = \sum_{i=1}^{k-1} \sum_{j \in supp(W_i)} \Big(p_l \bullet D_m^{-1} D_{m-k} D_i\ p_j\Big)\ \xi_{i,j}\ +\ \sum_{i=k}^{m-1} \sum_{j \in supp(W_i)} \Big(p_l \bullet D_m^{-1} D_k D_{m-i}\ p_j\Big)\ \xi_{i,j}$$
$$+\ p_l \bullet G_k \quad \text{for all} \quad l\ \in supp(V_k).$$

Here $\bullet$ means the inner product in $\mathbb{R}^{m-1}$.

Remark 7 *This is a generalization of the corresponding one in [KK99].*

Numerical experiments

Using the numerical algorithm of Theorem 3, we consider the Dirichlet problem:

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

Here

$U(x,y)$=sinh$(\pi x/2)$ sin$(\pi y/2)$ and $\Omega = \{(x,y) \mid \frac{(x-1/2)^2}{(1/4)^2} + \frac{(y-1/2)^2}{(1/8)^2} \ < \ 1\}$ that is the same geometry as one in [GPP94].

Then we get the following table of the choice of different mesh interval dh as for the maximum error (MaxEr) and the average error (AvEr) where $MaxEr = max\{|U(ih,jh){-}U_{i,j}|\ ;\ P(i,j) \in \Omega_h^0\}$ and $AvEr = \sum_{i,j=1}^{m-1}\{\ |U(ih,jh){-}U_{i,j}|;\ P(i,j) \in \Omega_h^0\}/N_h$ (N_h : the total number of the mesh points in Ω_h^0).

$dh = 1/n$	$MaxEr$	$AvEr$
$n = 16$	9.430569×10^{-3}	3.588982×10^{-3}
$n = 32$	5.492716×10^{-3}	1.065969×10^{-3}
$n = 64$	5.258107×10^{-3}	6.026563×10^{-4}
$n = 128$	2.969270×10^{-3}	3.058067×10^{-4}

Concluding remarks

We have presented the numerical algorithm of the direct method of lines coupled with the fictitious domain. This method which use the regular mesh is very simple and easy to perform the calculation, and yet the above maximum errors are same as one in the standard framework of the finite difference method in nonrectangular domain ([Joh67]). Therefore this argument shows that the finite difference method under the regular mesh is able to be applied to the case of general domains with the help of the fictitious domain.

Acknowledgment

The authors wish to thank Professor Hideo Kawarada, Chiba University, for his encouragement and useful comments.

References

[AKP95]Y. Achdou, Y.A. Kuznetsov, and O. Pironneau. Substructuring preconditioners for the q_1 mortar element method. *Numer.Math.*, 71:419–449, 1995.

[BGN70]B.L. Buzbee, G.H. Golub, and C.W. Nielson. On direct methods for solving Poisson's equations. *SIAM J.Numer.Anal.*, 7(4):627–656, 1970.

[FKK95]H. Fujita, H. Kawahara, and H. Kawarada. Distribution theoretical approach to fictitious domain method for Neumann problems. *East-West J.Math.*, 3(2):111–126, 1995.

[GPP94]R. Glowinski, T.W. Pan, and J. Periaux. A fictitious domain method for Dirichlet problem and applications. *Computer Methods in Applied Mechanics and Engineering*, 111:283–303, 1994.

[HH99]H. Han and Z. Huang. The direct method of lines for the numerical solutions of interface problem. *Comput. Meth. Appl. Mech. Engrg.*, 171(1-2):61–75, March 1999.

[Joh67]F. John. *Lectures on Advanced Numerical Analysis.* Gordon and Breach Science, 1967.

[Kaw89]H. Kawarada. *Free boundary problem-theory and numerical method.* Tokyo University Press, 1989.

[KK99]H. Koshigoe and K. Kitahara. Method of lines coupled with fictitious domain for solving Poisson's equation. *Gakuto international series, Mathematical Sciences and Applications*, 12:233–242, 1999.

[Lio73]J.L. Lions. *Perturbations singuliéres dans les problémes aux limites et en controle optimal*, volume 323 of *Lecture Notes in Mathematics.* Springer-Verlarg, 1973.

[Miz73]S. Mizohata. *The theory of partial differential equations.* Cambridge University Press, 1973.

[MKM86]G.I. Marchuk, Y.A. Kuznetsov, and A.M. Matsokin. Fictitious domain and domain decomposition methods. *Sov.J.Numer.Anal.Math.Modelling*, 1(1):3–35, 1986.

[Nak65]K. Nakashima. Numerical computation of elliptic partial differential equations I, method of lines. *Memoirs of the school of science and engineering, Waseda Univ.*, 1965.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau, ©2001 DDM.org

17. New Interface Conditions for Non-overlapping Domain Decomposition Iterative Procedures

Ohin Kwon[1] Dongwoo Sheen[2]

Introduction

A Seidel-type interface condition is considered for non-overlapping domain decomposition iterative methods. With a suitable pseudo-energy defined on interfaces, the convergence speed of the iterative scheme is shown to be as twice fast as that of the Jacobi scheme. Our analysis is entirely independent of the governing model problems of a specific type of partial differential equations, but depends only on the scheme of updating interface data. By this, our analysis covers Seidel-type schemes for a general class of problems, such as elliptic, Helmholtz, Maxwell, and elasticity problems, etc. In order to avoid the sequential nature of Seidel schemes and to implement them on parallel computers, *red-black Gauss-Seidel* schemes are also considered with equivalent efficiency to Seidel schemes.

Concerning domain decomposition iterative methods, P.-L. Lions [Lio88, Lio90] investigated the convergence properties by taking a suitable pseudo-energy with which he was able to show iterative solutions converge. This idea has been applied to a more difficult Helmholtz problem by Després [Des91, BD97]. An improved variant of Lions's method is proposed by Q. Deng and its convergence is analyzed in the Sobolev H^1 norm [Den97]. Exploiting the structure of mixed finite element, Douglas et al. obtained a more precise convergence rate by a spectral radius estimation of the iterative solution operator [DPRW93]. More efficient iterative schemes, such as Seidel-type and under-relaxation type domain decomposition iterative methods for elliptic, Helmholtz and electromagnetic problems have been considered in [CGJ98, CDJP97, DM97, Fen97, Gha97], and Seidel-type approaches based on nonconforming finite elements [DSSY99] were used in [HKS99, Kwo99, KS99] with estimations of spectral radii obtained. In this paper we show that the Seidel-schemes are exactly twice faster than the corresponding Jacobi-schemes.

[1] Department of Mathematics, Seoul National University, Seoul 151–742, Korea; E-mail: oikwon@math.snu.ac.kr

[2] Department of Mathematics, Seoul National University, Seoul 151–742, Korea; E-mail: sheen@math.snu.ac.kr, http://www.math.snu.ac.kr/~ sheen; This work was partially supported by Lotte Fellowship, and KOSEF 97-0701-01-01-3.

Domain decomposition iterative procedure

A model problem

Let Ω be a domain in $\mathbf{R}^N, N = 2, 3$, with the boundary $\Gamma = \partial\Omega$. Let us first consider the following model problem:

$$-\nabla \cdot (A\nabla u) + Bu = f \quad \text{in } \Omega,\ \nu \cdot A\nabla u + \alpha u = g \quad \text{on } \Gamma, \tag{1}$$

where ν is the unit outward normal vector to $\partial\Omega$. The coefficients $A = A(x), B = B(x) = B_R + iB_I$, and $\alpha = \alpha(x) = \alpha_R + i\alpha_I$ are assumed to satisfy

$$\begin{gathered} 0 < A_0|\xi|^2 \le A_{jk}(x)\xi_k\bar{\xi_j} \le A_1|\xi|^2 < \infty, \\ |B(x)| < B_1 < \infty, \qquad |\alpha(x)| < B_2 < \infty. \end{gathered}$$

Notice that (1) covers the case of Helmholtz equation and (1) may be regarded as a general form of first-order absorbing boundary condition.

Non-overlapping domain decomposition iterative procedure

Let $\{\Omega_j : j = 1, \cdots, J\}$ be a non-overlapping decomposition of Ω such that

$$\bar{\Omega} = \cup_{j=1}^J \bar{\Omega}_j,\ \Omega_j \cap \Omega_k = \emptyset,\ j \ne k,$$

and set

$$\Gamma_j = \partial\Omega \cap \partial\Omega_j, \quad \Gamma_{jk} = \Gamma_{kj} = \partial\Omega_j \cap \partial\Omega_k.$$

Denote by $v_j := v|_{\Omega_j}$ the restriction of a function v to Ω_j for all j, and set

$$\begin{gathered} V_j = H^1(\Omega_j)\ \forall j; \quad V = \{v|\ v|_{\Omega_j} \in V_j,\ \forall j\}; \\ \Lambda = \{w \mid w|_{\Gamma_{jk}} = Tr_{\Gamma_{jk}}(w_j) \in H^{-1/2}(\Gamma_{jk})\ \forall k\ \forall j\}, \end{gathered}$$

where $H^s(\Omega), H^s(\Omega_j), s \in \mathbf{R}$, are the usual complex-valued Sobolev spaces and $Tr_{\Gamma_{jk}}$ is the trace operator to Γ_{jk}.

Then the *domain decomposition iterative procedure* for solving (1) is as follows.

1. **Initialization Step.** An initial approximation $u^0 \in V$.

2. **Iterative Step.** For $n = 1, 2, \cdots$, solve iteratively the subdomain problems for $u_j^n, j = 1, \cdots, J$:

$$\begin{aligned} -\nabla \cdot (A\nabla u_j^n) + Bu_j^n &= f_j \quad \text{in } \Omega_j, && (2)\\ \nu_j \cdot A\nabla u_j^n + \alpha u_j^n &= g_j \quad \text{on } \Gamma_j, && (3) \end{aligned}$$

with the interface conditions

$$\nu_j \cdot A\nabla u_j^n + \beta u_j^n = -\nu_k \cdot A\nabla u_k^{n-1} + \beta u_k^{n-1} \quad \text{on } \Gamma_{jk},\ \forall k, \tag{4}$$

where ν_j is the unit outward normal vector to $\partial\Omega_j$, and β is a matching parameter such that $\beta|_{\Gamma_{jk}} = \beta|_{\Gamma_{kj}}\ \forall k\,\forall j$.

The weak problem for (2) is then to find $u^n \in V$ such that

$$a_j(u_j^n, \varphi) + \sum_k \langle \beta u_j^n, \varphi \rangle_{\Gamma_{jk}} = F_j(\varphi) + \sum_k \langle -\nu_k \cdot A\nabla u_k^{n-1} + \beta u_k^{n-1}, \varphi \rangle_{\Gamma_{kj}}, \quad \varphi \in V_j, \tag{5}$$

where

$$\begin{aligned} a_j(u_j, \varphi) &:= (A\nabla u_j, \nabla\varphi)_j + \big(Bu_j, \varphi\big)_j + \langle \alpha u_j, \varphi \rangle_{\Gamma_j}, \\ F_j(\varphi) &:= (f_j, \varphi) + \langle g_j, \varphi \rangle_{\Gamma_j}, \end{aligned}$$

with $(\cdot,\cdot)_j$ and $\langle \cdot,\cdot \rangle_{\Gamma_{jk}}$ being the $L^2(\Omega_j)$ and $L^2(\Gamma_{jk})$ inner products, respectively.

For each n, denote by $\lambda^n \in \Lambda$ the oblique normal traces:

$$\lambda_{jk}^n := \nu_j \cdot A\nabla u_j^n, \quad \Gamma_{jk}\, \forall k.$$

Then the interface condition (4) can be equivalently written in the form

$$\lambda_{jk}^n + \beta u_j^n = -\lambda_{kj}^{n-1} + \beta u_k^{n-1}, \quad \Gamma_{jk}\, \forall k, \tag{6}$$

and the weak formulation (5) takes the form

$$a_j(u_j^n, \varphi) + \sum_k \langle \beta u_j^n, \varphi \rangle_{\Gamma_{jk}} = F_j(\varphi) + \sum_k \langle \beta u_k^{n-1} + \beta u_k^{n-1}, \varphi \rangle_{\Gamma_{jk}}, \quad \varphi \in V_j. \tag{7}$$

Each **Iterative Step** consists of the following two substeps:
Substep 2a. Solve the subdomain problems (7) for $u^n \in V$;
Substep 2b. Update $\lambda^n \in \Lambda$ by (6).

The updating procedure (6) may be regarded as a *Jacobi-type scheme* with which subdomain problems (7) for all j can be easily parallelizable. Can we have a *Seidel type (Gauss-Seidel or red-black Gausss-Seidel type)* scheme for the updating procedure which guarantees faster convergence than the Jacobi-type scheme? The answer is *affirmatively* given. It will be clear from Remark 1 that Gauss-Seidel schemes will be as twice fast as the corresponding Jacobi ones, and from the next section that, by exploiting the red-black procedure, Gauss-Seidel schemes will guarantee such fast convergence when implemented in parallel.

Seidel-type Domain Decomposition Iterative Method

Gauss-Seidel iteration procedure

The *Seidel-type* domain decomposition iterative procedure is obtained by replacing the interface condition (4) by

$$\nu_j \cdot A\nabla u_j^n + \beta u_j^n = \begin{cases} -\nu_k \cdot A\nabla u_k^{n-1} + \beta u_k^{n-1}, & j < k, \\ -\nu_k \cdot A\nabla u_k^n + \beta u_k^n, & j > k, \end{cases} \quad \text{on } \Gamma_{jk},\ \forall k, \tag{8}$$

and hence (7) by

$$\begin{aligned} & a_j(u_j^n, \nabla\varphi)_j + \sum_k \langle \beta u_j^n, \varphi\rangle_{\Gamma_{jk}} \\ & = F_j(\varphi) + \left\{ \begin{array}{ll} \sum_k \langle -\lambda_{kj}^{n-1} + \beta u_k^{n-1}, \varphi\rangle_{\Gamma_{jk}}, & j < k, \\ \sum_k \langle -\lambda_{kj}^{n} + \beta u_k^{n}, \varphi\rangle_{\Gamma_{jk}}, & j > k. \end{array} \right. \end{aligned} \tag{9}$$

Let $\tilde{u}_j = u|_{\Omega_j}$ and $\tilde{\lambda}_{jk} = -\nu_j \cdot A\nabla\tilde{u}_j|_{\Gamma_{jk}}$ so that $\tilde{u}_j$ and $\tilde{\lambda}_{jk}$ satisfy the local equations

$$\begin{aligned} a_j(\nabla\tilde{u}_j, \nabla\varphi)_j - \sum_k \langle \tilde{\lambda}_{jk}, \varphi\rangle_{\Gamma_{jk}} &= F_j(\varphi),\ \varphi \in V_j, \\ \tilde{\lambda}_{jk} &= -\tilde{\lambda}_{kj} - \beta(\tilde{u}_j - \tilde{u}_k),\ \Gamma_{jk}\,\forall k. \end{aligned}$$

We will show the convergence of (u_j^n, λ_{jk}^n) to $(\tilde{u}_j, \tilde{\lambda}_{jk})$. Set

$$e_j^n = u_j^n - \tilde{u}_j, \qquad \mu_{jk}^n = \lambda_{jk}^n - \tilde{\lambda}_{jk}.$$

From (9) and (10), we have the error equations: for all j,

$$a_j(\nabla e_j^n, \nabla\varphi) - \sum_k \langle \mu_{jk}^n, \varphi\rangle_{\Gamma_{jk}} = 0,\ \varphi \in V_j, \tag{10}$$

$$\mu_{jk}^n = \left\{ \begin{array}{ll} -\mu_{kj}^{n-1} - \beta(e_j^n - e_k^{n-1}), & j < k, \\ -\mu_{kj}^{n} - \beta(e_j^n - e_k^{n}), & j > k, \end{array} \right. \quad \text{on } \Gamma_{jk},\ \forall k. \tag{11}$$

The choice $v = e_j^n$ in (10) gives

$$a_j(\nabla e_j^n, \nabla e_j^n) - \sum_k \langle \mu_{jk}^n, e_j^n\rangle_{\Gamma_{jk}} = 0. \tag{12}$$

We rewrite (11) as follows:

$$\begin{aligned} \mu_{jk}^n &= -\mu_{kj}^{n-1} - \beta(e_j^n - e_k^{n-1}), && j < k, \\ \mu_{jk}^n &= -\mu_{kj}^{n} - \beta(e_j^n - e_k^{n}), && j > k, \\ &= \mu_{jk}^{n-1} + \beta(e_k^n - e_j^{n-1}) - \beta(e_j^n - e_k^n) \\ &= \mu_{jk}^{n-1} - \beta e_j^n + 2\beta e_k^n - \beta e_j^{n-1}. \end{aligned} \tag{13}$$

This motivates us to define the *pseudo-energy* for the Seidel-type iterative procedure by

$$R^n := R(e^n, \mu^n) = \sum_{j<k} \left|\mu_{jk}^n + \beta e_j^n\right|_{0,\Gamma_{jk}} + \sum_{j>k} \left|\mu_{jk}^n + \beta(e_j^n - 2e_k^n)\right|_{0,\Gamma_{jk}}. \tag{14}$$

We observe that by (13), for $j > k$,

$$\mu_{jk}^n + \beta(e_j^n - 2e_k^n) = -\mu_{kj}^n - \beta e_k^n,$$

which implies that R^n given by (14) can be equivalently put in the simpler form:

$$R^n(e, \mu) = \sum_{j<k} \left|\mu_{jk}^n + \beta e_j^n\right|_{0,\Gamma_{jk}} + \sum_{j>k} \left|\mu_{kj}^n + \beta e_k^n\right|_{0,\Gamma_{jk}}. \tag{15}$$

Theorem 1 *For a given* $(u^0, \lambda^0) \in V \times \Lambda$*, if iterative solutions* $(u^n, \lambda^n) \in V \times \Lambda$ *are computed by using (9), the pseudo-energy given by (15) satisfies*

$$R^n(e, \mu) = R^{n-1}(e, \mu) - 8Re \sum_{j,k} \langle \mu_{jk}^{n-1}, \beta e_j^{n-1} \rangle_{\Gamma_{jk}}.$$

Proof. Stating from (15), by suitable swapping of the indices j and k, we have

$$\begin{aligned}
R^n &= 2\sum_{j<k} \left|\mu_{jk}^n + \beta e_j^n\right|_{0,\Gamma_{jk}} \\
&= 2\sum_{j<k} \left|\mu_{kj}^{n-1} - \beta e_k^{n-1}\right|_{0,\Gamma_{jk}} \qquad \text{by (13)} \\
&= 2\sum_{j>k} \left|\mu_{jk}^{n-1} - \beta e_j^{n-1}\right|_{0,\Gamma_{jk}} \\
&= 2\sum_{j>k} \left|-\mu_{kj}^{n-1} + \beta(e_k^{n-1} - 2e_j^{n-1})\right|_{0,\Gamma_{jk}} \qquad \text{by (13)} \\
&= 2\sum_{j<k} \left|\mu_{jk}^{n-1} - \beta(e_j^{n-1} - 2e_k^{n-1})\right|_{0,\Gamma_{jk}} \\
&= 2\sum_{j<k} \left|\mu_{jk}^{n-1} + \beta e_j^{n-1} - 2(e_j^{n-1} - \beta e_k^{n-1})\right|_{0,\Gamma_{jk}} \\
&= R^{n-1} - 8\mathrm{Re}\sum_{j<k} \left\langle \mu_{jk}^{n-1} + \beta e_j^{n-1}, \beta(e_j^{n-1} - e_k^{n-1})\right\rangle_{\Gamma_{jk}} \\
&\qquad +8\sum_{j<k} \left|\beta(e_j^{n-1} - e_k^{n-1})\right|_{0,\Gamma_{jk}} \\
&= R^{n-1} - 8\mathrm{Re}\sum_{j<k} \left\langle \mu_{jk}^{n-1} + \beta e_k^{n-1}, \beta(e_j^{n-1} - e_k^{n-1})\right\rangle_{\Gamma_{jk}} \\
&= R^{n-1} - 8\mathrm{Re}\sum_{j,k} \left\langle \mu_{jk}^{n-1}, \beta e_j^{n-1}\right\rangle_{\Gamma_{jk}}
\end{aligned}$$

since

$$\begin{aligned}
&\mathrm{Re}\left[\sum_{j<k} \left\langle -\beta e_k^{n-1}, \beta(e_j^{n-1} - e_k^{n-1})\right\rangle_{\Gamma_{jk}} + \sum_{j<k} \left\langle \mu_{jk}^{n-1}, \beta e_k^{n-1}\right\rangle_{\Gamma_{jk}}\right] \\
&= \mathrm{Re}\sum_{j<k} \left\langle \beta e_k^{n-1}, -\beta(e_j^{n-1} - e_k^{n-1}) + \mu_{jk}^{n-1}\right\rangle_{\Gamma_{jk}} \\
&= \mathrm{Re}\sum_{j>k} \left\langle \beta e_j^{n-1}, \beta(e_j^{n-1} - e_k^{n-1}) + \mu_{kj}^{n-1}\right\rangle_{\Gamma_{jk}} \\
&= -\mathrm{Re}\sum_{j>k} \left\langle \beta e_j^{n-1}, \mu_{jk}^{n-1}\right\rangle_{\Gamma_{jk}} \qquad \text{by (13)} \\
&= -\mathrm{Re}\sum_{j>k} \left\langle \mu_{jk}^{n-1}, \beta e_j^{n-1}\right\rangle_{\Gamma_{jk}}.
\end{aligned}$$

Remark 1 *The reader should observe that the form of pseudo-energy defined in (14) or (15) and both Theorem 1 and its proof are entirely independent of the sesquilinear form $a(\cdot,\cdot)$, and hence Theorem 1 is independent of governing model problem. (Our result depends only on the interface condition (8).) An implication of this observation is that Theorem 1 is valid for a wide range of problems, such as Maxwell and elasticity problems, obviously extending our model problem introduced in the previous section.*

Theorem 2 *The energy R^n can be expressed as*

$$R^n(e,\mu) = R^0(e,\mu) - 8\beta \sum_{k=1}^{n-1} \sum_{j=1}^{J} Re a_j(e_j^k, \beta e_j^k)_j.$$

Now, take the real part in (12) to obtain

$$\mathrm{Re} \sum_k \langle \mu_{jk}^n, e_j \rangle_{\Gamma_{jk}} = \mathrm{Re} a_j(\nabla e_j^n, \nabla e_j^n).$$

and choose $\beta = \beta_R + i\beta_I$ with positive real and nonnegative imaginary parts. Then, under additional assumptions on B_I and α_I such that $B_I \geq 0$ and $\alpha_I \geq 0$, which are indeed physically valid, we have

$$\begin{aligned} \mathrm{Re} a_j(e_j^n, \beta e_j^n)_j &= \beta_R \Big[(A\nabla e_j^n, \nabla e_j^n)_j + (B_R e_j^n, e_j^n)_j + \langle \alpha_R e_j^n, e_j^n \rangle_\Gamma \Big] \\ &\quad + \beta_I \Big[(B_I e_j^n, e_j^n)_j + \langle \alpha_I e_j^n, e_j^n \rangle_{\Gamma_j} \Big] > 0. \end{aligned}$$

In this case, we can conclude from (2) that e_j^n tends to zero as $n \to \infty$.

Remark 2 *For the Jacobi case with the same form of energy as in (15), it is well-known after Després [Des91] that the corresponding decay relations to Theorems 1 and 2 have the form*

$$R^n(e,\mu) = R^{n-1}(e,\mu) - 4Re \sum_{j,k} \langle \mu_{jk}^{n-1}, \beta e_j^{n-1} \rangle_{\Gamma_{jk}},$$

and

$$R^n(e,\mu) = R^0(e,\mu) - 4\beta \sum_{k=1}^{n-1} \sum_{j=1}^{J} Re a_j(e_j^k, \beta e_j^k)_j.$$

Therefore we conclude that the Seidel scheme is exactly as twice fast as the Jacobi scheme.

Red-black Gauss-Seidel procedure

Jacobi-type iterative algorithms are easily parallelizable, but Seidel-type are not easily parallelizable. In order to parallelize the introduced Seidel scheme, we propose a *red-black Seidel scheme* with efficiency equivalent to the Seidel-type one. For this, divide the subdomain indices into the two parts J_R and J_B, so that

$$\bar{\Omega} = \left[\cup_{j\in J_R} \bar{\Omega}_j \right] \bigcup \left[\cup_{j\in J_B} \bar{\Omega}_j \right], \quad \Omega_j \cap_{j\neq k} \Omega_k = \emptyset,$$

and every element $\Omega_j, j \in J_R$, is not adjacent to any element $\Omega_k, k \in J_B$.

With an initialization, the red-black iteration scheme is then the altenations of the following steps

1. $\forall j \in J_R$, solve (7) for $u^n \in V$ with

$$\lambda_{jk}^n \quad = \quad -\lambda_{kj}^{n-1} + \beta\big(u_j^n(\xi_{jk}) - u_k^{n-1}(\xi_{jk})\big)$$

2. $\forall j \in J_B$, solve (7) for $u^n \in V$ with

$$\lambda_{jk}^n \quad = \quad -\lambda_{kj}^n + \beta\big(u_j^n(\xi_{jk}) - u_k^n(\xi_{jk})\big).$$

The pseudo-energy for the red-black Seidel-type iterative procedure takes the similar form as (14) or (15) for errors

$$\begin{aligned} R^n \quad := \quad & R(e^n, \mu^n) = \sum_{j \in J_R} \left|\mu_{jk}^n + \beta e_j^n\right|_{0,\Gamma_{jk}} + \sum_{j \in J_B} \left|\mu_{jk}^n + \beta(e_j^n - 2\beta e_k^n)\right|_{0,\Gamma_{jk}} \\ = \quad & \sum_{j \in J_R} \left|\mu_{jk}^n + \beta e_j^n\right|_{0,\Gamma_{jk}} + \sum_{k \in J_B} \left|\mu_{kj}^n + \beta e_k^n\right|_{0,\Gamma_{jk}} \end{aligned}$$

The same arguments as the *Gauss-Seidel case* lead the analogous results as Theorems 1 and 2, and Remark 1.

References

[BD97]J. D. Benamou and B. Després. A domain decomposition method for the Helmholtz equation and related optimal control. *J. Comp. Phys.*, 136:68–82, 1997.

[CDJP97]P. Collino, G. Delbue, P. Joly, and A. Piacentini. A new interface condition in the non-overlapping domain decomposition. *Comput. Methods Appl. Mech. Engrg.*, 148:195–207, 1997.

[CGJ98]F. Collino, S. Ghanemi, and P. Joly. Domain decomposition method for harmonic wave propagation: A general presentation. Technical Report N° 3473, INRIA, 1998.

[Den97]Q. Deng. An analysis for a nonoverlapping domain decomposition iterative procedure. *SIAM J. Sci. Comput.*, 18:1517–1525, 1997.

[Des91]B. Després. *Méthodes de décomposition de demains pour les problèms de propagation d'ondes en régime harmonique.* PhD thesis, Université Paris IX Dauphine, 1991.

[DM97]J. Douglas, Jr. and D. B. Meade. Second-order transmission conditions for the Helmholtz equation. In P. E. Bjørstad, M. Espedal, and D. Keyes, editors, *Ninth International Conference on Domain Decomposition Methods*, pages 434–440. ddm.org, 1997.

[DPRW93]J. Douglas, Jr., P. L Paes Leme, J. E. Roberts, and J. Wang. A parallel iterative procedure applicable to the approxiamte solution of second order partial differential equations by mixed finite element methods. *Numer. Math.*, 65:95–108, 1993.

[DSSY99]J. Douglas, Jr., J. E. Santos, D. Sheen, and X. Ye. Nonconforming Galerkin methods based on quadrilateral elements for second order elliptic problems. *Mathematical Modelling and Numerical Analysis, M^2AN*, 33:747–770, 1999.

[Fen97]X. Feng. A non-overlapping domain decomposition method for solving elliptic problems by finite element methods. In P. E. Bjørstad, M. Espedal, and D. Keyes, editors, *Ninth International Conference on Domain Decomposition Methods*, pages 222–229. ddm.org, 1997.

[Gha97]S. Ghanemi. A domain decomposition method for Helmholtz scattering problems. In P. E. Bjørstad, M. Espedal, and D. Keyes, editors, *Ninth International Conference on Domain Decomposition Methods*, pages 105–112. ddm.org, 1997.

[HKS99]T. Ha, O. Kwon, and D. Sheen. Seidel-type nonconforming domain decomposition methods for Maxwell's equations. to appear, 1999.

[KS99]O. Kwon and D. Sheen. Seidel-type nonoverlapping domain decomposition iterative methods. to appear, 1999.

[Kwo99]O. Kwon. Spectral radius of red-black Gauss-Seidel nonconforming domain decomposition methods for second order elliptic problems. to appear, 1999.

[Lio88]Pierre-Louis Lions. On the Schwarz alternating method. I. In Roland Glowinski, Gene H. Golub, Gérard A. Meurant, and Jacques Périaux, editors, *First International Symposium on Domain Decomposition Methods for Partial Differential Equations*, pages 1–42, Philadelphia, PA, 1988. SIAM.

[Lio90]Pierre Louis Lions. On the Schwarz alternating method. III: a variant for nonoverlapping subdomains. In Tony F. Chan, Roland Glowinski, Jacques Périaux, and Olof Widlund, editors, *Third International Symposium on Domain Decomposition Methods for Partial Differential Equations , held in Houston, Texas, March 20-22, 1989*, Philadelphia, PA, 1990. SIAM.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau, ©2001 DDM.org

18. On the Interface and Two-Level Preconditioners in Newton-Schwarz Method

Daniel Lee[1]

Introduction

This paper is concerned with parallel computation in solving the convection-diffusion equation and the incompressible Navier-Stokes equation via Newton-Schwarz method, a nonlinear domain decomposition (DD) method. Various preconditioners are investigated here. An interface problem is tackled as a preconditioner for nonlinear block Jacobi DD approach, with an optional fine level interface problem solved as further preconditioner. Also a (global) coarse level preconditioner is considered. Examined also is the relaxation type preconditioner. Such preconditioned nonlinear DD methods exhibit impressive improvement over the basic non-preconditioned parallel Newton-Jacobi method.

A general review on Newton-Schwarz method is [GEMT98]. Our setup has the advantages of both the overlapped and the nonoverlap DD approach. The subdomain variables form a (nonoverlap) partition of the whole global system of equations. Solving the interface problem is regarded as a preconditioner to all subproblems. We note that the interface variables were excluded from subproblems in [LC98].

We describe in later sections the problem and solution procedure, the test cases and results, and a brief conclusion.

Solution Procedure and Newton-Schwarz

Following previous work [LC98], boundary-fitted cell-center type finite volumes with collocated grids were assumed for geometry discretization. Test problems ([Wan89]) are considered in integral form, involving properly defined numerical fluxes. All the definite integrals are further discretized ([Lee99], [JYL99]) as weighted averages involving the primitive (and the flux) variables at neighboring cells. To double-check our numerical observation, we solved both the coupled system, consisting of the continuity and the momentum equations, and also in a decoupled way (PISO, [Iss85]) for the system consisting of the momentum and the Pressure-Poisson equation. We tested also the Burgers equation. The discrete global nonlinear system is decomposed into partition of (nonoverlap) blocks of equations. Basic Parallel Newton-Jacobi (PNJ) method can be then carried out accordingly.

Interface Preconditioner

We regard an interface preconditioner as solving the interface problem on the interface B, before each nonlinear block Jacobi iteration. We prepose the following for further discussion.

[1] National Center for High-Performance Computing, Taiwan, R.O.C., c00dle00@nchc.gov.tw

Procedure IPPNJ (Interface-Preconditioned Parallel Newton-Jacobi) :

```
Do While $\left\{\left|X^{new} - X^{old}\right| \geq$ global-tolerance$\right\}$
    Step 1 : Solve the discrete algebraic nonlinear system $F_i(x) = 0$, $x \in \Omega_i$
             for all subdomain problems in parallel.
    Step 2a : Set up the interface problem, with infomation communicated
              from each subdomain.
    Step 2b : Solve the nonlinear system for the interface problem.
                  $F_B(x) = 0$, $x \in B$.
    Step 2c : Update via communication the interior boundary conditions
              for all subproblems with the interface variables just solved.
End Do
```

The interface problem is relatively small in size, easier to solve by an approximate matrix-free Newton-GMRes method [BS90], and the solution is supposed to offer more accurate interior boundary condition at the interface variables in an efficient way. This is the spirit of IPPNJ. That is, acceleration on (only) the interface variables yields a preconditioner for subsequent DD iterations. In implementation the interface preconditioner is squeezed into after and before two consecutive block Jacobi iterations. We mention that, in our setup, an interface-preconditioned Newton-Schwarz method corresponds actually to a mixing of nonlinear analogue of Schwarz type and Schur-complement type linear DD methods.

Preconditioned Block Newton-Schwarz Procedure

For general application, we recast the discrete system as $\Phi(u) = rhs$, where $\Phi = (\Phi_1, \; \cdot \; \cdot \;, \Phi_{n_d})^T$, $u = (u_1, \; \cdot \; \cdot, u_{n_d})^T$, with $u_i \in R^{ds} \equiv X_s$. Here s denotes a subdomain (and a subproblem), d_s denotes the dimension of subproblem (on subdomains). We assume equal size of the subproblems for simplicity. The space X_s is therefore where a solution to the discrete subproblem resides. Consider the subproblems $\widetilde{\Phi_i}(\widetilde{u_i}) = \widetilde{rhs_i}$ and $J(\Phi_i, u_i) = \frac{\partial \Phi_i}{\partial u_i}$. We assume regularity of such portion of the global Jacobian. We describe a more general version of previous procedure in details with these notation, based on partition of nonoverlap subdomains and some certain order of the equations and variables.

Procedure PPNJ (Preconditioned Parallel Newton-Jacobi) :

```
Do while (global convergence achieved or maximum DD iteration exceeded)
    1. Do $i = 1, \cdots, n_d$ (in parallel)
        a. set $\widetilde{u_i}$ by $u$ and canonical projection
        b. set $\widetilde{rhs_i}$
        c. obtain an approximated solution to $\widetilde{\Phi_i}(\widetilde{u_i}) = \widetilde{rhs_i}$
        d. evaluate for local convergence the residual $\widetilde{r_i} := \widetilde{rhs_i} - \widetilde{\Phi_i}(\widetilde{u_i})$
        e. evaluate for local convergence the difference $diff_i := \|\widetilde{u_i} - \widetilde{u_{i_{sav}}}\|$
       End Do
    2. Update global $u$ by communicating the $\widetilde{u_i}$ among relevant processors
    3. Check global convergence by evaluating $max_i\ diff_i$ or $\|\Phi(u)\|$ on $X_{n_s}^{n_d}$
    4. If global convergence is satisfied, then break. Otherwise
        a. Do an optional accelerator or preconditioner, such as interface
           preconditioner or global coarse level preconditioner.
        b. Update interface variables by communication, and approximation
           schemes in case of a two level preconditioner.
    5. Update the DD iteration counter.
End Do
```

Other Preconditioners

The solution procedure in solving the interface problem and all subproblems are identical by default. However, the interface problem can be solved optionally with a fine-level interface preconditioner. This is affordable if the two level setup for the interface problem remains relatively cheaper than the other subproblem both in storage and computation. It is therefore a natural iterative refinement procedure based on the consideration of load balance. A global coarse level preconditioner is also a good choice, hopefully produces global information update as motivated by the linear DD theory. Relaxation type strategy, in simple or hybrid form, can be used to accelerate the convergence on the interface variables, as to provide an interface preconditioner to the global problem. Furthermore, one can even over-relax the setup of the interface problems resulting in an accelerated interface preconditioner [LY00].

Numerical Results and Discussions

All cases were tested on a PC cluster at NCHC. The global region consists of nine subdomains in cases 1-5, and eight in case 6. Reynolds number is 1 in case 1, and 10 in cases 2-6. More about test is given in Table 1. The cpu time spent is shown in Table 2. Only partial results in accuracy and convergence are shown (Figures 1-12), due to limitation of space. Four-subdomain cases are also tested, showing behavior similar to what we described below.

Case 1 : Four DD methods are tested : Parallel Newton-Jacobi (PNJ), Interface-Preconditioned Parallel Newton-Jacobi (IPPNJ), Coarse-level-

Preconditioned Parallel Newton-Jacobi (CPPNJ), and Fine-level Interface-Preconditioned Parallel Newton-Jacobi (FIPPNJ). Convergence up to nine digits is enforced. The maximum DD-iterations is set to 100 to examine the stability. Although in practice this may be much smaller.

Shown in Figures 1-2 are the accuracy and convergence in relative sense. All yield very stable discrete dynamics. CPPNJ the fastest while PNJ the slowest. IPPNJ and FIPPNJ converges at about the same rate. About accuracy, CPPNJ is the worst and IPPNJ achieves the same accuracy both in absolute and relative sense, and takes about half iterations as PNJ. This is similar to what bewteen classical Gauss-Seidel and Jacobi iterations. We point out that the precision achieved with the two-level preconditioners depend certainly on local interpolation or approximation, and are confined therefore by the spatial grid resolution. It is witnessed that FIPPNJ saturated at some earlier stage, and was forced therefore to iterate 100 iterations. This hurt in the cpu-time contest (Table 2). Although, the convergence history does justify the stableness in computation. The FIPPNJ result seems not as appealing as that of IPPNJ. Therefore FIPPNJ is excluded in other test cases below.

Case 2 : PNJ is validated and applied to both the decoupled and the coupled approach, i.e., with or without PISO. Then we will test later in case 4 our proposed preconditioners within these two different approaches. Our algorithms and implementation is justified (Figure 3 and 4). The accuracy when without PISO is only slightly inferior to that with PISO. The time spent in computation of third order derivatives in Pressure-Poisson equation for the PISO formulation seems a disadvantage.

Case 3 : IPPNJ is applied with or without PISO. The findings, Figures 5 and 6, are similar to case 2.

Case 4 : The coupled system without PISO is solved with PNJ, IPPNJ and CPPNJ methods. CPPNJ is the most accurate and IPPNJ also outperforms PNJ (Figure 7). The convergence in the relatively severe maximum norm (Figure 8), although not as smooth as with the normalized two-norm (not shown), does indicate the relative spirits in these methods. Very heavy communications are seen on the coarse preconditioner, in the pre- and post- data processing in solving the global coarse problem. Therefore we exempted CPPNJ from subsequent cases.

Case 5 : Here we solve with PISO and compare PNJ and IPPNJ. The latter converges faster and is more accurate (Figures 9-10).

Case 6 : Relaxation type preconditioners are examined with various relaxation parameters. The kind of monotonicity of the effectiveness in accuracy (Figure 12) and convergence (Figure 11) is obviously seen. More thoughts along this line, including combination of different strategy and even working on over-relaxed interface problem, is investigated in [LY00].

Conclusion

Several preconditioners are designed for the interface problem in a Newton-Schwarz approach. With these the basic parallel block Jacobi precodure either converges faster or is more accurate. We found that IPPNJ converges faster while achieving same or better accuracy and costs less computation time, and that CPPNJ and FIPPNJ achieve moderate precision and converge faster in terms of the number of DD iterations, but both with apparently heavier communication overhead. The choice certainly depends on, among others, the spatial resolution and the required precision in computed results. We believe future technology improvement in system architecture will resolve the communication inefficiency to a large extent.

Acknowledgement

This work was supported by the National Science Council, Taiwan, R.O.C., under contract NSC-88-2113-E321-003. Some of the ideas was developed at CERFACS with helpful suggestions from Professor Iain Duff and Mr. Luc Giraud. Very sincere gratitude is expressed here.

References

[BS90]Peter N. Brown and Yousef Saad. Hybrid Krylov methods for nonlinear system of equations. *SIAM J. Sci. Stat. Comput.*, 11:450–481, 1990.

[GEMT98]W. D. Gropp, D. E.Keyes, L. C. McInnes, and M. D. Tidriri. Globalized Newton-Krylov-Schwarz algorithms and software for parallel implicit CFD. Technical Report 98-24, ICASE, August 1998. NASA/CR-1998-208435.

[Iss85]R. I. Issa. Solution of the implicitly discretised fluid flow equations by operator-Splitting. *Journal of Computational Physics*, 62:40–65, 1985.

[JYL99]Kang C. Jea, Mulder Yu, and Daniel Lee. A study on finite volume methods. In D. R. Kincaid et. al. eds, editor, *Iterative Methods on Scientific Computation II*. IMACS, 1999.

[LC98]Daniel Lee and Chih-Hua Chen. A study on domain based parallel flow computation. In *Proceeding of the Fifth National Conference in Computational Fluid Dynamics*, Taiwan, R.O.C., August 27 1998.

[Lee99]Daniel Lee. A new approach to finite volume-finite difference methods. *Chinese Journal of Numerical Mathematics and Applications*, 21(2):96–102, 1999.

[LY00]Daniel Lee and Mulder Yu. Parallel flow computation with domain based SOR type interface preconditioner. In *Proceedings of the Fourth International Conference on High-Performance Computing in the Asia-Pacific Region (HPC-Asia 2000)*, Beijing, China, May 2000. To appear.

[Wan89]C. Y. Wang. Exact solutions of the unsteady Navier-Stokes equations. *Appl. Mech. Rev.*, 42:269–282, 1989.

Table 1: Parameters of test runs with supremum norm; range of x, y and z in cases 1 and 6 is (0.0, 1.0); range of x and y in cases 2-5 is (1.0, 2.0).

case	eq.	DD_it	ht	ht_CFL	nx, ny, nz	PISO	method
1	2D Burgers	100	1e-3	9.00e-06	60, 60	0	various
2	2D NS	20	1e-3	5.62e-06	240, 240	0, 1	PNJ
3	2D NS	20	1e-3	5.62e-06	240, 240	0, 1	IPPNJ
4	2D NS	20	1e-3	5.62e-06	240, 240	0	various
5	2D NS	20	1e-3	5.62e-06	240, 240	1	various
6	3D Burgers	50	1e-2	4.16e-03	10, 10, 10	0	PNJ

Table 2: Cpu time in cases 1-6.

case		iter	total sub. solver	total sub. solver	other overhead	total cpu time
1	PNJ	81	4.05e+02	4.52e+02	9.00e+00	8.66e+02
	IPPNJ	38	2.43e+02	2.07e+02	3.00e+00	4.53e+02
	CPPNJ	11	1.01e+02	6.96e+01	2.40e+00	1.81e+02
	FIPPNJ	100	1.88e+03	4.34e+02	6.00e+00	2.32e+03
2	PISO 0	20	2.90e+03	3.28e+03	8.00e+01	6.26e+03
	PISO 1	20	3.42e+03	1.34e+04	8.00e+01	1.69e+04
3	PISO 0	20	2.50e+03	2.78e+03	8.00e+01	5.36e+03
	PISO 1	20	4.56e+03	2.38e+04	1.50e+02	2.84e+04
4	PNJ	20	2.86e+03	3.22e+03	9.00e+01	6.15e+03
	IPPNJ	20	2.44e+03	2.74e+03	9.00e+01	5.25e+03
	CPPNJ	20	1.23e+04	3.48e+03	1.20e+02	1.59e+04
5	PNJ	20	3.34e+03	1.34e+04	1.10e+02	1.69e+04
	IPPNJ	20	1.52e+04	1.44e+04	1.00e+02	2.97e+04
6	without SOR	10	6.07e+00	2.28e+01	8.73e+00	3.76e+01
	over with 1.1	9	4.86e+00	2.00e+01	8.24e+00	3.31e+01
	over with 1.2	10	5.52e+00	2.25e+01	8.48e+00	3.65e+01
	over with 1.3	12	5.32e+00	2.68e+01	9.19e+00	4.12e+01
	over with 1.4	19	9.39e+00	4.31e+01	1.21e+01	6.46e+01

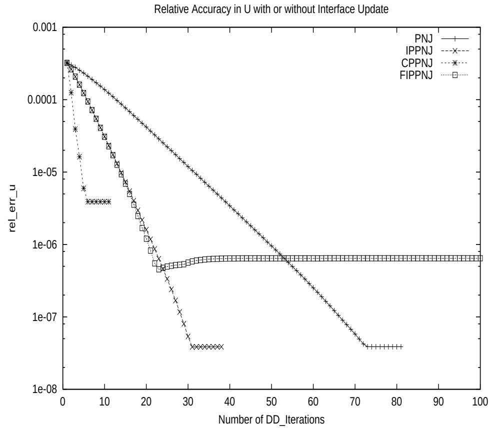


Figure 1: Relative accuracy in solving Burgers' eq. in case 1.

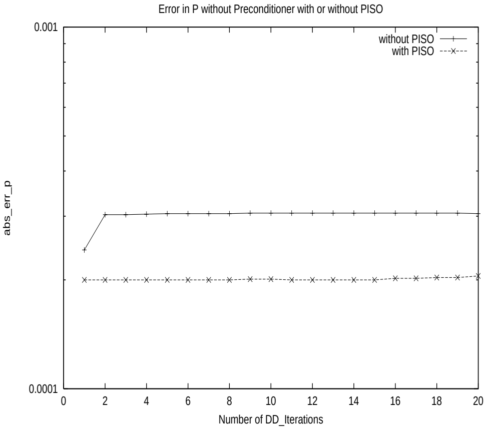


Figure 3: Accuracy in p with or without PISO in case 2.

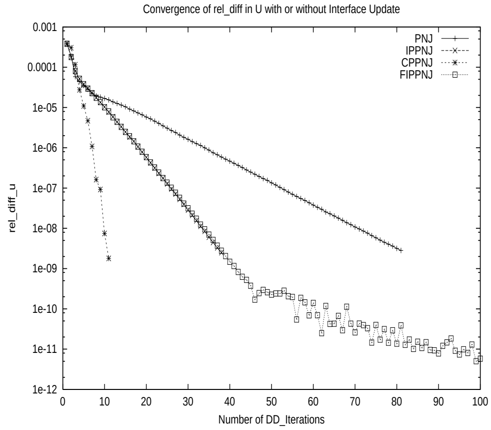


Figure 2: Relative convergence in solving Burgers' eq. in case 1.

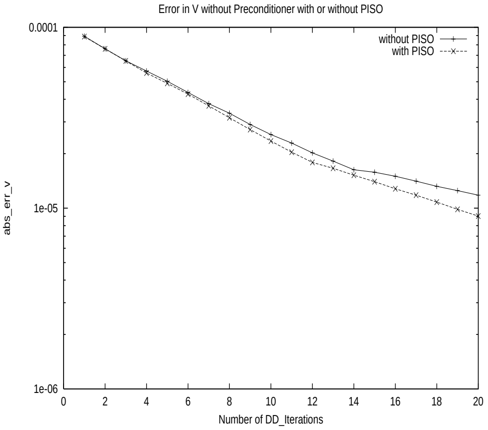


Figure 4: Accuracy in v with or without PISO in case 2.

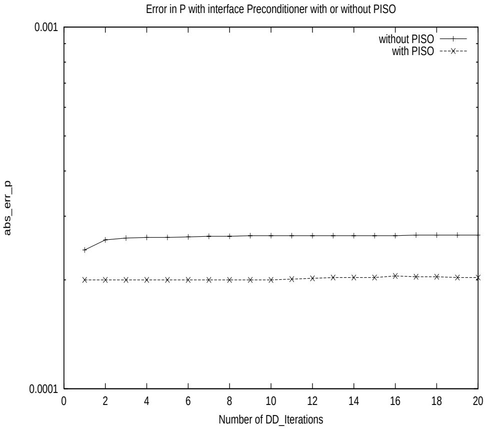


Figure 5: Accuracy in p with or without PISO in case 3.

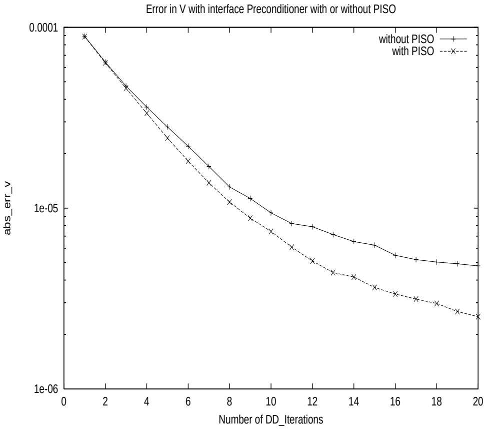


Figure 6: Accuracy in v with or without PISO in case 3.

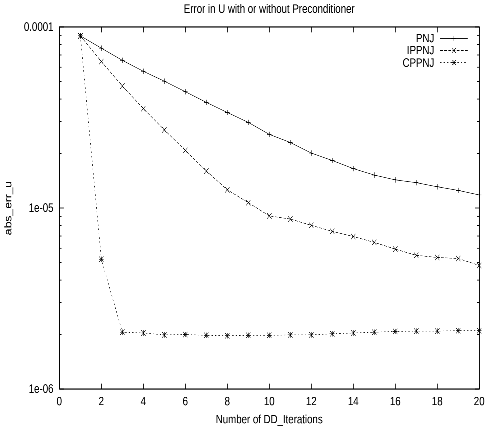


Figure 7: Accuracy in u with or without preconditioner in case 4.

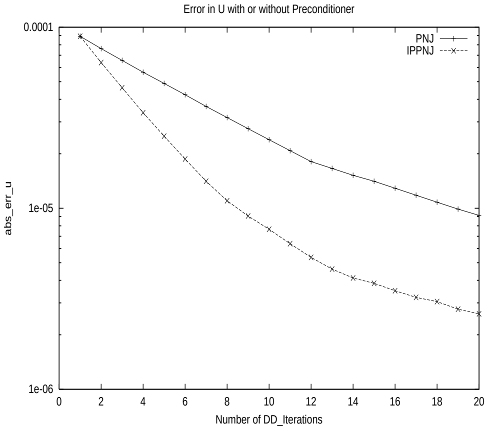


Figure 9: Accuracy in u with or without preconditioner in case 5.

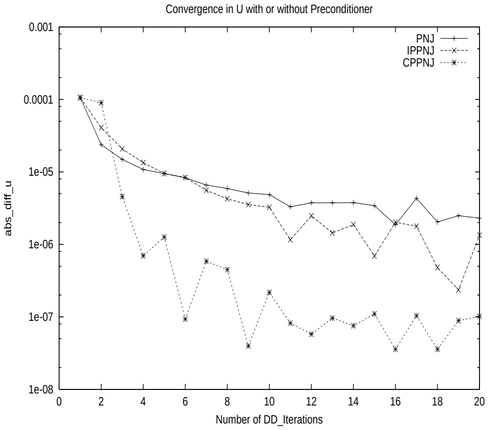


Figure 8: Convergence in u with or without preconditioner in case 4.

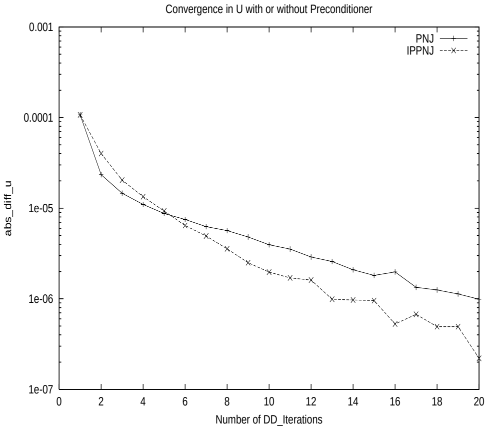


Figure 10: Convergence in u with or without preconditioner in case 5.

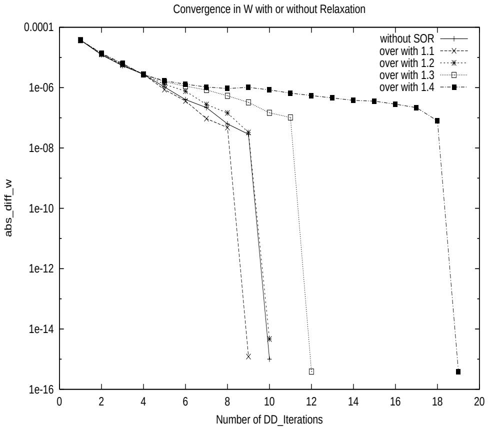


Figure 11: Convergence in w in solving 3D NS eq. with or without relaxation in case 6.

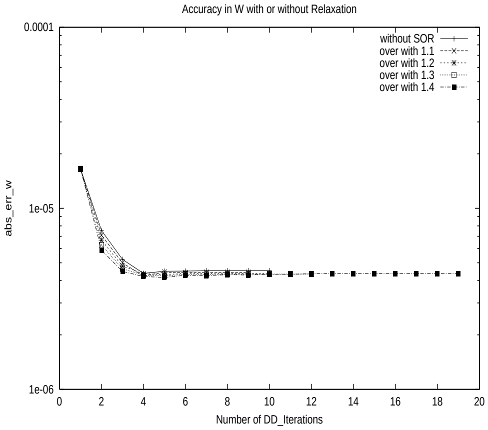


Figure 12: Accuracy in w in solving 3D NS eq. with or without relaxation in case 6.

Appendix

Analytic descriptions of the equations and solutions for our test are given here for easy reference. Dirichlet type boundary data are adopted for the test runs in this paper. We refer to [Wan89] for more explanation on the equations.

(1) 2D Burgers' equation:

$$u_t + u(u_x + u_y) - \frac{1}{Re}(u_{xx} + u_{yy}) \quad = \quad 0, \tag{1}$$

and one solution is :

$$u \quad = \quad \frac{1}{1 + e^{\frac{Re(2x+2y-2t)}{4}}}. \tag{2}$$

(2) 2D Navier-Stokes equation:
Continuity equation:

$$u_x + v_y \quad = \quad 0, \tag{3}$$

X-momentum equation:

$$u_t + uu_x + vu_y \quad = \quad -p_x + \frac{1}{Re}(u_{xx} + u_{yy}), \tag{4}$$

Y-momentum equation:

$$v_t + uv_x + vv_y \quad = \quad -p_v + \frac{1}{Re}(v_{xx} + v_{yy}), \tag{5}$$

and one solution is :

$$\begin{aligned} u &= -cos(x) * sin(y) * e^{\frac{-2t}{Re}}, & (6)\\ v &= sin(x) * cos(y) * e^{\frac{-2t}{Re}}, & (7)\\ p &= 10.0 - \frac{1}{4}(cos(2x) + cos(2y)) * e^{\frac{-4t}{Re}}). & (8) \end{aligned}$$

(3) 3D Burgers' equation:

$$u_t + u(u_x + u_y + u_z) - \frac{1}{Re}(u_{xx} + u_{yy} + u_{zz}) \quad = \quad 0, \tag{9}$$

and one solution is :

$$u \quad = \quad \frac{1}{1 + e^{\frac{Re(2x+2y+2z-3t)}{4}}}. \tag{10}$$

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau,

19. A Mortar Finite Element Method for Plate Problems

L. Marcinkowski [1]

Introduction

In the paper we discuss two versions of mortar finite element methods applied to clamped plate problems. The problems are approximated by the nonconforming Morley and Adini element methods in each subregion into which the original region of the discussed problems have been partitioned. On the interfaces between subdomains and at crosspoints of subregions some continuity conditions are imposed.

The main results of the paper are the proof of the solvability of the discrete problems and their error bounds.

The mortar method is a domain decomposition method that allow us to use discretizations of different type with independent discretizations parameters in nonoverlapping subdomains, see e.g. [BMP94], [BM97], [BB99] for a general presentation of the mortar method in the two and three dimensions for elliptic boundary value problems of second order.

In the paper mortar element methods for the locally nonconforming discretizations of the clamped plate problems are discussed. In [Lac98] there are formulated results for mortar method with nonconforming discrete Kirchoff triangle elements (DKT) for a similar problem while in [Bel97] the mortar method for the biharmonic problem is analyzed in the case of local spectral discretizations. The paper is based on the results which are obtained in the PhD thesis of the author, see [Mar99b], cf. also [Mar99a].

This paper is concerned with the mortar method where locally in the subdomains the nonconforming Adini and Morley plate finite elements are used. We restrict ourselves to the geometrically conforming version of the mortar method, i.e. the local substructures form a coarse triangulation. We first introduce independent local discretizations for the two discussed elements in each subdomain. The 2-D triangulations of two neighboring subregions do not necessarily match on their common interface, cf. Figure 1. The mortar technique for nonconforming plate elements which is discussed here requires the continuity of the solution at the vertices of subdomains and that the solution on two neighboring subdomains satisfies two mortar conditions of the L^2 type on their common interface. The form of these conditions depends on the local discretization methods and in some cases these conditions combine interpolants defined locally on interfaces. It follows from the fact that the respective traces of local functions also depend on the values of respective degrees of freedom at interior nodal points. We give error bounds for the both mortar methods. The results obtained in this paper can be generalized to analogous mortar discretizations of simply supported plate problems.

[1] Department of Mathematics, Warsaw University, ul. Banacha 2, 02-097 Warsaw, Poland, E-mail: lmarcin@mimuw.edu.pl. This work was partially supported by Polish Scientific Grant 237/PO3/99/16.

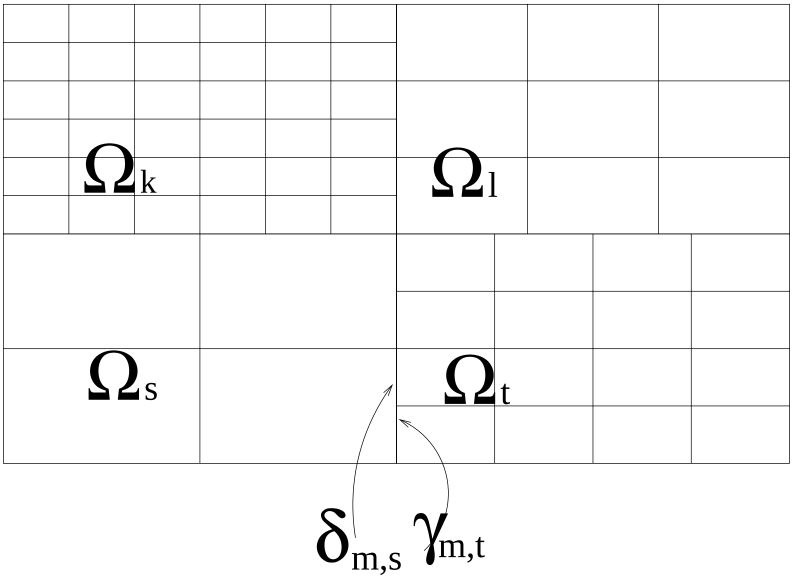


Figure 1: Nonmatching meshes.

Discrete problems

Clamped plate problem

Let Ω be a polygonal domain in $\mathbb{R}^2$. The differential problem is to find $u^* \in H_0^2(\Omega)$ such that

$$a(u^*, v) = \int_\Omega fv\,dx \quad \forall v \in H_0^2(\Omega), \tag{1}$$

where u^* is the displacement, $f \in L^2(\Omega)$ is the body force,

$$a(u,v) = \int_\Omega \left[\triangle u \triangle v + (1-\nu)\left(2u_{x_1x_2}v_{x_1x_2} - u_{x_1x_1}v_{x_2x_2} - u_{x_2x_2}v_{x_1x_1}\right)\right] dx.$$

Here

$$H_0^2(\Omega) = \{v \in H^2(\Omega) : \; v = \partial_n v = 0 \; \text{ on } \; \partial\Omega\},$$

∂_n is the normal unit derivative outward to $\partial\Omega$, and $u_{x_ix_j} := D_iD_ju$ for $i,j = 1,2$. The Poisson ratio ν satisfies $0 < \nu < 1/2$. It is well known that this problem has a unique solution, see e.g. [Cia91].

Let Ω be a union of non-overlapping polygonal subdomains that are arbitrary for the Morley element and are rectangles for the Adini element, i.e. $\overline{\Omega} = \bigcup_{k=1}^N \overline{\Omega}_k, \;\; \Omega_k \cap \Omega_l = \emptyset, \;\;\; k \neq l$. We assume that the intersection of boundaries of two different subdomains $\partial\Omega_k \cap \partial\Omega_l, k \neq l$, is either the empty set, a vertex or a common edge. We assume the shape regularity of that decomposition, cf. [BS94].

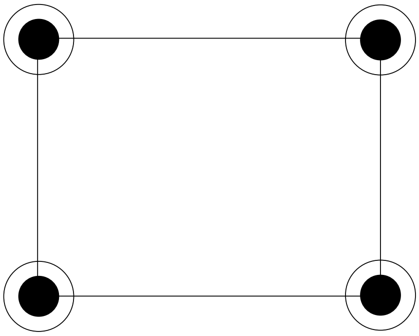


Figure 2: Adini element.

We triangulate each subdomain Ω_k into nonoverlapping rectangles for the Adini element and into triangles for the Morley one. The rectangles (or triangles) of this triangulation are denoted by τ_i and called elements. We assume that the arising fine triangulation $T_h(\Omega_k)$ is quasiuniform with parameter $h_k = \max(\text{ diam } \tau)$ for $\tau \in T_h(\Omega_k)$, cf. [BS94]. The triangulations for different Ω_k are independent and can be nonmatching across interfaces, i.e. on common edges of two subdomains, in general, cf. Figure 1.

Adini element

In this subsection, we introduce a mortar method that locally uses the Adini element, cf. Chapter 7, Section 49, p.298 in [Cia91]. The local finite element space $X_h^A(\Omega_k)$ of the Adini element is defined by

$$X_h^A(\Omega_k) = \{v \in L_2(\Omega_k) : \ \ v_{|\tau} \in P_3(\tau) \oplus \text{span}\{x_1^3 x_2, x_1 x_2^3\} \text{ for } \ \tau \in T_h(\Omega_k),$$

$$v, v_{x_1}, v_{x_2} \ \text{ continuous at the vertices of } \ \tau \ \text{ and}$$

$$v(a) = v_{x_1}(a) = v_{x_2}(a) = 0 \ \text{ for a vertex } \ a \in \partial\Omega_k \cap \partial\Omega\}$$

where $\tau \in T_h(\Omega_k)$ is a rectangular element, cf. Figure 2.

We also introduce the global space $X_h^A(\Omega) = \prod_k X_h^A(\Omega_k)$. For each interface $\overline{\Gamma}_{kl} = \partial\Omega_k \cap \partial\Omega_l$, we choose one side as a master denoted by $\gamma_{m,k} \subset \partial\Omega_k$ and the second one as a slave $\delta_{m,l} \subset \partial\Omega_l$ if $h_k \leq h_l$, see Figure 1. This assumption is necessary for the proof of some technical results and is due to the fact that any local finite element function is not sufficiently regular.

We introduce additional auxiliary spaces on each slave (nonmortar) $\delta_{m,l} \subset \partial\Omega_l$. Let the first one denoted by $M_{1,3}^{h_l}(\delta_{m,l})$ be the space of C^1 smooth functions that are piecewise cubic except for two elements that touch the ends of $\delta_{m,l}$, where are piecewise linear, and let the second one $M_{0,1}^{h_l}(\delta_{m,l})$ be the space of continuous piecewise linear functions which are constant on the two elements which touch the ends of $\delta_{m,l}$.

We say that $u_k \in X_h^A(\Omega_k)$ and $u_l \in X_h^A(\Omega_l)$ for $\partial\Omega_l \cap \partial\Omega_k = \overline{\Gamma}_{kl}$ satisfy the mortar conditions if

$$\int_{\delta_m} (u_k - u_l)\psi \, ds = 0 \;\; \forall \psi \in M_{1,3}^{h_l}(\delta_{m,l}), \tag{2}$$

$$\int_{\delta_m} (I_{h_k}\partial_n u_k - I_{h_l}\partial_n u_l)\psi \, ds = 0 \;\; \forall \psi \in M_{0,1}^{h_l}(\delta_{m,l}), \tag{3}$$

where I_{h_l}, I_{h_k} are the standard piecewise linear interpolants onto the h_l and h_k meshes of $\delta_{m,l}$ and $\gamma_{m,k}$, respectively. Note that $I_{h_i}\partial_n u_i$, for $i = k, l$, equals the normal derivative of piecewise bilinear interpolant defined over Ω_i by the values of $\partial_n u_i$ at the vertices of rectangular elements of $T_h(\Omega_i)$.

We now define the discrete space V_h^A as the subspace of $X_h^A(\Omega)$ formed by functions which satisfy the mortar conditions (2) and (3) on all slave sides and are continuous at all crosspoints.

The discretization of (1) using V_h^A is of the form:
Find $u_h^A \in V_h^A$ such that

$$a_h(u_h^A, v) = \int_\Omega f v \, dx \quad \forall v \in V_h^A, \tag{4}$$

where $a_h(u,v) = \sum_{k=1}^N a_{h,k}(u,v)$ and

$$a_{h,k}(u,v) = \sum_{\tau \in T_h(\Omega_k)} \int_\tau \triangle u \triangle v + (1-\nu)(2u_{x_1x_2}v_{x_1x_2} - u_{x_1x_1}v_{x_2x_2} - u_{x_2x_2}v_{x_1x_1}) \, dx. \tag{5}$$

The form $a_h(\cdot,\cdot)$ is positive definite over V_h^A what follows from the fact that $a_h(u,u) = 0$ implies that u is linear in all rectangles of $T_h(\Omega_k)$, then from the continuity of u, u_{x_1}, u_{x_2} at all vertices of the elements of $T_h(\Omega_k)$ follows that u is linear in Ω_k and from the mortar condition follows that u is linear in Ω. Then the boundary conditions yield $u = 0$.

Moreover, it has been proven in [Mar99b] that this form is uniformly elliptic on V_h^A what is stated in the following:

Theorem 1 *There exists a constant C independent of h_k and the number of subdomains such that for $u \in V_h^A$*

$$C \, \|u\|_{H_h^2(\Omega)}^2 \le a_h(u,u),$$

where $\|u\|_{H_h^2(\Omega)} = (\sum_{k=1}^N \sum_{\tau \in T_h(\Omega_k)} \|u\|_{H^2(\tau)}^2)^{1/2}$ is the so-called broken H^2-norm.

Hence

Proposition 1 *The problem (4) has a unique solution.*

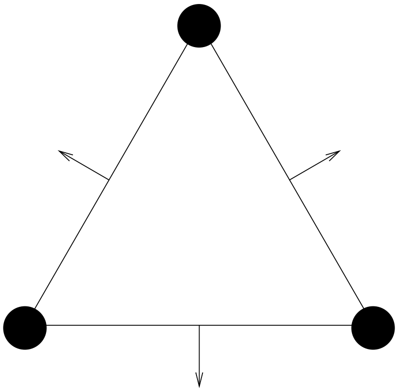


Figure 3: Morley element.

Morley element

In this subsection, we introduce a mortar method that locally uses the Morley element, e.g. cf. [LL75].

The local finite element space $X_h^M(\Omega_k)$ is defined by, see Figure 3,

$$X_h^M(\Omega_k) = \{v \in L_2(\Omega_k) : \ v_{|\tau} \in P_2(\tau), \ v \text{ continuous at vertices of}$$

$$\tau \in T_h(\Omega_k) \text{ and } \partial_n v \text{ continuous at midpoints of edges of } \tau \text{ and}$$

$$v(p) = \partial_n v(m) = 0 \ \text{ for a vertex } p \in \partial\Omega \text{ and a midpoint } m \in \partial\Omega\}.$$

We also introduce a global space $X_h^M(\Omega) = \prod_{k=1}^N X_h^M(\Omega_k)$ as in the previous subsection.

We now select an open disjoint side Γ_{kl} of $\partial\Omega_k$, $\overline{\Gamma}_{kl} = \partial\Omega_k \cap \partial\Omega_l$, denote it by $\gamma_{m,k}$ and name as master (mortar) if $h_k \leq h_l$, cf. Figure 1. This assumption like for the Adini element is necessary for the proof of some technical results and is due to the fact that any local finite element function is not sufficiently regular. The side of $\Gamma_{kl} \subset \partial\Omega_l$ is called slave (nonmortar) and is denoted by $\delta_{m,l}$. As $h_k \leq h_l$ and the both triangulations are quasiuniform, we can assume that the two end elements of the h_l-triangulation of the slave $\delta_{m,l}$, i.e. the ones that touch the ends of $\delta_{m,l}$, are longer than the respective elements of the h_k-triangulation of the master $\gamma_{m,k}$.

We introduce additionally two auxiliary spaces on each slave (nonmortar) $\delta_{m,l}$. Let the first one denoted by $M_{-1,0}^{h_l}(\delta_{m,l})$, be the space of functions which are piecewise constant on the h_l triangulation of $\delta_{m,l}$.

For the simplicity of presentation, we also assume that the both 1-D triangulations of the interface Γ_{kl}, the h_k one of its master $\gamma_{m,k}$ and the h_l one of its slave $\delta_{m,l}$, have even numbers of the elements. Let consider $\delta_{m,l}$ and let $\overline{\delta}_{m,l,h} = \{p_0, p_1, \dots, p_{N_{m,l}}\}$ be a set of vertices of the h_l triangulation of this slave, ($N_{m,l}$ is even). Then we introduce an operator $I_{2h_l,2} : C(\overline{\delta}_{m,l}) \to C(\overline{\delta}_{m,l})$ defined by the values of u at all points of $\overline{\delta}_{m,l,h}$ as follows:

- $I_{2h_l,2}u \in P_2$ on each $[p_i, p_{i+2}]$ for even $i < N_{m,l}$,
- $I_{2h_l,2}u(p_i) = u(p_i) \ \ p_i \in \overline{\delta}_{m,l,h}$.

The operator $I_{2h_k,2}$ that corresponds to the h_k mesh of master $\gamma_{m,k}$ is defined in the same way.

We next define an auxiliary space $M_{0,2}^{2h_l}(\delta_{m,l})$ as follows

$$M_{0,2}^{2h_l}(\delta_{m,l}) = \{v \in C(\overline{\delta}_{m,l}) : \ v \in P_2([p_i, p_{i+2}]) \text{ for even } i \neq 0, N_{m,l} - 2, \tag{6}$$

$$\text{and } v \in \ P_1([p_i, p_{i+2}]) \text{ for } i = 0, N_{m,l} - 2\}.$$

We now introduce the two mortar conditions on the interface $\Gamma_{kl} = \gamma_{m,k} = \delta_{m,l}$:

$$\int_{\delta_m} (I_{2h_k,2}u_k - I_{2h_l,2}u_l)\psi \, ds = 0 \ \ \forall \psi \in M_{0,2}^{2h_l}(\delta_{m,l}) \tag{7}$$

and

$$\int_{\delta_m} (\partial_n u_k - \partial_n u_l)\phi \, ds = 0 \ \ \forall \phi \in M_{-1,0}^{h_l}(\delta_{m,l}). \tag{8}$$

We next define the discrete space V_h^M as the subspace of $X_h^M(\Omega)$ formed by functions which satisfy the mortar conditions (7) and (8) on all slave sides and are continuous at all crosspoints.

The discretization of (1) using V_h^M is of the form:
Find $u_h^M \in V_h^M$ such that

$$a_h(u_h^M, v) = \int_\Omega fv \, dx \quad \forall v \in V_h^M, \tag{9}$$

where $a_h(u,v) = \sum_{k=1}^N a_{h,k}(u,v)$ and $a_{h,k}(u,v)$ are defined as in (5). The form $a_h(\cdot,\cdot)$ is positive definite over V_h^M. It follows from the fact that $a_h(u,u) = 0$ yields that u is piecewise linear in the triangles of $T_h(\Omega_k)$, then the continuity of u at all vertices and $\partial_n u$ at all midpoints of elements of $T_h(\Omega_k)$ yields that u is linear in Ω_k and finally from the mortar conditions follows that u is linear in Ω. The boundary conditions yield $u = 0$.

As in the case of Adini mortar method, we have the uniform ellipticity of the form $a_h(\cdot,\cdot)$ over V_h^M, cf. [Mar99a] and [Mar99b], i.e.

Theorem 2 *There exists a constant C independent of h_k and the number of subdomains such that for $u \in V_h^M$*

$$C \, \|u\|^2_{H_h^2(\Omega)} \leq a_h(u,u),$$

where $\|u\|_{H_h^2(\Omega)}$ is the broken H^2-norm.

Thus we obtain

Proposition 2 *The problem (9) has a unique solution.*

Error estimates

We have the following error estimates for the both elements:

Theorem 3 *Assume that u^*, the solution of (1), is in the space $H_0^2(\Omega) \cap H^4(\Omega)$. Then for the Adini element*

$$\|u^* - u_h^A\|^2_{H_h^2(\Omega)} \leq C_A \sum_{k=1}^{N} \left(h_k^2 |u^*|^2_{H^3(\Omega_k)} + h_k^4 |u^*|^2_{H^4(\Omega_k)} \right),$$

and for the Morley element

$$\|u^* - u_h^M\|^2_{H_h^2(\Omega)} \leq C_M \sum_{k=1}^{N} \left(h_k^2 |u^*|^2_{H^3(\Omega_k)} + h_k^4 |u^*|^2_{H^4(\Omega_k)} \right),$$

where u_h^A and u_h^M are the solutions of (4) and (9), respectively, $\|v\|_{H_h^2(\Omega)}$ is the broken H^2-norm, and C_A, C_M are positive constants independent of u^*, any h_k, and the number of subdomains.

Remark on Additive Schwarz Methods

In this section we make a brief remark on the parallel methods of Schwarz type for solving the discrete problems (4) and (9). The detailed discussion will be published elsewhere.

In [Mar99b] a parallel algorithm for solving (4) was constructed and analyzed. This is a iterative substructuring method, i.e. it is applied to the Schur complement of the discrete problem, i.e. interior variables are first eliminated using some direct methods. The method is described in terms of an Additive Schwarz Method (ASM), cf. [SBG96]. We decompose a discrete space into a sum of subspaces which consists of a coarse space, local one dimensional spaces associated with degrees of freedom of order one at vertices of subdomains, and certain local spaces associated with interfaces. The coarse space is not standard and can be named an exotic one.

A Neumann-Neumann method for solving systems of linear equations arising from conforming mortar discretizations of a plate problem which is constructed and analyzed in [Mar99b], can be adapted to the nonconforming cases of the Adini and Morley discretizations considered in this paper. The analysis of the Neumann-Neumann methods for the Adini case can be done in a similar way to that in [Mar99b] utilizing some technical results which can also be found in [Mar99b], while the case of the Morley element requires some new technical results which have been obtained and which will be published elsewhere.

The described methods are almost optimal, i.e. the number of iterations required to decrease the energy norm of the error by a conjugate gradient method is proportional to $(1 + \log(\frac{H}{\underline{h}}))$, where $H = \max_i(\operatorname{diam} \Omega_i)$ and $\underline{h} = \inf_i h_i$.

References

[BB99]Faker Ben Belgacem. The mortar finite element method with Lagrange multipliers. *Numer. Math.*, 84(2):173–197, 1999.

[Bel97]Z. Belhachmi. Nonconforming mortar element methods for the spectral discretization of two-dimensional fourth-order problems. *SIAM J. Numer. Anal.*, 34(15):1545–1573, 1997.

[BM97]Faker Ben Belgacem and Yvon Maday. The mortar element method for three dimensional finite elements. *RAIRO Mathematical Modelling and Numerical Analysis*, 31(2):289–302, 1997.

[BMP94]Christine Bernardi, Yvon Maday, and Anthony T. Patera. A new non conforming approach to domain decomposition: The mortar element method. In Haim Brezis and Jacques-Louis Lions, editors, *Collège de France Seminar*. Pitman, 1994. This paper appeared as a technical report about five years earlier.

[BS94]Susanne C. Brenner and L. Ridgway Scott. *The Mathematical Theory of Finite Element Methods*, volume 15 of *Texts in Applied Mathematics*. Springer-Verlag, New York, 1994.

[Cia91]P. G. Ciarlet. Basic error estimates for elliptic problems. In *Handbook of Numerical Analysis, Vol. II*, pages 17–351. North-Holland, Amsterdam, 1991.

[Lac98]C. Lacour. Non-conforming domain decomposition method for plate and shell problems. In J. Mandel, C. Farhat, and X.-C. Cai, editors, *Tenth International Conference on Domain Decomposition Methods*, pages 304–310. AMS, Contemporary Mathematics 218, 1998.

[LL75]P. Lascaux and P. Lesaint. Some nonconforming finite elements for the plate bending problem. *Rev. Francaise Automat. Informat. Recherche Operationnelle Ser. Rouge Anal. Numer.*, 9(R-1):9–53, 1975.

[Mar99a]Leszek Marcinkowski. A mortar element method for some discretizations of a plate problem. Tech. Report RW99-13 61, Institute of Applied Mathematics and Mechanics, Warsaw University, Warsaw, October 1999.

[Mar99b]Leszek Marcinkowski. *Mortar Methods for some Second and Fourth Order Elliptic Equations.* PhD thesis, Department of Mathematics, Warsaw University, January 1999.

[SBG96]Barry F. Smith, Petter E. Bjørstad, and William Gropp. *Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations.* Cambridge University Press, 1996.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau, ©2001 DDM.org

20. Bounds for Linear-Functional Outputs of Coercive Problems in Three Space Dimensions

Marius Paraschivoiu [1]

Introduction

A domain decomposition finite element technique for efficiently generating lower and upper bounds to outputs which are linear functionals of the solution to the convection-diffusion equation is presented. The bound method is particularly useful to investigate characteristic quantities of a physical system. These quantities, which we term "outputs", must be expressed as functionals of the field solution obtained from numerical simulations. Large computational gains can be obtained if a fast and accurate method can provide the output value without accurately calculating the expensive field solution. For the past few years, the bound method has been developed [PPP97, Par97, PP98] to calculate, instead of the output value, upper and lower quantitative bounds to this output. The advantages of this approach are the reduced computational time by calculating an approximation of the field solution and the mathematical proof that the bounds are rigorous.

The bound method has been extended to address outputs of the Helmholtz equation, the Burgers equation and the incompressible Navier–Stokes equations in two space dimensions [PP99, MPP00]. Initial work has been performed to address sensitivity derivatives as well as reduced-order approximations to solve design optimization problems [LPP00, MMO^{+}00]. However, two key extensions are still desired: application to compressible flows and extension to three space dimensions. In this paper, we address the latter. The Ladeveze procedure used to approximate the hybrid flux between sub-domains in two space dimensions does not extend to three space dimensions. Therefore a new procedure is needed. We investigate the finite element tearing and interconnecting (FETI) procedure which is independent of dimensionality. This iterative method is ideal to approximate the hybrid flux in the bound method, i.e. the inter-sub-domain connectivity.

The FETI procedure is well established both in the literature as well as in commercial softwares [Far91, FR92, FCM95, FCRR98]. It was shown that, for structural problems, the FETI procedure outperforms direct and iterative algorithms. For parallel processing the FETI procedure becomes even more attractive; it provides parallel scalability. Furthermore, the application of the FETI procedure in the bound method permits simple modifications which drastically reduce the computational time and memory. To be more precise, all the inverse problems do not need to be solved exactly, only an order of magnitude reduction in the residual error suffice. Similarly, the FETI global iterations can also be limited to only a few iterations because only an approximation of the hybrid fluxes is needed. The contribution of this paper is the description of an inexpensive procedure to calculate the inter-sub-domain connectivity

[1] Department of Mechanical and Industrial Engineering, University of Toronto, marius@mie.utoronto.ca

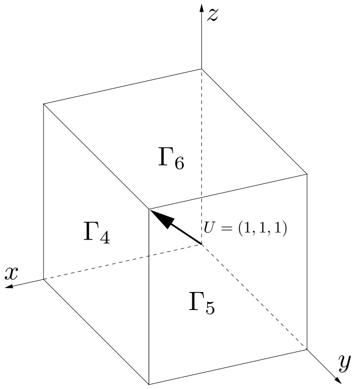


Figure 1: Convection-diffusion Geometry.

by exploiting simplifications made to the FETI method.

The convection-diffusion problem

The convection-diffusion problem is formulated in three space dimensions. This problem provides an example of a scalar non-symmetric problem,

$$-(\nu u_{,i})_{,i} + U_i u_{,i} = f \text{ in } \Omega,\ i = 1, ..., 3\ , \tag{1}$$

with inhomogeneous Dirichlet boundary conditions

$$u = g_D, \quad \text{on } \Gamma_D, \tag{2}$$

where ν is the positive viscosity and Ω is a bounded domain in $\mathbb{R}^3$.

The computational domain, Ω, is the unit cube, the six sides of which are denoted $\Gamma^j,\ j = 1, ..., 6$, as shown in Figure 1. We impose the boundary data on $g_D|_{\Gamma^4} = x_2 \times x_3,\ g_D|_{\Gamma^5} = x_1 \times x_3,\ g_D|_{\Gamma^6} = x_1 \times x_2$; and on $g_D|_{\Gamma^1} = g_D|_{\Gamma^2} = g_D|_{\Gamma^3} = 0$. The velocity is prescribed as $U = (1, 1, 1)$, and $f = 0$ to avoid any quadrature issues. Note that, for $U = (0, 0, 0)$ we recover the Poisson problem.

Bounds formulation

The bound method is based upon the construction of an augmented Lagrangian, in which the objective is a quadratic reformulation of the desired output, and the constraints are the finite element equilibrium equations and the inter-sub-domain continuity requirements. In the context of the bound method, computations are focused on evaluating a design quantity, i.e. an output. For simplicity, the particular linear functional investigated here is the average value of the field solution. Many engineering relevant linear functionals can be constructed, including the value at one point or the flux over a domain boundary [MPP99, Par97, PPP97].

For the discrete problems, we introduce a partition of the computational domain Ω into a set of N_k tetrahedra, $\mathcal{T}_H$. We also decompose each tetrahedron, $\Omega^{(k)}$, of $\mathcal{T}_H$, into a uniform refinement of tetrahedra $\mathcal{T}_h$ with characteristic diameter h.

Our discrete functional is constructed from the multiplication of the unit vector with the finite element mass matrix M, discretized on $\mathcal{T}_h$, $\ell = M\ \mathbf{1}$. The output of interest becomes $s = \ell^T u$, where u is an n-long vectors representing the discrete field solution to the Equations (1) and (2). Note that n is the number of nodes associated with the finite element discretization of u. For discretization, we exploit the finite element dimensional vector space X consisting of continuous piecewise linear functions on $\mathcal{T}_h$. For the convection-diffusion problem, the unknown nodal values of u, i.e. $\tilde{u}$, are obtained by solving the algebraic system

$$\tilde{L}\tilde{u} = \tilde{f} - b, \tag{3}$$

where $\tilde{L}$ is $r \times r$ non-symmetric positive-definite sparse matrix arising from the finite element discretization of the problem, and $\tilde{f}$ is a right-hand side r-long vector representing a prescribed force. The vector b contains the known data of u multiplied by L and transported to the right-hand side, i.e. the inhomogeneity. Clearly, the interior degrees-of-freedom r is less than n.

To avoid the expensive calculation of the system in (3), we introduce a discontinuous space $\hat{X}$ with jumps across the elements $\Omega^{(k)}$ and calculate bounds to s, i.e. $s_{LB} \leq s \leq s_{UB}$. Rigorous bounds are obtained by application of quadratic–linear duality theory [Str86], in which the candidate Lagrange multipliers are obtained from inexpensive calculations. The lower bound value is obtained from the Lagrangian, $\mathcal{L}(\hat{u}^{(k)}, \hat{\mu}, \hat{\lambda})$, where $\hat{\mu}$ and $\hat{\lambda}$ are approximations of the Lagrange multipliers. The lower bound to the output of interest, s, is

$$\begin{aligned} s_{LB} \quad &= \quad \sum_{k=1}^{N_k} \left(\hat{u}^{(k)^T} A^{(k)} \hat{u}^{(k)} - f^{(k)^T} \hat{u}^{(k)} + \ell^{(k)^T} \hat{u}^{(k)} \right) + C_u && (4) \\ & \quad -\hat{\mu}^T \sum_{k=1}^{N_k} \left(L^{(k)} \hat{u}^{(k)} - f^{(k)} \right) - \hat{\lambda}^T \sum_{k=1}^{N_k} B^{(k)} \hat{u}^{(k)}, && (5) \end{aligned}$$

where the superscript (k) is the restriction of the operator or the vector to the domain k. After simplifications we obtain

$$s_{LB} = -\sum_{k=1}^{N_k} \hat{u}^{(k)^T} A^{(k)} \hat{u}^{(k)} + C_u + f^{(k)^T} \hat{\mu}^{(k)}, \tag{6}$$

where all $\hat{u}^{(k)}$ are solutions of the decoupled local problems

$$2A^{(k)} \hat{u}^{(k)} = f^{(k)} - \ell^{(k)} + L^{(k)^T} \hat{\mu}^{(k)} - B^{(k)^T} \hat{\lambda}, \tag{7}$$

and $A^{(k)}$ is the finite element discretization of the symmetric term of $L^{(k)}$ and $B^{(k)}$ is the sign Boolean matrix which localizes the "jumps" at the interface. C_u is a boundary data value given by $C_u = G^T (L - A)\ G$, where G is a discrete function containing the Dirichlet boundary values and zero values elsewhere. In solving each Equation (7), care has to be taken to include the boundary condition for the elements lying on the boundary.

To guarantee solvability of each Equation (7), the candidate Lagrange multipliers must satisfy

$$(f^{(k)} - \ell^{(k)} + L^{(k)^T} \hat{\mu}^{(k)} - B^{(k)^T} \hat{\lambda}) \perp \mathrm{Ker}(A^{(k)}), \quad k = 1, ..., N_f, \tag{8}$$

where N_f is the number of pure Neumann problems.

The calculation of $\hat{\mu}$ has to be inexpensive such that the cost of calculating s_{LB} is considerabely less than the cost of calculating s. Hence, a coarse discretization $\mathcal{T}_H$ is exploited. We denote by X_H the corresponding conforming space of finite element functions, i.e. piecewise linear continuous functions in $\mathcal{T}_H$ including the Dirichlet boundary data. Following the bound method [Par97, PPP97], we solve

$$\tilde{L}_H \tilde{u}_H = \tilde{f}_H - b_H, \tag{9}$$

followed by

$$\tilde{L}_H^T \tilde{\mu}_H = 2\tilde{A}_H \tilde{u}_H - \tilde{f}_H + \tilde{\ell}_H + b_H, \tag{10}$$

where $\tilde{u}_H$ and $\tilde{\mu}_H$ are both in X_H. Note that there is no jump across the interface because of this continuous space. Afterward, $\hat{\mu}$ is interpolated on $\mathcal{T}_h$ to obtain $\hat{\mu} \in X$.

The FETI approach is employed to calculate the inter-sub-domain problems, i.e. calculation of $\hat{\lambda}$. Reformulating the FETI interface problem for the bounds gives

$$\begin{bmatrix} 2F_I & -G_I \\ -G_I^T & 0 \end{bmatrix} \begin{bmatrix} \hat{\lambda} \\ \alpha \end{bmatrix} = \begin{bmatrix} 2d \\ -e \end{bmatrix}, \tag{11}$$

where each of these terms is given by

$$F_I = \sum_{k=1}^{N_k} B^{(k)} A^{(k)^+} B^{(k)^T}, \tag{12}$$

$$G_I = \left[B^{(1)} R^{(1)} \ \ldots \ B^{(N_f)} R^{(N_f)} \right], \tag{13}$$

$$\alpha = \left[\alpha^{(1)} \ \ldots \ \alpha^{(N_f)} \right], \tag{14}$$

$$d = \sum_{k=1}^{N_k} B^{(k)} A^{(k)^+} (f^{(k)} - \ell^{(k)} + L^{(k)^T} \hat{\mu}^{(k)}), \tag{15}$$

$$e = \left[R^{(1)^T} f^{(1)} \right] \ \ldots \ \left[R^{(N_f)^T} f^{(N_f)} \right], \tag{16}$$

where $A^{(k)^+}$ is a generalized inverse of $A^{(k)}$ when the latter is singular. For subdomains with Dirichlet nodes, $A^{(k)^{-1}}$ is calculated. This domain decomposition based algorithm can be viewed as a two-step preconditioned conjugate gradient method to solve the interface problem [FCRR98]. The solution algorithm can be found abundantly in the literature [Far91, FR91, FR92, FCRR98, FM98, FCM95, Rix97]. Hence, it is not reviewed here.

We make several remarks regarding the computational simplifications in the context of the bound method. First, the inverse or the generalized inverse of $A^{(k)}$ is not calculated exactly, an iterative conjugate gradient solver is used. Note that this operation is need at each FETI iteration and such an approach may seem expensive. However, because we only require an approximation $\hat{\lambda}$, the conjugate gradient iterative procedure is terminated after the residual error is reduced by one order of magnitude. This approach requires less storage and fewer arithmetic operations than the Cholesky factorization used in the standard FETI approach. Note that this simplification is restricted to the bound method [FR91]. Second, we know that the null space, $R^{(k)}$, of the singular matrix $A^{(k)}$ is the unit vector avoiding the computational cost of its calculation. Third, the global FETI iterations can be stopped at any step and still provide rigorous bounds. Indeed, the constraint in the FETI interface problem (Equation 11) guarantees that the pure Neumann sub-domains are equilibrated. Numerical results will show that the sharpness of the bounds improves with the FETI iterations, however over solving the interface problem is not necessary as we will report in the numerical results Section.

Once $\hat{\mu}$ and $\hat{\lambda}$ are calculated, Equation (7) is solved for each subdomain to give $\hat{u}^{(k)}$ and finally s_{LB} is calculated from Equation (6). Similarly, the upper bound is obtained by taking the sign inverse of the lower bound of $-s$.

Numerical results

The convection-diffusion problem is investigated for the case where $f = 0$ and $\nu = 1/5$. The output of interest is the average of the solution on the fine discretization $\mathcal{T}_h$. This "triangulation" consists of 82,944 tetrahedron elements and 15,625 degrees-of-freedom. Three different coarse subdivisions are considered. The following notation, $\mathcal{T}_{(H,N)}$, is used to identify the coarse discretizations where N is the number of sub-domains per edge, i.e. $N \times N \times N \times 6$ sub-domains. Figure 2 presents, on the left, a coarse subdivision $\mathcal{T}_{(H,6)}$ and, on the right, the fine refinement $\mathcal{T}_h = \mathcal{T}_{(H,24)}$. A slice, at $z = 0.5$, of the finite element solution on $\mathcal{T}_{(H,6)}$ and of the reconstructed solution, $\mathcal{T}_{(H,N)}$, are presented in Figure 3.

To analyze the behavior of this method, we first report the values of the bounds and their convergence for different $\mathcal{T}_H$ meshes, Figure 4. The FETI iterations' stopping criterion is $\|r^n\|_2/\|r^0\|_2 < 10^{-2}$ where the numerator and the denominator are the n-th and the initial residual errors respectively. Obviously, for a given stopping criterion, the sharpness of the bounds depend on the richness of the coarse mesh. Recall that, the bound method guarantees that the output is within these values. Indeed, the expensive calculation on the fine mesh field solution is not required any longer as sufficient rigorous information is obtained for design.

Discussion of the computational cost of calculating the upper and lower bounds may be found in [Par00]. In this paper, we only pointout that the cost is related to the number of FETI iterations. As we have discussed previously, these iterations can be interrupted at any step and rigorous bounds can be calculated. The sharpness of the bounds depends on the number of iterations, or more precisely on the residual reached, as reported in Figure 5. The convergence curves show that the difference between of each bound and the fine mesh output value is considerably decreased in

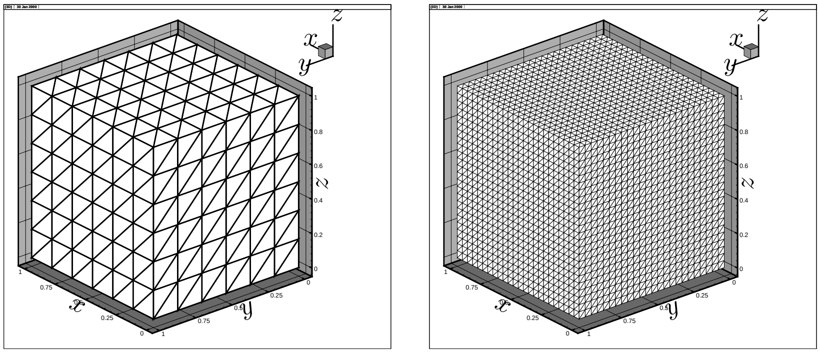


Figure 2: Two examples of meshes: (left) coarse mesh, $\mathcal{T}_{(H,6)}$; (right) fine mesh, $\mathcal{T}_h = \mathcal{T}_{(H,24)}$.

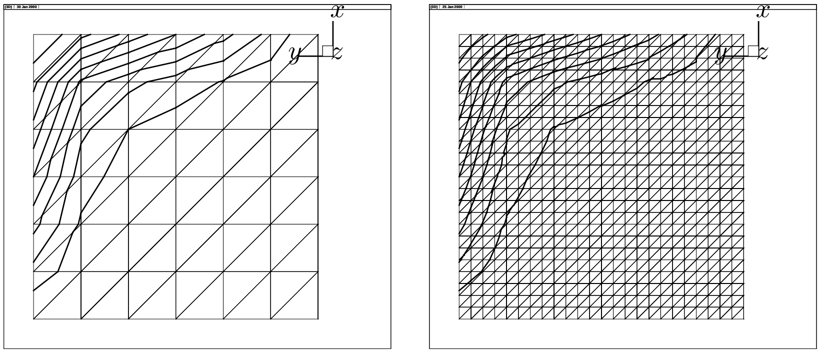


Figure 3: Isocontours (0 to 0.5 at intervals of 0.05) of (left) u_H on $\mathcal{T}_{H,6}$, and (right) $\hat{u}^{(k)},\ k = 1, ..., N_k$

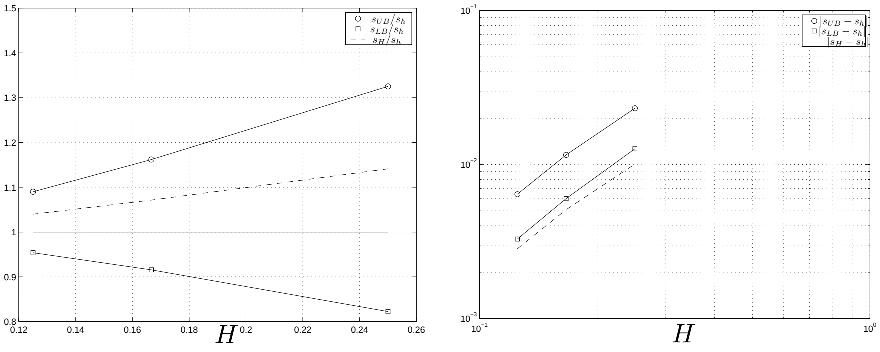


Figure 4: (left) Plots of s_{LB}/s_h, s_{UB}/s_h, and s_H/s_h as a function of the coarse mesh characteristic diameter H; (right) Log plots $|s_{LB} - s_h|$, $|s_{UB} - s_h|$ and $|s_H - s_h|$ as a function of H.

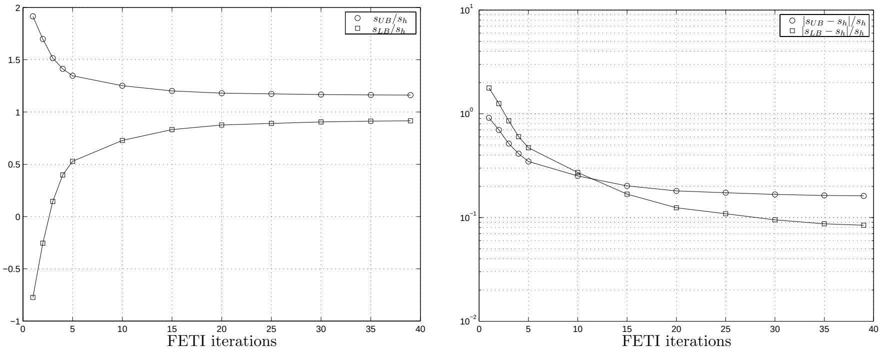


Figure 5: (left) Plots of s_{LB}/s_h and s_{UB}/s_h as a function of FETI iterations; (right) Log plots of $|s_{LB}-s_h|/s_h$ and $|s_{UB}-s_h|/s_h$ as a function of FETI iterations.

the first iterations. This indicates that an optimal number of iterations may exist. Both the hybrid flux and the adjoint approximations contribute to the bound gap. During the initial iterations, the bound gap is sensitive to the hybrid flux calculations. During later iterations, the adjoint interpolation dominates the bound gap. Clearly, resolving the interface problem more accurately will not improve the bounds because it does not improve the adjoint approximation. For improving the adjoint, there exists an adaptive approach to refine the coarse mesh in order to obtain the desired bound gap [PP97].

Acknowledgements

This work was supported in parts by the Natural Sciences and Engineering Research Council of Canada (NSERC). The author would like to thank Daniel Rixen for providing his FETI MATLAB script used for developing the C code employed in our computations.

References

[Far91]Charbel Farhat. A Lagrange multiplier based on divide and conquer finite element algorithm. *J. Comput. System Engng*, 2:149–156, 1991.

[FCM95]Charbel Farhat, Po-Shu Chen, and Jan Mandel. A scalable Lagrange multiplier based domain decomposition method for time-dependent problems. *Int. J. Numer. Meth. Eng.*, 38:3831–3853, 1995.

[FCRR98]C. Farhat, P.-S. Chen, F. Risler, and F.-X. Roux. A unified framework for accelerating the convergence of iterative substructuring methods with Lagrange multipliers. *Int. J. Numer. Meth. Engng.*, 42:257–288, 1998.

[FM98]C. Farhat and J. Mandel. The two-level feti method for static and dynamic plate problems - part i: an optimal iterative solver for biharmonic systems. *Computer Methods in Applied Mechanics and Engineering*, 155:129–152, 1998.

[FR91]Charbel Farhat and Francois-Xavier Roux. A Method of Finite Element Tearing and Interconnecting and its Parallel Solution Algorithm. *Int. J. Numer. Meth. Engng.*, 32:1205–1227, 1991.

[FR92]C. Farhat and F.X. Roux. An unconventional domain decomposition method for an efficient parallel solution of large-scale finite element systems. *SIAM J. Sc. Stat. Comput.*, 13:379–396, 1992.

[LPP00]R. M. Lewis, A. T. Patera, and J. Peraire. *A Posteriori* finite element bounds for sensitivity derivatives of partial-differential-equation outputs. *Finite Element Analysis & Design*, 34(3-4):271–290, 2000.

[MMO$^+$00]L. Machiels, Y. Maday, I. B. Oliveira, A. T. Patera, and D. V. Rovas. Output bounds for reduced-basis approximations of symmetric positive definite eigenvalue problems. *C. R. Acad. Sci. Paris, Série I*, 2000.

[MPP99]Y. Maday, A.T. Patera, and J. Peraire. A general formulation for *a posteriori* bounds for output functionals of partial differential equations; application to the eigenvalue problem. *C.R. Acad. Sci. I-Math.*, 328(9):823–828, 1999.

[MPP00]L. Machiels, J. Peraire, and A.T. Patera. *A Posteriori* finite element output bounds for incompressible Navier-Stokes equations; application to a natural convection problem. *Journal of Computational Physics*, 2000.

[Par97]M. Paraschivoiu. *A Posteriori finite element bounds for linear-functional outputs of coersive partial differential equations and of the Stokes problem.* PhD thesis, Massachusetts Institute of Technology, 1997.

[Par00]M. Paraschivoiu. A posteriori finite element output bounds in three space dimensions using the FETI method. *submitted to Comp. Meth. Appl. Mech. Engrg.*, 2000.

[PP97]J. Peraire and A.T. Patera. Bounds for linear-functional outputs of coercive partial differential equations: Local indicators and adaptive refinement. In P.Ladeveze and J.T. Oden, editors, *Workshop On New Advances in Adaptive Computational Methods in Mechanics.* Elsevier, 1997.

[PP98]M. Paraschivoiu and A. T. Patera. A hierarchical duality approach to bounds for the outputs of partial differential equations. *Comp. Meth. Appl. Mech. Engrg.*, 158:389–407, 1998.

[PP99]J. Peraire and A. T. Patera. Asymptotic *a posteriori* finite element bounds for the outputs of noncoercive problems: the Helmholtz and Burgers equations. *Comp. Meth. Appl. Mech. Engrg.*, 171:77–86, 1999.

[PPP97]M. Paraschivoiu, J. Peraire, and A. T. Patera. *A Posteriori* finite element bounds for linear–functional outputs of elliptic Partial Differential Equations. *Comput Methods Appl. Mech. Engrg.*, 150:289–312, 1997.

[Rix97]D. Rixen. *Substructuring and dual methods in structural analysis.* PhD thesis, University of Liège, Belgium, 1997.

[Str86]G. Strang. *Introduction to Applied Mathematics.* Wellesley-Cambridge Press, Wellesley, Massachusetts, 1986.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau,

21. Efficient and fast numerical methods to compute fluid flows in the geophysical β plane

T. SAKAJO [1]

Introduction

We consider a fluid flow in a rotating sphere with an unit radius. The flow is incompressible and inviscid, and covers the sphere with a constant density. This kind of flow is called a geophysical flow, since it is one of the simplest models of atmospheric flows in the earth. In practical study of geophysical flows, we are sometimes interested in a local flow in the neighborhood of a certain point in the sphere. In that case, we consider flows in a plane which is tangent to the point as an approximation model. The plane is called the geophysical β plane. In the present article, we introduce an equation which describes a motion in the β plane. And a numerical procedure to compute the equation is formulated. Furthermore, we suggest an efficient technique to compute it fast and accurately by using a fast algorithm and a parallelization based on the idea of Domain Decomposition. As an example of its application, we compute a two-dimensional flow problem in the β plane and investigate the effectiveness of the fast method and the effect of rotation on the evolution numerically.

Numerical computations of the geophysical flows play an important role in the atmospheric research, such as the weather forecast and the investigation of the environmental issues. In spite of its importance, it is not easy to obtain useful and practical results since it costs too much to compute these problems for sufficiently fine resolutions. That is why a fast and accurate numerical method is required. The purpose of the study is to give an efficient numerical method to compute such flows and to show its effectiveness by applying it to some fluid problem.

In the next section, we suggest the fast numerical method: We consider the equation of the flows in the β plane, whose detailed definition and formulation is explained. Our numerical method called the point potential vortex method is introduced. Then, some techniques to compute it fast and accurately are appearing. In the third section, we show some results of the numerical computation of a fluid flow in the β plane: (1) effectiveness of the fast method and (2) investigation of the influence of rotation on the evolution of the flow. The last section is conclusions.

Numerical methods

The equation of motion of fluids in the β plane

Now, we introduce an equation of motion of the flows in the geophysical β plane. Let ϕ and λ be a latitude and longitude of a point in the sphere, respectively. When fixing a point (λ_0, ϕ_0) in the sphere, we consider the plane which is tangent to the point and

[1] Graduate school of mathematics, sakajo@math.nagoya-u.ac.jp

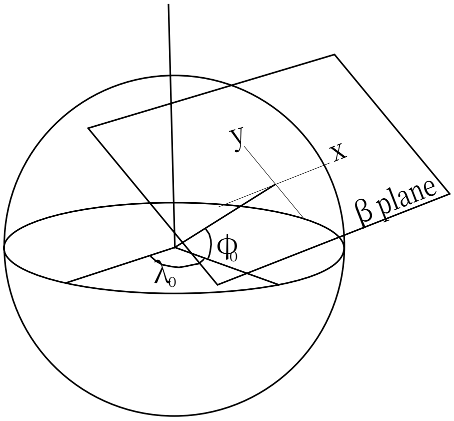


Figure 1: The β plane associated with the point (λ_0, ϕ_0) in the sphere.

introduce a new pair of variables (x, y) as follows (See Figure 1):

$$x = (\cos \phi_0)\lambda, \qquad y = \phi - \phi_0, \qquad (|x|, |y| << 1).$$

We define the stream function Ψ and the vorticity ω in the β plane as $\omega = \mathrm{rot}\mathbf{u}, \Delta\Psi = -\omega$. The velocity field $\mathbf{u}$ is recovered from the stream function by the formula $\mathbf{u} = (-\partial_y \Psi, \partial_x \Psi)$, and Δ is the two-dimensional Laplacian. Then, the equation of incompressible Euler flow in the β plane is given by

$$\frac{\partial}{\partial t}\Delta\Psi + \frac{\partial(\Psi, \Delta\Psi)}{\partial(x, y)} + \beta\frac{\partial\Psi}{\partial x} = 0, \tag{1}$$

where the second term is two-dimensional Jacobian:

$$\frac{\partial(a, b)}{\partial(x, y)} = \frac{\partial a}{\partial x}\frac{\partial b}{\partial y} - \frac{\partial a}{\partial y}\frac{\partial b}{\partial x}.$$

The equation differs from the usual two-dimensional Euler equation in the effect of rotation (β-effect) of the third term. Therefore, the vorticity is no longer an invariant quantity along the trajectory of the fluid element just like in two-dimensional case. However, we can define a “potential vorticity” by

$$q = \omega + \beta y.$$

Then, substituting it to the equation (1), we obtain the following simple equations:

$$\frac{Dq}{Dt} = (\frac{\partial}{\partial t} + u\frac{\partial}{\partial x} + v\frac{\partial}{\partial y})q = 0 \tag{2}$$

$$\Delta\Psi = -\omega, \qquad (u, v) = (-\partial_y \Psi, \partial_x \Psi), \tag{3}$$

where $\frac{D}{Dt}$ represents the derivative along the trajectory of the fluid element which moves together with the fluid flows (material derivative).

What the equation (2) indicates is the potential vorticity is invariant along the trajectory of the fluid element. That means, when $(x(t), y(t))$ is a position of the fluid element at some time t, the potential vorticity at the position is given by the initial potential vorticity at $(x(0), y(0))$:

$$q(x(t), y(t), t) = q(x(0), y(0), 0) \equiv q_0.$$

Hence, the vorticity ω at the position $(x(t), y(t))$ is represented by

$$\omega(x(t), y(t), t) = q_0 - \beta y(t). \tag{4}$$

Based on the considerations in the section, we show a numerical method and some techniques to compute the evolution of the flows in the β plane fast and accurately in the following subsections.

The point potential vorticity method

At first, the velocity field is obtained by solving the Laplace equation (3). For the sake of simplicity, we impose the periodic boundary condition in the x direction. Then, the velocity field $(u, v) = (-\partial_y\Psi, \partial_x\Psi)$ are given by

$$\begin{aligned} u(x,y,t) &= -\frac{1}{2}\int \frac{\omega(\tilde{x},\tilde{y},t)\sinh 2\pi(y-\tilde{y})}{\cosh 2\pi(y-\tilde{y}) - \cos 2\pi(x-\tilde{x})} d\tilde{x}d\tilde{y}, \\ v(x,y,t) &= \frac{1}{2}\int \frac{\omega(\tilde{x},\tilde{y},t)\sin 2\pi(x-\tilde{x})}{\cosh 2\pi(y-\tilde{y}) - \cos 2\pi(x-\tilde{x})} d\tilde{x}d\tilde{y}. \end{aligned} \tag{5}$$

We must note that the integrals on the right hand side are singular integral. What follows is our numerical procedure.

1. We discretize the computational domain which includes the vorticity field. Then we obtain the discretization points $\{(x_n, y_n)\}, (n = 1, ..., N)$. We must discretize a sufficiently large region including no vorticity at the beginning because of generation of the "ghost vorticity", which we will explain later section.

2. We assume that the vorticity concentrates in the discretization points, (Point potential vortices).Then we approximate the vorticity field as follows:

$$\omega(x,y,t) = \sum_{n=1}^{N}(q_{n0} - \beta y_n)\delta(x - x_n, y - y_n), \tag{6}$$

   where q_{n0} is initial potential vorticity and δ is Dirac's delta function.

3. Substituting (6) to (5), we compute the velocity field induced by the vorticity.

$$\begin{aligned} u_N(x,y,t) &= -\frac{1}{2N}\sum_{n=1}^{N}\frac{(q_{n0} - \beta y_n)\sinh 2\pi(y-y_n)}{\cosh 2\pi(y-y_n) - \cos 2\pi(x-x_n)}, \\ v_N(x,y,t) &= \frac{1}{2N}\sum_{n=1}^{N}\frac{(q_{n0} - \beta y_n)\sin 2\pi(x-x_n)}{\cosh 2\pi(y-y_n) - \cos 2\pi(x-x_n)}. \end{aligned} \tag{7}$$

4. The point potential vortices evolves by the induced velocity field:

$$\frac{dx_n}{dt} = u_N(x_n, y_n), \quad \frac{dy_n}{dt} = v_N(x_n, y_n), \quad (n = 1, ..., N).$$

We use the 4-th order Runge-Kutta method to compute the step 4. We refer the numerical scheme as the "point potential vortex method".

The fast algorithm and parallel implementation

Although the idea of the point potential vortex method is simple, it has not been easily applied to simulations of practical geophysical flows so far because of some difficulties. Here, we explain these difficulties and show some methods to overcome them.

Desingularization of the equation As we note in the previous section, the velocity field (5) is given as a singular integral. Due to the singularity, the round-off error has a seriously bad influence on numerical solutions. To get rid of the bad influence of the round-off error, we use the Krasny's desingularization technique[Kra86]. That is, taking a sufficiently small positive real number ϵ, we compute the following desingularized summation instead of (7):

$$\begin{aligned} u_N^\epsilon(x,y,t) &= -\frac{1}{2N}\sum_{n=1}^{N} \frac{(q_{n0} - \beta y_n)\sinh 2\pi(y-y_n)}{\cosh 2\pi(y-y_n) - \cos 2\pi(x-x_n) + \epsilon^2}, \\ v_N^\epsilon(x,y,t) &= \frac{1}{2N}\sum_{n=1}^{N} \frac{(q_{n0} - \beta y_n)\sin 2\pi(x-x_n)}{\cosh 2\pi(y-y_n) - \cos 2\pi(x-x_n) + \epsilon^2}. \end{aligned} \tag{8}$$

We can compute the velocity field (8) stably since they are bounded as long as $\epsilon \neq 0$.

Fast summation method Let N be the number of point potential vortices which is obtained by the discretizaion of the computational domain. The amount of computation which is required to compute the velocity field (8) for a point is $O(N)$. Therefore, it takes $O(N^2)$ operations to compute the Step 4 for all the points. Due to the rapid increase of the total amount of operations, it is difficult to use high resolution in the practical numerical computation. In order to overcome the difficulty, we apply the Draghicescu's fast algorithm. This algorithm reduces the total amount of operations to $O(N \log N)$, allowing approximation error to some extent. However, the method works well for a large number of N, since the approximation error reduces with $O(\frac{1}{N})$ and is negligible as N increases. As for the detailed algorithms and how to apply the algorithm to the periodic boundary condition, we would like the readers to refer to Draghicescu[Dra94] and Sakajo & Okamoto[SO98], respectively.

Parallel implementation We implement the fast numerical algorithm to a parallel computer. Now when the parallel computer has p CPUs, we assign $\frac{N}{p}$ point potential vortices to each processor and then compute the evolutions concurrently. Since the point potential vortices are obtained by discretizing the computational domain, the parallelization would be one of the Domain Decomposition techniques. In the present computation, we use a distributed parallel computer with four DEC Alpha 21264 processors.

N+M	8192	32768	131072
direct	143	2288	36789
fast	25	120	463
parallel+fast	12	51	156

Table 1: The elapsed time to compute the velocity field (8) in seconds

Results

We apply the point potential vortex method to the computation of a vortex sheet in the β plane. A vortex sheet is a surface across which the velocity of the fluid changes discontinuously. That means that initially the vorticity exists only in the vortex sheet, and outside of the vortex sheet there exists no vorticity. In the two-dimensional β plane, the vortex sheet is represented by a one-parameter curve: $(x(\Gamma,t), y(\Gamma,t))$, where Γ is circulation parameter along the sheet and t is time. We imposed the periodic boundary condition on the sheet as follows,

$$x(\Gamma+1,t) = x(\Gamma,t)+1, \qquad y(\Gamma+1,t) = y(\Gamma,t), \qquad (0 \le \Gamma < 1).$$

A flat vortex sheet $(x,y) = (\Gamma, 0)$ is a steady state. We add a small disturbance to the steady state and take it as an initial condition of the numerical computation:

$$x(\Gamma,t) = \Gamma + 0.01\sin 2\pi\Gamma, \qquad y(\Gamma,t) = -0.01\sin 2\pi\Gamma.$$

If we consider the ordinary two-dimensional vortex sheet, we have only to discretize the vortex sheet since the vorticity is invariant. However, since not the vorticity but the potential vorticity is invariant in the β plane approximation, the vorticity could be created as a result of the vertical movement of the point potential vortices even if it has no vorticity at the beginning. The created vorticity is called the ghost vorticity. Therefore, we have to discretize the sufficiently large regions to the y directions in this case by taking the creation of ghost vorticity into considerations.

We discretize the vortex sheet along the sheet and obtain N point potential vortices $(x_n, y_n), (n = 1, ..., N)$, whose initial position is

$$x_n(0) = \frac{n}{N} + 0.01\sin 2\pi\frac{n}{N}, \qquad y_n(0) = -0.01\sin 2\pi\frac{n}{N}, \qquad (n = 1, ..., N).$$

and initial potential vorticity is $q_{n0} = \frac{1}{N} + \beta y_n(0)$. We also discretize the outer regions by grids and obtain M point potential vortices, $(\tilde{x}_n(t), \tilde{y}_n(t))$, whose initial potential vorticity is $\tilde{q}_{n0} = 0 + \beta\tilde{y}_n(0)$, $(1 \le n \le M)$.

Effectiveness of the fast computation

We show the effectiveness of the fast algorithm and parallelization. The desingularization parameter is fixed to 0.1. Table 1 shows the elapsed time to compute the velocity field (8) in second when we change the number of discretization $N + M$. When we use the direct summation of $O(N^2)$, the computational time increases rapidly. On the other hand, the time is very small when we use the fast summation method. It takes

N+M	8192	32768	131072
error	2.26e-07	5.35e-08	9.46e-09

Table 2: Maximum approximation error of the fast algorithm

about 80 times faster than the direct summation when $N = 131072$ point potential vortices are used. Moreover, as a result of the implementation of the fast algorithm to the parallel computer, we achieve more than 230 times faster computation for $N = 131072$ points are used. Table 2 shows the maximum approximation error of the fast algorithm. As $N + M$ increase, the error gets smaller. The fast algorithm yields very accurate computation for a large number of point potential vortices.

These two results indicates that the more we use point potential vortices the more accurate and faster we can compute the velocity field.

An application - a vortex sheet in the β plane

We discretize the vortex sheet by 65536 points and the other no vorticity region $[0, 1] \times [-2.0, 2.0]$ by 128×512 grid points. The desingularization parameter ϵ is 0.1. We change only the parameter β to see the effect of rotation to the evolution.

Figure 2 shows the time evolution of the vortex sheet: (a) $\beta = 0$ (no rotation), (b) $\beta = 5$ (mild rotation) and (c) $\beta = 10$ (fast rotation). At first, when there exists no β-effect (column (a)), it evolves in the same way as the two-dimensional vortex sheet, which Krasny computed[Kra86]. Nearly flat vortex sheet becomes unstable and roll-up and then generates the spiral structure in the middle of the region. Next, when $\beta = 5$ (column (b)), it forms the spiral structure as well. However, the center of the spiral moves toward the northwest. This movement is due to the effect of rotation, which is well-known as the Rosby effect. Note that the number of winding of the spiral becomes few. At last, when $\beta = 10$ (column (c)), it begins forming the spiral structure at $t = 1.0$ but it hardly grows. Instead, an another spiral structure is generated at $t = 2.0$. The result indicates that faster rotation results in the appearance of the new spiral structure, which would be a new feature of the β-effect.

Conclusions

We suggest the point potential vortex method to compute the geophysical fluids in the β plane numerically. The fast summation method and the implementation to the parallel computer based on the Domain Decomposition approach makes us possible to execute the numerical computation accurately and fast. The hybrid combination of these two fast numerical method brings us a possibility to try the numerical computations of various practical geophysical flows.

As one of the examples, we apply the numerical scheme to the computation of the vortex sheet in the β plane. We find the northwestward movement of the spiral structure and the appearance of the new spiral structure due to the effect of rotation. The analytic investigation of these phenomena remains in the future.

The point potential vortex method could be extended to the case of the flows in the rotating sphere. The formulation is the same as the present method. That is, the potential vorticity is also preserved along the trajectory of the fluid element. However, since there is no fast summation method to compute the velocity field fast, the extension wouldn't be completed. The development of the fast algorithm for the velocity field in the sphere is challenging.

References

[Dra94]C. I. Draghicescu. An efficient implementation of particle methods for the incompressible euler equations. *SIAM J. Numer. Anal.*, 31(6):1090–1108, 1994.

[Kra86]R. Krasny. Desingularization of periodic vortex sheet roll-up. *J. Comput. Phys.*, 167:65–93, 1986.

[SO98]T. Sakajo and H. Okamoto. An application of draghicescu's fast summation algorithm to vortex sheet motion. *J. Phys. Soc. Japan*, 67(2):462–470, 1998.

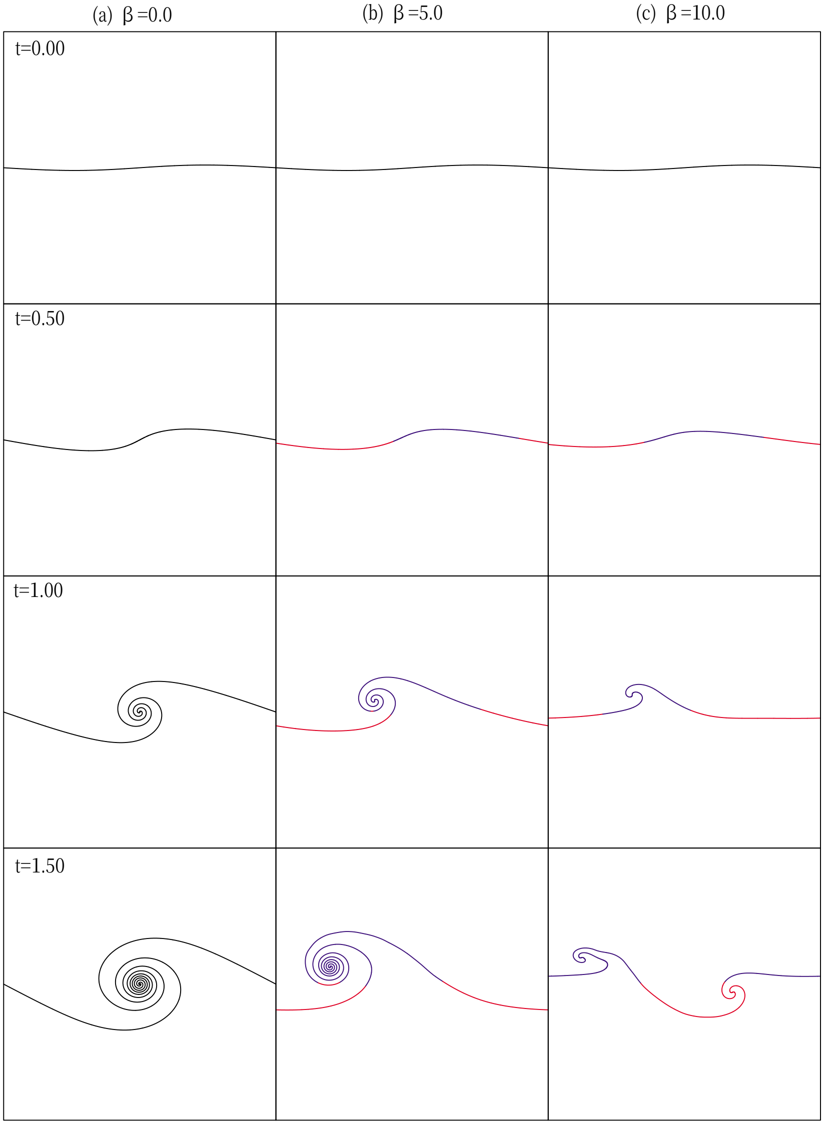


Figure 2: The time evolution of the vortex sheet in the β plane. (a) $\beta = 0.0$, (b) $\beta = 5.0$ and (c) $\beta = 10.0$

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau, ©2001 DDM.org

22. Boundary Element Scheme with Domain Decomposition Approach for Moving Interface Phenomenon

R. Sugino[1], H. Imai[2], N. Tosaka[3]

Introduction

In many situations of hydrodynamic phenomenon, two-layer fluid is a dominant feature of fluid motion. Such two-layer fluid often contains jumps in the density across the interface of fluid domains. The interface of air and water or salt water and fresh water is obvious example [Tho68]. The density jump may be assumed to occur in an infinitesimally thin interface in the mixture. And, the behavior of such an interface is important to understand various hydrodynamic phenomena [DR81].

In our previous study, we developed the efficient numerical procedure which is able to simulate the time evolution of an interface between two fluids with different densities. The numerical solution procedure based on the sub-region boundary element method with the mixed Eulerian - Lagrangian approach developed in the moving boundary problems in potential field [ST93]. Now, cluster computation by work stations or personal computers becomes available in many laboratories all over the world. Because of their potential for both high-performance and cost-effectiveness, cluster computing will attract much more attention of researches, and they will take the most important part in engineering computation in stead of vector computing in near future. Under this situation, investigation of parallel FEM algorithm [GDP83],[GASS93] based on the DDM is increasing. Recently, Kamiya et al. introduced DDM for the boundary element analysis in order to implement the parallel BEM computation [DM96],[KIK96]. They showed the utility of BEM analysis with domain decomposition scheme for some potential and elastic problems.

In this paper, we propose a new boundary element procedure for the density stratified flow based on the domain decomposition method. The final system of equations for the whole region is obtained by adding the set of boundary integral equations of governing equation for each sub-domain in conjunction with compatibility and equilibrium conditions between their interfaces. The present study is an attempt to develop parallel computation procedure for the interface motion of the two-layer fluid in a rectangular region based on domain decomposition method and two sub-domain boundary element method.

Mathematical Modeling of Moving Interface Flow

As shown in Figure1, Ω_1 and Ω_2 denote the portions of flow domain occupied by fluids 1 and 2, respectively. The fluid regions are separated by a sharp interface. Here, the

[1] Anan College of Technology,sugino@anan-nct.ac.jp
[2] Tokushima University,imai@pm.tokushima-u.ac.jp
[3] Nihon University,n7tosaka@ccu.cit.nihon-u.ac.jp

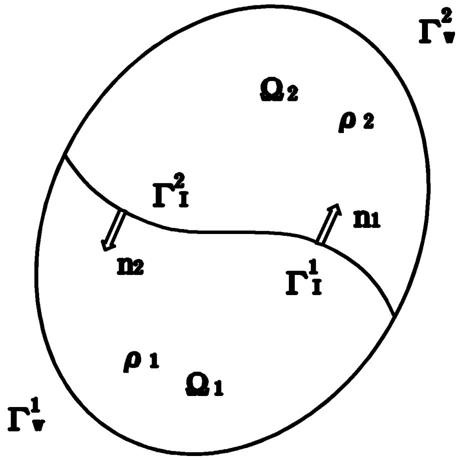


Figure 1: Geometrical configuration of problem.

subscript i denotes each flow region. And ρ denotes the density of fluids.

In this model, we assume the existence of velocity potentials $\phi_i\,(x, y, t)\,(i = 1, 2)$ in the fluids both sides of the interface. Then, the governing equations for the velocity vector $\boldsymbol{u}_i = (u_i, v_i)$ are given as follows:

$$\nabla^2\phi_i = \frac{\partial^2\phi_i}{\partial x^2} + \frac{\partial^2\phi_i}{\partial y^2} = 0 \qquad \text{in} \;\; \Omega_i \;\; (i = 1, 2), \tag{1}$$

$$\boldsymbol{u}_i = \nabla\phi_i \qquad \text{in} \;\; \Omega_i \;\; (i = 1, 2), \tag{2}$$

where $\nabla = (\partial/\partial x, \partial/\partial y)$ and ∇^2 denotes the two-dimensional Laplacian.

There are two kinds of the boundary conditions to be prescribed. The first one is the wall boundary condition given by

$$\frac{\partial\phi_i}{\partial n} = \boldsymbol{n} \;\cdot \nabla\phi_i = 0 \qquad \text{on} \;\; \Gamma_w^i \;\; (i = 1, 2), \tag{3}$$

where $\boldsymbol{n}$ denotes the outward unit normal vector on the boundary. The other is the so-called moving boundary conditions on the moving interface Γ_I. They are the kinematic and dynamic conditions. As the mathematical expressions of these conditions, we introduce its Lagrangian description in terms of the Lagrangian coordinates (ξ_i, η_i)for a marked particle on the moving interface. Consequently, the liquid particles on the interface must move with the interface in each domain. Then, the kinematic conditions for a particle are given by

$$\left.\begin{aligned} \frac{D\xi_i}{Dt} &= u_i = \frac{\partial\phi_i}{\partial x} \\ \frac{D\eta_i}{Dt} &= v_i = \frac{\partial\phi_i}{\partial y} \end{aligned}\right\} \qquad \text{on} \;\; \Gamma_I^i \;\; (i = 1, 2), \tag{4}$$

where D/Dt is used to express the Lagrangian derivative.

Next, we also can express the dynamic condition derived from Bernoulli's equation as the following equation on rate of change of ϕ_i :

$$\frac{\partial \phi_i}{\partial t} + \frac{1}{2}\left\{ \left(\frac{\partial \phi_i}{\partial x}\right)^2 + \left(\frac{\partial \phi_i}{\partial y}\right)^2 \right\} + g\eta_i + P_i/\rho_i = 0 \qquad \text{on } \Gamma_I^i \ \ (i = 1, 2), \qquad (5)$$

where g is the acceleration of gravity, and P_i are the pressure on the interface ${\Gamma_I}^i$.

The interfacial conditions should be introduced to this model. To require that two fluids do not separate or cross over at the interface, we must set the following kinematic condition :

$$\frac{\partial \phi_1}{\partial n} = -\frac{\partial \phi_2}{\partial n} \qquad \text{on } {\Gamma_I}^i \ \ (i = 1, 2). \qquad (6)$$

Next, the normal stress of the fluid is to be continuous at the interface. For an inviscid fluid, this means satisfaction of the following dynamical condition that the pressure is continuous:

$$P_1 = P_2 \qquad \text{on } {\Gamma_I}^i \ \ (i = 1, 2). \qquad (7)$$

In this paper, we consider the mathematical model given by equations (1)-(7) as the coupled problem of the boundary-value problem of Laplace equation (1) and the initial-value problem of the system of evolutional equations (4) and (5).

Theory of DD Approach for Moving Interface in Flow Region

DD Approach for Boundary Element Scheme

Let us consider the two layer flow with a moving interface in a domain Ω, which is decomposed into two sub-domains Ω_1 and Ω_2 as shown in Figure 2. Here, we can easily transform the field equation (1) into the following boundary integral equation by taking into consideration with the linearity of Laplace equation (1):

$$\int_{\Gamma_i} \phi_i(\,\boldsymbol{x}\,)\frac{\partial \omega_i^*}{\partial n}(\,\boldsymbol{x}\,,\ \boldsymbol{y}\,)d\Gamma(\,\boldsymbol{x}\,) = \int_{\Gamma_i} \frac{\partial \phi_i}{\partial n}(\,\boldsymbol{x}\,)\omega_i^*(\,\boldsymbol{x}\,,\ \boldsymbol{y}\,)d\Gamma(\,\boldsymbol{x}\,) \qquad (i = 1, 2), \quad (8)$$

in which ω^* is the well-known fundamental solution given by

$$\omega^*(\,\boldsymbol{x}\,,\ \boldsymbol{y}\,) = \frac{1}{2\pi}\ln\frac{1}{r} \quad , \quad r = \|\ \boldsymbol{x}\ -\ \boldsymbol{y}\ \|\ . \qquad (9)$$

If Dirichlet data on the moving interface Γ_I is known, then we can determine its derivative on Γ_I with solution of the above boundary integral equation. In order to solve approximately (8), we use the BE scheme.

In DD approach of (1),(6) and (7), several formulations can be derived according to treatment of inter boundary conditions of (6) and (7). In this study, we employ the continuity of Dirichlet data (i.e., velocity potential) and Neumann data (i.e., normal velocity) as follows:

$$\phi_2 = \alpha\phi_1 + \beta \qquad \text{on } \Gamma_I, \qquad (10)$$

$$\frac{\partial \phi_1}{\partial n} + \frac{\partial \phi_2}{\partial n} = 0 \qquad \text{on } \Gamma_I. \tag{11}$$

To treat the inter-subdomain boundary condition, the Lagrange multiplier λ is introduced as follows:

$$\phi_1 = \lambda = \frac{1}{\alpha}\phi_2 - \frac{\beta}{\alpha} \qquad \text{on } \Gamma_I. \tag{12}$$

Applying the above conditions to (8), the following inverse formulation is derived:

$$\sum_{i=1}^{2}\left[\int_{\Gamma^i_{I+w}} \phi_i \frac{\partial \omega_i^*}{\partial n} d\Gamma - \int_{\Gamma^i_{I+w}} \frac{\partial \phi_i}{\partial n} \omega_i^* d\Gamma\right] + \int_{\Gamma_I}\left(\frac{\partial \phi_1}{\partial n} + \frac{\partial \phi_2}{\partial n}\right)\delta\lambda d\Gamma = 0. \tag{13}$$

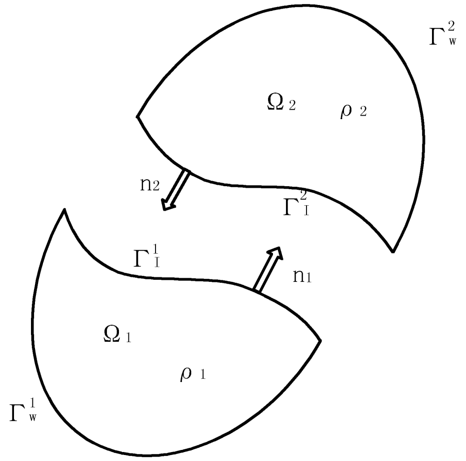


Figure 2: Problem splitted into two sub-domains.

Uzawa's Algorithm for DD Approach

Equation (13) consists of the usual boundary integral forms for the subdomain Ω_i and the constrain term derived from the energy equilibrium as well as the normal velocity continuity among subdomains. Uzawa's method[GDP83], which is one of iterative solution techniques, is employed here to solve (13).

Uzawa's algorithm is summarized as follows:

- `STEP 1:Initialization`

$$\lambda^0 = \widehat{\lambda}(\text{: constant}). \tag{14}$$

- `STEP 2:Computation in each subdomain :`

$$\sum_{i=1}^{2}\left[\int_{\Gamma^i_{I+w}} \phi_i \frac{\partial \omega_i^*}{\partial n} d\Gamma - \int_{\Gamma^i_{I+w}} \frac{\partial \phi_i}{\partial n} \omega_i^* d\Gamma\right] + \int_{\Gamma_I} \tau^n \lambda^n d\Gamma = 0, \tag{15}$$

$$\tau^n = \frac{\partial \phi_1}{\partial n} + \frac{\partial \phi_2}{\partial n} \quad \text{on} \quad \Gamma_I, \tag{16}$$

where τ denotes the residual value which is the continuity of flux.

In this step, we solve the Laplace equations under following boundary conditions:

$$\frac{\partial \phi_i}{\partial n} = 0 \quad \text{on} \quad \Gamma_w^i, \tag{17}$$

$$\phi_1 = \lambda^n, \quad \phi_2 = \alpha\lambda^n + \beta \quad \text{on} \quad \Gamma_I^i, \tag{18}$$

where superscript n indicates the n-th iterative step.

- `STEP 3:Modification of lagrange multiplier` λ^n

$$\lambda^{n+1} = \lambda^n + \omega\tau^n, \tag{19}$$

where ω denotes the convergence coefficient.

- `STEP 4:Judgement of convergence`

The criterion for convergence employed here is:

$$\int_{\Gamma_I} \frac{\tau^n \cdot \tau^n}{\tau^0 \cdot \tau^0} d\Gamma \leq \epsilon. \tag{20}$$

If λ^n has not converged yet, return to `STEP 1` by setting $n \leftarrow n+1$.

By implementation of the above iterative method, we can get the potential ϕ_i^{k+1} and $(\partial\phi_i/\partial n)^{k+1}$ on the ${\Gamma_I}^i$ and use these values for estimation of interfacial dynamics.

Formulation for Moving Interface Computations

We will determine the particles on the interface whose velocities $(\partial X_I/\partial t,\ \partial Y_I/\partial t)$ are a mean values of velocities of the two fluids. The kinematic condition (4) is modified to the following forms as given by :

$$\left.\begin{aligned} \frac{\partial X_I}{\partial t} &= \left[(1+\beta)\frac{\partial \phi_1}{\partial x} + (1-\beta)\frac{\partial \phi_2}{\partial x}\right]/2 \\ \frac{\partial Y_I}{\partial t} &= \left[(1+\beta)\frac{\partial \phi_1}{\partial y} + (1-\beta)\frac{\partial \phi_2}{\partial y}\right]/2 \end{aligned}\right\}, \tag{21}$$

where β is the constant in which $\beta = +1$ corresponds to the lower fluid, $\beta = -1$ is to the upper fluid and $\beta = 0$ is to mid-interface particles. In this computation, we adopt $\beta = 0$, and the velocity of interface is mid-interface particles of the layer. This system to be considered as the one of first-order ordinary differential equations can be solved approximately by using the time integration scheme. Applying the Euler scheme to the above system, we can determine the new value of ξ and η at the $(k+1)$-th time step. The procedure can be repeated to track the time history of the interface movement.

Numerical Experiments and Evaluations

In order to examine applicability of our method proposed, we show the obtained numerical results. We simulate the motion of two different density fluids in which are stratified for the vertical direction under gravitational force g. Two fluids are settled in the rectangular container with non-dimensional width, $L = 0.04$ and height, $H = 0.06$. And, the container is filled with the lower fluid to a height, $h = 0.03$ at the stationary state. This interface is initially flat, but a perturbation is supplied by specifying the y-coordinate component of its position at the interface as $\delta y = A_0 \cos(\pi x/L)$. Numerical computations are carried out for the case given by parameters such that $A_0 = 0.0001$,$g = 1.0$ and $\Delta t = 0.005$. The fluid domains is divided into 50 boundary segments and both interface parts are divided into 20 segments, respectively.

In Figure3, we show the profiles of interface at each time step in the case of $\rho_1/\rho_2 = 1.0/2.0$ as density ratio of the two fluids. The pertubation drives the unstable fluid interface, causing it to flow down along the right edge of the box in the form of a fluid spike, while a bubble moves up along the left box edge. Figure 4 shows the profiles of deformed interface at three cases at different time. Figure 5 shows the good convergence of Uzawa's iteration in Case I. Figure 6 shows the situations of convergence using Uzawa's method at each deformation level of interior-interface of DD computations. From this results, we can recognize the convergence speed of Case I is faster than the speed of Case II or Case III.

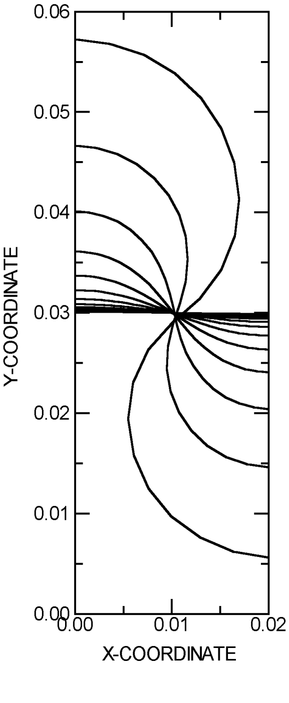


Figure 3: The schematic histories of time-dependent behaviour in a moving boundary phenomenon.

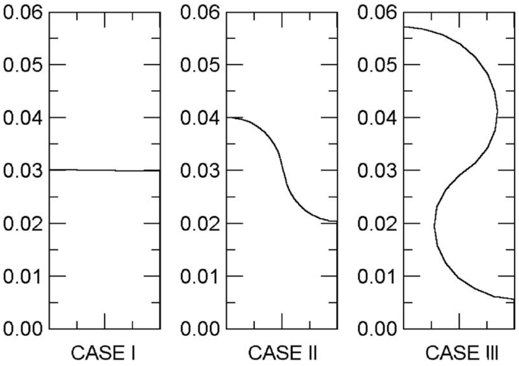


Figure 4: Three profiles showing interface deformations at different out put times. Case I :small deformation at t=0.000sec. Case II :middle deformation at t=0.130sec. Case III :large deformation at t=0.165sec.

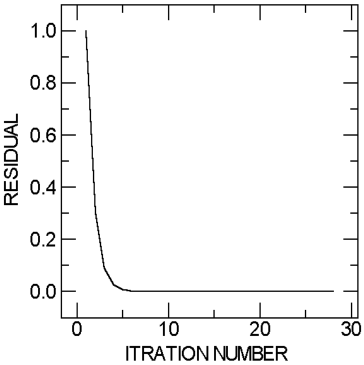


Figure 5: Convergence process of Uzawa's iterations at Case I.

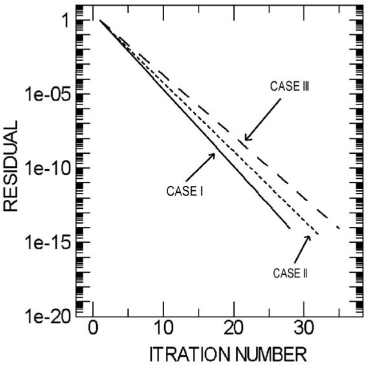


Figure 6: Comparison of convergence situations using DDM applied to three cases.

Concluding Remarks

In conclusions, we have shown the applicability for BE analysis with DDM to numerical simulation for moving boundary problems. We introduced the DDM which is based on construction of the set of BEM for each sub-domain in conjunction with compatibility and equilibrium conditions on the interface. Uzawa's method is effective to iterative computations for this type problem. This solution procedure can simulate the interfacial movement of density stratified flow. Obtained results show the tendency to increase of iteration number in computation at complicated shape of the internal boundary. Consequently, this DD-BEM procedure will contribute to the establishment of parallel BEM computation for further applications.

References

[DM96]A. J. Davies and J. Mushtaq. The domain decompostion boundary element method on a network of transputers. *Boundary Element Technology XI*, pages 397–406, 1996. Computational Mechanics Publications.

[DR81]P. Drazin and W. Reid. *Hydrodynamic Stability.* Cambridge Monographs on Mechanics and Applied Mathematics. Cambridge University Press, 1981.

[GASS93]G.Yagawa, A.Yoshioka, S.Yoshimuwa, and N. Soneda. A parallel finite element method with a supercomputer network. *Computers and Structures*, 47(3):407–418, 1993.

[GDP83]R. Glowinski, Q. V. Dinh, and J. Periaux. Domain decompostion methods for nonlinear problems in fluid dynamics. *Comp. Meth. Appl. Mech. Engng*, 40:27–109, 1983.

[KIK96]N. Kamiya, H. Iwase, and E. Kita. Parallel implementation of boundary element method with domain decomposition. *Eng. Anal. Bound. Elms.*, 18:209–216, 1996.

[ST93]R. Sugino and N. Tosaka. Numerical simulation of two-layer fluid by subregion boundary element method. *Boundary Elements Methods*, pages 109–118, 1993. Elsevier Science Publishers.

[Tho68]S. A. Thorpe. A method of producting a shear flow in a stratified fluid. *J. Fluid. Mech.*, 32(4):693–704, 1968.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau, ©2001 DDM.org

23. Two iterative substructuring methods for Maxwell's equations with discontinuous coefficients in two dimensions

A. Toselli [1]

Introduction

In this paper, we present some numerical results for a Balancing and a FETI method for the solution of a linear system arising from the edge element approximation of a vector field problem in two dimensions. The two methods are presented as projected preconditioned conjugate algorithms and give comparable performances in our tests. Our numerical results show that their condition number is independent of the number of substructures and grows only polylogarithmically with the number of unknowns associated with individual substructures. It is also independent of the jumps of both coefficients of the original problem.

We consider the following problem: Find $\mathbf{u} \in H_0(\mathrm{curl}\,;\Omega)$, such that

$$a(\mathbf{u},\mathbf{v}) = \int_\Omega \mathbf{f}\cdot\mathbf{v}\,dx, \quad \mathbf{v} \in H_0(\mathrm{curl}\,;\Omega), \tag{1}$$

where the bilinear form $a(\cdot,\cdot)$ is defined as

$$a(\mathbf{u},\mathbf{v}) := \int_\Omega \left(a\,\mathrm{curl}\,\mathbf{u}\,\mathrm{curl}\,\mathbf{v} + b\,\mathbf{u}\cdot\mathbf{v}\right)\,dx,$$

and $\mathbf{f} \in L^2(\Omega)^2$. Here, Ω is a bounded, open, connected polygon in $\mathbb{R}^2$, $H(\mathrm{curl}\,;\Omega)$ is the space of vectors in $L^2(\Omega)^2$, with curl in $L^2(\Omega)$, and $H_0(\mathrm{curl}\,;\Omega)$ its subspace of vectors with vanishing tangential component on $\partial\Omega$. The coefficients a and b are positive functions in $L^\infty(\Omega)$ bounded away from zero.

Finite element functions

For the discretization of problem (1), we consider a conforming triangulation $\mathcal{T}_h$ of Ω, of maximum diameter h, consisting of triangles or rectangles. We then define U as the space of edge elements of lowest degree, defined on $\mathcal{T}_h$, originally introduced in [N8́0]. Let $\mathcal{E}_h$ be the set of edges of $\mathcal{T}_h$. We recall that the tangential components of the vectors in U are constant along the edges of $\mathcal{T}_h$ and that these constant values can be chosen as the degrees of freedom in U.

We then consider a non–overlapping partition of the domain Ω, consisting of subdomains, also called substructures, $\mathcal{F}_H = \{\Omega_i |\, i = 1,\ldots,N\}$. The substructures are

[1] Courant Institute of Mathematical Sciences, 251 Mercer Street, New York, N.Y. 10012. E-mail: toselli@cims.nyu.edu URL: http://www.math.nyu.edu/~toselli. This work was supported in part by the Applied Mathematical Sciences Program of the U.S. Department of Energy under Contract DEFGO288ER25053.

connected polygonal domains the boundaries of which do not cut through the elements, and H is the maximum of their diameters. Let $\mathbf{t}_i$ be the unit tangent to $\partial\Omega_i$, having counterclockwise direction and restricted to $\partial\Omega_i \setminus \partial\Omega$. We will employ these unit vectors to define the coarse spaces of our algorithms. We also define the interface Γ as

$$\Gamma := \bigcup_{i=1}^{N} \partial\Omega_i \setminus \partial\Omega.$$

We remark that we only present numerical results for uniform meshes in this paper, but that our algorithms can be defined for more general cases. In particular, a theoretical bound for a FETI method, which is valid for triangulations that are shape–regular and locally quasi uniform, was proven in [TK99]. In the following, we assume, for simplicity, that the coefficient b is constant on each substructure and equal to b_i.

Given a substructure Ω_i, we define U_i as the space of restrictions of vectors in U to the Ω_i. We also define the local spaces W_i of tangential vectors as

$$W_i := \{(\mathbf{u}_i \cdot \mathbf{t}_i)\,\mathbf{t}_i \text{ restricted to } \partial\Omega_i \setminus \partial\Omega \ \mid \mathbf{u}_i \in U_i\}.$$

The vectors in W_i are uniquely determined by the degrees of freedom on $\partial\Omega_i$. In the following, the column vector of degrees of freedom of $\mathbf{u}_i \in W_i$ will be denoted by u_i, and it will be convenient to use the same notation for spaces of vectors and the corresponding spaces of degrees of freedom.

The finite element discretization of (1) gives rise to a symmetric, positive definite linear system. The degrees of freedom inside the substructures and on $\partial\Omega$ only belong to one substructure and can be eliminated in parallel by block Gaussian elimination. We are then left with a linear system involving only the degrees of freedom on Γ. Let S_i be the local Schur complement relative to the degrees of freedom on $\partial\Omega_i \setminus \partial\Omega$

$$S_i : W_i \longrightarrow W_i.$$

If a local vector on Ω_i is divided into two subvectors, of degrees of freedom corresponding to edges inside Ω_i and on $\partial\Omega_i \setminus \partial\Omega$, respectively, the local stiffness matrix of A_i can be written as

$$A_i = \begin{bmatrix} A_i^{II} & A_i^{IB} \\ A_i^{BI} & A_i^{BB} \end{bmatrix},$$

and the Schur complement S_i is defined as

$$S_i := A_i^{BB} - A_i^{BI} \left(A_i^{II}\right)^{-1} A_i^{IB}.$$

Before introducing our algorithms, we need to define a set of local scaling functions. These functions are constructed with the values of the coefficient b and ensure that the condition number of our iterative methods is independent of the jumps of *both* coefficients. For a substructure Ω_i, we define a piecewise constant function $\mu_i^\dagger$ on $\partial\Omega_i \setminus \partial\Omega$ such that

$$\mu^\dagger_{i\,|_e} \equiv \frac{b_i^\delta}{b_i^\delta + b_k^\delta}, \quad e \subset \partial\Omega_i \cap \partial\Omega_k, \quad e \in \mathcal{E}_h,$$

where $\delta \geq 1/2$ is arbitrary but fixed. Let D_i be the diagonal matrix that represents the multiplication of vectors in W_i by $\mu_i^{\dagger}$.

Conjugate Gradient algorithms

The two methods that we consider can be described as projected preconditioned conjugate gradient (PPCG) algorithms. We suppose that we are looking for the solution of a symmetric, positive definite linear system

$$Fz = d, \quad z \in V, \tag{2}$$

arising from a finite element discretization of an elliptic problem.

We first introduce a suitable subspace $V_0 \subset V$, of low dimension K, that will play the role of a *coarse space*, and define P_0 as the projection onto V_0 that is orthogonal with respect to the scalar product induced by F. The operator

$$P := I - P_0,$$

is also an orthogonal projection and, if

$$V = V_0 \oplus V^{\perp},$$

we have that $Range(P) = V^{\perp}$. Let $z_0 = P_0 z$ be the projection of the solution z onto V_0.

We consider the following preconditioned system

$$PMP^t\, Fz = PMP^t\, d, \quad z \in z_0 + V^{\perp}, \tag{3}$$

where the preconditioner M has the form

$$M := \sum_{i=1}^{N} M_i,$$

and the application of the *local component* M_i involves the solution of a local problem on the substructure Ω_i. Here, P^t denotes the transpose of the matrix P. Recalling the definition of P_0, we see that $P^t \neq P$, in general.

A full description of the conjugate gradient method applied to Equation (3) can be found in [FCM95, Tos00, TK99]. Here, we only remark that the action of the projection P on a vector can be evaluated at the expense of applying the matrix F and of solving a coarse problem of dimension K. Moreover, the action of P^t does not need to be calculated in practice.

A suitable choice of the projection P ensures that the condition number of the corresponding iterative method is independent of the number of substructures and depends only on the ratio $\rho = H/h$, which is a measure of the number of degrees of freedom in each substructure. In addition, a suitable choice of the preconditioner M ensures that the condition number is slowly increasing with ρ and is independent of possibly large jumps of the coefficients.

A Balancing method

The method that we present is a variant of the Neumann–Neumann algorithm introduced and analyzed in [Tos00]: it employs the same preconditioner M, but a different coarse space V_0. In [Tos00], the partition $\mathcal{F}_H$ is required to be a conforming coarse triangulation of Ω and V_0 is the standard edge element space defined on it, while, here, the partition is arbitrary and the basis functions of V_0 are associated to the substructures.

We consider the linear system obtained from the approximation of Problem (1) on the conforming finite element space U and define W as the space of tangential components of the vectors in U on Γ. We note that the restrictions of the vectors in W to $\partial\Omega_i \setminus \partial\Omega$ belong to W_i, for $i = 1, \ldots, N$. After eliminating the variables interior to the substructures, we are left with the system

$$Su = g, \quad u \in W, \tag{4}$$

where S is the global Schur complement matrix relative to Γ and g is the resulting right hand side. We define the operators

$$R_i^t : W_i \longrightarrow W, \quad i = 1, \ldots, N,$$

as the extensions by zero of local vectors in W_i on the whole Γ, and note that the R_i are the restriction operators from W to W_i. We can then write

$$S = \sum_{i=1}^{N} R_i^t S_i R_i.$$

Problem (4) then corresponds to the choice $F = S$, $d = g$, $V = W$, in (2). We define the coarse space as the span of the extensions to Γ of the vectors $\{\mathbf{t}_i\}$:

$$V_0 := span\{R_i^t\, t_i \mid i = 1, \ldots, N\}.$$

It can easily be checked that the dimension of V_0 is equal to the number of substructures minus one.

Following [Tos00], we define the local components of the preconditioner as

$$M_i := R_i^t\, D_i\, S_i^{-1} D_i\, R_i, \quad i = 1, \ldots, N.$$

A FETI method

The method presented in this section was originally developed and analyzed in [TK99]. We define the non–conforming space $\widehat{W}$ as

$$\widehat{W} := \prod_{i=1}^{N} W_i.$$

We note that the vectors in $\widehat{W}$ are in general discontinuous across Γ and, given two substructures, Ω_i and Ω_k, that share a common edge, there are two different fields on $\partial\Omega_i \cap \partial\Omega_k$ that correspond to a vector $\mathbf{u} \in \widehat{W}$. We define the block diagonal matrix

$$\widehat{S} := \mathrm{diag}\{S_1,\, S_2,\, \cdots, S_N\} \,:\, \widehat{W} \longrightarrow \widehat{W}.$$

We can then formulate our finite element problem as a constrained minimization problem: Find $u \in \widehat{W}$, such that

$$\left.\begin{array}{c}\frac{1}{2}u^t\,\widehat{S}\,u - u^t\,g \;\longrightarrow \min \\ Bu = 0\end{array}\right\} \tag{5}$$

where the matrix B evaluates the difference of the corresponding degrees of freedom on Γ and can be written as

$$B = \left[B^{(1)}\;B^{(2)}\;\cdots\;B^{(N)}\right].$$

Here, g is constructed with the local load vectors on the substructures. We then introduce a vector of Lagrange multipliers λ, to enforce the constraints, and obtain a saddle point formulation of (5). After eliminating the primal variable u, we obtain the following equation for the dual variable λ, see [FCM95, TK99],

$$B\widehat{S}^{-1}B^t\lambda = B\widehat{S}^{-1}g, \quad \lambda \in \mathit{Range}(B). \tag{6}$$

We consider a PPCG method for the solution of (6). This corresponds to the choice $F = B\widehat{S}^{-1}B^t$, $d = B\widehat{S}^{-1}g$, $V = \mathit{Range}(B)$, in (2). We note that V is the space of jumps of the tangential vectors in $\widehat{W}$. We then define the coarse space V_0 as a space of scaled jumps of the local vectors $\{\mathbf{t}_i\}$

$$V_0 := \mathit{span}\{B_i\,(I - D_i)\,t_i \mid i = 1, \ldots, N\}.$$

We refer to [TK99, Sect. 5] for additional details and for a discussion of the dimension of V_0. In particular, we note that the vectors $\{t_i\}$ also need to be scaled using the lengths of the edges in $\mathcal{E}_h$ if the mesh $\mathcal{T}_h$ is not uniform.

Following [TK99, KW99], we define the local components of the preconditioner as

$$M_i := (B\widehat{D}^{-1}B^t)^{-1}\,B_iD_i^{-1}\widehat{S}_iD_i^{-1}B_i^t\,(B\widehat{D}^{-1}B^t)^{-1},$$

where $\widehat{D} := \mathrm{diag}\{D_1, D_2, \cdots, D_N\}$.

Numerical results

We first remark that, for the Balancing method, at each conjugate gradient step, we need to solve one Neumann problem on each substructure for the application of the preconditioner, and two Dirichlet problems for the application of S and P (we recall that P is a projection that is orthogonal with respect to the scalar product induced by $F = S$). Similarly, for the FETI method, at each step, we need to solve two Neumann problems and one Dirichlet problem on each substructure. We refer to [FCM95, Tos00, TK99] for additional comments.

In our numerical tests, we consider the domain $\Omega = (0,1)^2$ and two uniform triangulations $\mathcal{T}_h$ and $\mathcal{F}_H$. The fine triangulation is made of triangles for the FETI method, and squares for the Balancing method. It consists of $2 * n^2$ triangles and n^2 squares, respectively, with $h = 1/n$. We note that, as opposed to the case of nodal

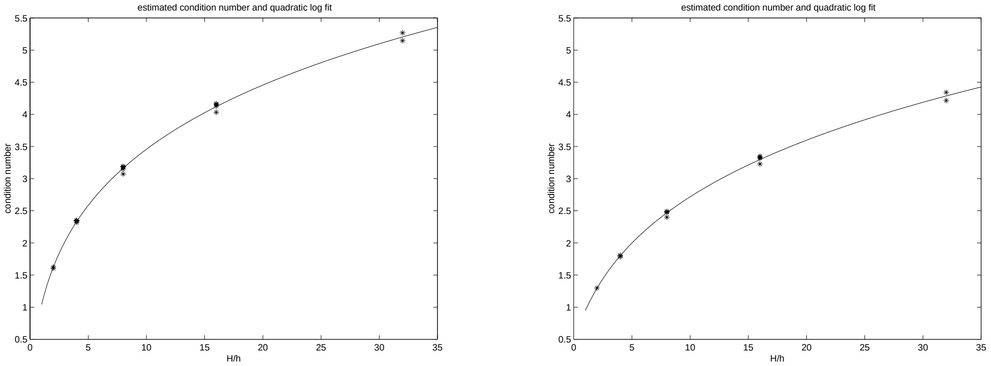


Figure 1: Case with $a = 1$, $b = 1$, $n = 32, 64, 128, 192, 256$. Estimated condition number (asterisk) and least–square second order logarithmic polynomial (solid line), versus $\rho = H/h$ for the Balancing (on the left) and the FETI (on the right) methods.

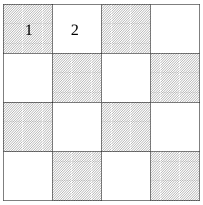


Figure 2: Checkerboard distribution of the coefficients in the unit square.

elements, for a fixed value of n, triangular and rectangular meshes do not give rise to edge element spaces of the same dimension. Nevertheless, the mesh size h and the order of accuracy is the same, see [N8́0], and our comparisons of the two methods are still reasonably fair. The coarse triangulation consists of nc^2 squares which are unions of fine elements, with $H = 1/nc$. The substructures Ω_i are the elements of the coarse triangulation $\mathcal{F}_H$. Throughout, we use the value $\delta = 1/2$.

We first consider a case with constant coefficients and meshes with $n = 32, 64, 128, 192, 256$. Figure 1 shows the estimated condition number (asterisks), for $a = b = 1$, as a function of $\rho = H/h$, for different values of n. The results for the FETI method are taken from [TK99]. For a fixed value of ρ, the condition number is quite insensitive to the dimension of the fine mesh. We have also plotted the best second order logarithmic polynomial least–square fits; our numerical results for both methods are consistent with the bound for the condition number

$$\kappa(PMP^tF) \leq C \left(1 + \log \frac{H}{h}\right)^2 ,$$

and suggest that this bound is sharp. We note that this bound was proved in [TK99] for the FETI method.

We then consider some cases where the coefficients have jumps. In Table 1, we show some results when the coefficient b has jumps across the substructures. We

b_2, ρ	4	8	16
10^{-4}	15.6 (22)	13.4 (22)	12.1 (22)
10^{-3}	15.1 (21)	13.2 (21)	12.1 (23)
10^{-2}	13.8 (20)	12.5 (21)	11.9 (23)
10^{-1}	10.8 (19)	10.8 (21)	11.5 (22)
1	6.31 (17)	7.55 (19)	10.2 (21)
10	3.87 (13)	5.41 (15)	7.36 (18)
10^2	2.33 (8)	3.12 (10)	3.87 (11)
10^3	3.70 (12)	4.77 (14)	5.56 (16)
10^4	3.96 (14)	4.33 (14)	4.64 (15)
10^5	3.27 (12)	3.55 (13)	4.34 (14)
10^6	2.99 (12)	3.44 (13)	4.28 (14)

4	8	16
4.12 (17)	5.99 (22)	8.42 (26)
4.09 (16)	5.96 (20)	8.37 (25)
4.04 (15)	5.88 (19)	8.25 (23)
3.88 (13)	5.65 (17)	7.91 (21)
3.44 (12)	5.02 (15)	6.99 (18)
2.56 (9)	3.73 (12)	5.16 (14)
1.76 (7)	2.41 (8)	3.10 (10)
2.51 (9)	3.37 (11)	3.99 (12)
2.74 (10)	3.09 (11)	3.51 (11)
2.20 (9)	2.73 (10)	3.35 (11)
2.09 (9)	2.65 (10)	3.34 (12)

Table 1: Checkerboard distribution for b: (b_1, b_2). Estimated condition number and number of CG iterations to obtain a relative preconditioned residual less than 10^{-6} (in parentheses), versus $\rho = H/h$ (columns) and b_2 (rows), for the Balancing (on the left) and the FETI (on the right) methods. Case of $n = 128$, $a = 1$, and $b_1 = 100$.

consider the checkerboard distribution shown in Figure 2, where b is equal to b_1 in the shaded area and to b_2 elsewhere. For a fixed value of $n = 128$, $b_1 = 100$, and $a = 1$, the estimated condition number and the number of iterations in order to obtain a reduction of the norm of the preconditioned residual by a factor 10^{-6}, are shown as a function of $\rho = H/h$ and b_2. For $b_2 = 100$, the coefficient b has a uniform distribution, and this corresponds to a minimum for the condition number and the number of iterations for both methods. When b_2 decreases or increases, the condition number and the number of iterations also increase, but they can still be bounded independently of b_2. We observe that the two methods give comparable iteration counts.

In Table 2, we show some results when the coefficient a has jumps. We consider the checkerboard distribution shown in Figure 2, where a is equal to a_1 in the shaded area and to a_2 elsewhere. For a fixed value of $n = 128$, $a_1 = 0.01$, and $b = 1$, the estimated condition number and the number of iterations are shown as a function of $\rho = H/h$ and a_2. We remark that for $a_2 = 0.01$, the coefficient a has a uniform distribution. For both methods, a slight increase in the number of iterations and the condition number is observed, when a_2 is decreased or increased and when H/h is large.

References

[FCM95]Charbel Farhat, Po-Shu Chen, and Jan Mandel. A scalable Lagrange multiplier based domain decomposition method for time-dependent problems. *Int. J. Numer. Meth. Eng.*, 38:3831–3853, 1995.

[KW99]Axel Klawonn and Olof B. Widlund. FETI and Neumann–Neumann iterative substructuring methods: Connections and new results. Technical Report 796, Department of Computer Science, Courant Institute, December 1999. To appear in Comm. Pure Appl. Math.

a_2, ρ	4	8	16	4	8	16
10^{-7}	2.56 (10)	4.33 (13)	8.02 (17)	2.80 (8)	4.49 (12)	7.29 (15)
10^{-6}	2.56 (10)	4.32 (13)	8.01 (17)	2.41 (8)	3.81 (11)	6.21 (14)
10^{-5}	2.56 (10)	4.30 (13)	7.96 (17)	1.82 (7)	2.65 (9)	4.05 (11)
10^{-4}	2.52 (9)	4.13 (13)	7.49 (16)	1.79 (7)	2.45 (8)	3.23 (10)
10^{-3}	2.39 (9)	3.51 (12)	5.59 (14)	1.78 (7)	2.42 (8)	3.07 (9)
10^{-2}	2.34 (9)	3.16 (11)	4.14 (13)	1.76 (7)	2.40 (8)	3.25 (10)
10^{-1}	2.32 (8)	3.12 (10)	3.87 (12)	1.77 (7)	2.41 (8)	3.10 (10)
1	2.33 (8)	3.12 (10)	3.87 (11)	1.77 (7)	2.46 (8)	3.26 (10)
10	2.34 (8)	3.16 (10)	4.11 (11)	1.77 (7)	2.46 (8)	3.26 (10)
10^2	2.34 (8)	3.16 (10)	4.14 (12)	1.77 (7)	2.46 (8)	3.26 (10)
10^3	2.34 (8)	3.17 (10)	4.14 (12)	1.77 (7)	2.46 (8)	3.26 (10)

Table 2: Checkerboard distribution for a: (a_1, a_2). Estimated condition number and number of CG iterations to obtain a relative preconditioned residual less than 10^{-6} (in parentheses), versus $\rho = H/h$ (columns) and a_2 (rows), for the Balancing (on the left) and the FETI (on the right) methods. Case of $n = 128$, $b = 1$, and $a_1 = 0.01$.

[N8́0]J.-C. Nédélec. Mixed finite elements in R^3. *Numer. Math.*, 35:315–341, 1980.

[TK99]Andrea Toselli and Axel Klawonn. A FETI domain decomposition method for Maxwell's equations with discontinuous coefficients in two dimensions. Technical Report 788, Department of Computer Science, Courant Institute, 1999. Submitted to SIAM J. Numer. Anal.

[Tos00]Andrea Toselli. Neumann–Neumann methods for vector field problems. *ETNA*, 11:1–24, 2000.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau, ©2001 DDM.org

24. FEM-FSM Combined Method for 2D Exterior Laplace and Helmholtz Problems

T. Ushijima[1]

Introduction

Consider the Poisson equation $-\Delta u = f$ in a planar exterior domain of a bounded domain $\mathcal{O}$. Assume that $f = 0$ in the outside of a disc with sufficiently large diameter. The solution u is assumed to be bounded at infinity. Discretizing the problem, we employ the finite element method (FEM, in short) inside the disc, and the charge simulation method (CSM, in short) outside the disc. A result of mathematical analysis for this FEM-CSM combined method is reported in this paper.

CSM is a typical example of the fundamental solution method (FSM, in short), through which the solution of homogeneous partial differential equation is approximated as a linear combination of fundamental solutions of differential operator. Hence the combined method for 2D exterior Laplace problem is extendable to the planar exterior reduced wave equations. Our discretization procedure for the reduced wave equation is also described in the paper.

Boundary bilinear forms of Steklov type for exterior Laplace problems and its CSM-approximation form

Let D_a be the interior of the disc with radius a having the origin as its center, and let Γ_a be the boundary of D_a. Let $\Omega_e = (D_a \cup \Gamma_a)^C$, which is said to be the exterior domain. We use the notation $\mathbf{r} = \mathbf{r}(\theta)$ for the point in the plane corresponding to the complex number $re^{i\theta}$ with $r = |\mathbf{r}|$ where $|\mathbf{r}|$ is the Euclidean norm of $\mathbf{r} \in \boldsymbol{R}^2$. Similarly we use $\mathbf{a} = \mathbf{a}(\theta)$, and $\vec{\rho} = \vec{\rho}(\theta)$, corresponding to $ae^{i\theta}$ with $a = |\mathbf{a}|$, and $\rho e^{i\theta}$ with $\rho = |\vec{\rho}|$, respectively.

Fix a positive integer N. Set

$$\theta_1 = \frac{2\pi}{N}, \qquad \theta_j = j\theta_1 \quad \text{for} \quad j \in \boldsymbol{Z}.$$

Fix a positive number ρ so as to satisfy $0 < \rho < a$. Let

$$\vec{\rho}_j = \vec{\rho}(\theta_j), \quad \mathbf{a}_j = \mathbf{a}(\theta_j), \quad 0 \le j \le N-1.$$

The points $\vec{\rho}_j$, and $\mathbf{a}_j$, are said to be the source, and the collocation, points, respectively. The arrangement of the set of source points and collocation points introduced as above is called **the equi-distant equally phased arrangement of source points and collocation points**, in this paper.

[1] The University of Electro-Communications, ushijima@im.uec.ac.jp

For functions $u(\mathbf{a}(\theta))$ and $v(\mathbf{a}(\theta))$ of $H^{1/2}(\Gamma_a)$, let us introduce the boundary bilinear form of Steklov type for exterior Laplace problem through the following formula:

$$b(u,v) = 2\pi \sum_{n=-\infty}^{\infty} |n| f_n \overline{g_n},$$

where f_n, and g_n, are continuous Fourier coefficients of $u(\mathbf{a}(\theta))$, and $v(\mathbf{a}(\theta))$, respectively. Namely f_n is defined through the following formula:

$$f_n = \frac{1}{2\pi} \int_{-\pi}^{\pi} u(\mathbf{a}(\theta)) e^{-in\theta} d\theta.$$

It is to be noted the following fact:

If $u(\mathbf{a}(\theta))$ is the boundary value on Γ_a of the function $u(\mathbf{r})$ satisfying the following boundary value problem (E) *of* (1) *with $\varphi = u(\mathbf{a}(\theta))$:*

$$\text{(E)} \quad \left\{ \begin{array}{rcll} -\Delta u & = & 0 & \text{in} \quad \Omega_e, \\ u & = & \varphi & \text{on} \quad \Gamma_a, \\ \sup\limits_{\Omega_e} & |u| < \infty, & & \end{array} \right. \tag{1}$$

then

$$b(u,v) = -\int_{\Gamma_a} \frac{\partial u}{\partial r} v d\Gamma. \tag{2}$$

(In (2), $d\Gamma$ is the curve element of Γ_a. Namely $d\Gamma = a d\theta$ in the polar coordinate expression.)

A CSM approximate form for $b(u,v)$, which is denoted by $\overline{b}^{(N)}(u,v)$, is represented through the following formula (3):

$$\overline{b}^{(N)}(u,v) = -\frac{2\pi}{N} \sum_{j=0}^{N-1} \frac{\partial u^{(N)}(\mathbf{a}_j)}{\partial r} v^{(N)}(\mathbf{a}_j), \tag{3}$$

where $u^{(N)}(\mathbf{r})$, and $v^{(N)}(\mathbf{r})$, are CSM-approximate solutions for $u(\mathbf{r})$ satisfying (E) of (1) with $\varphi = u(\mathbf{a}(\theta))$, and $\varphi = v(\mathbf{a}(\theta))$, respectively. Namely $u^{(N)}(\mathbf{r})$ is determined through the following problem $(\mathrm{E}^{(\mathrm{N})})$ of (4) with $f(\mathbf{a}(\theta)) = u(\mathbf{a}(\theta))$:

$$(\mathrm{E}^{(\mathrm{N})}) \quad \left\{ \begin{array}{rcl} u^{(N)}(\mathbf{r}) & = & \sum\limits_{j=0}^{N-1} q_j G_j(\mathbf{r}) + q_N, \\ u^{(N)}(\mathbf{a}_j) & = & f(\mathbf{a}_j), \quad 0 \le j \le N-1, \\ \sum\limits_{j=0}^{N-1} q_j & = & 0, \end{array} \right. \tag{4}$$

where

$$G_j(\mathbf{r}) = E(\mathbf{r} - \vec{\rho}_j) - E(\mathbf{r}), \qquad E(\mathbf{r}) = -\frac{1}{2\pi} \log r.$$

Problem ($\mathrm{E}^{(\mathrm{N})}$) of (4) is to find $N+1$ unknowns $q_j,\ 0 \leq j \leq N$, and it is uniquely solvable for any fixed $\rho \in (0, a)$. See [KO88], [Ush98a], and [Ush98b].

Let us use the parameter γ as

$$\gamma = \frac{\rho}{a},$$

and let

$$N(\gamma) = \frac{\log 2}{-\log \gamma}\ .$$

Modifying the treatment in [KO88] appropriately, we have the following Theorem:

Theorem 1 *Fix a positive number b, $0 < b < a$. Let $u(\mathbf{r})$ be harmonic in a domain containing the exterior domain of the disc with radius b having the origin as its center. And let $u^{(N)}(\mathbf{r})$ be the solution of the problem* ($\mathrm{E}^{(\mathrm{N})}$) *of* (4) *with the data $f(\mathbf{a}(\theta)) = u(\mathbf{a}(\theta))$. Let $N \geq N(\gamma)$. Then there exist constants $B > 0$ and $\beta \in (0, 1)$, independent of u (with the property above) and N, such that the following two estimates are valid:*

$$\max_{\mathbf{r}\in\overline{\Omega}_e} \left| u(\mathbf{r}) - u^{(N)}(\mathbf{r}) \right| \leq B \cdot \beta^N \cdot \max_{|\mathbf{r}|=b} |u(\mathbf{r})|\,,$$

$$\max_{\mathbf{r}\in\overline{\Omega}_e} \left| \text{grad } u(\mathbf{r}) - \text{grad } u^{(N)}(\mathbf{r}) \right|_{\mathbf{R}^2} \leq B \cdot \beta^N \cdot \max_{|\mathbf{r}|=b} |u(\mathbf{r})|\,.$$

FEM-CSM combined method for exterior Laplace problems

Fix a simply connected bounded domain $\mathcal{O}$ in the plane. Assume that the boundary $\mathcal{C}$ of $\mathcal{O}$ is sufficiently smooth. The exterior domain of $\mathcal{C}$ is denoted by Ω. Fix a function $f \in L^2(\Omega)$ with the property that the support of f, $\mathrm{supp}(f)$, is bounded. Choose a so large that the open disc D_a may contain the union $\mathcal{O} \cup \mathrm{supp}(f)$ in its interior. The following Poisson equation (E) of (5) is employed as a model problem.

$$\text{(E)} \quad \left\{ \begin{array}{rcll} -\Delta u & = & f & \text{in} \quad \Omega, \\ u & = & 0 & \text{on} \quad \mathcal{C}, \\ \sup\limits_{|\mathbf{r}|>a} |u| & < & \infty. & \end{array} \right. \tag{5}$$

The intersection of the domain Ω and the disc D_a is said to be the interior domain, denoted by Ω_i: $\Omega_i = \Omega \cap D_a$. Consider the Dirichlet inner product $a(u, v)$ for $u, v \in H^1(\Omega_i)$:

$$a(u, v) = \int_{\Omega_i} \text{grad } u \text{ grad } v \ d\Omega.$$

Since the trace $\gamma_a v$ on Γ_a is an element of $H^{1/2}(\Gamma_a)$ for any $v \in H^1(\Omega_i)$, the boundary bilinear form of Steklov type $b(u, v)$ is well defined for $u, v \in H^1(\Omega_i)$. Therefore we can define a continuous symmetric bilinear form:

$$t(u, v) = a(u, v) + b(u, v)$$

for $u, v \in H^1(\Omega_i)$. Let $F(v)$ be a continuous linear functional on $H^1(\Omega_i)$ defined through the following formula:

$$F(v) = \int_{\Omega_i} fv \; d\Omega.$$

A function space V is defined as follows:

$$V = \left\{ v \in H^1(\Omega_i) : v = 0 \quad \text{on} \quad \mathcal{C} \right\}.$$

Using these notations, the following weak formulation problem (Π) of (6) is defined.

$$(\Pi) \quad \left\{ \begin{array}{l} t(u,v) = F(v), \quad v \in V, \\ u \in V. \end{array} \right. \tag{6}$$

Admitting the equivalence between the equation (E) of (5) and the problem (Π) of (6), we consider the problem (Π) of (6) and its approximate ones hereafter.

Fix a positive number ρ so as to satisfy $0 < \rho < a$. For a fixed positive integer N, set the points $\vec{\rho}_j, \mathbf{a}_j, 0 \le j \le N-1$, as the equi-distant equally phased arrangement of source points and collocation points.

A family of finite dimensional subspaces of V :

$$\{V_N : N = N_0, N_0 + 1, \dots\}$$

is supposed to have the following properties:

$$(\mathrm{V_N} - 1) \quad V_N \subset C(\overline{\Omega_i}).$$

$$(\mathrm{V_N} - 2) \quad \left\{ \begin{array}{l} \textit{For any } v \in V_N, \; v(\mathbf{a}(\theta)) \textit{ is an equi-distant piecewise linear} \\ \textit{continuous } 2\pi\textit{-periodic function with respect to } \theta. \end{array} \right.$$

$$(\mathrm{V_N} - 3) \quad \min_{v \in V_N} a(v - v_N) \le \frac{C}{N} ||v||_{H^2(\Omega_i)}, \qquad v \in V \cap H^2(\Omega_i).$$

In the property $(\mathrm{V_N} - 3)$, C is a constant independent of N and v, and

$$a(v) = a(v,v)^{1/2}, \quad v \in V.$$

To construct a family $\{V_N\}$ with the conditions $(\mathrm{V_N} - 1)$, $(\mathrm{V_N} - 2)$ and $(\mathrm{V_N} - 3)$, we employ the curved element technique due to [Zlá73] .

For $u, v \in H^1(\Omega_i) \cap C(\overline{\Omega_i})$, we define a bilinear form $\overline{t}^{(\mathrm{N})}(u,v)$ as follows.

$$\overline{t}^{(N)}(u,v) = a(u,v) + \overline{b}^{(N)}(u,v).$$

A family of approximate problems $(\overline{\Pi}^{(\mathrm{N})})$ of (7) is stated as follows.

$$(\overline{\Pi}^{(\mathrm{N})}) \quad \left\{ \begin{array}{l} \overline{t}^{(\mathrm{N})}(\overline{u}_N, v) = F(v), \quad v \in V_N, \\ \overline{u}_N \in V_N. \end{array} \right. \tag{7}$$

We can show the following error estimate:

Theorem 2 *Suppose that* supp(f) *is contained in a disc D_b with the radius $b(< a)$ having the origin as its center. Let the function $D(\xi)$ of $\xi \in (0,1)$ be defined through*

$$D(\xi) = \frac{\xi}{(1-\xi)^3}.$$

Let $N \geq N(\gamma)$. Then there is a constant C such that

$$||u - \overline{u}_N||_{H^1(\Omega_i)} \leq C \left\{ B\beta^N + \frac{1 + D(\frac{b}{a})}{N} \right\} ||f||_{L^2(\Omega_i)},$$

where the constants B and $\beta \in (0,1)$ are described in Theorem 1 for the set of parameters $\{a, \rho, b\}$. In the above, the constant C is independent of the inhomogeneous data f and N.

Reduced wave problem in the outside of an open disc

Let k be the length of the wave number vector. Consider the following reduced wave problem $(\mathrm{E_f})$ of (8) in the exterior domain Ω_e of the circle Γ_a with radius a having the origin as its center.

$$(\mathrm{E_f}) \quad \left\{ \begin{array}{rcl} -\Delta u - k^2 u &=& 0 \ \text{ in } \Omega_e, \\ u &=& f \ \text{ on } \Gamma_a, \\ \lim\limits_{r\to\infty} \sqrt{r}\left\{\frac{\partial u}{\partial r} - iku\right\} &=& 0. \end{array} \right. \tag{8}$$

In the above, f is a complex valued continuous function on Γ_a.

The solution $u = u(\mathbf{r})$ of the problem $(\mathrm{E_f})$ of (8) is represented as

$$u = \sum_{n=-\infty}^{\infty} f_n \frac{H_n^{(1)}(kr)}{H_n^{(1)}(ka)} e^{in\theta},$$

where f_n is the continuous Fourier coefficient of the function $f(\mathbf{a}(\theta))$, and $H_n^{(1)}(z)$ is the n-th Hankel function of the first kind.

The boundary bilinear form $b(u,v)$ of Steklov type corresponding to the problem $(\mathrm{E_f})$ of (8) is given by the following formula:

$$b(u,v) = 2\pi \sum_{n=-\infty}^{\infty} \mu_{|n|} f_n \overline{g}_n,$$

where

$$\mu_n = k\frac{\dot{H}_n^{(1)}(ka)}{H_n^{(1)}(ka)} \quad \text{with} \quad \dot{H}_n^{(1)}(z) = \frac{d}{dz}H_n^{(1)}(z) \quad \text{for} \quad n = 0, 1, 2, \ldots .$$

FSM approximate problem for the reduced wave problem in the outside of an open disc

Fix a positive number ρ so as to satisfy $0 < \rho < a$. For a fixed positive integer N, set the points $\vec{\rho}_j, \mathbf{a}_j, 0 \le j \le N-1$, as the equi-distant equally phased arrangement of source points and collocation points.

The FSM approximate problem $(\mathrm{E}_\mathrm{f}^{(\mathrm{N})})$ of (9) for the problem (E_f) of (8) in the case of equi-distant equally phased arrangement of source points and collocation points is defined through the following:

$$(\mathrm{E}_\mathrm{f}^{(\mathrm{N})}) \quad \left\{ \begin{array}{rcl} u^{(N)}(\mathbf{r}) & = & \sum_{j=0}^{N-1} q_j G_j(\mathbf{r}), \\ \\ u^{(N)}(\mathbf{a}_j) & = & f(\mathbf{a}_j), \ 0 \le j \le N-1. \end{array} \right. \tag{9}$$

We use basis functions $G_j(\mathbf{r})$ in this problem represented as follows, with the use of the constant multiple of the fundamental solution of Helmholtz equation, $H_0^{(1)}(kr)$,

$$G_j(\mathbf{r}) = H_0^{(1)}(k|re^{i\theta} - \rho e^{i\theta_j}|), \quad 0 \le j \le N-1.$$

FSM approximate form for the boundary bilinear form of Steklov type

Setting

$$g(\theta) = H_0^{(1)}(k|ae^{i\theta} - \rho|),$$

we define for $l \in \boldsymbol{Z}$,

$$g_l = g(\theta_l).$$

The two-sided infinite sequence $\{g_l : l = 0, \pm 1, \pm 2, \dots\}$ has the period N. Further it is symmetric with respect to $N/2$. A **wave propagation matrix** G is defined through

$$G = (g_{jk})_{0 \le j,k \le N-1}, \qquad g_{jk} = g_{k-j}, \quad 0 \le j,k \le N-1.$$

It is to be noted that the matrix G is a complex valued symmetric cyclic square matrix of order N. The problem $(\mathrm{E}_\mathrm{f}^{(\mathrm{N})})$ of (9) is represented as

$$(\mathbf{E}) \quad G\mathbf{q} = \mathbf{f}, \qquad \text{with} \quad \mathbf{q} = (q_j)_{0 \le j \le N-1}, \quad \mathbf{f} = (f(\mathbf{a}_j))_{0 \le j \le N-1}.$$

Denote eigenvalues of the matrix G by $\lambda_j,\ 0 \le j \le N-1$. Then we have the following representation:

$$\lambda_j = \sum_{l=0}^{N-1} g_l \omega^{jl}, \quad 0 \le j \le N-1, \quad \text{with} \quad \omega = e^{i\theta_1}.$$

All the eigenvalues of G differ from zero if and only if the matrix G is regular. Therefore the problem $(\mathrm{E}_\mathrm{f}^{(\mathrm{N})})$ of (9) is uniquely solvable if and only if the following condition holds good:

$$\lambda_j \neq 0, \quad 0 \le j \le N-1.$$

Assuming the above condition, define an FSM approximate boundary bilinear form $\overline{b}^{(N)}(u,v)$ of the boundary bilinear form $b(u,v)$ through the same formula (3) as in the case of exterior Laplace problem, in which $u^{(N)}(\mathbf{r})$, and $v^{(N)}(\mathbf{r})$, are solutions of the FSM approximate problem $(\mathrm{E}_{\mathrm{f}}^{(\mathrm{N})})$ of (9) with the boundary data $f = u(\mathbf{a}(\theta))$, and $f = v(\mathbf{a}(\theta))$, respectively.

FEM-FSM combined method for the reduced wave problem in the exterior of a general scattering body

Fix a simply connected bounded domain $\mathcal{O}$ in the plane. Assume that the boundary $\mathcal{C}$ of $\mathcal{O}$ is sufficiently smooth. The exterior domain of $\mathcal{C}$ is denoted by Ω. Let g be a function representing the plane wave with the wave number vector (l,m). More precisely, set

$$g(x,y) = e^{i(lx+my)}, \qquad l^2+m^2=k^2.$$

Consider the following reduced wave problem (E) of (10).

$$(\mathrm{E}) \quad \left\{ \begin{array}{rcll} -\Delta u - k^2 u &=& 0 & \text{in} \quad \Omega, \\ u+g &=& 0 & \text{on} \quad \mathcal{C}, \\ \lim\limits_{r\to\infty} \sqrt{r}\left\{\frac{\partial u}{\partial r} - iku\right\} &=& 0. & \end{array} \right. \tag{10}$$

As in the case of Poisson equation in the second section, the intersection of the domain Ω and the disc D_a is said to be the interior domain, denoted by Ω_i.

For complex valued functions $u, v \in H^1(\Omega_i)$, consider the Dirichret inner product $a(u,v)$:

$$a(u,v) = \int_{\Omega_i} \text{grad } u \text{ grad } \overline{v} \, d\Omega,$$

where $\overline{v}$ represents the complex conjugate of v. Further the L^2 inner product for $u, v \in L^2(\Omega_i)$ is denoted by $m(u,v)$:

$$m(u,v) = \int_{\Omega_i} u\overline{v} \, d\Omega.$$

Since the trace $\gamma_a v$ on Γ_a is an element of $H^{1/2}(\Gamma_a)$ for any $v \in H^1(\Omega_i)$, we can see the boundary bilinear form of Steklov type $b(u,v)$ is well defined for $u, v \in H^1(\Omega_i)$ (See, for example, [Zeb92].). Therefore we can define a continuous bilinear form:

$$t(u,v) = a(u,v) - k^2 m(u,v) + b(u,v)$$

for $u, v \in H^1(\Omega_i)$. Hereafter, denoting the function space $H^1(\Omega_i)$ by W, let

$$V = \{v \in W : v = 0 \quad \text{on} \quad \mathcal{C}\}.$$

With these notations, the following weak formulation problem (Π) of (11) is defined.

$$(\Pi) \quad \left\{ \begin{array}{ll} t(u,v) = 0, & v \in V, \\ u+g=0 & \text{on} \quad \mathcal{C}, \\ u \in W. & \end{array} \right. \tag{11}$$

Admitting the equivalence between the equation (E) of (10) and the problem (Π) of (11), we consider the problem (Π) of (11) and its approximate ones hereafter.

A family of finite dimensional subspaces of W,

$$\{W_N : N = N_0, N_0 + 1, \dots\},$$

is supposed to have the following properties:

$$(\mathrm{W_N} - 1) \quad W_N \subset C(\overline{\Omega_i}).$$

$$(\mathrm{W_N} - 2) \quad \begin{cases} \textit{For any } v \in W_N,\ v(\mathbf{a}(\theta)) \textit{ is an equi-distant piecewise linear} \\ \textit{continuous } 2\pi\textit{-periodic function with respect to } \theta. \end{cases}$$

Define an approximate space V_N of V through

$$V_N = W_N \cap V.$$

For $u, v \in H^1(\Omega_i) \cap C(\overline{\Omega_i})$, set

$$\bar{t}^{(N)}(u, v) = a(u, v) - k^2 m(u, v) + \bar{b}^{(N)}(u, v).$$

Fix an element g_N of W_N which coincides with g at the nodal points on the interior boundary $\mathcal{C}$.

Now we can set the following approximate problem $(\overline{\Pi}^{(\mathrm{N})})$ of (12).

$$(\overline{\Pi}^{(\mathrm{N})}) \quad \begin{cases} \bar{t}^{(N)}(\bar{u}_N, v) = 0, & v \in V_N, \\ \bar{u}_N + g_N = 0 & \text{on} \quad \mathcal{C}, \\ \bar{u}_N \in W_N. \end{cases} \tag{12}$$

Thus we have formulated an FEM-FSM combined method for the reduced wave problem in the exterior of a general scattering body.

References

[KO88]M. Katsurada and H. Okamoto. A mathematical study of the charge simulation method I. *J. Fac. Sci. Univ. Tokyo, Sect. IA, Math.*, 35:507–518, 1988.

[Ush98a]T. Ushijima. An FEM-CSM combined method for 2D exterior Laplace problems, in Japanese. Abstract of Applied Mathematics Branch in Fall Joint Meeting of Mathematical Society of Japan, pp.126-129, October 1998.

[Ush98b]T. Ushijima. Some remarks on CSM approximate solutions of bounded harmonic function in a domain exterior to a circle, in japanese. Abstract of 1998 Annual Meeting of Japan Society for Industrial and Applied Mathematics, pp. 60–61, September 1998.

[Zeb92]A. Zebic. Equation de Helmholtz: Étude numérique de quelques préconditionnements pour la methode GMRES. Technical Report 1802, Raport de Recherche de INRIA, Décembre 1992.

[Zlá73]M. Zlámal. Curved elements in the finite element method. I. *SIAM J. Numer. Anal.*, 10:229–240, 1973.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau,

25. A Parallel Interface Preconditioner for the Mortar Element Method in Case of Jumping Coefficients

Yu. Vassilevski [1]

Introduction

The paper is devoted to designing an interface preconditioner for the mortar element method. After brief overview of the problem in Introduction, we discuss the mortar element method with different types of the Lagrange multiplier spaces. Next, we consider the domain decomposition technique for the solution of mortar element systems and outline the general framework of the solution of saddle-point systems which result from the mortar element system. In the last two sections, we constuct the interface preconditioner for the saddle-point Schur complement, which is the goal of the paper, and we present numerical experiments illustrating the basic properties of the interface preconditioner.

Designing the interface preconditioner is one of the most difficult problems in the mortar element method. In this paper we continue development of the Dirichlet-Dirichlet preconditioner [DA99, KV99]. We extend the method to the case of arbitrary type of Lagrange multiplier space and large jumps of coefficients. The proposed algorithm possesses natural parallelism. It is illustrated on a set of numerical experiments.

The mortar element method with Lagrange multipliers

We consider a macro-hybrid P_1 finite element method with respect to a decomposition of the computational domain $\Omega \subset \mathbb{R}^3$ into m nonoverlapping regular shaped polyhedral subdomains Ω_i, $1 \leq i \leq m$, i.e., $\bar{\Omega} = \cup_{i=1}^m \bar{\Omega}_i,\ \Omega_i \cap \Omega_j = \emptyset, 1 \leq i \neq j \leq m$. We assume this decomposition to be geometrically conforming in the sense that if $\bar{\Theta}_{ij} = \bar{\Omega}_i \cap \bar{\Omega}_j \neq \emptyset, i \neq j$, then $\bar{\Theta}_{ij}$ is either a common vertex, a common edge, or a common face of Ω_i and Ω_j. We refer to $\mathcal{S} := \bigcup \{\bar{\Theta}_{ij} : \mid \Theta_{ij} \mid \neq 0, 1 \leq i \neq j \leq m\}$ as the skeleton of the decomposition. We further decompose the skeleton, according to

$$\mathcal{S} = \bigcup_{k=1}^{K} \bar{\gamma}_k = \bigcup_{k=1}^{K} \bar{\delta}_k, \tag{1}$$

into the so-called mortars γ_k and non-mortars δ_k, $1 \leq k \leq K$, where each mortar is the entire open face of two adjacent subdomains $\Omega_{M(k)}$ and $\Omega_{\bar{M}(k)}, 1 \leq M(k) \neq \bar{M}(k) \leq m$, i.e., $\gamma_k = \Theta_{M(k),\bar{M}(k)}$. The non-mortars δ_k denote the corresponding opposite side

[1] Institute of Numerical Mathematics, Gubkina 8, 117333, Moscow, Russia, vasilevs@dodo.inm.ras.ru. This work was supported in part by the Russian Foundation for Basic Research Grant 99-01-01189, by the French-Russian A.M.Lyapounov Institute of Informatics and Applied Mathematics, and CNRS (France)

of the mortars γ_k. Choosing $H^{1/2}(\delta_k)$ as the trace space of $H^1(\Omega_{\bar{M}(k)})$ on δ_k, we introduce

$$V := \prod_{i=1}^{m} H^1(\Omega_i), \quad \Lambda := \prod_{k=1}^{K} H^{-1/2}(\delta_k).$$

We consider an elliptic problem in the macro-hybrid primal variational formulation [BF91]: Find $(u, \lambda) \in V \times \Lambda$ such that

$$\begin{aligned} a(u,v) + b(\lambda, v) &= l(v), \quad v \in V, \\ b(\mu, u) &= 0, \quad \mu \in \Lambda. \end{aligned} \tag{2}$$

Here, the bilinear forms $a(\cdot,\cdot) : V \times V \to \mathbb{R}, b(\cdot,\cdot) : \Lambda \times V \to \mathbb{R}$ and the functional $l(\cdot) : V \to \mathbb{R}$ are given by

$$a(v,w) := \sum_{i=1}^{m} a_i(v,w), \quad a_i(v,w) := \int_{\Omega_i} [\rho \nabla v \cdot \nabla w + \varepsilon v w] dx,$$

$$b(\mu, v) := \sum_{k=1}^{K} b_k(\mu, v), \quad b_k(\mu, v) :=< \mu, [v]_J >_{\delta_k}, \quad l(v) := \sum_{i=1}^{m} \int_{\Omega_i} f v dx,$$

where $[v]_J|_{\delta_k} := v|_{\Omega_{\bar{M}(k)}} - v|_{\Omega_{M(k)}}$, and $< \cdot, \cdot >_{\delta_k}$ refers to the dual pairing between $H^{-1/2}(\delta_k)$ and $H^{1/2}(\delta_k), f \in L_2(\Omega)$. For simplicity we assume that $\varepsilon(x) = \varepsilon_i \equiv const_i > 0$, $\rho(x) = \rho_i \equiv const_i > 0$ in Ω_i, $i = 1, \ldots, m$.

Let Ω_i^h be a conformal simplicial triangulation of Ω_i, $i = 1, \ldots, m$. We denote by V_i^h the space of P_1 conforming finite elements on Ω_i associated with triangulation Ω_i^h. It is obvious that the traces of $V_{\bar{M}(k)}^h$ and $V_{M(k)}^h$ on δ_k are, generally speaking, different.

We denote $\delta_k^h = \Omega_{\bar{M}(k)}^h \cap \delta_k$ and consider three choices of the discrete Lagrange multiplier space associated with continuous piecewise linear [BM94, Kuz95], piecewise constant [AT95] and the Dirac functions, respectively:

$$\Lambda^h(\delta_k) \;:=\; \Big\{ v = \sum_{i \in \{\mathcal{N}(\delta_k^h)\}} \beta_i \psi_i, \; \psi_i = \sum_{j \in \{\mathcal{B}(\delta_k^h)\}} \frac{(\varphi_i, \varphi_j)_{L_2(\delta_k)}}{\sum\limits_{l \in \{\mathcal{N}(\delta_k^h)\}} (\varphi_l, \varphi_j)_{L_2(\delta_k)}} \varphi_j + \varphi_i \Big\} \tag{3}$$

$$\Lambda^h(\delta_k) \;:=\; \Big\{ v|_\sigma \in P_0(\sigma), \; \sigma \in \mathcal{D}(\delta_k^h) \Big\} \tag{4}$$

$$\Lambda^h(\delta_k) \;:=\; \Big\{ v = \sum_{i \in \{\mathcal{N}(\delta_k^h)\}} \beta_i \boldsymbol{\delta}(x_i) \Big\} \tag{5}$$

Here $\mathcal{N}(\delta_k^h)$ is the set of inner nodes of δ_k^h, $\mathcal{B}(\delta_k^h)$ is the set of the nodes of δ_k^h lying on $\partial\delta_k$, and $\mathcal{D}(\delta_k^h)$ is the mesh dual to δ_k^h [Fei93]. The element σ of $\mathcal{D}(\delta_k^h)$ with a center node x_i is defined via the baricentric coordinates on elements e of δ_k^h surrounding x_i, $\lambda_j(e), j = 1,2,3 : \sigma = \{x | \lambda_j(x) \geq \max\limits_{l \neq j} \lambda_l, \lambda_j(x_i) = 1\}$. Notation $\boldsymbol{\delta}(x)$ stands for the Dirac function and φ_i stands for the standard Courant basis function, while $\{A\}$ denotes the set of indexes for nodes belonging to A.

Setting

$$V^h := \prod_{i=1}^{m} V_i^h \quad \text{and} \quad \Lambda^h := \prod_{k=1}^{K} \Lambda^h(\delta_k),$$

the mortar finite element approximation of (2) requires the computation of $(u, \lambda) \in V^h \times \Lambda^h$ such that

$$\begin{aligned} a(u,v) + b(\lambda, v) &= l(v),\ v \in V^h, \\ b(\mu, u) &= 0, \quad \mu \in \Lambda^h. \end{aligned} \tag{6}$$

We note that in contrast to (3),(4), in case (5) $\Lambda^h \not\subset \Lambda$ and the mortar finite elements are nonconforming ones. Since the paper is addressing a solution procedure for (6), we do not discuss approximation properties of (6) here.

In the sequel, we denote by $A \sim B$ the spectral equivalence between the matrices A and B or proportionality between values A and B, and by c or C, with or without subscripts, positive constants.

Domain decomposition solver

General framework

The finite element problem (6) results in the system of linear algebraic equations in the saddle-point form:

$$\begin{bmatrix} A & B^T \\ B & 0 \end{bmatrix} \begin{bmatrix} u \\ \lambda \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix}, \quad \text{or} \quad \begin{bmatrix} A_1 & & & 0 & B_1^T \\ & \cdot & & & \cdot \\ & & \cdot & & \cdot \\ & & & \cdot & \cdot \\ 0 & & & A_m & B_m^T \\ B_1 & \cdot & \cdot & \cdot\ B_m & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ \cdot \\ \cdot \\ \cdot \\ u_m \\ \lambda \end{bmatrix} = \begin{bmatrix} f_1 \\ \cdot \\ \cdot \\ \cdot \\ f_m \\ 0 \end{bmatrix}, \tag{7}$$

where the block representations of the matrices A and B are associated with the definition of the spaces V^h and Λ^h, while the matrix A and the vector f are specified by the bilinear form $a(u,v)$ and the functional $l(v)$, respectively. Under the assumptions made, matrices A_i are symmetric positive definite and the whole matrix of system (7) is nonsingular.

The linear problem (7) may be solved by several iterative techniques (the reader is referred to [HIK$^+$98, Kuz95] and references therein). The construction of a preconditioner R_λ for the matrix $BA^{-1}B^T$ is one of the most important issues. Usually R_λ is called to be an interface preconditioner, or a Lagrange multiplier preconditioner. One of possible constructions is the Dirichlet-Dirichlet preconditioner [DA99, KV99]. The goal of this paper is to develop the parallel version of the Dirichlet-Dirichlet preconditioner which is robust to both the types of Lagrange multipliers spaces and the jump of coefficients.

Interface preconditioner

Let $\Gamma_i := \partial\Omega_i \setminus \partial\Omega$, n_{Γ_i} be the number of nodes of $\Gamma_i^h := \partial\Omega_i^h \cap \Gamma_i$, $M_{\Gamma_i} \in \mathbb{R}^{n_{\Gamma_i} \times n_{\Gamma_i}}$ be the boundary mass matrix, d_i be the diameter of Ω_i, $i = 1, \dots, m$.

We introduce the matrix $P_{\Gamma_i} = w_{1,\Gamma_i} w_{1,\Gamma_i}^T$, where $w_{1,\Gamma_i} = \frac{1}{\sqrt{|\Gamma_i|}} e_{\Gamma_i}$, $e_{\Gamma_i} = [1 \dots 1]^T \in \mathbb{R}^{n_{\Gamma_i}}$. We note that $(M_{\Gamma_i} w_{1,\Gamma_i}, w_{1,\Gamma_i}) = 1$, and $P_{\Gamma_i} M_{\Gamma_i}$ are the M_{Γ_i}-orthogonal projectors, $i = 1, \dots, m$. Let $\varepsilon_i \le c\rho_i/d_i^2$ and let $\bar{A}_i$ be a matrix generated on Ω_i^h by the bilinear form $a_i(u,v)$ with $\varepsilon = \rho_i/d_i^2$. The matrices A_i and $\bar{A}_i$ have the block forms

$$A_i = \begin{bmatrix} A_{\Gamma_i} & A_{\Gamma_i I_i} \\ A_{I_i \Gamma_i} & A_{I_i} \end{bmatrix} \text{ and } \bar{A}_i = \begin{bmatrix} \bar{A}_{\Gamma_i} & \bar{A}_{\Gamma_i I_i} \\ \bar{A}_{I_i \Gamma_i} & \bar{A}_{I_i} \end{bmatrix},$$

where $A_{\Gamma_i}, \bar{A}_{\Gamma_i} \in \mathbb{R}^{n_{\Gamma_i} \times n_{\Gamma_i}}$.

Lemma 1 [HIK$^+$98, Kuz95] *Under the assumptions made*

$$\left(\bar{A}_{\Gamma_i} - \bar{A}_{\Gamma_i I_i} \bar{A}_{I_i}^{-1} \bar{A}_{I_i \Gamma_i}\right)^{-1} + \frac{1}{\varepsilon_i d_i} P_{\Gamma_i} \sim \left(A_{\Gamma_i} - A_{\Gamma_i I_i} A_{I_i}^{-1} A_{I_i \Gamma_i}\right)^{-1}. \tag{8}$$

The spectral equivalence takes place with constants independent of ρ_i, ε_i, d_i.

The above Lemma is used for the construction of a preconditioner to $BA^{-1}B^T$, since
$B_i A_i^{-1} B_i^T = B_{\Gamma_i} (A_{\Gamma_i} - A_{\Gamma_i I_i} A_{I_i}^{-1} A_{I_i \Gamma_i})^{-1} B_{\Gamma_i}^T$, where matrix B_{Γ_i} is the interface subblock of B_i, $B_i = (B_{\Gamma_i}, O)$. Using (8) we have

$$BA^{-1}B^T = \sum_{i=1}^m B_i A_i^{-1} B_i^T \sim \sum_{i=1}^m \frac{1}{\varepsilon_i d_i} B_{\Gamma_i} P_{\Gamma_i} B_{\Gamma_i}^T + \bar{G}, \tag{9}$$

$$\bar{G} = \sum_{i=1}^m B_{\Gamma_i} \left(\bar{A}_{\Gamma_i} - \bar{A}_{\Gamma_i I_i} \bar{A}_{I_i}^{-1} \bar{A}_{I_i \Gamma_i}\right)^{-1} B_{\Gamma_i}^T. \tag{10}$$

Theorem 1 [KV99] *Let the symmetric positive definite matrix D be such that the spectrum of $D\bar{G}$ belongs to the interval $[c_1, c_2]$, $0 < c_1 < c_2$ and let*

$$R_\lambda := \sum_{i=1}^m \frac{1}{\varepsilon_i d_i} B_{\Gamma_i} P_{\Gamma_i} B_{\Gamma_i}^T + D^{-1}. \tag{11}$$

Then

$$R_\lambda \sim BA^{-1}B^T. \tag{12}$$

The spectral equivalence takes place with constants independent of ρ_i, ε_i, d_i, m and dependent on c_1, c_2.

Matrix R_λ is a modification of D^{-1} by a low rank matrix $XX^T = \sum\limits_{i=1}^m \frac{1}{\varepsilon_i d_i} B_{\Gamma_i} P_{\Gamma_i} B_{\Gamma_i}^T$ with $X = \left(\dots, \frac{1}{\sqrt{\varepsilon_i d_i |\Gamma_i|}} B_{\Gamma_i} e_{\Gamma_i}, \dots\right)$. The solution of a system with matrix R_λ may be found by evaluations of matrix D:

$$R_\lambda^{-1} = D - DX(I_m^{-1} + X^T D X)^{-1} X^T D,$$

where $I_m \in \mathbb{R}^{m\times m}$ is the identity matrix. Thus, in order to construct a good preconditioner for $BA^{-1}B^T$ we have to find a preconditioner D to $\bar{G}$ such that $D\bar{G} \sim I$ and D is easily multiplied by a vector.

In order to motivate our further constructions, we briefly review already developed ones. Let us suppose for a moment that $\rho_i = 1$, $i = 1, \ldots, m$. In [KV99] and in [DA99] the following constructions were investigated, respectively:

$$\widetilde{D} = \sum_{i=1}^{m} B_{\Gamma_i}\left(\bar{A}_{\Gamma_i} - \bar{A}_{\Gamma_i I_i}\bar{A}_{I_i}^{-1}\bar{A}_{I_i\Gamma_i}\right)B_{\Gamma_i}^T, \tag{13}$$

$$\widetilde{D} = (BB^T)^{-1}B\bar{A}B^T(BB^T)^{-1}. \tag{14}$$

The choice (13) provides an easy parallel implementation, while (14) is not well parallelized, since the global matrix BB^T is to be factorized. A parallel iterative inversion of BB^T seems to be too expensive in view of large condition number of BB^T (of order of 100 in cases (3),(4)). On the other hand, the choice (13) yields the small ratio c_2/c_1 only in the case (5), in contrast to (14) providing satisfactory results in the case (3). The main reason for that is a mutual annihilation of the jump matrices in the product $\widetilde{D}\bar{G} = (BB^T)^{-1}B\bar{A}B^T(BB^T)^{-1}B\bar{A}B^T$.

A natural compromise between (13) and (14) is an approximation of $(BB^T)^{-1}$ by a block diagonal matrix whose blocks are associated with interfaces. The construction of this matrix will be considered later.

Another important modification stems from the properties of the Neumann-Neumann preconditioner [DRLT91, MB96]. Preconditioning the interface Schur complement by assembling Neumann problems requires certain weights for the Neumann problems [KMV93]. By analogy with the Neumann-Neumann preconditioner we weight the Dirichlet problems in (13) by diagonal matrices w_{Γ_i}. The entries of w_{Γ_i} are reciprocal to the number of host subdomains Ω_i, for any interface node.

We return to construction of block diagonal approximation of $(BB^T)^{-1}$. Let $B_{\Gamma_i,j}$ be the j-face block of matrix B_{Γ_i}, then

$$BB^T = \sum_{i=1}^{m} B_{\Gamma_i}B_{\Gamma_i}^T = \sum_{i=1}^{m}\begin{pmatrix} B_{\Gamma_i,1}B_{\Gamma_i,1}^T & B_{\Gamma_i,1}B_{\Gamma_i,2}^T & \cdots \\ B_{\Gamma_i,2}B_{\Gamma_i,1}^T & B_{\Gamma_i,2}B_{\Gamma_i,2}^T & \cdots \\ \vdots & \vdots & \ddots \end{pmatrix} = \tag{15}$$

$$\sum_{i=1}^{m}\begin{pmatrix} B_{\Gamma_i,1}w_{\Gamma_i}B_{\Gamma_i,1}^T & & \\ & B_{\Gamma_i,2}w_{\Gamma_i}B_{\Gamma_i,2}^T & \\ & & \ddots \end{pmatrix} + \sum_{i=1}^{m}\begin{pmatrix} B_{\Gamma_i,1}(1-w_{\Gamma_i})B_{\Gamma_i,1}^T & B_{\Gamma_i,1}B_{\Gamma_i,2}^T & \cdots \\ B_{\Gamma_i,2}B_{\Gamma_i,1}^T & B_{\Gamma_i,2}(1-w_{\Gamma_i})B_{\Gamma_i,2}^T & \cdots \\ \vdots & \vdots & \ddots \end{pmatrix}.$$

Such a decomposition of matrix BB^T turns out to be numerically reasonable in the sense that the inverse of the first term in (15) is a suitable substitution for $(BB^T)^{-1}$, according to numerical evidence.

Taking into account the above observations we present the parallel version of Dirichlet-Dirichlet preconditioner, in the case $\rho_i = 1$, $i = 1, \ldots, m$:

$$D = \sum_{i=1}^{m} F_{\Gamma_i}^{-1}B_{\Gamma_i}\omega_{\Gamma_i}\left(\bar{A}_{\Gamma_i} - \bar{A}_{\Gamma_i I_i}\bar{A}_{I_i}^{-1}\bar{A}_{I_i\Gamma_i}\right)\omega_{\Gamma_i}B_{\Gamma_i}^T F_{\Gamma_i}^{-1}, \tag{16}$$

$$F_{\Gamma_i} = blockdiag\{F_{\Gamma_i,j}\}, \quad F_{\Gamma_i,j} = B_{\Gamma_i,j}\,\omega_{\Gamma_i}\,B^T_{\Gamma_i,j} + B_{\Gamma_{i^\star},j^\star}\,\omega_{\Gamma_{i^\star}}\,B^T_{\Gamma_{i^\star},j^\star}.$$

Here, $\Omega_{i^\star}$ is the neighbor-subdomain to Ω_i with shared faces j and $j^\star$. We note the factorization of F_{Γ_i} is feasible since F_{Γ_i} is a sparse matrix.

Now let $\rho_i > 0$ be arbitrary. Then

$$\bar{G} = \sum_{i=1}^{m} B_{\Gamma_i}\left(\bar{A}_{\Gamma_i} - \bar{A}_{\Gamma_i I_i}\bar{A}^{-1}_{I_i}\bar{A}_{I_i\Gamma_i}\right)^{-1} B^T_{\Gamma_i} \equiv \tag{17}$$

$$\equiv \sum_{i=1}^{m} \frac{1}{\sqrt{\rho_i}} B_{\Gamma_i}\left(\rho_i^{-1}\bar{A}_{\Gamma_i} - \rho_i^{-1}\bar{A}_{\Gamma_i I_i}\bar{A}^{-1}_{I_i}\bar{A}_{I_i\Gamma_i}\right)^{-1} B^T_{\Gamma_i}\frac{1}{\sqrt{\rho_i}},$$

and the problem of construction D is reduced to the case $\rho_i = 1$ by substitutions $B_{\Gamma_i} \rightarrow B_{\Gamma_i}/\sqrt{\rho_i}$, $\bar{A}_i \rightarrow \bar{A}_i/\rho_i$. Thus, the general form of the Dirichlet-Dirichlet preconditioner is

$$D = \sum_{i=1}^{m} F^{-1}_{\Gamma_i}\frac{1}{\sqrt{\rho_i}} B_{\Gamma_i}\omega_{\Gamma_i}\left(\rho_i^{-1}\bar{A}_{\Gamma_i} - \rho_i^{-1}\bar{A}_{\Gamma_i I_i}\bar{A}^{-1}_{I_i}\bar{A}_{I_i\Gamma_i}\right)\omega_{\Gamma_i}B^T_{\Gamma_i}\frac{1}{\sqrt{\rho_i}}F^{-1}_{\Gamma_i},$$

$$F_{\Gamma_i} = blockdiag\{F_{\Gamma_i,j}\},$$

$$F_{\Gamma_i,j} = \frac{1}{\sqrt{\rho_i}}B_{\Gamma_i,j}\,\omega_{\Gamma_i}\,B^T_{\Gamma_i,j}\frac{1}{\sqrt{\rho_i}} + \frac{1}{\sqrt{\rho_{i^\star}}}B_{\Gamma_{i^\star},j^\star}\,\omega_{\Gamma_{i^\star}}\,B^T_{\Gamma_{i^\star},j^\star}\frac{1}{\sqrt{\rho_{i^\star}}},$$

which may be rewritten as:

$$D = \sum_{i=1}^{m} F^{-1}_{\Gamma_i}B_{\Gamma_i}\omega^\rho_{\Gamma_i}\left(\bar{A}_{\Gamma_i} - \bar{A}_{\Gamma_i I_i}\bar{A}^{-1}_{I_i}\bar{A}_{I_i\Gamma_i}\right)\omega^\rho_{\Gamma_i}B^T_{\Gamma_i}F^{-1}_{\Gamma_i}, \tag{18}$$

$$F_{\Gamma_i,j} = B_{\Gamma_i,j}\,\omega^\rho_{\Gamma_i}\,B^T_{\Gamma_i,j} + B_{\Gamma_{i^\star},j^\star}\,\omega^\rho_{\Gamma_{i^\star}}\,B^T_{\Gamma_{i^\star},j^\star}, \quad \omega^\rho_{\Gamma_i} = \omega_{\Gamma_i}/\rho_i.$$

It is clear that (18) differs from (16) only in the scaled count matrices $w^\rho_{\Gamma_i}$.

Remark. The presence of Dirichlet boundary conditions for the original problem reduces the rank of XX^T since subdomains with a Dirichlet part of the boundary do not contribute to XX^T [Kuz95].

Numerical experiments

We present the effect of the Dirichlet-Dirichlet preconditioner for the model operator $-\nabla \cdot \rho\nabla + \varepsilon$ with Neumann boundary conditions. The domain Ω is a union of four similar tetrahedra Ω_i sharing one common edge:

$$\Omega = \left\{ x \mid \sum_{i=1}^{3}|x_i| < \frac{1}{2},\, x_1 > 0 \right\},\ m = 4,$$

$$\Omega_1 = \{x \in \Omega, x_2 < 0, x_3 < 0\},\ \Omega_2 = \{x \in \Omega, x_2 < 0, x_3 > 0\},$$

$$\Omega_3 = \{x \in \Omega, x_2 > 0, x_3 < 0\},\ \Omega_4 = \{x \in \Omega, x_2 > 0, x_3 > 0\}.$$

We compare the Dirichlet-Dirichlet preconditioner R_λ for $BA^{-1}B^T$ in three cases of Lagrange multiplier spaces, (3), (4), (5). The comparison will be done for different types of tetrahedral meshes: quasi-uniform, shape-regular, and anisotropic. We distinguish the above types of meshes by the metric $H = diag\{H_1, H_2, H_3\}$ in which the meshes Ω_i^h become quasi-uniform, i.e. consist of the given number N_T of shape-regular (in metric H) tetrahedra of the same size (in metric H). In the tables below we show the estimated condition number of preconditioned Schur complement $BA^{-1}B^T$ and the number of PCG iterations applied to a system with $BA^{-1}B^T$ in order to reduce the Euclidean norm of residual by a factor of 10^6.

Coef.	Mesh	quasi-uniform			isotropic		anisotropic	
ρ_i	N_T	800	6000	39000	800	6000	800	6000
		$\Lambda^h(\delta_k)$ from (3)						
$\rho_{1,2,3,4} = 1$	cond(#it)	29(19)	45(25)	29(26)	37(17)	44(20)	26(17)	32(22)
$\rho_{1,2} = 1, \rho_{3,4} = 10^4$	cond(#it)	36(13)	18(11)	17(12)	81(23)	170(24)	59(18)	18(13)
$\rho_{1,3} = 1, \rho_{2,4} = 10^4$	cond(#it)	52(19)	83(23)	41(21)	91(21)	81(22)	44(17)	69(22)
		$\Lambda^h(\delta_k)$ from (4)						
$\rho_{1,2,3,4} = 1$	cond(#it)	29(20)	36(25)	28(24)	34(16)	36(17)	23(16)	29(19)
$\rho_{1,2} = 1, \rho_{3,4} = 10^4$	cond(#it)	32(13)	19(12)	17(12)	74(20)	170(22)	51(19)	18(12)
$\rho_{1,3} = 1, \rho_{2,4} = 10^4$	cond(#it)	49(19)	83(24)	35(21)	83(20)	81(22)	35(15)	65(23)
		$\Lambda^h(\delta_k)$ from (5)						
$\rho_{1,2,3,4} = 1$	cond(#it)	17(16)	18(20)	20(26)	17(14)	18(18)	16(13)	18(18)
$\rho_{1,2} = 1, \rho_{3,4} = 10^4$	cond(#it)	20(12)	41(19)	22(20)	20(11)	36(17)	23(12)	20(17)
$\rho_{1,3} = 1, \rho_{2,4} = 10^4$	cond(#it)	18(13)	19(15)	21(19)	18(10)	19(12)	18(10)	19(13)

Table 1: Condition number of $R_\lambda^{-1}BA^{-1}B^T$ and #PCG iteration, $\varepsilon = 1$.

In Table 1 the quasi-uniform mesh is obtained on the basis of the metric $H_1 = H_2 = H_3 = 1$, and the isotropic and anisotropic refinements to the common edge are defined by $H_1 = H_2 = H_3 = 0.5/(\sqrt{y^2 + z^2} + 0.01)$ and $H_1 = 1, H_2 = H_3 = 0.5/(\sqrt{y^2 + z^2} + 0.025)$, respectively. The meshes are generated in such a way that they do not match on the interfaces. It implies that the number of tetrahedra in Ω_i^h is equal to N_T only approximately. We consider three different distributions of coefficients ρ_i in Ω: no jump, two simply connected subdomains with constant coefficient, and the chess pattern.

In the next example we consider the effects of small value of coefficient ε_i and large number of subdomains m. The domain $\Omega = (0, 1)^3$ is split into $m = 6$ (resp. 48 or 384) tetrahedron subdomains Ω_i of the same diameter $d_i = \sqrt{3}$ (resp. $\sqrt{3}/2$ or $\sqrt{3}/4$), $i = 1, \ldots, m$. We consider the Helmholtz operator $-\Delta + \varepsilon$ with homogeneous Neumann boundary condition on $\partial\Omega$ and restrict the set of possible triangulations by quasi-uniform ones and take the Lagrange multiplier space (3). The number of tetrahedra N_T in Ω_i^h is chosen to be equal to 800. Slightly worse performance in the cases $m = 48, 384$ is due to presence of the crosspoints.

In Table 3 we present the parallel properties of the method in terms of the execution time of PCG iterations measured on different sets of processors. The measurement was obtained using a DEC TruCluster with Dec alpha processors running at 400 MHz.

$\varepsilon \setminus m$		$m=6$	$m=48$	$m=384$
$\varepsilon=1$	cond (# it)	26(26)	64(44)	84(45)
$\varepsilon=10^{-2}$	cond (# it)	35(20)	61(33)	62(28)
$\varepsilon=10^{-4}$	cond (# it)	29(15)	40(18)	19(7)

Table 2: Condition number of $R_{\lambda}^{-1}BA^{-1}B^T$ and #PCG iteration, quasi-uniform meshes, $\rho_{1,2,3,4}=1$, $\Lambda^h(\delta_k)$ from (3).

The Fortran code uses MPI library for interprocessor communications.

N_T	#Processors	2	4	8
800	time of PCG it.	3.0	1.5	0.9
2000	time of PCG it.	7.7	3.8	1.9

Table 3: Execution time of PCG iterations (sec), $m=48$, quasi-uniform meshes, $\varepsilon=1$, $\rho_{1,2,3,4}=1$.

Conclusions

The paper is addressing the construction of parallel interface preconditioner for the mortar element method. The new version of the Dirichlet-Dirichlet method is discussed. It is easy to parallel and it is robust to such "bad" parameters of an elliptic boundary value problem as the number of subdomains, the mesh refinement, the jump of the diffusion coefficient, the small value of perturbation parameter. Numerical experiments exhibited the basic properties of the method.

References

[AT95]A. Agouzal and J.-P. Thomas. Une méthode d'élement finis hybridesen décomposition de domaines. *RAIRO M²AN*, 29:749–764, 1995.

[BF91]F. Brezzi and M. Fortin. *Mixed and Hybrid Finite Element Methods.* Springer-Verlag, New-York, 1991.

[BM94]Faker Ben Belgacem and Yvon Maday. The mortar element method for three dimensional finite elements. Technical Report 94-16, Université Paul Sabatier, 1994.

[DA99]D.Stefanica and A.Klawonn. The FETI method for mortar finite elements. In C.-H. Lai, P.E. Bjorstad, M. Cross, and O.Widlund, editors, *Proceedings of 11th International Conference on Domain Decomposition Methods*, pages 121–129. DDM.org, 1999.

[DRLT91]Yann-Hervé De Roeck and Patrick Le Tallec. Analysis and test of a local domain decomposition preconditioner. In Roland Glowinski, Yuri Kuznetsov, Gérard Meurant, Jacques Périaux, and Olof Widlund, editors, *Fourth International Symposium on Domain Decomposition Methods for Partial Differential Equations*, pages 112–128. SIAM, Philadelphia, PA, 1991.

[Fei93]Miloslav Feistauer. *Mathematical Methods in Fluid Dynamics.* Longman Scientific & Technical, Harlow, England, 1993.

[HIK$^+$98]R. Hoppe, Yu. Iliash, Yu. Kuznetsov, Yu. Vassilevski, and B. Wohlmuth. Analysis and parallel implementation of adaptive mortar element methods. *East West J. Num. An.*, 6(3):223–248, 1998.

[KMV93]Yuri Kuznetsov, Petri Manninen, and Yuri Vassilevski. On numerical experiments with Neumann-Neumann and Neumann-Dirichlet domain decomposition preconditioners. Technical report, University of Jyväskylä, 1993.

[Kuz95]Yuri A. Kuznetsov. Efficient iterative solvers for elliptic problems on nonmatching grids. *Russ. J. Numer. Anal. Math. Modelling*, 10(3):187–211, 1995.

[KV99]Yu. Kuznetsov and Yu. Vassilevski. A Dirichlet-Dirichlet preconditioner for the mortar element method. Technical Report B 12/1999, University of Jyvaskyla, Dprt. of MIT, University of Jyvaskyla, Finland, 1999.

[MB96]J. Mandel and M. Bresina. Balancing domain decomposition for problems with large jumps in coefficients. *Math.Comp.*, 65(216):1387–1401, 1996.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau, ©2001 DDM.org

26. The Application of Operator Split Method to Large-Scale Reservoir Simulatios Part I. a Priori Estimates

H. Zhang[1]

Introduction

High performance computing technology offers the petroleum industry the ability to solve previously prohibitive large-scale reservoir problems. In July 1999, our group, in cooperation with the Petroleum Exploration and Development Institute of Daqing Oil Field, China, ran a million-gridblock-scale reservoir simulation on DAWN 2000, which is a home-made supercomputer, and a loosely coupled PC cluster seperately. The parallel computing methods that we used derived from the domain decomposition methods with no overlap. The next goal of our group is to solve reservoir simulations with millions of gridblocks on parallel machines. Unfortunately, it seems that the original computing method is not scalable enough. We believe that the reason is rather geologic than mathematical. As the simulating area becomes larger and larger, the geologic faults will be more and more complicated. Therefore, the non-matching grids on the interfaces of the substructures will be increasing largely, and possessing entirely different properties. This will inevitably lead to the poor performance of the original computing methods.

The purpose of this paper is trying to find an effective way to remove as many of the geologic non-matching grids as possible from the interfaces. The operator split method, not a very new technique, proposed by Douglas and Dupont[JD71], can solve this problem. Because, for quite a few reservoir problems, the reservoir Ω can be taken to be unions of right prisms. Or, mathematically, $\Omega = \cup\Omega_i$, where $\Omega_i = \Omega_{xy}^i \times [0, l_i]$, $\Omega_{xy}^i \subset \mathbb{R}^2$. When only upright wells are available, the original reservoir problem can be divided into an xy-direction, two-dimensional problem and a z-direction, one-dimensional problem in some of the subdomains. So, the geologic non-matching grids on the interfaces can be greatly reduced.

For a detailed introduction of operator split method, see [JD71] and [Mar90]. Generalizations of this method to parabolic problems on nonrectangular regions were presented by Hayes [Hay81]. Special treatments for convection-diffusion problems, parabolic and hyperbolic equations were considered by Krishnamachari, Hayes and Russell[SHR89] (without theoretical analysis), Bramble, Ewing and Li[BEL89], Bialecki and Fernandes[BF93], and Fernandes and Fairweather [FF91]. Applications of these methods to problems in fluid flow, physics of semiconductors and elastoplastic dynamics were described by Hayes and Krishnamachari [HK84], Berezin and Yanenko[BY84], and Migual, Pinsky and Taylor[MPT83].

The main purpose of using operator split method here is to reduce the geologic non-matching grids on the interfaces, instead of saving the memory costs and the

[1] R & D Center for Parallel Software, Institute of Software, Chinese Academy of Sciences, zhy@mail.rdcps.ac.cn

storage requirements as before. Based upon this method, we can further formulate the domain decomposition algorithms. We expect that this combining method can perform good scalability, and have nearly the same accuracy as the original methods, which will be proved in this paper.

In this paper, we will make some a priori estimates for the operator split method for reservoir problems. An optimal H^1 convergence rate will be proved. It is necessary to make a major, and probably unphysical assumption, as was done in [JR83], [Yua92], [Che94] and [JDEW83], that the sources and sinks are smoothly distributed and the resulting functions of interest are thus fairly smooth in space. The techniques involved to prove the error bounds are quite different from the standard ones presented by Douglas, Wheeler and Ewing, et.al.[JR83] [Yua92][JDEW83].

In this paper, we consider the single-phase, miscible displacement of one compressible fluid with another in a porous medium. A set of model equations is given as follows. For a more detailed description of the physical problem, see [Pea66]. Find the concentration $c = c(x,t)$ and $p = p(x,t)$ that satisfy the following equations:

$$d(c)\frac{\partial p}{\partial t} + \bigtriangledown \cdot u = d(c)\frac{\partial p}{\partial t} - \bigtriangledown \cdot (a(c) \bigtriangledown p) = q, \quad x \in \Omega, t \in \mathcal{J} \tag{1}$$

$$\phi\frac{\partial c}{\partial t} + b(c)\frac{\partial p}{\partial t} + u \cdot \bigtriangledown c - \bigtriangledown \cdot (D(u) \bigtriangledown c) = (\hat{c} - c)q, \quad x \in \Omega, t \in \mathcal{J} \tag{2}$$

where initial conditions and no flow boundary conditions are given by

$$p(x,0) = p_0(x), \qquad x \in \Omega \tag{3}$$

$$c(x,0) = c_0(x), \qquad x \in \Omega \tag{4}$$

and

$$u \cdot \nu = 0, \qquad x \in \partial\Omega \tag{5}$$

$$(D \bigtriangledown c - cu) \cdot \nu = 0, \qquad x \in \partial\Omega \tag{6}$$

For simplicity, denote $\Omega = \Omega_{xy} \times [0,l]$, $\mathcal{J} = (0,T]$, and ν is the outward unit normal vector on $\partial\Omega$, the boundary of Ω. Here $a(c), b(c), d(c), \phi = \phi(x)$ are specific reservoir and fluid properties, u is the Darcy velocity of the fluid, $D(u)$ is the diffusion coefficient matrix which combines the effects of molecular diffusion and mechanical dispersion, $\hat{c}$ is the specific concentration at injection wells and the resident concentration at production wells, and $q = q(x,t)$ is the imposed external flow, positive for injection and negative for production.

In [JR83] the authors presented and analyzed certain numerical approximations for a two dimensional model. Extensions of these methods to more efficient time-stepping procedures and methods of characteristics[Yua92] have since been developed.

The paper is organized as follows: In §2, the variational form and the elliptic projections of the problem are introduced. In §3, the numerical procedures are described. In §4, some a priori estimates are presented, and in §5, the amount of calculations of the operator split method are estimated.

Variations and Projections

To obtain a variational form of (1) and (2), we multiply (1) and (2) by test functions $v, w \in H^1(\Omega)$, and integrate by parts, respectively. This yields

$$(\phi \frac{\partial c}{\partial t}, w) + (b(c)\frac{\partial p}{\partial t}, w) + (u \cdot \bigtriangledown c, w) + (D(u) \bigtriangledown c, \bigtriangledown w)$$

$$= ((\hat{c} - c)q, w), \qquad w \in H^1(\Omega), t \in \mathcal{J} \tag{7}$$

$$(d(c)\frac{\partial p}{\partial t}, v) + (a(c) \bigtriangledown p, \bigtriangledown v) = (q, v), \qquad v \in H^1(\Omega), t \in \mathcal{J} \tag{8}$$

Let $\mathcal{M}_h = \mathcal{M}_{h_c}, \mathcal{N}_h = \mathcal{N}_{h_p} \subset W^{1,\infty}$ denote the finite element spaces spanned by tensor product bases, where $\mathcal{M}_h = \text{span}[\psi_i^{(xy)}(x,y) \times \psi_j^{(z)}(z)], \mathcal{N}_h = \text{span}\ [\bar{\psi}_i^{(xy)}(x,y) \times \bar{\psi}_j^{(z)}(z)]$, and $\mathcal{M}_h, \mathcal{N}_h$ satisfy

$$\inf_{w_h \in \mathcal{M}_h} \|w - w_h\|_{1,q} \le K\|w\|_{l+1,q} h_c^l, \quad w \in W^{l+1,q}, \quad 1 \le q \le \infty \tag{9}$$

and

$$\inf_{v_h \in \mathcal{N}_h} \|v - v_h\|_{1,q} \le K\|v\|_{r+1,q} h_p^r, \quad v \in W^{r+1,q}, \quad 1 \le q \le \infty \tag{10}$$

respectively. We assume that all standard inverse relations hold on $\mathcal{M}_h$ and $\mathcal{N}_h$.

We project the solution of the differential problem (1) and (2) into the finite element spaces by means of coercive elliptic forms associated with the differential system. First, for $t \in \mathcal{J}$, let $\tilde{c} = \tilde{c}_h : \mathcal{J} \to \mathcal{M}_h$ be determined by the relations:

$$(D(u) \bigtriangledown (c - \tilde{c}), \bigtriangledown w) + (u \cdot \bigtriangledown(c - \tilde{c}), w) + \sigma_1(c - \tilde{c}, w) = 0, \qquad w \in \mathcal{M}_h \tag{11}$$

where the constant σ_1 is chosen to be large enough to insure the coercivity of the bilinear form over $H^1(\Omega)$.

Similarly, let $\tilde{p} = \tilde{p}_h : calJ \to \mathcal{N}_h$ satisfy

$$(a(c) \bigtriangledown (p - \tilde{p}), \bigtriangledown v) + \sigma_2(p - \tilde{p}, v) = 0, \qquad v \in \mathcal{N}_h \tag{12}$$

where σ_2 is assumed to be coercive over $H^1(\Omega)$.

Let:

$$\zeta^n = c^n - \tilde{c}^n,\ \mathcal{E}^n = \tilde{c}^n - C^n,\ \eta^n = p^n - \tilde{p}^n,\ \pi^n = \tilde{p}^n - P^n$$

If the following restrictions are valid:

(i)q is smoothly distributed, the coefficients are smooth, therefore the solution is smooth.

(ii)The coefficients a, d and ϕ are positively bounded below, as well as being smooth.

$$0 < a_* \le a(c) \le a^*, \quad 0 < d_* \le d(c) \le d^*, \quad 0 < \phi_* \le \phi(x) \le \phi^* \tag{13}$$

$D = (D_{ij}(u))_{3\times 3}$ is a positive definite matrix, and there exist constants D_*, D^*, $0 < D_* \le D^*$, such that for $\forall w \in R^2$,

$$D_*|w|^2 \le (D(u)w, w) \le D^*|w|^2 \tag{14}$$

It follows from [JR83],[Che94] that:

$$\|\zeta\|_{L^2} + h_c\|\zeta\|_{H^1} + \|\frac{\partial\zeta}{\partial t}\|_{L^2} + h_c\|\frac{\partial\zeta}{\partial t}\|_{H^1} \le K\{\|c\|_{H^{l+1}} + \|\frac{\partial c}{\partial t}\|_{H^{l+1}}\}h_c^{l+1} \tag{15}$$

$$\|\eta\|_{L^2} + h_p\|\eta\|_{H^1} + \|\frac{\partial\eta}{\partial t}\|_{L^2} + h_p\|\frac{\partial\eta}{\partial t}\|_{H^1} \le K\{\|p\|_{H^{r+1}} + \|\frac{\partial p}{\partial t}\|_{H^{r+1}}\}h_p^{r+1} \tag{16}$$

$$\|\tilde{c}\|_{W^1_\infty(\mathcal{J};W^1_\infty)} + \|\tilde{p}\|_{W^1_\infty(\mathcal{J};W^1_\infty)} \le K, \quad \|\frac{\partial^2\eta}{\partial t^2}\|_{H^1} \le Kh_p^r \tag{17}$$

$$\|\frac{\partial^3\eta}{\partial t^3}\|_{L^\infty} + \|\bigtriangledown\frac{\partial^2\eta}{\partial t^2}\|_{L^\infty} + \|\bigtriangledown\frac{\partial^2\zeta}{\partial t^2}\|_{L^\infty} \le K \tag{18}$$

where K is a positive constant that does not depend on h_c and h_p.

The Numerical Procedures

In this section, we present the numerical procedures of (1) and (2) by using operator split methods. The associated matrix problem, however, will not factor, since, in general, ϕ and $d(c)$ are not single tensor products. So, on the left-hand side of (7) and (8), ϕ and $d(c)$ are replaced with certain type of patch approximations, respectively. Using the approximate $\tilde{\phi}$ and d^n, perturbation terms can be added to the matrix problem, so that it does factor as desired. For $C = \sum_{i=1}^{m_c}\mu_i\psi_i$, $P = \sum_{i=1}^{m_p}\gamma_i\bar{\psi}_i$ and $w = \sum_{j=1}^{m_c}\nu_j\psi_j$, $v = \sum_{j=1}^{m_p}\kappa_j\bar{\psi}_j$, define

$$(\tilde{\phi}C, w) = \int_\Omega\{\sum_{i,j=1}^{m_c}\mu_i\psi_i\nu_j\psi_j\tilde{\phi}_{ij}\}dx \tag{19}$$

$$(d^nP, v) = \int_\Omega\{\sum_{i,j=1}^{m_p}\gamma_i\bar{\psi}_i\kappa_j\bar{\psi}_j d^n_{ij}\}dx \tag{20}$$

where

$$\tilde{\phi}_{ij} = \sqrt{\phi(x^i)\cdot\phi(x^j)} \quad, x^i \in supp(\psi_i) \tag{21}$$

$$d^n_{ij} = \sqrt{d(x^i, C^n)\cdot d(x^j, C^n)} \quad, x^i \in supp(\bar{\psi}_i) \tag{22}$$

The three-level operator split method is defined by finding $\{C^n, P^n\} \in \mathcal{M}_h \times \mathcal{N}_h$ such that

$$(\tilde{\phi}\partial_t C^n, w) + (U^n\cdot\bigtriangledown C^n, w) + (D(U^n)\bigtriangledown C^n, \bigtriangledown w) + (b(C^n)\partial_t P^n, w)$$

$$+\lambda_1\Delta t(\tilde{\phi}\bigtriangledown\partial_t C^n, \bigtriangledown w) + \lambda_1^2(\Delta t)^2(\tilde{\phi}\frac{\partial^2}{\partial x\partial z}\partial_t C^n, \frac{\partial^2}{\partial x\partial z}w)$$

$$+\lambda_1^2(\Delta t)^2(\tilde{\phi}\frac{\partial^2}{\partial y\partial z}\partial_t C^n, \frac{\partial^2}{\partial y\partial z}w)$$

$$= ((\hat{C}^n - C^n)q^n, w) + ((\tilde{\phi} - \phi)\partial_t C^{n-1}, w), \quad w \in \mathcal{M}_h \tag{23}$$

suppose that U^n is given by

$$U^n = -a(C^n) \bigtriangledown P^n, \quad \text{for } \forall x \in \Omega \tag{24}$$

and

$$(d^n \partial_t P^n, v) + (a(C^n) \bigtriangledown P^n, \bigtriangledown v) + \lambda_2 \Delta t(d^n \bigtriangledown \partial_t P^n, \bigtriangledown v)$$

$$+\lambda_2^2(\Delta t)^2(d^n \frac{\partial^2}{\partial x\partial z}\partial_t P^n, \frac{\partial^2}{\partial x\partial z}v) + \lambda_2^2(\Delta t)^2(d^n \frac{\partial^2}{\partial y\partial z}\partial_t P^n, \frac{\partial^2}{\partial y\partial z}v)$$

$$= (q^n, v) + ((d^n - d(C^n))\partial_t P^{n-1}, v), \quad v \in \mathcal{N}_h \tag{25}$$

where the computing order is $C^1, P^2, U^2, C^2, P^3, U^3, \cdots$. For stability, we require that $\lambda_1 > \frac{1}{2}D^*/\phi_*$ and $\lambda_2 > a^*/d_*$. We assume that the initial time steps are chosen small enough, so that $P^1 = P^0 = p_0$, and the initial values of C^1 are derived through some kind of iterative methods.

If we notice the fact that the concentration equation is normally convection-dominated, a scheme combining the operator split procedure with the method of characteristics can be defined by employing an approximation to the following characteristic vector. For each (x, t), we let $\tau(x, t)$ be the unit vector in the indicated characteristic direction such that

$$\frac{\partial}{\partial \tau(x,t)} = \frac{u(x,c,\bigtriangledown p)}{\sqrt{|u(x,c,\bigtriangledown p)|^2 + \phi^2(x)}}\frac{\partial}{\partial x} + \frac{\phi(x)}{\sqrt{|u(x,c,\bigtriangledown p)|^2 + \phi^2(x)}}\frac{\partial}{\partial t} \tag{26}$$

$$= (|u|^2 + \phi^2)^{-1/2}(u_1\frac{\partial}{\partial x} + u_2\frac{\partial}{\partial y} + u_3\frac{\partial}{\partial z} + \phi\frac{\partial}{\partial t}) \tag{27}$$

Let $\phi_c = (|u|^2 + \phi^2)^{1/2}$, we then see that (2) is equivalent to

$$\phi_c \frac{\partial c}{\partial \tau(x,t)} + b(c)\frac{\partial p}{\partial t} - \bigtriangledown \cdot (D(u) \bigtriangledown c) = (\hat{c} - c)q \tag{28}$$

and the variational form (7) becomes

$$(\phi_c \frac{\partial c}{\partial \tau}, w) + (b(c)\frac{\partial p}{\partial t}, w) + (D(u) \bigtriangledown c, \bigtriangledown w) = ((\hat{c} - c)q, w), \quad w \in H^1, t \in \mathcal{J} \tag{29}$$

When solving for C^{n+1}, we define for each $x \in \Omega$,

$$\bar{x} = x - \frac{U^n(x)}{\phi(x)}\Delta t, \quad \bar{C}^n(x) = C^n(\bar{x}) \tag{30}$$

It is assumed that no flow occurs across the boundary. If $\bar{x}$ crosses over the boundary $\partial\Omega$, we can replace it with its mirror image point along the normal direction of $\partial\Omega$. We represented this point by $\bar{\bar{x}}$. Therefore, $\bar{C}^n$ is well defined. To approximate (29), we use a backward difference quotient for $\partial c/\partial\tau$ along the characteristic. Specifically, we take

$$(\frac{\partial c}{\partial \tau})^{n+1}(x) \approx \phi \frac{c^{n+1}(x) - c^n(\bar{x})}{\Delta t \phi_c} \tag{31}$$

so that

$$\phi_c \frac{\partial c^{n+1}}{\partial \tau} \approx \phi \frac{c^{n+1} - \bar{c}^n}{\Delta t} \tag{32}$$

The numerical scheme based on combining the operator split procedure with the method of characteristics for the concentration equation can be defined as

$$\begin{aligned}
&(\tilde{\phi}\partial_t C^n, w) + (U^n \cdot \triangledown C^n, w) + (D(U^n) \triangledown C^n, \triangledown w) + (b(C^n)\partial_t P^n, w) \\
&+\lambda_1 \Delta t(\tilde{\phi} \triangledown \partial_t C^n, \triangledown w) + \lambda_1^2 (\Delta t)^2 (\tilde{\phi} \frac{\partial^2}{\partial x \partial z} \partial_t C^n, \frac{\partial^2}{\partial x \partial z} w) \\
&+\lambda_1^2 (\Delta t)^2 (\tilde{\phi} \frac{\partial^2}{\partial y \partial z} \partial_t C^n, \frac{\partial^2}{\partial y \partial z} w) = ((\hat{C^n} - C^n) q^n, w) \\
&+((\tilde{\phi} - \phi)\partial_t C^{n-1}, w) - (\phi \frac{C^n - \bar{C}^n}{\Delta t}, w), \qquad w \in \mathcal{M}_h
\end{aligned} \tag{33}$$

The matrix problem associated with (23)-(25) , similarly for (33),(24),(25), is given by

$$\mathbb{K}_c^n (\mu^{n+1} - \mu^n) = \Phi^n \tag{34}$$

$$\mathbb{K}_p^n (\gamma^{n+1} - \gamma^n) = \Psi^n \tag{35}$$

where

$$\mathbb{K}_c^n = (Diag_c)^{1/2} \mathbb{K}_c (Diag_c)^{1/2} \quad , \mathbb{K}_p^n = (Diag_p^n)^{1/2} \mathbb{K}_p (Diag_p^n)^{1/2}$$

$$Diag_c = \begin{bmatrix} \tilde{\phi}(x^1) & & \\ & \ddots & \\ & & \tilde{\phi}(x^{m_c}) \end{bmatrix}, Diag_p^n = \begin{bmatrix} d(x^1, C^n) & & \\ & \ddots & \\ & & d(x^{m_p}, C^n) \end{bmatrix}$$

$$\mathbb{K}_c^{ij} = ((\psi_j, \psi_i) + \lambda_1 \Delta t(\triangledown \psi_j, \triangledown \psi_i) + \lambda_1^2 (\Delta t)^2 [(\frac{\partial^2 \psi_j}{\partial x \partial z}, \frac{\partial^2 \psi_i}{\partial x \partial z}) + (\frac{\partial^2 \psi_j}{\partial y \partial z}, \frac{\partial^2 \psi_i}{\partial y \partial z})])$$

$$\mathbb{K}_p^{ij} = ((\bar{\psi}_j, \bar{\psi}_i) + \lambda_2 \Delta t(\triangledown \bar{\psi}_j, \triangledown \bar{\psi}_i) + \lambda_2^2 (\Delta t)^2 [(\frac{\partial^2 \bar{\psi}_j}{\partial x \partial z}, \frac{\partial^2 \bar{\psi}_i}{\partial x \partial z})(\frac{\partial^2 \bar{\psi}_j}{\partial y \partial z}, \frac{\partial^2 \bar{\psi}_i}{\partial y \partial z})])$$

$$\begin{aligned}
\Phi_i^n &= ((\hat{C}^n - C^n) q^n, \psi_i) - (U^n \cdot \triangledown C^n, \psi_i) - (D(U^n) \triangledown C^n, \triangledown \psi_i) \\
&\quad -(b(C^n)\partial_t P^n, \psi_i) + ((\tilde{\phi} - \phi)\partial_t C^{n-1}, \psi_i) \\
\Psi_i^n &= (q^n, \bar{\psi}_i) - (a(C^n) \triangledown P^n, \triangledown \bar{\psi}_i) + ((d^n - d(C^n))\partial_t P^{n-1}, \bar{\psi}_i)
\end{aligned}$$

Notice that $\mathcal{M}_h$ and $\mathcal{N}_h$ are spanned by tensor product bases, so K_c and K_p can be rewritten in the following manner:

$$[\mathbb{I} \otimes (\mathbb{C}_{xy} + \lambda \Delta t \mathbb{A}_{xy})][(\mathbb{C}_z + \lambda \Delta t \mathbb{A}_z) \otimes \mathbb{I}] \tag{36}$$

where $\mathbb{C}_{xy}, \mathbb{A}_{xy}$ correspond to a two-dimensional problem in horizontal planes of Ω, while $\mathbb{C}_z, \mathbb{A}_z$ to a one-dimensional problem along the vertical lines in Ω.

"A Priori" Error Estimates

In order to derive the optimal H^1 error estimates for the procedures (23)-(25), and (33),(24),(25), We need to let ∂_t act on the both sides of the error equation of the pressure equation. Quite a few of the technical treatments were involved. After a careful calculation, we obtain

Theorem 1 *Suppose the restrictions of §2 be satisfied, and there is no flow at the initial time, i.e.* $p_0 \equiv const$. *The parameters* h_p *and* h_c *are chosen such that* $h_p^r = o(h_c), h_c^l = o(h_p)$, $r, l \geq 2$, $\Delta t = O(h_c^2) = O(h_p^2)$. *If* $\lambda_1 > \frac{1}{2} D^*/\phi_*$, $\lambda_2 > a^*/d_*$, *and the initial values of* C^0 *and* C^1 *satisfy*

$$\|C^1 - c^1\|_{H^1}^2 + \Delta t \|\partial_t (C-c)^0\|_{L^2}^2 \leq K(h_p^{2r} + h_c^{2l} + (\Delta t)^2)$$

Then for h_c *and* h_p *sufficiently small, we have*

$$\max_{1 \leq n \leq M} \{\|C^n - c^n\|_{H^1}^2 + \|U^n - u^n\|_{L^2}^2\}$$

$$+\Delta t \sum_{n=1}^{M-1} \{\|\partial_t (C-c)^n\|_{L^2}^2 + \|\partial_t (P-p)^n\|_{L^2}^2\} \leq K(h_p^{2r} + h_c^{2l} + (\Delta t)^2)$$

From the estimates, we know that the operator split method can maintain the optimal H^1 accuracy. Therefore, the new parallel computing method, the DDM combining with the operator split method, can have nearly the same numerical accuracy as the origianal method we used before.

Work Estimates

Suppose there are $m_p = m(h_p), m_c = m(h_c)$ unknowns for the pressure equation and the concentration equation respectively. The factorization of the matrices $\mathbb{K}_c$ and $\mathbb{K}_p$ requires $O(m_c^{3/2} + m_p^{3/2})$ operations, but this is done only once and used at all successive time steps. The evaluation of $(Diag_c)^{-1/2}$ requires $O(m_c)$ operations, while $(Diag_p^n)^{-1/2}$ requires $O(m_p)$ operations for each time level. The solution, given the factorization of $\mathbb{K}_c$ and $\mathbb{K}_p$, requires $O(m_c \log m_c + m_p \log m_p)$ operations. If $\Delta t = O(h_p^r) = O(h_c^l)$, i.e. $\Delta t = O(m_p^{-r/3}) = O(m_c^{-l/3})$, and $r, l \geq 2$, then the total number of operations needed is $O(m_p^{r/3+1} \log m_p + m_c^{l/3+1} \log m_c)$, which is nearly optimal since the solution is defined by $O(m_p^{r/3+1} + m_c^{l/3+1})$ parameters for a first-order correct-in-time method.

Acknowledgement. The author would like to thank Prof. Jiachang Sun and Prof. Yirang Yuan for suggesting this problem, and for many helpful discussions.

References

[BEL89]J.H. Bramble, R.E. Ewing, and G. Li. Alternating direction multistep methods for parabolic problems-iterative stabilization. *Siam J.Numer.Anal.*, 26(4):904–919, August 1989.

[BF93]B. Bialecki and R.I. Fernandes. Orthogonal spline collocation Laplace-modified and alternating-direction methods for parabolic problems on rectangles. *Mathematics of Computation*, 60(202):545–573, April 1993.

[BY84]Y.A. Berezin and N.N. Yanenko. The splitting method for the problem in physics of semiconductors. *Dokl.Acad.Sci.USSR*, 274(6), 1984. in Russian.

[Che94]A. Cheng. Optimal error estimate of finite element method for a model for miscible compressible displacement in porous media. *Numer.Math.J.Chinese Univ.*, 16(2):134–144, June 1994.

[FF91]R.I. Fernandes and G. Fairweather. An alternating direction Galerkin method for a class of second-order hyperbolic equations in two space variables. *Siam J.Numer.Anal.*, 28(5):1265–1281, October 1991.

[Hay81]L.J. Hayes. Galerkin alternating-direction methods for nonrectangular regions using patch approximations. *Siam J. Numer. Anal.*, 18:627–643, 1981.

[HK84]L.J. Hayes and S.V. Krishnamachari. Alternating-direction along flow lines in a fluid flow problem. *Comput. Methods Appl.Mech.Engrg.*, 47:187–203, 1984.

[JD71]J. Douglas Jr. and T. Dupont. Alternating-direction Galerkin methods on rectangles. In B. Hubbard, editor, *Proceedings Symposium on Numerical Solution of Partial Differential Equations, II.*, pages 133–164, New York, 1971. Academic Press.

[JDEW83]Jr. J. Douglas, R.E. Ewing, and M.F. Wheeler. The approximation of the pressure by a mixed method in the simulation of miscible displacement. *R.A.I.R.O Numerical Analysis*, 17(1):17–33, 1983.

[JR83]Jr. J.Douglas and J.E. Roberts. Numerical methods for a model for compressible miscible displacement in porous media. *Mathematics of Computation*, 41(164):441–459, October 1983.

[Mar90]G.I. Marchuk. *Handbook of Numerical Analysis, Splitting and alternating direction methods*, volume I. Elsevier Science Publishers B.V., North-Holland, Amsterdam, 1990.

[MPT83]O. Migual, P. Pinsky, and R. Taylor. Operator split methods for the numerical solution of the elastoplastic dynamic problem. *Comput. Methods Appl.Engrg.*, 39:137–157, 1983.

[Pea66]D.W. Peaceman. Improved treatment of dispersion in numerical calculation of multidimensional miscible displacement. *Soc.Pet.Eng.J.*, pages 213–216, 1966.

[SHR89]S.V.Krishnamchari, L.J. Hayes, and T.F. Russell. A finite element alternating-direction method combined with a modified method of characteristics for convection-diffusion problems. *Siam J. Numer. Anal.*, 26(6):1462–1473, December 1989.

[Yua92]Y. Yuan. Time stepping along characteristics for the finite element approximation of compressible miscible displacement in porous media. *Mathematica Numerica sinica*, 14(4):385–400, November 1992.

Part III

Applications

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau,

27. Minimum Overhead Data Partitioning Algorithms for Parallel Video Processing

D T Altılar[1], Y Paker[2]

Introduction

Data partitioning is important in many aspects, such as computational, load distribution, inter-process communication, load and data overhead considering different applications. In this paper, overhead due to data partitioning is discussed and two algorithms are proposed: Almost Square Tiles (AST) and Almost Square Tiles with aspect ratio (ASTwar). We exploit data parallelism, which is suitable for both SPMD and SIMD type of parallel computing.

The applications are selected from image/video processing arena most of which involve some neighbourhood operations that require surrounding pixels such as convolution or motion estimation. However, this never excludes the applicability of these algorithms to any other parallel applications, including linear or differential equation solvers. Both AST and ASTwar are to minimise the amount of overlapped data by defining a partition pattern that comprises rectangular tiles of similar sizes and having an aspect ratio of around 1.

A detailed explanation of the problem is introduced in Section "Background and Problem". "Approaches to Data Partitioning Problem" provides the reader a brief information about a recently proposed approach by Lee and Hamdi [LH95]. The proposed algorithms are defined in detail, and a brief comparison between the algorithms is given in "Two Proposed Algorithms:AST and ASTwar" Section. The paper concludes with a Section suggesting further research.

Background and the Problem

There are number of research areas in which data partitioning occupies an important role, such as instruction level data parallelism [AAL95], graph partitioning [KQR95], image processing for image space (2-D) or object space (3-D) [LH95, Whi92, LWY94, CQ95]. An extended survey on I/O intensive parallel computing is given in [Bre97] emphasising language support. As mentioned before, we take image/video processing domain to illustrate the partitioning ideas developed. Moreover, data partitioning is a very important issue in real-time video processing because any defined task should terminate within 40ms and acquire new data periodically.

Video processing algorithms we are interested in require neighbourhood pixels or blocks to be transmitted as shown in Figure 1. The original image is initially split into rectangles of size $a * b$. However, for the given application which includes a neighbourhood operation, rectangle is expanded in size by n in both directions.

[1] Department of Computer Science, Queen Mary, University of London, altilar@dcs.qmw.qc.uk
[2] Department of Computer Science, Queen Mary, University of London, paker@dcs.qmw.qc.uk

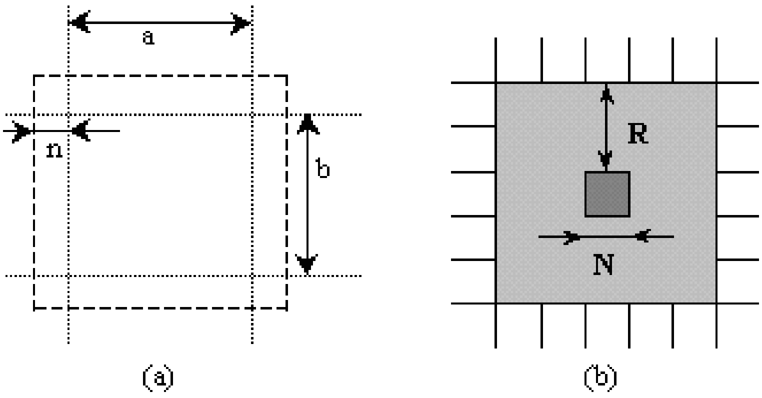


Figure 1: (a): Size expansion of a sub-image of $a * b$ due to the neighbourhood pixels, (b): A core block of *N*N* and search area of (2R+N)*(2R+N)

Figure 1a shows a sub-image of a by b pixels and the required n pixels size expansion. If we are to compute a convolution algorithm, the overall size of the sub-image becomes $(a + 2n)(b + 2n)$ into an associated coefficient matrix of *2n+1* by *2n+1*. The difference in size in pixels between these two sub-images is given in Eq. 1

$$((a + 2n)(b + 2n)) - (ab) = 2n(a + b) + 4n^2 \tag{1}$$

As another application, consider motion estimation, which is the most compute intensive part of MPEG video compression: a core block (called "macro block" in MPEG terminology) of N by N from the current frame to be matched with neighbouring blocks of previous frame (Figure 1b) which is a domain of $(2R + N)(2R + N)$ centred on the macro block. Considering the above given example, overhead data becomes significant as R could be up to 16. Thus, comparing with the previous application n could be up to 16 times bigger than a (or b) for this particular application.

When neighbourhood pixels are taken into account, different partition patterns yield different amount of additional data, i.e. data overhead, to be transferred giving rise to a minimisation problem.

Approaches to Data Partitioning Problem

In a recent article [LH95], Lee and Hamdi explain the experimental results of parallel image processing applications on a network of workstations. They exploited image parallelism on a client-server based application model, which they call Host-Node Model. The host splits the image and dispatches to a number of workstations to perform convolution. It is also responsible to collect the distributed sub-images. They consider a one-to-one communication between the host and the other nodes. Nodes are not allowed to communicate among themselves. Above given assumptions on system fit into our model as well.

One of the main concerns they stated in the paper is the impact of the overhead of neighbourhood pixels on the processing time. They proposed a heuristic method for data partitioning which comprises four steps: assuming that t is the number of sub-images (tiles) that the image will split into;

1. If t=1 then fetch another sub-image of whose t¿1,
   if there is no such a sub-image left then terminate.
2. If t is even, divide image into sub-images A and B,
   equally (with the ratio of 1:1)
   horizontally or vertically by keeping overlap minimum.
3. If t is odd, divide image into two sub-images A and B
   with the ratio of (t/2):(t/2)+1,
   horizontally or vertically by keeping overlap minimum.
4. Go to the first step for both sub-image A and sub-image B.

They compared their heuristic method with three standard partitioning methods: cross, column-wise, and row-wise. They indicated that the heuristic method is better than row (or column) partition method but not so good as cross partition. This heuristic method is a divide and conquer type of approach which could lead to undesired partition especially because of the third step of the partition algorithm.

The Core of the Proposed Approach

In order to find a better way of partitioning, we believe the decision should be made considering the original size of the image instead of dividing it into partitions recursively as in the divide and conquer type of approach.

Eq. 1 defines the overhead. If n is a constant as number of partitions, one needs to minimise $a+b$ to minimise the data overhead for $C = a*b$. Since C, load per partition, can be computed for a given n, one can define a generic minimisation problem for the issue: *For a given C, C=a*b , find Min(a+b)* . This is a well known minimisation problem having a solution of

$$a = b = \sqrt{C} \tag{2}$$

Eq. 2 shows that the minimum is achieved for $a = b$, i.e., for a square. In other words, square is the optimal shape for a constant area and minimum circumference. However, it is not always possible to divide a given image into squares of size k for any given number of partitions. Actually it is unlikely to have such a perfect partition except for a few special cases. The partition would comprise a mixture of squares and rectangles of different width and height. For achieving an acceptable solutions the height-width ratio of the rectangles should be close to one.

Two Proposed Algorithms: AST and ASTwar

To solve the above problem, two heuristic algorithms, Almost Square Tiles Data Partitioning Algorithm (AST) and Almost Square Tiles Data Partitioning Algorithm with aspect ratio (ASTwar), have been developed. Let k, the square of an integer, be the

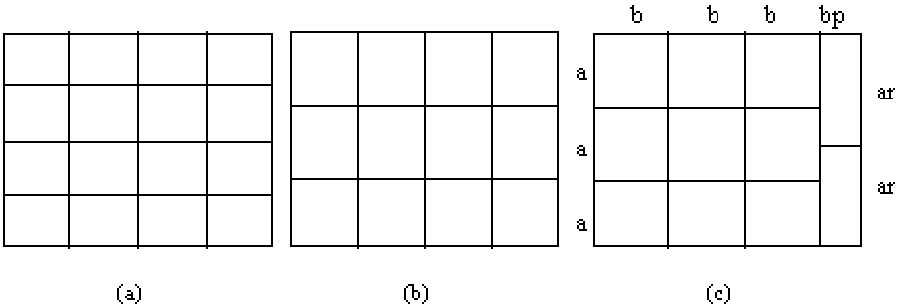


Figure 2: Internal steps of splitting in image of 576*720 into 11 partitions: a=192, ar=288, b=196, and bp=132 pixels.

least number which is greater than or equal to the number of data partitions p to be produced. The frame is split into k tiles with the concern that the height-width ratio of the rectangles should be close to one as much as possible. By changing the width and the height of sub-images afterwards, a partition producing minimum overhead is produced.

Both of the algorithms start by splitting the frame into $k = n^2$, n an integer providing that k is the smallest number greater than or equal to p. There is the possibility of reducing the number of rows (or columns with respect to the aspect ratio) by one for some cases which satisfy $n(n-1) > p$. The algorithm than proceeds to reduce the number of tiles by changing the size of the tiles column-wise.

For example, the above explained steps are shown in Figure 2, for 11 partitions: (a) The frame is split into 16 (4*4) initially although 11 is required, (b) For this particular case, the number of rows is reduced by one since $4(4-1) > 11$, (c) Column-wise changing on the width and reducing the number of tiles to 11 is the latter step of the overall algorithm. This third step comprises computing of height of the rows, i.e., the *a family* consists of *a, ap, ar* and *arp*, and computing width of the columns, i.e, the *b family* consists of *b,bp,* and *bpp* (Figure 3).

Computing the *a family* values is quite simple as number of rows for regular columns are known and number of rows for irregular columns is one less than regulars. *ap* and *arp* are the last tile heights (residues) of the regular and irregular columns respectively. For data balancing the area of tiles should be almost the same, i.e, $a * b = ar * bp$. On the other hand, $width = b * reg\_cols + bp * irr\_cols$. The solution of these two equations gives the value for b and bp.

Almost Square Tiles Data Partitioning Algorithm

In the AST algorithm it is assumed that the width of the frame is equal to or larger than its height. The flow of the developed algorithm can be summarised as follows:

```
 1)  k ← least_greater_or_equal_square(partitions)
 2)  first_square ← squareroot(k)                                       (A)
 3)  cols ← first_square
 ----------------------------------------------------------------------------
 4)  if (partitions is a square of an integer) rows ← first_square      (B)
     else if ((rows-1)*cols   partitions) rows ← first_square -1
 ----------------------------------------------------------------------------
 5)  irr_col ← cols * rows - partitions                                 (C)
 ----------------------------------------------------------------------------
 6)  a ← image_height/rows
 7)  ap ← image_height - a * (rows -1)
 8)  ar ← image_height/num_rows - 1                                     (D)
 9)  arp ← image_height - ar * (rows - 2)
 ----------------------------------------------------------------------------
10)  b ← (image_width/((ar/a) * (cols-irr_cols) + irr_cols)) * (ar/a)
11)  bp ← ( image_width - b * (cols-irr_cols) ) / irr_cols              (E)
12)  bpp ← image_width - b * (cols-irr_cols) - bp * (irr_cols - 1 )
```

Algorithm could be thought in five functional blocks from A to E. Lines 1 and 2 are to determine the maximum number of columns and rows. The number of tiles is assumed to be a square of an integer. If the number of partitions, *partitions*, is a square number, the number of columns, *columns*, and the number of rows, *rows*, would be the same. Set the number of columns for every row(Line 3). By the end of Block A, the number of column which equals the number of rows is known.

Line 4 is to search for the possibility of dividing the data into less rows than the current value of *rows.* If /em partitions is not a square number then there is such a possibility as shown in Figure 2. The final value of the *rows* is set while terminating Block B.

Block C (Line 5)is to compute the number of irregular columns which is one less than the regular ones.

Block D comprises lines to compute the values of the *a family*, i.e., *a,ap,ar,* and *apr.* *a* and *ar* are tile height for regular and irregular columns respectively where *ap* and *arp* are the last tile heights (residues) of the regular and irregular columns respectively. Computing values for the *a family* members is quite simple as *image_height* is known, the number of rows for regular columns is computed in previous blocks, and the number of rows for irregular columns is one less than regulars. The height of a regular tile, a, can be computed by dividing the image height by the number of the rows (Line 6). Line 7 is to check out whether there is a residue row having different height, *ap.* If there is an irregular column, there will be a repeating tile height as well, which is *ar* (Line 8). There might be a residue row having different height than *ar*, which is *arp* (Line 9).

The *b family* members are computed through Block E. Line 10 possesses the solution of two equations to numerate *b* and *bp.* In order to make the areas of most of the tiles equal: *a*b=bp*ar.* Since image width should be covered by columns: $width = cols * b + irr\_cols * bp$. As *a, ar, cols,* and *irr_cols* are computed previously *b*, in Line 10, and *bp*, in Line11 can be numerated. Block E ends with checking out for size of the residue column.

Thus, one could produce at most six different types of tiles through the given algorithm. Tile type names and sizes are (Figure 3):

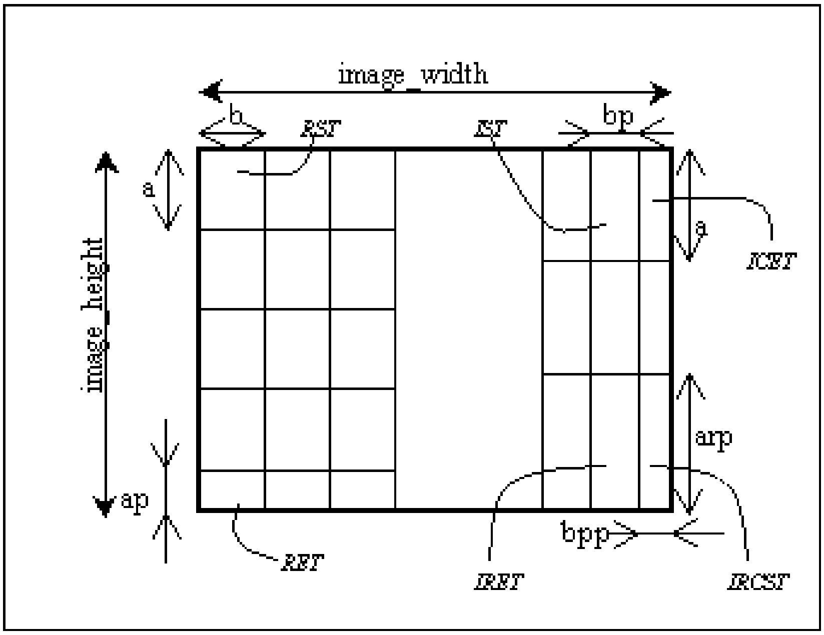


Figure 3: All possible tiles and sizes of tiles to be produced by the proposed algorithms

RST - regular standard ones $(a * b)$,
RET - regular excess ones $(ap * b)$,
IST - irregular standard one $(bp * ar)$,
ICET - irregular column excess ones $(bpp * ar)$,
IRET - irregular row excess ones $(bp * arp)$,
IRCET - irregular double excess one $(bpp * arp)$;

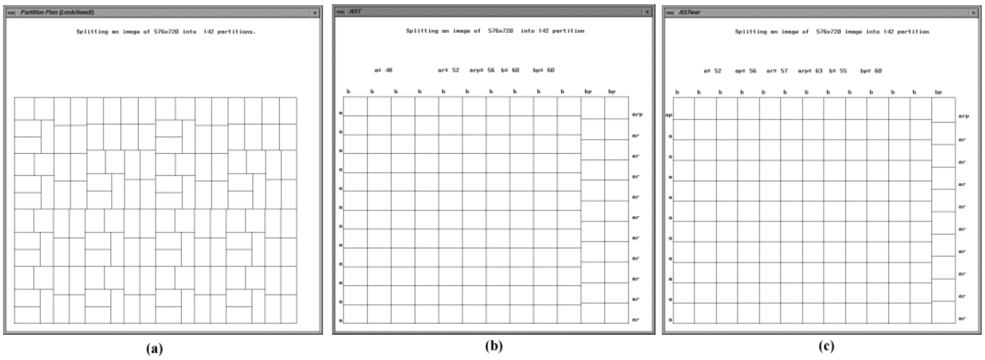
where

a is the standard tile height, ap is the height of the last standard tile in a regular column, ar is the height of irregular column tiles, arp is the height of the last tile in an irregular column, b is the width of the tile of standard regular column, bp is the width of the tiles of irregular columns, bpp is the width of the tiles of the last irregular column.

Almost Square Tiles with aspect ratio Data Partitioning Algorithm

The aspect ratio of the image is taken into account in the ASTwar algorithm. Therefore instead of dividing image into the same number of columns and rows initially, considering the aspect ratio, an image can be divided into different numbers of columns and rows ensuring that widths and heights of rectangles should be as close as possible.

The ASTwar requires the overall ratio for the image. The aspect ratio of the image



Partition Plan (Lee&Hamdi)

Splitting an image of 576x720 into 142 partitions.

AST

Splitting an image of 576x720 into 142 partition

a= 48 ar= 52 arp= 56 b= 60 bp= 60

ASTwar

Splitting an image of 576x720 image into 142 partition

a= 52 ap= 56 ar= 57 arp= 63 b= 55 bp= 60

Figure 4: Partition patterns for 142 tile: (a)Lee-Hamdi, (b)ATS, (c)ATSwar

is multiplied by the ratio of the number of rows to columns to compute the *overall ratio*: $overall\_ratio = aspect\_ratio\_of\_image * (num\_rows/num\_cols)$

In the ASTwar algorithm *overall_ratio* to be close to 1 where in the AST number of columns is equal to the number of rows as the image ratio is expected as one (or close to one) implicitly. Thus, the ASTwar algorithm is the same as the AST except the block (A) which comprises a loop to set a value for the *overall ratio* as close as possible to 1.

Comparison of the Algorithms

Both AST and ASTwar data partitioning algorithms provide better solution than the one suggested in Lee and Hamdi. The actual values of data overhead for a neighbourhood of 16 pixels are given in Table 1. Even for a neighbourhood of 16 pixels, two proposed algorithms reduce I/0 data amount by upto 10%. Obviously more significant reductions are available for larger values of *n*.

	Lee-Hamdi	AST	ASTwar			
Partitions	(A)	(B)	(C)	(A)-(B)	(A-C)	(B-C)
12	78336	74496	74496	3840	3840	0
24	112128	107904	107520	4224	4608	384
110	249792	237200	244544	12592	5248	-7344
130	275072	268672	259440	6400	15632	9232
142	290816	283264	282336	7552	8480	928

Table 1: Actual data overhead in pixels for an image of 576x720 requiring neighbouring pixels of 16.

Partition patterns for 142 partitions are drawn in Figure 4. One should pay attention to the irregularity of shapes in Figure 4a and regularity in Figure 4b and Figure 4c.

Conclusion and Further Research

Two new algorithms, AST and ASTwar, have been introduced to reduce this overhead for data transmission for parallel algorithms requiring neighbourhood pixels. They are both based on the concept that the more tiles are close to squares the less data overhead is to be introduced. Therefore, a global data partition pattern creation, keeping every rectangles height and width as close as possible is the basic approach lying under the two algorithms. ASTwar is slightly different from the first one as it takes the image aspect ratio into account as well. The partition patterns and numerical analysis have shown that the ASTwar algorithm has better performance than the AST algorithm. All of the algorithms are currently being tested for images of different aspect ratios for image/video processing area. These two algorithms for optimal data partitioning are also applicable to other types of parallel applications since optimisation is on overlapped (shared) data. Applying these two algorithms is for parallel numerical solution of partial differential equations is in progress.

References

[AAL95]M. J. Anderson, S. P. Amarasighe, and M. S. Lam. Data and computation transformations for multi-processors. In *Proceedings of the Fifth ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming (PpoPP'95) Santa Barbara CA*, July 19-21 1995.

[Bre97]P. Brezany. *Input/Output Intensive Parallel Computing.* Lecture Notes in Computer Science Series 1220. Springer-Verlag, 1997.

[CQ95]P. E. Crandall and M. J. Quinn. A partitioning advisory system for networked data-parallel processing. *Concurrency: Practice and Experience*, 7(5):479–495, 1995.

[KQR95]M. Kaddoura, C. W. Qu, and S. Ranka. Partitioning unstructured graphs for non-uniform and adaptive environments. *IEEE Parallel and Distributed Technology*, 3, 1995.

[LH95]C. Lee and M. Hamdi. Parallel image processing applications on a network of workstations. *Parallel Computing*, 21:137–160, 1995.

[LWY94]C. Lee, Y. F. Wang, and T. Yang. Static global scheduling for optimal computer vision and image processing operations on distributed memory processors. Technical Report TRC94-23, Dept. of Computer Science, Santa Barbara, CA, December 1994.

[Whi92]S. Whitman. *Multiprocessor Methods for Computer Graphics Rendering.* Jones and Bartlett publishers, Boston, MA., 1992.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau,

28. The Mortar Element Method for 3D Maxwell's equations: analysis and application to magnetodynamics

A. Buffa[1] , Y. Maday[2 3] , F. Rapetti[3]

Introduction

In this paper, we describe the main ideas of the mortar element method combined with $H(\mathbf{curl})$–conforming finite elements for the numerical approximation of Maxwell's equations. This method turns out to be a new non–conforming, non–overlapping domain decomposition technique where non–matching grids are allowed at the interface between adjacent sub–domains. We report the results on the method's convergence and error estimate together with the description of the main implementation details and some numerical results.

Position of the problem

We are interested in a system that can be modeled by the set of Maxwell's equations when the displacement currents are neglected:

$$\begin{array}{lll} (a) & \mathbf{curl}\,\mathbf{E} = -\partial_t \mathbf{B} & \text{in } \Omega\times]0,T[\\ (b) & \mathbf{curl}\,\mathbf{H} = \mathbf{J} & \text{in } \Omega\times]0,T[\\ (c) & \mathbf{J} = \sigma \mathbf{E} & \text{in } \Omega\times]0,T[\\ (d) & \mathbf{B} = \mu\, \mathbf{H} & \text{in } \Omega\times]0,T[\end{array} \qquad (1)$$

where $\Omega \subset \mathbb{R}^3$ is bounded, $\mathbf{E}$, $\mathbf{H}$ are the electric and magnetic fields, $\mathbf{B}$ the magnetic induction and $\mathbf{J}$ the current density. In system (1), ∂_t stands for the first derivative in time.

The physical parameters of the problem are: the magnetic permeability μ and the electric conductivity σ. Without loss of generality we assume that μ is a positive constant and σ is simply bounded. We set $\mathcal{C} = supp\{\sigma\}$ the conducting part and we assume that $\mathcal{C}$ is simply connected.

In three–dimensional magnetodynamic applications, system (1) is usually reformulated in terms of a primary variable which is either the magnetic field $\mathbf{H}$ or the magnetic vector potential $\mathbf{A}$. In both cases, by re–writing system (1) in terms of the chosen primary variable, we obtain the following parabolic equation which is the object of our study:

$$\partial_t(\alpha\, \mathbf{u}) + \mathbf{curl}\,(\beta\, \mathbf{curl}\,\mathbf{u}) = \mathbf{f} \qquad \text{in } \Omega\times]0,T[\,, \qquad (2)$$

[1] Università degli Studi di Pavia, Dip. di Matematica, annalisa@ian.pv.cnr.it

[2] Laboratoire d'Analyse Numérique BC187, Université Pierre et Marie Curie, maday@ann.jussieu.fr

[3] ASCI–UPR 9029 CNRS, Université Paris Sud, rapetti@asci.fr

with α and β two functions related to the physical parameters σ and μ and $\mathbf{f}$ related to the external sources. Equation (2) can be non–strictly parabolic: this may occur when working with the magnetic vector potential. In this case, a jauge condition (e.g., $\mathrm{div}(\mathbf{u}) = 0$ where $\alpha = 0$) has to be added to equation (2) to ensure the uniqueness of the solution. Moreover, we assume:

$$\mathbf{u} \wedge \mathbf{n} = \mathbf{0} \ \text{ on } \ \partial\Omega \times]0, T[\qquad \text{and} \qquad \mathbf{u}(\mathbf{x}, 0) = 0 \ \text{ a.e. in } \ \mathbf{x} \in \overline{\mathcal{C}}. \tag{3}$$

We introduce the following Hilbert spaces (endowed with the corresponding graph norms)

$$\begin{aligned} H(\mathbf{curl}, \Omega) &= \left\{ \mathbf{u} \in L^2(\Omega)^3 \mid \mathbf{curl}\,\mathbf{u} \in L^2(\Omega)^3 \right\}, \\ H_0(\mathbf{curl}, \Omega) &= \left\{ \mathbf{u} \in H(\mathbf{curl}, \Omega) \mid (\mathbf{u} \wedge \mathbf{n})_{|\partial\Omega} = \mathbf{0} \right\}. \end{aligned} \tag{4}$$

The variational formulation of the problem (2) reads:

$$\begin{gathered} \text{Find } \mathbf{u} \in H_0(\mathbf{curl}, \Omega) \text{ such that } \forall \mathbf{v} \in H_0(\mathbf{curl}, \Omega) : \\ \textstyle\int_\Omega \partial_t \alpha \ \mathbf{u} \cdot \mathbf{v} \ d\mathbf{x} \ + \int_\Omega \beta \ \mathbf{curl}\,\mathbf{u} \cdot \mathbf{curl}\,\mathbf{v} \ d\mathbf{x} \ = \int_\Omega \mathbf{f} \cdot \mathbf{v} \ d\mathbf{x} \,. \end{gathered} \tag{5}$$

It can be proved that this problem admits a unique solution when suitably interpreted in time and a jauge condition is imposed where $\alpha = 0$. Note that when $\alpha = 0$ everywhere, (2) is the magnetostatic problem and (5) its variational formulation.

The main concern of our work is to propose an efficient domain decomposition method for this type of equations, discretized by using edge element approximation in three dimensions which allows for non-matching grids. The outline of the paper is the following: in the second section the mortar element method is proposed, the analysis is sketched and some details of the implementation are given. The third section is devoted to the applications: we present some preliminary numerical results in the magnetostatic case and the governing equations for the magnetodynamic problem in moving geometries. Numerical simulations in the latter case are in progress.

Definition and analysis of the mortar element method

Since the definition and analysis of a domain decomposition procedure for (2) is strictly related to the choice of the spatial discretization, in this section, without loss of generality, we consider the following model problem:

$$\begin{gathered} \text{Find } \mathbf{u} \in \ H_0(\mathbf{curl}, \Omega) \text{ such that } \forall \ \mathbf{v} \in \ H_0(\mathbf{curl}, \Omega) \\ \int_\Omega \mathbf{curl}\,\mathbf{u} \cdot \mathbf{curl}\,\mathbf{v} \ d\mathbf{x} \ + \int_\Omega \mathbf{u} \cdot \mathbf{v} \ d\mathbf{x} \ = \int_\Omega \mathbf{f} \cdot \mathbf{v} \ d\mathbf{x}. \end{gathered} \tag{6}$$

This problem admits obviously a unique solution in $H_0(\mathbf{curl}, \Omega)$ and it is worth noting that it is strictly related to (2): when the parameters of the problem are set equal to 1 and an implicit time stepping procedure is applied, (6) is the problem that we have to solve at each time step. The case of vanishing α will be the object of further remarks.

Approximation spaces

Partition of the domain and local spaces

Assume here that the domain Ω is a convex bounded (Lipschitz) polyhedral[4] subset of $\mathbb{R}^3$. Let $\Omega_k \subseteq \Omega$, for $1 \leq k \leq K$, be a non–overlapping, polyhedral partition of Ω, that is:

$$\overline{\Omega} = \cup_{k=1}^{K} \overline{\Omega}_k \text{ with } \Omega_k \cap \Omega_l = \emptyset \text{ if } k \neq l. \tag{7}$$

For every k ($1 \leq k \leq K$) we denote by $\mathbf{n}_k$ the outer normal to Ω_k and we call $\{\Gamma^{k,i}\}_{1\leq i\leq F(k)}$ the $F(k)$ faces of the polyhedron Ω_k. We define the **skeleton** Σ as $\Sigma = \cup_{k=1}^{K} \cup_{l=1}^{K} \partial\Omega_k \setminus \partial\Omega$. Let $\tau^{k,i}$ be the counterclockwise tangential vector to $\partial\Gamma^{k,i}$; we define also the outer normal to $\partial\Gamma^{k,i}$ as $\mathbf{n}^{k,i} = \tau^{k,i} \wedge \mathbf{n}_k$.

Among several possibilities we choose a splitting of the skeleton Σ as the disjoint union of some closed faces $\{\overline{\Gamma}^{k,i}\}_{k,i}$ that we call *slave faces*. A unique set of indices corresponds to this choice and we denote it by:

$$\mathcal{I}_M = \{m = (k,i) \text{ such that } \Gamma^{k,i} \text{ is a slave face } \}.$$

To shorten the notations we denote the slave faces by Γ^m ($1 \leq m \leq M$) and, as prescribed, we have: $\overline{\Sigma} \equiv \cup_{m=1}^{M} \overline{\Gamma}^m$ and $\Gamma^m \cap \Gamma^n = \emptyset$ if $m \neq n$. Moreover we set: $\overline{\gamma}^{l,k} = \partial\Omega_l \cap \partial\Omega_k$. We define the following "broken" space:

$$X := \{\mathbf{u} \in L^2(\Omega)^3 \mid \mathbf{u}_{|\Omega_k} \in H(\mathbf{curl}, \Omega_k) \ , \ (\mathbf{u} \wedge \mathbf{n})_{|\,\partial\Omega\cap\partial\Omega_k} = \mathbf{0} \ 1 \leq k \leq K \ \}. \tag{8}$$

As standard, X is a Hilbert space endowed with the following broken norm: $||\mathbf{u}||^2_{\star,\mathbf{curl}} := \sum_{k=1}^{K} ||\mathbf{u}_{|\Omega_k}||^2_{\mathbf{curl},\Omega_k}$.

For each index k ($1 \leq k \leq K$), we introduce a regular quasi–uniform triangulation $\mathcal{T}^k_{h(k)}$ on the sub–domain Ω_k and we denote by h the maximum of the mesh sizes. These partitions can be composed either of tetrahedra or parallelepipeds; they are completely independent and thus, in general, non–matching at the interfaces $\{\gamma^{k,l}\}_{k,l}$.

Let $\hat{K}$ be the reference tetrahedron or cube. For every $K_i \in \mathcal{T}^k_{h(k)}$, we denote by $\mathcal{F}_i : \hat{K} \to K_i$ a bijective mapping from $\hat{K}$ to K_i. These mappings can be chosen as *linear* both in the case of tetrahedra and parallelepipeds: $\mathcal{F}_i(\hat{\mathbf{x}}) = \mathcal{B}_i\hat{\mathbf{x}} + \mathbf{c}_i$ where $\mathcal{B}_i \in \mathbb{R}^{3\times3}$ is an invertible matrix and $\mathbf{c}_i \in \mathbb{R}^3$ is a constant field. Over each sub–domain Ω_k we define the finite dimensional space which is at the base of the domain decomposition method:

$$Y^k_h := \{\mathbf{v}^k_h \in H(\mathbf{curl}, \Omega_k) \mid \mathcal{B}^T_i(\mathbf{v}^k_{h_{|K_i}} \circ \mathcal{F}_i) \in \mathcal{P}_{p(k)} \ \forall K_i \in \mathcal{T}^k_{h(k)}\}, \tag{9}$$

where $\mathcal{P}_{p(k)}$ denotes a family of Nédélec type finite elements for Maxwell's equations of degree $p(k)$ (see [N\'80, N\'86] for a complete definition). Furthermore we set:

$$X_h \ := \ \{\mathbf{v}_h \in X \mid \mathbf{v}^k_h := \mathbf{v}_{h\,|\,\Omega_k} \in Y^k_h \text{ and } (\mathbf{v}^k_h \wedge \mathbf{n})_{\partial\Omega\setminus\partial\Omega_k} = \mathbf{0}\}, \tag{10}$$

and in the following we denote the elements $\mathbf{v}_h \in X_h$ both as functions and as K-uplets $\mathbf{v}_h = (\mathbf{v}^1_h, \mathbf{v}^2_h, \cdots, \mathbf{v}^K_h)$ where $\mathbf{v}^k_h \in Y^k_h$ ($1 \leq k \leq K$). We use both notations

[4] The whole theory applies even when Ω is a regular bounded subset of $\mathbb{R}^3$. Of course, in this case the subdomain $\{\Omega_k\}_k$, defined afterwards, can not be polyhedra but curved polyhedra.

since it is never misleading. Since we deal with tangential traces of these vector fields, we introduce some further definitions. For any index $m = (k,i) \in \mathcal{I}_M$, we set $T_h^{k,i} = \{(\mathbf{v}_h^k \wedge \mathbf{n}_k)_{|\Gamma^{k,i}} \mid \mathbf{v}_h^k \in Y_h^k\}$ and its subset $T_{h,0}^{k,i} = \{\lambda_h \in T_h^{k,i} \mid (\lambda_h \cdot \mathbf{n}^{k,i})_{|\partial\Gamma^{k,i}} = 0\}$.

Let Γ^m be a *slave face* with $m = (k,i)$ the corresponding indices and $\mathbf{v}_h \in X_h$: for almost every $\mathbf{x} \in \Gamma^m$ there exists an l $(1 \le l \le K\ ,\ l \ne k)$, such that $\mathbf{x} \in \Gamma^m \cap \gamma^{k,l}$. At this point $\mathbf{x}$, we have two fields, namely $\mathbf{v}_h^k$ and $\mathbf{v}_h^l$. In general, since the macro–decomposition is non–conforming, the value of l depends on the point $\mathbf{x}$ and we denote by I_m the set of indices l $(1 \le l \le K\ ,\ l \ne k)$ such that $\Gamma^m \cap \gamma^{k,l} \ne \emptyset$. We then set $\mathbf{v}_h^{-k}(\mathbf{x}) = \mathbf{v}_h^l(\mathbf{x}), \quad \forall \mathbf{x} \in \Gamma^m \cap \gamma^{k,l}, \quad l \in I_m$. The function $\mathbf{v}_h^{-k}(\mathbf{x})$ is defined at almost every $\mathbf{x} \in \Gamma^m$. Due to the non–conformity of the macro–decomposition, $\mathbf{v}_h^{-k}(\mathbf{x})$ is not in general the tangential trace at Γ^m of a field $\mathbf{v}$ in $H(\mathbf{curl}, \Omega_k)$.

Constraint problem and matching condition

Let $\mathbf{v} \in H_0(\mathbf{curl}, \Omega)$, we have that $(\mathbf{v}^{-k} \wedge \mathbf{n})_{|\Gamma^m} = (\mathbf{v}^k \wedge \mathbf{n})_{|\Gamma^m}$ in $\left(H_{00}^{1/2}(\Gamma^m)\right)'$. The purpose of this section is to express how to impose this condition on the discrete broken space in a weak sense. To this aim, we define, for any $m \in I_m$, $M_h^m \subseteq T_h^m, \quad \dim\{M_h^m\} = \dim\{T_{h,0}^m\}$. We set:

$$M_h := \{\psi_h \in L^2(\Sigma)^2 \mid \forall m \in I_m\ ,\ \psi_{h\,|\Gamma^m} \in M_h^m\}. \tag{11}$$

As before we also adopt the vector notation $\psi_h = (\psi_h^1, \psi_h^2, \ldots, \psi_h^M)$ when it is convenient. Then, we propose the following non–conforming approximation space for $H_0(\mathbf{curl}, \Omega)$:

$$X_h^c = \{\mathbf{v} \in X_h \mid \forall m\ ,\ \int_{\Gamma^m} (\mathbf{v}_h^k \wedge \mathbf{n}_k - \mathbf{v}_h^{-k} \wedge \mathbf{n}_k) \cdot \psi_h \, d\Gamma = 0\ \forall \psi_h \in M_h^m\}. \tag{12}$$

The discrete problem reads: find $\mathbf{u}_h \in X_h^c$ such that $\forall\ \mathbf{v}_h \in\ X_h^c$:

$$\sum_{k=1}^{K} \left[\int_{\Omega_k} \mathbf{curl}\,\mathbf{u}_h^k \cdot \mathbf{curl}\,\mathbf{v}_h^k \, d\mathbf{x}\ + \int_{\Omega_k} \mathbf{u}_h^k \cdot \mathbf{v}_h^k \, d\mathbf{x} \right] = \int_\Omega \mathbf{f} \cdot \mathbf{v}_h \, d\mathbf{x} \qquad \forall\ \mathbf{v}_h \in\ X_h^c. \tag{13}$$

The numerical properties of the space X_h^c depend strongly on the choice of the Lagrange multiplier space M_h. In the following, we discuss our choice for this space and, we proceed to the analysis of the method. We refer to [Hop99] for a different approach using the first family of edge elements.

Thanks to the locality in the definition (11) of M_h, we focus our attention on a single slave face Γ^m. Γ^m is decomposed by $\mathcal{T}_{h(k)|\Gamma^m}^k$ either into triangles or parallelograms. For every parallelogram (resp. triangle) of $\mathcal{T}_{h_{(k)}|\Gamma^m}^k$, there exists a linear mapping F_i satisfying $T_i = F_i(\hat{T})$ where $\hat{T}$ is the reference square $]-1,1[^2$ (resp. the reference triangle $\hat{T} := \{(x,y) \in \mathbb{R}^2 \mid 0 < x < 1\ ,\ 0 < y < 1 - x\}$). The construction of M_h^m consists in imposing additional constraints at T_h^m on the parallelograms (resp. triangles) which meet the boundary $\partial\Gamma^m$. We denote by BT^m the set of all these elements T_i and assume that the mapping F_i associates to (one of) the boundary edge(s)

$(\overline{T}_i \cap \partial\Gamma^m)$ an edge of $\hat{T}$ that is parallel to a Cartesian axis (this is exhaustive up to a rotation). Our choice of the Lagrange multiplier space turns out to be:
Case of parallelograms:

$$M_h^m := \left\{ \lambda_h^m \in T_h^m \mid B_i^{-1}(\lambda_h^m \circ F_i) \in \mathbb{Q}_{p,p}(\hat{S}) \times \mathbb{Q}_{p,p-1}(\hat{S}) \ , \ T_i \in BT^m \right\} \tag{14}$$

where $\mathbb{Q}_{p,p'}$ denotes the space of polynomials which are of degree p in the first variable and of degree p' in the second one. Of course, if a corner of Γ^m belongs to the parallelogram $\overline{T}_i$, then the Lagrange multiplier λ_h^m is chosen so that $B_i^{-1}(\lambda_h^m \circ F_i) \in \mathbb{Q}_{p-1,p}(\hat{S}) \times \mathbb{Q}_{p,p-1}(\hat{S})$.
Case of triangles:

$$M_h^m := \left\{ \lambda_h^m \in T_h^m \mid B_i^{-1}(\lambda_h^m \circ F_i) \in P_p(\hat{T}) \times P_{p-1}(\hat{T}) \ , \ T_i \in BT^m \right\}. \tag{15}$$

As before, if a corner of Γ^m belongs to the triangle $\overline{T}_i$, then the Lagrange multiplier λ_h^m is chosen so that $B_i^{-1}(\lambda_h^m \circ F_i) \in P_{p-1}(\hat{T}) \times P_{p-1}(\hat{T})$.

The spaces T_h^m and M_h^m are $H(\text{div})$–conforming and the degrees of freedom are related to the normal components of the fields along the edges. We refer to [BBM00] for a complete characterization.

The following proposition holds in both cases of triangles and parallelograms:

Proposition 1 *Let* $\Pi_h^m : L^2(\Gamma^m)^2 \to T_{h,0}^m$ *be defined by*

$$\int_{\Gamma^m} (\mathbf{u} - \Pi_h^m \mathbf{u}) \cdot \varphi_h \, d\Gamma \ = 0 \qquad \forall \varphi_h \in M_h^m. \tag{16}$$

There exists a constant C *independent of* h *such that the following stability estimate holds:*

$$\forall \mathbf{u} \in L^2(\Gamma^m)^2 \quad , \quad ||\Pi_h^m \, \mathbf{u}||_{0,\Gamma^m} \leq C \, ||\mathbf{u}\,||_{0,\Gamma^m}. \tag{17}$$

Remark 1 *If one deals with the first family of Nédélec type finite elements (see [N8́0]), then at the interface* Γ^m *the space* T_h^m *is of Raviart-Thomas type and the Lagrange multiplier space can be similarly defined (see [Hop99]).*

Convergence result

In this section we simply state the convergence results concerning problem (13) whose proofs can be found in [BBM00]. We have

Theorem 1 *Let* $\mathbf{u} \in H_0(\mathbf{curl}, \Omega)$ *be the solution of problem (6) and* $\mathbf{u}_h$ *the solution of problem (13) with* M_h *defined by (14) or (15). We assume that* $\mathbf{u}_k \in H^{p+1}(\Omega_k)$ *with* $\mathbf{curl}\,\mathbf{u}_k \in H^{p+1}(\Omega_k)$ $(1 \leq k \leq K)$ *and we suppose that there exists a uniform constant* γ *such that* $\max_k\{h_k\} \leq \gamma \, \min_k\{h_k\}$. *We set* $h := \max_k\{h_k\}$. *The following estimate holds:*

$$||\mathbf{u} - \mathbf{u}_h||_{\star,\Omega} \leq C_1 \, h^p \left(\sum_{k=1}^{K} ||\mathbf{u}||^2_{p+1,\Omega_k} \right)^{\frac{1}{2}} + C_2 \, h^p \, \sqrt{|\ln h|} \, \left(\sum_{k=1}^{K} ||\mathbf{curl}\,\mathbf{u}||^2_{p+1,\Omega_k} \right)^{\frac{1}{2}} \tag{18}$$

where $C_1, \ C_2$ *are uniform constants depending only on the macro-decomposition.*

Note that the first term comes from the best approximation error and the second one from the consistency error.

Remark 2 *The same error estimate holds when the coefficients are not set equal to one but they jump through the different sub–domains: the constant in front of the right hand side will depend on the size of their jumps.*

Remark 3 - ***Imposing a jauge condition*** **-** *When the parameter α is vanishing on a part of the domain, equation (6) must be replaced by:*

$$\int_\Omega \alpha \mathbf{u}\cdot\mathbf{v}\ d\mathbf{x}\ +\int_\Omega \beta\ \mathbf{curl}\,\mathbf{u}\cdot\mathbf{curl}\,\mathbf{v}\ d\mathbf{x}\ =\int_\Omega \mathbf{f}\cdot\mathbf{v}\ d\mathbf{x}\ ;\quad div\ (\mathbf{u})_{|\{\alpha=0\}}=0. \tag{19}$$

The definition of the proposed method in this case would involve a non–conforming mortar element approximation of the mixed problem related to (19) which is still not understood. Nevertheless, it is worth noticing that when the partition (7) is chosen in a way that $\alpha = 0$ in one sub–domain only, say $\Omega_{\bar{k}}$, the sub–domains are decomposed in polyhedra and none of the faces of $\Omega_{\bar{k}}$ is slave; then problem (19) can be suitably approximated. The discrete problem is: find $\mathbf{u}_h \in X_h^c$ such that $\forall \mathbf{v}_h \in X_h^c$ and $p_h \in \mathcal{S}^{p+1}(\mathcal{T}_h^{\bar{k}}, \Omega_{\bar{k}}) \cap H_0^1(\Omega_{\bar{k}})$:

$$\begin{aligned}\int_{\Omega\setminus\Omega_{\bar{k}}} \alpha\mathbf{u}_h\cdot\mathbf{v}_h\ d\mathbf{x}\ +\int_\Omega \beta\ \mathbf{curl}\,\mathbf{u}_h\cdot\mathbf{curl}\,\mathbf{v}_h\ d\mathbf{x} &= \int_\Omega \mathbf{f}\cdot\mathbf{v}_h\ d\mathbf{x} &(20)\\ \textit{and}\qquad \int_{\Omega_{\bar{k}}} \mathbf{u}_h\cdot \textit{grad}\, p_h\ d\mathbf{x} &= 0 &(21)\end{aligned}$$

where $\mathcal{S}^{p+1}(\mathcal{T}_h^{\bar{k}}, \Omega_{\bar{k}})$ is the standard scalar space of Lagrange finite elements of degree $p+1$. Making use of the approximation results proved in [ABDG98] and the ones of the previous section, it can be proved that (20)–(21) admits a unique solution and the error estimate (18) holds true when $\mathbf{u}$ is solution of (19) and $\mathbf{u}_h$ of (20)–(21).

On the other hand, when the quantity of interest is the magnetic vector potential, a unique solution can be selected by using a suitable iterative solver and expressing $\mathbf{J} = \mathbf{curl}\ \mathbf{T}$ *for a vector* $\mathbf{T}$*. Note that only the* **curl** *of the magnetic vector potential is needed: so, the magnetic induction is uniquely determined in any case.*

Reduction of the computational cost

The use of the second family of Nédélec type finite elements is often out of range in realistic three–dimensional computations and the use of the first family is often preferred. In the standard approximation context, with respect to the first one, the second family does not give a substantial improvement in the accuracy while it increases the number of degrees of freedom. In this section we show how these two families of edge elements can be merged in a way to obtain, on one hand, “quasi-optimal” convergence of the mortar element method and, on the other hand, a sensible reduction in the algebraic system dimension. We consider here the case where each sub–domain is discretized by a finite number of tetrahedra, first or second order edge elements are chosen and we will focus the attention on one slave face Γ^m.

First family: six degrees of freedom per tetrahedron – Given a tetrahedron K, let $\mathbf{r}_j$ (j=1,4) be the position vectors of its vertices and $\lambda_j(\mathbf{r})$ be the barycentric coordinate of a point $P \in K$ (with vector position $\mathbf{r}$) with respect to the vertex j. It is clear that $\lambda_j(\mathbf{r})$ is a linear function in the tetrahedron with $\lambda_j(\mathbf{r}_k) = \delta_{jk}$ $(j, k \in \{1, 2, 3, 4\})$. The vector basis function corresponding to an edge e_{ij} going from $\mathbf{r}_i$ to $\mathbf{r}_j$, is given by

$$\mathbf{w}_{ij}(\mathbf{r}) = \lambda_i(\mathbf{r}) \text{ grad } \lambda_j(\mathbf{r}) - \lambda_j(\mathbf{r}) \text{ grad } \lambda_i(\mathbf{r}) \quad , \quad i, j = 1, 2, 3, 4, \, i < j \, ; \tag{22}$$

let us denote by $\mathcal{P}_1(K)$ the space generated by the basis functions settled in (22). The interpolating function $\mathbf{u}_h$ on K for the vectorial state variable $\mathbf{u} \in (C^0(\overline{K}))^3$ has the following form

$$\mathbf{u}_h = \sum_{i=1}^{3} \sum_{j=i+1}^{4} \mathbf{w}_{ij} \, \alpha_{ij}(\mathbf{u}) \quad \text{with} \quad \alpha_{ij}(\mathbf{u}) = |e_{ij}| \, (\mathbf{u} \cdot \mathbf{t}_{e_{ij}})(\mathbf{x}_{ij}^M)$$

where $|e_{ij}|$ is the length of e_{ij}, $\mathbf{x}_{ij}^M$ its midpoint and $\mathbf{t}_{e_{ij}}$ its tangent unit vector.

Second family: twelve degrees of freedom per tetrahedron – A complete linear interpolation of a three-dimensional vector in a tetrahedron needs twelve degrees of freedom. The corresponding edge element can be obtained by taking two unknowns over each edge of the tetrahedron. Keeping the same notations as the ones used to introduce the first family of edge elements, one of the possibilities is to define the vector basis functions corresponding to an edge e_{ij} going from $\mathbf{r}_i$ to $\mathbf{r}_j$, as follows

$$\mathbf{w}_{ij}(\mathbf{r}) = \lambda_i(\mathbf{r}) \text{ grad } \lambda_j(\mathbf{r}) \quad , \quad i, j = 1, 2, 3, 4, \, i \neq j \, ; \tag{23}$$

let us denote by $\mathcal{P}_2(K)$ the space generated by the basis functions defined in (23). The interpolating function $\mathbf{u}_h$ on K for the vectorial state variable $\mathbf{u} \in (C^0(\overline{K}))^3$ has the following form

$$\mathbf{u}_h = \sum_{i=1}^{4} \sum_{j \neq i, j=1}^{4} \mathbf{w}_{ij} \, \beta_{ij}(\mathbf{u}) \quad \text{with} \quad \begin{array}{l} \beta_{ij}(\mathbf{u}) = |e_{ij}| \, (\mathbf{u} \cdot \mathbf{t}_{e_{ij}})(\mathbf{x}_i) \\ \beta_{ji}(\mathbf{u}) = |e_{ij}| \, (\mathbf{u} \cdot \mathbf{t}_{e_{ij}})(\mathbf{x}_j) \end{array}$$

where $\mathbf{x}_i$ and $\mathbf{x}_j$ are the end points of the edge e_{ij}.

Merging the two families – In paper [BBM00], the authors have shown that the mortar method combined with edge elements in three dimensions leads to an approximation which is slightly sub-optimal with the second family and give indications that with the first family non-optimal results could be feared. On the other hand, by using the second family of edge elements in one domain, the number of unknowns for a given mesh is multipled by two. To overcome the difficulties, the idea is based on the following two facts:

- taking the difference of $\mathbf{w}_{ij}$ and $\mathbf{w}_{ji}$ defined in (23) we get the old element $\mathbf{w}_{ij}$ defined in (22); moreover, one element $\mathbf{v} \in \mathcal{P}_1$ can be thought as an element $\mathbf{v} \in \mathcal{P}_2$ with the corresponding degrees of freedom $(\beta_{ij}, \beta_{ji}) = (\alpha_{ij}, -\alpha_{ij})$;
- the Lagrange multipliers of the mortar method are defined locally on Γ^m.

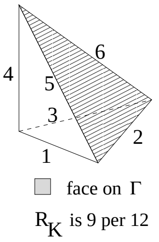
The compromise to have a good approximation without too many unknowns is to limit the use of the second family to all edges that belong to the interface Γ^m. The first family is then adopted to approximate the problem solution along all edges that do not belong to the interface (i.e. over each tetrahedron that does not meet the interface). The space of edge elements $\mathcal{P}$ involved in the definition (9) is the following:

$$\mathcal{P}(K) = \{\mathbf{u} \,|\, \mathbf{u}_{|e} \in \mathcal{P}_1\,,\ \forall e \notin \partial K \cap \Gamma^m \ \text{ and } \mathbf{u}_{|e} \in \mathcal{P}_2\,,\ \forall e \in \partial K \cap \Gamma^m\,\}\,. \tag{24}$$

From the implementation point of view, the merging can be done by introducing a rectangular matrix R_K that depends on the current tetrahedron K as follows:

$$\begin{array}{ll} R_K \in \mathcal{M}(6,12) & \partial K \cap \Gamma^m = \emptyset \ \text{ or reduced to one point} \\ R_K \in \mathcal{M}(7,12) & \partial K \cap \Gamma^m \text{ consists of one edge of } K\,, \\ R_K \in \mathcal{M}(9,12) & \partial K \cap \Gamma^m \text{ consists of one face of } K\,, \\ R_K \in \mathcal{M}(11,12) & \partial K \cap \Gamma^m \text{ consists of two faces of } K\,. \end{array}$$

$\mathcal{M}(n,m)$ denotes the set of matrices with n rows and m columns. Moreover, the local stiffness matrix associated to each tetrahedron is built using the second family for only those elements K that meet the interface, i.e. $S_K \in \mathcal{M}(12,12)$ if $\partial K \cap \Gamma^m \neq \emptyset$ nor to one point. In this case, the assembling process does not involve the full matrix S_K but the smaller one given by $R_K S_K R_K^T$ (we have got rid of the additional unknowns for all edges of K that do not lie on Γ^m).



$$R_K^T = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Figure 1: The elements of matrix R_K^T for a given tetrahedron K; to each edge among those with numbers 1, 3, 4, is associated one circulation and to those with numbers 2, 5, 6, are associated two.

Dealing with the first family

The use of the first family inside each sub-domain together with the second one at the interface glued together with the mortar element method as defined in the second section does not pollute the general accuracy of the problem. In order to analyse this we refer to the standard tool for the analysis of non-conforming approximation: the Berger-Scott-Strang Lemma. This Lemma allows to state:

$$|\mathbf{u} - \tilde{\mathbf{u}}_h|_{*,\Omega} \leq \inf_{\mathbf{v}_h \in \tilde{X}_h^c} |\mathbf{u} - \mathbf{u}_h|_{*,\Omega} + \sup_{\mathbf{v}_h \in \tilde{X}_h^c} \frac{\sum_{k=1}^{K} \langle \mathbf{v}_h^k \wedge \mathbf{n}_k, \mathbf{curl}\ \mathbf{u} \rangle_{-\frac{1}{2},\frac{1}{2},\partial\Omega_k}}{|\mathbf{v}_h|_{*,\Omega}} \tag{25}$$

where $\tilde{X}_h^c$ denotes the subspace of X_h^c composed of all functions that are of the first family inside the subdomains as described in the previous subsection, and $\tilde{\mathbf{u}}_h$ denotes the solution of problem (13) where X_h^c is replaced by $\tilde{X}_h^c$. The first contribution is known as the best fit of $\mathbf{u}$ by elements of $\tilde{X}_h^c$ and the second contribution is the consistancy error. This second contribution is exactly the same as in the analysis of problem (13) while the former is analyzed following the same steps of the proof of Theorem 2.3: starting from the local approximation od $\mathbf{u}_{|\Omega_k}$ by elements of the first family (e.g. the interpolation $\mathcal{I}_h^k\mathbf{u}_{|\Omega_k}$), we correct the trace value on the slave side of the interfaces by substructing from $\mathcal{I}_h^k\mathbf{u}_{|\Omega_k}$ the function obtained by prolongating by 0 the difference $\Pi_h^m(\mathcal{I}_h^k\mathbf{u}_{|\Omega_k} - \mathcal{I}_h^{-k}\mathbf{u}_{|\Omega_{-k}})$ between the current value on the slave face and the value derived from the application of the mortar condition.

Since the local interpolation operator has the same asymptotic approximation properties in the first and second family, the previous correction is optimal and we can state that the same error bound holds for $\tilde{\mathbf{u}}_h$ as what is stated in Theorem 2.3.

Applications

The flexibily and performance of a numerical method for the simulation of electromagnetic field distributions relies, in several cases, on the possibility of working with non-matching grids at the interface between adjacent sub–domains. One example is given by the treatment of moving structures. Our choice is to work in Lagrangian ones, dealing with non-conforming discretizations at the level of the sliding interface between the stator and the rotor. The second choice is less expensive from the computational point of view if we use a method that avoids re-meshing or interpolation procedures. Another example is the optimization of the structure shape for an electromagnetic device. We can re-mesh either the whole domain or only a region containing the shape to be optimized: in the second case it may be useful to work with non-matching grids to simplify the local re-meshing task and successive solution of the problem. A third example consists in the possibility of coupling variational methods of different others or with unknowns associated to different geometric entities.

Some preliminary results in magnetostatics

Currently, the work in progress consists in applying the described method to compute the distribution of induced currents in moving structures: this is an information of great importance for performances prediction and devices design. Nevertheless, the magnetostatic problem is of great interest due to the fact that we have to face all the difficulties of the method's implementation even if the geometry does not move. The movement treatment would add the additional cost of discretizing the coupling condition at each new position of the free part.

As an example of application, we present some results obtained by solving the magnetostatic problem in terms of the magnetic vector potential $\mathbf{A}$, i.e. the equation $\mathbf{curl}\,(\mu^{-1}\mathbf{curl}\,\mathbf{A}) = \mathbf{J}$, with homogeneous boundary conditions. We consider a hexahedrical domain divided into two sub–domains which are discretized by non–structured tetrahedrical coarse meshes. The computational domain in presented in Figure 2 while the magnetic induction $\mathbf{B} = \mathbf{curl}\,\mathbf{A}$ computed on matching and non–matching grids is displayed in Figure 3. In both cases, the information (i.e. the tangential component of the unknown) is well transmitted from one domain to the other.

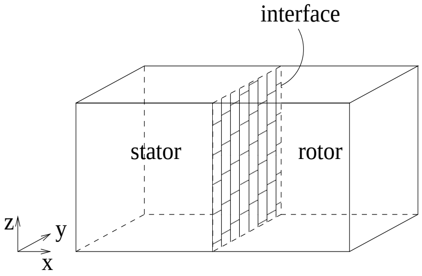


Figure 2: The domain Ω: a flat interface separates the two sub–domains.

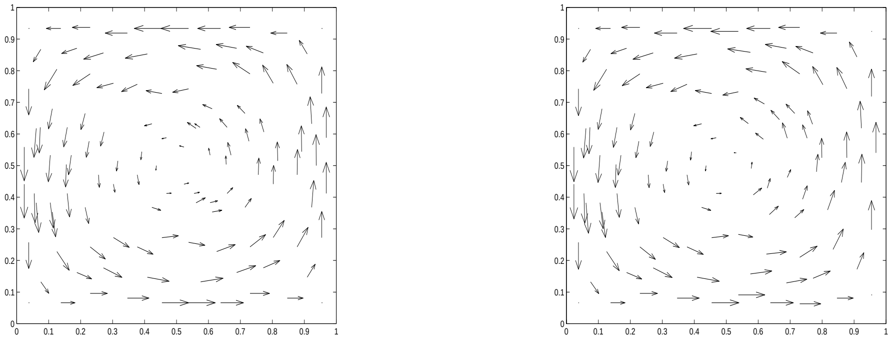


Figure 3: Field $\mathbf{B}$ on the plane $y = .25$ computed on matching (left) and non-matching (right) grids. The stored magnetic energy is ≈ 1.9 MJ in both cases.

Formulation and discretization of the magnetodynamic problem

We are given with a domain $\Omega \subset \mathbb{R}^3$, decomposed in a rotating part (rotor) Ω_1 and a static one (stator) $\Omega_2 = \Omega \setminus \bar{\Omega}_1$. Ω_1 is a cylinder that turns around its axis. Let $\theta \in C^1(0,T)$ be the law of rotation, i.e., $\theta(t)$ denotes the rotation angle at time t and $r_t : \Omega_1 \to \Omega_1$ the rotation operator which turns the domain Ω_1 with an angle $\theta(t)$ and r_{-t} its inverse. Here we suppose for simplicity that $\alpha > 0$ everywhere.

In both domains Ω_1 and Ω_2, we have to solve the equation (2) while the transmission conditions at Γ take into account the movement. They are:

$$r_t \mathbf{u}_1(r_{-t}\mathbf{x}, t) \wedge \mathbf{n}_\Gamma = \mathbf{u}_2(\mathbf{x}, t) \wedge \mathbf{n}_\Gamma \ , \tag{26}$$

$$r_t \beta(r_{-t}\mathbf{x}, t) \mathbf{curl}\, \mathbf{u}_1(r_{-t}\mathbf{x}, t) \wedge \mathbf{n}_\Gamma = \beta(\mathbf{x}, t) \mathbf{curl}\, \mathbf{u}_2(\mathbf{x}, t) \wedge \mathbf{n}_\Gamma \ . \tag{27}$$

Set $\mathcal{H} = H(\mathbf{curl}, \Omega_1) \times H_{0,\partial\Omega}(\mathbf{curl}, \Omega_2)$, we then are led to introduce

$$\mathcal{H}^t = \{\mathbf{u} = (\mathbf{u}_1, \mathbf{u}_2) \in \mathcal{H} \mid r_t \mathbf{u}_1(r_{-t}\mathbf{x}, t) \wedge \mathbf{n}_\Gamma = \mathbf{u}_2(\mathbf{x}, t) \wedge \mathbf{n}_\Gamma \ \forall \mathbf{x} \in \Gamma\}. \tag{28}$$

The problem obtained by considering equation (2) in both domain together with homogeneous boundary condition at $\partial\Omega$ and the transmission conditions (26-27) admits a unique solution $\mathbf{u} \in L^\infty(0, T, \mathcal{H}) \cap H^1(0, T, L^2(\Omega))$ when suitably interpreted in a variational sense both in time and space. Note that here the essential transmission condition (26) is strongly imposed in the definition of the functional space, while the natural one (27) is weakly imposed through the variational formulation (this is a consequence of the integration by parts). We are now in the position of making a discretization of this problem and the key point will be the discrete counterpart of the time-dependent constraint characterizing the definition of the space $\mathcal{H}^t$.

The mortar element method proposed in the second section provides an "optimal" spatial discretization of the stated problem. The computational domain is split up into two sub–domains Ω_1 and Ω_2 and the skeleton consists of 3 interfaces (see the Figure 4). Over each sub–domain, we consider the finite element discretization derived in the first part of the second section. We call $\mathcal{H}_h^t$ the resulting broken edge element space. The Lagrange multiplier spaces are chosen according to the second section , namely we have M_h^i, $i = 1, 2, 3$. At each interface, the matching condition turns out to be time-dependent, namely, for any $i = 1, 2, 3$ and $\mathbf{u}_h = (\mathbf{u}_{1,h}, \mathbf{u}_{2,h}) \in \mathcal{H}_h^t$ we have:

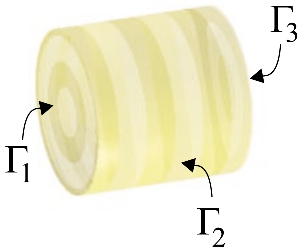


Figure 4: Interfaces for the definition of the mortar element method.

$$\int_{\Gamma_i} \big(r_t \mathbf{u}_{1,h}(r_{-t}\mathbf{x}, t) - \mathbf{u}_{2,h}(\mathbf{x}, t)\big) \wedge \mathbf{n}_\Gamma \cdot \psi_h^i d\Gamma = 0 \quad \forall \psi \in M_h^i.$$

The problem is then discretized in time by means of an implicit Euler method. The analysis of such a formulation is available in the 2D case together with some numerical results (see [BMR99], [Rap00]), and it is in progress for the 3D problem.

References

[ABDG98]C. Amrouche, C. Bernardi, M. Dauge, and V. Girault. Vector potentials in three-dimensional non-smooth domains. *Math. Meth. Appl. Sci.*, 21:823–864, 1998.

[BBM00]F. Ben Belgacem, A. Buffa, and Y. Maday. The mortar finite element method for $3d$ maxwell's equations: first results. SIAM J. Num. An. (submitted), 2000.

[BMR99]A. Buffa, Y. Maday, and F. Rapetti. A sliding mesh-mortar method for a two dimensional eddy-currents model of electric engines. Math. Meth. Anal. Num. (submitted), 1999.

[Hop99]R. Hoppe. Mortar edge element method in $\searrow^3$. *East West J. Num. An.*, 7(3):159–222, 1999.

[N80́]J.-C. Nédélec. Mixed finite elements in R^3. *Numer. Math.*, 35:315–341, 1980.

[N8́6]J.-C. Nédélec. A new family of mixed finite elements in R^3. *Numer. Math.*, 50:57–81, 1986.

[Rap00]F. Rapetti. The mortar edge element method on non-matching grids for eddy current calculations in moving structures. Int. J. Num. An. (submitted), 2000.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau, ©2001 DDM.org

29. Solving non-linear electronic packaging problems on parallel computers using domain decomposition

P.Chow[1], C.Bailey[2], C.Addison[1]

Introduction

Miniaturisation of electronic equipment, such as those found in a notebook computer, palm held devices, cell phones etc., requires high-density packing of electronic components onto printed circuit boards (PCB). To join the interconnections, solder materials are used to bond microprocessor chips and board during assembly. In the Reflow process case, the board assembly passes through a furnace where the solder bump initially in the form of solder paste, melts, reflows, and then solidifies to bond the interconnections. A number of defects may occur during and after this process such as, respectively, bridging of the liquid solder and cracking of solder joint, chip or board. With the increasing drive towards miniaturisation and smaller pitch sizes (gap between interconnection of solder bumps), these are serious issues to industry in manufacturing and component reliability in operation.

Finite Element Analysis (FEA) is used extensively in the electronic packaging community to calculate stress of solders and components, for reliability analyses [Lau93] [SYS97]. Computer simulations, together with some experiments, provides an effective design and optimization route to reducing these defects and in assessing solder and board integrity and reliability. For models to fully characterise the physical phenomena of the process that govern the integrity of the final joints requires the representing physics of:

- Heat transports with solder solidification involving latent heat evolutions
- Residual stress evolution involving thermal miss-match between materials.

Also, an integrated solution procedure is needed to solve governing equations of temperature, evolving solder shape, solidification, and stress, as they are interdependent. For example, stress analysis is dependent on temperature changes in solid regions. While for the solder joint formation, the solder material will initially, after heating, be liquid and when the board exits the furnace it starts to solidify and stress developments begin.

A microprocessor chip commonly has large number of interconnects that bonds to a circuit board. The general modelling practice is to take a Macro-Micro approach that simulates a single interconnect or assumes each interconnect behaves like a beam in the finite element analysis. In the Macro-Micro case, there is a data transfer between the models at each time step, see Figure 1. A detailed 3D model requires a sizeable mesh and long computing time, i.e. solving non-linear equations of thermal and mechanical systems; thereby, constraining the number of the number of cycles possible for the

[1] Fujitsu European Centre for Information Technology Ltd, UK, {chow, addison}@fecit.co.uk
[2] University of Greenwich, UK, c.bailey@gre.ac.uk

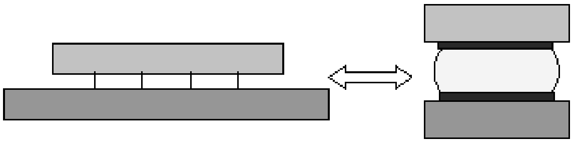


Figure 1: Solder Modelled as Beams (Macro) and Continuum (Micro)

design and optimization process. In the multiple chips case, it can easily leads to models with mesh sizes having millions of elements.

Parallel computing technology opens up the possibility of undertaking such detailed and large-scale analyses, and delivers the solution in a practical timeframe. In application areas such as automobile and aerospace, parallel computing has significantly reduced the time for analyses and increased the size of models (both of physical models and mesh sizes) that can be performed. Such success is also due to the advances in the Domain Decomposition method, now a key element in the majority of parallel models such as mesh or domain partitioning, linear and non-linear solver strategies, and matching and non-matching overlapping grids. Here, we show some parallel computations of 3D electronic packaging models that involves cooling, solidification, and residual stresses of solder joints and throughout the component during assembly. All the computations are performed on a Fujitsu AP3000 system using up to 12 processing elements, with the largest model completed having over 1 million elements.

Heat Transport Equations

The equations governing the physics of heat transport and solidification can be expressed as:

$$\rho c \frac{\partial T}{\partial t} + \nabla \cdot (\rho c \underline{v} T) = \nabla \cdot (k \nabla T) + S$$

where T, t, ρ, c, k, $\underline{v}$ and S are the temperature, time, density, specific heat, thermal conductivity, velocity vector, and source term, respectively. The equation for evolution of latent heat during solidification is represented by the source term, and expressed by:

$$S = -L\rho \frac{\partial f}{\partial t} - L\rho \nabla (\underline{v} f)$$

where L and f are the latent heat and liquid-fraction, respectively. The relationship between the liquid-fraction and temperature describes how the material (here it is the solder) solidifies between the liquidus and solidus temperatures range. For isothermal materials the latent heat release is instantaneous; this means liquidus and solidus temperatures are the same and translates to a vertical jump in the curve between liquid-fraction and temperature. Such numerical discontinuity needs to be addressed properly to maintain energy conservation, if not, it is possible to artificially gain or lose energies in the system. To fully conserve energy, the Enthalpy Source-Based method [VS91] is used to address such discontinuity.

Stress-Strain Equations

For stress analysis, the incremental equilibrium equations governing solid deformation are [ZT89] [TBC95]:

$$\Delta\sigma_{ij,j} = 0 \qquad (i, j = x, y, z)$$

where $\Delta\sigma_{ij,j}$ are the Cartesian components of the Cauchy stress tensor. The incremental stress $\Delta\underline{\sigma}$, $(\Delta\sigma_{xx} \ \ \Delta\sigma_{yy} \ \ \Delta\sigma_{zz} \ \ \Delta\sigma_{xy} \ \ \Delta\sigma_{xz} \ \ \Delta\sigma_{yz})$ is due to the elastic strain given by:

$$\Delta\underline{\sigma} = [D]\Delta\underline{e}^{el}$$

where $\Delta\underline{e}^{el}$ and $[D]$ are the elastic strain vector and elastic materials matrix respectively. The elastic strains are dependent on the total $\Delta\underline{e}$, thermal $\Delta\underline{e}^{th}$, and visco-plastic strain $\Delta\underline{e}^{vp}$ vectors given by:

$$\Delta\underline{e}^{el} = \Delta\underline{e} - \Delta\underline{e}^{th} - \Delta\underline{e}^{vp}$$

For small strains, the total strain, $\Delta\underline{e}$, is given by the gradient in displacements, which in matrix form is:

$$\Delta\underline{e} = [L]\Delta\underline{d}$$

where $[L]$ is the matrix of differentials and $\Delta\underline{d}$ is the displacement vector. The visco-plastic strains in this analysis are represented by the Perzyna [Per66] constitutive model give by:

$$\underline{\dot{e}}^{vp} = \frac{\partial e^{vp}}{\partial t} = \frac{2\lambda}{3\sigma^{eq}} \left(\frac{\sigma^{eq}}{\sigma^{y}} - 1 \right)^{n} S_{ij}$$

where λ, σ^{eq}, σ^{y}, n and S_{ij} are the fluidity, von-mises stress, yield stress, strain rate sensitivity, and deviatoric stress, respectively. Within a time increment, Δt, the incremental visco-plastic strain is:

$$\Delta\underline{e}^{vp} = \Delta t\underline{\dot{e}}^{vp}$$

An Integrated Procedure

The solution procedure for the coupling of solidification and stress, plus others such as thermal convection (not included in the solder for the present study) is given in reference [BCF^{+}96], and is in the PHYSICA toolkit. Figure 2 shows the coupled solution procedure for transient analysis of temperature, solidification, and stress.

Within the time step loop the thermal variables, temperature and liquid-fraction, are first solved and the temperature changes, ΔT, calculated. To account for latent heat evolution during solidification, and other non-linearity, an iterative procedure is generally used. Next, the resulting changes of temperature and liquid-fraction are used in the stress calculations.

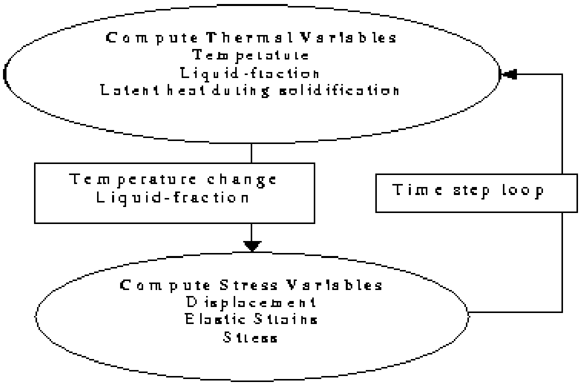


Figure 2: Coupled solution procedure in PHYSICA

Based on temperature changes the incremental displacements are calculated. Using the new displacements and current total stress, σ^o, the incremental total and viscoplastic strains can be calculated. The incremental elastic strain and stress can then be obtained for the time step. This incremental stress will update the total stress ($\sigma^n = \sigma^o + \Delta\sigma$) that will change the values for viscoplastic strain. Due to non-linearity and coupling, an iterative procedure is commonly used.

After the thermal and stress variables have been solved within the time step, the values for these become the old values for calculations at the next time step. As cooling progresses the liquid solder region solidifies and the resulting "solid" elements becoming eligible for stress calculations. The solution procedure continues until the simulation finishes.

Software & Parallel Model

The PHYSICA toolkit [Phy] [CCB$^+$96] from University of Greenwich is used for the study. It has an open single-code component-based software framework [CBM$^+$99] for coupled and Multiphysics applications. The code is 3D, unstructured mesh, with analysis models for fluid flow, heat transfer, solidification, elastic/visco plastic, combustion and radiosity. PHYSICA's parallel model (see refs [CBM$^+$99] [MCJ97]) is based on the Single Program Multiple Data (SPMD) paradigm, where each processing element runs the same program on a sub-portion of the model domain. The mesh, representing the model domain, is partitioned using the graph-partitioning tool JOSTLE [Jos] into sub-domains that are minimized for data exchanges between the overlapped region. Message passing is then used to perform any data exchange needed between these sub-domains on each processing element (PE).

For parallel codes to scale well for performance, the non-scalable portions needs to be eliminated - if not possible, it will be a point of concern in the solution procedure's critical path. Some common examples of non-scalable parts are, reading and writing to files (parallel input and output are currently system dependent, if available), and global summation operatives commonly found in popular linear solvers. In the version

Table 1: Parallel performance for electronic package case

CPU time in minutes			
PE	Solution time	Total time	Speedup
1	17.40	17.92	
4	6.00	6.75	2.65
8	4.87	5.62	3.19

of PHYSICA used for this work, the embedded JOSTLE is scalar and it is a critical point in the overall scalability. Also, by default, the whole mesh is first read into memory for JOSTLE to perform the partitioning before distribute to the PEs for processing. For larger model sizes this non-stop processing can be inappropriate due to memory demands by JOSTLE and PHYSICA - together, the optimal performance configuration of the computing system can degrade significantly. In this work, the mesh partitioning and analysis are executed separately for all the large models; first partitioning the mesh and save the PE index for each mesh element to file, then the analysis phase reads the save index data and distributes the mesh element to each PE for processing. For multiple run cases with the same number of PEs, this ”partition and save” approach may, in some instances, be more advantages than the non-stop approach; since for any amount of multiple runs the partitioning only occurs once. A parallel version of JOSTLE is underway to address the non-stop processing and other related matters.

Parallel Results

Table 1 shows the computing times for a model solve in electronic packing (consisting of 21,413 vertices, 57,577 faces and 18,150 elements) in CPU minutes for 1, 4 and 8 PEs. For this model the total time for 8 PEs is a speedup of 3.19 over a single PE, the solution speedup (without initial setup, such as mesh partitioning, and reading and writing to files) is 3.57. This means the non-solution portion takes about 11 and 13 percents of the total CPU time, respectively for 4 and 8 PEs, compared to 0.03 percent for single PE.

Figure 3 shows one quarter of a chip bonding to a PCB example being modelled during the reflow process, and Figure 4 showing an enlarged view of the solder bumps with two different attachment materials at top and bottom. The model consists of 273,504 vertices, 1,133,207 faces and 425,890 elements. Figure 5 shows the solidification fronts of the solder bumps during cooling phase of the reflow process. The corner solder bump is solidifying at a rate faster than its neighbours as indicated by the solidification front in dark colours. Figure 6 shows the magnitude of visco-plastic strain and deformation throughout the solder bumps at the end of reflow when all the solder bumps are solid. Again the corner solder bump has a higher amount of strain than all the other solid bumps. The deformation, as shown by the inclining solder bump, is board contracting more than the chip because of different thermal coefficients in the material properties.

Table 2 gives the computing times from 2 up to 12 PEs in CPU hours. The model

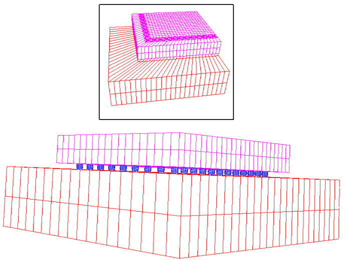


Figure 3: Chip bonding to PCB

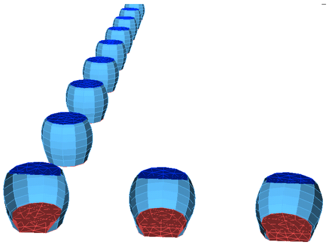


Figure 4: Solder bumps

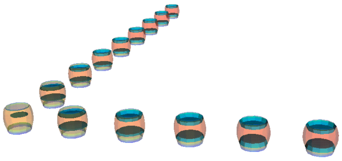


Figure 5: Solidifying solder bumps

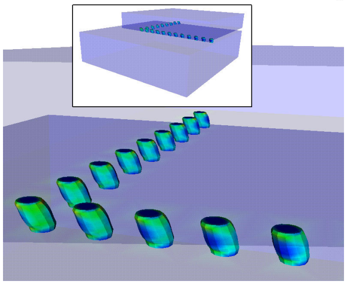


Figure 6: Solder bumps deformation

is too big for single PE on the AP3000, as it reports out of memory. The CPU runtime for 2 PEs is under 7 hours and 12 PEs is under 2 hours. This gives a speedup factor of about 8 for 12 PEs, representing a saving of about 5 hours in analysis time or an extra 1 to 2 cycles in the design and optimization process. For lower PE runs the speedup factor moves nearer to the linear scaling mark.

To get an idea of a single PE runtime, the same model was run on a Sun Enterprise 10000 (E10000) with 2GB memory in scalar mode. With UltraSPARC processors inside the AP3000 and E10000 systems, U170 and U250 respectively, a total CPU time of 15.34 hours was reported on the E10000 with solution time being 15.24 hours. If we put the E10000 result with the 12 PEs of AP3000, it represents a saving of over 13 hours in analysis time or giving an extra 5 to 6 cycles in the design & optimization process. In terms of speedups, it represents a factor of 9 (compared to 8) for analysis time and 11 (compared to 9) for the solution period. Figures 7 and 8, respectively, show graphs of parallel performance for total and solution times; the triangle markers indicates an idea of the true speedup if the 1 PE time had been possible. These estimates are obtained by substituting the E10000 single PE result in the calculation for speedups.

From the performance graphs, it is encouraging to see the curve for total time shows there are potential gains for this model case by adding more PEs (12 PEs is the highest we have access to at present). A downside is the non-solution portion of the analysis time is also increasing with PEs, some 19 percent (or 20 minutes) for the 12 PEs case. To see how larger models may fair, a similar problem with model size of

Table 2: Parallel performance for a chip bonding to a PCB example

CPU time in hours			
PE	Solution time	Total time	Speedup
2	6.281	6.748	
3	4.233	4.621	2.921
4	3.261	3.648	3.699
5	2.703	3.070	4.397
6	2.350	2.701	4.997
7	2.082	2.488	5.425
8	1.816	2.153	6.268
9	1.684	2.023	6.671
10	1.530	1.848	7.305
11	1.460	1.762	7.658
12	1.379	1.698	7.951

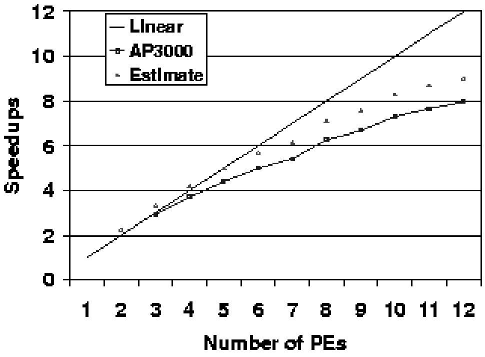


Figure 7: Total time performance

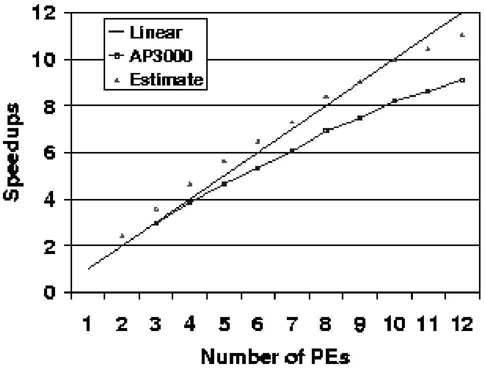


Figure 8: Solution time performance

1,205,997 vertices, 3,504,048 faces and 1,149,312 elements was conducted on 12 PEs. The parallel performance reports an analysis time of 6.01 hours with a solution period of 4.94 hours, representing some 18 percent or 1 hour for non-solution activities. This is encouraging, as the percentage figure has not altered significantly.

Conclusion

The above results indicate significant reduction in analysis time for electronic packaging applications, even for a model mesh size of 18,000 elements. As for larger models with elements in the millions, such as multiple chips on board cases, it can exceed the memory capacity of today's workstations. Parallel computing with domain decomposition offers a solution to run and deliver the analysis within a design and development timeframe.

The numerical experiments conducted indicate some 20 percent of the analysis time on 12 PEs are in non-solution activities, such as data retrieving and saving to files and setup period for parallel computation. Therefore, there is great potential in reducing this figure even further and improving parallel performance by removing the present scalar events in the procedure's critical path such as having parallel IO and parallel mesh partitioning. Memory usage is lot higher in mechanical analysis section than in the thermal section; a ratio of about 2 to 1 has been observed - this is primarily due to the segregated method used in thermal analysis section as to the full-system employed in the mechanical.

References

[BCF$^+$96]C. Bailey, P. Chow, Y. Fryer, M. Cross, and K. Pericleous. Multiphysics modelling of the metals casting processes. *Proceedings of Royal Society of London A*, 452:459–486, 1996.

[CBM$^+$99]P. Chow, C. Bailey, K. McManus, C. Addison, and M. Cross. A single-code model for multiphysics analysis-engine in parallel and distributed computers with the physica toolkit. In Choi-Hong Lai, Petter E. Bjørstad, Mark Cross, and Olof B. Widlund, editors, *Domain Decomposition Methods in Sciences and Engineering*, pages 396–402, Bergen, 1999. Domain Decomposition Press.

[CCB$^+$96]M. Cross, P. Chow, C. Bailey, N. Croft, J. Ewer, P. Leggett, K. McManus, K.A. Pericleous, and M.K. Patel. PHYSICA – a software environment for the modelling of multiphysics phenomena. *ZAMM*, 76:101–104, 1996.

[Jos]JOSTLE. University of Greenwich, London, UK (www.gre.ac.uk/˜wc06/jostle).

[Lau93]J. Lau. *Thermal stress and strain in micro-electronic packaging*. Van Nostrand Reinhold, New York, 1993.

[MCJ97]K. McManus, M. Cross, and S. Johnson. Issues and strategies in the parallelisation of unstructured multiphysics codes. In *Parallel and distributed computing for computational mechanics*, 1997.

[Per66]P. Perzyna. Fundamental problems in viscoplasticity. *Advanced Applied Mechanics*, 9:243–377, 1966.

[Phy]PHYSICA. University of Greenwich, London, UK (physica.gre.ac.uk).

[SYS97]H. Sakai, N. Yasuda, and K. Seyama. Reliability study of flip chip organic bga/csp packages. In *Semicon West 97, BGA Technology Conference*, 1997.

[TBC95]G. Taylor, C. Bailey, and M. Cross. Solution of elastic/visco-plastic constitutive equations: a finite volume approach. *Apply Mathematical Modelling*, 19:746–760, 1995.

[VS91]V. Voller and C. Swaminathan. General source-based methods for solidification phase change. *Numerical Heat Transfer*, 19:175–190, 1991.

[ZT89]O.C. Zienkiewicz and R. Taylor. *The finite element method.* McGraw-Hill, 1989.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau, ©2001 DDM.org

30. A heterogeneous domain decomposition for initial– boundary value problems with conservation laws and electromagnetic fields

C.A. Coclici, W.L. Wendland[1], J. Heiermann, M. Auweter–Kurtz[2]

Introduction

In this paper a nonoverlapping domain decomposition method for the numerical treatment of compressible viscous plasma flows inside a self–field magnetoplasmadynamic (MPD) accelerator is developed. The high–enthalpy magneto–plasma flow is modelled by a system of conservation laws extended by partial differential equations describing the electromagnetic field. Due to the tremendous computational time needed for the numerical solution of the complex equations, the flow–field domain is decomposed into two model zones, characterized by different physical properties of the flow. The complete model of the extended Navier–Stokes equations in the near field of the accelerator is coupled with a simplified model of the extended Euler equations in the far field. The coupling is realized by appropriate transmission conditions at the artificial coupling boundary.

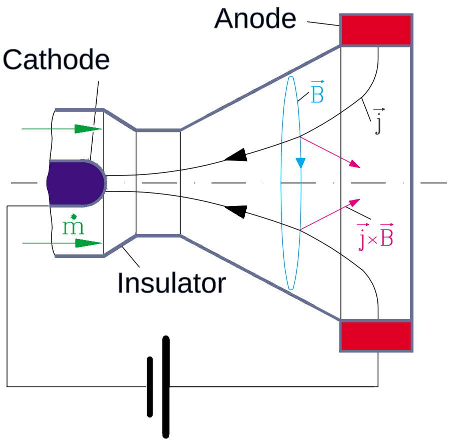


Figure 1: MPD thruster

The principle of a self–field thruster is shown in Figure 1. A cold gas (argon) enters the accelerator and is heated up by an electric discharge to a hot plasma. The plasma expands thermally and accelerates into a test tank in the laboratory. In addition, the plasma is accelerated by electromagnetic Lorentz forces. The flow is described by the conservation equations for mass, momentum and energy for the heavy particles (argon atoms Ar^0 and ions Ar^{1+}, Ar^{2+}), by the conservation equation for the electron and the ionization energy, and by the Maxwell equations of classical electrodynamics.

Furthermore, reaction equilibrium, thermal non–equilibrium (two–fluid model), and laminar flow are assumed at this time.

The complete system of governing equations is employed within an essentially smaller near–field region Ω_1, containing the thruster, and is coupled by appropriate

[1] Mathematical Institute A, University of Stuttgart, {coclici, wendland}@mathematik.uni-stuttgart.de

[2] Institute of Space Systems, University of Stuttgart {heierman, auweter}@irs.uni-stuttgart.de

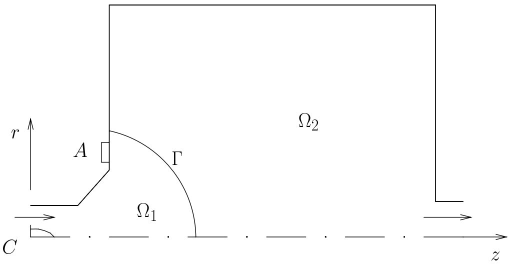


Figure 2: Decomposition of the computational domain

transmission conditions across the artificial boundary Γ with a simplified model in the complementary far field Ω_2, corresponding to the test tank. Generally, the far–field simplifications should be chosen in such a way, that on one hand the flow in the far–field domain is still modelled accurately enough, and on the other hand, the numerical treatment can be performed efficiently.

The axisymmetric plasma flow is described in cylindrical coordinates by the vector–valued function

$$\mathbf{W} = \mathbf{W}(r, z;\, t) := \big[\,\mathbf{w}, p_H, T_H;\ \mathbf{w}_e;\ \mathbf{w}_{EB}\,\big]^\top(r, z;\, t), \quad (r, z) \in \Omega,\ t \in [0, T].$$

Here, $\mathbf{w} = (\rho, \rho v_r, \rho v_z, E_H)^\top$ collects the conservative variables with the density ρ, the velocity vector $\mathbf{v} = (v_r, v_z)^\top$, and the energy of the heavy particles E_H. The pressure and the temperature of the heavy particles are denoted by p_H and T_H, respectively. The function $\mathbf{w}_e = (\mathrm{e}_{ei}, p_e, T_e)^\top$ describes the electron component of the plasma, with e_{ei} containing the electron and the ionization energy, and with p_e and T_e representing the pressure and the temperature of the electron component, respectively. Finally, $\mathbf{w}_{EB} = (\mathbf{E}, \mathbf{B}, \mathbf{j})^\top$ contains the electromagnetic field $(\mathbf{E}, \mathbf{B})$ and the electric current density $\mathbf{j}$.

Modelling of the near field

The heavy–particle flow is modelled by the compressible Navier–Stokes equations which are extended due to the influence of an arc discharge. They take in cylindrical coordinates the form

$$\frac{\partial \mathbf{w}_1}{\partial t} + \mathrm{div}_{(r,z)}\mathbf{F}(\mathbf{W}_1) = \mathrm{div}_{(r,z)}\mathbf{R}(\mathbf{w}_1, \nabla_{(r,z)}\mathbf{w}_1) + \mathbf{G}(\mathbf{W}_1) \ \text{ in } \ \Omega_1 \times [0, T]. \tag{1}$$

The function $\mathbf{F}$ contains the convective part of the Navier–Stokes equations (here, with the pressure field $p = p_H + p_e$), and, in addition, an electromagnetic pressure term derived from the source terms. We represent $\mathbf{F}$ as

$$\mathbf{F} = (\mathbf{F}_r, \mathbf{F}_z)(\mathbf{W}) = (\mathbf{f}_r, \mathbf{f}_z)(\mathbf{w}, \mathbf{w}_e) + (\mathbf{g}_r, \mathbf{g}_z)(\mathbf{w}_{EB}),$$

where, with the purely azimuthal magnetic field $\mathbf{B} = (0, B, 0)^\top$ and with the magnetic permeability of vacuum $\mu_0 > 0$,

$$\begin{aligned}
\mathbf{f}_r(\mathbf{w}, \mathbf{w}_e) &:= \big(\rho v_r,\ \rho v_r^2 + (p_H + p_e),\ \rho v_r v_z,\ [E_H + (p_H + p_e)]\ v_r\big)^\top,\\
\mathbf{f}_z(\mathbf{w}, \mathbf{w}_e) &:= \big(\rho v_z,\ \rho v_z v_r,\ \rho v_z^2 + (p_H + p_e),\ [E_H + (p_H + p_e)]\ v_z\big)^\top,\\
\mathbf{g}_r(\mathbf{w}_{EB}) &:= \big(0,\ B^2,\ 0,\ B^2 v_r\big)^\top\!/(2\mu_0),\ \mathbf{g}_z(\mathbf{w}_{EB}) := \big(0,\ 0,\ B^2,\ B^2 v_z\big)^\top\!/(2\mu_0).
\end{aligned}$$

The viscous terms are collected in the function $\mathbf{R} = (\mathbf{R}_r, \mathbf{R}_z)(\mathbf{w}, \nabla_{(r,z)}\mathbf{w})$ where

$$\begin{aligned}
\mathbf{R}_r(\mathbf{w}, \nabla_{(r,z)}\mathbf{w}) &:= \big(0,\ \tau_{rr},\ \tau_{rz},\ \tau_{rr}v_r + \tau_{rz}v_z + \lambda_H\ \partial T_H/\partial r\big)^\top, \\
\mathbf{R}_z(\mathbf{w}, \nabla_{(r,z)}\mathbf{w}) &:= \big(0,\ \tau_{zr},\ \tau_{zz},\ \tau_{zr}v_r + \tau_{zz}v_z + \lambda_H\ \partial T_H/\partial z\big)^\top,
\end{aligned}$$

with the heat conductivity $\lambda_H > 0$ of the heavy–particle flow, and with

$$\tau_{rr} = \mu\left[2\frac{\partial v_r}{\partial r} - \frac{2}{3}\ \mathrm{div}\ \mathbf{v}\right],\ \tau_{rz} = \tau_{zr} = \mu\left[\frac{\partial v_r}{\partial z} + \frac{\partial v_z}{\partial r}\right],\ \tau_{zz} = \mu\left[2\frac{\partial v_z}{\partial z} - \frac{2}{3}\ \mathrm{div}\ \mathbf{v}\right]$$

defining the components of the viscous part of the stress tensor; $\mu > 0$ represents the viscosity coefficient. The function $\mathbf{G}$ contains the electromagnetic force and heat terms as well as quantities describing the heat transfer due to the collisions between the plasma components:

$$\begin{aligned}
\mathbf{G}(\mathbf{W}) := \Bigg(0,\ &\frac{1}{r}\left[p_H + p_e - \frac{2}{3}\mu\Big(2\frac{v_r}{r} - \frac{\partial v_r}{\partial r} - \frac{\partial v_z}{\partial z}\Big) - \frac{B^2}{2\mu_0}\right],\ 0, \\
&\Big(p_e + \frac{B^2}{2\mu_0}\Big)\,\mathrm{div}\,\mathbf{v} - \frac{v_r}{r}\frac{B^2}{\mu_0} + \sum_{\nu=0}^{2} n_\nu n_e \alpha_{e\nu}(T_e - T_H)\Bigg)^\top.
\end{aligned}$$

Here, n_ν ($\nu = 0, 1, 2$) and n_e are the densities of the heavy particles and of the electrons, respectively, and $\alpha_{e\nu}$ are heat transfer coefficients. Note that, by including the Lorentz terms $\mathbf{j} \times \mathbf{B}$ as $B^2/(2\mu_0)$ in the fluxes, our formulation observes as much conservation as possible. Consequently, conservative numerical methods (as e.g. the finite volume method) are good candidates to be used for the numerical treatment of the problem. The conservation of the electron and ionization energy is given in $\Omega_1 \times [0, T]$ by

$$\begin{aligned}
\frac{\partial \mathrm{e}_{ei}}{\partial t} + \mathrm{div}\,(\mathrm{e}_{ei}\mathbf{v}) - \mathrm{div}\,(\lambda_{ei}\nabla T_e) = {}& -p_e\,\mathrm{div}\,\mathbf{v} + \frac{5}{2}\frac{k}{e}\mathbf{j}\cdot\nabla T_e - \frac{1}{n_e e}\mathbf{j}\cdot\nabla p_e \\
&+ \sum_{\nu=0}^{2} n_\nu n_e \alpha_{e\nu}(T_H - T_e) + \frac{|\mathbf{j}|^2}{\sigma}. \qquad (2)
\end{aligned}$$

Here, λ_{ei} denotes the heat conductivity for the electron component of the flow, k is the Boltzmann constant, and σ is the electric conductivity. The Maxwell equations and Ohm's law for plasmas read

$$\mathrm{rot}\,\mathbf{B} = \mu_0\mathbf{j},\quad \mathrm{rot}\,\mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t},\quad \mathrm{div}\,\mathbf{B} = 0; \qquad \mathbf{E} = \frac{\mathbf{j}}{\sigma} - \mathbf{v}\times\mathbf{B} + \beta\,\mathbf{j}\times\mathbf{B} - \beta\,\nabla p_e$$

(β – Hall parameter), leading to the discharge equation

$$\frac{\partial^2 B}{\partial r^2} + \frac{\partial^2 B}{\partial z^2} + \left[\frac{1}{r} - \mu_0\sigma v_r\right]\frac{\partial B}{\partial r} - \mu_0\sigma v_z\frac{\partial B}{\partial z} - \left[\frac{1}{r^2} + \mu_0\sigma\Big(\frac{\partial v_r}{\partial r} + \frac{\partial v_z}{\partial z}\Big)\right]B = F_B, \qquad (3)$$

where $F_B = F_B(\sigma, p_e, \mu_0)$ denotes a source term. Additional equilibrium reactions are incorporated into our model but, for brevity, they are not given here explicitly (for more details, see e.g. [Sle99]).

In order to get a more profound understanding of the complex physical processes involved, theoretical and numerical investigations have been performed [Sle99, WKAK98]. Continuing this work, the mathematical formulations of the conservation equations have been extended in [HAKE$^+$99], where an advanced numerical finite–volume code has been written in order to capture the plasma flow physics accurately.

However, due to the high complexity of the model and the tremendous computational costs, the system has been discretized only in the vicinity Ω_1 of the MPD thruster, identified here as the near field. Γ is there considered as outflow (freestream) boundary and characteristic boundary conditions, using data obtained from measurements, are used. In that model one faces the problem that a certain amount of ambient (cold) far–field gas *recirculates* into the hot plasma jet in the near field. Consequently, parts of the outflow boundary Γ get “inflow” properties and require additional information about the flow quantities. This makes the numerical treatment of the plasma flow in the far field Ω_2 necessary. We consider in Ω_2 a simplified model which takes into account the physical properties of the flow, and couple this model to the complete system in Ω_1. Our coupling procedure extends previous work on heterogeneous domain decomposition in aerodynamics (see, e.g. [QS95, Coc98, CW01]) to the case of compressible magneto–plasma flows.

Simplified modelling of the far field

In a first approximation we assume that far away from the MPD accelerator the shear stresses $\tau_{rr}, .., \tau_{zz}$ and the heat conduction terms $\lambda_H \partial_{r(z)} T_H$, defining the quantity $\mathbf{R}$, are strongly dominated by the convective part. Hence, we assume the heavy–particles flow to be inviscid in Ω_2. At the moment, we also assume that the magnetic field $\mathbf{B}$ vanishes identically in Ω_2. The system of conservation laws takes the simplified form

$$\frac{\partial \mathbf{w}_2}{\partial t} + \operatorname{div}_{(r,z)}(\mathbf{f}_r, \mathbf{f}_z)(\mathbf{w}_2, \mathbf{w}_{e,2}) = \mathbf{H}(\mathbf{w}_2, \mathbf{w}_{e,2}) \quad \text{in} \quad \Omega_2 \times [0, T], \tag{4}$$

with the simplified source term

$$\mathbf{H}(\mathbf{w}_2, \mathbf{w}_{e,2}) := \left(0,\ \frac{p_H + p_e}{r},\ 0,\ p_e \operatorname{div} \mathbf{v} + \sum_{\nu=0}^{2} n_\nu n_e \alpha_{e\nu}(T_e - T_H)\right)^{\top}.$$

Furthermore, as a consequence of $\mathbf{j} = \operatorname{rot} \mathbf{B}/\mu_0 \equiv 0$ in Ω_2, the equation of conservation of electron and the ionization energy (2) becomes in $\Omega_2 \times [0, T]$

$$\frac{\partial \mathrm{e}_{ei}}{\partial t} + \operatorname{div}(\mathrm{e}_{ei}\mathbf{v}) - \operatorname{div}(\lambda_{ei} \nabla T_e) = -p_e \operatorname{div} \mathbf{v} + \sum_{\nu=0}^{2} n_\nu n_e \alpha_{e\nu}(T_H - T_e). \tag{5}$$

Transmission conditions

These conditions should be chosen in such a way, that on one hand, the fundamental physical laws are respected, and on the other hand, the resulting coupled problem is well–posed and consistent with the full original problem. The continuity of the characteristic variables could be chosen as transmission condition, but according to the theory of hyperbolic equations, this can be required only across that part of the interface, where the corresponding characteristics enter the hyperbolic region, see e.g. [Hir88]. We also refer to [QS95, Coc98, CW01], where the continuity of the Riemann invariants across the inflow part and compatibility conditions across the outflow part of the boundary are used. In accordance with the conservation laws, the continuity of the normal flux yields a transmission condition on the complete interface Γ: the total flux associated with the full model in Ω_1 (containing the inviscid as well as the viscous contributions) is set equal to the normal inviscid flux, that results from the simplified equations in Ω_2:

$$-\big[\mathbf{R}_r(\mathbf{w}_1,\nabla\mathbf{w}_1)n_r+\mathbf{R}_z(\mathbf{w}_1,\nabla\mathbf{w}_1)n_z\big]+\big[\mathbf{f}_r(\mathbf{w}_1,\mathbf{w}_{e,1})+\mathbf{g}_r(\mathbf{w}_{EB,1})\big]\,n_r$$

$$+\big[\mathbf{f}_z(\mathbf{w}_1,\mathbf{w}_{e,1})+\mathbf{g}_z(\mathbf{w}_{EB,1})\big]\,n_z=\mathbf{f}_r(\mathbf{w}_2,\mathbf{w}_{e,2})\,n_r+\mathbf{f}_z(\mathbf{w}_2,\mathbf{w}_{e,2})\,n_z \qquad (6)$$

across Γ. The flux condition has successfully been used for pure flow problems (see, for example, [QS95, CW01]). However, it implies that the solutions of the coupled problem may exhibit jumps at the interface, depending on the magnitude of the viscosity and heat transfer terms neglected in the far field (for more details, see [Coc98]). Since the solution of the original problem should satisfy the *natural* transmission conditions at the artificial interface (i.e. continuity of the solution and of the total normal flux), the approximate extended Navier–Stokes / extended Euler solution can only be a first approximation and needs to be corrected by special terms accounting for the loss of continuity and maintaining the continuity of the normal flux. A boundary layer correction for a simplified transmission problem is presented in [CW00] in the framework of singular perturbation theory. This analysis is extended for the problem under consideration in [CMW00].

In order to assure the electron heat transfer across the interface, we impose the continuity of the co–normal derivative of the electron temperature T_e:

$$\Big[\lambda_{ei,1}\frac{\partial T_{e,1}}{\partial\mathbf{n}}\Big](r,z)=\Big[\lambda_{ei,2}\frac{\partial T_{e,2}}{\partial\mathbf{n}}\Big](r,z)\quad\text{for all}\quad(r,z)\in\Gamma. \qquad (7)$$

Finally, we impose $B\equiv 0$ on Γ.

Numerical aspects and results

In the numerical code, the extended conservation laws (1) and (4), describing the heavy–particle motion, as well as the electron and the ionization energy equations (2) and (5) are solved on an unstructured, dual mesh by using a second–order finite volume upwind scheme based on explicit Euler time–stepping. The discharge equation (3) is currently solved by triangular finite elements with linear ansatz functions using an SOR scheme. For a detailed description we refer to [HAKE$^+$99]. A finite volume formulation is in preparation for this conservation equation.

The full computational domain including the near field of the MPD accelerator and the far field corresponding to the tank, are shown in Figure 3. The area of the far field is about 80 times larger than that of the near field, emphasizing the necessity of simplifying the mathematical model in the far field.

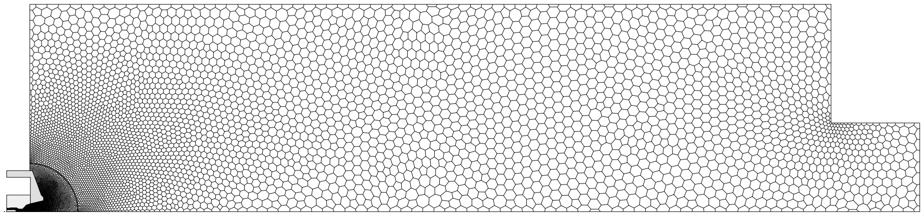


Figure 3: Full computational domain

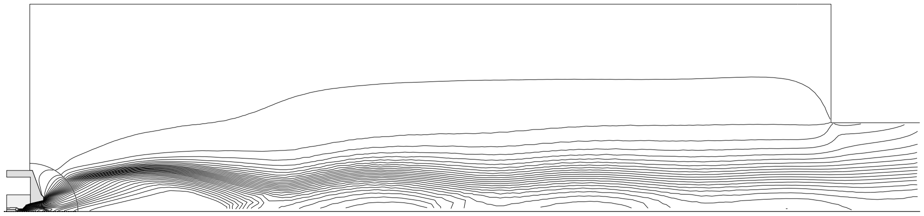


Figure 4: Isolines of v_z

The isolines of the axial velocity component v_z give an overall impression of the plasma flow: The plasma is accelerated in the MPD accelerator, then it is expanded into the tank, and finally it flows out of the tank at the far right.

The coupling domain, where dual cells on both sides of the coupling boundary touch each other, is shown in Figure 5 (left). Both meshes are produced with an advancing front algorithm. The mesh generator enforces the global mesh to be conforming at the artificial coupling boundary.

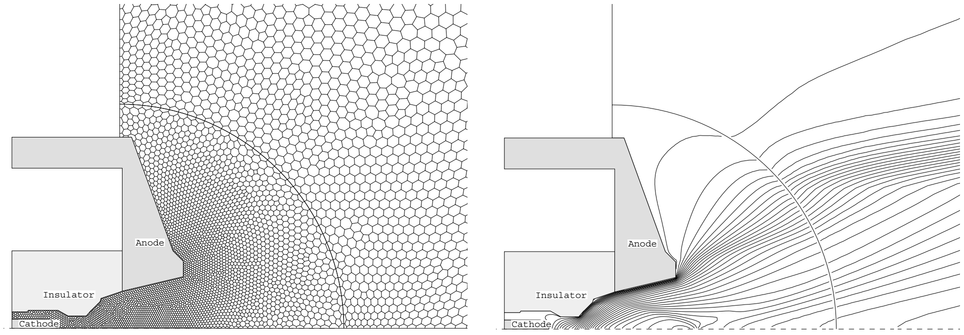


Figure 5: Computational domain for the coupling (left); isolines of v_z (right)

The isolines of the longitudinal velocity v_z for the coupled solution are presented on the right. Obviously, the coupling method works very well for the central, hot plasma jet. Up to one nozzle radius above the centerline, v_z passes smoothly the coupling boundary Γ. However, farther away from the centerline, the isolines become slightly discontinuous and do not cross the interface smoothly.

It turns out that the neglection of the heat conduction terms corresponding to the heavy–particle flow is significant, see Figure 6 (left). While the isolines of the heavy–particle temperature T_H behave smoothly across the part of the interface Γ contained in the central plasma jet, we can see the discontinuities of the solution in the region above very clearly. The explanation for this is that the heavy–particle heat conduction is still physically relevant with respect to the inviscid Euler energy flux, such that the heavy particle heat conduction cannot be neglected in this geometrical decomposition. The local discontinuity of T_H also causes a slight discontinuity and non–smoothness of v_z in the critical region. Therefore, the far field domain will be further decomposed into a small intermediate domain attached to the near field and the complementary far–field region. In the intermediate field, the heavy–particle heat conduction will be considered, while the components of the viscous stress tensor will be neglected. Also a rigorous dimension analysis of the flow quantities is necessary to justify the use of the intermediate model.

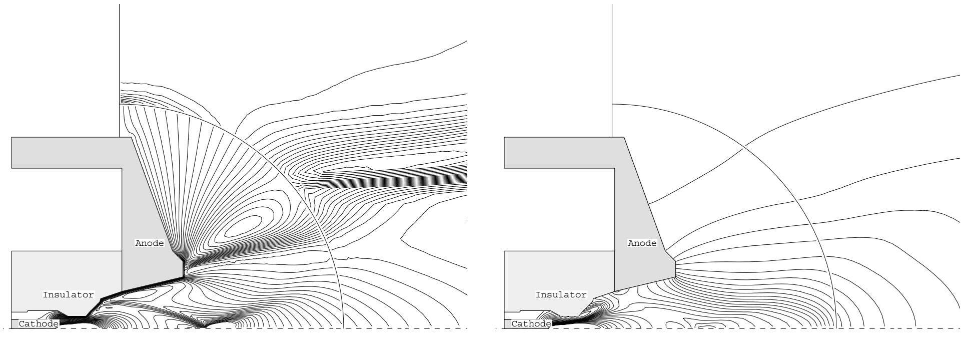


Figure 6: Isolines of the heavy–particles temperature T_H (left) and of the electron temperature T_e (right)

The approximate coupled solution also shows that the electron temperature T_e passes the artificial interface smoothly, as can be seen in Figure 6 (right). Thus, the *natural* transmission condition (7), used for the coupling of the equations (2) and (5) is justified also numerically.

Finally, we outline that by using the heterogeneous domain decomposition, the very complex compressible magneto–plasma flow has been computed for the first time within the whole MPD accelerator plus tank configuration, and that the influence of the far field through the recirculating amount of gas has been simulated numerically. Our investigation shows that the heterogeneous domain decomposition method is an excellent tool which can be efficiently used in the numerical treatment of nonlinear boundary value problems of high complexity.

Acknowledgement

This work has been supported by the German Research Foundation DFG (Deutsche Forschungsgemeinschaft) through the projects “Domain Decomposition Methods and Far–Field Boundary Conditions for Initial Boundary Value Problems with Conservation Laws and Electromagnetic Fields” (We 659/35–1) and “Conservation Equations and Numerical Solutions for Technically Relevant Magneto–Plasmas” (Au 85/9–1,2). The authors thank H.J. Kaeppeler for many stimulating discussions and helpful contributions.

References

[CMW00]C.A. Coclici, G. Morosanu, and W.L. Wendland. Asymptotic correction of the numerical solution to a heterogeneous domain decomposition for compressible plasma flows. In preparation, 2000.

[Coc98]C.A. Coclici. *Domain Decomposition Methods and Far–Field Boundary Conditions for Compressible Flows Around Airfoils.* PhD thesis, Mathematical Institute A, University of Stuttgart, Pfaffenwaldring 57, D-70569 Stuttgart, Germany, 1998.

[CW00]C.A. Coclici and W.L. Wendland. The coupling of hyperbolic and elliptic boundary value problems with variable coefficients. *Math. Meth. Appl. Sci.*, 23:401–440, 2000.

[CW01]C.A. Coclici and W.L. Wendland. Analysis of a heterogeneous domain decomposition for compressible viscous flow. *Math. Models Meth. Appl. Sci.*, 4(11), 2001. accepted for publication.

[HAKE$^+$99]J. Heiermann, M. Auweter-Kurtz, A. Eberle, H.J. Kaeppeler, U. Iben, and P.C. Sleziona. Recent improvements of numerical methods for the simulation of mpd thruster flow on adaptive meshes. In *Proceedings of the 26th International Electric Propulsion Conference, Kitakyushu*, 1999.

[Hir88]C. Hirsch. *Numerical Computation of Internal and External Flows.* Wiley, 1988.

[QS95]A. Quarteroni and L. Stolcis. Homogeneous and heterogeneous domain decomposition for compressible fluid flows at high reynolds numbers. *Numerical Methods for Fluid Dynamics*, 5:113–128, 1995.

[Sle99]P.C. Sleziona. *High Enthalpy Flows for Spaceflight Applications (in German).* Shaker Verlag, Aachen, 1999.

[WKAK98]H.P. Wagner, H.J. Kaeppeler, and M. Auweter-Kurtz. Instabilities in mpd thruster flows: 1. space charge instabilities in unbounded and inhomogeneous plasmas. *Journal of Physics D: Appl. Phys.*, 31:519–528, 1998.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau,

31. A Defect Correction Method for the Retrieval of Acoustics Waves

G.S. Djambazov[1], C.-H. Lai[2], K.A. Pericleous[3]

Introduction

For a given mathematical problem and a given approximate solution, the residue or defect may be defined as a quantity to measure how well the problem has been solved. Such information may then be used in a simplified version of the original mathematical problem to provide an appropriate correction quantity. The correction can then be applied to correct the approximate solution in order to obtain a better approximate solution to the original mathematical problem. Such idea has been around for a long time and in fact has been used in a number of different ways.

A famous example of defect correction is the computation of a solution to the nonlinear equation $f(x) = 0$. Suppose $\bar{x}$ is an approximate solution, then $-f(\bar{x})$ is the defect. One possible version of the original problem is to define $\bar{f}(x) \equiv f'(\bar{x})(x - \bar{x}) + f(\bar{x}) = 0$. In fact, if one replaces $x - \bar{x}$ as v, then v is the correction which is obtained by solving $f'(\bar{x})v = -f(\bar{x})$ and an updated approximation can be obtained by evaluating $x := \bar{x} + v$. Most defect correction are used in conjunction with discretisation methods and two-level multigrid methods [BS84]. This paper is not intended to give an overview of defect correction methods but to use the basic concept of a defect correction in conjunction with fluctuations in flow field variables for sound and noise retrieval.

Recall that sound waves - manifested as pressure fluctuations - are typically several orders of magnitude smaller than the pressure variations in the flow field that account for flow acceleration. Furthermore, they propagate at the speed of sound in the medium, not as a transported fluid quantity. A decomposition of variables was first introduced in [DLP97] and has been further examined in [Dja98] to include three types of components. These components include (1) the mean flow, (2) flow perturbations or aerodynamic sources of sound, and (3) the acoustic perturbation. We have demonstrated the accurate computation of (1) and (2) in [DLP98]. Mathematically, the flow variable U may be written as $\bar{u} + u$ where $\bar{u}$ denotes the mean flow and part of aerodynamic sources of sound and u denotes the remaining part of the aerodynamic sources of sound and the acoustic perturbation.

While flow perturbation or aerodynamic sources of sound may be easier to recover, it is not true for the acoustic perturbation because of its comparatively small magnitude. In fact, the solutions of the Reynolds averaged Navier-Stokes equations reveal only a truncated part of the full physical quantities. This paper follows the basic principle of the defect correction as discussed above and applies the concept to the recovery of the propagating acoustic perturbation. The method relies on the use of a lower order partial differential equation defined on the same computational domain

[1] University of Greenwich, G.Djambazov@gre.ac.uk
[2] University of Greenwich, C.H.Lai@gre.ac.uk
[3] University of Greenwich, K.Pericleous@gre.ac.uk

where a residue exists such that the acoustic perturbation may be retrieved through a properly defined coarse mesh.

This paper is organised as follows. First, derivation of a lower order partial differential equation resulting from the Navier-Stokes equations is given. Second, accurate representation of residue on a coarse mesh is discussed. The coarse mesh is designed in such a way as to allow various frequencies of noise to be studied. Suitable interpolation operators are studied for the two different meshes. Third, 1-D and 2-D examples are used to illustrate the concept. Finally, future work is discussed.

The Defect Correction Method

The aim here is to solve the non-linear equation

$$\mathcal{L}U := \mathcal{L}(\bar{u} + u) = 0 \tag{1}$$

where $\mathcal{L}$ is a non-linear operator depending on $U := \bar{u} + u$. Using the concept of decomposition of variable, U is now written as $\bar{u} + u$, where $\bar{u}$ is the mean flow and u is the acoustic perturbation. Note that $u \ll \bar{u}$. In the case of sound generated by the motion of fluid, it is natural to imagine $\mathcal{L}$ as the Navier-Stokes operator. For a 2-D problem,

$$\bar{u} = \begin{bmatrix} \bar{\rho} \\ \bar{v}_1 \\ \bar{v}_2 \end{bmatrix} \qquad u = \begin{bmatrix} \rho \\ v_1 \\ v_2 \end{bmatrix}$$

where ρ is the density of fluid and v_1 and v_2 are the velocity components along the two spatial axes. Using the summation notation of subscripts, the 2-D Navier-Stokes problem $\mathcal{L}u = 0$ is written as

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho v_j}{\partial x_j} = 0$$

and

$$\frac{\partial v_i}{\partial t} + v_j \frac{\partial v_i}{\partial x_j} + \frac{1}{\rho}\frac{\partial P}{\partial x_i} - \frac{f_i}{\rho} = 0$$

where P is the pressure and f_i is the external force along i-th axis.

Suppose (1) may be split and re-written as

$$\mathcal{L}(\bar{u} + u) \equiv \mathcal{L}\bar{u} + E\{\bar{u}\}u + K[\bar{u}, u] \tag{2}$$

where $E\{\bar{u}\}$ is an operator depending on the knowledge of $\bar{u}$ and $K[\bar{u}, u]$ is a functional depending on the knowledge of both $\bar{u}$ and u. Following the concept of defect correction, $\bar{u}$ may be considered as an approximate solution to (1). Hence one can evaluate the residue of (1) as

$$R \equiv \mathcal{L}(\bar{u} + u) - \mathcal{L}\bar{u} = -\mathcal{L}\bar{u}$$

which may then be substituted into (2) to give

$$E\{\bar{u}\}u + K[\bar{u}, u] = R \tag{3}$$

In many cases, $K[\bar{u}, u]$ is small and can then be neglected. In those cases, the problem in (3) is a linear problem and may be solved more easily to obtain the acoustics fluctuation u. A non-linear iterative solver is required in order to obtain u for cases when $K[\bar{u}, u]$ is not negligible. Finally, to obtain the approximate solution $\bar{u}$, one only needs to solve $\mathcal{L}\bar{u} = 0$.

Expanding $\mathcal{L}(\bar{u} + u) = 0$ for $\mathcal{L}$ being the Navier-Stokes operator and re-arranging we obtain

$$\frac{\partial \rho}{\partial t} + \bar{v}_j \frac{\partial \rho}{\partial x_j} + \bar{\rho}\frac{\partial v_j}{\partial x_j} + [v_j \frac{\partial(\bar{\rho} + \rho)}{\partial x_j} + \rho \frac{\partial(\bar{v}_j + v_j)}{\partial x_j}] = -[\frac{\partial \bar{\rho}}{\partial t} + \bar{v}_j \frac{\partial \bar{\rho}}{\partial x_j} + \bar{\rho}\frac{\partial \bar{v}_j}{\partial x_j}]$$

and

$$\frac{\partial v_i}{\partial t} + \bar{v}_j \frac{\partial v_i}{\partial x_j} + \frac{1}{\bar{\rho}}\frac{\partial P}{\partial x_i} - \frac{f_i}{\bar{\rho}}$$

$$+[\frac{\rho}{\bar{\rho}}\frac{\partial(\bar{v}_i + v_i)}{\partial t} - (v_j + \frac{\rho}{\bar{\rho}}(\bar{v}_j + v_j))\frac{\partial(\bar{v}_i + v_i)}{\partial x_j}] = -[\frac{\partial \bar{v}_i}{\partial t} + \bar{v}_j \frac{\partial \bar{v}_i}{\partial x_j} + \frac{1}{\bar{\rho}}\frac{\partial \bar{P}}{\partial x_i} - \frac{\bar{f}_i}{\bar{\rho}}] \quad (4)$$

It can be seen that (4) may be written in the form of (3) where

$$E\{\bar{u}\}u = \left[\begin{array}{c} \frac{\partial \rho}{\partial t} + \bar{v}_j \frac{\partial \rho}{\partial x_j} + \bar{\rho}\frac{\partial v_j}{\partial x_j} \\ \frac{\partial v_i}{\partial t} + \bar{v}_j \frac{\partial v_i}{\partial x_j} + \frac{1}{\bar{\rho}}\frac{\partial P}{\partial x_i} - \frac{f_i}{\bar{\rho}} \end{array} \right] \quad (5)$$

$$K[\bar{u}, u] = \left[\begin{array}{c} v_j \frac{\partial(\bar{\rho}+\rho)}{\partial x_j} + \rho \frac{\partial(\bar{v}_j + v_j)}{\partial x_j} \\ \frac{\rho}{\bar{\rho}}\frac{\partial(\bar{v}_i + v_i)}{\partial t} - (v_j + \frac{\rho}{\bar{\rho}}(\bar{v}_j + v_j))\frac{\partial(\bar{v}_i + v_i)}{\partial x_j} \end{array} \right] \quad (6)$$

$$R = \left[\begin{array}{c} -[\frac{\partial \bar{\rho}}{\partial t} + \bar{v}_j \frac{\partial \bar{\rho}}{\partial x_j} + \bar{\rho}\frac{\partial \bar{v}_j}{\partial x_j}] \\ -[\frac{\partial \bar{v}_i}{\partial t} + \bar{v}_j \frac{\partial \bar{v}_i}{\partial x_j} + \frac{1}{\bar{\rho}}\frac{\partial \bar{P}}{\partial x_i} - \frac{\bar{f}_i}{\bar{\rho}}] \end{array} \right] \equiv -\mathcal{L}\bar{u} \quad (7)$$

From the knowledge of physics of fluids, the acoustic perturbations ρ and v_j are of very small magnitude (this is not true for their derivatives), therefore, K may be considered negligible due to the reason that any feedback from the propagating waves to the flow can be completely ignored, except in some cases of acoustic resonance, which is impossible to occur in the examples of this paper. Hence the equation $E\{\bar{u}\}u = R$, with E given by (5), which is known as the linearised Euler equation, can be solved in an easier way. The remaining question is to obtain the approximate solution $\bar{u}$ to the original problem (2). It is well known that CFD analysis packages provide excellent methods for the solution of $\mathcal{L}\bar{u} = 0$. Therefore one requires to use a Reynolds averaged Navier-Stokes package supplemented with turbulence models such as [CPC95] to provide a solution of $\bar{u}$. Physically, one requires $\bar{u}$ to be as accurate as possible to capture all the physics such as turbulence and vortices.

A Two-Level Numerical Scheme

In order to simulate accurately the approximate solution, $\bar{u}$, to the original problem, $\mathcal{L}U = 0$, the QUICK differencing scheme [Leo79] is used which produces sufficiently accurate results of $\bar{u}$ for the purpose of evaluating the residue as defined in (7). A sufficiently fine mesh has to be used in order to preserve vorticity motion. However, situations are different for the numerical solutions of linearised Euler equations [DLP97][Dja98], where the mesh has to be much coarser in order to obey Courant limit and to account for the fact that the acoustic wavelength is larger than the vortex diameter. An account on various high order finite difference schemes and its mesh requirements for the numerical solution of linearised Euler equations can be found in [Dja98]. After evaluating the residue on the fine mesh, it is then required to transfer these residuals onto the coarser mesh. Physically, the residue is in fact the sound source that will disappear without the proper retrieval technique as discussed in this paper.

Let h denote the mesh to be used in the Reynolds averaged Navier-Stokes solver. Instead of evaluating $\bar{u}$, one would solve the discretised approximation $\mathcal{L}_h \bar{u}_h = 0$ to obtain $\bar{u}_h$. The residue on the fine mesh h can be computed as $\mathcal{L} u_h$ by means of a higher order approximation [Dja98]. Let H denote the mesh for the linearised Euler equations solver. Again instead of evaluating u, one would solve the discretised approximation $E_H\{\bar{u}_H\} u_H = R_H$ to obtain u_H. Here R_H is the projection of R onto the mesh H. Let $I_{\{h,H\}}$ be a projection operator to project the residue computed on the fine mesh h to the coarser mesh H. The projected residue can then be used in the numerical solutions of linearised Euler equations. Let $I_{\{H,h\}}$ be an interpolation operator to interpolate the acoustic signals from the coarser mesh back to the finer mesh. Therefore the two-level numerical scheme is

```
For non-resonance problems:
    Solve $\mathcal{L}_h \bar{u}_h = 0$
    $R_H := -I_{\{h,H\}} \mathcal{L} \bar{u}_h$
    $\bar{u}_H := I_{\{h,H\}} \bar{u}_h$
    Solve $E_H\{\bar{u}_H\} u_H = R_H$
    $u_h := I_{\{H,h\}} u_H$
    $\bar{u} := \bar{u}_h + u_h$
```

Note that R_H cannot be computed as $\mathcal{L} I_{\{h,H\}} \bar{u}_h$ because $\mathcal{L}$ is a non-linear operator.

In the actual implementation, a pressure-density relation which also defines the speed of sound c in air is used:

$$\frac{\partial P}{\partial \rho} = c^2 \approx 1.4 \frac{\bar{P}}{\bar{\rho}} \tag{8}$$

and the first component of the linearised Euler equations in (5) becomes

$$\frac{\partial P}{\partial t} + \bar{v}_j \frac{\partial P}{\partial x_j} + \bar{\rho} c^2 \frac{\partial v_j}{\partial x_j} = -c^2 [\frac{\partial \bar{\rho}}{\partial t} + \bar{v}_j \frac{\partial \bar{\rho}}{\partial x_j} + \bar{\rho} \frac{\partial \bar{v}_j}{\partial x_j}] \tag{9}$$

The purpose of this substitution is to make sure that the new fluctuations P and v_i do not contain a hydrodynamic component, and hence it allows them to be resolved on regular Cartesian meshes [Dja98] which is essential for the accurate representation of the acoustic waves or the fluctuation quantity u. On the other hand, an unstructured mesh is usually used to obtain $\bar{u}_h$. The two different meshes overlapped one another on the computational domain. The computational domain for the linearised Euler equations is not necessarily exactly the same as the one for the CFD solutions. However, the computational domain for the linearised Euler equations must be large enough to contain the longest wavelength of a particular problem under consideration. The numerical examples as shown in next section do not involve any solid objects, therefore $I_{\{h,H\}}$ and $I_{\{H,h\}}$ are simply arithmetic averaging processes.

Numerical Examples

To test the feasibility of this approach the simple 1-D example is considered of an initial-value wave propagation problem with exact solution

$$\begin{array}{rcl} P & = & f(x-ct)+f(x+ct) \\ \bar{\rho}cv_1 & = & f(x-ct)-f(x+ct) \\ f(x) & = & \left\{ \begin{array}{l} \frac{A}{2}(1+\cos 2\pi\frac{x}{\lambda}), |x|<\frac{\lambda}{2} \\ 0, |x|\geq\frac{\lambda}{2} \end{array} \right. \end{array} \tag{10}$$

where A is the amplitude and λ is the wavelength of the pressure pulses that start from the origin ($x=0$) at $t=0$.

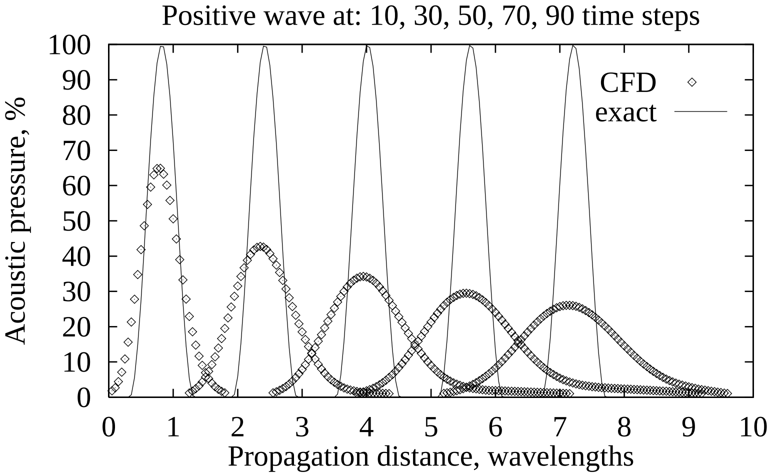


Figure 1: Analytic and preliminary CFD solutions of test problem

For the fine grid problem, $\mathcal{L}_h\bar{u}_h=0$, the initial conditions $P=2f(x)$ and $\bar{v}_1=0$ were prescribed for the CFD solution which uses a structured finite volume code [CHA95] with QUICK differencing scheme for the momentum equations. The time dependent result pictured in Figure 1 agrees with the analytic solution only in phase, but not in amplitude. Refining of the mesh does not improve the result at all. Since the problem is symmetrical with respect to the origin, only the right part ($x>0$) is shown (and solved for). Uniform mesh was used to avoid averaging of the residual sources with this test. The time step of the CFD simulation can be several times larger than the time step of the explicit Euler solver (which has to obey the Courant limit). In this example the CFD makes 12.5 time steps per cycle with 20 points per wavelength. In fact, time steps smaller than this produce greater numerical errors over the same propagation distance. This is most probably due to the false diffusion of the CFD schemes which accumulates with every time step.

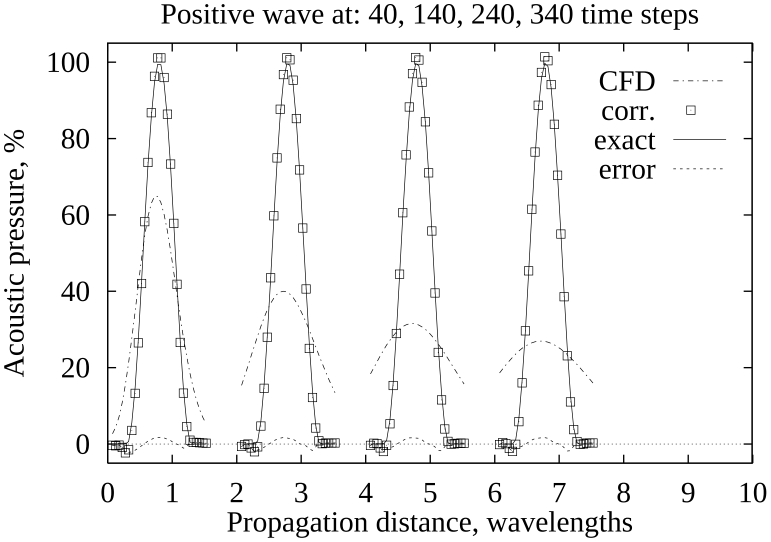


Figure 2: Defect corrected solution of test problem

The coarse grid solver, $E_H\{\bar{u}_H\}u_H = R_H$, or the acoustic module starts with zero initial conditions, and gradually accumulates the differences between the real pressure and velocity fields and their CFD representations. This process is driven by the source terms of (4) which are discretised in a time-accurate way. The solution in Figure 2 is obtained with second order approximation of the CFD quantities along the temporal axis, and its maximum error is about 2%. If linear approximations are used (which require only two stored CFD steps) the overall error becomes a little higher than 6%.

There are no external sources of mass and no external forces are acting on the fluid in this example. Also, the viscous stresses can be completely ignored with these 1-D sound waves: $\overline{f_i} = 0$ (see equation 4).

In Figure 2 the defect corrected solution (which is the sum of the CFD solution and the linearised Euler solution) is shown at regular intervals in order to trace the wave propagating from left to right. The time step with the acoustic module is 4 times smaller than the CFD step, and this is equivalent to 50 time steps per cycle. Since the acoustic procedure is fully explicit, these correction steps are computationally very inexpensive (the acoustic module needs less than 10 s to compute the correction of this example including the input and output of disk files). It can be seen that the result of this one-dimensional test is very encouraging.

As a more realistic example, a 2-D production of sound waves due to the generation of a vortex series within a flow medium as depicted in Figure 3 is examined. There are no solid bodies in the flow domain, and no acoustic source cells have been predefined. The vortices are initiated by the time-dependent source patch in the middle of the left boundary of the CFD domain. There is a background flow at a rate of 160 m/s from left to right which is not shown in Figure 3. An additional source of mass is associated with the sinusoidal in time (with an amplitude of 12 m/s) source of momentum in the vertical direction, active during time steps 1–30. Both of these cooperate in the production of acoustic waves which originate at the source patch.

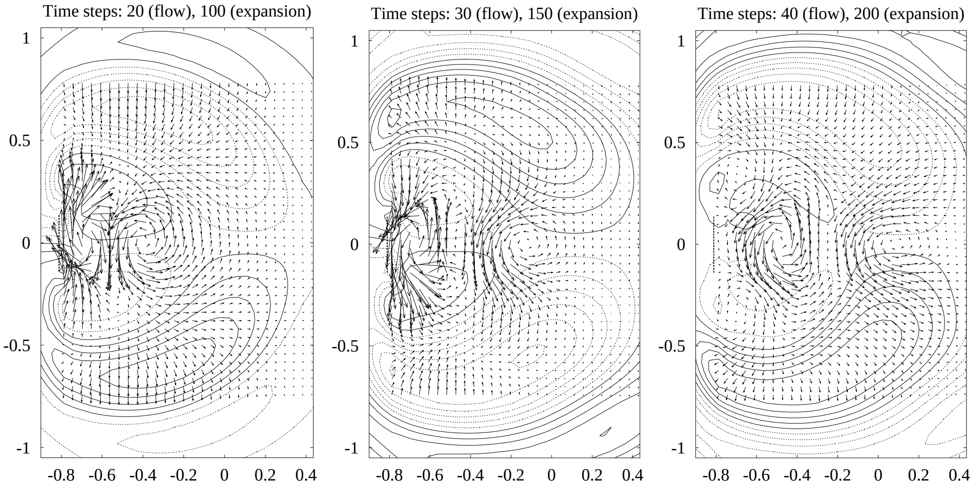


Figure 3: Vortex generation and acoustic correction pressure contours (positive - solid lines, negative - dashed lines, spacing: 20 Pa). Velocity vector scale: 3 m/s to 0.2 m. Vertical dashed line marks vortex generation source patch.

The same fully-implicit in time CFD code [CHA95] with QUICK differencing scheme for the momentum equations was used to simulate the generation and the convection of vortices. The mesh density is indicated in Figure 3 by the density of the arrows representing velocity vectors. As expected, no acoustic waves can be identified in the resulting CFD pressure field. After the correction steps (re-discretisation, mapping of residuals, and linearised Euler solution), the missing part of the pressure field is obtained, and it is shown in Figure 3 by contours. In this case of regular meshes with no solid objects, the mapping procedure is simple: two CFD cells in the vertical direction constitute one acoustic cell, and simple arithmetic averaging is used for the mapping.

Figure 3 shows clearly the acoustic waves that have been produced at the vortex generation patch propagating upwards, downwards, and out of the domain. There is no analytical validation for this example, but the results obtained are physically correct.

Conclusions

A framework of defect correction has been established for computational aeroacoustics. Based on this defect correction framework, it is possible to study aerodynamic sound generation in a systematic way. A 1-D example with validation and a 2-D example with physically correct results are shown. The authors are currently investigating a 2-D example with proper validation and are extending the concept to 3-D problems.

References

[BS84]K. Böhmer and H. J. Stetter, editors. *Defect Correction Methods: Theory and Applications.* Springer-Verlag, Wien, 1984.

[CHA95]CHAM Ltd, Wimbledon, UK. *PHOENICS, Version 2.1.3*, 1995.

[CPC95]N. Croft, K. Pericleous, and M. Cross. PHYSICA: A multiphysics environment for complex flow processes. In C. Taylor, editor, *Num. Meth. Laminar & Turbulent Flow '95*, page 1269. Pineridge Press, U. K., 1995.

[Dja98]Georgi S. Djambazov. *Numerical Techniques for Computational Aeroacoustics.* PhD thesis, Computing and Mathematical Sciences, University of Greenwich, 30 Park Row, Greenwich, London SE10 9LS, United Kingdom, September 1998.

[DLP97]G. Djambazov, C.-H. Lai, and K. Pericleous. Development of a domain decomposition method for computational aeroacoustics. In Petter E. Bjørstad, Magne S. Espedal, and David E. Keyes, editors, *Domain Decomposition Methods in Science and Engineering.* John Wiley & Sons, 1997. Proceedings from the Ninth International Conference, June 1996, Bergen, Norway.

[DLP98]G.S. Djambazov, C.-H. Lai, and K.A. Pericleous. Efficient computation of aerodynamic noise. In J. Mandel, C. Farhat, and X.-C. Cai, editors, *Tenth International Conference on Domain Decomposition Methods*, pages 506–512. AMS, Contemporary Mathematics 218, 1998.

[Leo79]B.P. Leonard. A stable and accurate convective modelling procedure based on quadratic upstream interpolation. *Computer Methods in Applied Mechanics and Engineering*, 19:59–98, 1979.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau, ©2001 DDM.org

32. Numerical solution of vascular flows by heterogeneous domain decomposition methods

L. Fatone[1], P. Gervasio[2], A. Quarteroni[3]

Introduction

In this note we investigate a problem arising from fluid dynamics for hemodynamics, using heterogeneous domain decomposition techniques. In particular we will couple Navier-Stokes equations with Oseen or Stokes equations, as advocated in papers [FGQ99] and [FGQ00].

Our interest is twofold. On one hand we would like to assess the quality of the coupled heterogeneous models; in particular we want to compare two options where the Oseen flux or the Stokes flux is matched continuously at the interface. On the other hand, we wish to carry out iterative substructuring method to solve the coupled problem. This iterative procedure has been introduced and analyzed in [FGQ00] for a general problem.

More generally in multi-field domain decomposition problems, different physical, mathematical or numerical models are adopted in different parts of the computational domain. One motivation is to develop parallel algorithms, the other is to provide an efficient way to reduce the complexity of the problem in certain regions, by using there a simpler mathematical model.

Given a bounded domain $\Omega \subset \mathbb{R}^2$, with a Lipschitz boundary $\partial\Omega$, $T > 0$, a vector field $\mathbf{f}$, a constant viscosity $\nu > 0$, we are interested in approximating the velocity field $\mathbf{u} = \mathbf{u}(\mathbf{x}, t)$ and the pressure field $p = p(\mathbf{x}, t)$ for the incompressible Navier-Stokes equations:

$$\partial_t \mathbf{u} - \nu\Delta\mathbf{u} + (\mathbf{u}\cdot\nabla)\mathbf{u} + \nabla p = \mathbf{f}, \quad \nabla\cdot\mathbf{u} = 0 \quad \text{in } \Omega\times(0,T) \tag{1}$$

by a multi-field approach. The idea is to consider two disjoint subregions Ω_1 and Ω_2 of Ω such that $\overline{\Omega}_1 \cup \overline{\Omega}_2 = \overline{\Omega}$, and to couple the Navier-Stokes equations (1) restricted to the subregion Ω_1 with the following linear Oseen equations

$$\partial_t \mathbf{u} - \nu\Delta\mathbf{u} + (\mathbf{u}_\infty\cdot\nabla)\mathbf{u} + \nabla p = \mathbf{f}, \quad \nabla\cdot\mathbf{u} = 0 \quad \text{in } \Omega_2\times(0,T), \tag{2}$$

where $\mathbf{u}_\infty$ is a prescribed solenoidal vector field. Sometimes the Oseen equations can be replaced by the Stokes equations, which are a special case of (2) with $\mathbf{u}_\infty = \mathbf{0}$.

The Navier-Stokes subregion Ω_1 can be a suitable internal domain of Ω and the Oseen subregion Ω_2 an exterior subdomain. Otherwise Ω_1 can be the part of Ω where the flow is quite perturbed by the presence of an obstacle. On the common boundary

[1] Department of Mathematics, Politecnico di Milano, Milano (Italy) (on leave from EPFL, Lausanne (CH)), Lorella.Fatone@epfl.ch

[2] Department of Mathematics, University of Brescia, Brescia (Italy), Paola.Gervasio@unibs.it

[3] Department of Mathematics EPFL, Lausanne (CH) and Department of Mathematics, Politecnico di Milano, Milano (Italy), Alfio.Quarteroni@epfl.ch

of the two subdomains, $\Gamma := \partial\Omega_1 \cap \partial\Omega_2$, correct transmission conditions have to be imposed.

The mathematically admissible transmission conditions at subdomain interfaces have been determined and analyzed in [FGQ99]. A Dirichlet/Neumann iterative procedure among subdomains has been proposed to solve the coupled Navier-Stokes/Oseen (or Navier-Stokes/Stokes) problem and its analysis was carried out in [FGQ00].

In the first and second Sections we recall the general problem, while in the last Section we carry out the numerical results and the assessment of the proposed method.

Multi-domain formulations and transmission conditions

We consider a vector field $\mathbf{w} : \Omega :\rightarrow \mathbb{R}^2$ such that $\mathbf{w}_i = \mathbf{w}_{|\Omega_i}$, for $i = 1, 2$ and

$$\mathbf{w}_1 = \mathbf{u}_{|\Omega_1} \quad \text{and} \quad \mathbf{w}_2 \;\; \text{is equal either to} \;\; \mathbf{u}_{\infty|\Omega_2} \;\; \text{or to} \;\; \mathbf{0}. \tag{3}$$

The multi-domain formulation corresponding to (1) (restricted to Ω_1) - (2), is: find $\mathbf{u}_i : \Omega_i \rightarrow \mathbb{R}^2$ and $p_i : \Omega_i \rightarrow \mathbb{R}$, for $i = 1, 2$ satisfying

$$\begin{aligned}
\partial_t \mathbf{u}_i - \nu\Delta\mathbf{u}_i + (\mathbf{w}_i \cdot \nabla)\mathbf{u}_i + \nabla p_i = \mathbf{f}, &\quad \text{in } \Omega_i \times (0,T) \quad i = 1,2 && (4)\\
\nabla \cdot \mathbf{u}_i = 0 &\quad \text{in } \Omega_i \times (0,T) \quad i = 1,2 && (5)\\
\mathbf{u}_1 = \mathbf{u}_2 &\quad \text{on } \Gamma \times (0,T) && (6)\\
-p_1\mathbf{n} + \nu(\mathbf{n} \cdot \nabla)\mathbf{u}_1 = -p_2\mathbf{n} + \nu(\mathbf{n} \cdot \nabla)\mathbf{u}_2 &\quad \text{on } \Gamma \times (0,T) && (7)
\end{aligned}$$

and suitable boundary conditions on $\partial\Omega \times (0,T)$, where $\mathbf{u}_i = \mathbf{u}_{|\Omega_i}$, $p_i = p_{|\Omega_i}$, for $i = 1, 2$, and $\mathbf{n}$ denotes the normal unit vector on Γ directed from Ω_1 to Ω_2.

The choice $\mathbf{w}_1 = \mathbf{u}_1$ and $\mathbf{w}_2 = \mathbf{0}$ corresponds to a *Navier-Stokes/Stokes* coupling, while the choice $\mathbf{w}_1 = \mathbf{u}_1$ and $\mathbf{w}_2 = \mathbf{u}_{\infty|\Omega_2}$ corresponds to a *Navier-Stokes/Oseen* coupling.

The transmission conditions (6) and (7) ensure the continuity of the velocity field and the continuity of the normal stress across the interface, respectively.

For the Navier-Stokes/Oseen coupling, the transmission condition (7) can be replaced on Γ by the following one [FS98]:

$$-p_1\mathbf{n} + \nu(\mathbf{n} \cdot \nabla)\mathbf{u}_1 - \frac{1}{2}(\mathbf{w}_1 \cdot \mathbf{n})\mathbf{u}_1 = -p_2\mathbf{n} + \nu(\mathbf{n} \cdot \nabla)\mathbf{u}_2 - \frac{1}{2}(\mathbf{w}_2 \cdot \mathbf{n})\mathbf{u}_2, \tag{8}$$

and it is associated to the skew-symmetric form of the convective term in (4).

Besides, from now on, given a sufficiently regular vector field $\mathbf{w}$, we set:

$$\begin{aligned}
&T_S(\mathbf{u}, p)\mathbf{n} = -p\mathbf{n} + \nu(\mathbf{n} \cdot \nabla)\mathbf{u} && \textit{Stokes normal stress},\\
&T_O(\mathbf{w}; \mathbf{u}, p)\mathbf{n} = -p\mathbf{n} + \nu(\mathbf{n} \cdot \nabla)\mathbf{u} - \tfrac{1}{2}(\mathbf{w} \cdot \mathbf{n})\mathbf{u} && \textit{Oseen normal stress}.
\end{aligned} \tag{9}$$

The mathematical justification for the use of either (7) or (8) is provided in [FGQ99].

The time-dependent system (4)-(7) can be discretised in time, e.g., by a finite-difference scheme, so that a steady problem has to be solved at each time step. The discretisation of time derivative gives rise to a mass term with constant coefficient α dependent from the time scheme.

The boundary conditions we will consider for the coupled problem (4)-(7) will be of Dirichlet type on $\partial\Omega_D$ (e.g., no-slip boundary conditions $\mathbf{u} = \mathbf{0}$ on fixed walls, or inflow conditions $\mathbf{u} = \mathbf{g}$, for a suitable given vector field $\mathbf{g}$) and of Neumann type on $\partial\Omega_N$ (such as $T_S(\mathbf{u}, p)\mathbf{n} = \mathbf{0}$).

Dirichlet/Neumann iterations

In order to solve the multi-domain problem (4)-(7) an iterative procedure was introduced in [FGQ99], based on the solution of a sequence of boundary value problems on each subdomain, plus relaxation conditions at the interface Γ, (see [QV99], Ch. 3). In the current case, the idea consists of solving problems like (4)-(5) for every $i = 1, 2$, for which the transmission conditions (6) and (7) (or (8)) provide Dirichlet and Neumann boundary conditions on the interface Γ, respectively.

Precisely, a Dirichlet/Neumann iteration scheme for problem (4)-(5) with transmission conditions (6), (8), can be set up as follows: given $\boldsymbol{\lambda}^0$ defined on Γ, for each $k \geq 1$ find $(\mathbf{u}_1^k, p_1^k)$ such that:

$$\begin{array}{lll} \alpha\mathbf{u}_1^k - \nu\Delta\mathbf{u}_1^k + (\mathbf{w}_1^k \cdot \nabla)\mathbf{u}_1^k + \nabla p_1^k = \mathbf{f}, & \nabla \cdot \mathbf{u}_1^k = 0 & \text{in } \Omega_1 \\ \mathbf{u}_1^k = \boldsymbol{\lambda}^{k-1} & & \text{on } \Gamma \end{array} \tag{10}$$

then find $(\mathbf{u}_2^k, p_2^k)$ such that:

$$\begin{array}{lll} \alpha\mathbf{u}_2^k - \nu\Delta\mathbf{u}_2^k + (\mathbf{w}_2^k \cdot \nabla)\mathbf{u}_2^k + \nabla p_2^k = \mathbf{f}, & \nabla \cdot \mathbf{u}_2^k = 0 & \text{in } \Omega_2 \\ T_O(\mathbf{w}_2; \mathbf{u}_2^k, p_2^k)\mathbf{n} = T_O(\mathbf{w}_1^k; \mathbf{u}_1^k, p_1^k)\mathbf{n} & & \text{on } \Gamma \end{array} \tag{11}$$

where, for $k \geq 1$, the interface values are updated as follows:

$$\boldsymbol{\lambda}^k = \theta\mathbf{u}_{2|\Gamma}^k + (1-\theta)\boldsymbol{\lambda}^{k-1} \quad \text{on } \Gamma,$$

and θ is a positive relaxation parameter that will be determined in order to ensure, and possibly, to accelerate the convergence of the iterative scheme. We note that the restrictions $\mathbf{u}_{2|\Gamma}^k$ will be understood in the sense of the traces and in the linear case, (i.e. when $\mathbf{w}$ is given independently of $\mathbf{u}$), $\mathbf{w}_1^k = \mathbf{w}_1$ for all $k \geq 1$.

In the case in which the Stokes interface condition (7) is considered, the last equation in (11) is replaced by

$$T_S(\mathbf{u}_2^k, p_2^k)\mathbf{n} = T_S(\mathbf{u}_1^k, p_1^k)\mathbf{n} \quad \text{on } \Gamma. \tag{12}$$

We point out that "parallel" versions of the previous iterative schemes are obtained replacing $\mathbf{u}_1^k$ by $\mathbf{u}_1^{k-1}$ and p_1^k by p_1^{k-1} (and $\mathbf{w}_1^k$ by $\mathbf{w}_1^{k-1}$) in the last set of equations (11) (in (12)).

The convergence of this iterative scheme for a suitable range $(0, \theta^*)$ of relaxation parameters has been proven in [FGQ00], using Schauder fixed point theorem for the Steklov-Poincaré operator in the space of traces on Γ.

Numerical results

We consider the two dimensional model of pulsatile Newtonian flow in the human carotid bifurcation. This model problem is considered in biomechanical literature as a simplification of the more complex 3-D problem. The computational domain is shown in Fig. 3. The basic shape of the model agrees with the model of Bharadvaj et al. ([BMG82]) and the geometry parameters are based upon the data described by Ku et al. ([KGZG85]). Using the common carotid diameter $D = 0.62$cm as characteristic length and a reference blood viscosity $\nu = 0.035$, the maximum Reynolds number within a period of the motion is $Re_{\max} \simeq 800$. The assumed pulse frequency is 72 strokes per minute, so that the motion is periodic with period $T = 5/6$.

At the inflow boundary (the left vertical side) a fully developed time-dependent velocity profile $\mathbf{g}(x_2, t)$, such that $g_2(x_2, t) = h(x_2) \cdot \phi(t)$, is prescribed (where $\phi(t)$ is the function described in Fig. 1 (top) and $h(x_2)$ is a parabolic profile); at the rigid walls the no-slip condition $\mathbf{u} = \mathbf{0}$ is applied, while at the outflow boundary a no-friction condition is imposed (i.e. $T_S(\mathbf{u}, p)\mathbf{n} = \mathbf{0}$).

The two-domains formulation (10)-(11) is here extended to four subdomains (see Fig. 2): one Navier-Stokes domain and three Oseen domains with $\mathbf{u}_\infty(t) = \mathbf{u}_{Stokes}(t)$, that is the Stokes solution subjected to the fully developed time-dependent velocity profile $\mathbf{g}(x_2, t)$. The Euler Semi-Implicit (ESI) finite difference scheme is used to discretise the time derivative, with $\Delta t = 10^{-2}$. At each time step of the ESI scheme, we make use of the Dirichlet/Neumann algorithm. The relaxation parameter θ was chosen dinamically so as to minimize the interface error at each D/N step. In order to test the convergence of the D/N algorithm we check that

$$\max_{i=1,2} \left[\|\mathbf{u}_i^k - \mathbf{u}_i^{k-1}\|_{H^1(\Omega_i)} / \|\mathbf{u}_i^k\|_{H^1(\Omega_i)} \right] \leq 5 \cdot 10^{-6},$$

where k is the iteration counter. The numerical approximation is carried out by considering stabilised Spectral Element Methods, with 25 elements and polynomial degree $N = 5$.

In Fig. 1 the two components of the velocity are shown for the full Navier-Stokes approximation and the NS/OS coupling with either Oseen flux or Stokes flux across the interfaces. Note that the coupling based on the Stokes flux at the interfaces provide a much more accurate solution, as already noticed in [FGQ99] for other problems.

In Fig. 2 (bottom) we show the number of D/N iterations needed to converge, at each time step, for the NS/OS coupling with either Oseen or Stokes flux across the interfaces.

In Fig. 3 we report the relative errors between the NS/OS ($\mathbf{u}_{NS/OS}$) and the full Navier-Stokes ($\mathbf{u}_{NS}$) solution for the two different decompositions illustrated in Fig. 4. We denote by Ω_0 the domain of the left decomposition of Fig. 4 in which Navier-Stokes equations are solved, and we define the error as:

$$e_{H^1(\Omega_0)} = \frac{\|\mathbf{u}_{NS} - \mathbf{u}_{NS/OS}\|_{H^1(\Omega_0)}}{\|\mathbf{u}_{NS}\|_{H^1(\Omega_0)}}.$$

As expected, the second partition, featuring Navier-Stokes subdomain larger than in the first one, provides more accurate results.

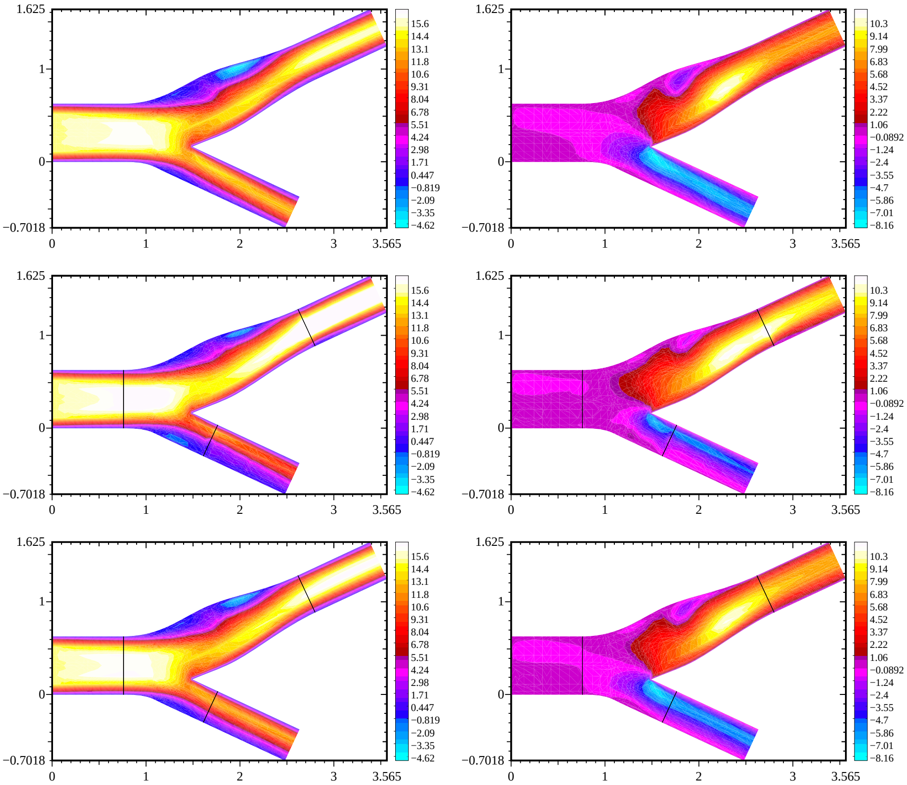


Figure 1: First (left) and second (rigth) components of the velocity for the full Navier-Stokes solution (top), the NS/OS coupling with Oseen flux at the interfaces (intermediate), the NS/OS coupling with Stokes flux at the interfaces (bottom). The results refer to $t = .3$ when the difference between the full Navier-Stokes solution and the coupled NS/OS solution is maximum.

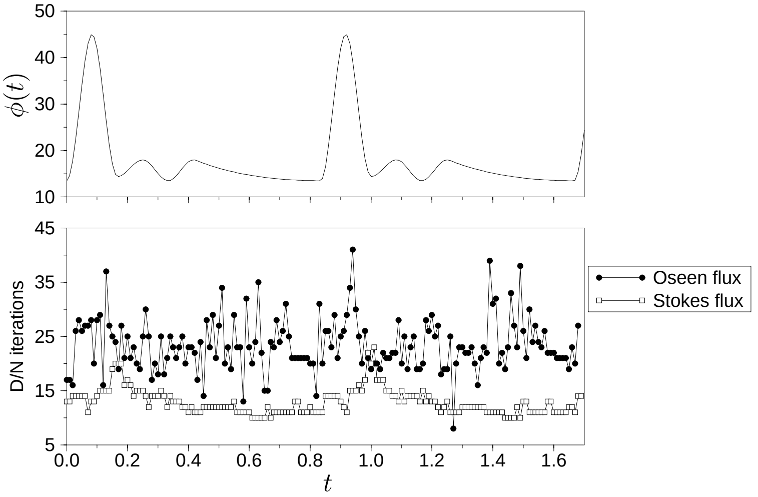


Figure 2: The fully developed time-dependent velocity profile (top) and the D/N iterations for the coupling with either Oseen or Stokes flux.

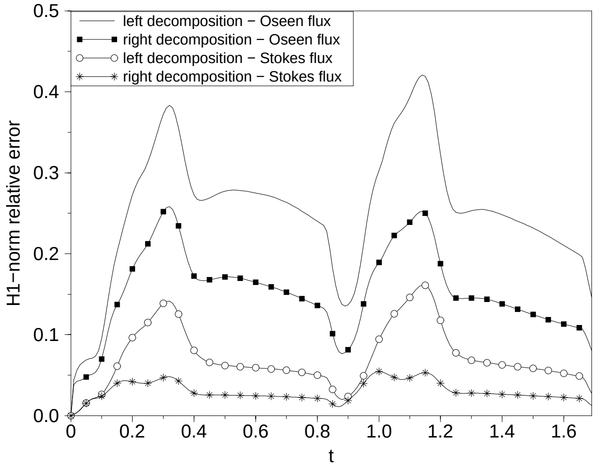


Figure 3: The errors $e_{H^1(\Omega_0)}$ between the NS/OS coupling and the full Navier-Stokes solution, with either Oseen or Stokes flux across the interfaces.

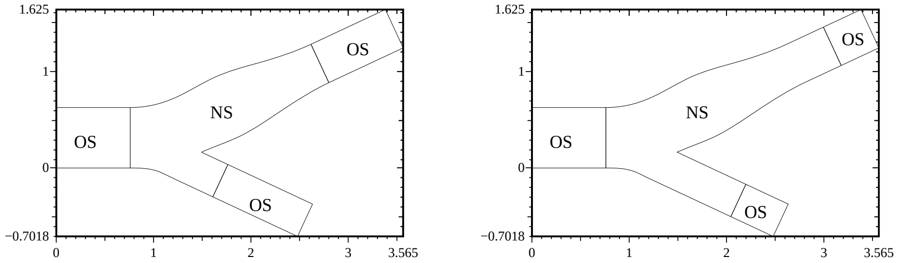


Figure 4: The two decompositions used for the error analysis of Fig. 3.

References

[BMG82]B.K. Bharadvaj, R.F. Mabon, and D.P. Giddens. Steady flow in a model of the human carotid bifurcation. part i - flow visualisation. *J. Biomechanics*, 15:349–362, 1982.

[FGQ99]L. Fatone, P. Gervasio, and A. Quarteroni. Multimodels for incompressible flows. Technical Report 09.99, EPFL, Lausanne (CH), 1999. Accepted for publication in JMFM, Vol.2 (2000), no. 2.

[FGQ00]L. Fatone, P. Gervasio, and A. Quarteroni. Multimodels for incompressible flows: iterative solutions for the Navier-Stokes/Oseen coupling. Technical Report n. 15/2000, Seminario Matematico di Brescia, Brescia, (Italy), 2000.

[FS98]M. Feistauer and C. Schwab. Coupling of an interior Navier-Stokes problem with an exterior Oseen problem. Technical Report 98/01, ETH, Zurich (CH), 1998.

[KGZG85]D.N. Ku, D.P. Giddens, C.Z. Zarins, and S. Glagov. Pulsatile flow and atherosclerosis in the human carotid bifurcation. *Arteriosclerosis*, 5:293–302, 1985.

[QV99]A. Quarteroni and A. Valli. *Domain Decomposition Methods for Partial Differential Equations.* Oxford Science Publications, 1999.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau, ©2001 DDM.org

33. Domain decomposition methods for a coupled vibration between an acoustic field and a plate

Xiaobing Feng[1], Zhenghui Xie[2]

Introduction

The coupled vibration between an acoustic field and a plate is encountered in many engineering and industrial applications. The interaction between the wind and a windshield of a car is an interesting example found in the automobile industry. Mathematically, such an interaction is described by the coupled system of the second order scalar wave equation and the fourth order plate vibration equation. Since the thickness of the plate is negligible, the plate serves a dual role in the model. It is the solid medium and in the same time it is the interface between the acoustic field and the solid (so it is a part of the boundary of the acoustic field), where they interact each other.

Let $\Omega \subset R^3$ be a three-dimensional acoustic field and $\Gamma_0 \subset R^2$, a part of the boundary $\partial\Omega$, denote the domain of the plate. Let $\Gamma_1 = \partial\Omega\backslash\Gamma_0$ be the remaining portion of the boundary of Ω. Let $p = p(x_1, x_2, x_3)$ denote the pressure function of the fluid in the acoustic field Ω and $u = u(x')$ $(x' = (x_1, x_2)^t)$ denote the vertical displacement of the plate Γ_0. Then the governing partial differential equations of the fluid–plate interaction is given by [CS76]

$$\begin{aligned}
\tfrac{1}{c^2}p_{tt} - \Delta p = f, & \qquad \text{in } \Omega \times (0,T), & (1)\\
\tfrac{1}{c}p_t + \tfrac{\partial p}{\partial n} = 0, & \qquad \text{on } \Gamma_1 \times (0,T), & (2)\\
\tfrac{\partial p}{\partial n} + \rho_f u_{tt} = 0, & \qquad \text{on } \Gamma_0 \times (0,T), & (3)\\
\rho_s u_{tt} + D\Delta^2_{\Gamma_0} u = p, & \qquad \text{on } \Gamma_0 \times (0,T), & (4)\\
u = \tfrac{\partial u}{\partial \nu} = 0, & \qquad \text{on } \partial\Gamma_0 \times (0,T), & (5)\\
p(x,0) = p_0(x), \quad p_t(x,0) = p_1(x), & \qquad \text{in } \Omega, & (6)\\
u(x',0) = u_0(x'), \quad u_t(x',0) = u_1(x'), & \qquad \text{on } \Gamma_0, & (7)
\end{aligned}$$

where c is the sound speed in the fluid, D flexural rigidity of plate. ρ_f and ρ_s are the air mass density and plate mass density, and n and ν are the outward normal vector on Γ_0 and $\partial\Gamma_0$, respectively. $\Delta^2_{\Gamma_0}$ stands for the biharmonic operator defined on Γ_0 in variables x_1, x_2.

In the model, equations (3) and (4) are the interface condition which describe the interaction between the acoustic field and the plate. Equation (2) is the first order absorbing boundary condition for the acoustic wave. We use this boundary condition,

[1] Department of Mathematics, The University of Tennessee, Knoxville, TN 37996, U.S.A. xfeng@math.utk.edu

[2] LASG, Institute of Atmospheric Physics, Chinese Academy of Sciences, Beijing 100080, China. xzh@lasgsgi4.iap.ac.cn

instead of the Dirichlet condition as used in [CS76], to limit the (computational) size of the acoustic domain. Using the energy method it is not hard to show the following theorem.

Theorem 1 *For $f \in H^{-1}(\Omega)$, $p_j \in H^{1-j}(\Omega)$, and $u_j \in H^{2-j}(\Gamma_0)$, $j = 0, 1$, the problem (1)–(7) has a unique solution $(p, u) \in L^2(H^1(\Omega)) \cap H^1(L^2(\Omega)) \times L^2(H^2(\Gamma_0)) \cap H^1(H^1(\Gamma_0))$.*

The goal of this paper is to develop some parallelizable non–overlapping domain decomposition iterative methods for effectively solving the problem (1)–(7). Due to the heterogeneous nature of the problem, the non–overlapping domain decomposition approach is a very practical and natural way to solve the problem. In §2 we introduce two classes of domain decomposition iterative methods to decouple the problem into fluid and plate subdomain problems. In §3 we establish usefulness of these methods by showing their strong convergence in the energy norm of the underlying problem. Finally, in §4 we present some numerical models based on finite difference methods, and some numerical tests to validate the theory and to show effectiveness of the methods, in particular, with respect to different choices of the relaxation parameter.

Domain decomposition methods

In this section we first propose a family of new interface conditions which are equivalent to the original interface condition (3). This is the key step towards developing non-overlapping domain decomposition methods for the problem. Based on these new interface conditions, we then introduce two classes of parallelizable non-overlapping domain decomposition iterative algorithms for solving the system (1)–(7) and show their strong convergence in the energy norm of the underlying interaction problem. The methods and the analysis of this paper are inspired by its companion paper [Fen98], where non-overlapping domain decomposition methods were developed for a general fluid–solid interaction problem in which the solid is a 3–dimensional elastic body. For applications of domain decomposition methods to other heterogeneous problem, we refer to [CF99, Fen98, QPV92] and references therein.

To decouple the problem on the interface, we rewrite the interface condition (3) as

$$\frac{\partial p}{\partial n} + \alpha p_t = -\rho_f u_{tt} + \alpha p_t, \qquad \text{on } \Gamma_0 \times (0, T), \tag{8}$$

for any nonzero constant α.

Hence, the problem (1)–(7) is equivalent to the problem consisting equations (1), (2), (8), and (4)–(7). That is,

$$\frac{1}{c^2}p_{tt} - \Delta p = f, \qquad \text{in } \Omega \times (0,T), \tag{9}$$
$$\frac{1}{c}p_t + \frac{\partial p}{\partial n} = 0, \qquad \text{on } \Gamma_1 \times (0,T), \tag{10}$$
$$\frac{\partial p}{\partial n} + \alpha p_t = -\rho_f u_{tt} + \alpha p_t, \qquad \text{on } \Gamma_0 \times (0,T), \tag{11}$$
$$\rho_s u_{tt} + D\Delta^2_{\Gamma_0} u = p, \qquad \text{on } \Gamma_0 \times (0,T), \tag{12}$$
$$u = \frac{\partial u}{\partial \nu} = 0, \qquad \text{on } \partial\Gamma_0 \times (0,T), \tag{13}$$
$$p(x,0) = p_0(x), \quad p_t(x,0) = p_1(x), \qquad \text{in } \Omega, \tag{14}$$
$$u(x',0) = u_0(x'), \quad u_t(x',0) = u_1(x'), \qquad \text{on } \Gamma_0. \tag{15}$$

Domain decomposition algorithms

Based on the above new form of the interface conditions we propose the following two types of iterative algorithms. The first one resembles to block Gauss-Seidel iteration and the other resembles block Jacobi iteration.

Algorithm 1

Step 1: $\forall p^0 \in H^1(L^2(\Gamma_0))$.

Step 2: Compute $\{(u^k, p^k)\}_{k\geq 1}$ by solving

$$\rho_s u^k_{tt} + D\Delta^2_{\Gamma_0} u^k = p^{k-1}, \qquad \text{on } \Gamma_0 \times (0,T), \tag{16}$$
$$u^k = \frac{\partial u^k}{\partial \nu} = 0, \qquad \text{on } \partial\Gamma_0 \times (0,T), \tag{17}$$
$$u^k(x',0) = u_0(x'), \quad u^k_t(x',0) = u_1(x'), \qquad \text{on } \Gamma_0 \times (0,T); \tag{18}$$
$$\frac{1}{c^2}p^k_{tt} - \Delta p^k = f, \qquad \text{in } \Omega \times (0,T), \tag{19}$$
$$\frac{\partial p^k}{\partial n} + \alpha p^k_t = -\rho_f u^k_{tt} + \alpha p^{k-1}_t, \qquad \text{on } \Gamma_0 \times (0,T), \tag{20}$$
$$\frac{\partial p^k}{\partial n} + \frac{1}{c}p^k_t = 0, \qquad \text{on } \Gamma_1 \times (0,T), \tag{21}$$
$$p^k(x,0) = p^k_0(x), \quad p^k_t(x,0) = p_1(x), \qquad \text{in } \Omega. \tag{22}$$

Algorithm 2

Step 1: $\forall H^2(L^2(\Gamma_0)) \times H^1(L^2(\Gamma_0))$.

Step 2: Compute $\{(u^k, p^k)\}_{k\geq 1}$ by solving

$$\rho_s u^k_{tt} + D\Delta^2_{\Gamma_0} u^k = p^{k-1}, \qquad \text{on } \Gamma_0 \times (0,T), \tag{23}$$
$$u^k = \frac{\partial u^k}{\partial \nu} = 0, \qquad \text{on } \Gamma_0 \times (0,T), \tag{24}$$
$$u^k(x',0) = u_0(x'), \quad u^k_t(x',0) = u_1(x'), \qquad \text{on } \Gamma_0 \times (0,T); \tag{25}$$
$$\frac{1}{c^2}p^k_{tt} - \Delta p^k = f, \qquad \text{in } \Omega \times (0,T), \tag{26}$$
$$\frac{\partial p^k}{\partial n} + \alpha p^k_t = -\rho_f u^{k-1}_{tt} + \alpha p^{k-1}_t, \qquad \text{on } \Gamma_0 \times (0,T), \tag{27}$$
$$\frac{\partial p^k}{\partial n} + \frac{1}{c}p^k_t = 0, \qquad \text{on } \Gamma_1 \times (0,T), \tag{28}$$
$$p^k(x,0) = p_0(x), \quad p^k_t(x,0) = p_1(x), \qquad \text{in } \Omega. \tag{29}$$

Convergence analysis

In this subsection we shall establish the utility of Algorithm 1 and 2 by showing their strong convergence. Since the proof of the convergence of Algorithm 1 and 2 are similar, we only give the proof for Algorithm 1.

Define the error functions $e^k = p - p^k, r^k = u - u^k$. It follows from (1)–(7) and (9)–(15) that

$$\rho_s r^k_{tt} + D\Delta^2_{\Gamma_0} r^k = e^{k-1}, \qquad \text{on } \Gamma_0 \times (0,T), \tag{30}$$
$$r^k = \tfrac{\partial r^k}{\partial \nu} = 0, \qquad \text{on } \Gamma_0 \times (0,T), \tag{31}$$
$$r^k(x',0) = r^k_t(x',0) = 0, \qquad \text{in } \Omega \times (0,T); \tag{32}$$
$$\tfrac{1}{c^2} e^k_{tt} - \Delta e^k = 0, \qquad \text{in } \Omega \times (0,T), \tag{33}$$
$$\tfrac{\partial e^k}{\partial n} + \alpha e^k_t = -\rho_f r^k_{tt} + \alpha e^{k-1}_t, \qquad \text{on } \Gamma_0 \times (0,T), \tag{34}$$
$$\tfrac{\partial e^k}{\partial n} + \tfrac{1}{c} e^k_t = 0, \qquad \text{on } \Gamma_1 \times (0,T), \tag{35}$$
$$e^k(x,0) = e^k_t(x,0) = 0, \qquad \text{in } \Omega. \tag{36}$$

Lemma 1 *For* $\forall \tau \in (0,T]$, *we have*

$$\textstyle\int_0^\tau \int_{\Gamma_0} e^{k-1}_t r^k_{tt} dsdt = \frac{1}{2}[\|\sqrt{\rho_s} r^k_{tt}(\cdot,\tau)\|^2_{0,\Gamma_0} + \|\sqrt{D}\Delta_{\Gamma_0} r^k_t(\cdot,\tau)\|^2_{0,\Gamma_0}]. \tag{37}$$

$$\textstyle\int_0^\tau \int_{\Gamma_0} \frac{\partial e^k}{\partial n} e^k_t dsdt = \frac{1}{2}\left[\left\|\frac{1}{c} e^k_t(\cdot,\tau)\right\|^2_{0,\Omega} + \|\nabla e^k(\cdot,\tau)\|^2_{0,\Omega}\right] + \int_0^\tau \left\|\frac{1}{\sqrt{c}} e^k_t(\cdot,t)\right\|^2_{0,\Gamma_1} dt \tag{38}$$

Proof: Testing (30)) against r^k_{tt} after taking one derivative with respect to t, we get

$$\int_{\Gamma_0} e^{k-1}_t r^k_{tt} dsdt = \frac{1}{2}\frac{d}{dt}\|\sqrt{\rho_s} r^k_{tt}\|^2_{0,\Gamma_0} + \frac{1}{2}\frac{d}{dt}\left\|\sqrt{D}\Delta_{\Gamma_0} r^k_t\right\|^2_{0,\Gamma_0}. \tag{39}$$

Integrating (39) in t from 0 to τ yields (37).

Similarly, test (33) against e^k_t, we get

$$\int_{\Gamma_0} \frac{\partial e^k}{\partial n} e^k_t ds = \frac{1}{2}\frac{d}{dt}\left[\left\|\frac{1}{c} e^k_t\right\|^2_{0,\Omega} + \left\|\nabla e^k\right\|^2_{0,\Omega}\right] + \left\|\frac{1}{\sqrt{c}} e^k_t\right\|^2_{0,\Gamma_1}. \tag{40}$$

Integrating (40) in t from 0 to τ gives (38).

Notice that in the proof we have used the fact that

$$r^k(\cdot,0) = r^k_t(\cdot,0) = r^k_{tt}(\cdot,0) = \Delta_{\Gamma_0} r^k(\cdot,0) = e^k_t(\cdot,0) = e^k(\cdot,0) = 0, \ \nabla e^k(\cdot,0) = 0.$$

Next, define the "pseudo-energy"

$$E_k(\tau) \equiv \left\|\frac{\partial e^k}{\partial n} + \alpha e^k_t\right\|^2_{L^2((0,\tau),L^2(\Gamma_0))} = \int_0^\tau \int_{\Gamma_0} \left[\frac{\partial e^k}{\partial n} + \alpha e^k_t\right]^2 dsdt. \tag{41}$$

By a direct calculation, we can show that $\{E_k(\tau)\}$ satisfy the following identity.

Lemma 2 *For $k \geq 1$ there holds the following identity*

$$E_k(\tau) = E_{k-1}(\tau) - R_{k-1}(\tau), \tag{42}$$

where

$$\begin{aligned} R_{k-1}(\tau) \quad = \quad & \left\| \frac{\partial e^{k-1}}{\partial n} \right\|^2_{L^2((0,\tau),L^2\Gamma_0))} - \rho_f^2 \|r_{tt}^k\|^2_{L^2((0,\tau),L^2(\Gamma_0))} \\ & +2\alpha \int_0^\tau \int_{\Gamma_0} \frac{\partial e^{k-1}}{\partial n} e_t^{k-1} ds dt + 2\alpha\rho_f \int_0^\tau \int_{\Gamma_0} e_t^{k-1} r_{tt}^k ds dt. \end{aligned} \tag{43}$$

An immediate consequence of Lemma 1 is the following lemma.

Lemma 3 *For $k \geq 1$ there holds the equality*

$$\begin{aligned} R_{k-1}(\tau) \quad = \quad & [\alpha\rho_f \|\sqrt{\rho_s} r_{tt}^k(\cdot,\tau)\|^2_{0,\Gamma_0} - \rho_f^2 \|r_{tt}^k\|^2_{L^2((0,\tau);L^2(\Gamma_0))}] \\ & +\alpha \left[\rho_f \|\sqrt{D}\Delta_{\Gamma_0} r_t^k(\cdot,\tau)\|^2_{0,\Gamma_0} + \left\| \frac{1}{c} e_t^k(\cdot,\tau) \right\|^2_{0,\Omega} + \|\nabla e^k(\cdot,\tau)\|^2_{0,\Omega} \right] \\ & + \left\| \frac{\partial e^{k-1}}{\partial n} \right\|^2_{L^2((0,\tau);L^2(\Gamma_0))} + \alpha \int_0^\tau \left\| \frac{1}{\sqrt{c}} e_t^k(\cdot,t) \right\|^2_{0,\Gamma_1} dt. \end{aligned} \tag{44}$$

Theorem 2 *If $\alpha > T\rho_f/\rho_s$, then*
(1) $p^k \to p$ strongly in $L^2((0,T);H^1(\Omega)) \cap H^1((0,T);L^2(\Omega))$.
(2) $u^k \to u$ strongly in $H^1((0,T);H^2(\Gamma_0)) \cap H^2((0,T);L^2(\Gamma_0))$.

Proof: It is easy to check that (42) implies that

$$\int_0^T E_k(\tau) d\tau = \int_0^T E_0(\tau) d\tau - \sum_{l=0}^{k-1} \int_0^T R_l(\tau) d\tau. \tag{45}$$

Since

$$\int_0^T \|r_{tt}^k\|^2_{L^2((0,\tau);L^2(\Gamma_0))} d\tau = \int_0^T \left(\int_0^\tau \|r_{tt}^k(\cdot,t)\|^2_{0,\Gamma_0} dt \right) d\tau \leq T \|r_{tt}^k\|^2_{L^2((0,T);L^2(\Gamma_0))}, \tag{46}$$

we have

$$\begin{aligned} & \textstyle\int_0^T \left[\alpha\rho_f \|\sqrt{\rho_s} r_{tt}^k(\cdot,\tau)\|^2_{0,\Gamma_0} - \rho_f^2 \|r_{tt}^k\|^2_{L^2((0,\tau);L^2(\Gamma_0))} \right] d\tau \\ & \qquad \geq \rho_f(\alpha\rho_s - T\rho_f) \|r_{tt}^k\|^2_{L^2((0,T);L^2(\Gamma_0))}. \end{aligned} \tag{47}$$

Hence, if $\alpha > T\rho_f/\rho_s$, every term on the right hand side of (43) is a nonnegative term. Now it follows from (45) that

$$\sum_{l=0}^{\infty} \int_0^T R_l(\tau) d\tau < \infty,$$

which implies that

$$\lim_{l\to\infty}\int_0^T R_l(\tau)d\tau = 0. \tag{48}$$

Finally, the proof is completed by combining (44), (47) and (48).

Numerical experiments

We shall present some numerical tests for the domain decomposition algorithms developed in the previous sections. Finite difference methods are used to discretize the differential equations. The acoustic field is chosen as the unit cubic $\Omega = [0,1]^3$ and the plate domain is the unit square $\Gamma_0 = [0,1]^2$ on the x_1x_2- plane. Zero source function $f \equiv 0$ and the parameters $c = 2.5, D = 2, \rho_f = 5, \rho_s = 50$ are assumed in all tests. Also, the uniform meshes are used in both acoustic domain and the plate domain. The mesh size of the acoustic domain is $\Delta x_1 = \Delta x_2 = \Delta x_3 = 0.1$ and the mesh size of the the plate domain is $\Delta x_1 = \Delta x_2 = 0.05$. The time step size $\Delta t = 0.01$ is used in all tests. Finally, we choose the following initial conditions.

$$p_0(x) = 1, \quad p_1(x) = 0.1, \quad u_0(x_1,x_2) = \sin\pi x_1 \sin\pi x_2, \quad u_1(x_1,x_2) = 0.1.$$

Figure 1 shows the plate displacement (u) profiles at four different time steps, in which (a)–(d) are plots of u at $t = 4\Delta t, 8\Delta t, 12\Delta t, 16\Delta t$, respectively. Figure 2 gives the pressure (p) profiles on the interface $x_3 = 0$ at (a) $t = 4\Delta t$, (b) $t = 8\Delta t$, (c) $t = 12\Delta t$, (d) $t = 16\Delta t$, which show the acoustic wave action on the plate. Figure 3 shows the contour plots of the pressure p on the cross section of Ω at $x_1 = 0.5$ at (a) $t = 4\Delta t$, (b) $t = 8\Delta t$, (c) $t = 12\Delta t$, (d) $t = 16\Delta t$. Figure 4 presents a comparison of the iteration numbers for different choices of the relaxation parameter α at various time steps. Graph (a) compares the iteration numbers for $\alpha = 10^{-9}$ and $\alpha = 10$, while Graph (b) compares for $\alpha = 1$ and $\alpha = 100$. The criterion used to stop the domain decomposition iteration at all time steps is that the relative error of successive iterates should be less than 10^{-3}. These comparisons suggest that the algorithms perform better with large relaxation parameter α, which is predicted by the convergence analysis. It is also interesting to note that for a fixed α the number of iterations required at different time steps varies significantly. We believe that this is caused mainly by the fact that the solution varies significantly at different time steps.

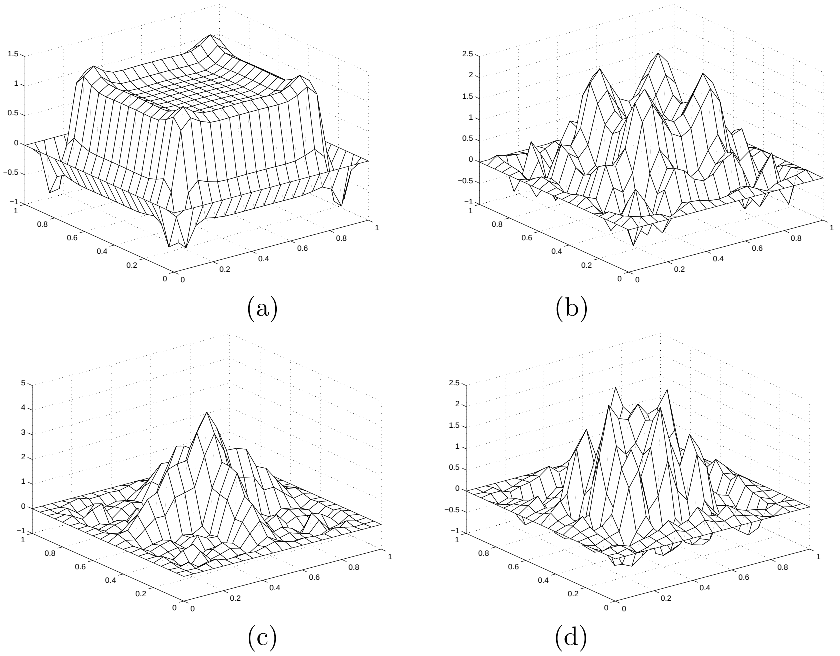
References

[CF99]P. Cummings and X. Feng. Domain decomposition methods for a system of coupled acoustic and elastic Helmholtz equations. In C-H. Lai, P. Bjørstad, M. Cross, and O Widlund, editors, *Eleventh International Conference on Domain Decomposition Methods*, pages 203–210, Bergen, Norway, 1999. Domain Decomposition Press.

[CS76]A. Craggs and G. Stead. Sound transmission between enclosures - a study using plate and acoustic finite elements. *Acustica*, 35:89–98, 1976.

[Fen98]X. Feng. Interface conditions and non-overlapping domain decomposition methods for a fluid-solid interface problem. In J. Mandel, C. Farhat, and X.-C. Cai, editors, *Tenth International Conference on Domain Decomposition Methods*, pages 417–424. AMS, Contemporary Mathematics 218, 1998.

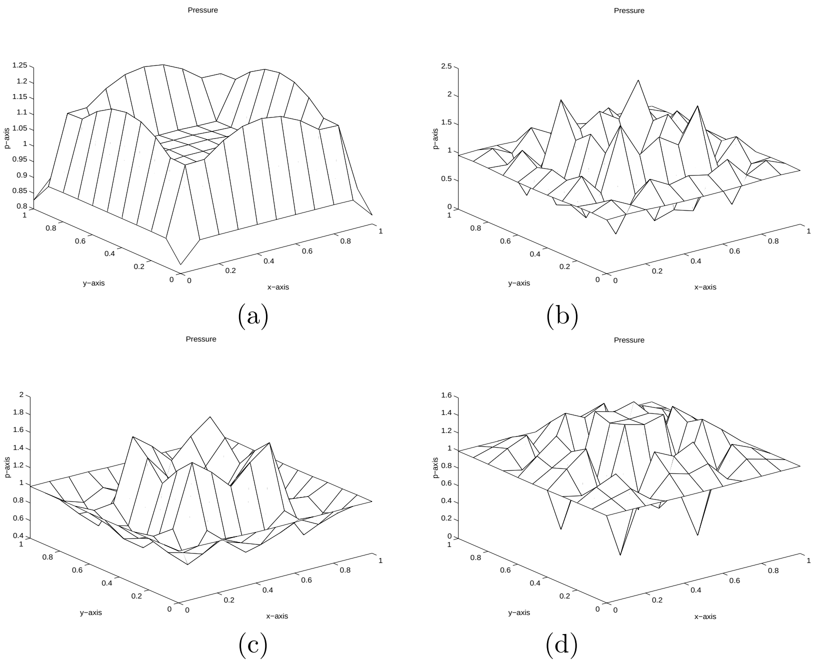
[QPV92]A. Quarteroni, F. Pasquarelli, and A. Valli. Heterogeneous domain decomposition principles, algorithms, applications. In David E. Keyes, Tony F. Chan, Gérard A. Meurant, Jeffrey S. Scroggs, and Robert G. Voigt, editors, *Fifth International Symposium on Domain Decomposition Methods for Partial Differential Equations*, pages 129–150, Philadelphia, PA, 1992. SIAM.



(a) (b)

(c) (d)

Figure 1: *Plate displacement profiles at different time steps*



(a) (b)

(c) (d)

Figure 2: *Pressure profiles on the interface at different time steps*

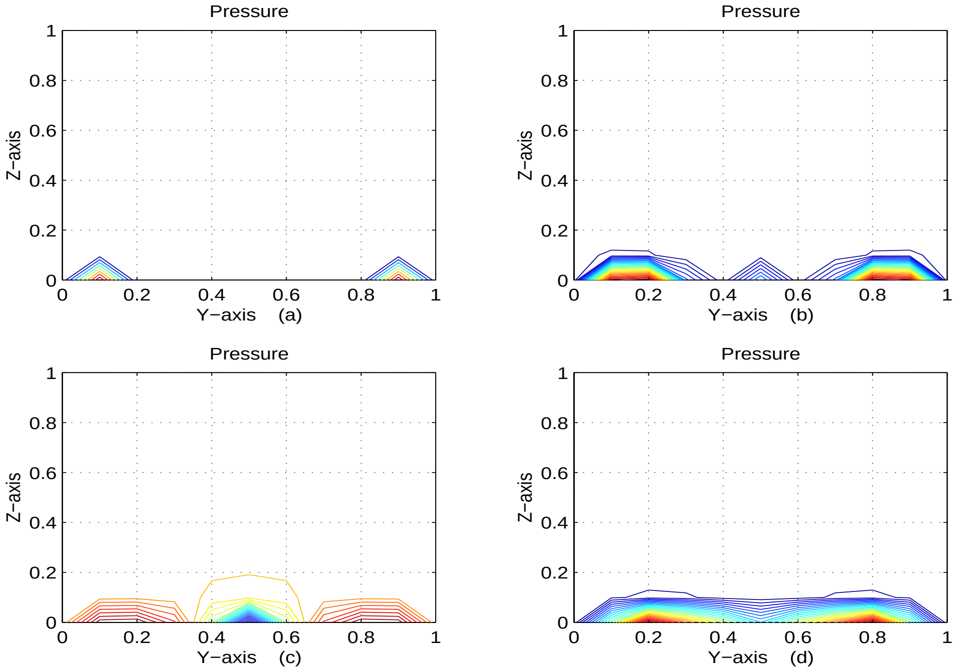
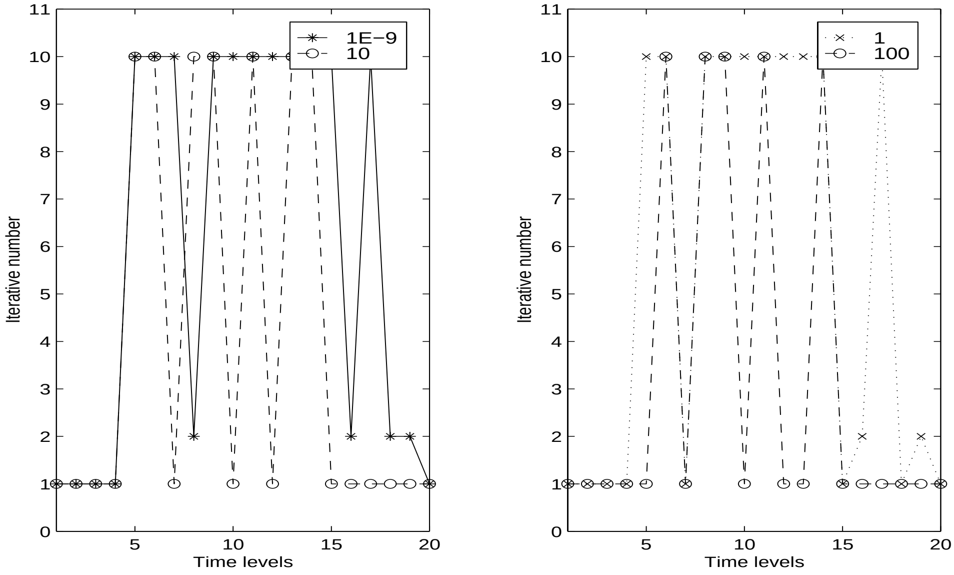


Figure 3: *Contour plots of the pressure p on the cross section $x_1 = 2$ at different time steps*



(a) (b)

Figure 4: Comparison of the iteration numbers for different value α at various time steps

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau,

34. Domain Decomposition for Kalman Filter Method and Its Application to Tidal Flow at Onjuku Coast

Maki FUJIMOTO [1], Mutsuto KAWAHARA [2]

Introduction

Recently remarkable progress has been done in the civil engineering field and many hi-level public works have been carried out. With these progress, numerical analysis in the field of civil engineering also advances more and more, and it plays an important role in the building of many kinds of structures. To grasp actual phenomenon about civil engineering more faithfully and briefly by using numerical analysis, it is necessary to further make progress customary analysis procedure. We have been studying tidal flow using Kalman filter method. But we have a serious problem that Kalman filter using FEM cannot be computed about problem with huge analytical domain. Then we decide to apply domain decomposition for Kalman filter with FEM. Using domain decomposition, finite element matrix becomes smaller and memory is economized. Computational time becomes very short and numerical examples with small mesh can be calculated. Kalman filter using FEM with many nodes could be calculated. Schwarz Alternating Procedure (SAP),one of domain decomposition technique, is used in this case. SAP is easy to treat and to apply for Kalman filter.

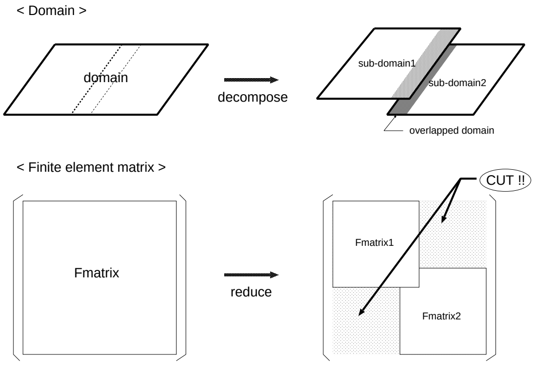


Figure 1: Image of domain decomposition

In this analysis, applying domain decomposition for actual problem of tidal flow, we verify effectiveness of this technique.

[1] Civil Engineering, Chuo University
[2] Civil Engineering, Chuo University

Finite Element Method

Basic Equations

Linear shallow water equations, which consist of momentum equation and continuity equation, are shown as follows ;

$$\dot{u}_i + g\eta_{,i} = 0, \tag{1}$$

$$\dot{\eta} + hu_{i,i} = 0, \tag{2}$$

where u_i are velocities, η is water elevation , h is water depth and g is the gravity acceleration.

Boundary Conditions

The boundary of water region consists of two parts S_1 and S_2. S_1 is the land boundary and S_2 is the open boundary. The boundary conditions are shown as follows ;

$$u_n = ul + vm = \hat{u}_n \qquad \text{on } S_1, \tag{3}$$

$$\eta = \hat{\eta} \qquad \text{on } S_2, \tag{4}$$

where we use ˆ to represent the given values.

Finite Element Equations

To obtain finite element equations, the shallow water equations can be discretized spatially and temporally. As regards spatial directions, Galerkin method with triangles finite element is used. As regards numerical integration in time, explicit method with BTD term is used. Then finite element equations can be obtained in the following form.

$$\bar{M}_{\alpha\beta}u_\beta^{n+1} = M_{\alpha\beta}u_\beta^n - \Delta t\{gS_{\alpha\beta,x}\eta_\beta^n + \frac{\Delta t}{2}gh(H_{\alpha,x\beta,x}u_\beta^n + H_{\alpha,x\beta,y}v_\beta^n)\}, \tag{5}$$

$$\bar{M}_{\alpha\beta}v_\beta^{n+1} = M_{\alpha\beta}v_\beta^n - \Delta t\{gS_{\alpha\beta,y}\eta_\beta^n + \frac{\Delta t}{2}gh(H_{\alpha,y\beta,x}u_\beta^n + H_{\alpha,y\beta,y}v_\beta^n)\}, \tag{6}$$

$$\bar{M}_{\alpha\beta}\eta_\beta^{n+1} = M_{\alpha\beta}\eta_\beta^n - \Delta t\{h(S_{\alpha\beta,x}u_\beta^n + S_{\alpha\beta,y}v_\beta^n) + \frac{\Delta t}{2}gh(H_{\alpha,x\beta,x}\eta_\beta^n + H_{\alpha,y\beta,y}\eta_\beta^n)\}, \tag{7}$$

where BTD(Balancing Tensor Diffusivity) term is numerical viscosity.

Kalman Filter

Basic Equations

The basic equations of Kalman filter are shown as follows;

$$x_{k+1} = F_k x_k + G_k w_k, \tag{8}$$

$$y_k = H_k x_k + v_k. \tag{9}$$

Eq.8 and Eq.9 are called system equation and observation equation respectively. Where x_k is the state vector, y_k is observed vector, F_k is state transition matrix that is called as finite element matrix, H_k is the observation matrix which has information about placement of observation points, G_k is driving matrix, w_k is the system noise and v_k is the observation noise. w_k and v_k are the white noise which are independent each other.

Domain Decomposition Technique

An Introduction to Schwarz Alternating Procedure(SAP)

Schwarz Alternating Procedure(SAP) is one of many Domain Decomposition Techniques. The decomposed domain with overlapping area is performed in this method. In order to define the SAP algorithm, computational domain Ω is decomposed into some overlapping subdomains. It is assumed that Ω is partitioned into $J \geq 2$ intersecting subdomains which can be written as follow;

$$\Omega = \bigcup_{k=1}^{J} \Omega^k, \tag{10}$$

where Ω^k is subdomain. The overlapping area is shown by $\Omega^k \cap \Omega^{k+1}$ for $k = 1, 2, \cdots, (J-1)$. And the boundaries of Ω^k are defined as;

$$\Gamma^k = \partial\Omega^k \cap \Gamma, \quad k = 1, 2, \cdots, J \tag{11}$$

where $\partial\Omega^k$ is whole boundary of subdomain Ω^k, Γ^k is internal boundary which is written in another way from Eq.(9);

$$\Gamma^k = \partial\Omega^k \backslash (\partial\Omega^k \cap \partial\Omega^{k+1}), \quad k = 1, 2, \cdots, J,$$

and Γ is a set of internal boundary Γ^k. This is written as follow;

$$\Gamma = \bigcup_{k=1}^{J} \Gamma^k. \tag{12}$$

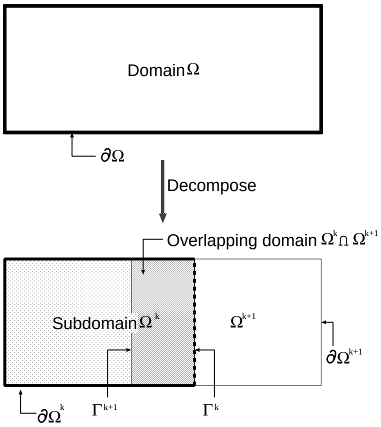


Figure 2: Model of Decomposed Domain

Application for Finite Element Method

If Domain Decomposition Technique with SAP, which was explained above, was applied for Finite Element Method, basic equation could be written as follows;

$$\dot{u}_i^k + g\eta_{,i}^k = 0, \quad \dot{\eta}^k + hu_{i,i}^k = 0 \qquad \text{in } \Omega^k$$

$$u_n^k = u^k l + v^k m = \hat{u}_n^k \qquad \text{on } \partial\Omega_1^k,$$

$$\eta = \hat{\eta}^k \qquad \text{on } \partial\Omega_2^k,$$

$$k = 1, 2, \cdots, J.$$

In order to obtain finite element equations about this case, these equations are discretized in each domain. And using finite element equations that are obtained by the discretization, we carry out numerical analysis.

Numerical Example

Case 1

Computational domain is Onjuku Coast in Chiba Prefecture. This domain is divided into 300 nodes and 526 elements. In this analysis, two types of examples computed and these results are compared. One is the case where the analytical domain is decomposed into 2 sub-domains and the other is that not using SAP. Time increment Δt is 0.0020, system noise Q is 0.020 and observation noise R is 0.0010. Calculation is carried out by using observation data from July 16, 1997 to July 20 at 5 observation points.

In this case, computational domain decomposed into 2 subdomains with overlapping area is used. Sub-domain 1 has 169 nodes and sub-domain 2 has 173 nodes.

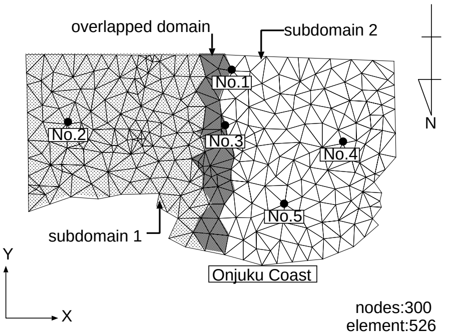


Figure 3: Computational domain of case 1

Algorithm of Case 1

Kalman filter with FEM applying SAP is calculated as following algorithm:

1. Kalman-gain is calculated about sub-domain 1 and converged.
2. Kalman-gain is calculated about sub-domain 2 and converged.
3. Using Kalman-gain converged at 1., optimal estimated value is computed about sub-domain1.
4. Using Kalman-gain converged at 2., optimal estimated value is computed about sub-domain2.
5. In overlapping area, results at 3. and 4. are compared.
   If difference between these results is not small enough, go to 3..
6. Results are written.

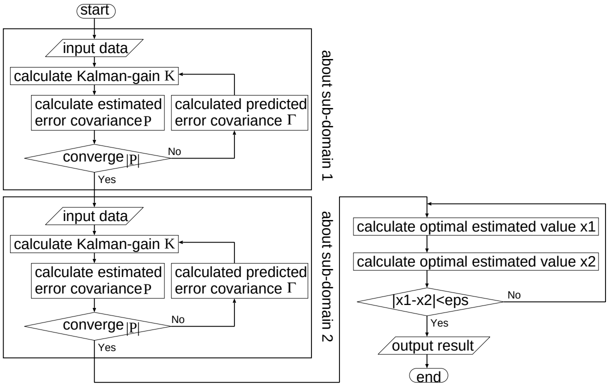

start
input data
calculate Kalman-gain K
calculate estimated error covariance P
calculated predicted error covariance Γ
converge |P|
No
Yes
about sub-domain 1
input data
calculate Kalman-gain K
calculate estimated error covariance P
calculated predicted error covariance Γ
converge |P|
No
Yes
about sub-domain 2
calculate optimal estimated value x1
calculate optimal estimated value x2
|x1-x2|<eps
No
Yes
output result
end

Figure 4: Algorithm of Kalman filter using SAP

Numerical Result

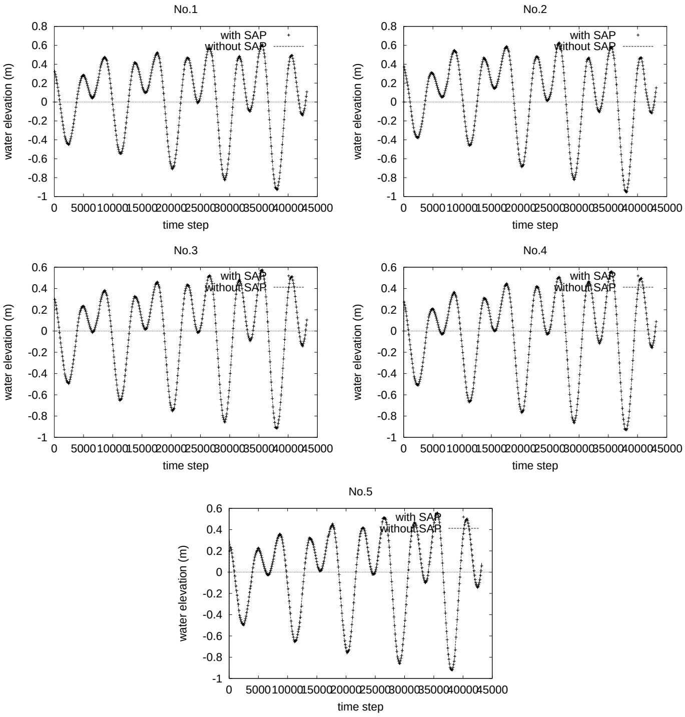
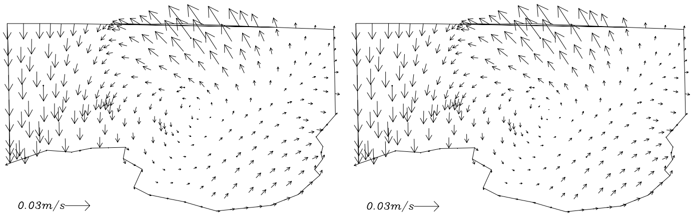


Figure 5: Comparison about water elevation



(a)With SAP (b)Without SAP

Figure 6: Velocity distribution (Residual flow)

Fig.5 shows comparing two results, one is using SAP and the other is not using SAP. According to this figure, it is recognized that difference between two values is very small. Fig.6 shows velocity distributions of residual flow. Fig.6(a) is result of analysis with SAP and Fig.6(b) is that without SAP. Comparing two figures, flow conditions are almost the same each other.

Case 2

Computational domain is simple model of channel. This domain is divided into 33 nodes and 40 elements. Time increment Δt is 0.0020, system noise Q is 0.000001 and observation noise R is 0.50. Quasi-observation data is given for 3 observation points.

In this case, computational domain decomposed into 3 subdomains with overlapping area is used. Each sub-domain has 15 nodes.

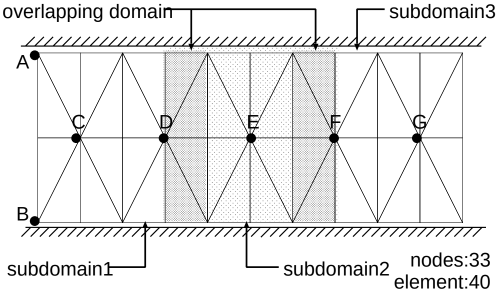


Figure 7: Computational domain of case 2

Numerical Result

Fig.8 shows comparing two results, one is using SAP and the other is not using SAP. According to this figure, it is recognized that difference between two values is very small. Fig.9 shows velocity distributions of residual flow. Fig.9(a) is the result of analysis with SAP and Fig.9(b) is that without SAP. Comparing two figures, small difference exists but direction of velocities can be estimated.

Conclusion

In this analysis, domain decomposition is applied for model problem of shallow water flow and actual tidal flow problem using Kalman filter.

In case 1, comparing results of two computations, two values almost agree (Fig.5 and Fig.6). These results show that algorithm of SAP with two sub-domains is established. And CPU time becomes 3 times shorter than that without SAP. In case2, time histories of results of water elevation also agree each other (Fig.8). Comparing velocity distributions using SAP and without using SAP shows that small difference exists. The estimation using SAP is smaller than that not using SAP. But direction of velocity is almost the same. If we refine the iteration method, more faithful analysis could be carried out.

Not only water elevation but also velocity distribution show that SAP with three sub-domains is not perfect. It seems that cause of this failure is overlapping areas are plural. If overlapping areas exist more than one, values obtained by calculation are not reflecting results of whole domain.

Algorithm of Kalman filter using SAP with three sub-domains has to be established. If we would succeed, CPU time becomes shorter.

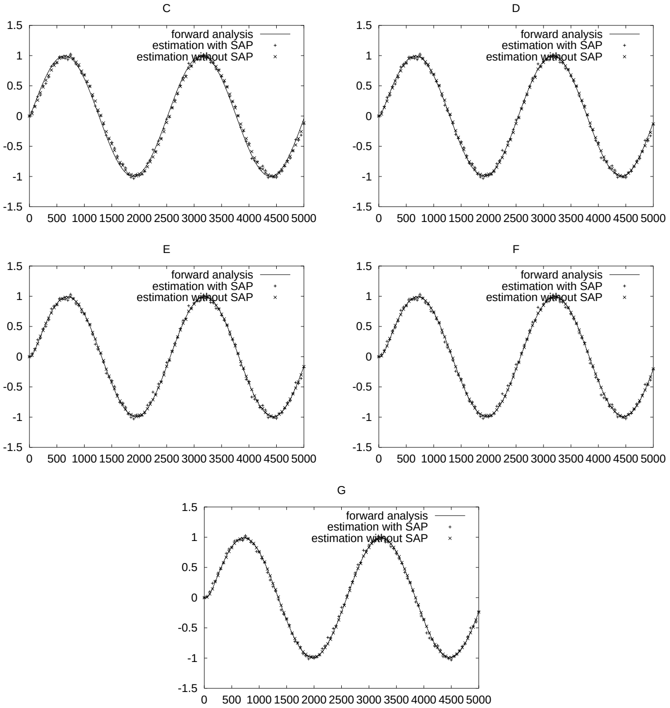
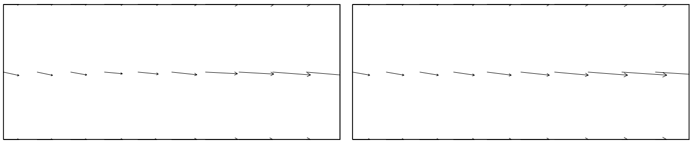


Figure 8: Comparison about water elevation



(a)With SAP (b)Without SAP

Figure 9: Velocity distribution (Residual flow)

References

[May94]A.S. Mayer. Application of domain decomposition techniques for multiphase groundwater problems. *Computational Method in Water Resources*, X:951–958, 1994.

[Sbo91]H. Sbosny. Parallel multigrid methods on composite method. Fifth International Symposium on Domain Decomposition, 1991.

[Tho91]J.M. Thomas. Finite element matching methods. Fifth International Symposium on Domain Decomposition, 1991.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau,

35. Two level Domain Decomposition for Multi-clusters

M. Garbey , D. Tromeur-Dervout [1] [2]

Introduction

We discuss the design of parallel algorithms to solve elliptic problems on multi-clusters computers. Multi-clusters can be seen as two-level parallel architecture machines, since communication between clusters are usually much slower than communication or access to memory within each of the clusters. We introduce special algorithms that use two levels of parallelism and match the multi-cluster architecture. Efficient parallel algorithms that rely on fast uniform communication have been extensively developed in the past: we intend to use them for parallel computation within the clusters. On top of these local parallel algorithms, new *robust and parallel* algorithms are needed that can work with few clusters linked by a slow communication network. We present a two level domain decomposition algorithm that uses Aitken or Steffensen acceleration procedure combined to Schwarz for the outer loop and standard parallel domain decomposition for the inner loop. We demonstrate finally the interest of our algorithm for metacomputing.

We consider the design of parallel algorithms for multi-cluster architecture with few heterogeneous clusters linked by an affordable network of order 10Mb/s bandwidth. Each cluster can be a shared multiprocessors machine or an MIMD computer with a fast internal Network. The elapse time to access memory from a given processor to a given data on such architecture is then strongly dependent on the location of the datas. Fast scalable parallel algorithm for the Laplace problem with domain decomposition and/or multigrid on a uniform MIMD architecture have usually very poor efficiency on multi-cluster machine with slow inter-cluster network.

On the contrary a numerically unefficient iterative domain decomposition algorithm such as the classical additive Schwarz procedure for the Laplace problem, is easy to implement, robust and scalable on multi-cluster architecture. So our goal is the design of an acceleration procedure for iterative domain decomposition analogous to additive Schwarz that increases the numerical efficiency of the basic underlined algorithm but stay easy to implement, robust and scalable on multi-clusters. The common procedure to accelerate additive Schwarz method is the introduction of a coarse-grid operator [LSFQ97]. The resulting modified Schwarz algorithms becomes numerically efficient but the coarse grid computation might be a bottle neck for the parallel processing. We adopt here a different point of view and try to extract from a finite sequence of the interfaces generated by the Schwarz iterative procedure or analogous relaxation method, an accurate prediction of the interface's limit. We will show in simple case as finite difference approximation of Elliptic operator with con-

[1] This work was supported by the Région Rhône Alpes

[2] CDCSP/ISTIL - University Lyon 1, 69622 Villeurbanne, France
{garbey,dtromeur}@cdcsp.univ-lyon1.fr, http://cdcsp.univ-lyon1.fr

stant coefficient on regular grids, that we can obtain a fast *direct* solver so called Aitken-Schwarz procedure. In more complex situation, we shall derive a fast iterative solver by alternating few Schwarz iterations with Aitken acceleration [SB80]. We will call this methodology Steffensen-Schwarz following the spirit of the Steffensen method in non-linear context [Hen64]. The main advantage of our approach is that the new algorithm requires only the coding of an independent subroutine that processes the sequences of interfaces generated by the basic domain decomposition method. In addition, we will show that this subroutine does not require too many communications and performs efficiently on multi-clusters with slow inter-cluster network. We will report in particular on a successful metacomputing experiment with distanced parallel computers.

The plan of this article is as follows. Next section presents a new family of domain decomposition algorithms in the one dimensional case. Then we generalize the method to multidimensional elliptic operator with strip domain decomposition, before presenting in an another section some extension of the results to linear elliptic operator with varying coefficients and non linear elliptic operators. Some results on large scale parallel computing are reported in the last section before our conclusion.

Basic idea in one D

two subdomains with Dirichlet-Dirichlet BC

Let us consider a linear problem

$$L[U] = f \; in \; \Omega, \; U_{|\partial\Omega} = 0. \tag{1}$$

L can be the continuous problem or the discrete one. We restrict ourselves to two subdomains and start with the additive Schwarz algorithm. For simplicity of the description of the method, we assume implicitly in the following notations that the homogeneous Dirichlet boundary condition in (1) is satisfied by all intermediate subproblems.

$$L[u_1^{n+1}] = f \; in \; \Omega_1, \; u_{1|\Gamma_1}^{n+1} = u_{2|\Gamma_1}^{n}, \tag{2}$$

$$L[u_2^{n+1}] = f \; in \; \Omega_2, \; u_{2|\Gamma_2}^{n+1} = u_{1|\Gamma_2}^{n}. \tag{3}$$

We observe that the operator T,

$$u_{i|\Gamma_i}^{n} - U_{\Gamma_i} \rightarrow u_{i|\Gamma_i}^{n+2} - U_{\Gamma_i} \tag{4}$$

is *linear*.

Let us consider first the one-dimensional case $\Omega = (0,1)$: the sequence $u_{i|\Gamma_i}^{2n}$ is a sequence of real numbers. Note that as long as the operator T is linear, the sequence $u_{i|\Gamma_i}^{n+2}$ has pure linear convergence (or divergence); that is, it satisfies the identity $u_{i|\Gamma_i}^{n+2} - U_{|\Gamma_i} = \delta(u_{i|\Gamma_i}^{n} - U_{|\Gamma_i})$, where δ is the amplification factor of the sequence. Let us assume $\delta \neq 1$. The Aitken acceleration procedure gives the *exact* limit of the sequence on the interface Γ_i based on three successive Schwarz iterates $u^j_{i|\Gamma_i}$, $j = 1, 2, 3$, and the initial condition $u^0_{i|\Gamma_i}$, namely,

$$u^\infty_{\Gamma_i} = \frac{u^0_{i|\Gamma_i} u^3_{i|\Gamma_i} - u^1_{i|\Gamma_i} u^2_{i|\Gamma_i}}{u^3_{i|\Gamma_i} - u^2_{i|\Gamma_i} - u^1_{i|\Gamma_i} + u^0_{i|\Gamma_i}}.$$

An additional solve of each subproblem (2,3) with boundary conditions $u^\infty_{\Gamma_i}$ gives the solution of (1). The Aitken acceleration thus transforms the additive Schwarz procedure into an *exact* solver regardless of the speed of convergence of the original Schwarz method.

With the previous algorithm, we do need 3 solves of each subproblem to apply the Aitken acceleration and an additional solve of each subproblem to get the solution. We can derive a more numerically efficient algorithm that requires 3 solves of each subproblems in the following way: we have

$$u^{n+1}_{1|\Gamma_2} - U_{|\Gamma_2} = \delta_1 (u^n_{2|\Gamma_1} - U_{|\Gamma_1}), \tag{5}$$

$$u^{n+1}_{2|\Gamma_1} - U_{|\Gamma_1} = \delta_2 (u^n_{1|\Gamma_2} - U_{|\Gamma_2}), \tag{6}$$

where δ_1 (resp δ_2) is the damping factor associated to the operator L in subdomain Ω_1 (resp Ω_2) [Gar96]. Consequently

$$u^2_{1|\Gamma_2} - u^1_{1|\Gamma_2} = \delta_1 (u^1_{2|\Gamma_1} - u^0_{2|\Gamma_1}),$$

$$u^2_{2|\Gamma_1} - u^1_{2|\Gamma_1} = \delta_2 (u^1_{1|\Gamma_2} - u^0_{1|\Gamma_2}),$$

So except if the initial boundary conditions $u^0_{2|\Gamma_1}$ or $u^0_{1|\Gamma_2}$ matches with the exact solution U at the interfaces Γ_i , the amplification factors δ_1 and δ_2 can be computed from (5) and (6). Then if $\delta_1 \delta_2 \neq 1$ the limit $U_{|\Gamma_i}, i = 1, 2$ is obtained as the solution of the linear system (5, 6).

We observe that δ_1, δ_2 are dependent only on the operator and the partitioning of the domain. δ_1 for example can be computed before hand as follows. Let $v_{1/2}$ be the solution of

$$L[v_{1/2}] = 0 \; in \; \Omega_{1/2}, \; v_{|\Gamma_{1/2}} = 1. \tag{7}$$

We have $\delta_{1/2} = v_{|\Gamma_{2/1}}$. When $\delta_{1/2}$ is a priori known, we need only *one* Schwarz iterate to accelerate the interface and an additional solves for each subproblems. This is a total of two solves per subdomain. This feature is particularly attractive when the elliptic problem (1) has to be solved many times.

two subdomains with Dirichlet-Newman BC

It is interesting that the same idea applies to other well-known iterative procedures such as the Dirichlet-Newman iterative procedure that has the advantage of using non overlapping partitioning but the disadvantage of possible divergence. The relaxation procedure of the Funaro-Quarteroni algorithm [FQZ88], can fix this convergence problem when the relaxation parameter is chosen correctly. However the Aitken acceleration procedure may solve the artificial interface problem whether the original Dirichlet-Neumann iterative procedure converges or diverges, as long as the sequence of solution at the interface behaves linearly! To be more specific let us consider for example the Helmholtz problem:

$$-\frac{d^2}{dx^2}U + \mu U = f \; in \; \Omega, \; U = 0 \; on \; \partial\Omega.$$

The domain Ω is split into two non overlapping subdomains that share the interface Γ. We consider the iterative procedure,

$$-\frac{d^2}{dx^2}u_1^n + \mu u_1^n = f \; in \; \Omega_1, \; u_1^n = u_2^n \; on \; \Gamma, \tag{8}$$

$$-\frac{d^2}{dx^2}u_2^n + \mu u_2^n = f \; in \; \Omega_2, \; \frac{\partial u_2^n}{\partial x} = \frac{\partial u_1^n}{\partial x} \; on \; \Gamma. \tag{9}$$

We approximate this problem with 2^d order finite differences for (8) and one side first order finite differences for the boundary condition in (9). The computation of each subproblems (8) and (9) is a priori a sequential process. The sequence of real numbers u_1^n generated by this algorithm has linear convergence to U_Γ or linear divergence depending on the interface location Γ, that is $u_{1|\Gamma}^{n+1} - U_{|\Gamma} = \delta(u_{1|\Gamma}^n - U_{|\Gamma})$, where δ is the amplification factor of the sequence. Once again the Aitken acceleration procedure gives the *exact* limit of this sequence no matter the value of $\delta \neq 1$, with

$$u_\Gamma^\infty = \frac{u_{1|\Gamma}^0 u_{1|\Gamma}^2 - u_{1|\Gamma}^1 u_{1|\Gamma}^1}{u_{1|\Gamma}^2 - 2u_{1|\Gamma}^1 + u_{1|\Gamma}^0}.$$

So far, we have restricted ourselves to domain decomposition with two subdomains. Next we will introduce a generalized Aitken acceleration technique that can be applied to an arbitrary number $q > 2$ of subdomains.

more than 2 subdomains case with Dirichlet-Dirichlet BC

Let $\Omega_i = (x_i^l, x_i^r), \; i = 1..q$ be a partition of Ω with $x_2^l < x_1^r < x_3^l < x_2^r, ..., x_q^l < x_{q-1}^r$. We consider the additive Schwarz algorithm

```
$for \; i = 1..q, do$
    $L[u_i^{n+1}] = f \; in \; \Omega_i, \; u_i^{n+1}(x_i^l) = u_{i-1}^n(x_i^l), \; u_i^{n+1}(x_i^r) = u_{i+1}^n(x_i^r),$
$enddo$
```

Let us denote $u_i^{l,n+1} = u_i^{n+1}(x_i^l)$, $u_i^{r,n+1} = u_i^{n+1}(x_i^r)$ and $\tilde{u}^n$ (respt $\tilde{u}$) be the n iterated (respt exact) solution restricted at the interface, i.e

$$\tilde{u}^n = (u_2^{l,n}, u_1^{r,n}, u_3^{l,n}, u_2^{r,n}, ..., u_q^{l,n}, u_{q-1}^{r,n})$$

The operator $\tilde{u}^n \rightarrow \tilde{u}^{n+1}$ is linear. Let us denote P its matrix. P has the following pentadiagonal structure:

$$\begin{vmatrix}
0 & \delta_1^r & 0 & 0 & .... & & & \\
\delta_2^{l,l} & 0 & 0 & \delta_2^{l,r} & ... & & & \\
\delta_2^{r,l} & 0 & 0 & \delta_2^{r,r} & ... & & & \\
 & & & & & & & \\
 & & & ... & \delta_{q-1}^{l,l} & 0 & 0 & \delta_{q-1}^{l,r} \\
 & & & ... & \delta_{q-1}^{r,l} & 0 & 0 & \delta_{q-1}^{r,r} \\
 & & & ... & 0 & 0 & \delta_q^r & 0
\end{vmatrix}$$

δ_1^r and δ_q^r can be computed as in the two subdomain cases.

The subblocks $P_i = \begin{vmatrix} \delta_i^{l,l} & \delta_i^{l,r} \\ \delta_i^{r,l} & \delta_i^{r,r} \end{vmatrix}$ $i = 2..q-1$ can be computed with 3 Schwarz iterates as follows.
We have $(u_{i-1}^{r,n+1} - \tilde{u}_{i-1}^r, u_{i+1}^{l,n+1} - \tilde{u}_{i+1}^l)^t = P_i(u_i^{l,n} - \tilde{u}_i^l, u_i^{r,n} - \tilde{u}_i^r)^t$. Therefore

$$\begin{pmatrix} u_{i-1}^{r,n+3} - u_{i-1}^{r,n+2} & u_{i-1}^{r,n+2} - u_{i-1}^{r,n+1} \\ u_{i+1}^{l,n+3} - u_{i+1}^{l,n+2} & u_{i+1}^{l,n+2} - u_{i+1}^{l,n+1} \end{pmatrix} = P_i \begin{pmatrix} u_i^{l,n+2} - u_i^{l,n+1} & u_i^{l,n} - u_i^{l,n} \\ u_i^{r,n+2} - u_i^{r,n+1} & u_i^{r,n} - u_i^{r,n} \end{pmatrix} \quad (10)$$

In practice the last matrix on right hand side of the previous equation is non singular and P_i can be computed, but it cannot be guaranty. However, one can always compute before hand the coefficients of P_i as follows. Let v be the solution of

$$L[v] = 0 \; in \; \Omega_i, \; v(x_i^l) = 1, \; v(x_i^r) = 0, \quad (11)$$

and w be the solution of

$$L[w] = 0 \; in \; \Omega_i, \; w(x_i^l) = 0, \; w(x_i^r) = 1. \quad (12)$$

We have then $\delta_i^{l,l} = v(x_{i-1}^r)$, $\delta_i^{l,r} = v(x_{i+1}^l)$ $\delta_i^{r,l} = w(x_{i-1}^r)$ and $\delta_i^{r,r} = w(x_{i+1}^l)$. We observe that this computation of the subblocks P_i can be done in parallel.

In addition, for the Helmotz operator $L[u] = u'' - \lambda u$, or generally speaking elliptic problems with constant coefficients, the matrix P is known analytically.

From the equality

$$\tilde{u}^{n+1} - \tilde{u} = P(\tilde{u}^n - \tilde{u}),$$

one writes the generalized Aitken acceleration as follows:

$$\tilde{u}^\infty = (Id - P)^{-1}(\tilde{u}^{n+1} - P\tilde{u}^n). \quad (13)$$

If the additive Schwarz method converges, then $||P|| < 1$ and $Id - P$ is non singular. The algorithm is then

• step1 : compute analytically or numerically in parallel each subblocks P_i from each subproblems (11,12).
• step2: apply one additive Schwarz iterate.
• step3: apply generalized Aitken acceleration on the interfaces based on (13) with $n = 0$.
• step4: compute in parallel the solution for each subdomain.

Algorithm I

From the point of view of parallelism step1 and step4 does not requires any communication. step2 requires local communication between subdomains that overlap. Step3 on the contrary requires global communication. We will see in the next section, how theses basic ideas can be extended on multidimensional elliptic operators and how to minimize the global communications involved in step3.

Multidimensional elliptic operator

general formal framework

Next, let us consider the multidimensional case with the discretized version of the problem (1). We restrict ourselves for simplicity to the two overlapping subdomain case and the additive Schwarz algorithm (2, 3). Let us denote $E_i^h, i = 1, 2$ some finite vector space used to approximate the solution restricted to the artificial interface $\Gamma_i, i = 1, 2$. Let $b_i^j, j = 1..N$ be a set of basis functions for this vector space and P be the corresponding matrix of the linear operator T

$$u^n_{i|\Gamma_i} - U_{\Gamma_i} \rightarrow u^{n+2}_{i|\Gamma_i} - U_{\Gamma_i}.$$

We denote by $u^n_{i,j}, j = 1, .., N$ the components of $u^n_{i|\Gamma_i}$, and we have then

$$(u^{n+2}_{i,j} - U_{j|\Gamma_i})_{j=1,..,N} = P(u^n_{i,j} - U_{j|\Gamma_i})_{j=1,..,N}.$$

let us suppose that the interface sequence is such that the matrix $(u^{2(j+1)}_{k,i} - u^{2j}_{k,i})_{i=1,..,N,j=0,..,N-1}$ is non singular. Let Id be the matrix for the identity operator. We introduce a generalized Aitken acceleration with the following formula: first

$$P = (u^{2(j+1)}_{k,i} - u^{2j}_{k,i})_{i=1,..,N,j=1,..,N}(u^{2(j+1)}_{k,i} - u^{2j}_{k,i})^{-1}_{i=1,..,N,j=0,..,N-1}, k = 1, 2,$$

and second, if $Id - P$ is non singular, the trace of the exact solution $(u_{k,i})_{i=1,..,N}$ on interface $\Gamma_k, k = 1, 2$ is the solution of the linear system

$$(Id - P)(u^\infty_{k,i})_{i=1,..,N} = (u^{2N+2}_{k,i})_{i=1,..,N} - P\ (u^{2N}_{k,i})_{i=1,..,N}.$$

If this generalized Aitken procedure works, it should be a priori independently of the spectral radius of P, that is, the convergence of the underlined Schwarz additive iterative procedure is not needed. In conclusion, $2N + 1$ Schwarz iterates produce a priori enough data to compute via this generalized Aitken acceleration the interface value $U_{|\Gamma_k}, k = 1, .., 2$. This computation is amenable to $N+1$ Schwarz iterates, if one accelerates the sequence of coupled interfaces corresponding to the linear mapping

$$(u^n_{1|\Gamma_1} - U_{\Gamma_1}, u^n_{2|\Gamma_2} - U_{\Gamma_2}) \rightarrow (u^{n+1}_{1|\Gamma_1} - U_{\Gamma_1}, u^{n+1}_{2|\Gamma_2} - U_{\Gamma_2}).$$

However, we can expect that the matrix $(u^{2(j+1)}_{k,i} - u^{2j}_{k,i})_{i=1,..,N,j=0,..,N-1}$ is ill-conditioned and that the computed value of P is very sensitive to the data. In addition N or $2N$ Schwarz iterates is too many iterates to be considered as an efficient procedure.

Nevertheless, we have numerical evidence that this procedure can perform on two dimensional linear elliptic problems with stiff coefficients [GTD99]

We are currently investigating diverse strategies to make this algorithm useful and efficient in the framework of unstructured grid but we will restrict ourselves in this paper to the case of regular grids for which sine or cosine expansion of the traces generated by additive Schwarz is a natural tool.

Aitken-Schwarz method for Elliptic Operator

Let us consider first the Poisson problem $u_{xx} + u_{yy} = f$ in the square $(0, \pi)^2$ with Dirichlet boundary conditions. We partition the domain into an arbitrary number nd of overlapping strips: $\Omega = \bigcup_{j=1..nd} \Omega_j$. We introduce the regular discretization in the y direction $y_i = (i-1)h$, $h = \frac{1}{N-1}$, and central second-order finite differences of the u_{yy} derivative. Let us denote by $\hat{u}_i$ (resp. $\hat{f}_i$) the coefficient of the sine expansion of u (resp. f). The Poisson problem decomposes then into N independents semi-discretized equation corresponding to sinus waves $sin(iy), i = 1..N$,

$$\hat{u}_{i,xx} - 4/h^2 \; sin^2(i\frac{h}{2}) \; \hat{u}_i = \hat{f}_i, \tag{14}$$

The matrix P for the set of basis functions $b_i = sin(i\frac{y}{\pi})$ is therefore *diagonal.* The Aitken Schwarz algorithm is very similar to the algorithm derived in the one dimensional case. In particular the coefficients of each wave number of the trace of the solutions generated by the Schwarz algorithm has its own linear rate of convergence, the high frequencies terms being damped the fastest. The algorithm writes:

• step1 : compute analytically or numerically in parallel each subblocks P_i from each subproblems (11,12) and each operator $L_i[v] = v_{xx} - 4/h^2 \; sin^2(i\frac{h}{2}) \; v$.

• step2: apply one additive Schwarz iterate to the Poisson problem with block solver of choice i.e multigrids, FFT etc...

• step3:

- compute the sine expansion $\hat{u}^n_{j|\Gamma_i}, n = 0, 1$ of the traces on the artificial interface $\Gamma_i, i = 1..nd$ for the initial boundary condition $u^0_{|\Gamma_i}$ and the solution given by one Schwarz iterate $u^1_{|\Gamma_i}$.
- apply generalized Aitken acceleration based on (13) with $n = 0$ *separately* to each wave coefficients in order to get $\hat{u}^\infty_{j|\Gamma_i}$.
- recompose the trace $u^\infty_{j|\Gamma_i}$ in physical space.

• step4: compute in parallel the solution in each subdomains Ω_j, with new inner BCs and block solver of choice.

Algorithm II

This algorithm has a very high potential of parallelism. step 1 and 4 are fully parallel. Step 2 requires only local communication and scale well with the number of processors. Step 3 requires global communication of interfaces in Fourier space. But high frequency have very fast decay and little influence on the final solution. Therefore one can restrict adaptively the Aitken acceleration process of step3 to a subset

$\hat{u}_j^n, j = 1..M$, with $M < N$, and minimize the amount of global communications. In addition the arithmetic complexity of step3 that is the kernel of the method is negligible compare to step2. Further, this procedure works independently of the discretization and grids in x direction as long as the block solvers for each subproblems are exact. The same idea can be applied to Elliptic problems with constant coefficients or x dependent coefficients since the matrix P in such cases stays diagonal. Let us notice that for Elliptic problem with homogeneous Neumann BC instead of Dirichlet BC, one has to accelerate the cosine expansion of the interface's sequence. For Elliptic problem with non homogeneous BC, it is convenient to work on a shifted sequence that satisfies the homogeneous BC.

To exemplify the Aitken Schwarz procedure with a slightly more difficult case, let us consider the transmission problem:

$$-\mu_1 \Delta u_1 + u_1 = f \ \ in \ \ (0, \frac{\pi}{2}) \times (0, \pi) \tag{15}$$

$$-\mu_2 \Delta u_2 + u_2 = f \ \ in \ \ (\frac{\pi}{2}, 1) \times (0, \pi) \tag{16}$$

with homogeneous Dirichlet boundary conditions. μ_1 and μ_2 are positive constants. Let us discretize this simple problem with second order central differences and iterate with a Dirichlet-Neumann domain decomposition. For $\mu_1 = 1$ and $\mu_2 = 8$ this procedure is *linearly* divergent, but the following Aitken acceleration applied to the sine expansion of the trace of the solution $u_1(.,y)$ at $x = \frac{\pi}{2}$,

$$\hat{u}_k^\infty = \hat{u}_k^0 - \frac{(\hat{u}_k^1 - \hat{u}_k^0)^2}{\hat{u}_k^2 - 2\hat{u}_k^1 + \hat{u}_k^0},$$

generates the sine expansion of the exact interface solution modulo the residual error of each subdomain solve. Fig 1 reports on the numerical result obtained with matlab for a small test case i.e 25 by 25 grid points. This example is interesting because the convergence history has not the classical behavior that one may expect!.

Let us now describe briefly some key aspect of the stability of the Aitken Schwarz algorithm.

sensitivity analysis

It is interesting to understand how behaves the Aitken-Schwarz method if one use inexact block solver or approximation of the matrix of operator T. This is obviously related to the stability of the acceleration procedure with respect to perturbation of P or perturbation of $\tilde{u}^n$. Let us summarize briefly the results we found for discrete linear elliptic operators that satisfies a maximum principle. Extension of the results and details of the analysis will be available in a forthcoming paper.

We assume for simplicity a uniform strip domain decomposition and writes

$$\begin{pmatrix} \delta_1 & 0 \; 0 \; \delta_2 \\ \delta_2 & 0 \; 0 \; \delta_1 \end{pmatrix} \tag{17}$$

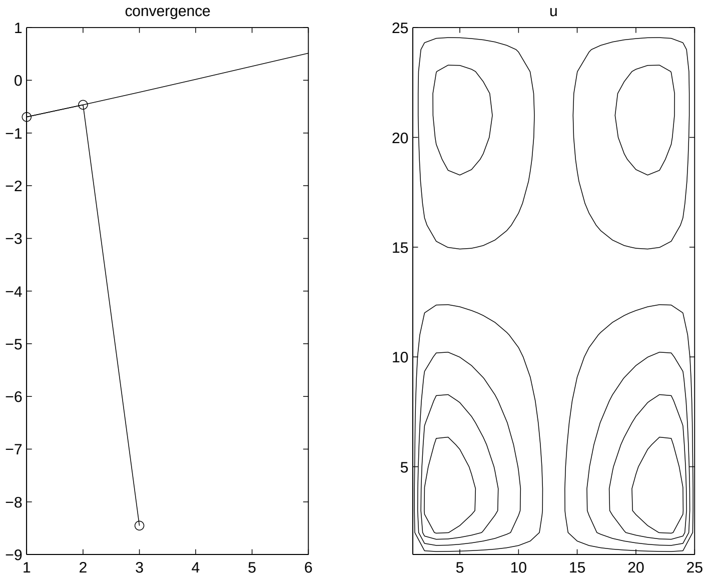


Figure 1: Dirichlet-Newman Algorithm for a Transmission Problem. Solid line (resp. -o- line) gives the log_{10} (error in maximum norm) on the discrete solution additive with basic procedure (resp. new method)

the generic subblock of P for a given wave number k.

Let $\tilde{P}$ be an approximation of P. The relative error on the artificial interface vector $\tilde{u}$ is then bounded by

$$2\frac{||(Id-P)^{-1}||^2||(P-\tilde{P})||}{1-||(Id-P)^{-1}(P-\tilde{P})||}+||(Id-P)^{-1}(P-\tilde{P})||.$$

Since the operator L satisfied a maximum principle, this corresponds to the global error. A straightforward application of this estimate is the minimization of the communication constraint in step 3 of Aitken-Schwarz'Algorithm, if one neglects interactions between subdomains that are not neighbors. It is equivalent to approximate P with the following matrix $\tilde{P}$ for acceleration:

$$\left|\begin{array}{cccccccc}
0 & \delta_1 & 0 & 0 & .... & & & \\
\delta_1 & 0 & 0 & 0 & ... & & & \\
0 & 0 & 0 & \delta_1 & ... & & & \\
 & & & & & & & \\
 & & & ... & \delta_1 & 0 & 0 & 0 \\
 & & & ... & 0 & 0 & 0 & \delta_1 \\
 & & & ... & 0 & 0 & \delta_1 & 0
\end{array}\right|$$

The error on the corresponding predicted wave amplitude of the interface given by the incomplete Aitken acceleration is then bounded by $(2\delta_2\frac{1+\delta_1}{1-\delta_1}+\delta_2)/(1-\delta_1^2)$. It is clear that δ_1 and δ_2 decrease as the corresponding frequency increases. One can therefore decouple adaptively the computation depending on the wave number, preserving the overall accuracy of the method.

One can also analyze the impact of inexact sub-block solver. Let us restrict ourselves to the Poisson problem in two space dimensions with five point schemes. If P_i is computed either analytically or independently with high accuracy, the numerical error is then bounded by $\frac{\eta}{h}$ where η stands for the maximum error in each inexact block solves and h for the time step. If P is computed numerically from Schwarz iterates with inexact sub-block solve the situation is more complicated. The acceleration procedure is much more sensitive and we get an upper bound of order $\frac{\eta}{h^3}$.

Because the accuracy of the Aitken-Schwarz procedure deteriorates with the uncomplete construction of the matrix P or the inexact sub-block solve, it is natural to apply the same acceleration procedure in a loop until appropriate convergence. We name this procedure a Steffensen-Schwarz algorithm and we are going to show that this algorithm is suitable to solve elliptic problems far more complicated than the Poisson problem.

Steffensen-Schwarz method for linear and non linear elliptic operator

Let us consider first the Linear case $L = -\Delta u + a(x,y)u$, with a varying smooth coefficient a. In all numerical experiments, thereafter, we will consider strip domain decomposition with *minimum overlap*, i.e one mesh overlap.

Linear Elliptic operator

For simplicity of the presentation, we consider (4) with only two overlapping subdomains. The elementary methods described for the Poisson problem in Section *Multi-dimensional elliptic operator* fails to be an exact solver if the grid has a non constant space step in the y direction or if the operator has coefficients depending on the x *and* y variable, because P is no longer diagonal but rather a dense matrix!. However if one approximates the coefficients a by its Z truncated Cosine expansions as follows,

$$a(x,y) \approx \Sigma_{k=1..Z}\hat{a}_k(x)cos((k-1)y),$$

matrix P is then a sparse matrix of bandwidth $2Z+1$. Our heuristic strategy is therefore to try to rebuild from the sequence of $2Z+1$ consecutive interfaces generated by Schwarz, a band approximation P_Z of P. We look then for P_Z such that,

$$(\hat{u}_i^{2Z+2} - \hat{u}_i^{2Z+1}, ..., \hat{u}_i^3 - \hat{u}_i^2, \hat{u}_i^2 - \hat{u}_i^1) = (P_{i,i-Z}, ..., P_{i,i+Z}) \times S_B, \qquad (18)$$

where S_B is the following subblock

$$\begin{pmatrix} \hat{u}_{k-Z}^{2Z+1} - \hat{u}_{k-Z}^{2Z} & ...\hat{u}_{k-Z}^{1} - \hat{u}_{k-Z}^{0} \\ . & ... \\ . & ... \\ \hat{u}_{k+Z}^{2Z+1} - \hat{u}_{k+Z}^{2Z} & ...\hat{u}_{k+Z}^{1} - \hat{u}_{k+Z}^{0} \end{pmatrix} \qquad (19)$$

provided by the Schwarz iterative process. (18) holds for $Z < i \leq N - Z$. Similar equation can be written with appropriate reduced dimension for the end terms of the diagonal of P_Z that is when $i \leq Z$ or $i > N - Z$. If S_B is non singular, the k^{ieme} row of P_Z is well defined. Otherwise, we have to decrease Z for this specific row until the subblock is non singular. In practice the conditioning of the subblock deteriorates when the frequency increases but only low frequencies needed to be accelerated since high frequencies are damped very fast by the Schwarz method itself.

Fig 2a and Fig 2b give numerical illustration of the method for different coefficient functions $a(x, y)$ and different choices for the bandwidths. Convergence curves are commented with + sign for Z=1, o sign for Z=2 and v sign for Z=3. We have chosen coefficients $a = 1 + y$ and $a = 1. + exp(sin(y)$ that have cosine expansion with growing speed of convergence. Our numerical experiment seems to confirm that the faster the cosine expansion of $a(x, .)$ converges, the faster converges the Steffensen approximation with the diagonal approximation $Z = 1$ of P. On the contrary the $Z = 3$ approximation improves best the convergence compare to the algorithm with $Z = 1$, when the convergence of the Fourier expansion of a is slow -see Fig 2a.

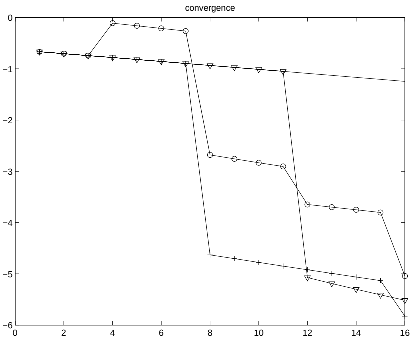


Fig 2a: $a(x, y) = 1. + y$

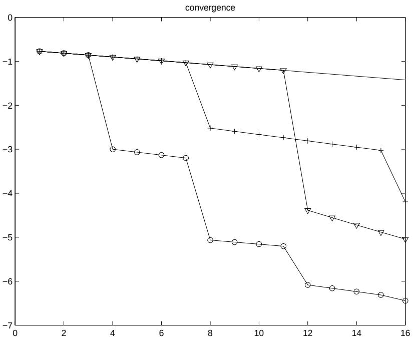


Fig 2b: $a(x, y) = 1. + exp(sin(y))$

Fig 3a and Fig 3b report on similar results but for the Poisson problem on an irregular domain that is a square except on one side, that is replaced by a reentry corner.

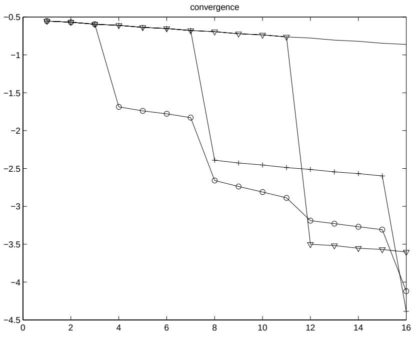


Fig 3a:Convergence history for irregular geometry

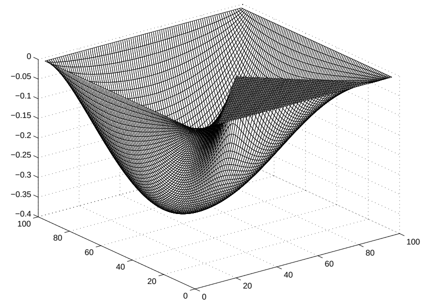


Fig 3b:Solution of the problem

the non linear case

We consider a one dimensional nonlinear problem that is a simplified model of a semiconductor device [Sel84]. The model writes

$$\Delta u = e^u - e^{-u} + f,\ in(0,d), \tag{20}$$

$$f = tanh(20(\frac{x}{d} - \frac{1}{2})),\ x \in (0,d), \tag{21}$$

$$u(0) = asinh(\frac{f(0)}{2}) + u_o, u(d) = asinh(\frac{f(d)}{2}) \tag{22}$$

The problem is discretized by means of second-order central finite differences. We apply Steffensen-Schwarz method with two subdomains and minimum overlap. In particular, we solve a non linear problem in each subblock at each iteration step of additive Schwarz. Fig 4a reports on the numerical results with 80 grid points. The convergence history shows that the closer the iterate gets to the final solution, the better is the result of the Aitken acceleration. This Newton like property of convergence of the algorithm can be actually proven using the monotonicity of the discrete non linear operator -see also [Hen64].

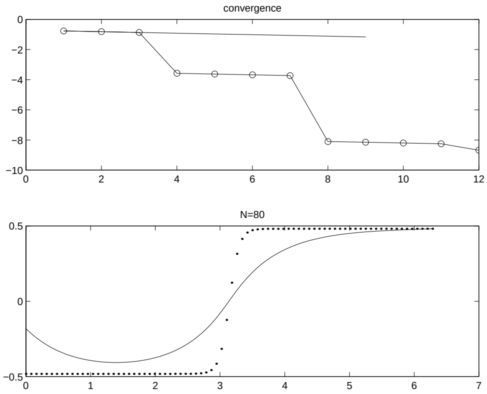


Fig 4a: One D semi conductor problem

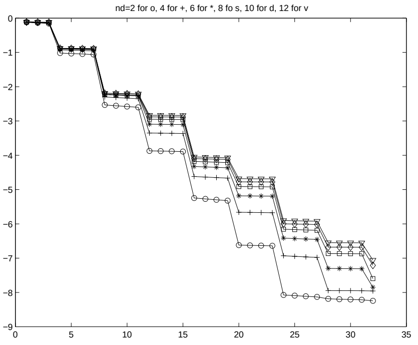


Fig 4b: Convergence for the two-D Bratu problem

We consider second the *Bratu* problem [Wie96],

$$-\Delta u = \lambda e^u,\ in\ \Omega = (0,1)^2, \tag{23}$$

$$u_{|\partial\Omega} = 0 \tag{24}$$

This problem has a smooth solution for $\lambda \in (0, 6.81)$. We have experimented the Steffensen Schwarz algorithm for the classical five points finite difference scheme with strip domain decomposition, an arbitrary number of subdomains and $\lambda = 6$. Our numerical experiments have shown that the Steffensen-Schwarz algorithm with diagonal approximation of P is best. Let us notice that $u(x_i, .)$ restricted to artificial interfaces of strip domain decomposition is a continuous periodic function of period 1; non homogeneous boundary conditions might then lead to a different choice for Z.

Fig 4b shows the solution and the convergence of our methods with a grid of approximatively fixed size 60×60 and an increasing number of subdomains from 2 to 12. It can be seen that unfortunately the one dimensional quadratic convergence property is lost in multidimensional problems, because the linear approximation of the operator has coefficients depending on space. As a matter of fact each step between two plateau in the convergence history has about the same size. However, it is most interesting to notice that the number of Steffensen-Schwarz iterates required to reach a given level of accuracy depends slightly on the number of subdomains. The total number of Schwarz iterates to reach an error less than 10^{-7} in maximum norm is 24 with 3 subdomains, and 32 with 12 subdomains.

We are going now to return to parallel efficiency of this new domain decomposition domain that was our motivation.

Application to distributed computing

We report on performance of Aitken Schwarz algorithm for three dimensional Poisson problem. Each subblock will be solved on a parallel system itself with a "classical" parallel algorithm. We will therefore referee to subblocks as macro subblocks since there are also decomposed into subdomains. To be more precise our Aitken Schwarz code is part of a 3 dimensional Navier Stokes code and is used to solve simultaneously 3 Laplace problems for each component of the flow speed [TD93]. The Aitken-Schwarz method in three D is similar to the two D algorithm II, except that we use two dimensional FFT for interfaces. In addition, the matrix $P_{i,j}$ corresponding to each couples of sine waves $[sin(iy), sin(jz)]$ can be precomputed analytically. Each macro subblock is solved with a parallel algorithm that combines multigrid and Schur dual complement method (**MCSD**). This parallel macro-block solver is very efficient and scalable on large MIMD system with uniform network.

We first compare the Aitken Schwarz method with MCSD on a SGI Origin 2000 system thanks to the **Centre Informatique National de l'Enseignement Supérieur** support.

Table 1 gives elapse time for 3 Laplace solve with 8388608 grid points. The "one Macro-Subdomain row" corresponds to MCSD algorithm. The three next rows correspond to the combination of Aitken Schwarz for the macro domain decomposition and MCSD in each macro subdomain. Our actual implementation of Aitken Schwarz is not optimum, since we use blocking communications, redundant interface treatment, and gather of all the interfaces. However we see that this new method can compete with our former optimized implementation of MCSD technique.

On large MIMD machine the salient feature of our multilevel domain decomposition is not used because we have not been able to allocate the processes in order to get the best performance of SGI network. In metacomputing experiments, we obtain our main result: table 2 shows that Aitken Schwarz performs 10 times better than MCSD when one use two clusters linked by a 10Mb/s network. In this experiment we have used two different generations of Compaq clusters with one or two 4 ev5 hypernodes called 4100 Dec alpha servers and dual ev6 hypernode called DS20. The elapsed time in this table are given for 3 Laplace solves and a total of 197000 unknowns. Table 3 shows that our Aitken Schwarz gives also very good results for a slow non dedicated network i.e 2Mb/s that is the France Telecom regular link between University Lyon1-Claude Bernard and Ecole Normale Supérieure of Lyon (**ENSL**) 10 kilometers away. The total number of unknowns in this last experiment is 288000, and we use in addition to Dec alpha4100 and DS20 alpha servers (respectively CDCSP-MOBY and CDCSP DS20), the sun Enterprise 10000 parallel computer of the Pole of Numerical Simulation and Modeling of ENSL (PSMN-SDF1). Let us mention, that in this last case, it is hopeless to use MCSD.

Number of Macro Subdomains	Time in second	Error in Maximum norm	FFT	gather interface
1	46.5	1.3 E-12		
2	71.5	2.0 E-12	2.0s	0.3s
4	56.1	1.0 E-12	4.6s	6.5s
8	59.6	2.6 E-12	11.0s	13.0s

Table 1: *Performance of the analytical Aitken Schwarz algorithm on SGI system.*

Cluster 1 3 or 4 processors	Cluster 2 2 or 3 or 4 processors	Elapse time in second	Bandwidth of network
4 ev5 CDCSP-MOBY	4 ev5 CDCSP-MOBY	28.4s	100 Mb/s
4 ev5 CDCSP-MOBY	2 ev6 CDCSP-DS20	29.4s	10 Mb/s
2 ev5 CDCSP-MOBY 1 ev6 CDCSP-DS20	2 ev5 CDCSP-MOBY 1 ev6 CDCSP-DS20	220.7s	10 Mb/s

Table 2: *Performance of the analytical Aitken Schwarz algorithm on intranet.*

We are currently running similar experiment with metacomputing between large parallel systems located in different countries in order to validate our approach on 3D large scale complex problems.

Conclusion

We have developed in this paper a new two levels domain decomposition method designed to work efficiently on multi-cluster architecture. We have combined fast parallel solvers such as Multigrids and Schur complement dual that are scalable and efficient inside the clusters and acceleration of robust solvers as additive Schwarz algorithm that does not require too many inter-cluster communications. We have shown that

Cluster 1 4 processors	Cluster 2 4 processors	Cluster 3 4 or 2 processors	Elapse time in second	Bandwidth of network
4 PSMN-SDF1	4 PSMN-SDF1	4 PSMN-SDF1	28.8 s	not available
4 ev5 CDCSP-MOBY	4 ev5 CDCSP-MOBY	4 ev5 CDCSP-MOBY	20.7s	100 Mb/s
4 PSMN-SDF1	4 ev5 CDCSP-MOBY	2 ev6 CDCSP-DS20	31.2s	2 Mb/s

Table 3: *Performance of the analytical Aitken Schwarz algorithm on City's Network.*

the basic idea of acceleration of relaxation domain decomposition method via Aitken transform is a possible efficient alternative to acceleration that use multilevel grid concepts for the efficient solution of Elliptic problem with regular grids and we hope to extend similar ideas in the context of unstructured meshes.

References

[FQZ88]D. Funaro, A. Quarteroni, and P. Zanolli. An iterative procedure with interface relaxation for domain decomposition methods. *SIAM J. Numer. Anal.*, 25(6):1213–1236, 1988.

[Gar96]M. Garbey. A schwarz alternating procedure for singular perturbation problems. *SIAM J. Sci. Comput.*, 17:1175–1201, 1996.

[GTD99]M. Garbey and D. Tromeur-Dervout. Operator splitting and domain decomposition for multiclusters. In D. Keyes and al editors, editors, *Proc. Parallel CFD99*, 1999. to appear.

[Hen64]P. Henrici. *Elements of Numerical Analysis.* John Wiley & Sons Inc, New York-London-Sydney, 1964.

[LSFQ97]L.Paglieri, A. Scheinine, L. Formaggia, and A. Quarteroni. Parallel conjugate gradient with schwarz preconditioner applied to fluid dynamics problems. In P. Schiano et al., editor, *Parallel Computational Fluid Dynamics, Algorithms and Results using Advanced Computer, Proceedings of Parallel CFD'96*, pages 21–30, 1997.

[SB80]J. Stoer and R. Burlish. *Introduction to numerical analysis.* TAM 12 Springer, New York, 1980.

[Sel84]S. Selberherr. *Analysis and simulation of semiconductor devices.* Springer Verlag, Wien, New York, 1984.

[TD93]D. Tromeur-Dervout. *Résolution des Equations de Navier-Stokes en Formulation Vitesse Tourbillon sur Systèmes multiprocesseurs à Mémoire Distribuée.* PhD thesis, University Paris 6 / ONERA, January 1993.

[Wie96]C. Wieners. A parallel newton multigrid method for high order finite elements and its application to numerical existence proofs for elliptic boundary value problem. *Combustion Theory and Modelling*, 76(3):175–180, 1996.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau,

36. A Domain Embedding Method for the Direct Numerical Simulation of Fluidization and Sedimentation Phenomena

R. Glowinski[1], T.-W. Pan[2], D.D. Joseph[3]

Introduction

Motivated by the *direct numerical simulation of particulate flow* (i.e., of the motion of fluid-particle mixtures) the authors of this paper, with the assistance of several collaborators, have introduced some years ago a computational methodology based on a *fictitious domain* formulation involving distributed Lagrange multipliers defined over the particles; this approach allows the flow computations to be done on a fixed space region of simple shape, giving thus to the practitioners the possibility of very fast solvers to treat, for example, the diffusion and the incompressibility if we assume that the fluid is viscous and incompressible. The above methodology will be briefly discussed in the next section and then applied to the direct simulations of the fluidization of 1204 identical rigid solid spherical particles contained in a "bed" of simple shape and of the sedimentation of 6400 disks in a 2D rectangular box. For more details on the methodology briefly discussed in this paper and for further numerical results obtained with it see [GHJ+97, GPH+98, PGH+98, GPHJ99, GPH+99, Pan99].

Mathematical Models for Particulate Flow: A Fictitious Domain Based Equivalent Formulation.

Modeling of the fluid-particle interaction.

Let $\Omega \subset I\!R^d (d = 2, 3)$ be a space region; we suppose that Ω is filled with an *incompressible viscous fluid* of density ρ_f and that it contains J moving rigid bodies $P_1,\ P_2, ..., P_J$ (see Figure 1 for a particular case where $d = 2$ and $J = 3$). We denote by $\mathbf{n}$ the unit normal vector on the boundary of $\Omega \setminus \cup_{j=1}^{J} \overline{P}_j$, pointing outward to the flow region. Assuming that the only external force acting on the mixture is *gravity*, then, between *collisions* (assuming that collisions take place), the *fluid flow* is modeled by the following *Navier-Stokes equations*

[1] Department of Mathematics, University of Houston, Houston, TX 77204-3476, roland@math.uh.edu

[2] Department of Mathematics, University of Houston, Houston, TX 77204-3476, pan@math.uh.edu

[3] Aerospace Engineering and Mechanics, University of Minneapolis, 107 Ackerman Hall, 110 Union Street, Minneapolis, MN 55455, joseph@aem.umn.edu

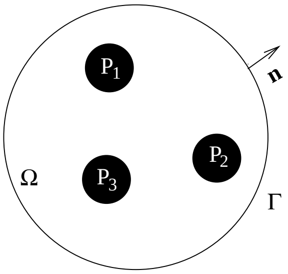


Figure 1: An example of two-dimensional flow region with three particles.

$$\begin{cases} \rho_f \left[\dfrac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \boldsymbol{\nabla})\mathbf{u} \right] = \rho_f \mathbf{g} + \boldsymbol{\nabla} \cdot \boldsymbol{\sigma} \ in \ \Omega \setminus \cup_{j=1}^{J} \overline{P_j(t)}, \\ \boldsymbol{\nabla} \cdot \mathbf{u} = 0 \ in \ \Omega \setminus \cup_{j=1}^{J} \overline{P_j(t)}, \\ \mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0(\mathbf{x}), \ \forall \mathbf{x} \in \Omega \setminus \cup_{j=1}^{J} \overline{P_j(0)}, \ with \ \boldsymbol{\nabla} \cdot \mathbf{u}_0 = 0, \end{cases} \tag{1}$$

to be completed by

$$\mathbf{u} = \mathbf{g}_0 \ on \ \Gamma \ with \ \int_\Gamma \mathbf{g}_0 \cdot \mathbf{n} d\Gamma = 0 \tag{2}$$

and by the following *no-slip boundary condition* on the boundary ∂P_j of P_j

$$\mathbf{u}(\mathbf{x}, t) = \mathbf{V}_j(t) + \boldsymbol{\omega}_j(t) \times \overrightarrow{\mathbf{G}_j(t)\mathbf{x}}, \ \forall \mathbf{x} \in \partial P_j(t), \tag{3}$$

where, in (3), $\mathbf{V}_j$ (resp., $\boldsymbol{\omega}_j$) denotes the *velocity of the center of mass* $\mathbf{G}_j$ (resp., the *angular velocity*) of the j^{th} particle, for $j = 1, ..., J$. In (1), the *stress-tensor* $\boldsymbol{\sigma}$ verifies

$$\boldsymbol{\sigma} = \boldsymbol{\tau} - p\mathbf{I}, \tag{4}$$

typical situations for $\boldsymbol{\tau}$ being

$$\boldsymbol{\tau} = 2\nu \mathbf{D}(\mathbf{u}) = \nu(\boldsymbol{\nabla}\mathbf{u} + \boldsymbol{\nabla}\mathbf{u}^t) \ \ (Newtonian \ case), \tag{5}$$

$$\boldsymbol{\tau} \ is \ a \ nonlinear \ function \ of \ \boldsymbol{\nabla}\mathbf{u} \ \ (non-Newtonian \ case). \tag{6}$$

The motion of the particles is modeled by the following *Newton-Euler* equations

$$\begin{cases} M_j \dfrac{d\mathbf{V}_j}{dt} = M_j \mathbf{g} + \mathbf{F}_j, \\ \mathbf{I}_j \dfrac{d\boldsymbol{\omega}_j}{dt} + \boldsymbol{\omega}_j \times \mathbf{I}_j \boldsymbol{\omega}_j = \mathbf{T}_j, \end{cases} \tag{7}$$

for $j = 1, ..., J$, where in (7):

- M_j is the *mass* of the j^{th} particle.
- $\mathbf{I}_j$ is the *inertia tensor* of the j^{th} particle.

- $\mathbf{F}_j$ is the resultant of the *hydrodynamical forces* acting on the j^{th} particle, i.e.

$$\mathbf{F}_j = (-1)\int_{\partial P_j} \boldsymbol{\sigma}\mathbf{n}d(\partial P_j). \tag{8}$$

- $\mathbf{T}_j$ is the torque at $\mathbf{G}_j$ of the hydrodynamical forces acting on the j^{th} particle, i.e.

$$\mathbf{T}_j = (-1)\int_{\partial P_j} \overrightarrow{\mathbf{G}_j\mathbf{x}} \times \boldsymbol{\sigma}\mathbf{n}d(\partial P_j). \tag{9}$$

- We have

$$\frac{d\mathbf{G}_j}{dt} = \mathbf{V}_j. \tag{10}$$

Equations (7) to (10) have to be completed by the following *initial conditions*:

$$P_j(0) = P_{0j},\ \mathbf{G}_j(0) = \mathbf{G}_{0j},\ \mathbf{V}_j(0) = \mathbf{V}_{0j},\ \boldsymbol{\omega}_j(0) = \boldsymbol{\omega}_{0j},\ \forall j = 1, ..., J. \tag{11}$$

Remark 1 *If the flow-rigid body motion is two-dimensional, or if P_j is a spherical body made of an homogeneous material, then the nonlinear term $\boldsymbol{\omega}_j \times \mathbf{I}_j\boldsymbol{\omega}_j$ vanishes in (7).*

A global variational formulation of the fluid-particle interaction via the virtual power principle.

We suppose, in this section, that the fluid is *Newtonian* of viscosity ν. Let us denote by $P(t)$ the space region occupied at time t by the particles; we have thus $P(t) = \cup_{j=1}^J P_j(t)$. To obtain a *variational formulation* for the system of equations described as described above, we introduce the following *functional space* of *compatible test functions*:

$$\begin{cases} W_0(t) = \{(\mathbf{v}, \mathbf{Y}, \boldsymbol{\theta}) | \mathbf{v} \in (H^1(\Omega \setminus \overline{P(t)}))^d,\ \mathbf{v} = \mathbf{0}\ on\ \Gamma, \\ \qquad \mathbf{Y} = \{\mathbf{Y}_j\}_{j=1}^J,\ \boldsymbol{\theta} = \{\boldsymbol{\theta}_j\}_{j=1}^J,\ with\ \mathbf{Y}_j \in I\!R^d,\ \boldsymbol{\theta}_j \in I\!R^3, \\ \qquad \mathbf{v}(\mathbf{x}, t) = \mathbf{Y}_j + \boldsymbol{\theta}_j \times \overrightarrow{\mathbf{G}_j(t)\mathbf{x}}\ on\ \partial P_j(t),\ \forall j = 1, ..., J\}; \end{cases} \tag{12}$$

in (12) we have $\boldsymbol{\theta}_j = \{0, 0, \theta_j\}$ if $d = 2$.

Applying the *virtual power principle* to the *whole* mixture (i.e., to the fluid *and* the particles) yields the following *global* variational formulation:

$$\begin{cases} \rho_f \int_{\Omega \setminus \overline{P(t)}} \left[\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \boldsymbol{\nabla})\mathbf{u}\right] \cdot \mathbf{v}d\mathbf{x} + 2\nu \int_{\Omega \setminus \overline{P(t)}} \mathbf{D}(\mathbf{u}) : \mathbf{D}(\mathbf{v})d\mathbf{x} \\ -\int_{\Omega \setminus \overline{P(t)}} p\boldsymbol{\nabla} \cdot \mathbf{v}d\mathbf{x} + \sum_{j=1}^J M_j\dot{\mathbf{V}}_j \cdot \mathbf{Y}_j + \sum_{j=1}^J (\mathbf{I}_j\dot{\boldsymbol{\omega}}_j + \boldsymbol{\omega}_j \times \mathbf{I}_j\boldsymbol{\omega}_j) \cdot \boldsymbol{\theta}_j \\ = \rho_f \int_{\Omega \setminus \overline{P(t)}} \mathbf{g} \cdot \mathbf{v}d\mathbf{x} + \sum_{j=1}^J M_j\mathbf{g} \cdot \mathbf{Y}_j,\ \forall \{\mathbf{v}, \mathbf{Y}, \boldsymbol{\theta}\} \in W_0(t), \end{cases} \tag{13}$$

$$\int_{\Omega\setminus\overline{P(t)}} q\boldsymbol{\nabla}\cdot\mathbf{u}(t)d\mathbf{x}=0,\ \forall q\in L^2(\Omega\setminus\overline{P(t)}), \tag{14}$$

$$\mathbf{u}(t)=\mathbf{g}_0(t)\ on\ \Gamma, \tag{15}$$

$$\mathbf{u}(\mathbf{x},t)=\mathbf{V}_j(t)+\boldsymbol{\omega}_j(t)\times\overrightarrow{\mathbf{G}_j(t)\mathbf{x}},\ \forall\mathbf{x}\in\partial P_j(t),\ \forall j=1,...,J, \tag{16}$$

$$\frac{d\mathbf{G}_j}{dt}=\mathbf{V}_j, \tag{17}$$

to be completed by the following *initial conditions*

$$\mathbf{u}(\mathbf{x},0)=\mathbf{u}_0(\mathbf{x}),\ \forall\mathbf{x}\in\Omega\setminus\overline{P(0)}, \tag{18}$$

$$P_j(0)=P_{0j},\ \mathbf{G}_j(0)=\mathbf{G}_{0j},\ \mathbf{V}_j(0)=\mathbf{V}_{0j},\boldsymbol{\omega}_j(0)=\boldsymbol{\omega}_{0j},\ \forall j=1,...,J. \tag{19}$$

In relations (13) to (19):

- We have denoted functions such as $\mathbf{x}\rightarrow\varphi(\mathbf{x},t)$ by $\varphi(t)$.
- We have used the following notation

$$\begin{array}{l}\mathbf{a}\cdot\mathbf{b}=\sum_{k=1}^d a_kb_k,\ \forall\mathbf{a}=\{a_k\}_{k=1}^d,\ \mathbf{b}=\{b_k\}_{k=1}^d,\\ \mathbf{A}:\mathbf{B}=\sum_{k=1}^d\sum_{l=1}^d a_{kl}b_{kl},\ \forall\mathbf{A}=(a_{kl})_{1\le k,l\le d},\ \mathbf{B}=(b_{kl})_{1\le k,l\le d}.\end{array}$$

- We have $\boldsymbol{\omega}_j(t)=\{0,0,\omega_j(t)\}$ if $d=2$.
- We assume that $\mathbf{u}(t)\in(H^1(\Omega\setminus\overline{P(t)}))^d$ and $p(t)\in L^2(\Omega\setminus\overline{P(t)})$.

A distributed Lagrange multiplier based fictitious domain method.

Following references [GHJ+97, GPH+98, PGH+98, GPHJ99, GPH+99, Pan99] we introduce the following variant of the virtual power formulation (13)-(19):

For a.e. $t>0$, *find* $\mathbf{u}(t),\ p(t),\{\mathbf{V}_j(t),\ \mathbf{G}_j(t),\ \boldsymbol{\omega}_j(t)\}_{j=1}^J$, *such that*

$$\begin{cases}\rho_f\displaystyle\int_\Omega\left[\frac{\partial\mathbf{u}}{\partial t}+(\mathbf{u}\cdot\boldsymbol{\nabla})\mathbf{u}\right]\cdot\mathbf{v}d\mathbf{x}-\int_\Omega p\boldsymbol{\nabla}\cdot\mathbf{v}d\mathbf{x}+2\nu\int_\Omega\mathbf{D}(\mathbf{u}):\mathbf{D}(\mathbf{v})d\mathbf{x}\\ +\sum_{j=1}^J(1-\rho_f/\rho_j)\left[M_j\dfrac{d\mathbf{V}_j}{dt}\cdot\mathbf{Y}_j+(\mathbf{I}_j\dfrac{d\boldsymbol{\omega}_j}{dt}+\boldsymbol{\omega}_j\times\mathbf{I}_j\boldsymbol{\omega}_j)\cdot\boldsymbol{\theta}_j\right]\\ =\rho_f\displaystyle\int_\Omega\mathbf{g}\cdot\mathbf{v}d\mathbf{x}+\sum_{j=1}^J(1-\rho_f/\rho_j)M_j\mathbf{g}\cdot\mathbf{Y}_j,\ \forall\{\mathbf{v},\mathbf{Y},\boldsymbol{\theta}\}\in\tilde{W}_0(t),\end{cases} \tag{20}$$

$$\int_\Omega q\boldsymbol{\nabla}\cdot\mathbf{u}d\mathbf{x}=0,\ \forall q\in L^2(\Omega), \tag{21}$$

$$\mathbf{u}=\mathbf{g}_0\ on\ \Gamma, \tag{22}$$

$$\mathbf{u}(\mathbf{x},t)=\mathbf{V}_j(t)+\boldsymbol{\omega}_j(t)\times\overrightarrow{\mathbf{G}_j(t)\mathbf{x}},\ \forall\mathbf{x}\in P_j(t),\ \forall j=1,...,J, \tag{23}$$

$$\frac{d\mathbf{G}_j}{dt}=\mathbf{V}_j, \tag{24}$$

$$P_j(0)=P_{0j},\ \mathbf{V}_j(0)=\mathbf{V}_{0j},\ \boldsymbol{\omega}_j(0)=\boldsymbol{\omega}_{0j},\mathbf{G}_j(0)=\mathbf{G}_{0j},\ \forall j=1,...,J, \tag{25}$$

$$\mathbf{u}(\mathbf{x},0)=\mathbf{u}_0(\mathbf{x}),\ \forall\mathbf{x}\in\Omega\setminus\cup_{j=1}^J\overline{P_{0j}} \tag{26}$$

$$\mathbf{u}(\mathbf{x},0)=\mathbf{V}_{0j}+\boldsymbol{\omega}_{0j}\times\overrightarrow{\mathbf{G}_{0j}\mathbf{x}},\ \forall\mathbf{x}\in\overline{P_{0j}},\ \forall j=1,...,J, \tag{27}$$

with, in relation (20), space $\tilde{W}_0(t)$ defined by

$$\tilde{W}_0(t) = \{(\mathbf{v}, \mathbf{Y}, \boldsymbol{\theta}) | \mathbf{v} \in (H_0^1(\Omega))^d, \mathbf{Y} = \{\mathbf{Y}_j\}_{j=1}^J, \boldsymbol{\theta} = \{\boldsymbol{\theta}_j\}_{j=1}^J, \ \textit{with} \\ \mathbf{Y}_j \in I\!R^d, \ \boldsymbol{\theta}_j \in I\!R^3, \ \mathbf{v}(\mathbf{x}, t) = \mathbf{Y}_j + \boldsymbol{\theta}_j \times \overrightarrow{\mathbf{G}_j(t)\mathbf{x}} \ \textit{in} \ P_j(t), \ \forall j = 1, ..., J\}.$$

In order to relax the *rigid body motion constraint* (23) we are going to employ a family $\{\boldsymbol{\lambda}_j\}_{j=1}^J$ of *Lagrange multipliers* so that $\boldsymbol{\lambda}_j(t) \in \Lambda_j(t)$ with

$$\Lambda_j(t) = (H^1(P_j(t)))^d, \ \forall j = 1.,,,.J. \tag{28}$$

We obtain, thus, the following *fictitious domain formulation with Lagrange multipliers: For a.e. $t > 0$, find $\mathbf{u}(t)$, $p(t)$, $\{\mathbf{V}_j(t), \ \mathbf{G}_j(t), \ \boldsymbol{\omega}_j(t), \ \boldsymbol{\lambda}_j(t)\}_{j=1}^J$, such that*

$$\begin{cases} \mathbf{u}(t) \in (H^1(\Omega))^d, \ \mathbf{u}(t) = \mathbf{g}_0(t) \ \textit{on} \ \Gamma, p(t) \in L^2(\Omega), \\ \mathbf{V}_j(t) \in I\!R^d, \ \mathbf{G}_j(t) \in I\!R^d, \ \boldsymbol{\omega}_j(t) \in I\!R^3, \ \boldsymbol{\lambda}_j(t) \in \Lambda_j(t), \ \forall j = 1, ..., J, \end{cases} \tag{29}$$

and

$$\begin{cases} \rho_f \displaystyle\int_\Omega \left[\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \boldsymbol{\nabla})\mathbf{u}\right] \cdot \mathbf{v} d\mathbf{x} - \int_\Omega p \boldsymbol{\nabla} \cdot \mathbf{v} d\mathbf{x} \\ \quad +2\nu \displaystyle\int_\Omega \mathbf{D}(\mathbf{u}) : \mathbf{D}(\mathbf{v}) d\mathbf{x} + \sum_{j=1}^J (1 - \rho_f/\rho_j) M_j \frac{d\mathbf{V}_j}{dt} \cdot \mathbf{Y}_j \\ \quad + \displaystyle\sum_{j=1}^J (1 - \rho_f/\rho_j)(\mathbf{I}_j \frac{d\boldsymbol{\omega}_j}{dt} + \boldsymbol{\omega}_j \times \mathbf{I}_j \boldsymbol{\omega}_j) \cdot \boldsymbol{\theta}_j \\ \quad - \displaystyle\sum_{j=1}^J < \boldsymbol{\lambda}_j, \ \mathbf{v} - \mathbf{Y}_j - \boldsymbol{\theta}_j \times \overrightarrow{\mathbf{G}_j\mathbf{x}} >_j \\ = \rho_f \displaystyle\int_\Omega \mathbf{g} \cdot \mathbf{v} d\mathbf{x} + \sum_{j=1}^J (1 - \rho_f/\rho_j) M_j \mathbf{g} \cdot \mathbf{Y}_j, \\ \forall \mathbf{v} \in (H_0^1(\Omega))^d, \ \forall \mathbf{Y}_j \in I\!R^d, \ \forall \boldsymbol{\theta}_j \in I\!R^3, \end{cases} \tag{30}$$

$$< \boldsymbol{\mu}_j, \ \mathbf{u}(t) - \mathbf{V}_j(t) - \boldsymbol{\omega}_j(t) \times \overrightarrow{\mathbf{G}_j(t)\mathbf{x}} >_j = 0, \ \forall \boldsymbol{\mu}_j \in \Lambda_j(t), \ \forall j = 1, ..., J, \tag{31}$$

completed by relations (21), (24)-(27). The two most natural choices for $< \cdot, \cdot >_j$ are defined by

$$< \boldsymbol{\mu}, \mathbf{v} >_j = \int_{P_j(t)} (\boldsymbol{\mu} \cdot \mathbf{v} + \delta_j^2 \boldsymbol{\nabla}\boldsymbol{\mu} : \boldsymbol{\nabla}\mathbf{v}) d\mathbf{x}, \ \forall \boldsymbol{\mu} \ \textit{and} \ \mathbf{v} \in \Lambda_j(t), \tag{32}$$

$$< \boldsymbol{\mu}, \mathbf{v} >_j = \int_{P_j(t)} (\boldsymbol{\mu} \cdot \mathbf{v} + \delta_j^2 \mathbf{D}(\boldsymbol{\mu}) : \mathbf{D}(\mathbf{v})) d\mathbf{x}, \ \forall \boldsymbol{\mu} \ \textit{and} \ \mathbf{v} \in \Lambda_j(t), \tag{33}$$

with δ_j a *characteristic length* (the diameter of P_j, for example). Other choices are possible as shown in, e.g, ref. [GPHJ99].

On the discretization of problem (29)-(31).

The *space approximation* (resp., *time discretization*) of problem (29)-(31) by *finite element method* (resp., *operator splitting*) methods is discussed in refs. [GHJ$^+$97, GPH$^+$98, PGH$^+$98, GPHJ99, GPH$^+$99, Pan99]; the above references also include a discussion of the numerical treatment of *particle/particle* and *particle/boundary collisions.*

Numerical Simulations

The Fluidization of a Bed of 1204 Particles.

We consider here the simulation of the *fluidization* in a bed of 1,204 spherical particles. The computational domain is $\Omega = (0, 0.6858) \times (0, 20.3997) \times (0, 44.577)$. The thickness of this bed is slightly larger than the diameter of the particles which is $d = 0.635$, so there is only one layer of balls in the $0x_2$ direction (the above lengths are in centimeters). In [FJL87] many experimental results related to this type of "almost two-dimensional" beds are presented. The fluid is incompressible, viscous, and Newtonian; its density is $\rho_f = 1$ and its viscosity is $\nu_f = 10^{-2}$. We suppose that at $t = 0$ the fluid and the particles are at rest. The boundary condition for the velocity field is

$$\mathbf{u}(t) = \begin{cases} \mathbf{0} & on\ the\ four\ vertical\ walls, \\ 5\begin{pmatrix} 0 \\ 0 \\ 1 - e^{-50t} \end{pmatrix} & on\ the\ two\ horizontal\ walls. \end{cases}$$

The density of the balls is $\rho_s = 1.14$. We suppose that the fluid can enter and leave the bed. The mesh size for the velocity field is $h_\Omega = 0.06858$ (corresponding to 2×10^6 vertices for the velocity mesh), while it is $h_p = 2h_\Omega$ for the pressure (corresponding to 2.9×10^5 vertices for the pressure mesh). The time step is $\Delta t = 10^{-3}$. The initial position of the balls is shown in Figure 2. After starting pushing the balls up, we observe that the inflow creates cavities propagating among the balls in the bed. Since the inflow velocity is much higher than the critical fluidization velocity (of the order of 2.5 here), many balls are pushed directly to the top of the bed. Those balls at the top of the bed are stable and closely packed while the others are circling around at the bottom of the bed. Those numerical results are very close to experimental ones obtained at the University of Minnesota and have been visualized in Figures 2 and 3 (where the lengths are in inches this time). In the simulation, the maximum particle Reynolds number is 1,512 while the maximum averaged particle Reynolds number is 285. The computations were done on an *SGI Origin 2000*, using a partially parallelized code; the computational time is approximately 110 sec./time step.

Sedimentation of 6,400 circular particles in a two-dimensional cavity. Rayleigh-Taylor instability for particulate flow.

The test problem that we consider now concerns the simulation of the motion of 6,400 sedimenting circular disks in the closed cavity $\Omega = (0, 8) \times (0, 12)$. The diameter d of the disks is 1/12 and the position of the disks at time $t = 0$ is shown in Figure 4. The solid fraction in this test case is 34.9%. The disks and the fluid are at rest a time $t = 0$. The density of the fluid is $\rho_f = 1$ and the density of the disks is $\rho_s = 1.1$. The viscosity of the fluid it $\nu_f = 10^{-2}$. The time step is 10^{-3}. The mesh size for the velocity field is $h_\Omega = 1/192$ (the velocity triangulation has thus about 3.5×10^6 vertices) while the pressure mesh size is $h_p = 2h_\Omega$ implying, approximately, 885,000 vertices for the pressure triangulation. For this test problem where many particles "move around" a fine mesh is required essentially everywhere. The computational time per time step

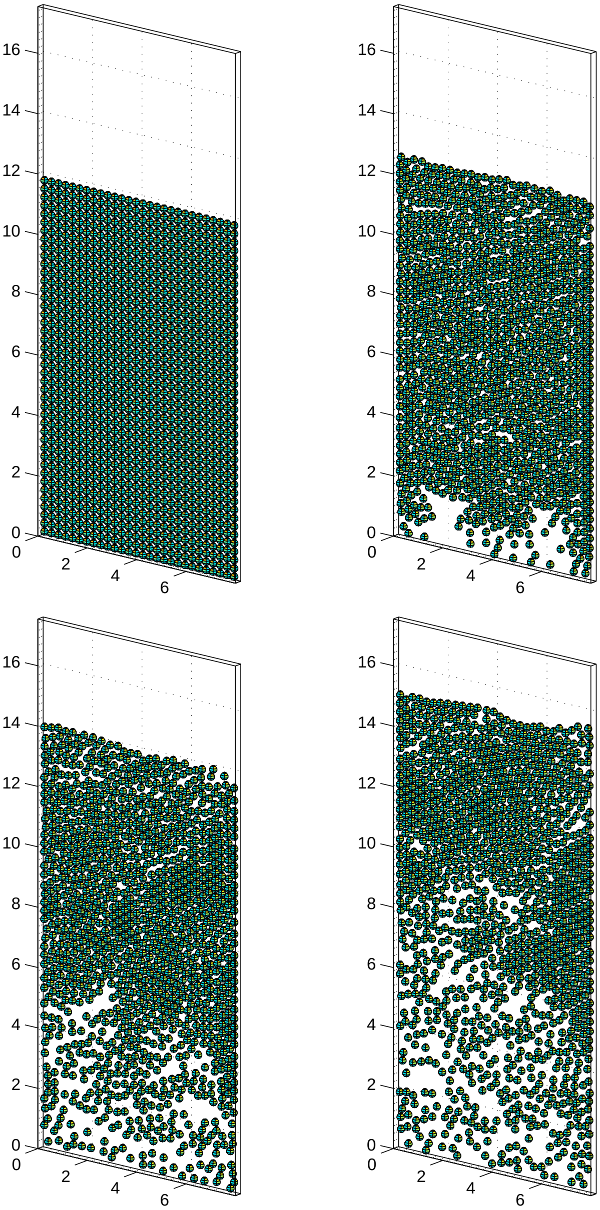


Figure 2: Fluidization of 1,204 spherical particles: positions of the particles at $t = 0$, 1.5, $t = 3$ and 4.5 (from left to right and from top to bottom).

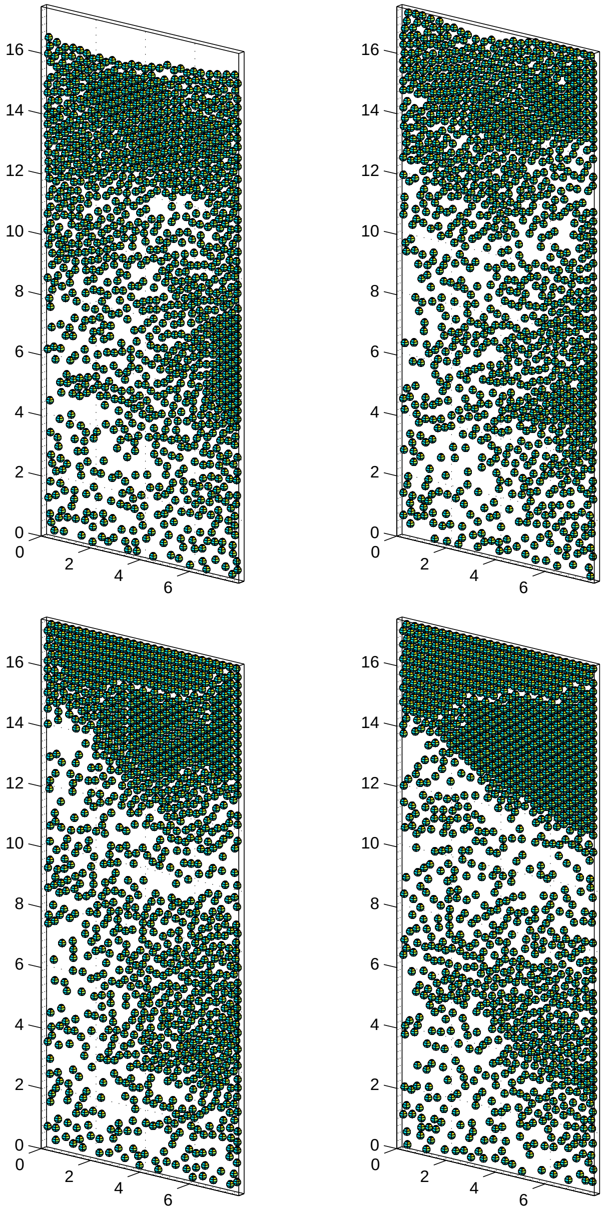


Figure 3: Fluidization of 1,204 spherical particles: positions of the particles at $t = 6$, 7, 8 and 10 (from left to right and from top to bottom).

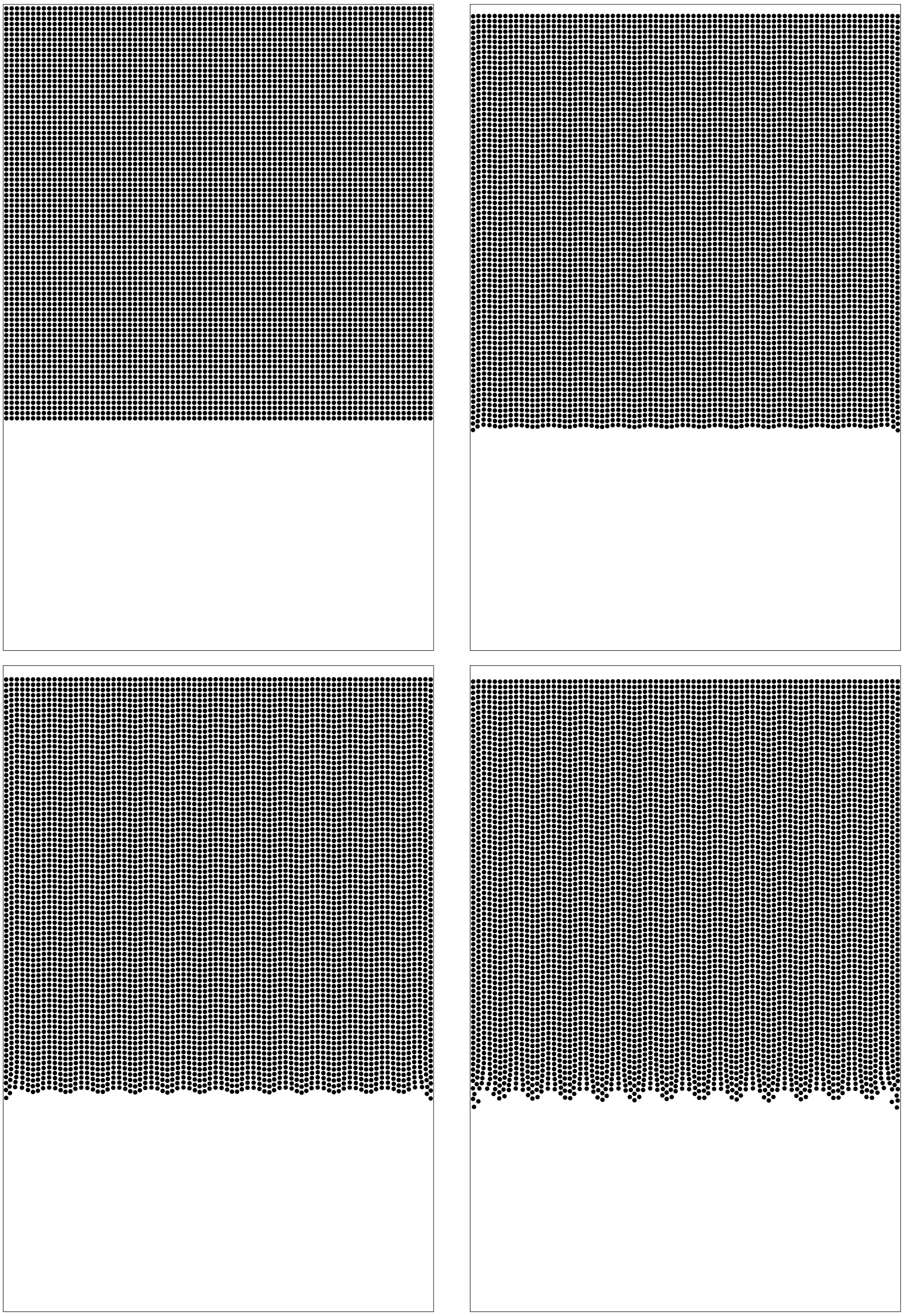


Figure 4: Sedimentation of 6,400 particles: positions at $t = 0$, 0.4, 0.5, 0.6 (from left to right and from top to bottom), and visualization of the Rayleigh-Taylor instability.

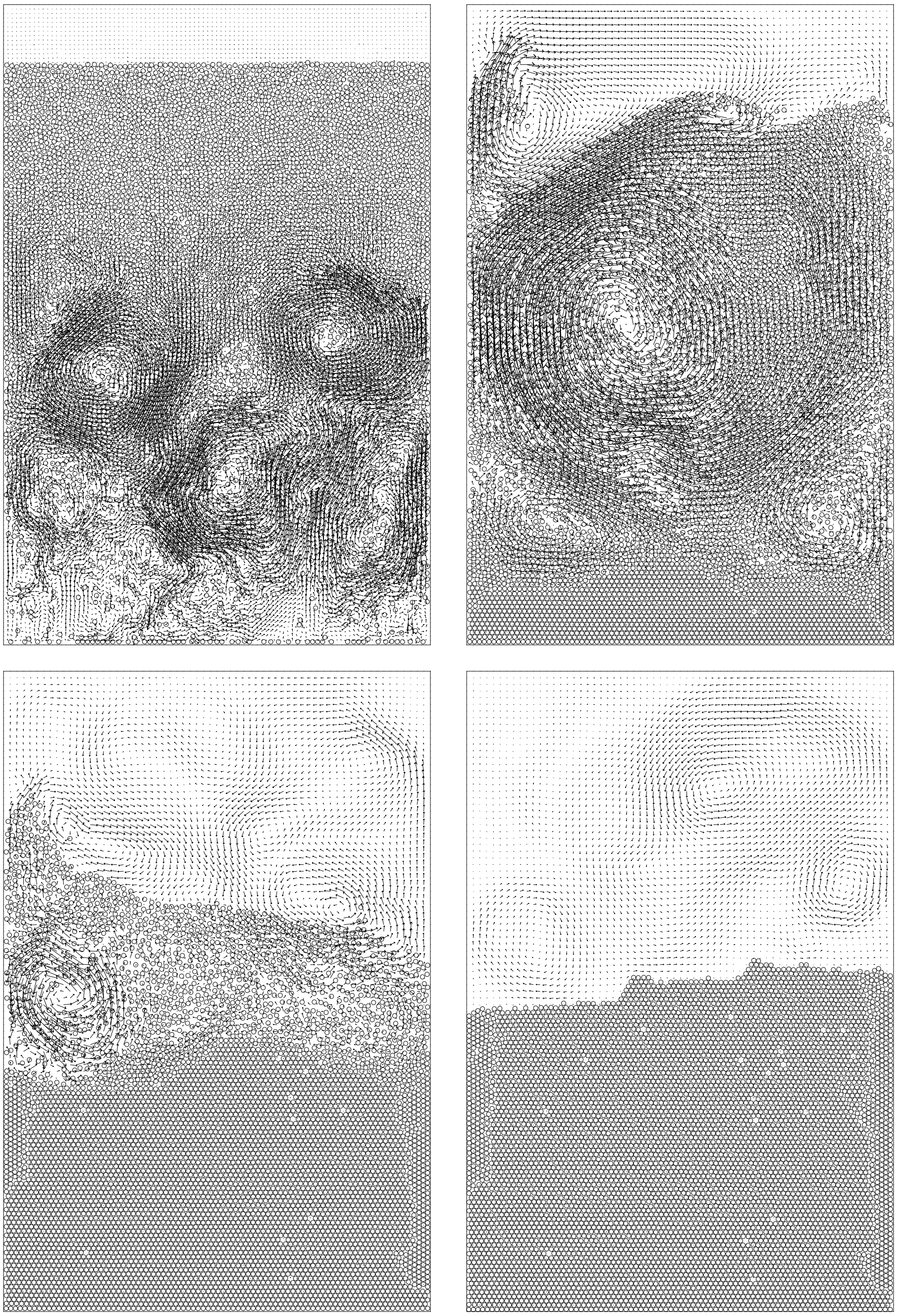


Figure 5: Sedimentation of 6,400 particles: positions at $t = 2.6$, 5, 9, 13 (from left to right and from top to bottom), and visualization of the Rayleigh-Taylor instability.

is approximately 10 min. on a *DEC Alpha 500-au* workstation, implying that to simulate *one time unit* of the phenomenon under consideration *we need, practically, a full week.* The evolution of the 6,400 disks sedimenting in Ω is shown in Figures 4 and 5. The maximum particle Reynolds number in the entire evolution is 72.64. Figure 4 clearly shows the development of a "text-book" *Rayleigh-Taylor instability.* This instability develops into a fingering phenomenon and many symmetry breaking and other bifurcation phenomena, including drafting, kissing and tumbling, take place at various scales and times; similarly vortices of various scales develop and for a while the phenomenon is clearly chaotic, which is not surprising after all for a 6,400-body problem. Finally, the particles settle at the bottom of the cavity and the fluid returns to rest.

Acknowledgments

We acknowledge the helpful comments and suggestions of E. J. Dean, V. Girault, J. He, T.I. Hesla, Y. Kuznetsov, J. Periaux, G. Rodin, A. Sameh, V. Sarin, and P. Singh, and also the support of the Supercomputing Institute at the University of Minnesota concerning the use of an SGI Origin2000. We acknowledge also the support of the NSF (Grants ECS-9527123, CTS-9873236, and DMS-9973318) and Dassault Aviation.

References

[FJL87]A. F. Fortes, D. D. Joseph, and T. S. Lundgren. Nonlinear mechanics of fluidization of beds of spherical particles. *J. Fluid Mech.*, 177:467–483, 1987.

[GHJ$^+$97]R. Glowinski, T. I. Hesla, D.D. Joseph, T.W. Pan, and J. Periaux. Distributed Lagrange multiplier methods for particulate flows. In M.O. Bristeau, G.J. Etgen, W. Fitzgibbon, J.L. Lions, J. Periaux, and M.F. Wheeler, editors, *Computational Science for the 21st Century*, pages 270–279, Chichester, 1997. Wiley.

[GPH$^+$98]R. Glowinski, T.W. Pan, T.I. Hesla, D.D. Joseph, and J. Periaux. A fictitious domain method with distributed Lagrange multipliers for the numerical simulation of particulate flow. In J. Mandel, C. Farhat, , and X.C. Cai, editors, *Domain Decomposition Methods 10*, pages 121–137, Providence, RI, 1998. AMS.

[GPH$^+$99]R. Glowinski, T. W. Pan, T. I. Hesla, D. D. Joseph, and J. Periaux. A distributed Lagrange multiplier/fictitious domain method for flows around moving rigid bodies: Application to particulate flow. *Int. J. Numer. Meth. Fluids*, 30:1043–1066, 1999.

[GPHJ99]R. Glowinski, T.W. Pan, T.I. Hesla, and D.D. Joseph. A distributed Lagrange multiplier/fictitious domain method for particulate flow. *International Journal of Multiphase Flow*, 25:755–794, 1999.

[Pan99]T. W. Pan. Numerical simulation of the motion of a ball falling in an incompressible viscous fluid. *C. R. Acad. Sci. Paris*, 327, Série 2, b:1035–1038, 1999.

[PGH$^+$98]T. W. Pan, R. Glowinski, T. I. Hesla, D. D. Joseph, and J. Periaux. Numerical simulation of the Rayleigh-Taylor instability for particulate flow. In M. Hafez and J. C. Heinrich, editors, *Proceedings of the Tenth International Conference on Finite Elements in Fluids*, pages 217–222, 1998.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau,

37. Parallel 3D Maxwell Solvers based on Domain Decomposition Data Distribution

G. Haase[1], M. Kuhn[2], U. Langer[3]

Introduction

The most efficient solvers for finite element (fe) equations are certainly multigrid, or multilevel methods, and domain decomposition methods using local multigrid solvers. Typically, the multigrid convergence rate is independent of the mesh size parameter, and the arithmetical complexity grows linearly with the number of unknowns. However, the standard multigrid algorithms fail for the Maxwell finite element equations in the sense that the convergence rate deteriorates as the mesh-size decreases. To overcome this drawback, R. Hiptmair proposed to modify the smoothing iteration by adding a smoothing step in the discrete potential space [Hip99]. Similarly, D. Arnold, R. Falk and R. Winther suggested a special block smoother that has the same effect [AFW00].

The parallelization of these or, more precisely, of appropriately modified multigrid solvers is certainly the only principle way to enhance the efficiency of these algorithms. Due to the peculiarities of the multigrid methods for the Maxwell equations, the parallelization is not straightforward. In this paper, we propose a unified approach to the parallelization of multigrid methods and domain decomposition methods. In order to develop a basic parallel Maxwell solver that can be used for more advanced problems as basic module, it is sufficient to consider the magnetostatic case. In the magnetostatic case, the Maxwell equation can be reduced to the curl-curl–equation that is not uniquely solvable because of the large kernel of the curl-operator (potential fields). In practice, a gauging condition is imposed in order to pick out a unique solution. The so-called Coulomb gauging aims at a divergence-free solution (vector potential). The weak formulation of the curl-curl–equation and the gauging condition together with a clever regularization leads to a regularized mixed variational formulation of the magnetostatic Maxwell equations in $\mathrm{H}_0(\mathbf{curl}) \times \mathrm{H}_0^1(\Omega)$ that has a unique solution. The discretization by the Nédélec and Lagrange finite elements results in a large, sparse, symmetric, but indefinite system of finite element equations. Eliminating the Lagrange multiplier from the mixed finite element equations, we arrive at a symmetric and positive definite (spd) problem that can be solved by some parallel multigrid preconditioned conjugate gradient (pcg) method. More precisely, this pcg solver contains a standard scaled Laplace multigrid regularizer in the regularization part and a special multigrid preconditioner for the regularized Nédélec finite element equations that we want to solve. (see second section). The parallelization of the

[1] Johannes Kepler University Linz, Institute of Analysis and Computational Mathematics, ghaase@numa.uni-linz.ac.at

[2] Johannes Kepler University Linz, SFB “Numerical and Symbolic Scientific Computing”, kuhn@sfb013.uni-linz.ac.at

[3] Johannes Kepler University Linz, Institute of Analysis and Computational Mathematics, ulanger@numa.uni-linz.ac.at

pcg algorithm, the Laplace multigrid regularizer and the multigrid preconditioner are based on a unified domain decomposition (dd) data distribution concept that will be briefly described in the following two sections. From the parallelization point of view, we prefer Hiptmair's multigrid method with some modifications for the construction of the special multigrid preconditioner. We also propose a concept for coupling finite elements with boundary elements in 3D. As in 2D, a really efficient parallel solver should be based on a hybrid parallelization concept using some Dirichlet dd preconditioner the components of which are a dd parallelized global multigrid preconditioner for the finite element part and algebraically parallelized components for the boundary element parts. The final part contains some results of our numerical experiments on a parallel machine with distributed memory that show the high efficiency of our approach for a real-life application.

3D Magnetostatic Field Problems

The magnetostatic equations, in which we are interested throughout the paper, can be rewritten as

$$\operatorname{curl}(H) = J, \quad H = \nu B, \quad \operatorname{div}(B) = 0, \tag{1}$$

where H and B denote the magnetic field intensity and the magnetic flux density, respectively. The permeability μ ($\nu := 1/\mu \geq \nu_{min} > 0$) and current density J are given. Furthermore, we note that the current density J is physically divergence-free, i.e., $\operatorname{div}(J) = 0$. Theoretically, the computational domain Ω coincides with the space $\mathbb{R}^3$ in any case. The behavior of the magnetic field at infinity is described by radiation conditions. In practice, one may often simplify the problem by considering a bounded, simply connected computational domain $\Omega \subset \mathbb{R}^3$ with Lipschitz boundary $\Gamma = \partial\Omega$ and by replacing the radiation condition by the boundary condition

$$B \cdot n = 0 \quad \text{on } \partial\Omega, \tag{2}$$

where n stands for the unit outward normal with respect to $\partial\Omega$. Introducing some vector potential u for the B-field $B = \operatorname{curl}(u)$ and taking into account the Coulomb gauging condition $\operatorname{div}(u) = 0$ ensuring uniqueness, we arrive at the following mixed variational formulation that is fundamental for our approach to the numerical solution of the magnetostatic Maxwell equations (1):
Find $(u, p) \in X \times M := H_0\,(\operatorname{curl}, \Omega) \times H_0^1(\Omega)$ such that

$$\begin{aligned} a(u,v) + b(v,p) &= \langle f, v\rangle \quad \forall\, v \in H_0\,(\operatorname{curl}, \Omega), & (3)\\ b(u,q) &= 0 \qquad\quad \forall\, q \in H_0^1(\Omega), & (4) \end{aligned}$$

where $a(u,v) := \int_\Omega \nu \operatorname{curl}(u) \cdot \operatorname{curl}(v)\, dx$, $b(v,p) := \int_\Omega v \cdot \nabla p\, dx$, and $\langle f, v\rangle := \int_\Omega J \cdot v\, dx$. Now it is not difficult to conclude from the Brezzi-Babuška theory that the mixed variational problem (3) - (4) has a unique solution. Moreover, choosing $v = \nabla p \in H_0(\operatorname{curl}, \Omega)$ in (3), we immediately observe that $p = 0$. This simple observation is crucial for our approach. Indeed, adding an arbitrary spd bilinear form $c(\cdot,\cdot) : M \times M \to \mathbb{R}^1$ to the second equation of our mixed variational problem (3) - (4) we arrive at the equivalent mixed variational problem: Find $(u,p) \in X \times M$ such that

$$\begin{aligned} a(u,v) + b(v,p) &= \langle f, v\rangle \quad \forall\, v \in X, && (5)\\ b(u,q) - c(p,q) &= 0 \qquad\quad \forall\, q \in M. && (6)\end{aligned}$$

Let now be $X_h := \mathcal{N}_h^1 \subset X$ and $M_h := \mathcal{S}_h^1 \subset M$ the lowest order edge element space (see [N86́]) and the space of piecewise linear nodal elements on a shape-regular tetrahedral triangularization of Ω with the mesh-width h, respectively [Cia78]. Then the mixed fe approximation to the regularized mixed variational problem (5) - (6) leads us to the following symmetric, but indefinite system

$$\begin{pmatrix} A & B^T \\ B & -C \end{pmatrix} \begin{pmatrix} \underline{u}_h \\ \underline{p}_h \end{pmatrix} = \begin{pmatrix} \underline{f}_h \\ 0 \end{pmatrix} \tag{7}$$

of linear finite element equations for defining the edge unknowns $\underline{u}_h$ and the nodal unknowns $\underline{p}_h$, where the matrices A, B, C and the first component $\underline{f}_h$ of the right-hand side are derived from the bilinear forms $a(\cdot,\cdot)$, $b(\cdot,\cdot)$, $c(\cdot,\cdot)$, and the linear form $\langle f,\cdot\rangle$, respectively.

Eliminating $\underline{p}_h = C^{-1}B\underline{u}_h$ from the second equation in (7) and inserting it into the first equation, we obtain the spd Schur complement system

$$G\underline{u}_h := (A + B^T C^{-1} B)\underline{u}_h = \underline{f}_h. \tag{8}$$

Let $\widetilde{C}$ be some spd matrix that is spectrally equivalent to C (briefly, $\widetilde{C} \approx C$). Then the original Schur complement system (8) is equivalent to the modified Schur complement system

$$\widetilde{G}\underline{u}_h := (A + B^T \widetilde{C}^{-1} B)\underline{u}_h = \underline{f}_h. \tag{9}$$

Instead of solving the symmetric, but indefinite system (7), we solve the spd modified Schur complement system (9).

Let us choose the spd bilinear form

$$c(p,q) := \frac{1}{\nu_{\min}} \int\limits_{\Omega} \nabla p \, \nabla q \, dx, \tag{10}$$

corresponding to the Laplace operator scaled by $1/\nu_{\min}$, and let us consider a spd preconditioner C_H for the spd matrix $H := A + \widetilde{M}$, where $\widetilde{M}$ is here the appropriately scaled mass matrix in X_h defined by $(\widetilde{M}\underline{u}_h, \underline{v}_h) := \nu_{\min} \int_\Omega u_h \, v_h \, dx$. The discrete LBB–condition and the spectral equivalence $C_H \approx H$ imply that $C_H \approx G \approx \widetilde{G}$ (see [Kuh98] for the detailed proof). Once a good preconditioner C_H and an appropriate regularizer $\widetilde{C}$ is available, we can solve the modified Schur complement system (9) by the pcg method. In practice, we choose the multigrid preconditioner $C_H := H(I - M_H)^{-1}$ and the multigrid regularizer $\widetilde{C} := C_C := C(I - M_C)^{-1}$, where M_H and M_C are the corresponding multigrid iteration operators with respect to H and C. Choosing appropriate symmetric multigrid cycles, we can now conclude from the results of [AFW00, Hac85, Hip99] that the pcg method is asymptotically optimal with respect to the operation count and to the memory demand [JLM$^+$89]. The numerical results obtained from the serial implementation of this algorithm confirm this statement [KLS00]. In this paper, we are interested in the parallel implementation of this algorithm. The parallelization of this algorithm is far from being straightforward because of the peculiarities connected with the multigrid regularizer C_C and with the special multigrid preconditioner C_H.

A Unified Data Distribution Concept

Vector and matrix types

We decompose $\overline{\Omega}$ in P non-overlapping subdomains $\overline{\Omega}_s$ which are discretized by a mesh $\tau_{h,s}$, such that the whole triangulation $\tau_h = \bigcup_{s=1}^{P} \tau_{h,s}$ of Ω is conform. The index set of the N_s unknowns in Ω_s is denoted by ω_s. The mapping of a vector $\underline{u} \in \mathbb{R}^N$ in global numbering onto a local vector $\underline{u}_s \in \mathbb{R}^{N_s}$ in subdomain $\overline{\Omega}_s$ $(s = \overline{1,P})$ is represented symbolically by subdomain connectivity matrices $\mathcal{A}_s$ of dimension $N_s \times N$ with entries $\mathcal{A}_s^{[i,j]} := 1$ if $j \in \omega$ is the global number of $i \in \omega_s$ and $\mathcal{A}_s^{[i,j]} := 0$ otherwise.

The index set of all those subdomains, an unknown $u^{[j]}$, $j \in \omega$ belongs to, is denoted by $\sigma^{[j]} := \{s \,|\, \exists i \in \omega_s : \mathcal{A}_s^{[i,j]} \neq 0\}$. We store the data of a vector component $u^{[i]}$ in the subdomain Ω_s if $s \in \sigma^{[i]}$.

There are two opportunities to store those components and finally that vector. A vector $\underline{\mathfrak{u}}$ is called an accumulated vector if each vector component $\mathfrak{u}^{[i]}$ is stored in all subdomains Ω_s, $s \in \sigma^{[i]}$ with its full value. The local vectors $\underline{\mathfrak{u}}_s$ can be represented as $\underline{\mathfrak{u}}_s := \mathcal{A}_s \cdot \underline{\mathfrak{u}}$. We name a vector $\underline{\mathsf{r}}$ as distributed vector if it is decomposed into local vectors $\underline{\mathsf{r}}_s$ such that $\underline{\mathsf{r}} = \sum_{s=1}^{P} \mathcal{A}_s^T \cdot \underline{\mathsf{r}}_s$ holds, i.e., all subdomains Ω_s, $s \in \sigma^{[i]}$ store only $\underline{\mathsf{r}}_s$ and possess a portion of the full vector value $\mathsf{r}^{[i]}$ which can be determined only by taking the sum. The conversion of a distributed vector $\underline{\mathsf{v}}$ into an accumulated vector $\underline{\mathfrak{w}}$ can be done by evaluating the sum above and restrict the result afterwards, i.e.,

$$\underline{\mathfrak{w}} \leftarrow \underline{\mathsf{v}} \qquad : \qquad \underline{\mathfrak{w}}_s := \mathcal{A}_s \cdot \underline{\mathfrak{w}} = \mathcal{A}_s \cdot \sum_{s=1}^{P} \mathcal{A}_s^T \cdot \underline{\mathsf{v}}_s \ . \tag{11}$$

With respect to an element-wise domain decomposition, the matrix defined by the bilinear form in (3) can also be stored in two ways. A matrix $\mathfrak{M}$ is called accumulated if its local restrictions $\mathfrak{M}_s$ possess the full entries of it, and we can write $\mathfrak{M}_s := \mathcal{A}_s \cdot \mathfrak{M} \cdot \mathcal{A}_s^T$. We call a matrix K distributed if we have locally stored matrices K_s such that $\mathsf{K} := \sum_{s=1}^{P} \mathcal{A}_s^T \cdot \mathsf{K}_s \cdot \mathcal{A}_s$ holds, i.e., each subdomain $\overline{\Omega}_s$ stores only a part of its full values. We obtain distributed system matrices K_s automatically in our approach.

Basic operations

The inner product of different type vectors requires one global reduce operation of the local inner products, for details see [Haa98, Haa99, HLM91]. The multiplication of a distributed matrix with an accumulated vector results in a distributed vector and its local realization $\underline{\mathsf{v}}_s = \mathsf{K}_s \cdot \underline{\mathfrak{w}}_s$ requires no communication at all:

$$\langle \underline{\mathfrak{w}}, \underline{\mathsf{r}} \rangle \;=\; \sum_{s=1}^{P} \langle \underline{\mathfrak{w}}_s, \underline{\mathsf{r}}_s \rangle \qquad \text{and} \qquad \mathsf{K} \cdot \underline{\mathfrak{w}} \;=\; \underline{\mathsf{v}} \;. \tag{12}$$

The situation changes if we use an accumulated matrix $\mathfrak{M}$. If the pattern of $\mathfrak{M}$ fulfills the condition

$$\forall i, j \in \omega \;: \qquad \sigma^{[i]} \not\subseteq \sigma^{[j]} \implies \mathfrak{M}^{[i,j]} = 0 \;, \tag{13}$$

then no communication is needed for the operations $\underline{\mathfrak{w}} = \mathfrak{M} \cdot \underline{\mathfrak{u}}$ and $\underline{\mathsf{d}} = \mathfrak{M}^T \cdot \underline{\mathsf{r}}$, i.e., we performed locally $\underline{\mathfrak{w}}_s = \mathfrak{M}_s \cdot \underline{\mathfrak{u}}_s$ and $\underline{\mathsf{d}}_s = \mathfrak{M}_s^T \cdot \underline{\mathsf{r}}_s$, $\forall s = \overline{1, P}$.

Basic algorithms

The operations (12) allow us already to formulate a parallel pcg algorithm for solving the matrix equation $Ku = f$ with a preconditioner C_K. Besides the inner products, only the preconditioning step $\underline{\mathfrak{w}} \Leftarrow C_K^{-1} \cdot \underline{\mathsf{r}}$ involves communication indicated by using $\Leftarrow$ instead of $\leftarrow$. In the case of $C_K = I$, i.e., no preconditioning, this step reduces to a type conversion (11) involving communication. We require that the communication costs for applying any other preconditioner C_K^{-1} are in the same range.

Algorithm 1 Parallel pcg method PCG($\mathsf{K}, \underline{\mathfrak{u}}, \underline{\mathsf{f}}, C_K$)

```
repeat
  $\underline{\mathsf{v}} \leftarrow \mathsf{K} \cdot \underline{\mathfrak{s}}$
  $\alpha \leftarrow \sigma / \langle \underline{\mathfrak{s}}, \underline{\mathsf{v}} \rangle$
  $\underline{\mathfrak{u}} \leftarrow \underline{\mathfrak{u}} + \alpha \cdot \underline{\mathfrak{s}}$
  $\underline{\mathsf{r}} \leftarrow \underline{\mathsf{r}} - \alpha \cdot \underline{\mathsf{v}}$
  $\underline{\mathfrak{w}} \Leftarrow C_K^{-1} \cdot \underline{\mathsf{r}}$
  $\sigma \leftarrow \langle \underline{\mathfrak{w}}, \underline{\mathsf{r}} \rangle$ , $\beta \leftarrow \sigma / \sigma_{\text{old}}$ , $\sigma_{\text{old}} \leftarrow \sigma$
  $\underline{\mathfrak{s}} \leftarrow \underline{\mathfrak{w}} + \beta \cdot \underline{\mathfrak{s}}$
until termination
```

One possible choice for the preconditioner is $C_K^{-1} = (I - M_K)K^{-1}$, with M_K being the multigrid iteration operator for K. The parallel multigrid iteration is presented in Alg. 2, where ℓ denotes the level such that $\ell = 1$ stands for the coarsest grid. The algorithm needs a smoother SMOOTH with a good parallel performance, e.g., a block Jacobi smoother with Gauss-Seidel smoothing in blocks containing interior unknowns of the subdomains. Furthermore, the interpolation $\mathfrak{P}$ has to fulfill the pattern condition (13) and we take $\mathfrak{P}^T$ as restriction. The coarse grid system can be solved directly or by some iterative method similar to the PCG in Alg. 1. Despite the coarse grid solver, only the smoothing sweep requires communication.

Parallel Multigrid Maxwell Solver

We want to solve (9) by the pcg algorithm (Alg. 1) using a multigrid preconditioner (Alg. 2) for the realization of C_K^{-1}. In this section, we will discuss how these compo-

Algorithm 2 Parallel multigrid PMG($\mathsf{K}, \underline{\mathfrak{u}}, \underline{\mathsf{f}}, \ell$)

```
if $\ell == 1$ then
    $\underline{\mathfrak{u}} \Leftarrow$ SOLVE $( \sum_{s=1}^{P} \mathcal{A}_s^T \mathsf{K}_s \mathcal{A}_s \cdot \underline{\mathfrak{u}} \; = \; \underline{\mathsf{f}} )$
else
    $\widetilde{\underline{\mathfrak{u}}} \Leftarrow$ SMOOTH$(\mathsf{K}, \underline{\mathfrak{u}}, \underline{\mathsf{f}})$
    $\underline{\mathsf{d}} \leftarrow \underline{\mathsf{f}} - \mathsf{K} \cdot \widetilde{\underline{\mathfrak{u}}}$
    $\underline{\mathsf{d}}^H \leftarrow \mathfrak{P}^T \cdot \underline{\mathsf{d}}$
    $\underline{\mathfrak{w}}^H \Leftarrow$ PMG$(\mathsf{K}^H, \underline{\mathfrak{w}}^H \leftarrow 0, \underline{\mathsf{d}}^H, \ell - 1)$
    $\underline{\mathfrak{w}} \leftarrow \mathfrak{P} \cdot \underline{\mathfrak{w}}^H$
    $\underline{\mathfrak{u}} \Leftarrow$ SMOOTH$^T(\mathsf{K}, \widetilde{\underline{\mathfrak{u}}} + \underline{\mathfrak{w}}, \underline{\mathsf{f}})$
end if
```

nents have to be adapted to the case of our Maxwell solver presented in the second section.

Our reduced primal formulation (9) has been derived from (7). As discussed in the previous section, the matrices are generated locally, such that the local components A_s, B_s, C_s are available. Denoting the subdomain connectivity matrices with respect to the spaces $\mathbb{X}_h := \mathbb{R}^{n_h}$ and $\mathbb{M}_h := \mathbb{R}^{m_h}$ by $\mathcal{A}_{\mathbb{X},s}$ and $\mathcal{A}_{\mathbb{M},s}$, respectively, we have the following relations:

$$\mathsf{A} = \sum_{s=1}^{P} \mathcal{A}_{\mathbb{X},s} \mathsf{A}_s \mathcal{A}_{\mathbb{X},s}^T, \quad \mathsf{B} = \sum_{s=1}^{P} \mathcal{A}_{\mathbb{M},s} \mathsf{B}_s \mathcal{A}_{\mathbb{X},s}^T, \quad \mathsf{C} = \sum_{s=1}^{P} \mathcal{A}_{\mathbb{M},s} \mathsf{C}_s \mathcal{A}_{\mathbb{M},s}^T.$$

The system matrix in (9) is defined by $\widetilde{\mathsf{G}} := \mathsf{A} + \mathsf{B}^\mathsf{T} \widetilde{C}^{-1} \mathsf{B}$, where $\widetilde{C}$ is a preconditioner for C. In order to apply PCG$(\widetilde{\mathsf{G}}, \underline{\mathfrak{u}}, \underline{\mathsf{f}}, C_{\widetilde{G}})$ we explain in Alg. 3 how the matrix-by-vector operation is defined for the distributed matrix $\widetilde{\mathsf{G}}$. Hereby, the required operation $\widetilde{C}^{-1}$ is being realized by one multigrid iteration step PMG$(\mathsf{C}, \underline{\mathfrak{p}}, \underline{\mathsf{q}}, \ell)$ in the space $\mathbb{M}_h$. Although the matrix $\widetilde{\mathsf{G}}$ is distributed, the corresponding matrix-by-vector operation requires as many communications as one multigrid iteration step for C in $\mathbb{M}_h$.

Algorithm 3 The operation $\underline{\mathsf{v}} \Leftarrow \widetilde{\mathsf{G}} \cdot \underline{\mathfrak{s}}$.

```
$\underline{\mathsf{q}} \leftarrow \mathsf{B} \cdot \underline{\mathfrak{s}}$
$\underline{\mathfrak{p}} \Leftarrow$ PMG$(\mathsf{C}, \underline{\mathfrak{p}}, \underline{\mathsf{q}}, \ell)$
$\underline{\mathsf{v}} \leftarrow \mathsf{A} \cdot \underline{\mathfrak{s}} + \mathsf{B}^\mathsf{T} \cdot \underline{\mathfrak{p}}$
```

Furthermore, the operation C_K^{-1} in Alg. 1 is now realized by one iteration step of HYBRIDPMG$(\widetilde{\mathsf{H}}, \mathsf{C}, \underline{\mathfrak{u}}, \underline{\mathsf{f}}, \ell)$ defined in Alg. 4. Comparing Alg. 4 and Alg. 2 we observe that only the smoother has to be adapted to our special application.

In particular we use a hybrid smoother as proposed in [Hip99] which is suitable for parallelization. As in [Hip99], we introduce the lifting operator $\mathfrak{L} : \; \mathbb{X}_h \to \mathbb{M}_h$ where $\mathfrak{L}^{[i,j]} := -1$, $\mathfrak{L}^{[i,k]} := 1$ if the oriented edge with the unknown index i in $\mathbb{X}_h$ connects the two unknowns with the indices j and k in $\mathbb{M}_h$. Otherwise we have $\mathfrak{L}^{[i,j]} := 0$. We observe that $\mathfrak{L}^T$ satisfies the pattern condition (13). Now Alg. 5 is the correct definition of the parallel hybrid smoother HYBRIDSMOOTH$(\mathsf{H}, \mathsf{C}, \underline{\mathfrak{u}}, \underline{\mathsf{f}})$. Note, using the

Algorithm 4 Parallel hybrid multigrid HYBRIDPMG$(\mathsf{H}, \mathsf{C}, \underline{\mathfrak{u}}, \underline{\mathsf{f}}, \ell)$

```
if $\ell == 1$ then
    $\underline{\mathfrak{u}} \Leftarrow \text{Solve}\,( \sum_{s=1}^{P} \mathcal{A}_s^T \mathsf{H}_s \mathcal{A}_s \cdot \underline{\mathfrak{u}} \;=\; \underline{\mathsf{f}} )$
else
    $\underline{\widetilde{\mathfrak{u}}} \Leftarrow \text{HybridSmooth}(\mathsf{H}, \mathsf{C}, \underline{\mathfrak{u}}, \underline{\mathsf{f}})$
    $\underline{\mathsf{d}} \leftarrow \underline{\mathsf{f}} - \mathsf{H} \cdot \underline{\widetilde{\mathfrak{u}}}$
    $\underline{\mathsf{d}}^H \leftarrow \mathfrak{P}^T \cdot \underline{\mathsf{d}}$
    $\underline{\mathfrak{w}}^H \leftarrow 0$
    $\underline{\widetilde{\mathfrak{w}}}^H \Leftarrow \text{HYBRIDPMG}(\mathsf{H}^H, \mathsf{C}^H, \underline{\mathfrak{w}}^H, \underline{\mathsf{d}}^H, \ell - 1)$
    $\underline{\mathfrak{w}} \leftarrow \mathfrak{P} \cdot \underline{\widetilde{\mathfrak{w}}}^H$
    $\underline{\widehat{\mathfrak{u}}} \leftarrow \underline{\widetilde{\mathfrak{u}}} + \underline{\mathfrak{w}}$
    $\underline{\mathfrak{u}} \Leftarrow \text{HybridSmooth}^T(\mathsf{H}, \mathsf{C}, \underline{\widehat{\mathfrak{u}}}, \underline{\mathsf{f}})$
end if
```

Algorithm 5 Parallel hybrid smoother HYBRIDSMOOTH$(\mathsf{H}, \mathsf{C}, \underline{\mathfrak{u}}, \underline{\mathsf{f}})$

```
$\underline{\widetilde{\mathfrak{u}}} \Leftarrow \text{Smooth}(\mathsf{H}, \underline{\mathfrak{u}}, \underline{\mathsf{f}})$
$\underline{\mathsf{q}} \leftarrow \mathfrak{L} \cdot (\underline{\mathsf{f}} - \mathsf{H} \cdot \underline{\widetilde{\mathfrak{u}}})$
$\underline{\mathfrak{p}} \leftarrow 0$
$\underline{\widetilde{\mathfrak{p}}} \Leftarrow \text{Smooth}(\nu_{\min}^2 \cdot \mathsf{C}, \underline{\mathfrak{p}}, \underline{\mathsf{q}})$
$\underline{\mathfrak{u}} \leftarrow \underline{\widetilde{\mathfrak{u}}} + \mathfrak{L}^T \cdot \underline{\widetilde{\mathfrak{p}}}$
```

matrix C derived from (10) for defining the smoother in $\mathbb{M}_h$ we have to use the correct scaling by $\nu_{\min}^2$ corresponding to the definition of scaled mass matrix $\widetilde{M}$. Now, the smoother SMOOTH can be any standard smoother with a good parallel performance, e.g., a block Jacobi smoother with Gauss-Seidel smoothing in blocks containing interior unknowns of the subdomains. Since HYBRIDSMOOTH involves at least two smoothing steps, one in $\mathbb{X}_h$ and one in $\mathbb{M}_h$, at least two subsequent communications are required.

Note, the post–smoothing step HYBRIDSMOOTH$^T(\mathsf{H}, \mathsf{C}, \underline{\mathfrak{u}}, \underline{\mathsf{f}})$. in Alg. 4 is obtained from Alg. 5 by executing step 1 after steps 2-3-4-5 instead of executing the given order 1-2-3-4-5.

Parallel Domain Decomposition Maxwell Solver

If one is interested in the exterior magnetic field, then the coupling of the FEM with the BEM is certainly the natural technique to handle this problem. For simplicity of the presentation, let us consider the case where the magnetic sources and the ferromagnetic materials are located in some bounded and simply connected Lipschitz domain Ω_F where we will use the FEM for approximating the magnetic field. Thus, we suppose that in the exterior BEM subdomain $\Omega_B := (\bar{\Omega}_F)^c$ the electric current density vanishes, i.e., $J = 0$, and $\nu = \nu_B > 0$ (air). We can again introduce the vector potential u for the B–field $B = \text{curl}(u)$ in Ω_F. However, in the exterior domain Ω_B, the H–field can now be represented as a gradient field of some scalar potential φ, i.e. $H = \text{grad}(\varphi)$ in Ω_B.

Therefore, in the exterior subdomain Ω_B, the magnetostatic Maxwell equations (1) are essentially reduced to the scaled Laplace equation for the scalar potential φ. The Cauchy data for the solution of this equation are related by Calderon's integral equations $\varphi = (\frac{1}{2}\mathcal{I} + \mathcal{K})\varphi - \nu_B \mathcal{V}\lambda$ and $\lambda = -\frac{1}{\nu_B}\mathcal{D}\varphi + (\frac{1}{2}\mathcal{I} - \mathcal{K}^*)\lambda$ on the interface $\Gamma := \partial\Omega_F = \partial\Omega_B$, where φ denotes the trace of the scalar potential on Γ, $\lambda = \frac{1}{\nu_B}\frac{\partial\varphi}{\partial n} = B \cdot n$ on Γ, $n :=$ outer unit normal to Ω_F, $\mathcal{V} :=$ single layer potential operator on Γ, $\mathcal{K} :=$ double layer potential operator on Γ, $\mathcal{D} :=$ hypersingular operator on Γ.

Using now Coulomb's gauging condition $\mathrm{div}(u) = 0$ in Ω_F and Cauchy's representation formula of the Cauchy data together with the interface condition predicting the continuity of the tangential part $H \times n$ of the H–field and of the normal component $B \cdot n$ of the B–field, we arrive at the mixed coupled fe-be variational formulation: Find $(u, \varphi, p) \in V := X \times \Phi \times M$ such that:

$$\begin{aligned} a(u,\varphi;v,\psi) + b(v,p) &= \langle f, v\rangle \quad \forall\,(v,\psi) \in X \times \Phi, && (14)\\ b(u,q) - c(p,q) &= 0 \qquad\quad \forall\, q \in M, && (15) \end{aligned}$$

where $X := H(\mathrm{curl}, \Omega_F)$, $\Phi := H_\star^{1/2}(\Gamma)$, and $M := H_\star^1(\Omega_F)$. The bilinear forms are defined by the identities

$$a(u,\varphi;v,\psi) := \int\limits_{\Omega_F} \nu\, \mathrm{curl}(u) \cdot \mathrm{curl}(v)\, dx + \langle \nu_B \mathcal{V}(\mathrm{curl}(u)\cdot n), \mathrm{curl}(v)\cdot n\rangle_\Gamma$$

$$-\langle(\frac{1}{2}\mathcal{I} + \mathcal{K})\varphi, \mathrm{curl}(v)\cdot n\rangle_\Gamma + \langle \frac{1}{\nu_B}\mathcal{D}\varphi, \psi\rangle_\Gamma + \langle(\frac{1}{2}\mathcal{I} + \mathcal{K}^*)(\mathrm{curl}(u)\cdot n), \psi\rangle_\Gamma,$$

$$b(u,q) := \int\limits_{\Omega_F} u \cdot \nabla q\, dx, \quad c(p,q) := \frac{1}{\nu_{min}} \int\limits_{\Omega_F} \nabla p \cdot \nabla q\, dx, \quad \langle f, v\rangle := \int\limits_{\Omega_F} J \cdot v\, dx.$$

The subscribe "$\star$" means that the function of the corresponding space should be L_2–orthogonal to the constant functions. Again one can show existence and uniqueness of the solution. Moreover, $p = 0$ if $\int_{\Omega_F} J\, \nabla q\, dx = 0 \quad \forall\, q \in H_\star^1(\Omega_F)$ (see [Kuh98] for the proof).

Choosing the finite (boundary) element subspaces $X_h := \mathcal{N}_h^1 \subset X$, $\Phi_h := \mathcal{S}_h^1 \subset \Phi$ and $M_h := \mathcal{S}_h^1 \subset M$, we derive from (14) the symmetric coupled fe-be Galerkin scheme: Find $(u_h, \varphi_h, p_h) \in V_h := X_h \times \Phi_h \times M_h$ such that

$$\begin{aligned} a(u_h,\varphi_h;v_h,\psi_h) + b(v_h,p_h) &= \langle f, v_h\rangle \quad \forall\,(v_h,\psi_h) \in X_h \times \Phi_h, && (16)\\ b(u_h,q_h) - c(p_h,q_h) &= 0 \qquad\quad \forall\, q_h \in M_h, && (17) \end{aligned}$$

that is again equivalent to the following symmetric, but indefinite system of coupled fe-be equations

$$\begin{pmatrix} A & K^T & B^T \\ K & -D & 0 \\ B & 0 & -C \end{pmatrix} \begin{pmatrix} \underline{u}_h \\ \underline{\varphi}_h \\ \underline{p}_h \end{pmatrix} = \begin{pmatrix} \underline{f}_h \\ 0 \\ 0 \end{pmatrix}, \qquad (18)$$

where $A = A^F + A^B$ consists of the contributions from the first two terms of the bilinear form $a(\cdot,\cdot)$. Eliminating again $\underline{p}_h = C^{-1}B\underline{u}_h$ from the third equation in (18) and inserting it into the first equation, we obtain the Schur complement system

$$\begin{pmatrix} \widetilde{A} & K^T \\ K & -D \end{pmatrix} \begin{pmatrix} \underline{u}_h \\ \underline{\varphi}_h \end{pmatrix} = \begin{pmatrix} \underline{f}_h \\ 0 \end{pmatrix},$$

where $\widetilde{A} = A^F + A^B + B^T C^{-1} B$. In contrast to the finite element case, here the Schur complement system remains symmetric and indefinite. Similar to the 2D case discussed in [Lan94], we can now construct efficient solvers on the basis of the Bramble-Pasciak transformation [BP88]. In [KS00], M. Kuhn and O. Steinbach describe the ingredients of the preconditioner and present numerical results showing the high efficiency of this solver for coupled fe-be equations in 3D.

Numerical Results

In this section, we present an example from magnetostatics and we apply the algorithms presented above. The geometry of our model problem together with the corresponding coarse surface mesh is shown in Fig. 1 on the left. We consider the model of a transformer with a kernel, three coils and the air domain around these parts. The outer boundary is given by an iron casing of high conductivity that motivates the boundary condition (2). The magnetic field which is to be computed is generated by tangential currents within the three coils. The iron core has a permeability of 1000. The resulting magnetic flux density B is shown in Fig. 1 on the right.

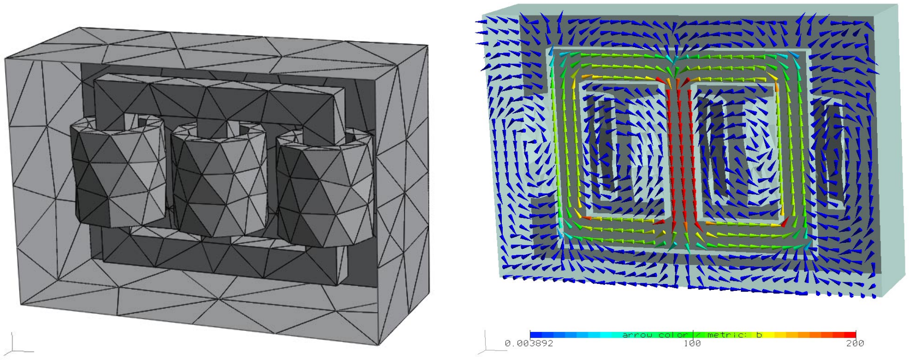


Figure 1: Geometry, initial mesh (left), resulting magnetic flux density B (right).

The basis for our domain decomposition is a tetrahedral mesh generated fully automatically by NETGEN (see [KLS00]). The surface mesh shown in Fig. 1 corresponds to a volume mesh of 2465 tetrahedra. We apply a modified recursive spectral bisection (rsb) algorithm which allows to use any number P of subdomains. The use of the rsb ensures that the elements are distributed almost equally to the P processors being used. Finer meshes corresponding to the levels $\ell = 2, 3, 4$ are obtained from uniform refinement resulting in overall 1262080 tetrahedra. This refinement is purely local and can be realized without communication. However, newly created unknowns at interfaces between different subdomains have to be identified uniquely by all surrounding processors. For this purpose, one root process per interface receives data from all surrounding subdomains, it identifies all unknowns uniquely and it finally distributes this information again to all adjacent processors. This setup phase is required once after each refinement step.

The numerical experiments presented below are carried out on a SGI Origin 2000 machine with 64 CPU R12000, 300 MHz and overall 20 GB main memory. The numerical simulations are carried out using the object oriented C++ code *FEPP* [KLS00]. The message passing is based on MPI from the *SGI Message Passing Toolkit 1.2.* The wall-clock time has been measured by *MPI_WTIME().*

The system (9) has been solved using the PMG algorithm with the relative accuracy 10^{-4}. For the multigrid preconditioner C_H a V–cycle with 1 pre– and 1 post–smoothing step HYBRIDSMOOTH has been used. The multigrid regularisator $\widetilde{C}$ has been realized by a V–cycle with 2 pre– and 2 post–smoothing steps using a standard smoother with good parallel efficiency as described before. Table 1 shows the corresponding results including number of unknowns (dof), number of iterations (It.) and wall-clock time in seconds (T[sec]). We increase the number of unknowns from top to bottom, while the number of processors is increased from left to right. First, we observe that the number of iterations is almost independent of the number of unknowns. This shows the optimality of our algorithm. However, the number of iterations depends slightly on the number of processors. That is because our smoothers depend on the partition of the mesh. In particular, the blocks of unknowns at interfaces where Jacobi steps are performed grow with the number of subdomains. The wall-clock times for the pcg are given in the upper part for each level ℓ separately. Additionally we present the overall time for the finest grid ($\ell = 4$) and together for all grids ($\Sigma\ell$) in the lower part of Table 1. This time includes the grid refinement together with the setup phase for the vector accumulation, the assembling of the matrices and the solution of the system. Table 2 shows the corresponding speedup results. First the overall speedup for the accumulated time over all 4 levels is given. It performs well until $P = 16$ and is no longer optimal for $P = 60$. This loss of efficiency is due to the setup phase for interface unknowns which shows rather bad scalability in the current implementation. So it scales from 32 sec for $P = 16$ to 15 sec for $P = 60$ only. The

		1		4	16	32	48	60	
ℓ	dof	It.	T	T	T	T	T	It.	T
1	3466	5	0.1	0.1	0.1	0.1	0.1	5	0.1
2	26907	8	7.1	2.9	1.5	1.4	1.4	11	1.7
3	212597	9	84	28	7.3	4.7	3.8	13	4.1
4	1691370	11	1398	496	81	42	28	14	24
T($\ell = 4$)		2593		806	160	86	61	54	
T($\Sigma\ell$)		2852		886.4	188.4	108.5	83.8	79.6	

Table 1: Wall-clock time T for the solver on each level (upper part), overall wall-clock time for $\ell = 4$ and accumulated for all levels ($\Sigma\ell$) in seconds.

P	1	4	16	32	48	60
$\Sigma\ell$	1.0	3.2	15.1	26.2	34.1	35.6
$\ell = 4$	1.0	3.2	16.2	30.0	42.5	48.0
pcg ($\ell = 4$)	1.0	2.8	17.2	33.0	49.6	58.5
1 Iter. ($\ell = 4$)	1.0	3.3	21.9	42.1	63.2	74.4

Table 2: Speedup results for overall time $\Sigma\ell$, time for $\ell = 4$, pcg (solver) for $\ell = 4$ and one iteration of pcg for $\ell = 4$.

speedup computed for $\ell = 4$ shows a slightly better behavior since coarse grid effects are neglected. However, the speedup computed for the solver and $\ell = 4$ only, shows much better results. Here the speedup is almost optimal for $P = 60$. For a more detailed analysis we consider the speedup with respect to one iteration for $\ell = 4$. Now we observe even super-speedups. However this is due to cache effects.

References

[AFW00]Douglas N. Arnold, Richard S. Falk, and Ragnar Winther. Multigrid in H(div) and H(curl). *Numer. Math.*, 85(2):197–217, 2000.

[BP88]James H. Bramble and Joseph E. Pasciak. A preconditioning technique for indefinite systems resulting from mixed approximations of elliptic problems. *Mathematics of Computation*, 50(181):1–17, 1988.

[Cia78]Philippe G. Ciarlet. *The Finite Element Method for Elliptic Problems.* North-Holland, Amsterdam, 1978.

[Haa98]Gundolf Haase. Parallel incomplete Cholesky preconditioners based on the non-overlapping data distribution. *Parallel Computing*, 24(11):1685–1703, 1998.

[Haa99]Gundolf Haase. *Parallelisierung numerischer Algorithmen für partielle Differentialgleichungen.* Teubner-Verlag, Stuttgart/Leipzig, 1999.

[Hac85]Wolfgang Hackbusch. *Multigrid Methods and Applications.* Springer, Berlin, 1985.

[Hip99]Ralf Hiptmair. Multigrid methods for Maxwell's equations. *SIAM J. Numer. Anal.*, 36:204–225, 1999.

[HLM91]Gundolf Haase, Ulrich Langer, and Arnd Meyer. The approximate Dirichlet decomposition method. part I,II. *Computing*, 47:137–167, 1991.

[JLM$^+$89]Michael Jung, Ulrich Langer, Arnd Meyer, Werner Queck, and Manfred Schneider. Multigrid preconditioners and their applications. In G. Telschow, editor, *Third Multigrid Seminar, Biesenthal 1988*, pages 11–52, Berlin, 1989. Karl–Weierstrass–Institut. Report R–MATH–03/89.

[KLS00]Michael Kuhn, Ulrich Langer, and Joachim Schöberl. Scientific computing tools for 3D magnetic field problems. In John R. Whiteman, editor, *The Mathematics of Finite Elements and Applications X*, pages 239 – 258. Elsevier, 2000.

[KS00]Michael Kuhn and Olaf Steinbach. FEM–BEM coupling for 3D exterior magnetic field problems. In T. Tiihonen and P. Neittaanmäki, editor, *ENUMATH 99 - Proceedings of the 3rd European Conference on Numerical Mathematics and Advanced Applications, Jyväskylä, Finland, July 26-30, 1999*, pages 180–187, Singapore, 2000. World Scientific.

[Kuh98]Michael Kuhn. *Efficient Parallel Numerical Simulation of Magnetic Field Problems.* PhD thesis, Johannes Kepler University Linz, Institute of Analysis and Computational Mathematics, 1998.

[Lan94]Ulrich Langer. Parallel iterative solution of symmetric coupled BE/FE–equations via domain decomposition. *Contemp. Math.*, 157:335–344, 1994.

[N8́6]J.-C. Nédélec. A new family of mixed finite elements in R^3. *Numer. Math.*, 50:57–81, 1986.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau,

38. A Fictitious Domain Decomposition Method for High Frequency Acoustic Scattering Problems

U. Hetmaniuk[1], C. Farhat[2]

Introduction

It is well known that most PDE problems defined over an axisymmetric domain can be efficiently solved by a Fourier based solution method. However, for many applications, the underlying computational domain is not entirely axisymmetric, but has one or several major axisymmetric subdomains. For such problems, an axisymmetric analysis method is not applicable, and a straightforward one can be inefficient because it does not exploit the geometrical properties of the axisymmetric components. The objective of this paper is to fill this existing gap, and propose a computationally efficient method for solving problems on a class of partially axisymmetric domains [FUR99].

We illustrate our method for a submarine problem. Indeed, a submarine can be represented as the assembly of a major cylindrical component, and a few minor "features" that are however essential for the application itself. Our approach is presented here in the context of the finite element solution of the three-dimensional exterior Helmholtz problem in the high frequency regime. This problem is challenging because it leads to large-scale computations. For example, at a wave length equal to the length of a submarine divided by 360, the finite element discretization of such a problem requires hundreds of millions of grid points.

The proposed methodology is based on a fictitious domain approach (for example, see [DGH$^+$92]) where the original exterior Helmholtz problem is extended into an axisymmetric exterior problem, and where parts of the genuine boundary conditions are enforced through the utilization of Lagrange multipliers. The axisymmetry of the enlarged domain is then exploited by expanding the solution into a Fourier series. The Fourier modes of the solution are computed by solving a series of bidimensional problems coupled altogether by the Lagrange multipliers. The associated constrained problem is treated by extension of the FETI-H method [FML00], and a special coarse problem is constructed for accelerating the convergence of the corresponding interface problem. The resulting fictitious domain decomposition method is a fast solver because it transforms a 3D problem into a series of 2D ones.

For simplicity but without any loss of generality, we consider in this paper the case of a scatterer with a single component and one arbitrarily shaped feature. The generalization to an arbitrary number of axisymmetric components and features is straightforward.

[1] University of Colorado at Boulder, hetmaniu@hawkeye.colorado.edu
[2] University of Colorado at Boulder, charbel@boulder.colorado.edu

Mathematical formulation

Extension of the solution to a fictitious domain

We consider an impenetrable obstacle Ω composed of two substructures

$$\bar{\Omega} = \bar{C} \cup \bar{W}$$

where C and W are two disjoint open sets and C is axisymmetric, as illustrated on Fig. 1.

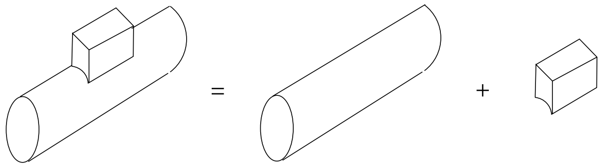


Figure 1: Physical decomposition of the scatterer

B_R is the ball of radius R centered at the center of geometry of Ω, n is the outward normal to ∂B_R and $\frac{\partial}{\partial n}$ is the normal derivative operator. We define the following exterior domains and their intersection with B_R.

$$\left\{ \begin{array}{l} \Omega_e = \mathbb{R}^3 \backslash \bar{\Omega} \\ \Omega_{e,R} = \Omega_e \cap B_R \end{array} \right. \qquad \left\{ \begin{array}{l} C_e = \mathbb{R}^3 \backslash \bar{C} \\ C_{e,R} = C_e \cap B_R \end{array} \right.$$

The surface Γ is defined as the intersection of ∂W with $C_{e,R}$.

The focus model problem is given by

Find $u \in H^1(\Omega_{e,R})$ such that

$$\left\{ \begin{array}{rcllc} \Delta u + k^2 u & = & f & \text{in} & \Omega_{e,R} \\ u & = & 0 & \text{on} & \partial\Omega \\ \frac{\partial u}{\partial n} & = & iku & \text{on} & \partial B_R \end{array} \right. \tag{1}$$

where u is the acoustic scattered field and f belongs to $L^2(\Omega_{e,R})$.

In this paper we consider only a spherical artificial boundary with a first-order approximation of the Sommerfeld condition. But any other axisymmetric boundary or absorbing condition could be used to ensure that the waves are outgoing.

In order to obtain an axisymmetric computational domain, we embed the original domain $\Omega_{e,R}$ into $C_{e,R}$ which satisfies

$$\bar{C}_{e,R} = \bar{\Omega}_{e,R} \cup \bar{W}$$

We extend u from $\Omega_{e,R}$ to the enlarged domain $C_{e,R}$ to a function (still denoted by u for simplicity) with $H^1(C_{e,R})$ regularity. This regularity requirement implies the continuity of the trace of u across the surface Γ.

Solving problem (1) is *equivalent* to solving the following problem

Find $u \in V = \{v \in H^1(C_{e,R}) \quad | \quad v = 0 \text{ on } \Gamma\}$ such that

$$\left\{\begin{array}{rcll} \Delta u + k^2 u & = & \tilde{f} & \text{in} \quad C_{e,R} \\ u & = & 0 & \text{on} \quad \partial C \\ \frac{\partial u}{\partial n} & = & iku & \text{on} \quad \partial B_R \end{array}\right. \tag{2}$$

in the sense that the solution of problem (2) restricted to $\Omega_{e,R}$ satisfies the boundary value problem (1), and $\tilde{f}$ is an L^2-extension of f, for example, by 0.

We include the boundary condition on ∂C into the definition of the functional space

$$Y = \{v \in H^1(C_{e,R}) \quad | \quad v = 0 \text{ on } \partial C\}$$

We can rewrite problem (2) into the following saddle-point problem :

$$\text{Find } (u, \mu) \in Y \times H^{-1/2}(\Gamma) \text{ such that}$$

$$\left\{\begin{array}{l} \int_{C_{e,r}} \nabla u.\nabla v - k^2 uvdx + \int_{\partial B_R} ikuvd\sigma = \int_{C_{e,R}} fvdx + \int_\Gamma \mu v d\sigma, \quad \forall v \in W \\ \\ \int_\Gamma \zeta.ud\sigma = 0, \quad \forall \zeta \in H^{-1/2}(\Gamma) \end{array}\right. \tag{3}$$

Domain decomposition

For high-frequency acoustic scattering problems, numerical discretization leads to large-scale systems of equations. Thus a domain decomposition technique is useful for solving these systems. For the sake of clarity, but without any loss of generality, we present our domain decomposition method for the case of two subdomains.

We describe the axisymmetric domain $C_{e,R}$ in cylindrical coordinates (r, θ, z). $C_{e,R}$ is generated by rotation around the z-axis of a meridian plane $c_{e,R}$. We partition $c_{e,R}$ into two non-overlapping subdomains c^1 and c^2. The decomposition of $c_{e,R}$ induces a partition of $C_{e,R}$

$$\bar{C}_{e,R} = \bar{C}^1_{e,R} \cup \bar{C}^2_{e,R}$$

where $C^1_{e,R}$ (resp. $C^2_{e,R}$) is generated by the rotation of c^1 (resp. c^2) around the z-axis.

Let u^s denote the restriction to $C^s_{e,R}$ of the solution of problem (2), for $s = 1, 2$. The interface between $C^1_{e,R}$ and $C^2_{e,R}$ is denoted Σ_I, which is axisymmetric. Now, we are looking for the functions u^s in the following functional spaces

$$V_s = \{v \in H^1(C^s_{e,R}) \quad | \quad v = 0 \text{ on } \Gamma \cap C^s_{e,R}\}$$

for $s = 1, 2$.

For solving problem (2) on a partitioned domain, we adopt the FETI-H method [FML00] which introduces the two following problems

$$\text{Find } (u^1, u^2) \in V_1 \times V_2 \text{ such that}$$

$$\left\{\begin{array}{rcll} \Delta u^1 + k^2 u^1 &=& \tilde{f}_{|C^1_{e,R}} & \text{in} \quad C^1_{e,R} \\ u^1 &=& 0 & \text{on} \quad \partial C \cap \partial C^1_{e,R} \\ \frac{\partial u^1}{\partial n} &=& iku^1 & \text{on} \quad \partial B_R \cap \partial C^1_{e,R} \\ \frac{\partial u^1}{\partial \nu^1} + iku^1 &=& \lambda & \text{on} \quad \Sigma_I \end{array}\right.$$
$$\left\{\begin{array}{rcll} \Delta u^2 + k^2 u^2 &=& \tilde{f}_{|C^2_{e,R}} & \text{in} \quad C^2_{e,R} \\ u^2 &=& 0 & \text{on} \quad \partial C \cap \partial C^2_{e,R} \\ \frac{\partial u^2}{\partial n} &=& iku^2 & \text{on} \quad \partial B_R \cap \partial C^2_{e,R} \\ -\frac{\partial u^2}{\partial \nu^2} + iku^2 &=& \lambda & \text{on} \quad \Sigma_I \end{array}\right. \tag{4}$$

with the constraint

$$u^1 - u^2 = 0 \text{ on } \Sigma_I \tag{5}$$

Here, ν^s denotes here the unit outward normal on the interface boundary between $C^1_{e,R}$ and $C^2_{e,R}$, and λ is a Lagrange multiplier field for enforcing the continuity at the interface of the solution.

Similarly to the previous section, we can introduce a saddle-point problem with the boundary condition on ∂C inside a functional space and two Lagrange multipliers: λ for the continuity at the interface of the solution, μ for enforcing the genuine boundary condition on Γ.

A Fourier based finite element discretization

Each function u^s is 2π-periodical with respect to the cylindrical coordinate θ. Hence, it can be expanded in a Fourier series with respect to θ as follows

$$u^s(r,\theta,z) = \sum_{n=-\infty}^{\infty} u^s_n(r,z)e^{in\theta} \tag{6}$$

The Fourier coefficients of u^s are now functions of (r,z) defined on c^s.

Discretizing the two-dimensional subdomains c^s by finite elements and truncating the Fourier expansions leads to the following discrete expression of u^1 and u^2

$$\left\{\begin{array}{rcl} u^1(r,\theta,z) &=& \sum_{n=-n_\theta}^{n_\theta} \sum_{j=1}^{n^1_{cyl}} u^1_{n,j} X^1_j(r,z) e^{in\theta} \\ u^2(r,\theta,z) &=& \sum_{n=-n_\theta}^{n_\theta} \sum_{j=1}^{n^2_{cyl}} u^2_{n,j} X^2_j(r,z) e^{in\theta} \end{array}\right. \tag{7}$$

where n_θ denotes the selected number of Fourier modes, $X^s_j(r,z)$ denote the shape functions associated with the chosen two-dimensional finite element discretization in c^s and $u^s_{n,j}$ denote the corresponding nodal values.

We enforce all the constraints pointwise with discrete Lagrange multipliers, assuming the subdomains have matching discrete interfaces.

This discretization leads to the following algebraic system

$$\left\{\begin{array}{rcl} (\boldsymbol{K^1_{n_\theta}} - \boldsymbol{k^2 M^1_{n_\theta}} - \boldsymbol{ikM^1_{S,n_\theta}} + \boldsymbol{ikB^{1^T}_{n_\theta} M_{bb} B^1_{n_\theta}})\boldsymbol{u^1_{n_\theta}} + \boldsymbol{B^{1^T}_{n_\theta}\lambda} + \boldsymbol{C^{1^T}_{n_\theta}\mu} & = & \boldsymbol{F^1_{n_\theta}} \\ (\boldsymbol{K^2_{n_\theta}} - \boldsymbol{k^2 M^2_{n_\theta}} - \boldsymbol{ikM^2_{S,n_\theta}} - \boldsymbol{ikB^{2^T}_{n_\theta} M_{bb} B^2_{n_\theta}})\boldsymbol{u^2_{n_\theta}} + \boldsymbol{B^{2^T}_{n_\theta}\lambda} + \boldsymbol{C^{2^T}_{n_\theta}\mu} & = & \boldsymbol{F^2_{n_\theta}} \\ \boldsymbol{B^1_{n_\theta} u^1_{n_\theta}} + \boldsymbol{B^2_{n_\theta} u^2_{n_\theta}} & = & \boldsymbol{0} \\ \boldsymbol{C^1_{n_\theta} u^1_{n_\theta}} + \boldsymbol{C^2_{n_\theta} u^2_{n_\theta}} & = & \boldsymbol{0} \end{array}\right. \tag{8}$$

$\boldsymbol{K^s_{n_\theta}}$, $\boldsymbol{M^s_{n_\theta}}$ are the so-called stiffness and mass matrices for the substructure $C^s_{e,R}$. Matrix $\boldsymbol{M^s_{S,n_\theta}}$ is induced by the Sommerfeld radiation condition and is non-zero only at the degrees of freedom lying on the outer boundary of the domain. Matrix $\boldsymbol{M_{bb}}$ is an interface mass matrix introduced in the FETI-H method for local damping, in order to avoid local resonance. The vectors $\boldsymbol{u^s_{n_\theta}}$ and $\boldsymbol{F^s_{n_\theta}}$ are respectively the vectors of Fourier coefficients of the solution and the load on substructure $C^s_{e,R}$, and $\boldsymbol{\lambda}$ is the vector of Lagrange multipliers for enforcing the continuity at the interface of the Fourier coefficients. The matrices $\boldsymbol{B^s_{n_\theta}}$ depend on the shape functions $X^s_j(r,z)$ and on the discretization of the Lagrange multiplier field λ. With our assumptions, each $\boldsymbol{B^s_{n_\theta}}$ becomes a Boolean substructure connectivity matrix. $\boldsymbol{\mu}$ is the vector of Lagrange multipliers for enforcing pointwise the constraints on Γ. The matrices $\boldsymbol{C^s_{n_\theta}}$ depend on the discretization of u^s and of the Lagrange multiplier field μ. For each node k lying on $\Gamma \cap c^s$ and for which the second cylindrical coordinate in $C^s_{e,R}$ is denoted by θ^k, a constraint equation can be written as follows

$$\sum_{n=-n_\theta}^{n=n_\theta} u^s_{n,k} e^{in\theta^k} = 0 \tag{9}$$

The system of equations (8) has the pattern of the FETI-H equations with a set of multipoint constraints (MPCs). Therefore, it is most efficiently solved by the numerically scalable FETI-H solver [FML00] coupled with an appropriate treatment of the MPCs [FLR98].

By gathering the Lagrange multipliers $\boldsymbol{\lambda}$, $\boldsymbol{\mu}$ together and also the matrices $\boldsymbol{B^s_{n_\theta}}$, $\boldsymbol{C^s_{n_\theta}}$ together, we can define an extended dual interface problem. We solve this dual problem with the FETI-H solver, where at each iteration the MPCs are exactly satisfied and where the Krylov space for the search directions is enriched by the range of a coarse matrix Q [FML00] based now on the Fourier coefficients of planar waves.

The generalization to an arbitrary number of subdomains is straightforward. One needs only to follow the methodology defined in [FML00] for signing efficiently all the interfaces of the subdomains.

Numerical experiments

We illustrate our embedding method with the resolution of the Helmholtz equation on the exterior domain of an obstacle. The structure is composed of a large cylindrical component and a conical tower of 45 degrees.

The problem is formulated as follows

$$\left\{ \begin{array}{rcll} \Delta u + k^2 u & = & 0 & \text{in} \quad \Omega_{e,R} \\ u & = & 1 & \text{on} \quad \partial\Omega \\ \frac{\partial u}{\partial n} & = & 0 & \text{on} \quad S_R \end{array} \right. \qquad \left\{ \begin{array}{l} \text{Diameter of the cylinder : } a = 1 \\ \text{Length of the cylinder : } L = 10 \\ \text{Wavenumber : } kL = 10 \\ \text{Wavelength : } \lambda = 2\pi \\ \text{Mesh size : } h = \lambda/25 \\ \text{Distance } S_R \text{ - obstacle : } 0.5\lambda \end{array} \right. \tag{10}$$

For this computation, S_R has a cylindrical shape.

We discretize the domain $\Omega_{e,R}$ by 343,680 8-noded brick elements. We compute a reference solution by performing a global finite element analysis with Q1 functions, using the classical FETI-H method.

We compute a solution obtained by our methodology with 40 Fourier modes and 172 Lagrange multipliers for enforcing part of the boundary condition. The two-dimensional mesh for computing the Fourier coefficients is made of 1,072 Q1 elements.

As shown on Fig. 2 and Fig. 3, the results obtained by the fictitious method are in excellent agreement with those obtained by a global analysis method.

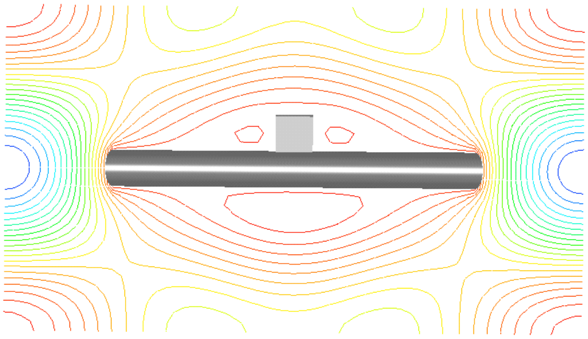


Figure 2: Isovalues of the reference solution

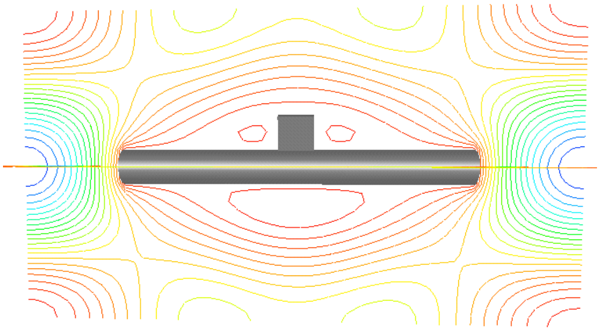


Figure 3: Isovalues of the solution with 40 modes

In all the cases, we use the following convergence criterion

$$\| \tilde{\boldsymbol{K}}\boldsymbol{u} - \boldsymbol{f} \| \leq \mathbf{10^{-6}} \| \boldsymbol{f} \|$$

where $\tilde{\boldsymbol{K}}$ denotes the generalized stiffness matrix of the system to be solved, $\boldsymbol{u}$ denotes either the nodal values of the 3D solution or the nodal values of the Fourier coefficients of the solution and $\boldsymbol{f}$ the corresponding right-hand side.

The performance results of the FETI-H method applied to the solution of the 3D computation are reported on the table 1. These results are achieved for 200 subdomains on a single processor Origin 2000 computer. The size of the coarse grid problem, on which the GCR solver iterates, is 1,577.

Number of iterations	Total CPU time	Total memory cost
130	2,548 s	2,172 Mb

Table 1: Performance results for the 3D computation on a single processor Origin 2000

The performance results of the method with fictitious domain are reported on the table 2, using 1, 3 and 5 subdomains for the axisymmetric component on a single processor. Note that for the case of one subdomain, the constrained problem is solved by a direct method.

Nb. of subd.	Size of pb. coarse	Number of iterations	Total CPU time	Total memory cost
1	172	DIRECT	145 s	449 Mb
3	216	10	232 s	402 Mb
5	260	12	253 s	426 Mb

Table 2: Performance results for the fictitious methodology

As expected, the fictitious domain method is an order magnitude faster and less memory intensive than the 3D domain decomposition based FETI-H method, because this fictitious domain transforms a 3D problem into a series of 2D ones. We also note that for the considered wavenumber k, the size of the 2D mesh is such that solving the 2D Fourier problems by a direct method is faster than solving them by a domain decomposition one. However, one can expect this trend to reverse for larger values of ka.

Conclusion

In this paper, we have presented a fictitious domain decomposition method that allows exploiting a potential partial axisymmetry of a given computational domain. This in turn results in a dramatic reduction of the size of the system of equations to be solved, without a loss of accuracy. Therefore, this fictitious domain decomposition method enables the solution of high frequency 3D acoustic scattering problems on contemporary computational platforms.

Acknowledgment

The authors acknowledge the support by the Office of Naval Research under Grant N-00014-95-1-0663. The first author also acknowledges the support by the Corps des Ponts et Chaussées, France.

References

[DGH$^+$92]Q. V. Dihn, R. Glowinski, J. He, V. Kwock, T. W. Pan, and J. Périaux. Lagrange multiplier approach to fictitious domain methods: Application to fluid dynamics and electromagnetics. In David E. Keyes, Tony F. Chan, Gérard A. Meurant, Jeffrey S. Scroggs, and Robert G. Voigt, editors, *Fifth International Symposium on Domain Decomposition Methods for Partial Differential Equations*, pages 151–194, Philadelphia, PA, 1992. SIAM.

[FLR98]C. Farhat, C. Lacour, and D. Rixen. Incorporation of linear multipoint constraints in substructure based iterative solvers. part 1 : A numerically scalable algorithm. *Int. J. Numer. Meth. Eng.*, 43:997–1016, 1998.

[FML00]C. Farhat, A. Macedo, and M. Lesoinne. A two-level domain decomposition method for the iterative solution of high-frequency exterior Helmholtz problems. *Numer. Math.*, 85(2):283–303, 2000.

[FUR99]C. Farhat, U.Hetmaniuk, and D. Rixen. An efficient substructuring method for analyzing structures with major axisymmetric components. In *AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference and Exhibit, 40th, St. Louis, MO, Apr 12-15 1999*, pages 832–837, 1999. AIAA Paper 99-1283.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau,

39. Numerical computation for some competition-diffusion systems on a parallel computer

R. Ikota[1], M. Mimura[2], T. Nakaki[3]

Introduction

In theoretical biology, spatial segregation of biological species has been investigated by many scientists (see [DHMP99], [IMY98] and the references therein). Among several models explaining such a phenomenon, we deal with the systems of competition-diffusion type.

We consider n kinds of species U_i $(1 \leq i \leq n)$. Let $u_i(x,t)$ be the population density of the species U_i $(1 \leq i \leq n)$ at time $t > 0$ and the position $x \in \Omega$, where Ω is a bounded domain in $\mathbf{R}^N$. Then our model can be described by

$$\frac{\partial u_i}{\partial t} = d_i \Delta u_i + (r_i - \sum_{j=1}^{n} a_{ij} u_j) u_i \qquad (i = 1, 2, \ldots, n), \quad x \in \Omega, \ \ t > 0, \tag{1}$$

where d_i is the diffusion rate, r_i the intrinsic growth rate, a_{ii} the intraspecific competition rate, and a_{ij} $(i \neq j)$ the interspecific competition rate between U_i and U_j. We assume that all these parameters are nonnegative and impose initial and Neumann boundary conditions on (1):

$$u_i(x,0) = u_{i0}(x) \qquad (i = 1, 2, \ldots, n), \quad x \in \Omega, \tag{2}$$

$$\frac{\partial u_i}{\partial \nu} = 0 \qquad (i = 1, 2, \ldots, n), \quad x \in \partial\Omega, \quad t > 0, \tag{3}$$

where ν is the unit outer normal to $\partial\Omega$, and u_{i0} is a nonnegative function.

Specifically we are interested in the case where the competition is extremely strict. In order to treat such situations we rewrite the equations (1) and obtain the following:

$$\frac{\partial u_i}{\partial t} = d_i \Delta u_i + (r_i - a_{ii} u_i) u_i - k \sum_{\substack{j=1 \\ j \neq i}}^{n} b_{ij} u_i u_j \qquad (i = 1, 2, \ldots, n) \quad x \in \Omega, \ \ t > 0. \tag{4}$$

The parameter k represents the magnitude of interspecific competition. We study (4) when k is very large. As $k \to \infty$, we can observe in our numerical computations that the region Ω is divided into each region Ω_i which only a single species U_i occupies. One of our interests is to analyze the behavior of interfaces between $\{\Omega_i\}$. If we use typical numerical methods, we have some difficulties to track the interfaces. That is

[1] University of Tokyo, ikota@ms.u-tokyo.ac.jp

[2] Hiroshima University, mimura@math.sci.hiroshima-u.ac.jp

[3] Kyushu University, nakaki@math.kyushu-u.ac.jp

because they appear in the limit case where $k \to \infty$. In fact, for a fixed large value of k, we can easily perform numerical computations to (4), however, we have no criterion to determine the numerical interfaces by using numerical solutions to u_i.

In this paper we propose a method to analyze (4) when $k \to \infty$, by which we can track the interfaces. Our method is described as follows:

A-Method

Step 1: For $u_i(\cdot, t)$, solve the following PDE:

$$\begin{aligned} \frac{\partial \bar{u}_i}{\partial \tau} &= d_i \Delta \bar{u}_i + (r_i - a_{ii}\bar{u}_i)\bar{u}_i && \text{in } \Omega, \quad 0 < \tau < \Delta t, \\ \frac{\partial \bar{u}_i}{\partial \nu} &= 0 && \text{on } \partial\Omega, \quad 0 < \tau < \Delta t, \\ \bar{u}_i(x,0) &= u_i(x,t) && \text{in } \Omega, \end{aligned}$$

where $\Delta t > 0$ is a given constant $(1 \le i \le n)$.

Step 2: Solve the following ODE until $\tau = \infty$, that is, compute the equilibrium points for $1 \le i \le n$:

$$\begin{aligned} \frac{d\check{u}_i(x,\tau)}{d\tau} &= -\sum_{\substack{j=1 \\ j \ne i}}^{n} b_{ij}\check{u}_i\check{u}_j && \text{in } \Omega, \quad 0 < \tau < \infty, \\ \check{u}_i(x,0) &= \bar{u}_i(x,\Delta t) && \text{in } \Omega. \end{aligned}$$

Step 3: Put $u_i(x, t + \Delta t) = \check{u}_i(x, \infty)$ $(1 \le i \le n)$.

This method has the advantage that we can determine the interfaces naturally as shown in Fig. 5 without complicated procedure even in the multi-component $(n \ge 2)$ and multi-dimensional $(N \ge 2)$ cases.

The aim of this paper is as follows: We show the mathematical justification of A-Method when $n = 2$ and $d_1 = d_2$ in the second section. The condition $d_1 = d_2$ is imposed by the mathematical reason. We also propose a parallel algorithm to A-Method in the third section. We describe the algorithm and perform numerical simulations for the typical case $n = 3$. When $n \ne 3$, we can similarly treat the problem.

Mathematical justification for the two-component case

Known Results

We consider the following two-component (that is, $n = 2$) system:

$$u_t = d_1 \Delta u + f(u)u - kuv \quad \text{in } Q = \Omega \times \mathbf{R}^+, \tag{5}$$

$$v_t = d_2 \Delta v + g(v)v - \alpha kuv \quad \text{in } Q = \Omega \times \mathbf{R}^+, \tag{6}$$

$$\frac{\partial u}{\partial \nu} = 0, \quad \frac{\partial v}{\partial \nu} = 0 \quad \text{on } S = \partial\Omega \times \mathbf{R}^+, \tag{7}$$

$$u(x,0) = u_0^k(x), \quad v(x,0) = v_0^k(x) \quad \text{for } x \in \Omega, \tag{8}$$

where $f(u) = r_1 - a_{11}u$ and $g(v) = r_2 - a_{22}v$.

Let $(u^{(k)}, v^{(k)})$ be a solution to (5)–(8) and put $w^{(k)} = u^{(k)} - v^{(k)}/\alpha$. If $u_0^{(k)}$ and $v_0^{(k)}$ converge to u_0 and v_0 respectively, then by Proposition 2.1 in [DHMP99], $w^{(k)}$ converges to a weak solution w of the following problem as $k \to \infty$:

$$w_t = \nabla(d(w)\nabla w) + h(w) \quad \text{in} \quad Q, \tag{9}$$

$$\frac{\partial w}{\partial \nu} = 0 \quad \text{on} \quad S, \tag{10}$$

$$w(x,0) = w_0(x) \equiv u_0(x) - \frac{v_0(x)}{\alpha} \quad \text{for} \quad x \in \Omega, \tag{11}$$

where

$$d(s) = \begin{cases} d_1 & \text{if} \quad s > 0, \\ d_2 & \text{if} \quad s < 0, \end{cases}$$

$$h(s) = \begin{cases} f(s)s & \text{if} \quad s > 0, \\ g(-\alpha s)s & \text{if} \quad s < 0. \end{cases}$$

Under certain conditions, by putting $u = [w]^+$ and $v = \alpha[w]^-$, we observe that the above problem (9)–(11) is equivalent to the following problem (see [DHMP99]):

$$u_t = d_1\Delta u + f(u)u \quad \text{in} \quad Q^{\text{int}}, \tag{12}$$

$$v_t = d_2\Delta v + g(v)v \quad \text{in} \quad Q^{\text{ext}}, \tag{13}$$

$$u = 0 \quad \text{and} \quad v = 0 \quad \text{on} \quad \Gamma, \tag{14}$$

$$d_1\frac{\partial u}{\partial n} = -\frac{d_2}{\alpha}\frac{\partial v}{\partial n} \quad \text{on} \quad \Gamma, \tag{15}$$

$$\frac{\partial v}{\partial n} = 0 \quad \text{on} \quad \partial\Omega \times (0,T], \tag{16}$$

$$u(x,0) = u_0(x), \;\; v(x,0) = v_0(x) \quad \text{for} \quad x \in \Omega, \tag{17}$$

where

$$Q^{\text{int}} = \{(x,t) \in \mathbf{R} \times (0,T];\, u(x,t) > 0 \text{ and } v(x,t) = 0\},$$

$$Q^{\text{ext}} = \{(x,t) \in \mathbf{R} \times (0,T];\, u(x,t) = 0 \text{ and } v(x,t) > 0\}.$$

Definition of the Approximation and Results

In this subsection, we show a mathematical justification that A-Method gives an approximation to our problem. In Step 1 of A-Method we solve the following systems:

$$(P_u)\begin{cases} u_t = d_1\Delta u + f(u)u & \text{in} \quad Q, \\ \frac{\partial u}{\partial \nu} = 0 & \text{on} \quad S, \\ u(x,0) = u_0(x) \in C(\bar{\Omega}) & \text{for} \quad x \in \Omega, \end{cases}$$

$$(P_v)\begin{cases} v_t = d_2\Delta v + g(v)v & \text{in} \quad Q, \\ \frac{\partial v}{\partial \nu} = 0 & \text{on} \quad S, \\ v(x,0) = v_0(x) \in C(\bar{\Omega}) & \text{for} \quad x \in \Omega. \end{cases}$$

We denote the solutions to (P_u) and (P_v) by $\mathcal{H}^u(t)u_0$ and $\mathcal{H}^v(t)v_0$, respectively. In Step 2, we solve the following ordinary differential equations:

$$\begin{cases} \dfrac{du}{dt} = -uv, \\ \dfrac{dv}{dt} = -\alpha uv, \\ u(0) = u_0, \quad v(0) = v_0. \end{cases}$$

Recalling

$$\frac{d}{dt}(u - \frac{v}{\alpha}) = 0,$$

then we obtain

$$\lim_{t\to\infty}(u(t), v(t)) = ([u_0 - \frac{v_0}{\alpha}]^+, \alpha[u_0 - \frac{v_0}{\alpha}]^-). \tag{18}$$

Let us define an operator $\mathcal{K}(t)$ parameterized with non-negative number t by

$$\mathcal{K}(t)z_0 \equiv \mathcal{H}^u(t)[z_0]^+ - \frac{1}{\alpha}\mathcal{H}^v(t)(\alpha[z_0]^-). \tag{19}$$

Then we can describe the approximated solution constructed by A-Method as

$$\mathcal{K}(T/n)^n w_0. \tag{20}$$

If $d_1 = d_2$, under certain conditions imposed on w_0 we have proven

$$\begin{aligned} \|\mathcal{K}(T/n)^n w_0 - w(T)\|_{L^2(\Omega)} &\leq C_1(T/n)^{1/2}, \\ \|\mathcal{K}(T/n)^n w_0 - w(T)\|_{L^1(\Omega)} &\leq C_2(T/n). \end{aligned}$$

These inequalities implies that the numerical solutions of A-Method converges as $\Delta t \to 0$. Unfortunately at present we can not prove the convergence when $d_1 \neq d_2$. However our numerical computations suggest that the solution also converges.

Parallel computations for the three-component case

Algorithm

Our algorithm here is shown when $n = 3$. For $n \neq 3$, it is quite easy to extend our algorithm. We describe our algorithm for a computer with three CPUs which are called CPU1, CPU2 and CPU3. To CPUi we assign three arrays, say Array$i$-u, Array$i$-v and Array$i$-w ($i = 1, 2, 3$).

The first step (Fig. 1): First of all, we put the data u, v and w into Array1-u, Array2-v and Array3-w, respectively

The second step (Fig. 1): Then we solve

$$\begin{array}{llll} u_t = d_1\Delta u + (r_1 - a_{11}u)u & \text{on} & \text{Array1-u} & \text{using CPU1,} \\ v_t = d_2\Delta v + (r_2 - a_{22}v)v & \text{on} & \text{Array2-v} & \text{using CPU2,} \\ w_t = d_3\Delta w + (r_3 - a_{33}w)w & \text{on} & \text{Array3-w} & \text{using CPU3.} \end{array}$$

The third step (Fig. 2): We copy Array1-u into Array2-u and Array3-u, Array2-v into Array1-v and Array3-v, Array3-w into Array1-w and Array2-w.

The fourth step (Fig. 3): We compute the ODE system. We separate the region into three parts. We assign each part to CPUi ($i = 1, 2, 3$) respectively.

The fifth step (Fig. 4): Gather data u into Array1-u, v into Array2-v and w into Array2-w.

We note that the second and fourth steps stated above correspond to Steps 1 and 2 of A-Method, respectively.

Numerical experiments

Let us demonstrate our numerical simulations when the region Ω is the two dimensional interval $(0, 1)^2$. We use the workstation Sun Enterprise 450 (4 CPUs, Total memory 2GB). The programs are written in Sun Fortran 77 (Option: `-fast -O5`) and MPI [GLS94].

Numerical parameters we use are 256×256 space mesh and $\Delta t = 0.001$. Computations are halted if one of three species u, v or w becomes extinct.

We obtain the following table which shows the CPU times of the single and parallel computations. We have used 3 CPUs and obtained about 2.3 times speed-up. In our experience, the parallel performance goes up when the nodal points near the interfaces are assigned equally to each CPU.

CPU times

case	Single	Parallel	ratio
a	769sec.	328sec.	2.34
b	2544sec.	1104sec.	2.30
c	2562sec.	1112sec.	2.30
d	2951sec.	1242sec.	2.38
e	3967sec.	1742sec.	2.28

On this table, we remark the following:

- *Single* in the table means the computation using a usual code without MPI.
- *Parallel* means that the computation by our algorithm with 3 CPUs.
- We vary the initial function and parameters $\{r_i\}$ and $\{a_{ij}\}$ in *cases* (a)–(e).

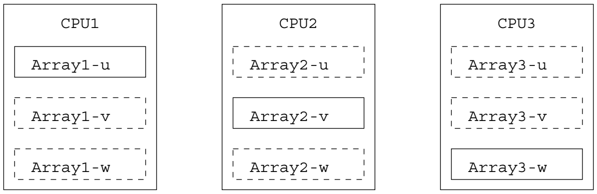


Figure 1: The first and second steps. The data u, v and w are stored in Array1-u, Array2-v and Array3-w, respectively. Then solve the PDE on each CPUs.

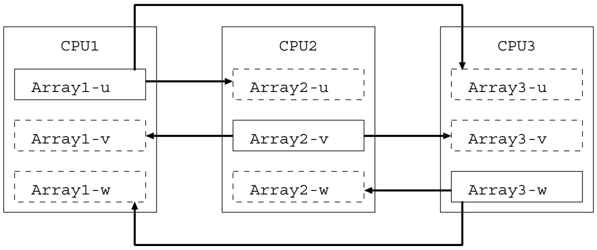


Figure 2: The third step. Message passing between CPUs.

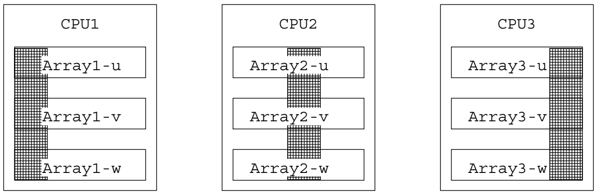


Figure 3: The fourth step. Solve the ODE on each CPUs.

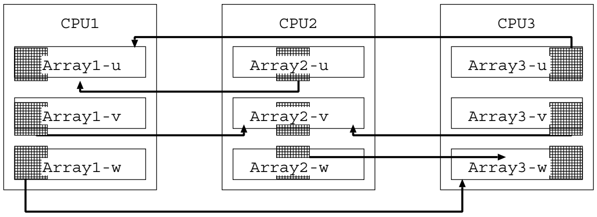


Figure 4: The fifth step. Gather the data.

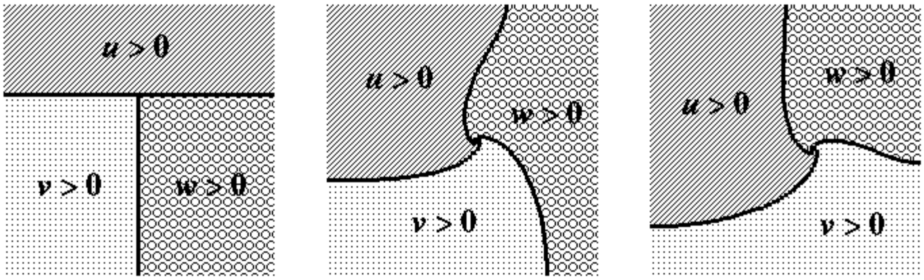


Figure 5: Numerical solutions by the present method for the three-component case in two dimensional space $(0,1)^2$. The solutions are drawn at $t = 0$ (left), $t = 0.5$ (center) and $t = 1$ (right). We can clearly observe the interfaces between regions $\{\Omega_i\}$.

Concluding remarks

A problem in mathematical biology is considered. The method, which we propose in this paper, has the advantages that we can determine the interfaces naturally and clearly as shown in Fig. 5 and that an implementation to the parallel computer can be easily done. We obtained 2.3 times speed-up by using 3 CPUs.

For the two-component case, we justified the method rigorously when $d_1 = d_2$. We can expect that the condition $d_1 = d_2$ is not essential.

Acknowledgments.

We would like to thank Mr. Atsushi Suzuki of Kyushu University for setting us a good computational environment.

References

[DHMP99]E. N. Dancer, D. Hilhorst, M. Mimura, and L. A. Peletier. Spatial segregation limit of a competition-diffusion system. *European J. Appl. Math.*, 10(2):97–115, 1999.

[Eva80]L. C. Evans. A convergence theorem for a chemical diffusion-reaction system. *Houston J. Math.*, 6(2):259–267, 1980.

[GLS94]W. Gropp, E. Lusk, and A. Skjellum. *Using MPI: Portable Parallel Programming with the Message-Passing Interface.* MIT press, 1994.

[IMY98]M. Iida, M. Mimura, and E. Yanagida. A free boundary problem as a singular limit of a competition-diffusion system. In Y. Nishiura, I. Takagi, and E. Yanagida, editors, *Proceedings of the International Conference on Asymptotics in Nonlinear Diffusive Systems*, pages 217–221, Sendai, 1998. Tohoku Univ.

[MBO94]B. Merriman, J. K. Bence, and S. J. Osher. Motion of multiple junctions: a level set approach. *J. Comput. Phys.*, 112:334–363, 1994.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau,

40. Domain Decomposition Method Applied to Radiation Problems

T. Kako [1], T. Kano[2], X.-J. Liu [3], T. Yamashita [4]

Introduction

The main purpose of this paper is to investigate some typical problems of wave motion in unbounded region which are related to radiation or scattering phenomena. The Helmholtz equation is one of the most important mathematical models which is used to describe the time harmonic behavior of various vibration and wave propagation phenomena.

The motivation of research is to understand main characteristics of wave propagation phenomena in obstacle scattering and/or wave radiation process through its numerical computation based on its mathematical analysis.

The importance of the wave propagation resides in the fact that it transmits information and transports energy. Some examples of research fields related to the wave propagation include acoustics, elasticity, electromagnetism with various applications such as sound emission from a speaker, human speech production, sound production of musical instruments, noise reduction, diagnostics or detection by ultrasonic wave, propagation of waves in optical fiber scope, heating by wave for various kinds of materials and others. Some of the characteristic quantities to be calculated in these problems include scattering amplitudes, transmission and reflection coefficients and resonance frequencies.

To investigate numerically the wave propagation phenomena in unbounded region using computers, we have to approximate the original problem which is formulated in some infinite dimensional function space by the one in an appropriate finite dimensional linear space. For this purpose, we first use the knowledge of the analytical properties of the solution to the original problem such as the radiation condition at infinity and/or the expression of the solution by a series of special functions or by an integral involving Green's function. We then reduce the problem into the boundary value problem in a bounded region with some truncation error for its solution and apply a finite element discretization method to get the linear equation in a finite dimensional approximation space.

Especially, we will show the effectiveness of the radiation condition at infinity which describes the asymptotic behavior of the solution and singles out the physical solution. We then use the domain decomposition method which divides the original problem in an unbounded region into the problem in a bounded region and the one in an outer region with simple shape.

[1] Department of Computer Science, The University of Electro-Communications, Chofu, Tokyo 182-8585, JAPAN. email: kako@im.uec.ac.jp

[2] Address same as above.

[3] Address same as above

[4] Hitachi, Ltd., JAPAN

More specifically, we treat the following three types of problems in different shapes of spatial regions.

The first one is a two-dimensional obstacle scattering problem, where we introduce a (higher order) radiation condition and the corresponding artificial boundary condition on the circular boundary of a truncated bounded disk region. The outer region is then the complement of the disk.

The second one is a two-dimensional half space problem where we consider a two component elastic wave propagation. There is a difficulty in this problem that the analytical asymptotic behavior at infinity is much more complicated than that in the scalar case due to the existence of the Rayleigh wave which propagates along the surface on the half space.

The third one is a two-dimensional wave-guide problem where we use the exact boundary condition given by the Diriclet to Neumann map on the boundary between a bounded region and an outer unbounded region which is cylindrical with a bounded cross section. We also consider a one-dimensional problem related to this original two-dimensional problem.

We will show some numerical examples in each case. In particular, in the second case, we discuss the relationship between 2D and 1D cases and show some numerical examples which indicate the efficiency of the 1D model as the good approximation of the 2D problem in the sense that it gives similar frequency response curves.

Mathematical Formulation

The main mathematical framework of the study consists of the scattering theory based on the perturbation theory for linear operators and the finite element method for partial differential equations.

The first difficulty in studying the radiation or scattering problem comes from the unboundedness of the region where we consider the partial differential equation and we have to choose an appropriate function space. The second problem we have to treat appropriately is the indefiniteness of the bilinear form which appears in the weak formulation used for the finite element method in the artificial bounded region and we have to consider the problem with non-real variables as well.

In this paper, we restrict our study to the two-dimensional case although the real physical phenomena occur in three-dimensional space. However, at least the theoretical part of our study can be extended to the three-dimensional case without any essential difficulty. The main problem we may have to solve is the practical computational complexity due to the large number of unknowns in 3D case and the shortage of memory and speed of the present computers together with the human resources in programming.

Two-dimensional wave propagation problem

The wave propagation phenomena in two-dimensional space R^2 can be described by the following mathematical model of the wave equation in $\Omega \subset R^2$:

$$\begin{aligned}(\frac{\partial^2}{\partial t^2} - \Delta)u(t,x,y) &= f(t,x,y) \text{ in } (-\infty,\infty)\times\Omega, \ \ \Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}, \quad (1)\\ (\alpha\frac{\partial}{\partial n} + \beta)u(t,x,y) &= g(t,x,y) \text{ on } (-\infty,\infty)\times\partial\Omega, \quad (2)\end{aligned}$$

where $\frac{\partial}{\partial n}$ denotes the outward normal derivative on the boundary $\partial\Omega$ of Ω.

In the following, we consider a stationary time harmonic solution of the problem: $u(t,x,y) = e^{i\omega t}u(x,y)$ for inhomogeneous data: $f(t,x,y) = e^{i\omega t}f(x,y)$ and $g(t,x,y) = e^{i\omega t}g(x,y)$ from which we can calculate almost every important quantity. Then u satisfies the Helmholtz equation:

$$\begin{aligned}(-\Delta - \omega^2)u(x,y) &= f(x,y) \text{ in } \Omega, \quad (3)\\ (\alpha\frac{\partial}{\partial n} + \beta)u(x,y) &= g(x,y) \text{ on } \partial\Omega \quad (4)\end{aligned}$$

with some radiation condition at infinity ($r = (x^2+y^2)^{1/2} \to +\infty$). For the existence and uniqueness of this problem, see [Wil75] or [ST70].

We assume that the boundary $\partial\Omega$ consists of two mutually distinct parts: $\partial\Omega = \Gamma_H \cup \Gamma_S$ where $g = g_S$ on the source boundary Γ_S and $g = 0$ on the homogeneous boundary Γ_H. The existence and uniqueness of the solution to this radiation or scattering problem can be proved by the limiting absorption principle which claims that the physical solution is the limit of the solution for the problem with positive absorption when the absorption tends to zero. In case that we know Green's function of the corresponding free space problem which satisfies the radiation condition at infinity, we can construct the solution solving the integral equation on the boundary.

Reduction to a problem in a bounded region

We introduce an artificial boundary in Ω which includes the source boundary Γ_S and we assume that the shape of the outside the boundary is simple. For example, it is the outside of a disk or a cylindrical region. The, using the knowledge of the solution outside the boundary we impose the boundary condition on the artificial boundary which may the Diriclet to Neumann (DtN in short) map or its approximation. We sometimes call it a radiation boundary condition (or artificial boundary condition).

The artificial boundary condition on the artificial boundary was introduced by B. Engquist - A. Majda [EM77], C. Goldstein [Gol81], T.Kako [Kak81], G.A. Kriegsman - C.S. Morawetz [KM80] and others. M. Masmoudi [Mas87] used the DtN map and there are several researches to this direction (see a book by D. Givoli [Giv92].

In the followings, we show more concretely three cases where we introduce different artificial boundary conditions for respective problems.

Radiation boundary conditions in obstacle scattering

In this section, using the analytic expression of solutions, we introduce a higher order radiation condition. We assume that Ω^c has a non-empty interior and includes the origin: $0 \in \Omega^c$. Choosing a number R_0 with the property: $\Omega^c \subset \mathbf{B}_{R_0} \equiv \{x \big| \, |x| \leq R_0\}$ and a smooth function $\chi_{R_0}(x)$ such that

$$\chi_{R_0}(x) = \left\{ \begin{array}{ll} 1 & (|x| \leq R_0) \\ 0 & (|x| \geq R_0 + 1), \end{array} \right. \tag{5}$$

we define a function $f \equiv (-\Delta - k^2)(1 - \chi_{R_0}(x))u(x)$. The zeroth order Hankel function of the first kind, $\frac{i}{4}H_0^{(1)}(k|x-x'|)$ is Green's function of $(\mathbf{H_t})$. Hence the solution of $(\mathbf{H_t})$ has the expression:

$$v(x) = \int_{B_{R_0+1} \setminus B_{R_0}} \frac{i}{4} H_0^{(1)}(k|x-x'|) f(x') dx'. \tag{6}$$

Using the asymptotic expansion formula for the Hankel function, putting $B(0) \equiv 1$ and defining the operators $L(p)$ and $B(p)$, $p = 1, 2, ...,$ as $L(p) \equiv \frac{1}{2ikp}\{\Lambda_\theta + p(p-1) + \frac{1}{4}\}$ and $B(p) \equiv L(p)L(p-1)...L(1)$, we have the following expression of the solution $u(r, \theta)$ an asymptotic expansion as r tends to infinity:

$$u(r, \theta) = \frac{1}{\sqrt{r}} e^{ikr} (\sum_{p=0}^{N} \frac{B(p)}{r^p}) a_0(\theta) + O(r^{-N-1-1/2}), \tag{7}$$

and we also have the asymptotic expansion:

$$\frac{\partial u}{\partial r} = iku - \frac{1}{2r}u + \frac{1}{\sqrt{r}} e^{ikr} (\sum_{p=1}^{N} \frac{-p}{r^{p+1}} B(p)) a_0(\theta) + O(r^{-N-2-1/2}). \tag{8}$$

In particular, we have, for $N = 1$,

$$u(r, \theta) = \frac{1}{\sqrt{r}} e^{ikr} (1 + \frac{1}{r} B(1)) a_0(\theta) + O(r^{-2-1/2}) \tag{9}$$

and

$$\frac{\partial u}{\partial r} - iku + \frac{1}{2r}u + \frac{1}{\sqrt{r}} e^{ikr} \frac{1}{r^2} B(1) a_0(\theta) = O(r^{-3-1/2}). \tag{10}$$

We define an operator $T_r \equiv \frac{1}{r}B(1)(1 + \frac{1}{r}B(1))^{-1}$. Since $B(1)$ is skew-selfadjoint, the operator T_r is bounded in $L^2(S^1)$ with norm $||T_r||_{L^2(S^1)} \leq 1$. Using this operator and eliminating $a_0(\theta)$ from the equations (9) and (10), we have the following theorem:

Theorem 2.1([LK98a]) *There exists one and only one solution of the Helmholtz equation (1) and (2) which satisfies the followings:*

$$\left\{ \begin{array}{rcll} -\Delta u(x) - k^2 u(x) & = & 0 & \text{in } \Omega^c, \\ u(x) & = & -\varphi_0 & \text{on } \partial\Omega, \\ ||\frac{\partial u}{\partial r} - iku + \frac{1}{2r}u + \frac{1}{r}T_r u||_{L^2(S^1)} & = & O(r^{-7/2}), & r \to \infty. \end{array} \right. \tag{11}$$

The equation (1) is considered in an unbounded region, which causes some difficulty to find approximate numerical solutions. To resolve this problem, we introduce a sequence of problems in bounded region. Put $R \gg 1$, and let u_R be the solution of the boundary value problem:

$$\left\{\begin{array}{rcll} -\Delta u_R - k^2 u_R & = & 0 & \text{in } \Omega_R^c \equiv \Omega^c \cap B_R, \\ u_R & = & -\varphi_0 & \text{on } \partial\Omega, \\ \frac{\partial u_R}{\partial r} - iku_R + \frac{1}{2R}u_R + \frac{1}{R}T_R u_R & = & 0 & \text{on } S_R = \partial B_R. \end{array}\right. \tag{12}$$

If we introduce the operators H_R and Q_R as $H_R u = -\Delta u$ with $\mathcal{D}(H_R) \equiv \{u|\ u \in H^2(\Omega_R^c), u|_{\partial\Omega} = 0$ and $\frac{\partial u}{\partial r}|_{S_R} = 0$ on $S_R\}$ and $Q_R u = (\frac{2}{R}T_R + \frac{1}{r} - 2ik)\frac{\partial u}{\partial r} - \{(\frac{1}{R}T_R)^2 + \frac{1}{4r^2} - 2ik\frac{1}{R}T_R\}u - u$ with $\mathcal{D}(Q_R) = \mathcal{D}(H_R)$, the equation (1) becomes an operator equation:

$$(H_R + 1)w_R + Q_R w_R = f_R. \tag{13}$$

We have the following theoretical result for the unique existence of the solution:

Theorem 2.2([LK98a]) *The equation (13) has a unique solution in $L^2(\Omega_R^c)$ which is given as*

$$w_R = (H_R + 1)^{-1}(1 + Q_R(H_R + 1)^{-1})^{-1} f_R. \tag{14}$$

The proof of this theorem is given by using Rellich's compactness theorem and the Fredholm alternative theorem. We can estimate the difference $e_R \equiv u - u_R$ as follows:

Theorem 2.3([LK98a]) *When $R \gg 1$, for a fixed R_0, the following estimates hold with some constant C which is independent of R:*

$$\int_{S_R} |e_R|^2 dS_R \le CR^{-6} \qquad \textit{and} \qquad \sup_{x \in B_{R_0}} |e_R(x)| \le CR^{-3}. \tag{15}$$

Radiation boundary condition for seismic wave

In the case of the elastic wave in half space which describes the seismic wave, we have to treat correctly the Rayleigh wave which propagate along the boundary surface. As far as we know, the asymptotic behavior of wave motion at infinity is not well investigated. This makes it difficult to introduce the reasonable artificial boundary and the boundary condition on it. In [YT97] , T. Yamashita and the present author proposed the artificial boundary condition on the half circle and the radiation boundary condition which is the linear combination of those for bulk P and S waves and that for the Rayleigh wave.

The basic time harmonic governing equation is written as

$$\rho\omega^2\mathbf{u} = (\lambda + 2\mu)\ \text{grad div}\ \mathbf{u} - \mu\ \text{rot rot}\ \mathbf{u} \quad \text{in } \Omega\backslash\mathbf{S}, \tag{16}$$

$$\mathbf{u} = \mathbf{f}(\mathbf{x}) \quad \text{in } \mathbf{S}, \tag{17}$$

$$\sigma(\mathbf{u}) = \mathbf{0} \quad \text{on } \Gamma_F, \tag{18}$$

where Γ_F is a boundary surface and we imposed the free traction condition with surface traction force $\sigma(\mathbf{u})$.

We need, in this time independent case, some asymptotic condition at infinity. Using this condition, we might get the artificial boundary condition on the artificial boundary which we take the half circle with radius R. The heuristic radiation boundary condition which we impose on this half circle is given as

$$\mathcal{D}\mathbf{u} = \mathbf{0} \quad \text{on } \Gamma_R, \tag{19}$$

$$\mathcal{D}\mathbf{u} \equiv i\rho\omega \begin{pmatrix} n_1 & n_2 \\ n_2 & -n_1 \end{pmatrix} \begin{pmatrix} V_{RP}(\mathbf{x}) & 0 \\ 0 & V_{RS}(\mathbf{x} \end{pmatrix} \begin{pmatrix} n_1 & n_2 \\ n_2 & -n_1 \end{pmatrix} \mathbf{u} + \sigma(\mathbf{u}),$$

where

$$\begin{aligned} V_{RP}(\mathbf{x}) &\equiv V_P - (V_P - V_R)\exp\left(-\omega x_2(1 - V_R^2/V_P^2)^{1/2}/V_R\right), \\ V_{RS}(\mathbf{x}) &\equiv V_S - (V_S - V_R)\exp\left(-\omega x_2(1 - V_R^2/V_S^2)^{1/2}/V_R\right). \end{aligned}$$

where V_P, V_S and V_R are the wave speeds of the primary, the secondary and the Rayleigh waves respectively. The main idea of this condition is to mix up the transparent conditions for respective waves in case of a plain wave with the ratio of the amplitude of the Rayleigh wave which decreases exponentially to the perpendicular direction to the free surface. The theoretical as well as numerical analysis for this approximation method is a future work.

Dirichlet to Neumann map in 2D wave-guide

In the case of 2D wave-guide problem with a cylindrical unbounded semi-infinite channel, the radiation condition in the cylindrical is written as:

$$\frac{\partial p}{\partial n}(= \frac{\partial p}{\partial x}) = \Lambda p \quad \text{on}\Gamma_R, \tag{20}$$

where Γ_R is an artificial boundary which is a cross section of the cylindrical region and Λ is the Dirichlet to Neumann map in the outer cylindrical region given as

$$\Lambda p = \sum_{n=0}^{\infty} \gamma_n C_n(p) \cos(\frac{n\pi}{L} y) \tag{21}$$

with

$$C_n(p) = \begin{cases} \frac{1}{L}\int_0^L p(x,y)dy & (n = 0) \\ \frac{2}{L}\int_0^L p(x,y)\cos(\frac{n\pi}{L}y)dy & (n \geq 1), \end{cases} \tag{22}$$

$$\gamma_n \begin{cases} i\zeta_n, & \zeta_n = \{\omega^2 - (\frac{n\pi}{L})^2\}^{1/2}, \quad 0 < \frac{n\pi}{L} < \omega \\ -\eta_n, & \eta_n = \{(\frac{n\pi}{L})^2 - \omega^2\}^{1/2}, \quad \omega \leq \frac{n\pi}{L}. \end{cases} \tag{23}$$

Then the Helmholtz equation in the inner domain Ω_i is given as:

$$\begin{array}{rcl} (-\omega^2 - \Delta)p & = & 0 \ \text{ in } \Omega_i, \\ \dfrac{\partial p}{\partial n} = 0 \text{ on } \Gamma_H, \quad \dfrac{\partial p}{\partial n} & = & g_S \ \text{ on } \Gamma_S, \quad \dfrac{\partial p}{\partial n} = \Lambda p \ \text{ on } \Gamma_R. \end{array} \tag{24}$$

Related to this 2D wave-guide problem, we can consider the corresponding 1D Webster's horn equation given as:

$$-\frac{\partial v}{\partial t}\frac{A(x)}{\rho}\frac{\partial p}{\partial x}, \qquad -\frac{\partial p}{\partial t}\frac{\rho c^2}{A(x)}\frac{\partial v}{\partial x}, \tag{25}$$

where p is the pressure and v is the velocity, and $A(x)$ denotes the area of the cross section. Eliminating v, we have the 1D approximation model called Webster's horn equation:

$$\frac{\partial^2 p}{\partial t^2} - \frac{1}{A(x)}c^2\frac{\partial}{\partial x}(A(x)\frac{\partial p}{\partial x}) = 0. \tag{26}$$

Week Formulation and Discretization

In this paper, we use the finite element method to dicretize the problem in the artificially truncated region with an artificial boundary condition. We start with a weak formulation of the problem in an appropriate closed subspace $\mathcal{V}$ of the Sobolev space $H^1(\Omega_i)$ defined through the boundary condition and then restrict the problem into a finite dimensional subspace of $\mathcal{V}$ which is a set of all piece-wise linear continuous functions in $\mathcal{V}$ with respect to a regular triangulation of Ω_i. We note that we have to introduce an appropriate approximation of the boundary integral which corresponds to the non-local boundary condition such as the higher order radiation boundary condition or the Dirichlet to Neumann map. In the following, we show the case of the 2D wave-guide problem in some detail.

Application to 2D wave-guide problem

The weak formulation for the Helmholtz problem (3) and (4) with the artificial boundary condition is given as:

Find $p \in \mathcal{V} \subset H^1(\Omega)$:

$$a(p,q) = (f,q)(= a_0(g,q)) \qquad \forall q \in \mathcal{V}$$

where, together with its approximation $a_N(\cdot,\cdot)$,

$$\begin{array}{rcl} a(p,q) & = & a_0(p,q) + b_1(p,q) + b_2(p,q), \\ a^N(p,q) & = & a_0(p,q) + b_1(p,q) + b_2^N(p,q) \end{array}$$

with

$$\begin{aligned}
a_0(p,q) &= \int_\Omega \nabla p \cdot \overline{\nabla q} + p\overline{q}dxdy, \\
b_1(p,q) &= -(\omega^2+1)\int_\Omega p\overline{q}dxdy, \\
b_2(p,q) &= -(\Lambda p(x_R,\cdot), q(x_R,\cdot)) = b_{2,i}(p,q) + b_{2,r}^\infty(p,q), \\
b_{2,i}(p,q) &= -i\omega L C_0(p)C_0(q) - i\sum_{0<\frac{n\pi}{L}<\omega} \zeta_n(\frac{L}{2})C_n(p)C_n(q), \\
b_{2,r}^\infty(p,q) &= \sum_{\omega\leq\frac{n\pi}{L}} \eta_n(\frac{L}{2})C_n(p)C_n(q),
\end{aligned}$$

where ζ_n and η_n are all nonnegative constants in (23), and

$$\begin{aligned}
b_2^N(p,q) &= -(\Lambda^N p(x_R,\cdot), q(x_R,\cdot)) = b_{2,i}(p,q) + b_{2,r}^N(p,q), \\
b_{2,r}^N(p,q) &= \sum_{\frac{L}{\pi}\omega\leq n\leq N} \eta_n(\frac{L}{2})C_n(p)C_n(q).
\end{aligned}$$

Now the finite element method is formulated as:

Find $p_h \in \mathcal{V}_h \subset H^1(\Omega)$:

$$a(p_h,q_h) = (f,q_h)(= a_0(g,q_h)) \qquad \forall q_h \in \mathcal{V}_h.$$

Error Analysis

We develop the error analysis for the finite element discretization for the Helmholtz equation with the DtN boundary condition. We give rather abstract results which is essentially known but in an operator theoretical formulation. In application to 2D wave-guide problem, we use the result of Mikhlin (see [Mik64]) and the results of compact perturbation theory as well as the uniqueness of the analytic solution.

Abstract results for error analysis of finite element method

We consider the following four problems:

1: $(\mathrm{E})_\mathrm{w}$: Find $u \in \mathcal{V}$ such that

$$a(u,v) = (f,v) \quad \text{for all} \quad v \in \mathcal{V}.$$

2: $(\mathrm{E_h})_\mathrm{w}$: Find $u_h \in \mathcal{V}_h$ such that

$$a(u_h,v_h) = (f,v_h) \quad \text{for all} \quad v_h \in \mathcal{V}_h.$$

3: $(\mathrm{E^N})_\mathrm{w}$: Find $u^N \in \mathcal{V}$ such that

$$a^N(u^N,v) = (f,v) \quad \text{for all} \quad v \in \mathcal{V}.$$

4: $(\mathrm{E}_{\mathrm{h}}^{\mathrm{N}})_{\mathrm{w}}$: Find $u_h^N \in \mathcal{V}_h$ such that

$$a^N(u_h^N, v_h) = (f, v_h) \quad \text{for all} \quad v_h \in \mathcal{V}_h.$$

Then, we have the above four equations are equivalent to the following operator equations respectively:

1. $(\mathrm{E})_{\mathrm{op}}: \quad Au = f.$

2. $(\mathrm{E}_{\mathrm{h}})_{\mathrm{op}}: \quad A_h u_h = f_h \quad \text{with} \quad A_h = P_h A, f_h = P_h f.$

3. $(\mathrm{E}^{\mathrm{N}})_{\mathrm{op}}: \quad A^N u^N = f.$

4. $(\mathrm{E}_{\mathrm{h}}^{\mathrm{N}})_{\mathrm{op}}: \quad A_h^N u_h^N = f_h \quad \text{with} \quad A_h^N = P_h A^N, f_h = P_h f.$

By Riesz's representation theorem, two operators A and A_N are defined as:

$$a(u, v) = (Au, v) \quad \text{and} \quad a^N(u, v) = (A^N u, v) \quad \text{for all} \quad v \in \mathcal{V}.$$

Using the relations $Au = A^N u^N = f$ and

$$P_h A u_h = A_h u_h = f_h = A_h^N u_h^N = P_h f = P_h A u = P_h A^N u^N,$$

we can transform the expression of the error $u - u_h^N$ as follows:

$$\begin{aligned}
u - u_h^N &= u - v_h + v_h - u_h^N \\
&= u - v_h + (A_h^N)^{-1} A_h^N v_h - u_h^N \\
&= u - v_h + (A_h^N)^{-1} A_h^N v_h - (A_h^N)^{-1} f_h \\
&= u - v_h + (A_h^N)^{-1} A_h^N v_h - (A_h^N)^{-1} P_h f \\
&= u - v_h + (A_h^N)^{-1} A_h^N v_h - (A_h^N)^{-1} P_h A u \\
&= u - v_h + (A_h^N)^{-1} \{A_h^N v_h - P_h A u\} \\
&= u - v_h + (A_h^N)^{-1} \{P_h A^N v_h - P_h A u\} \\
&= u - v_h + (A_h^N)^{-1} \{P_h A^N (v_h - u) + P_h A^N u - P_h A u\} \\
&= \{I - (A_h^N)^{-1} P_h A^N)\}(u - v_h) + (A_h^N)^{-1} P_h (A^N - A) u.
\end{aligned}$$

Hence we can estimate the above difference as:

$$\|u - u_h^N\| \le (I + \|(A_h^N)^{-1}\| \|A^N\|) \inf_{v_h \in \mathcal{V}_h} \|u - v_h\| + \|(A_h^N)^{-1}\| \|(A^N - A)u\|.$$

Therefore, our next task is to prove the followings:

1. The uniform boundedness of $\|(A_h^N)^{-1}\|$: $\|(A_h^N)^{-1}\| \le M < +\infty$ with respect to h and N.

2. The truncation error estimate: $\|(A^N - A)u\| \le \frac{C}{N^\alpha} \|u\|_{\mathcal{W}}$ under the regularity condition for u: $u \in \mathcal{W} \subset \mathcal{V}$.

Actually, we have proved these conditions for the obstacle scattering case in [LK98a]. In the next section, we treat the case of wave-guide.

Application to the wave-guide problem

We can apply the abstract error estimation based on the following observations:

1. The sesquilinear form $b^{\infty}_{2,r}(p,q)$ is bounded and nonnegative in $\mathcal{V}$. Hence $a_{0,DN}(p,q) \equiv a_0(p,q) + b^{\infty}_{2,r}(p,q)$ is an inner product in $\mathcal{V}$

2. The form $b_1(p,q) + b_{2,i}(p,q)$ is compact with respect to $a_{0,DN}(p,q)$ in $\mathcal{V}$.

3. We can then apply the results by Mikhlin [Mik64] (see also Kako [Kak81]) and we can prove the convergence of the finite element method under some additional condition on the non-existence of a positive eigenvalue.

4. The difference between $a(p,q)$ and $a^N(p,q)$ is written as:

$$a(p,q) - a^N(p,q) = \sum_{N<n} \eta_n(\frac{L}{2})C_n(p)C_n(q) = (\{\Lambda - \Lambda^N\}p, q).$$

   and $\|\{\Lambda - \Lambda^N\}p\|_{L^2(0,L)}$ tends to zero exponentially with respect to N or estimated from above by $\frac{C}{N^\alpha}\|u\|_{\mathcal{W}}$ with any α and a corresponding higher order Sobolev space $\mathcal{W}$.

Some Numerical Examples

In this section, we show some numerical examples calculated by using the methods introduced in the previous sections.

Obstacle scattering (by X.-J. Liu)

Fig.1 shows a typical wave profile computed by the method introduced in [XJK96], [LK98a] and [LK98b].

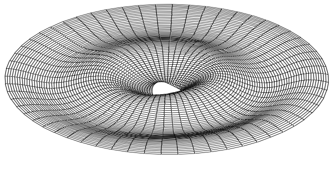


Figure 1: Wave profile of 2D obstacle scattering

Seismic wave in 2D foundation (by T. Yamashita)

We show two numerical results in Fig.2 where a single source is placed inside the foundation [YT97]). The left figure is the case of the artificial boundary with radius $R = 1$ and the right one is the case with $R = 1,25$. There is a good coincidence between these two results and the Rayleigh wave is well captured.

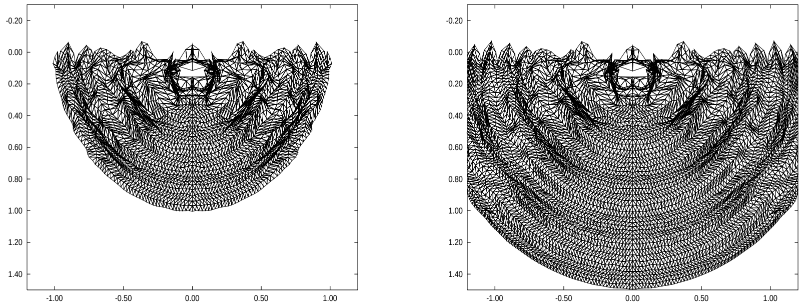


Figure 2: Stationary 2D elastic wave propagation

Voice generation problem (by T. Kano)

Lastly, we show an numerical example of 2D wave propagation in the vocal tract open to an infinite cylinder. The Fig.3 shows a wave profile with a time frequency 7.5 kHz. The source is placed on the left edge and the right side is a radiation boundary. The figure on the right shows a frequency response curve measured at the mid point on the radiation boundary. We can see that, as the shape of the vocal tract becomes flatter, the response curve approaches nearer to the one of 1D model.

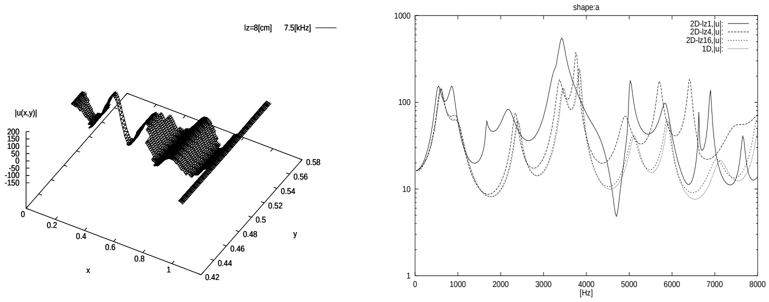


Figure 3: Comparison between 1D and 2D frequency response curves

Concluding Remarks

We have developed a methodology to calculate problems in several unbounded regions by use of the DtN mapping or its approximations. Error analysis is given as an extension of the standard method. Application to problems having resonance phenomena is presented and some typical phenomena have been captured in these numerical experiments. Applications to more realistic industrial problems are future subject.

References

[EM77]B. Engquist and A. Majda. Absorbing boundary conditions for the numerical simulation of waves. *Math. Comp.*, 31(139):629–651, 1977.

[Giv92]D. Givoli. *Numerical methods for problems in infinite domains.* Elsevier, 1992.

[Gol81]C. Goldstein. The finite element method with non-uniform mesh sizes applied to the exterior Helmholtz problem. *Numer. Math.*, 1981.

[Kak81]T. Kako. Approximation of the scattering state by means of the radiation boundary condition. *Math. Meth. in the Appl. Sci.*, 3:506–515, 1981.

[KM80]G.A. Kriegsman and C.S. Morawetz. Solving the Helmholtz equation for exterior problems with variable index of refraction. *SIAM J. Sci. Stat. Comput.*, 1980.

[LK98a]X.-J. Liu and T. Kako. Higher order radiation boundary condition and finite element method for scattering problem. *Advances in Mathematical Sciences and Applications*, 8:801–819, 1998.

[LK98b]X.-J. Liu and T. Kako. Iteration algorithm for the Helmholtz equation by means of domain decomposition method. *Transaction of Japan SIAM*, 8(4):435–446, 1998.

[Mas87]M. Masmoudi. Numerical solution for exterior problems. *Numer. Math.*, 51:87–101, 1987.

[Mik64]S.G. Mikhlin. *Variational Methods in Mathematical Physics.* Pergamon, Oxford, 1964.

[ST70]N. Shenk and D. Thoe. Outgoing solutions of $(-\delta + q - k^2)u = f$ in an exterior region. *J. Math. Anal. Appl.*, 31:81–96, 1970.

[Wil75]C. Wilcox. *Scattering Theory for the d'Alembert Equation in Exterior Domains*, volume 442 of *Lecture Notes of Mathematics.* Springer, Berlin, 1975.

[XJK96]X.-J. Liu X.-J and T. Kako. Numerical calculation of scattering state by means of higher order radiation boundary condition. *RIMS Kokyuroku (Kyoto University)*, 944:68–76, 1996.

[YT97]T. Yamashita and T.Kako. Finite element method for elastic wave in 2-dimensional semi-infinite region, in japanese. Proceedings of the Conference on Computational Engineering Science, Vol.2., May 1997.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau, ©2001 DDM.org

41. Application of the Domain Decomposition Method to the Flow around the Savonius Rotor

Testuya Kawamura[1], Tsutomu Hayashi[2], Kazuko Miyashita[3]

Introduction

In this study, we focus on the Savonius Rotor and try to compute the flow field under its operation and make clear the running performance by means of the numerical simulation. Our final objective is to simulate the flow field around the whole rotor and estimate the effect of the sidewall or the other rotor. Incompressible Navier-Stokes equations are solved in a few regions separately where the fixed coordinate and the rotating coordinate are used respectively. We employ domain decomposition method in order to connect these regions with adequate accuracy. The basic equations in each region are solved by using standard MAC method[HW65]. The physical quantities such as the velocity and the pressure in each region are transferred through the overlapping region, which is common in each domain. Reasonable results are obtained in the present calculations.

Recently, the wind force is widely recognized as the environmentally friendly energy and attracts public attention. The wind power plant using windmills is the typical example. In order to make an effective windmill, it is very important to analyze the flow field around a windmill. In this case, numerical simulation becomes a promising method. The most important part of the investigation is to analyze the flow field near the rotating rotor of the windmill. On the other hand, it is also very important to investigate the interaction among the windmills if they are placed without long distance.

For the numerical simulation of rotating body, it is convenient to use the rotating coordinate system, which rotates with the same speed. However, if there is another body which is not rotating or if there are many rotors which rotate at different position and with different speed, it is very difficult to choose one special rotating coordinate system. In these situations, it is natural to use many coordinate systems separately, which are suitable for the flow simulation around each rotor and connect these coordinates adequately. We focus on these points and simulate the flow fields around a windmill by using domain decomposition method in which the whole computation region is divided into several domains and they are connected adequately.

The Savonius rotor[Sav31] is chosen for the simulation since in this case the rotating bluff body generates the complex flow field with large separation and it is very interesting to investigate such flow from the fluid dynamical point of view. Figure 1 is the schematic figure of the Savonius rotor. The features of this windmill are easy to make, independent of the direction of the wind, low speed and high torque. The Savonius rotor is usually used as the pump.

[1] Ochanomizu University, kawamura@ns.is.ocha.ac.jp
[2] Tottori University, hayashi@damp.tottori-u.ac.jp
[3] Ochanomizu University, miya@ns.is.ocha.ac.jp

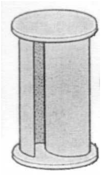


Figure 1: Savonius rotor

There are several experimental and numerical works concerning with Savonius rotor [RESF78] [Oga83] [IST94]. Among them, Ishimatsu et al.[IST94] calculated the flow around a Savonius rotor. Their objective is to compute running performance of one windmill. Therefore, they ignore the effect of the sidewall, ground and other windmills. Their numerical method is based on the finite volume method with unstructured grids. As is discussed above, one of the important objectives of the present study is to investigate the effect of the obstacles. Therefore, we employ the domain decomposition method in this study.

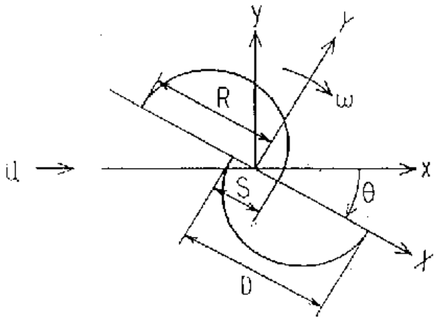
Numerical Method

Since the rotational frequency is low enough, the flow around the Savonius rotor is assumed as incompressible. The basic equations are

$$\nabla \mathbf{v} = 0$$

$$\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla \mathbf{v}) = -\nabla p + \frac{1}{Re} \nabla^2 \mathbf{v}$$

where Re is the Reynolds number. We use both Cartesian coordinate system (x,y) and



R: radius of rotation
D: bucket diameter
ω: angular velocity
θ: attack angle

Figure 2: Savonius Rotors without walls & obstacle

the rotating coordinate system (X,Y) which rotates around vertical axis with constant angular velocity ω. If we use the symbols indicated in Figure 2, the relation between two coordinate systems is

$$X = x\cos\theta - y\sin\theta,$$

$$Y = x\sin\theta + y\cos\theta,$$

where θ is the angle between two coordinate systems. The basic equations are expressed in the rotating coordinate system as

$$\frac{\partial U}{\partial X} + \frac{\partial V}{\partial Y} = 0$$

$$\frac{\partial U}{\partial t} + U\frac{\partial U}{\partial X} + V\frac{\partial U}{\partial Y} - \omega^2 X + 2\omega V = -\frac{\partial P}{\partial X} + \frac{1}{Re}\left(\frac{\partial^2 U}{\partial X^2} + \frac{\partial^2 U}{\partial Y^2}\right)$$

$$\frac{\partial V}{\partial t} + U\frac{\partial V}{\partial X} + V\frac{\partial V}{\partial Y} - \omega^2 Y - 2\omega U = -\frac{\partial P}{\partial Y} + \frac{1}{Re}\left(\frac{\partial^2 V}{\partial X^2} + \frac{\partial^2 V}{\partial Y^2}\right)$$

where (U,V) are the velocity components in (X,Y) direction while (u,v) are those in the fixed Cartesian coordinate system. These velocity components are connected to each other through the following relations:

$$U = u\cos\theta - v\sin\theta - \omega Y,$$

$$V = u\sin\theta + v\cos\theta + \omega X.$$

We use two computational domains. One domain(region1) includes the rotating rotor and another(region2) includes the fixed walls. Since the shape of the Savonius rotor is semicircular, it is convenient to use a semicircular region. The region including rotors consists of two semicircular regions whose centers are located at different positions. These two regions are connected by one line which passes two centers as is shown in Figure 3. Clearly, it is convenient to use the grid system based on the cylindrical coordinate. Another domain(region2) is rectangular and includes the fixed walls(Figure 4). The Cartesian coordinate system is used and the non-uniform rectangular grid is employed in this region. The grid points do not coincide with each other in both x(X) and y(Y) direction. The computations in the two domains, which have the overlapping region are performed alternatively at every time step. Figure 5 indicates the whole computational region. The physical quantities (velocity and pressure) are exchanged through the common overlapping region as is shown in Figure 6. When we compute

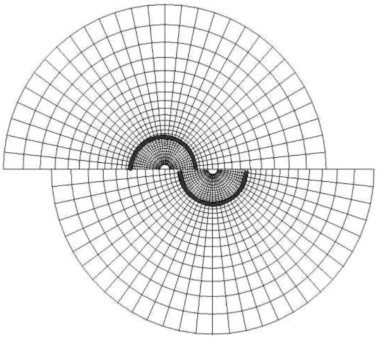


Figure 3: Inner region(region1). The bold lines indicate two blades

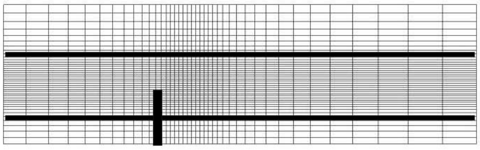


Figure 4: Outer region(region2). The bold lines indicate the sidewalls

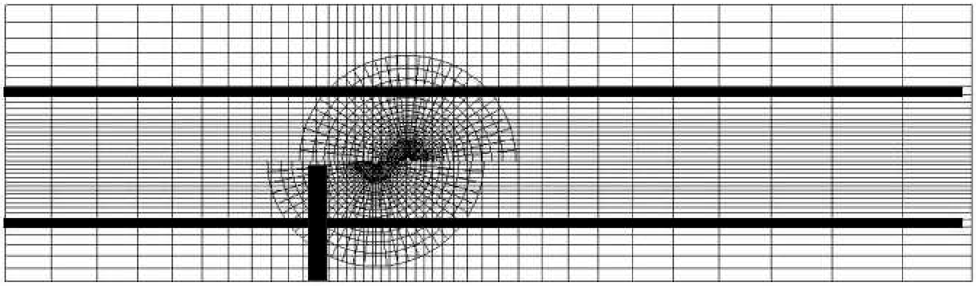


Figure 5: Whole computational region

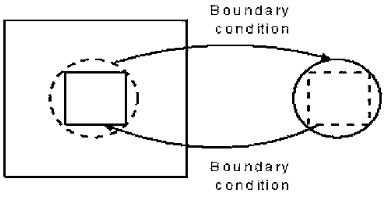


Figure 6: Domain decomposition by the overlapping region

the flow field of region1, the boundary conditions are required on each boundary. If the boundary locates outside of the region, the boundary conditions are determined by the usual way, i.e. free stream condition or something like this. If the boundary locates inside of the region2, the boundary values can be obtained from the computational results of the region2. In this case, some interpolations are required since the grid systems in both regions are different. In this study, the interpolation shown in Figure 7 is used. Since this formula requires only the distance from the four corners in one grid cell, it can be used even if the grid cell is highly deformed.

$$f_P = \frac{1}{R}\left(\frac{1}{r_Q}f_Q + \frac{1}{r_R}f_R + \frac{1}{r_S}f_S + \frac{1}{r_T}f_T\right)$$

$$where\ \ R = \frac{1}{r_Q} + \frac{1}{r_R} + \frac{1}{r_S} + \frac{1}{r_T}$$

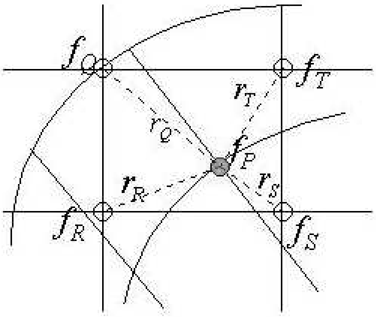


Figure 7: Interpolation in the overlapping region

Similar technique is used for determining the boundary conditions of region2 from the computational results of region1. In region1, two regions of semicircular shape are connected through one line without overlapping region. The boundary conditions on this line are given by the average value of the nearest grid points in each region(Figure 8) as follows: The numerical method to solve incompressible Navier-Stokes equation

$$f_P = (f_Q + (1-r)f_R + rf_S)/2.$$

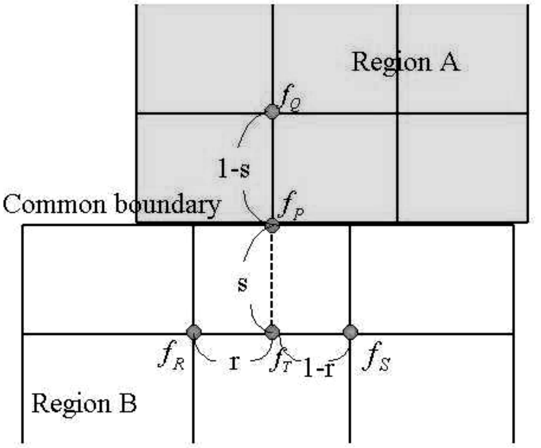


Figure 8: Interpolation along the line

is the standard MAC method. All the spatial derivatives except the nonlinear term of the Navier-Stokes equation is approximated by the second order central difference. Nonlinear terms are approximated by the third order upwind scheme[KK84] due to the numerical stability. Euler explicit scheme is employed for the time integration.

Result

Typical results obtained by the present study are shown here. The dimensionless gap width(=S/D, see Figure 2) is chosen to 0.15 and tip speed ratio $\lambda(= R\omega/u_\infty)$ is changed from 0.25 to 1.25. Figure 9 indicates the initial position of the rotor. In this case, the rotational angle θ is defined as zero and the rotor begins to rotate clockwise from this position. Figure 10 is an example of the instantaneous velocity vectors. Both the vectors in the inner region and the outer region are plotted in the same figure. The vectors vary continuously from the inner region to the outer region, which indicates the interpolation works well in this calculation. Figure 11 is time history of the torque coefficient. The torque coefficient $C_r(= T/qRA$ where T is the torque, q is the dynamic pressure, R is the radius of the rotor, and A is the projection area). The tip speed ratio is 0.25 and no walls exist. Clearly, it has a period of 180 degree. The torque becomes maximum and minimum around 30 and 150 degree respectively and becomes zero around 120 and 180 degree. Figure 12 is also time history of the

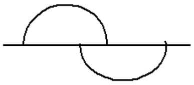


Figure 9: Initial position of the rotor

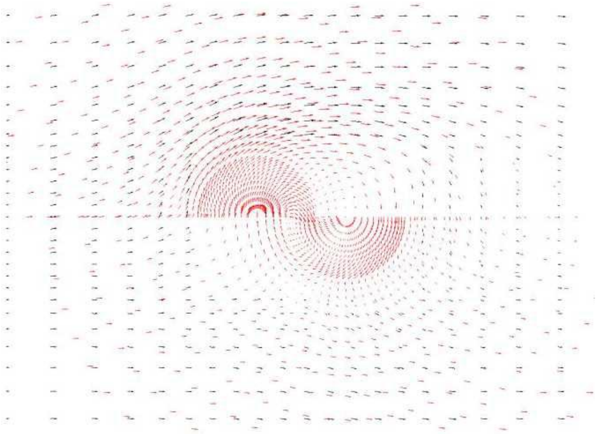


Figure 10: An example of the instantaneous velocity fields in the whole region

torque coefficient but the tip speed ratio is 0.5 and 0.75. As tip speed ratio becomes large, the negative part of the curve becomes large indicating the total torque becomes small. Figure 13 is the result of the calculation with walls. It corresponds to Figure 11 and Figure 12. Although the shape of each curve is similar, the absolute value becomes large for the latter case. Figure 14 is the time-averaged torque coefficient

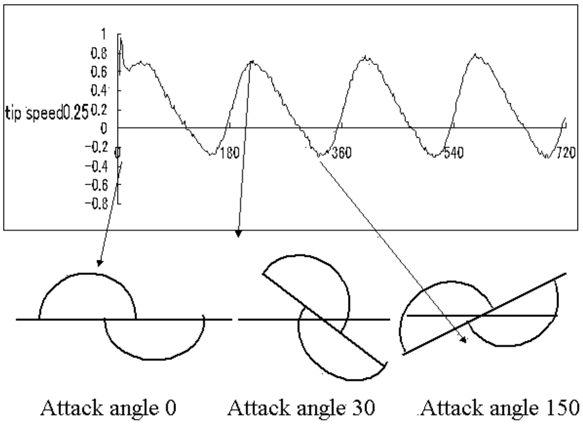


Figure 11: Time history of the torque coefficient without walls(Tip speed ratio is 0.25)

for various tip speed ratio λ. Both the results of the calculations with and without walls are indicated in the same figure. Torque coefficients decrease nearly linear as the tip speed ratio increases and become negative around 0.8. They become almost twice when the walls exist. Figure 15 is the time-averaged power coefficients $C_p(=\lambda C_r)$ for various tip speed ratio. Both the results with and without walls are shown. The power coefficient has its maximum value around $\lambda = 0.5$ and $\lambda = 0.4$ for the case with and without walls respectively. The maximum value is almost twice for the case with

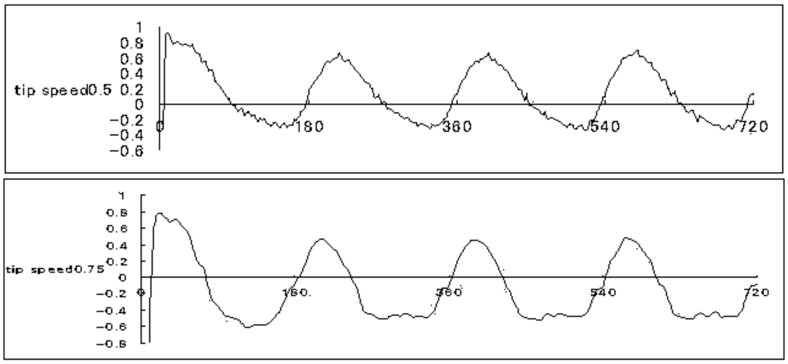


Figure 12: Time history of the torque coefficient without walls(Tip speed ratio is 0.5 and 0.75)

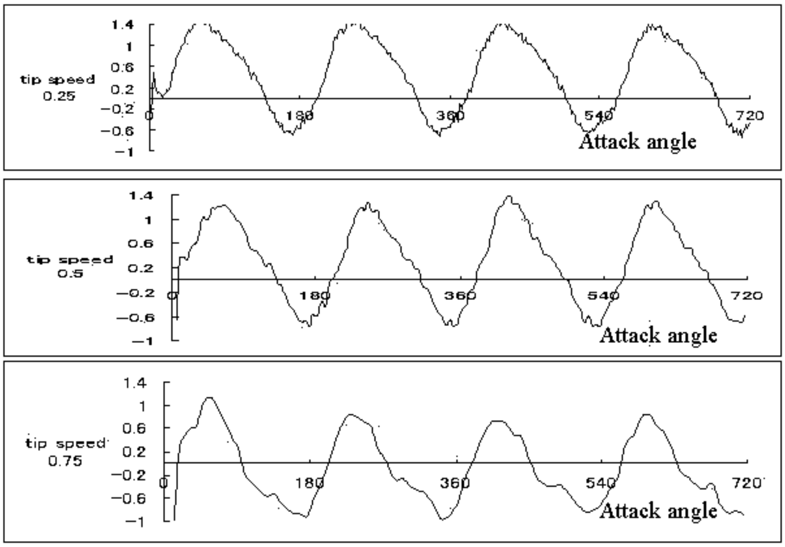


Figure 13: Time history of the torque coefficient with walls(Tip speed ratio is 0.25, 0.5 and 0.75)

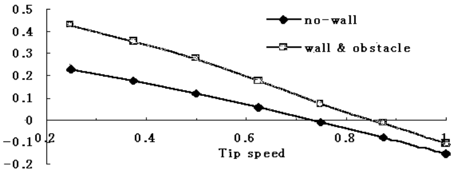


Figure 14: Time-averaged torque coefficient for various tip speed ratios for the cases with and without walls

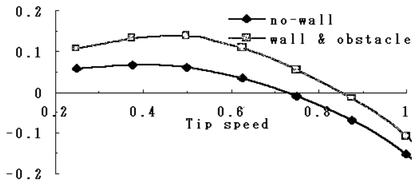


Figure 15: Time-averaged power coefficient for various tip speed ratios for the cases with and without walls

walls.

Summary

In this study, the flow field around the windmill is computed by using domain decomposition method. Although the Savonius rotor is chosen for the present computation, this method can be applied for the computations of other windmills. Two computational domains are used and connected to each other. One domain contains the rotating rotor and rotational coordinate system is employed. Another contains the fixed walls and the Cartesian coordinate system is used. Both regions have common overlapping region. The physical quantities on the boundary of one domain in the overlapping region are calculated by interpolating the physical values at the grid points in another region which are located inside of the region. The running performance of the Savonius rotor such as the torque coefficient and the power coefficient is obtained for various tip speed ratios. The effect of the walls on the running performance is also investigated. It is found that torque coefficient and the power coefficient become almost twice when the walls are placed adequately.

References

[HW65]F. H. Harlow and J. E. Welch. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *Physics of Fluids*, 8(12):2182–2189, December 1965.

[IST94]K. Ishimatsu, T. Shinohara, and F. Takuma. Numerical simulation for savonius rotors(running performance and flow fields). *JSME(B)*, 60(569):154–160, 1994.

[KK84]T. Kawamura and K. Kuwahara. Computation of high Reynolds number flow around a circular cylinder with surface roughness. *AIAA paper*, 84(0340), 1984.

[Oga83]Ogawa. The study on the savonius wind turbine (1st. report; theoretical analysis). *JSME*, 49(441), 1983.

[RESF78]B. F. Blackwell R. E. Sheldahl and L. V. Feltz. Wind tunnel performance data for two- and three-bucket savonius rotors. *J. Energy*, 1978.

[Sav31]S. J. Savonius. The s-roter and its application. *Mech. Eng.*, 53:333, 1931.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau,

42. An Artificial Boundary Condition for the Numerical Computation of Scattering Waves

D. Koyama[1]

Introduction

We consider the *controllability method*, which is proposed by Bristeau-Glowinski-Périaux [BGP98], for computing numerical solutions of the exterior problem for the Helmholtz equation. In the controllability method, we need to introduce an artificial boundary in order to reduce the computational domain to a bounded domain, and need to solve, in the bounded computational domain, the wave equation and an elliptic problem iteratively. We first introduce a new *artificial boundary condition* (ABC) for the wave equation, which is suitable for the controllability method. Our ABC is constructed by using the *Dirichlet-to-Neumann* (DtN) operator associated with the Helmholtz equation. We next discuss uniqueness for *semi-discrete solution* of the controllability method in the case when the artificial boundary is a circle. Then we need spectral properties of the DtN operator, which are deduced from some properties of the Hankel functions. We finally present numerical examples, which show that numerical solutions obtained by using our ABC are more accurate than those obtained by using another well-known ABC, and that by using our ABC, accurate numerical solutions are obtained whether the artificial boundary is large or small. These numerical results suggest that by using our ABC and by taking a small artificial boundary, we can reduce the computational costs.

We consider the exterior problem for the Helmholtz equation:

$$\left\{\begin{array}{rcll} -\Delta U - k^2 U &=& 0 & \text{in } \Omega, \\ U &=& G & \text{on } \gamma, \\ \displaystyle\lim_{r\longrightarrow+\infty} r^{\frac{1}{2}}\left(\frac{\partial U}{\partial r} - ikU\right) &=& 0 & \text{(outgoing radiation condition).} \end{array}\right. \tag{1}$$

Here k is a positive constant and Ω is an unbounded domain of $\mathbb{R}^2$ with boundary γ. We assume that $\mathcal{O} = \mathbb{R}^2 \setminus \overline{\Omega}$ is a bounded open set. Further G is a function on γ and $r = |x|$ for $x \in \mathbb{R}^2$. When computing numerical solutions of (1) by using the controllability method, we choose the artificial boundary as follows: $\Gamma_a = \{x \in \mathbb{R}^2 \mid |x| = a\}$, where a is a positive number such that $\overline{\mathcal{O}} \subset \{x \in \mathbb{R}^2 \mid |x| < a\}$. Then the bounded computational domain becomes as follows: $\Omega_a = \{x \in \Omega \mid |x| < a\}$. In the controllability method, we solve, in the bounded domain Ω_a, the wave equation with an ABC. We propose a new ABC for the wave equation as follows:

$$\frac{\partial u}{\partial n} + \frac{\partial u}{\partial t} = -\mathcal{S}u - iku \quad \text{on } \Gamma_a, \tag{2}$$

where n is the unit normal vector on Γ_a being toward infinity and $\mathcal{S}$ is the *Dirichlet-to-Neumann* (DtN) operator for the Helmholtz equation with the outgoing radiation

[1] The University of Electro-Communications, koyama@im.uec.ac.jp

condition. Bristeau et al. use the following ABC mainly:

$$\frac{\partial u}{\partial n} + \frac{\partial u}{\partial t} = 0 \quad \text{on } \Gamma_a, \tag{3}$$

and do not mention our ABC (2).

Further we discuss the uniqueness for the solution of the semi-discrete problem of the following problem: find $\boldsymbol{u} = \{u_0, u_1\} \in E_g$ such that

$$\left\{ \begin{array}{rcll} u_{tt} - \Delta u &=& 0 & \text{in } \Omega_a \times (0, T), \\ u &=& g & \text{on } \gamma \times (0, T), \\ \dfrac{\partial u}{\partial n} + \dfrac{\partial u}{\partial t} &=& -\mathcal{S}u - iku & \text{on } \Gamma_a \times (0, T), \\ u(x, 0) &=& u_0(x), \quad u_t(x, 0) = u_1(x) & \text{in } \Omega_a, \\ u(x, T) &=& u_0(x), \quad u_t(x, T) = u_1(x) & \text{in } \Omega_a, \end{array} \right. \tag{4}$$

where $T = 2\pi/k$, $g(x, t) = G(x)e^{-ikt}$, $E_g = V_g \times L^2(\Omega_a)$, and

$$V_g = \left\{ v \in H^1(\Omega_a) \mid v = g(0) \text{ on } \gamma \right\}.$$

Bardos-Rauch [BR94] discuss uniqueness for the solution of the problem (4) in the case when the ABC is replaced by the ABC (3). However, their analysis is not sufficient to prove the uniqueness for the solution of (4), which is yet to be proved.

The DtN operator for the Helmholtz equation

The DtN operator $\mathcal{S}$ can be analytically represented as follows (see Grote-Keller [GK95]):

$$\mathcal{S}U(a, \theta) = \sum_{n=-\infty}^{\infty} -k \frac{H_n^{(1)'}(ka)}{H_n^{(1)}(ka)} U_n(a) Y_n(\theta), \tag{5}$$

where r, θ are the polar coordinates, $H_n^{(1)}$ are the cylindrical Hankel functions of the first kind of order n, Y_n are the spherical harmonics defined by $Y_n(\theta) = e^{in\theta}/\sqrt{2\pi}$, and $U_n(a) = \int_0^{2\pi} U(a, \theta) Y_n(\theta)\, d\theta$.

Uniqueness for the semi-discrete solution

We discretize the problem (4) by finite element method, and show that the obtained semi-discrete problem has a unique solution under hypotheses described below. For this purpose, we choose a finite dimensional subspace W_h of $H^1(\Omega_a)$, and define $V_h = \{v_h \in W_h \mid v_h = 0 \text{ on } \gamma\}$. Let $\phi_1, \phi_2, \ldots, \phi_N$ be a base of V_h, and $\phi_1, \phi_2, \ldots, \phi_N, \phi_{N+1}, \ldots, \phi_{N'}$ a base of W_h. Then we may assume that $\phi_1, \phi_2, \ldots, \phi_{N'}$ are real-valued functions. The semi-discrete problem of the problem (4) can be written as follows: find $\{\boldsymbol{\xi}_0, \boldsymbol{\eta}_0\} \in \mathbb{C}^N \times \mathbb{C}^N$ such that

$$\left\{ \begin{array}{l} B\dfrac{d^2\boldsymbol{\xi}}{dt^2}(t) + C\dfrac{d\boldsymbol{\xi}}{dt}(t) + (A + S + ikC)\,\boldsymbol{\xi}(t) = e^{-ikt}\boldsymbol{f} \quad \text{in } (0, T), \\ \boldsymbol{\xi}(0) = \boldsymbol{\xi}_0, \quad \boldsymbol{\xi}_t(0) = \boldsymbol{\eta}_0, \\ \boldsymbol{\xi}(T) = \boldsymbol{\xi}_0, \quad \boldsymbol{\xi}_t(T) = \boldsymbol{\eta}_0, \end{array} \right. \tag{6}$$

where B, C, A, and S are matrices defined as follows:

$$\begin{aligned}
B &= ((\phi_l,\, \phi_j))_{1\le j,\, l\le N}\,, \quad (u,\, v) = \int_{\Omega_a} u\overline{v}\, dx, \\
C &= (\langle\phi_l,\, \phi_j\rangle)_{1\le j,\, l\le N}\,, \quad \langle u,\, v\rangle = \int_{\Gamma_a} u\overline{v}\, d\gamma, \\
A &= (a(\phi_l,\, \phi_j))_{1\le j,\, l\le N}\,, \quad a(u,\, v) = \int_{\Omega_a} \nabla u\cdot\nabla\overline{v}\, dx, \\
S &= (s(\phi_l,\, \phi_j))_{1\le j,\, l\le N}\,, \quad s(u,\, v) = \int_{\Gamma_a} \mathcal{S}u\overline{v}\, d\gamma,
\end{aligned}$$

and $\boldsymbol{f}$ is a vector defined as follows:

$$\begin{aligned}
\boldsymbol{f} &= (f_j)_{1\le j\le N}\,, \\
f_j &= \sum_{l=N+1}^{N'} \left[k^2(\phi_l,\, \phi_j) - a(\phi_l,\, \phi_j) - s(\phi_l,\, \phi_j)\right] G_l.
\end{aligned}$$

Here the non-homogeneous Dirichlet data G is approximated by the following function: $G_h = \sum_{j=N+1}^{N'} G_j\phi_j|_\gamma$, where $G_j \in \mathbb{C}$ $(j = N+1,\, \ldots\, ,\, N')$.

Now we define a square matrix $\mathcal{A}$ of order $2N$ as follows:

$$\mathcal{A} = \begin{bmatrix} O & I \\ -B^{-1}(A+S+ikC) & -B^{-1}C \end{bmatrix},$$

where I is the unit matrix of order N. To show that the problem (6) has a unique solution, we use the following proposition:

Proposition 1 *The problem* (6) *has a unique solution if and only if*

$$ikl \notin \sigma(\mathcal{A}) \quad \textit{for all } l \in \mathbb{Z}, \tag{7}$$

where $\sigma(\mathcal{A})$ is the set of all eigenvalues of the matrix $\mathcal{A}$.

We show that the problem (6) has a unique solution under two hypotheses described below.

Hypothesis 1 *For a positive λ and $u_h \in V_h$, if we have*

$$a(u_h,\, v_h) = \lambda(u_h,\, v_h) \quad \textit{for all } v_h \in V_h,$$

and if we have $u_h = 0$ on Γ_a, then we have $u_h = 0$ in Ω_a.

Hypothesis 1 can be interpreted as follows. The discrete problems of the two eigenvalue problems:

$$\left\{\begin{aligned}
-\Delta u &= \lambda u && \text{in } \Omega_a, \\
u &= 0 && \text{on } \gamma, \\
\frac{\partial u}{\partial n} &= 0 && \text{on } \Gamma_a
\end{aligned}\right.$$

and

$$\left\{\begin{array}{rcll} -\Delta u & = & \lambda u & \text{in } \Omega_a, \\ u & = & 0 & \text{on } \gamma, \\ u & = & 0 & \text{on } \Gamma_a \end{array}\right.$$

have no same eigenpair.

Hypothesis 2 *For the wave number k, we take the radius a of the artificial boundary such that*

$$\operatorname{Im}\left\{\frac{H_0^{(1)'}(ka)}{H_0^{(1)}(ka)}\right\} < 2.$$

To explain Hypothesis 2, we here state the following lemma:

Lemma 1 $\operatorname{Im}\left\{\dfrac{H_0^{(1)'}(x)}{H_0^{(1)}(x)}\right\}$ *is a decreasing function on $(0, \infty)$, and further*

$$\begin{array}{lcl} \operatorname{Im}\left\{\dfrac{H_0^{(1)'}(x)}{H_0^{(1)}(x)}\right\} & \longrightarrow & 1 \quad (x \longrightarrow +\infty), \\ \operatorname{Im}\left\{\dfrac{H_0^{(1)'}(x)}{H_0^{(1)}(x)}\right\} & \longrightarrow & +\infty \quad (x \longrightarrow +0). \end{array}$$

By Lemma 1, there exists only one $\alpha > 0$ such that

$$\operatorname{Im}\left\{\frac{H_0^{(1)'}(\alpha)}{H_0^{(1)}(\alpha)}\right\} = 2.$$

If the radius a of the artificial boundary is taken to satisfy $a > \alpha/k$, then Hypothesis 2 is satisfied. Here we note that $\alpha \approx 0.088426$.

Theorem 1 *The problem* (6) *has a unique solution under Hypotheses* 1 *and* 2.

To prove Theorem 1, we use the following two lemmas:

Lemma 2 *For all $x > 0$ and for all $\nu \in \mathbb{R}$, we have*

$$\operatorname{Re}\left\{\frac{H_\nu^{(1)'}(x)}{H_\nu^{(1)}(x)}\right\} < 0.$$

Lemma 3 *For all $x > 0$ and for all ν, $\nu' \in \mathbb{R}$ satisfying $|\nu| > |\nu'|$, we have*

$$0 < \operatorname{Im}\left\{\frac{H_\nu^{(1)'}(x)}{H_\nu^{(1)}(x)}\right\} < \operatorname{Im}\left\{\frac{H_{\nu'}^{(1)'}(x)}{H_{\nu'}^{(1)}(x)}\right\}.$$

Proof of Theorem 1: Because of Proposition 1, our task is now to show that (7) holds. The proof is by contradiction. Assume that $ikl \in \sigma(\mathcal{A})$ $(l \in \mathbb{Z})$. Then there is $\boldsymbol{\xi}\ (\neq\ \boldsymbol{o}) \in \mathbb{C}^N$ such that

$$\mathcal{A}\begin{bmatrix} \boldsymbol{\xi} \\ \boldsymbol{\eta} \end{bmatrix} = ikl \begin{bmatrix} \boldsymbol{\xi} \\ \boldsymbol{\eta} \end{bmatrix}.$$

Then we have

$$-(kl)^2 B\boldsymbol{\xi} + iklC\boldsymbol{\xi} + (A + S + ikC)\boldsymbol{\xi} = \boldsymbol{o}. \tag{8}$$

Now we write $\boldsymbol{\xi} = [\xi_1,\, \xi_2,\, \dots\, ,\, \xi_N]^T$ and set $u_h = \sum_{j=1}^{N} \xi_j \phi_j$. Then (8) is written as follows: for all $v_h \in V_h$,

$$-(kl)^2(u_h,\, v_h) + ikl\langle u_h,\, v_h\rangle + a(u_h,\, v_h) + s(u_h,\, v_h) + ik\langle u_h,\, v_h\rangle = 0. \tag{9}$$

Here if we take $v_h = u_h$ in (9), then we obtain

$$-(kl)^2(u_h,\, u_h) + ikl\langle u_h,\, u_h\rangle + a(u_h,\, u_h) + s(u_h,\, u_h) + ik\langle u_h,\, u_h\rangle = 0.$$

The real part of this identity is:

$$a(u_h,\, u_h) - (kl)^2(u_h,\, u_h) - \frac{k}{a}\sum_{n=-\infty}^{\infty} \mathrm{Re}\left\{\frac{H_n^{(1)'}(ka)}{H_n^{(1)}(ka)}\right\}\left|\left\langle u_h,\, \frac{e^{in\theta}}{\sqrt{2\pi}}\right\rangle\right|^2 = 0, \tag{10}$$

and the imaginary part is:

$$\frac{k}{a}\sum_{n=-\infty}^{\infty}\left[l + 1 - \mathrm{Im}\left\{\frac{H_n^{(1)'}(ka)}{H_n^{(1)}(ka)}\right\}\right]\left|\left\langle u_h,\, \frac{e^{in\theta}}{\sqrt{2\pi}}\right\rangle\right|^2 = 0. \tag{11}$$

We here consider three different cases.

Case 1: When $l \le -1$. By Lemma 3,

$$l + 1 - \mathrm{Im}\left\{\frac{H_n^{(1)'}(ka)}{H_n^{(1)}(ka)}\right\} < 0 \quad \text{for all } n \in \mathbb{Z},$$

and hence, by (11),

$$\left\langle u_h,\, \frac{e^{in\theta}}{\sqrt{2\pi}}\right\rangle = 0 \quad \text{for all } n \in \mathbb{Z}.$$

This implies

$$u_h = 0 \quad \text{on } \Gamma_a. \tag{12}$$

From this identity and (9), we get

$$a(u_h,\, v_h) = (kl)^2(u_h,\, v_h) \quad \text{for all } v_h \in V_h. \tag{13}$$

From (12), (13), and Hypothesis 1, we have $u_h = 0$ on Ω_a, i.e., $\boldsymbol{\xi} = \boldsymbol{o}$. This contradicts the assumption that $\boldsymbol{\xi} \neq \boldsymbol{o}$. Therefore we can see $ikl \notin \sigma(\mathcal{A})$.

Case 2: When $l = 0$. By (10), we obtain

$$a(u_h,\, u_h) - \frac{k}{a} \sum_{n=-\infty}^{\infty} \mathrm{Re}\left\{ \frac{H_n^{(1)'}(ka)}{H_n^{(1)}(ka)} \right\} \left| \left\langle u_h,\, \frac{e^{in\theta}}{\sqrt{2\pi}} \right\rangle \right|^2 = 0.$$

From this identity and Lemma 2, it follows that $u_h = 0$ in Ω_a. Therefore $0 \notin \sigma(\mathcal{A})$.

Case 3: When $l \geq 1$. By Lemma 3 and Hypothesis 2, we have

$$l + 1 - \mathrm{Im}\left\{ \frac{H_n^{(1)'}(ka)}{H_n^{(1)}(ka)} \right\} > 2 - \mathrm{Im}\left\{ \frac{H_0^{(1)'}(ka)}{H_0^{(1)}(ka)} \right\} > 0 \quad \text{for all } n \in \mathbb{Z}.$$

By the same argument as Case 1, we can conclude that $ikl \notin \sigma(\mathcal{A})$. ■

Numerical examples

Scattering by a disk

We compare the accuracy of numerical solutions obtained by using our ABC (2) and the ABC (3) via numerical experiments. We consider a test problem, where the obstacle $\mathcal{O} = \{x \in \mathbb{R}^2 \mid |x| < 1\}$, the wave number $k = 1$, and the Dirichlet data G is chosen as the exact solution U becomes as follows: $U(r,\, \theta) = H_1^{(1)}(r)\cos\theta$. We locate the artificial boundary Γ_a at $r = 2$. We use the conforming finite element method using piecewise linear elements. The triangulation has 2176 vertices and 4096 triangles. The length h of each side of every triangle satisfies $\lambda/129 < h < \lambda/54$, where λ is the wave length, i.e., $\lambda = 2\pi/k$. To solve the wave equation numerically, we use explicit second order finite difference centered scheme with the step size $\Delta t = T/200$, where $T = 2\pi/k$. When we use our ABC, we have to truncate the infinite series of (5) at a finite value N. We denote the truncated DtN operator by $\mathcal{S}^N$. In this test problem, we choose $N = 1$, and then we note that $u = U(r,\, \theta)e^{-ikt}$ satisfies

$$\frac{\partial u}{\partial n} + \frac{\partial u}{\partial t} = -\mathcal{S}^1 u - iku \quad \text{on } \Gamma_a.$$

We show contour lines of the real part of the exact solution and the numerical solution obtained by the ABC (3) in Figure 1, where solid lines display the numerical solution, and dotted lines the exact solution. We can see that the numerical solution is very different from the exact solution. We show the exact solution and the numerical solution obtained by our ABC in Figure 2, where the numerical solution is exactly coincident with the exact solution. From these figures we can see that numerical solutions obtained by our ABC are more accurate than those obtained by the ABC (3).

Scattering by a Π-shaped open resonator

We compute scattering of an incident plane wave $\exp[ik(x_1 \cos\alpha + x_2 \sin\alpha)]$ by an obstacle, where α is an incident angle. The wave number $k = 8\pi$ and then the wave

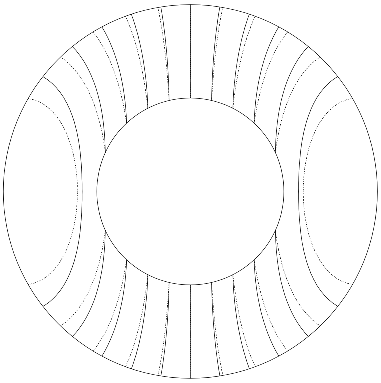


Figure 1: Contour lines of the real part of the exact solution and of the real part of the numerical solution obtained by using the ABC (3).

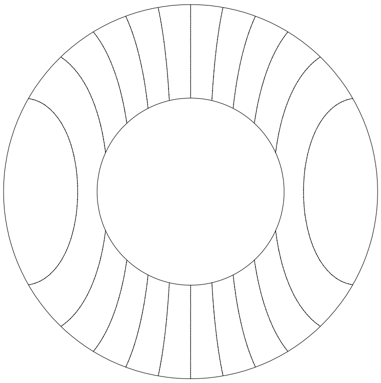


Figure 2: Contour lines of the real part of the exact solution and of the real part of the numerical solution obtained by using our ABC (2).

length $\lambda = 0.25$. The obstacle is a Π-shaped open resonator. The size of its interior rectangle is $4\lambda \times \lambda$, and the thickness of the wall is $\lambda/5$. The incident angle $\alpha = 30°$. First we choose the radius of the artificial boundary $a = 3\lambda$. Then the DtN operator is truncated at $N = 135$, and the triangulation has 42648 vertices and 83808 triangles. The length h of each side of every triangle satisfies $\lambda/51 < h < \lambda/20$. We numerically solve the wave equation by the explicit second order finite difference centered scheme with the step size $\Delta t = T/100$. Next we choose the radius of the artificial boundary $a = 4\lambda$. Then the DtN operator is truncated at $N = 150$, and the triangulation has 77808 vertices and 153888 triangles. The conditions of the mesh size h and the time step size Δt are the same as above. In Figure 3, we display the contour lines of the real part of the numerical solutions in the cases when $a = 3\lambda$ and when $a = 4\lambda$. Figure 3 shows good coincidence of those numerical solutions, and suggests that if we use our ABC, we can get accurate numerical solutions without enlarging radius of the artificial boundary.

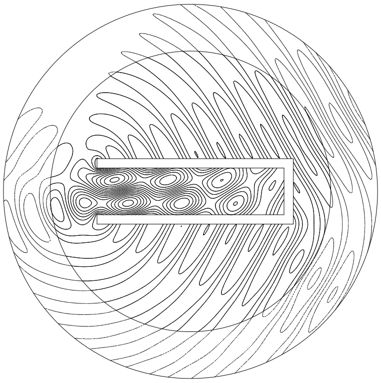


Figure 3: Contour lines of the real part of the numerical solutions in the cases when $a = 3\lambda$ and when $a = 4\lambda$.

Acknowledgments

The author would like to thank Professor Takashi Kako for introducing the controllability method to him, Professor Teruo Ushijima for advising him to prove the uniqueness for the semi-discrete solution, and Mr. Yoshio Tsukuda for giving him a software for triangulation, which was used in computations of the second numerical example.

References

[BGP98]M. O. Bristeau, R. Glowinski, and J. Périaux. Controllability methods for the computation of time-periodic solutions; application to scattering. *J. Comput. Phys.*, 147(2):265–292, 1998.

[BR94]C. Bardos and J. Rauch. Variational algorithms for the Helmholtz equation using time evolution and artificial boundaries. *Asymptotic Analysis*, 9:101–117, 1994.

[GK95]M. J. Grote and J. B. Keller. On nonreflecting boundary conditions. *J. Comput. Phys.*, 122(2):231–243, 1995.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau, ©2001 DDM.org

43. Domain decomposition methods for welding problems

C.-H. Lai[1], C.S. Ierotheou[2], C.J. Palansuriya[3], and K.A. Pericleous[4]

Introduction

The welding of metals and alloys is a widely used industrial process. Many types of analysis have been carried out on such problems [MUB67]. The numerical thermal analysis of welding is required to take into account such features as temperature dependent material properties, phase change, non-uniform distribution of energy from heat source etc. In this paper, a 2-D non-linear electric arc-welding problem is considered. It is assumed that the moving arc generates an unknown quantity of energy which makes the problem an inverse problem with an unknown source. Robust algorithms to solve such problems efficiently, and in certain circumstances in real-time, are of great technological and industrial interest.

There are other types of inverse problems which involve inverse determination of heat conductivity or material properties [CDJ63][TE98], inverse problems in material cutting [ILPP98], and retrieval of parameters containing discontinuities [IK90]. As in the metal cutting problem, the temperature of a very hot surface is required and it relies on the use of thermocouples. Here, the solution scheme requires temperature measurements lied in the neighbourhood of the weld line in order to retrieve the unknown heat source. The size of this neighbourhood is not considered in this paper, but rather a domain decomposition concept is presented and an examination of the accuracy of the retrieved source are presented.

This paper is organised as follows. The inverse problem is formulated and a method for the source retrieval is presented in the second section. The source retrieval method is based on an extension of the 1-D source retrieval method as proposed in [ILP$^+$99] for metal cutting problems. A parallel algorithm based on the concept of coupling heterogeneous numerical models in different subdomains is given in the third section. Accuracy of the numerical simulation is compared with results that are generated by a known heat source [ASW85][DM93] and with temperature measurements that are obtained by using experimental thermocouples as shown in [ASW85].

The inverse welding problem

Three assumptions are needed in this problem. These assumptions are (1) the material properties are homogeneous across the domain of interest, (2) application of a welding tool along a weld path is equivalent to the application of a heat source along

[1] University of Greenwich, C.H.Lai@gre.ac.uk
[2] University of Greenwich, C.Ierotheou@gre.ac.uk
[3] University of Greenwich, C.J.Palansuriya@gre.ac.uk
[4] University of Greenwich, K.A.Pericleous@gre.ac.uk

the path and (3) the rate of change of temperature on either side of the weld is directly proportional to the strength of the heat source [ILP$^+$99]. The welding problem considered in this paper is the welding of two thin metal plates using the technology of arc-welding. For simplicity, the electric arc is moving along the weld path, $y = y_w$ with a speed of U_w. A straight weld line is depicted in 1 as a dotted line. Without loss of generality, the welding line can be a straight line or a general path. If the welding path was a straight line and that the welding tool travelled along $y = y_w = 0$, then due to the symmetry of the problem only the upper half of the domain needs to be considered. This simplifies the model description and programming effort. Since the thickness of the plate, d, is small compared to the other dimensions, only 2-D heat conduction needs to be considered. Hence, using the first two assumptions, the mathematical model which governs the heat conduction of the plate can be written as the following 2-D nonlinear, unsteady, parabolic, heat conduction equation,

$$c_e\frac{\partial T}{\partial t} = \frac{\partial}{\partial x}(k(T)\frac{\partial T}{\partial x}) + \frac{\partial}{\partial y}(k(T)\frac{\partial T}{\partial y}) - 2h_{eff}A(T - T_a) + \delta(y-y_w)Q_w \quad (1)$$

subject to the initial condition $T(x,y,0) = T_i(x,y)$ and boundary conditions defined by the functionals $B_0[T(0,y,t),0,y,t] = 0$, $B_1[T(l,y,t),l,y,t] = 0$, $C_0[T(x,-h,t),x,-h,t] = 0$ and $C_1[T(x,h,t),x,h,t] = 0$. Here $T(x,y,t)$ is the temperature distribution, $k(T)$ is the conductivity of the metal plates, t is the time, h_{eff} is the effective heat transfer, A is the surface area, T_a is the ambient temperature, $c_e = \rho c - L\frac{\partial f_l}{\partial T}$ is the effective specific heat, ρ is the density, c is the specific heat capacity, L is the latent heat, $\frac{\partial f_l}{\partial T}$ is the variation of liquid fraction, $\delta(y - y_w)$ is the Dirac Delta function, $Q_w = Q_w(x,t)$ is the heat transfer rate generated from the moving arc. T_i, B_0, B_1, C_0 and C_1 are known functions. The source term, Q_w, in (1) is an unknown, and the inverse problem here is to retrieve this unknown heat source.

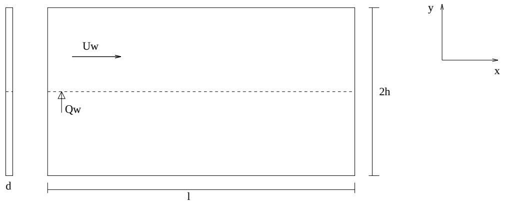


Figure 1: A simple welded work-piece.

In order to deal with this additional unknown, temperature measurements near the weld line is required (see Figure 2). Thermocouples are attached at $y = y_s$, such that $y_w < y_s < h$. Let the temperature measured by means of the thermocouples be $T(x,y_s,t) = T^*(x,t)$. The measured temperatures are used as interior boundary conditions, as described in next Section, along subdomain interfaces and to retrieve the temperature distribution at the welding points. The heat source retrieval is based on the third assumption, i.e. in the neighbourhood of the weld,

$$\frac{\partial T}{\partial t} = \alpha(x,t)\delta(y-y_w)Q_w(x,t) \quad (2)$$

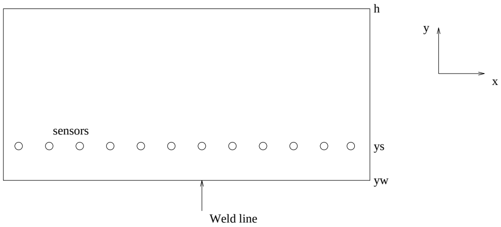


Figure 2: Thermocouples are located near the weld line.

where $\alpha > 1$ is a time dependent function that also depends on x. The condition $\alpha > 1$ is to ensure an increase in temperature at the weld due to an increase in Q_w. Integrating (2) across the weld at a given value of x gives

$$\int_{y_w^-}^{y_w^+} \frac{\partial T}{\partial t} dy \;=\; \alpha(x,t) Q_w(x,t) \tag{3}$$

where y_w^+ to y_w^- is the width of the weld along y-axis at a given instance of time under immediate influence of the electric arc. Integrating (1) across the weld and equating the result to (3) lead to

$$\frac{1}{c_e}\{k(T)\frac{\partial T}{\partial y}|_{y_w^+} \;-\; k(T)\frac{\partial T}{\partial y}|_{y_w^-} \;+\; \frac{\partial}{\partial x}(k(T)\frac{\partial T}{\partial x})(y_w^+ - y_w^-) \;-\; 2h_{eff}A(T-T_a)(y_w^+ - y_w^-)\}$$

$$= \;(\alpha(x,t) - \frac{1}{c_e})Q_w(x,t) \tag{4}$$

Let $\beta(x,t) = c_e\alpha(x,t) - 1$ and define the predicted heat source as $Q_p = \beta(x,t)Q_w(x,t)$ which may be computed as

$$Q_p(x,t) = k(T)\frac{\partial T}{\partial y}|_{y_w^+} \;-\; k(T)\frac{\partial T}{\partial y}|_{y_w^-} + \frac{\partial}{\partial x}(k(T)\frac{\partial T}{\partial x})(y_w^+ - y_w^-)$$

$$-\; 2h_{eff}A(T - T_a)(y_w^+ - y_w^-) \tag{5}$$

Then Q_p can be substituted into (1) to replace Q_w and the non-linear heat conduction problem may then be solved as a direct problem with $T_p(x,t)$ being the corresponding temperature distribution. Hence it is possible to evaluate $\beta(x,t)$ from the knowledge of $T_p(x,t)$ and $T(x,y_w,t)$ as

$$\beta(x,t) \;=\; \frac{T_p(x,t)}{T(x,y_w,t)} \;=\; \frac{Q_p(x,t)}{Q_w(x,t)} \tag{6}$$

where $T(x,y_w,t)$ is the temperature at the weld line corresponds to $Q_w(x,t)$. Hence $Q_w(x,t)$ may then be determined once $\beta(x,t)$ is known. Note that it is not necessary to compute $c_e\alpha(x,t) - 1$.

The domain decomposition method

Since the only unkown involved in the p.d.e. is the heat source, it makes sense to eliminate the unknown source term of the p.d.e. [ILPP98] for the governing equations on both sides of the welding path. The monitored thermocouple data provides an ideal interior partitioning. For the present study, y_w is chosen as zero. Hence the problem become symmetric and only half of the entire problem needs to be considered. The original domain is partitioned to two well defined, homogeneous, continuous and properly connected subdomains denoted by $D_1 = \{(x, y) : 0 < x < l \text{ and } 0 < y < y_s\}$ and $D_2 = \{(x, y) : 0 < x < l \text{ and } y_s < y < h\}$ and they are depicted as in Figure 3. The two subproblems can be written as follows:

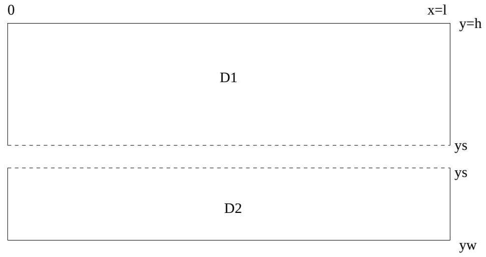


Figure 3: Visualization of subdomains.

$$SP_1:\quad c_e\tfrac{\partial T_1}{\partial t} = \tfrac{\partial}{\partial x}(k(T_1)\tfrac{\partial T_1}{\partial x}) + \tfrac{\partial}{\partial y}(k(T_1)\tfrac{\partial T_1}{\partial y}) - 2h_{eff}A(T_1 - T_a)\ inD_1$$
subject to $T_1(x, y, 0) = T_i(x, y)$,
$B_0[T_1(0, y, t)0, y, t] = 0$, $B_1[T_1(l, y, t), l, y, t] = 0$,
$T_1(x, y_s, t) = T^*(x, t)$, $C_1[T_1(x, h, t), x, h, t] = 0$.

$$SP_2:\quad c^e\tfrac{\partial T_2}{\partial t} = \tfrac{\partial}{\partial x}(k(T_2)\tfrac{\partial T_2}{\partial x}) + \tfrac{\partial}{\partial y}(k(T_2)\tfrac{\partial T_2}{\partial y}) - 2h_{eff}A(T_2 - T_a)\ inD_2$$
subject to $T_2(x, y, 0) = T_i(x, y)$,
$B_0[T_2(0, y, t), 0, y, t] = 0$, $B_1[T_1(l, y, t), l, y, t] = 0$,
$T_2(x, y_s, t) = T^*(x, t)$, $\frac{\partial T_2(x,0,t)}{\partial y} = 0$.

and are defined on two different subdomains of different sizes which are subjected to different set of boundary conditions. They are non-linear in nature and are completely decoupled from each other. Therefore they may be solved simultaneously or concurrently by using the Newton's method. Let $F(T)$ be defined as

$$F(T) = c_e\frac{\partial T}{\partial t} - \frac{\partial}{\partial x}(k(T)\frac{\partial T}{\partial x}) - \frac{\partial}{\partial y}(k(T)\frac{\partial T}{\partial y}) + 2h_{eff}A(T - T_a) \equiv 0 \qquad (7)$$

which leads to the corresponding Jacobian $J(T)$ as follows,

$$J(T) = c_e\frac{\partial}{\partial t} - k'\frac{\partial^2 T}{\partial x^2} - k\frac{\partial^2}{\partial x^2} - k''(\frac{\partial T}{\partial x})^2 - 2k'\frac{\partial T}{\partial x}\frac{\partial}{\partial x} - k'\frac{\partial^2 T}{\partial y^2}$$

$$-k\frac{\partial^2}{\partial y^2} - k''(\frac{\partial T}{\partial y})^2 - 2k'\frac{\partial T}{\partial y}\frac{\partial}{\partial y} + 2h_{eff}A \tag{8}$$

The linearisation leads to an iterative scheme, to be performed in each of the subdomain, $T^{new} = T^{old} - J^{-1}(T^{old})F(T^{old})$ where superscript *new* denotes new iterates and *old* denotes old iterates. $F(T)$ and $J(T)$ are obtained by a second order finite volume method which leads to a set of large sparse linear system and it can be solved by means of a standard domain decomposition software package such as PETSc [BGMS97]. More processors may be used to achieve a secondary level of parallelism for the Newton's iterative scheme, which are separately controlled by different hosts assigned to each of the subproblems. Therefore the inverse welding problem has two levels of parallelism. One level being the differential equation level and the other being the discretised level [ILPP98]. The solution of SP_2 retrieves the temperature at the weld line. The temperature at and around the weld line is, in turn, used to retrieve the unknown source term as described above.

Numerical examples

In this Section, a validation problem for comparison purposes is defined. The true source as given in [ASW85, DM93] is depicted in Figure 9 and the physical data for (1), as given below, are used to derive validation data for comparison. Geometry of the two plates is chosen as $l = 0.5m$, $2h = 0.33m$, $d = 0.008m$, $U_w = 0.00333m/s$. The model problem gives the temperature field of the steel plate. The physical data used in the numerical example are $Q_w = 1350\ W$, $T_a = 293\ K$, $h_{eff}\ =\ 60\ W/m^2K$, $\rho\ =\ 7850\,kg/m^3, c\ =\ 607\,J/kgK, L\ =\ 272\,kJ/kg, T_s\ =\ 1843\,K$ and $T_l\ =\ 1863\,K$. Here T_s is the solidus temperature and T_l is the liquidus temperature. For the present purpose, the liquid fraction f_l is evaluated as,

$$f_l = \begin{cases} 0 & \text{if } T < T_s \\ \frac{(T-T_s)}{(T_l-T_s)} & \text{if } T_s \le T \le T_l \\ 1 & \text{if } T > T_l \end{cases}$$

The nonlinear conductivity of steel is given by,

$$k(T) = \begin{cases} \frac{-27.2}{762}T\ +\ 64.9448 & \text{if } T \le 1035K \\ \frac{8}{881}T\ +\ 18.6016 & \text{if } T > 1035K \end{cases}$$

The initial condition is $T_i(x,y) = T_a$ and the boundary conditions are $B_0 = B_1 = k\frac{\partial T}{\partial x}\ +\ h_{eff}(T - T_a) = 0$ and $C_1 = k\frac{\partial T}{\partial y}\ +\ h_{eff}(T - T_a) = 0$. The source is applied only at cells which are at a given instant of time under immediate influence of the electric arc. A mesh size of 50 × 50 is used to obtain the following numerical results. Figure 4 shows the 2-D temperature distribution at $t = 75s$. At this time the arc passes the midsection of the plate ($x = 0.25m$,$y = y_w = 0m$). Therefore, the temperature is at its highest value at this section. Thermocouple temperature measurements are available for comparison from MPA, Stuttgart [ASW85]. Figures 5 to 7 show the comparison of numerical results with the thermocouple measurements. Figure 5 compares the numerical results with measured results when the arc passes the midsection of the plate. Figure 6 shows the comparison at a further 7.5s later, as expected cooling has begun (since the arc has moved away from the midsection). Figure 7 shows the temperature history at the midsection, it illustrates the rapid heating to the melting point when the arc approaches the midsection and the gradual cooling thereafter when the arc has passed the section. These results show that the derived data matches with the experimental data. Thermocouples are now placed at $y_s = 0.0033m$ as suggested in[ASW85]. The inverse problem given by (1) is solved by using the mesh configuration of 200 × 200. Figure 8 shows the accuracy of the retrieved temperature field at $x = 0.25m$ and $t = 75s$. At this time step, the electric arc passes over the point $x = 0.25m$ and $y = y_w = 0m$ (midsection), and as expected it generates high temperature values (and gradients) around this point. Figure 9 shows the accuracy of the retrieved source term at $x = 0.25m$ and $y = y_w = 0m$ using the

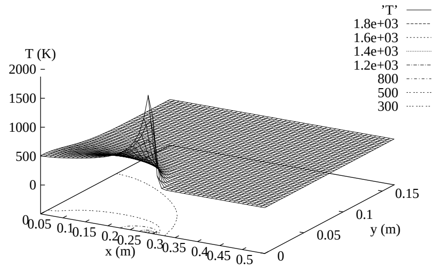


Figure 4: Temperature distribution at t = 75s.

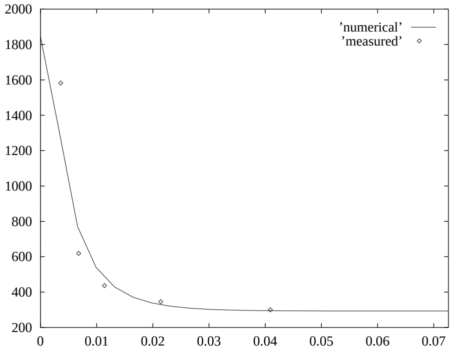


Figure 5: Temperature distribution at x=0.25m and t = 75s (Vertical axis - T in Kelvin and horizontal axis - y coordinates).

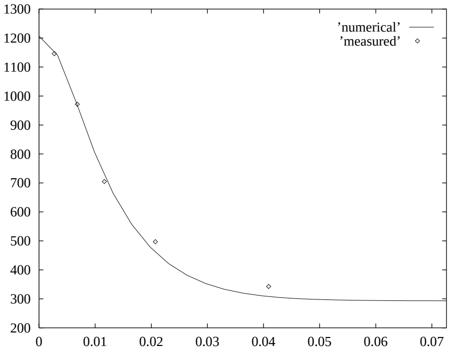


Figure 6: Temperature distribution at x=0.25m and t = 82.5s (Vertical axis - T in Kelvin and horizontal axis - y coordinates).

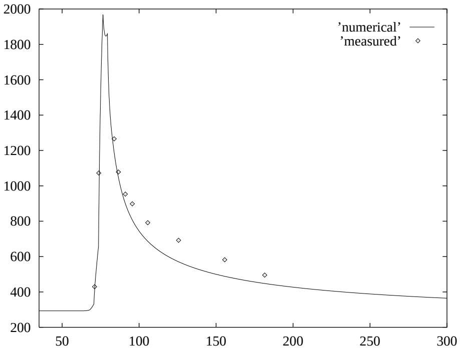


Figure 7: Temperature history at x=0.25m and y=0m (Vertical axis - T in Kelvin and horizontal axis - time in seconds).

proposed method as shown above. The source retrieval is only activated when the electric arc actually passes over this point.

Conclusion

A domain decomposition method is proposed for an inverse problem in arc-welding. The method is based on the partitioning of problems at the continuous problem level where the unknown heat source can be eliminated from the mathematical model and where the subproblems may be completely decoupled. The retrieved heat source compares well with the results generated by using a known heat source of a typical arc-welding and by using experiments.

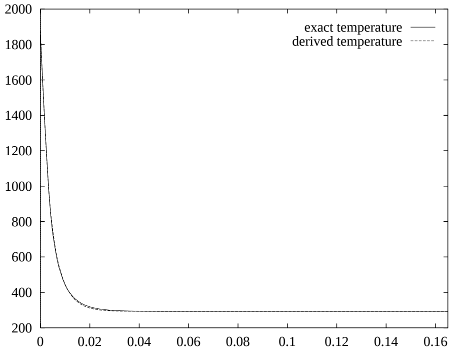


Figure 8: Accuracy of the temperature distribution at x=0.25m and t=75s (Vertical axis - T in Kelvin and horizontal axis - time in seconds).

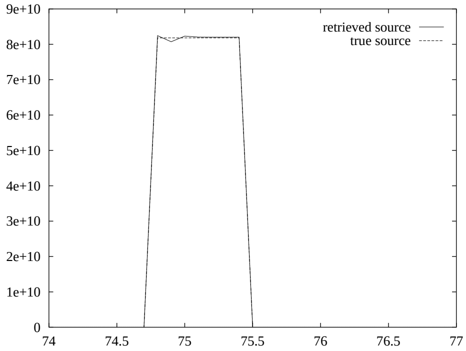


Figure 9: Accuracy of the source retrieval (Vertical axis - source strength and horizontal axis - y coordinates).

References

[ASW85]J.H. Argyris, J. Szimmat, and K.J. William. Finite element analysis of arc-welding processes. In R.W. Lewis and K. Morgan, editors, *Numerical Methods in Heat Transferr Vol 3*, pages 1–34. John Wiley and Sons, 1985.

[BGMS97]S. Balay, W. Gropp, L.C. McInnes, and B. Smith. *PETSc 2.0 User Manual.* Argonne National Laboratory, http://www.mcs.anl.gov/petsc/, 1997.

[CDJ63]J.R. Cannon, J. Douglas, and B.F. Jones. Determination of the diffusivity in an anisotropic medium. *Int. J. Eng. Sci.*, 1, 1963.

[DM93]I. Demirdzic and D. Martinovic. Finite volume method for thermo-elasto-plastic stress analysis. *Computer Methods in Apllied Mechanics and Engineering*, 109:331–349, 1993.

[IK90]K. Ito and K. Kunisch. The augmented lagrangian method for parameter estimation in elliptic systems. *SIAM J. Control Optim.*, 28:113–136, 1990.

[ILP$^+$99]C.S. Iertotheou, C.-H. Lai, C.J. Palansuriya, K.A. Pericleous, M.S. Espedal, and X.-C. Tai. Accuracy of a domain decomposition method for the recovering of discontinuous heat sources in metal sheet cutting. *Computing and Visualisation in Science*, 2:149–152, 1999.

[ILPP98]C.S. Ierotheou, C.-H. Lai, C.J. Palansuriya, and K.A. Pericleous. Simulation of 2-d metal cutting by means of a distributed algorithm. *The Computer Journal*, 41:57–63, 1998.

[MUB67]P.S Myers, O.A. Uyehara, and G.L. Borman. Fundamentals of heat flow in welding. *Welding Research Council Bulletin*, 123, 1967.

[TE98]X.-C. Tai and M. S. Espedal. Rate of convergence of some space decomposition method for linear and non-linear elliptic problems. *SIAM J. Numer. Anal.*, 35:1558–1570, 1998.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau,

44. FETI-DP: An Efficient, Scalable and Unified Dual-Primal FETI Method

M. Lesoinne[1], K. Pierson[2]

Introduction

The FETI algorithms are numerically scalable iterative domain decomposition methods. These methods are well documented for solving equations arising from the Finite Element discretization of second or fourth order elasticity problems. The one level FETI method equipped with the Dirichlet preconditioner was shown to be numerically scalable for second order elasticity problems while the two level FETI method was designed to be numerically scalable for fourth order elasticity problems (see [FR94, Far91b, Far91a, FR91, FR92, FM98, Rou95]).

The second level coarse grid is an enriched version of the original one level FETI method with coarse grid. The coarse problem is enriched by enforcing transverse displacements to be continuous at the corner points. This coarse problem grows linearly with the number of subdomains. Current implementations use a direct solution method to solve this coarse problem. However, the current implementation gives rise to a full matrix system. This full matrix can lead to increased storage requirements especially if working within a distributed memory environment. Also, the factorization and subsequent forward/backward substitutions of the second level coarse problem becomes the dominant factor in solving the global problem as the number of subdomains becomes large ($N_s > 1000$).

We introduce an alternative formulation of the two level coarse problem that leads to a sparse system better suited for a direct method. Then we show extensions to the alternate formulation that allow optional admissible constraints to be added to improve convergence. Lastly, we report on the numerical performance, parallel efficiency, memory requirements, and overall CPU time as compared to the classical two level FETI on some large scale fourth order elasticity problems.

The Dual-Primal FETI Method

Let Ω be partitioned into a set of N_s, non-overlapping subdomains (or substructures) Ω^s. Points where 3 or more subdomains intersect, are labeled as corner points which will remain primal variables. The mechanical interpretation of this particular method of mesh splitting can be viewed as making incisions into the mesh but leaving the corner points attached. This is analogous to the "tearing" stage of FETI. The "interconnecting" stage occurs only on the subdomain interfaces which now excludes the corner points. Typically, in fourth order elasticity problems, the corner points have 6 degrees of freedom (3 translations and 3 rotations). This method of mesh splitting

[1] Professor, Department of Aerospace Engineering and Sciences and Center for Aerospace Structures University of Colorado at Boulder Boulder, CO 80309-0429, U.S.A.

[2] Senior Member Technical Staff, Sandia National Labs, Albuquerque, NM 87111, U.S.A.

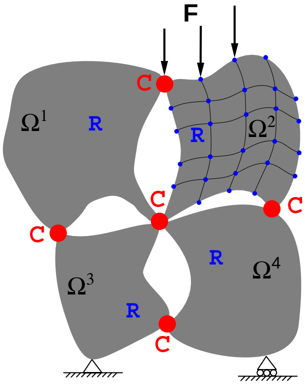


Figure 1: Dual-Primal Mesh Partitions

and corner point identification is illustrated in Figure 1: By splitting, u^s into two sub-vectors such that:

$$u = \begin{bmatrix} u_r \\ u_c \end{bmatrix} = \begin{bmatrix} u_r^1 \\ \vdots \\ u_r^{N_s} \\ u_c \end{bmatrix} \tag{1}$$

where u_r^s is the remaining subdomain solution vector and u_c is a global/primal solution vector over all defined corner degrees of freedom. The solution at the corner points is continuous by definition when the solution vector is constructed in this manner. Using this notation, we can split the subdomain stiffness matrix into:

$$K^s = \begin{bmatrix} K_{rr}^s & K_{rc}^s \\ K_{rc}^{s^T} & K_{cc}^s \end{bmatrix} \tag{2}$$

Then the original FETI equilibrium equations can be modified using the following matrix partitioning where the subscripts c and r denote the corner and the remainder degrees of freedom.

$$\begin{bmatrix} K_{rr}^1 & \dots & 0 & K_{rc}^1 B_c^1 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & K_{rr}^{N_s} & K_{rc}^{N_s} B_c^{N_s} \\ B_c^{1^T} K_{rc}^{1^T} & \dots & B_c^{N_s{}^T} K_{rc}^{N_s{}^T} & \sum_{s=1}^{N_s} B_c^{s^T} K_{cc}^s B_c^s \end{bmatrix} \begin{bmatrix} u_r^1 \\ \vdots \\ u_r^{N_s} \\ u_c \end{bmatrix} = \begin{bmatrix} f_r^1 - B_r^{1^T} \lambda \\ \vdots \\ f_r^{N_s} - B_r^{N_s{}^T} \lambda \\ \sum_{s=1}^{N_s} B_c^{s^T} f_c^s \end{bmatrix} \tag{3}$$

$$\sum_{s=1}^{N_s} B_r^s u_r^s = 0 \tag{4}$$

Where the corner stiffness matrix, $K_{cc} = \sum_{s=1}^{N_s} B_c^{s^T} K_{cc}^s B_c^s$ is a global stiffness quantity, B_c^s maps the local corner equation numbering to global corner equation numbering, f_r^s is the external force applied on the r degrees of freedom, $B_r^{s^T}$ is a boolean matrix that extracts the interface of a subdomain, and λ are the Lagrange multipliers. Let K_{rr} denote the block diagonal subdomain stiffness matrix restricted to the remaining, r, points, K_{rc} the block column vector of subdomain coupling stiffness matrices, f_r the block column vector of subdomain force vectors, K_{cc} the global corner stiffness matrix and using the "rc" notation, we can rewrite the equilibrium equations in the more compact form:

$$\begin{bmatrix} K_{rr} & K_{rc} \\ K_{rc}^T & K_{cc} \end{bmatrix} \begin{bmatrix} u_r \\ u_c \end{bmatrix} = \begin{bmatrix} f_r - B_r^T \lambda \\ f_c \end{bmatrix} \tag{5}$$

Now we can invert the first equation for u_r noting that K_{rr} is a symmetric positive definite matrix due to the guarantee of enough corner points that remove all singularities. This is in contrast to all the previous FETI methods where the correct computation of the the null spaces was required to be accurately computed, leading to a natural coarse problem. This null space computation was seen as a liability when working with nonlinear structures where the size of the null space would vary from one tangent stiffness matrix to the next ([FPL00]). Then substitute the result into the compatibility equation (Eq. 4). With some algebraic manipulation we can derive the Dual-Primal FETI interface problem where the unknowns are λ, the Lagrange multipliers and u_c, the global corner degrees of freedom.

$$\begin{bmatrix} F_{rr} & F_{rc} \\ F_{rc}^T & -K_{cc}^* \end{bmatrix} \begin{bmatrix} \lambda \\ u_c \end{bmatrix} = \begin{bmatrix} d_r \\ -f_c^* \end{bmatrix} \tag{6}$$

where $F_{rr} = \sum_{s=1}^{N_s} B_r^s K_{rr}^{s^{-1}} B_r^{s^T}$, $F_{rc} = \sum_{s=1}^{N_s} B_r^s K_{rr}^{s^{-1}} K_{rc}^s B_c^s$, $d_r = \sum_{s=1}^{N_s} B_r^s K_{rr}^{s^{-1}} f_r^s$, and $f_c^* = \sum_{s=1}^{N_s} (B_c^{s^T} f_c^s - B_c^{s^T} K_{rc}^{s^T} K_{rr}^{s^{-1}} f_r^s)$. The corner degrees of freedom, u_c, are condensed out to form the following symmetric positive definite Dual-Primal FETI interface problem which we solve using a preconditioned conjugate gradient method. For a detailed derivation of this equation, please see [CFR00].

$$\left[F_{rr} + F_{rc} K_{cc}^{*^{-1}} F_{rc}^T \right] \lambda = d_r - F_{rc} K_{cc}^{*^{-1}} f_c^* \tag{7}$$

It can be seen that the new FETI operator has a coarse grid problem for which the stiffness matrix can be written as follows

$$K_{cc}^* = \sum_{s=1}^{N_s} \left[B_c^{s^T} K_{cc}^s B_c^s - (K_{rc}^s B_c^s)^T K_{rr}^{s^{-1}} (K_{rc}^s B_c^s) \right] \tag{8}$$

This new coarse problem has some highly beneficial properties over the existing two-level FETI coarse problem (see [FM98]). First, this new coarse problem is sparse

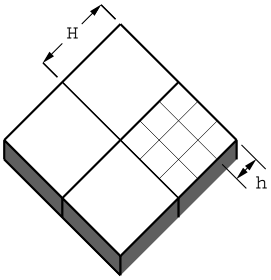


Figure 2: Model problem for numerical scalability studies

symmetric positive definite. Secondly, only one forward/backward substitution has to be performed per FETI iteration in comparison with two per iteration of the original FETI algorithms. We also note that the coarse problem is easily formed in parallel with subdomain operations. As with the original FETI coarse problem, it couples all of the subdomains and propagates the error at each FETI iteration.

Numerical Scalability

Now we would like to test the numerical scalability of the Dual-Primal FETI method for fourth order elasticity problems. The chosen tests show the numerical scalability with respect to the number of subdomains, size of the subdomains, and the size of the elements. The model problem for these tests is a 1×1 square mesh discretized into 3 node shell elements. Let h denote the size of an individual element and H denote the size of one subdomain. The first numerical test keeps the number of subdomains at 64, while varying the size of h and the effect on the number of iterations to converge to $1.0E-6$ is observed.

H	h	N_{dof}	FETI-2	FETI-DP
1/8	1/40	5,166	23 itr.	17 itr.
1/8	1/80	19,926	30 itr.	22 itr.
1/8	1/160	78,246	36 itr.	28 itr.
1/8	1/320	310,086	44 itr.	34 itr.
1/8	1/640	1,234,566	51 itr.	41 itr.

One can see that the number of iterations remains roughly constant for both the two level FETI method and for the Dual-Primal method as the size of the problem is increased from $5,166$ dof to over 1 million dof.

The second numerical test fixes the size of the problem and varies the number of subdomains used to solve the problem. Again the number of iterations remains approximately constant over a large range of N_s for both the two level FETI method and the Dual-Primal FETI method..

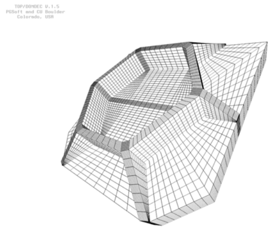


Figure 3: Finite element model of a diffraction grating

H	h	N_s	FETI-2	FETI-DP
1/8	1/640	64	51 itr.	17 itr.
1/10	1/640	100	47 itr.	22 itr.
1/16	1/640	256	47 itr.	28 itr.
1/20	1/640	400	47 itr.	34 itr.
1/40	1/640	1,600	40 itr.	41 itr.
1/64	1/640	4,096	36 itr.	28 itr.

The last numerical test holds the size of the subdomains constant while increasing the size of the overall problem. In this test, the condition number of the two level FETI method should remain roughly the same (see [FM98]). We see that both methods exhibit this trend for a large range of N_s.

H	h	N_s	FETI-2	FETI-DP
1/2	1/20	4	12 itr.	12 itr.
1/4	1/40	16	24 itr.	19 itr.
1/8	1/80	64	30 itr.	22 itr.
1/16	1/160	256	32 itr.	24 itr.
1/32	1/320	1,024	34 itr.	25 itr.
1/64	1/640	4,096	36 itr.	28 itr.

The Augmented Dual-Primal FETI Method

After testing FETI-DP on a range of fourth order problems, we decided to test a second order elasticity problem. The motivation was to see if we could improve the existing one level FETI technology. As the reader can see, the following results were not encouraging for this diffraction grating problem with 120,987 degrees of freedom.

N_s	FETI-1	FETI-DP	Augmented FETI-DP
56	81 itr. (281 sec.)	190 itr. (534 sec.)	63 itr. (284 sec.)
128	51 itr. (115 sec.)	115 itr. (273 sec.)	38 itr. (129 sec.)

The initial thought was to investigate how the new Dual-Primal coarse problem could be extended to improve convergence. This can be accomplished by forcing the residual to be orthogonal to a chosen set of vectors at each iteration of the FETI algorithm.

Let Q be a matrix of arbitrarily chosen vectors, r the residual, then we can enforce the following equation to enhance convergence:

$$Q^T r = Q^T \sum_{s=1}^{N_s} B_r^s u_r^s = Q_r^T u_r = 0 \tag{9}$$

We insert these equations within the formulation by introducing new Lagrange multipliers, μ, to enforce the constraints associated with Eq. 9.

$$\begin{bmatrix} K_{rr} & K_{rc} & Q_r \\ K_{rc}^T & K_{cc} & 0 \\ Q_r^T & 0 & 0 \end{bmatrix} \begin{bmatrix} u_r \\ u_c \\ \mu \end{bmatrix} = \begin{bmatrix} f_r - B_r^T \lambda \\ f_c \\ 0 \end{bmatrix} \tag{10}$$

The resulting FETI operator has the same form as given in 7. Following the same procedure used to derive Eq. 6, we arrive at the following expression for the augmented Dual-Primal FETI coarse grid which is non-singular for a well-posed non-floating structure but because of the μ Lagrange multiplier, we have negative eigen values.

$$\tilde{K}_{cc}^* = \sum_{s=1}^{N_s} \begin{bmatrix} B_c^{s^T} K_{cc}^s B_c^s - B_c^{s^T} K_{rc}^{s^T} K_{rr}^{s^{-1}} K_{rc}^s B_c^s & -B_c^{s^T} K_{rc}^{s^T} K_{rr}^{s^{-1}} Q_r^s \\ -Q_r^{s^T} K_{rr}^{s^{-1}} K_{rc}^s B_c^s & -Q_r^{s^T} K_{rr}^{s^{-1}} Q_r^s \end{bmatrix} \tag{11}$$

These Q matrices can be chosen to be the average x,y, or z jump along a subdomain edge resulting in an edge by edge sparsity pattern for the augmented set of equations.. There has been a clear advantage to writing the equations on a per edge basis as it has improved convergence dramatically, improved CPU times, and restored numerical scalability with respect to second order elasticity problems.

For higher order elements, such as 10 node tetrahedron, FETI-DP has shown to be much more efficient than the one level FETI method. The following results were obtained from a large-scale structural solid model discretized using 10 node tetrahedrons of a BMW engine. The entire engine model has over 1 million degrees of freedom which was decomposed into 823 subdomains and computed on an Origin 2000 machine. It took the one level FETI method 243 iterations to converge while it took 90 iterations for FETI-DP.

N_p	FETI-1	Augmented FETI-DP
3	1,476 sec.	604 sec.
6	773 sec.	334 sec.
12	461 sec.	247 sec.
24	207 sec.	140 sec.

Parallel Scalability

We conclude this paper with a large-scale example problem that highlights the advantages of FETI-DP. The following problem is a shell model of a wheel rim composed of over 313856 elements, 156017 nodes, and containing $936,102$ degrees of freedom. Three points were fixed along the inner rim, effectively constraining the model. Then a gravity load was applied to the model which was decomposed into 500 subdomains.

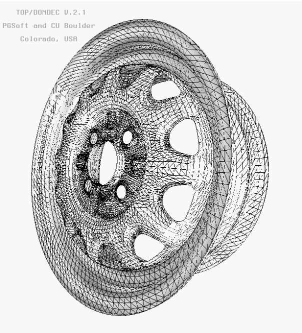


Figure 4: Finite element model of wheel rim

As one can see, the reduction in CPU time is dramatic for the FETI-DP method. The PSLDLT parallel sparse solver shows a large improvement over the two level FETI method for low numbers of processors while the FETI-DP method is faster for $Np = 1$ all the way to $Np = 24$. The speed-up numbers for the two level FETI method and the FETI-DP method are nearly identical for these runs on an Origin 2000.

N_p	FETI-2	PSLDLT	FETI-DP
1	2,995 s (1.0)	1,631 s (1.0)	1,594 s (1.0)
4	789 s (3.8)	502 s (3.2)	370 s (4.3)
8	371 s (8.1)	301 s (5.4)	196 s (8.1)
16	214 s (13.9)	218 s (7.5)	116 s (13.7)
20	179 s (16.7)	200 s (8.2)	99 s (16.1)
24	157 s (19.0)	200 s (8.2)	86 s (18.5)

Conclusion

We have shown a modification to the classical FETI method where the local operators are symmetric positive definite. This eliminates the necessity for computing the local null spaces. which also removes the original FETI coarse problem. The new Dual-Primal FETI method has a global coarse problem associated with the global corner displacements. This coarse grid was shown to have as good as or better than convergence for fourth order plates and shells problems with respect to the two level FETI method. For second order problems, the new Dual-Primal FETI coarse grid has to be augmented with optional constraints to remain numerically scalable. The Dual-Primal FETI method is more robust, more efficient and typically faster than the classical FETI methods for large numbers of subdomains.

References

[CFR00]P. LeTallec K. Pierson C. Farhat, M. Lesoinne and D. Rixen. Feti-dp: A dual-primal unified feti method - part i: A faster alternative to the two-level feti method. *International Journal for Numerical Methods in Engineering*, 2000. in press.

[Far91a]C. Farhat. A saddle-point principle domain decomposition method for the solution of solid mechanics problem. In D.E. Keyes, T.F. Chan, G.A. Meurant, J.S. Scroggs, and R.G. Voigt, editors, *Proc. Fifth SIAM Conference on Domain Decomposition Methods for Partial Differential Equations*, pages 271–292. SIAM, 1991.

[Far91b]Charbel Farhat. A Lagrange multiplier based on divide and conquer finite element algorithm. *J. Comput. System Engng*, 2:149–156, 1991.

[FM98]C. Farhat and J. Mandel. The two-level feti method for static and dynamic plate problems - part i: an optimal iterative solver for biharmonic systems. *Computer Methods in Applied Mechanics and Engineering*, 155:129–152, 1998.

[FPL00]C. Farhat, K. H. Pierson, and M. Lesoinne. The second generation of feti methods and their application to the parallel solution of large-scale linear and geometrically nonlinear structural analysis problems. *Computer Methods in Applied Mechanics and Engineering*, 184:333–374, 2000.

[FR91]Charbel Farhat and Francois-Xavier Roux. A Method of Finite Element Tearing and Interconnecting and its Parallel Solution Algorithm. *Int. J. Numer. Meth. Engng.*, 32:1205–1227, 1991.

[FR92]C. Farhat and F.X. Roux. An unconventional domain decomposition method for an efficient parallel solution of large-scale finite element systems. *SIAM J. Sc. Stat. Comput.*, 13:379–396, 1992.

[FR94]Charbel Farhat and François-Xavier Roux. Implicit parallel processing in structural mechanics. In J. Tinsley Oden, editor, *Computational Mechanics Advances*, volume 2 (1), pages 1–124. North-Holland, 1994.

[Rou95]F. X. Roux. Parallel implementation of a domain decomposition method for nonlinear elasticity problems. In *Domain-Based Parallelism and Problem Decomposition Methods in Computational Science and Engineering*, pages 161–175. SIAM, 1995.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau, ©2001 DDM.org

45. A blackbox reduced-basis output bound method for shape optimization

L. Machiels[1], Y. Maday[2], A. T. Patera[3], D. V. Rovas[4]

Introduction

We present a two-stage off-line/on-line blackbox reduced-basis output bound method for the prediction of outputs of coercive partial differential equations with affine parameter dependence. The computational complexity of the on-line stage of the procedure scales only with the dimension of the reduced-basis space and the parametric complexity of the partial differential operator. The method is both efficient and certain: thanks to rigorous *a posteriori* error bounds, we may retain only the minimal number of modes necessary to achieve the prescribed accuracy in the output of interest. The technique is particularly appropriate for applications such as design and optimization, in which repeated and rapid evaluation of the output is required.

Reduced-basis methods [ASB78, Nag79, NP80] — projection onto low-order approximation spaces comprising solutions of the problem of interest at selected points in the parameter/design space — are efficient techniques for the prediction of linear functional outputs. These methods enjoy an optimality property which ensures rapid convergence even in high-dimensional parameter spaces; good accuracy is obtained even for very few modes (basis functions), and thus the computational cost is typically very small.

It is often the case that the parameter enters affinely in the differential operator. This allows us to separate the computational steps into two stages: (i) the *off-line* stage, in which the reduced-basis space is constructed; and (ii) the *on-line*/real time stage, in which for each new parameter value the reduced-basis approximation for the output of interest is calculated. The on-line stage is "blackbox" in the sense that there is no longer any reference to the original problem formulation: the computational complexity of this stage scales only with the dimension of the reduced-basis space and the parametric complexity of the partial differential operator.

Although *a priori* theory [FR83, Por85] suggests the optimality of the reduced-basis space approximation, for a particular choice of the reduced-basis space the error in the output of interest is typically not known, and hence the minimal number of basis functions required to satisfy the desired error tolerance can not be ascertained. As a result, either too many or too few basis functions are retained; the former results in computational inefficiency, the latter in uncertainty and unacceptably inaccurate

[1] Department of Mechanical Engineering, Massachusetts Institute of Technology, Room 3-264, 77 Massachusetts Avenue, Cambridge, MA 01239, USA

[2] ASCI, UPR 9029, bâtiment 506, Université Paris-Sud, 91405 Orsay cedex, France; et Analyse numérique, Université Paris VI, 4, place Jussieu, 75252 Paris cedex 05, France

[3] Department of Mechanical Enginering, Massachusetts Institute of Technology, Room 3-264, 77 Massachusetts Avenue, Cambridge, MA 01239, USA

[4] Department of Mechanical Engineering, Massachusetts Institute of Technology, Room 3-243, 77 Massachusetts Avenue, Cambridge, MA 01239, USA

predictions. In this paper we develop blackbox *a posteriori* methods that address these shortcomings. We consider here equilibrium solutions of coercive problems within the context of shape optimization; see also [MPR00] for treatment of noncoercive equilibrium problems and [MMO$^+$00] for symmetric eigenvalue problems.

Numerical Method

Preliminaries

Let Y be a Hilbert space with an associated inner product $(\cdot,\cdot)_Y$ and an induced norm $\|\cdot\|_Y$. We define our parameter space to be $\mathcal{D} \subset \mathbb{R}$; a point in that space is denoted μ. Our problem is then to find $u \in Y$ such that

$$a(u, v; \mu) = \ell(v),\ \forall v \in Y, \tag{1}$$

and subsequently the output of interest $s(u) = \ell^0(u)$; $\ell(\cdot)$ and $\ell^0(\cdot)$ are both in Y', the dual space of Y. The bilinear form a is assumed to be continuous; symmetric, $a(w, v; \mu) = a(v, w; \mu),\ \forall w, v \in Y$; and coercive, $a(v, v; \mu) \geq c\|v\|_Y^2 > 0,\ \forall v \in Y, \forall \mu \in \mathcal{D}$, where c is a strictly positive real constant. Associated with the above primal problem we define the dual problem for $\psi \in Y:\ a(v, \psi; \mu) = -\ell^0(v),\ \forall v \in Y$. The need for this problem will become clear in the error estimation discussion.

We next introduce a symmetric positive-definite form $\hat{a}(w, v)$, and define $\lambda^1_{\hat{a}}(\mu)$ to be the minimum eigenvalue of $a(\varphi, v; \mu) = \lambda(\mu)\hat{a}(\varphi, v),\ \forall v \in Y$. A lower bound for this eigenvalue is required by the output bound procedure: we assume that a $g(\mu)$ is known such that

$$a(v, v; \mu) \geq g(\mu)\hat{a}(v, v) > 0,\ \forall v \in Y \text{ and } \forall \mu \in \mathcal{D}. \tag{2}$$

It is also possible to include approximation of $\lambda^1_{\hat{a}}(\mu)$ as part of the reduced basis approximation [MPR00].

Finally, for the blackbox method, we shall assume that, for some finite integer Q, there exists a decomposition of $a(w, v; \mu)$ of the form

$$a(w, v; \mu) = \sum_{q=1}^{Q} \sigma^q(\mu) a^q(w, v), \forall w, v \in Y \text{and } \forall \mu \in \mathcal{D}, \tag{3}$$

where we make no assumptions on the a^q other than continuity and bilinearity.

Reduced-Basis Approximation

We choose $N/2$ points in our parameter space $\mathcal{D}$, and form the sample set $S_N = \{\mu_1, \ldots, \mu_{N/2}\}$. The reduced-basis spaces associated with the primal and dual problems are then given by $W_N^{pr} = \operatorname{span}\{u(\mu_1), \ldots, u(\mu_{N/2})\}$ and $W_N^{du} = \operatorname{span}\{\psi(\mu_1), \ldots, \psi(\mu_{N/2})\}$ respectively; we can now form

$$W_N = \operatorname{span}\{u(\mu_1), \psi(\mu_1), \ldots, u(\mu_{N/2}), \psi(\mu_{N/2})\} \equiv \operatorname{span}\{\zeta_1, \ldots, \zeta_N\}. \tag{4}$$

The space W_N defined this way has good approximation properties both for the primal and the dual problems.

For each new desired $\mu \in \mathcal{D}$, we now apply a standard Galerkin procedure over W_N to obtain $u_N(\mu)$ and $\psi_N(\mu)$ according to $a(u_N(\mu), v; \mu) = \ell(v),\ \forall v \in W_N$, and $a(v, \psi_N(\mu); \mu) = -\ell^0(v),\ \forall v \in W_N$. The output can then be calculated as $s_N(\mu) = \ell^0(u_N(\mu))$.

Bounds Evaluation

We start by defining the residuals associated with the primal and dual reduced-basis approximations, $R^{pr}(v; \mu) = \ell(v) - a(u_N(\mu), v; \mu),\ \forall v \in Y$, and $R^{du}(v; \mu) = -\ell^0(v) - a(v, \psi_N(\mu); \mu),\ \forall v \in Y$, respectively. The Riesz representations $\hat{e}^{pr}(\mu)$ and $\hat{e}^{du}(\mu)$ of the primal and dual residuals can then be defined as $\hat{a}(\hat{e}^{pr}(\mu), v) = R^{pr}(v; \mu),\ \forall v \in Y,\ \hat{a}(\hat{e}^{du}(\mu), v) = R^{du}(v; \mu), \forall v \in Y$.

We then define, as in [MMO$^+$00, MPR00],

$$\begin{aligned} \bar{s}_N(\mu) &= s_N(\mu) - \frac{1}{2g(\mu)} \hat{a}(\hat{e}^{pr}(\mu), \hat{e}^{du}(\mu)), \\ \Delta_N(\mu) &= \frac{1}{2g(\mu)} \hat{a}^{1/2}(\hat{e}^{pr}(\mu), \hat{e}^{pr}(\mu))\ \hat{a}^{1/2}(\hat{e}^{du}(\mu), \hat{e}^{du}(\mu)), \end{aligned} \tag{5}$$

and compute lower and upper estimators $s_N^{\pm} = \bar{s}_N \pm \Delta_N$.

It can be shown [MMO$^+$00, MPR00] that s_N^+ (respectively s_N^-) will be an upper (respectively lower) bound for s provided that $g(\mu)$ is a lower bound for the eigenvalue $\lambda^1_{\hat{a}}(\mu)$ (or equivalently satisfies (2)). Note that in the general case, where an $\hat{a}$ and $g(\mu)$ which satisfy (2) may not be readily available, the reduced-basis space must be augmented with eigenmodes corresponding to the minimum eigenvalue of the problem $a(\varphi, v; \mu) = \lambda(\mu)\hat{a}(\varphi, v),\ \forall v \in Y$ [MPR00].

Also of interest is the quality of the bounds — how well they approximate the actual error. We measure the quality of the bounds by the effectivity $\eta_N(\mu)$, defined as the ratio of the bound gap Δ_N to $|s - \bar{s}_N|$. From the bound result we know that $\eta_N(\mu) \geq 1$. We can further prove [MPR00] that $\eta_N(\mu)$ is bounded independent of N; in practice, $\eta_N(\mu)$ is typically $O(1)$, as desired.

Blackbox Method

The parametric dependence assumed in (3) permits us to decouple the computation into two stages: the *off-line* stage, in which (i) the reduced basis is constructed and, (ii) the necessary error-estimation preprocessing is performed; and the *on-line* stage, in which for each new desired value of μ, μ_d, we compute $s_N(\mu_d)$ and the associated bounds. The essential "enabler" is the absence of μ dependence in $\hat{a}$, which allows us to precompute (and later assemble) all the "pieces" of $\hat{e}^{pr}(\mu_d)$, and $\hat{e}^{du}(\mu_d)$ by linear superposition. The details of the blackbox technique follow. For convenience we define $\mathcal{N}$ as the set $\{1, \ldots, N\}$, and $\mathcal{Q}$ as the set $\{1, \ldots, Q\}$.

***Off-line* Stage**

1. Calculate $u(\mu_i)$ and $\psi(\mu_i), i = 1, \ldots, N/2$, to form W_N as in (4).
2. Compute $\underline{A}^q \in \mathbb{R}^{N\times N}$ as $A^q_{i,j} = a^q(\zeta_i, \zeta_j), \forall i, j \in \mathcal{N}^2$ and $\forall q \in \mathcal{Q}$.
3. Solve for $\hat{z}^{0,pr} \in Y$ and $\hat{z}^{0,du} \in Y$ from $\hat{a}(\hat{z}^{0,pr}, v) = \ell(v)$, $\forall v \in Y$, and $\hat{a}(\hat{z}^{0,du}, v) = -\ell^0(v)$, $\forall v \in Y$, respectively. Also, compute $\hat{z}^q_j \in Y$ from $\hat{a}(\hat{z}^q_j, v) = -a^q(\zeta_j, v)$, $\forall v \in Y$, $\forall j \in \mathcal{N}$ and $\forall q \in \mathcal{Q}$.
4. Calculate and store $c^{pr}_0 = \hat{a}(\hat{z}^{0,pr}, \hat{z}^{0,pr})$; $c^{du}_0 = \hat{a}(\hat{z}^{0,du}, \hat{z}^{0,du})$; $c^{pr,du}_0 = \hat{a}(\hat{z}^{0,pr}, \hat{z}^{0,du})$; $F^{pr}_{N,j} = \ell(\zeta_j)$ and $F^{du}_{N,j} = \ell^0(\zeta_j)$, $\forall j \in \mathcal{N}$; $\Lambda^{q,pr}_j = \hat{a}(\hat{z}^{0,pr}, \hat{z}^q_j)$ and $\Lambda^{q,du}_j = \hat{a}(\hat{z}^{0,du}, \hat{z}^q_j)$, $\forall j \in \mathcal{N}$ and $\forall q \in \mathcal{Q}$; $\Gamma^{pq}_{ij} = \hat{a}(\hat{z}^p_i, \hat{z}^q_j)$, $\forall i, j \in \mathcal{N}^2$ and $\forall p, q \in \mathcal{Q}^2$.

This stage requires $(NQ + N + 2)$ Y-linear system solves; $(N^2Q^2 + 2NQ + 3)$ $\hat{a}$-inner products; and $2N$ evaluations of linear functionals.

***On-line* Stage**

For each new desired design point μ_d we then compute the reduced-basis prediction and error bound based on the quantities computed in the off-line stage.
1. Form $\underline{A}_N = \sum_{q=1}^{Q} \sigma^q(\mu_d)\underline{A}^q$ and solve for $\underline{u}_N \equiv \underline{u}_N(\mu_d) \in \mathbb{R}^N$ and $\underline{\psi}_N \equiv \underline{\psi}_N(\mu_d) \in \mathbb{R}^N$ from $\underline{A}_N\ \underline{u}_N = \underline{F}^{pr}_N$ and $\underline{A}_N\ \underline{\psi}_N = -\underline{F}^{du}_N$, respectively.
2. Evaluate the bound average and bound gap as

$$\bar{s}_N = (\underline{F}^{du}_N)^T \underline{u}_N - \frac{1}{2g(\mu_d)}\Big(\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{p=1}^{Q}\sum_{q=1}^{Q} u_{N,i}\psi_{N,j}\sigma^p(\mu_d)\sigma^q(\mu_d)\Gamma^{pq}_{ij} + \sum_{j=1}^{N}\sum_{q=1}^{Q}\psi_{N,j}\sigma^q(\mu_d)\Lambda^{q,pr}_j + \sum_{j=1}^{N}\sum_{q=1}^{Q} u_{N,j}\sigma^q(\mu_d)\Lambda^{q,du}_j + c^{pr,du}_0\Big),$$

and

$$\Delta_N(\mu_d) = \frac{1}{2\,g(\mu_d)} \times \Big(\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{p=1}^{Q}\sum_{q=1}^{Q} u_{N,i}u_{N,j}\sigma^p(\mu_d)\sigma^q(\mu_d)\Gamma^{pq}_{ij} + 2\sum_{j=1}^{N}\sum_{q=1}^{Q} u_{N,j}\sigma^q(\mu_d)\Lambda^{q,pr}_j + c^{pr}_0\Big)^{\frac{1}{2}} \times \Big(\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{p=1}^{Q}\sum_{q=1}^{Q} \psi_{N,i}\psi_{N,j}\sigma^p(\mu_d)\sigma^q(\mu_d)\Gamma^{pq}_{ij} + 2\sum_{j=1}^{N}\sum_{q=1}^{Q} \psi_{N,j}\sigma^q(\mu_d)\Lambda^{q,du}_j + c^{du}_0\Big)^{\frac{1}{2}}.$$

respectively.

For each μ_d, $O(N^2Q^2 + N^3)$ operations are required to obtain the reduced-basis solution and the bounds. Since $\dim(W_N) \ll \dim(Y)$, the cost to compute $s_N(\mu_d)$, $\bar{s}_N(\mu_d)$, and $\Delta_N(\mu_d)$ in the on-line stage will typically be much less than the cost to directly evaluate $u(\mu_d)$ and $s(\mu_d) = \ell^0(u(\mu_d))$ from (1).

Results

Instantiation: Fin Problem

To illustrate our method we consider the problem of designing the thermal fin of Figure 1 to cool (say) an electronic component at the fin base, Γ_1. The ith "radiator" of the fin has thermal conductivity k_i (normalized relative to the conductivity of the central post); and the fluid surrounding the fin is characterized by a heat convection coefficient expressed in nondimensional form by a Biot number, Bi. The fin geometry is described by the radiator length β and thickness α, both nondimensionalized with respect to the width of the fin base. We thus obtain $P = 7$, with a typical point in $\mathcal{D} \in \mathbb{R}^7$ given by $\mu = \{k_1, k_2, k_3, k_4, \mathrm{Bi}, \alpha, \beta\}$. For the output of interest we choose the mean temperature of the base, $s(u) = \ell^0(u) = \int_{\Gamma_1} u$, which is directly related to the cooling efficiency of the fin.

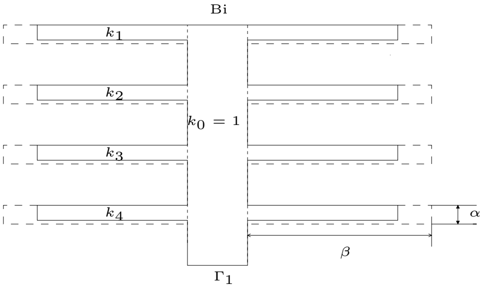


Figure 1

On the original domain the bilinear and linear forms are given by $\int_{\Omega_0} \nabla u \cdot \nabla v + \sum_{i=1}^{4} k_i \int_{\Omega_i} \nabla u \cdot \nabla v + Bi \int_{\partial\Omega \backslash \Gamma_1} uv$, and $\ell(v) = \int_{\Gamma_1} v$; here Ω_0 is the fin central post domain, and Ω_i is the ith radiator domain. (Note $\ell(v) = \ell^0(v)$, and thus the primal and dual problems coincide; this particular case is denoted compliance, and leads to considerable simplification of the numerical procedure.) We then map the domain Ω to a reference fin geometry $\hat{\Omega}$, shown by solid lines in Figure 1. The problem now takes the desired form (1) with $Y = H^1(\hat{\Omega})$ — more exactly, Y is a very fine (and hence very high-dimensional) finite element approximation of $H^1(\hat{\Omega})$ defined over a suitable triangulation of $\hat{\Omega}$. We can readily verify that the resulting form a is symmetric and positive-definite.

Taking advantage of the natural domain decomposition afforded by our mapping, it is then not difficult to cast the problem such that (3) is satisfied with $Q = 16$; the σ^q induced by the variable geometry appear as domain-dependent effective orthotropic conductivities and Bi numbers. Choosing $\hat{a}(u, v) = \sum_{q=1}^{Q} a^q(u, v) = \int_{\hat{\Omega}} \nabla u \cdot \nabla v + \int_{\partial\hat{\Omega} \backslash \Gamma_1} uv$, $g(\mu) = \min_{q \in \{1,\dots,Q\}} \sigma^q(\mu)$ (the σ^q are all bounded from below by a positive constant), we are able to verify (2). Thus all our requirements are honored, and the bound method can be applied.

Accuracy and Effectivity

We first investigate how the dimension of the reduced-basis space affects the accuracy of the bounds. We choose for the design space $\mathcal{D} = [0.1, 10]^4 \times [0.01, 1.] \times [0.1, 0.5] \times [2.0, 3.0]$, and for μ_d the value $\{0.4, 0.6, 0.8, 1.2, 0.1, 0.3, 2.8\}$. To form the reduced space we choose randomly $N/2$ points in $\mathcal{D}$. We plot in Table 1 the bound gap and effectivity as a function of N.

N	Δ_N	η_N
10	1.5987×10^{-1}	2.9947
20	1.5691×10^{-2}	2.8607
30	2.4267×10^{-3}	2.7557
40	7.2616×10^{-4}	2.6250
50	3.0620×10^{-4}	2.6085

Table 1

As we can see from Table 1, even for small N, the accuracy is very good; furthermore, convergence with N is quite rapid. This is particularly noteworthy given the high-dimensional parameter space; even with $N = 50$ points we have less than two points (effectively) in each parameter coordinate. We also note that the effectivity remains roughly constant with increasing N: the estimators are not only bounds, but relatively sharp bounds — good predictors for when N is "large enough." The behavior we observe at this particular value of μ_d is representative of most points in (a random sample over) $\mathcal{D}$, however there can certainly be points where the effectivity is larger: more systematic study is required.

Shape Optimization

Target Temperature

We suppose we wish to find the configuration which yields a base (e.g., chip) temperature of s_* (say 1.8) to within $\epsilon = .01$ by varying only the height (α) of the radiators. To start, we choose a relatively large number of basis functions in the design space $\mathcal{D}$ defined above, and perform the off-line stage of the blackbox method. For efficiency in the *on-line* stage, we then enlist only a subset of these basis functions [Kae00] — those which are closer in the design space to the desired evaluation point — and refine when higher accuracy is required. A binary chop algorithm, summarized below, is implemented to effect the coupled approximation–optimization; we assume monotonicity for simplicity of exposition.

```
for $i$ = 1:maxiter
  Choose $\overline{\alpha} := (\alpha_l + \alpha_r)/2$
  blackbox for $\overline{\alpha} \Rightarrow s_N^+,\ s_N^-$
  $d_1 := \max(|s_* - s_N^+|, |s_* - s_N^-|)$
  $d_2 := \min(|s_* - s_N^+|, |s_* - s_N^-|)$
  if($d_2 > \epsilon$)
    if($s_N^+ > s_*$ and $s_N^- > s_*$) $\alpha_l := \overline{\alpha}$
    if($s_N^+ < s_*$ and $s_N^- < s_*$) $\alpha_r := \overline{\alpha}$
    else $N := N + N^+$
  if($d_1 < \epsilon$) stop
  else
    $N := N + N^+$
next
```

In the particular test case shown in Table 2, we begin with $N = 10$ points and set $N^+ = 10$ as well; we initialize $\alpha_l = 0.1$ and $\alpha_r = 0.5$. During the optimization process, refinement is effected twice, such that a total of $N = 30$ basis functions are invoked (considerably less than the 50 available). The savings are significant, yet we are still ensured, thanks to the bounds, that our design requirement is met to the desired tolerance of $\varepsilon = .01$. One can also apply a dynamic adaptation strategy in which only a minimal number of basis functions are generated (initially) in the off-line stage: if these prove inadequate, we return to the off-line stage for additional basis functions and also revision of the necessary matrices and inner products.

i	$\bar{\alpha}$	s_N^+	s_N^-	α_l	α_r
1	0.3	1.683	1.753	0.1	0.5
2	0.2	1.716	2.056	0.1	0.3
3	0.2	1.766	1.807	0.1	0.3
4	0.2	1.771	1.778	0.1	0.3
5	0.15	1.817	1.840	0.1	0.2
6	0.175	1.792	1.806	.15	0.2

Table 2

If we choose a tighter tolerance ε, or if we wish to investigate many different set points s_*, or if we perform the optimization permitting all 7 design parameters to vary, we would of course greatly increase the number of output predictions required — and hence greatly increase the efficiency of the reduced-basis blackbox technique relative to conventional approaches.

Achievable Set

In multicriterion optimization we consider various (competing) outputs of interest, say volume, $\mathcal{V}$, and root temperature, s. Changing the dimensions of the fin by selecting different α and β will (say) decrease the volume of the fin, and hence material requirements - but also (typically) increase the fin base temperature. It is thus of interest to determine all possible operating points, that is, to generate the map of the "achievable set." In general this will be prohibitively expensive unless one has recourse to a very low-dimensional representation such as the reduced-basis approximation.

We consider this problem for constant conductivities $k_i = 1.$, $i = 0, \dots, 4$, and Biot number Bi$= 0.001$. We then select 100,000 points in the two dimensional design space $[\alpha, \beta] = [0.1, 0.5] \times [2.0, 3.0]$ and evaluate our bounds for s with an error tolerance of 0.1%. Since in this design we wish to be sure that the actual temperature will be less than our prediction, we choose to construct our map based on s_N^+. We are thus insured that at each design point the actual temperature will be lower than that on our curve.

Each evaluation produces a point on the s–$\mathcal{V}$ plane, thus generating the achievable set. Obvious optimality conditions require that we remain on the left or lower boundaries of the achievable set, known as the efficient frontier or trade-off curve in Pareto analysis. As we can see from Figure 2, we can decrease the volume with no real increase in temperature up to the point were the left and lower boundaries cross; after that, the small further possible volume reduction results in a steep rise in base temperature.

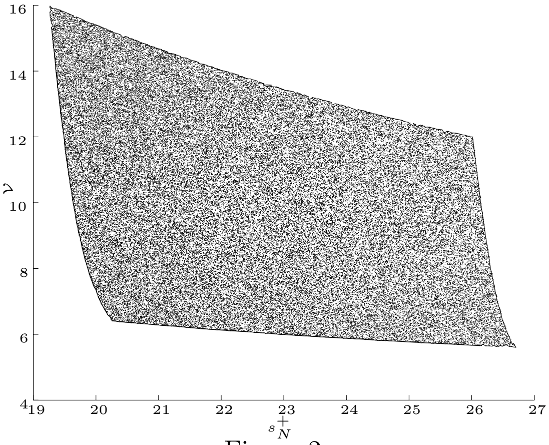


Figure 2

Acknowledgements This work was supported by the Singapore–MIT Alliance, by AFOSR Grant F49620-97-1-0052, and by NASA Grant NAG1-1978.

References

[ASB78]B. O. Almroth, P. Stern, and F. A. Brogan. Automatic choice of global shape functions in structural analysis. *AIAA Journal*, 16:525–528, May 1978.

[FR83]J. P. Fink and W. C. Rheinboldt. On the error behaviour of the reduced basis technique for nonlinear finite element approximations. *Z. Angew. Math. Mech.*, 63(1):21–28, 1983.

[Kae00]R. Kaenel. Reduced basis methods and output bounds for partial differential equations. Master's thesis, MIT, EPFL, 2000.

[MMO$^+$00]L. Machiels, Y. Maday, I. B. Oliveira, A. T. Patera, and D. V. Rovas. Output bounds for reduced-basis approximations of symmetric positive definite eigenvalue problems. *C. R. Acad. Sci. Paris, Série I*, 2000.

[MPR00]Y. Maday, A. T. Patera, and D. V. Rovas. A blackbox reduced-basis output bound method for noncoercive linear problems. *College de France Series; also MIT-FML Report 00-2-1*, 2000.

[Nag79]D. A. Nagy. Modal representation of geometrically nonlinear behaviour by the finite element method. *Computers and Structures*, 10:683–688, 1979.

[NP80]A. K. Noor and J. M. Peters. Reduced basis technique for nonlinear analysis of structures. *AIAA Journal*, 18(4):455–462, April 1980.

[Por85]T. A. Porsching. Estimation of the error in the reduced basis method solution of nonlinear equations. *Mathematics of Computation*, 45(172):487–496, October 1985.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau,

46. Best N-term capacitance approximation on sparse grids

P. Oswald[1]

Introduction

In [GOS99], adaptive sparse grid spaces spanned by a finite number of tensor-product L_2-orthogonal Haar functions have been applied to capacitance calculations on a unit screen. In this note, we state asymptotically optimal approximation rates for this problem when choosing the best possible adaptive sparse grid space of a given dimension N. We also compare the results with other recent approaches to efficiently solve this problem and comment on some numerical tests. Details of the proofs and a discussion of the approximation-theoretical aspects have appeared in [Osw99].

For a flat square screen $I^2 \equiv [0,1]^2$, we consider the *single layer potential equation*

$$\frac{1}{4\pi}\int_{I^2} \frac{f(y)}{|x-y|_2}\,dy = g(x)\ , \qquad x \in I^2\ . \tag{1}$$

As this problem can be cast in variational form and leads to a symmetric $H^{-1/2}$-elliptic problem, Galerkin methods can be set up and allow for a straightforward analysis. E.g., convergence and error estimates in Sobolev norms (most naturally in the $H^{-1/2}$-related energy norm) can be obtained for many natural discretization spaces. There are two obstacles that trigger further investigations. First, one is interested in as small as possible computational subspaces since the discretization leads to *dense matrices* which is in contrast to the situation in finite element or finite difference methods for partial differential equations. Several approaches are under investigation (see [GOS99] for a brief discussion) to overcome this problem. We only mention adaptive wavelet compression schemes [Dah97, vPS97] and the hp-version of the boundary element method [Ste96] which will be used for comparison below. These methods also deal with the second obstacle: solutions of problems such as (1) exhibit very low global Sobolev smoothness due to dominant corner and edge singularities. For the important special case $g(x) \equiv 1$, the so-called *capacitance problem*

$$\frac{1}{4\pi}\int_{I^2} \frac{f(y)}{|x-y|_2}\,dy = 1\ , \quad x \in I^2\ , \tag{2}$$

the variational solution $f \in H^{-1/2}(I^2)$ does not even belong to $L_2(I^2)$. This leads to very slow convergence rates of any standard Galerkin method, both theoretically and practically. However, in analogy to elliptic problems in polyhedral domains, the 'bad' behavior of a solution f of (1) for smooth data g can be separated into a few singularity components associated with the edges and corners of I^2, i.e., one can write

$$f = f^{sing} + f^{reg}\ , \tag{3}$$

[1] Bell Laboratories, poswald@research.bell-labs.com

where the *singular part* f^{sing} is a finite linear combination of specific, prescribed singularity functions (usually composed of terms of the form $\mathrm{dist}(x,F)^{\alpha}$ and $\log(\mathrm{dist}(x,F))$, where F is an edge or an vertex of I^2) while the *regular part* can be as smooth as wanted (limits are set by the smoothness class of g). See [vP89, vPa90] for details on the singularity decomposition (3) for (1) and similar screen problems. Thus, to obtain improved rates of convergence it would be enough to adapt the computational subspace such that it approximates the singularity functions in f^{sing} as well as the smooth part f^{reg}. In practical algorithms, this basic idea is implemented a priori (e.g., by using *graded meshes* in h- and hp-version boundary element methods [HMS97]) or by using *feedback adaptivity schemes*, e.g., based on *a posteriori error estimators*, as suggested by different authors [HMS97, Dah97].

Without becoming too detailed, let us mention some theoretical approximation results for the h- and hp-version of the boundary element method, on the one hand, and the wavelet schemes, on the other. Throughout the paper, wavelets are *semi-orthogonal spline wavelets* of low order m, even though results for this class of ansatz spaces are valid under much more general assumptions [Dah97]. The boundary element spaces for the h-version are piecewise polynomials or splines of order m on certain sequences of partitions of I^2 (both quasi-uniform and adaptively refined ones) while in an hp-method, in addition, the polynomial degree may vary in each element of the underlying partition. Subsequently, we will specialize to the simplest case $m = 1$ of piecewise constant approximation. Our model problem will be (2). From [vPa90] it follows that the singular part f^{sing} of the solution f of (2) is representable as a sum of singularity functions, with the leading singularities of the form $\sim \mathrm{dist}(x,e)^{-1/2}$ near the interior part of any edge e of I^2, and $\sim \mathrm{dist}(x,P)^{\gamma-1}$, $\gamma \approx 0.2966...$, if x approaches a vertex P of I^2. This leads to $f \in H^{-\varepsilon}(I^2)$ for any $\varepsilon > 0$, a result which brakes down (due to the edge singularity) for $\varepsilon = 0$. Throughout the paper, the notation ε stands for an arbitrarily small positive parameter.

To make a fair comparison between different approximation methods, we will relate error quantities to the dimension $N = \dim V_N$ of the computational subspace V_N from which the Galerkin solution is determined, and not to a meshsize parameter of the underlying partition or to the level number of a space in a wavelet multiresolution analysis. We admit that this way of comparison is still disputable since computational work and storage limitations may be quite different for subspaces with the same N. All estimates are given for the best approximations in the $H^{-1/2}(I^2)$ norm,

$$e_N(f)_{-1/2} = \inf_{v_N \in V_N} \|f - v_N\|_{H^{-1/2}} \ ,$$

which is equivalent to estimating the error in energy norm between the Galerkin solution $f_N \in V_N$ and the solution f of (2). Moreover, capacitance errors $\delta_N \equiv |\mathcal{C} - \mathcal{C}_N|$, where

$$\mathcal{C} = \frac{1}{4\pi}\int_{I^2} f(y)\,dy\ , \qquad \mathcal{C}_N = \frac{1}{4\pi}\int_{I^2} f_N(y)\,dy\ , \tag{4}$$

are covered, too, since

$$\delta_N = \mathcal{C} - \mathcal{C}_N \asymp \|f - f_N\|^2_{H^{-1/2}} \asymp e_N(f)^2_{-1/2}\ . \tag{5}$$

- *h-version with quasi-uniform partitions and fixed polynomial degree resp. non-adaptive wavelet spaces.* Here, standard estimates

  $$e_N(u)_{-1/2} \le C h_N^{t+1/2} \|u\|_{H^t} , \quad u \in H^t(I^2) , \; -1/2 < t \le m ,$$

  hold with a mesh parameter $h_N \approx N^{-1/2}$, and lead in conjunction with the above regularity result for f to the estimate

  $$e_N(f)_{-1/2} = \mathrm{O}(N^{-(1/4-\varepsilon)}) , \qquad N \to \infty , \tag{6}$$

  The asymptotic behavior in (6) is independent of m, and much worse than the saturation order $\mathrm{O}(N^{-(m/2+1/4)})$ valid for approximating smooth functions from $H^m(I^2)$ with respect to the same spaces V_N.

- *h-version with graded meshes.* The estimate (6) can be improved if graded meshes are allowed for partitioning I^2, see [vPa90]. For I^2, these are based on tensor-product partitions where the univariate partitions have $n \asymp \sqrt{N}$ grid points $\xi_i \in (0,1)$ which behave like $\sim (i/n)^\beta$ near the left endpoint (analogous refinement is assumed at the right endpoint of $[0,1]$). For appropriate β, the above mentioned saturation order can be reached:

  $$e_N(f)_{-1/2} = \mathrm{O}(N^{-(1/4+m/2)}) , \qquad N \to \infty , \tag{7}$$

  for the associated spaces of piecewise polynomials or splines of order m on the above partitions, see [vP89, vPa90]. This improvement is achieved by allowing high aspect ratios of the rectangles (anisotropic refinement) near the edges.

- *hp-version on geometric meshes.* The best asymptotic estimates are known for the hp-method and a geometric tensor-product mesh (now the univariate meshes are given by $\xi_i \sim \sigma^{n/2-i}$, $\sigma < 1$, near the left endpoint of $[0,1]$). The result for the particular case under consideration (see [HMS97, Ste96]) is

  $$e_N(f)_{-1/2} = \mathrm{O}(e^{-cN^{1/4}}) , \qquad N \to \infty . \tag{8}$$

- *Adaptive wavelet approximation.* The basic idea is to determine a wavelet space V_N as the linear span of N carefully selected wavelets ψ_λ from different levels of the underlying multiresolution analysis. Theoretically, assuming that $f = \sum_\lambda c_\lambda \psi_\lambda$ is decomposed into a wavelet series, and that $\Psi = \{\psi_\lambda\}$ forms a Riesz basis in $H^{-1/2}$, the best one can do is to select the terms with the N largest $|c_\lambda|$. An algorithm which uses this basic idea has been described in [Dah97]. The supporting approximation-theoretical result behind it has been known for some years [DJP92, Osw90]. It is now referred to under the name *nonlinear N-term approximation* and has found important applications to image compression and adaptive algorithms, see [DeV98]. The bad news, however, is that for our f this only leads to an estimate of

  $$e_N(f)_{-1/2} = \mathrm{O}(N^{-(1/2-\varepsilon)}) , \qquad N \to \infty , \tag{9}$$

  again independently of m. This is a slight improvement over (6) but even the h-version on optimally chosen graded meshes with piecewise constants does asymptotically better than any adaptive wavelet space. The main reason is that for the wavelet bases considered in [Dah97, DeV98] (and in most of the literature on solving boundary integral equations by wavelet methods), the nonlinear N-term approximation models *optimal isotropic local h-refinement.* Thus, for resolving the dominating edge singularities in the solution of (2), too many wavelet functions are necessary to improve the resolution along edges. This effect does not occur for point singularities and is practically invisible for edge singularities that are weaker than those exhibited by the solutions of screen problems (compare [DD97]).

Clearly, from the above one would prefer graded resp. geometric meshes (combined with h- resp. hp-methods) over wavelet type methods for the application under consideration. The exponential convergence of the hp-method is hard to beat in the asymptotic range. However, since the implementation of an hp-method for integral equations is by no means trivial, simpler and less optimal methods may still have a chance. E.g., well-understood adaptivity and compression strategies, preconditioning, and canonical data structures are some advantages of wavelet methods that one might wish to explore.

Improving upon the relatively weak approximation potential for solutions of screen problems while still working in a wavelet multiresolution analysis is suggested by the results on *adaptive sparse grid spaces* in [GOS99]. In the present note we describe the approximation rates obtainable from these spaces in more quantitative terms. Roughly speaking, our general claim is that under the same assumptions on f, by changing from the traditional, isotropic wavelet constructions on I^2 to tensor-product, anisotropic wavelet systems Ψ^*, the unsatisfactory rates of (9) can be replaced by

$$e_N^*(f)_{-1/2} = \mathrm{O}(N^{-(1/4+m)}) , \qquad N \to \infty , \tag{10}$$

where $e_N^*(f)_{-1/2}$ describes now the best N-term approximation with respect to the new wavelet system Ψ^*. Our point is that, even without going to graded meshes, we can expect good results if standard wavelet systems are replaced by tensor-product wavelet systems. We give precise statements for the case $m = 1$ (piecewise constant approximation) in the next section. Numerical experiments are presented in the last part.

N-TERM APPROXIMATION BY HAAR FUNCTIONS

Let us give the definition of the Haar-wavelet systems ($m = 1$) under consideration. The characteristic function of a set Ω will be denoted by χ_Ω. Let $\mathcal{D}_j$ be the system of dyadic intervals Δ of length $|\Delta| = 2^{-j}$, $j \geq 0$, of $I \equiv [0, 1]$. Any $\Delta \in \mathcal{D}_j$ uniquely splits into left (Δ^+) and right (Δ^-) half-intervals from $\mathcal{D}_{j+1}$. Set

$$\phi_\Delta = |\Delta|^{-1/2}\chi_\Delta , \qquad \psi_\Delta = |\Delta|^{-1/2}(\chi_{\Delta^+} - \chi_{\Delta^-}) , \quad \Delta \in \mathcal{D} = \cup_{j\geq 0}\mathcal{D}_j ,$$

for the univariate scaled box functions and Haar functions, respectively. The *standard bivariate Haar system* is given by

$$\Psi_H = \cup_{j\geq 0}\ \Psi_j ,$$

where Ψ_0 consists of the only function χ_{I^2}, and

$$\Psi_j = \{\psi_\Delta(x_1)\phi_{\Delta'}(x_2),\ \phi_\Delta(x_1)\psi_{\Delta'}(x_2),\ \psi_\Delta(x_1)\psi_{\Delta'}(x_2),\ \Delta, \Delta' \in \mathcal{D}_{j-1}\}$$

for $j \geq 1$.The supports of Haar functions from Ψ_j are dyadic squares of sidelength 2^{-j+1}, $j \geq 1$. In contrast, the Haar functions in the tensor-product bivariate Haar system

$$\Psi_H^* = \cup_{j_1,j_2 \geq 0}\ \Psi_{j_1,j_2}^*\ ,$$

where

$$\Psi_{j_1,j_2}^* = \{\psi_\Delta(x_1)\psi_{\Delta'}(x_2)\,,\ \Delta \in \mathcal{D}_{j_1-1}, \Delta' \in \mathcal{D}_{j_2-1}\}\ ,$$

possess rectangular support. For notational convenience, we defined $\mathcal{D}_{-1} = \{[0,2]\}$ and $\psi_{[0,2]} = \phi_I$. Obviously, both systems are complete orthonormal systems in $L_2(I^2)$.

We are interested in the behavior of best N-term approximations with respect to Ψ_H^*

$$e_N^*(f)_s = \inf_{\Psi_N^* \equiv \{\psi_1,\dots,\psi_N\} \subset \Psi_H^*}\ \inf_{v_N \in V_N^* \equiv \text{span}\,\Psi_N^*} \|f - v_N\|_{H^s}\ , \quad N \geq 1\ , \tag{11}$$

in the $H^s(I^2)$-norm. Due to the approximation and smoothness properties of piecewise constant functions, only the range $-1 < s < 1/2$ is of interest. Two main theorems are established (for a detailed exposition and proofs, we refer to [Osw99]). The first theorem serves functions from spaces of functions with dominating mixed derivatives which can be defined as tensor products of univariate Sobolev spaces:

$$H_{\text{mix}}^t(I^2) = H^t(I) \otimes H^t(I)\ , \qquad -\infty < t < \infty\ .$$

For $t = 0$, we have $H_{\text{mix}}^0(I^2) \cong L_2(I^2)$ while $f \in H_{\text{mix}}^1(I^2)$ if f belongs to $H^1(I^2)$ and additionally possesses a weak mixed derivative $\partial_{11} f \in L_2(I^2)$.

Theorem 1 *Let $f \in H_{\text{mix}}^t(I^2)$ for some $-1/2 < t \leq 1$. Then its best N-term approximations with respect to Ψ_H^* in $H^s(I^2)$, $N \geq 1$, satisfy*

$$e_N^*(f)_s \leq C\|f\|_{H_{\text{mix}}^t} \begin{cases} N^{-(t-s)}\ , & 0 < s < 1/2\ ,\ \ s < t \leq 1\ , \\ N^{-t}(1+\log N)^t\ , & s = 0 < t < 1\ , \\ N^{-1}(1+\log N)^{3/2}\ , & s = 0\ ,\ t = 1\ , \\ N^{-(t-s/2)}\ , & -1 < s < 0\ ,\ \ s/2 < t \leq 1\ . \end{cases}$$

In particular, if $f \in H_{\text{mix}}^1(I^2)$ then

$$e_N^*(f)_{-1/2} \leq CN^{-5/4}\|f\|_{H_{\text{mix}}^1}\ , \quad N \to \infty\ . \tag{12}$$

This estimate is applicable to the smooth part f^{reg} of solutions to (1), to achieve the O$(N^{-5/4})$ error bound in practice, one can, e.g., take subspaces spanned by the following subset of $\asymp 2^J$ Haar functions:

$$\Psi_{H,J}^* = \cup_{j_1,j_2 \geq 0\,:\, j_1+j_2+\frac{1}{4}\max(j_1,j_2) \leq \frac{9}{8}J}\ \Psi_{j_1,j_2}^*\ . \tag{13}$$

Note that $\Psi^*_{H,J}$ spans a subspace of the standard sparse grid space of level J.

The second result covers certain types of *singularity functions.* We call $f \in L_1(I^2)$ an *edge singularity function with exponent* $\alpha \in [0,1)$ if it has a continuous derivative $\partial_{11} f$ in the open square $(0,1)^2$ and satisfies

$$|\partial_{kl} f(x_1,x_2)| \le C(\min(x_1, 1-x_1))^{-\alpha-k}(\min(x_2,1-x_2))^{-\alpha-l}$$

for all $(x_1,x_2) \in (0,1)^2$ and $0 \le k,l \le 1$. E.g., the singular part f^{sing} in (3) of the solution f of (2) possesses this property with $\alpha = 0.7034...$ (a more detailed analysis shows that for the capacitance problem better representations of f^{sing} can be found which would lead to edge singularity functions with $\alpha = 1/2$ as the appropriate value).

Theorem 2 *Let* $-1 < s < 1/2$, *and* f *be an edge singularity function with exponent* α, *where* $0 \le \alpha < \min(1/2 - s, 1/2 - s/2)$. *Then* $f \in H^s(I^2)$ *and*

$$e^*_N(f)_s \le C \begin{cases} N^{-(1-s)}\,, & 0 < s < 1/2\,, \\ N^{-1}(\log N)^{3/2}\,, & s = 0\,, \\ N^{-(1-s/2)}\,, & -1 < s < 0\,, \end{cases} \qquad N \to \infty\,. \tag{14}$$

Roughly speaking, by optimally choosing N Haar functions from $\Psi^*_H(I^2)$, an edge singularity function with exponent α satisfying the above condition possesses the same asymptotic N-term approximation rate as smooth functions from $H^1_{\text{mix}}(I^2)$. For the case $s = -1/2$, we can have $0 \le \alpha < 3/4$ which leads according to our above remarks to

$$e^*_N(f^{sing})_{-1/2} \le CN^{-5/4}\,, \quad N \to \infty\,, \tag{15}$$

for the singular part of the solution f of the capacitance problem (2). Since, at the same time, we can assume that $f^{reg} \in H^1_{\text{mix}}(I^2)$ in (3), the two estimates (12) and (15) yield an analogous estimate for f itself. Finally, from (4) we see that the capacitance $\mathcal{C}$ of the unit square screen can be approximated at a rate of $\mathrm{O}(N^{-5/2})$ if optimal selections of N Haar functions from $\Psi^*_H(I^2)$ are used to build discretization spaces.

NUMERICAL TESTS

In Section 3.3-4 of [GOS99], capacitance approximations have been computed for full grid (fg-), sparse grid (sg-) and adaptive sparse grid (asg-) spaces. To reach a relative capacitance error δ^{rel}_N of approximately 10^{-3}, subspaces V_N of dimension $N = 65536$, $N = 1280$, and $N = 68$, respectively, were needed. The proofs of Theorem 1 and 2 suggest the use of new asg-spaces with slightly improved convergence properties (see Table 1). In order to achieve the above-mentioned asymptotical error estimate $\mathrm{O}(N^{-5/2})$, it should be sufficient to take the union of the set $\Psi^*_{J_0,H}$ defined in (13) which serves the regular part f^{reg}, and a set Ψ^*_N consisting of $N \asymp 2^{J_0}$ functions from $\Psi^*_H(I^2)$ producing the N largest contributions to the upper bound

$$\|f^{sing}\|^2_{H^{-1/2}} \le C \sum_{j_1,j_2} \sum_{\psi \in \Psi^*_{j_1,j_2}} 2^{-\max(j_1,j_2)} |c_\psi(f^{sing})|^2 \tag{16}$$

Table 1: Relative capacitance errors for various V_N

fg-spaces		sg-spaces		asg-spaces [GOS99]		new asg-spaces	
N	δ_N^{rel}	N	δ_N^{rel}	N	δ_N^{rel}	N	δ_N^{rel}
4	0.08302	3	0.08302	20	0.00921	9	0.02516
16	0.04584	8	0.04589	32	0.00495	17	0.00738
64	0.02490	20	0.02511	44	0.00268	25	0.00254
256	0.01310	48	0.01340	56	0.00150	33	0.00130
1024	0.00677	112	0.00708	68	0.00090	41	0.00098
4096	0.00346	256	0.00373	80	0.00060	61	0.00069
16384	0.00175	576	0.00197	92	0.00044	81	0.00051
65536	0.00089	1280	0.00105	104	0.00037	101	0.00036

for the singular part f^{sing} from (3). The proof of Theorem 2 also shows that the unknown Haar-Fourier coefficients $c_\psi(f^{sing})$ can be replaced by computable upper bounds. These, in turn, can be obtained from using appropriate majorants for f^{sing} (such as appearing in the definition of edge singularity functions with exponent α or obtained directly from the available singularity decompositions, see [HMS97, GOS99]). Tuning N, J_0, and choosing different majorants may lead to further improvement.

In our experiments, the sets Ψ_N^* have been obtained from thresholding the sequence $\{2^{-\max(j_1,j_2)}|c_\psi(f^\alpha)|^2\}$ for the function $f^\alpha(x_1,x_2) = x_2^{-\alpha}$ (which mimics a singularity along the edge $x_2 = 0$ of the unit square) and a straightforward symmetrization step (note that for the solution of (2) satisfies $f(x_1,x_2) = f(x_1, 1-x_2) = f(1-x_1,x_2)$). Using the values $\alpha = 1/2$, $J_0 = 3$, we found that the above-mentioned relative error of 10^{-3} can be reached by using 41 ansatz functions (the constant function from $\Psi_{0,0}^*$ and four functions with support along the edges from each of the sets $\Psi_{0,j}^* \cup \Psi_{j,0}^*$, $j = 2,\ldots,11$). This hints at the importance of dealing with the edge singularities adequately, and in the first place.

We also performed some a posteriori analysis by first computing the numerical solution on a sufficiently large adaptive sparse grid space (dimensions $N_{\max} = 277$ and $N_{\max} = 409$ have been tried), and then applying the above thresholding procedure to the obtained set of approximate Fourier coefficients. For small $N << N_{\max}$, this procedure leads to essentially the same spaces as used to produce the results of the last column of Table 1. Lack of space prevents us from giving more details (see the extended version of this note at `http://cm.bell-labs.com/who/poswald`).

Conclusion

It is demonstrated that properly selected, small subsystems of the tensor-product Haar system can be used as ansatz functions in a Galerkin scheme for the single layer potential equation to obtain the capacitance of a square screen with a relative accuracy of up to 10^{-4} in a highly efficient way. Theoretical support is given by providing sharp asymptotic estimates for the best N-term approximation with regard to this Haar system in Sobolev norms and various classes of functions (including those typical for the solutions of the single layer potential equation), and comparing them with analogous results for other popular approximation schemes for this problem.

The results also highlight, under model assumptions, the importance of anisotropic refinement along the edges of the screen and represent an interesting improvement over the use of graded meshes. The advantage is that only mesh-structures based on coordinate-wise dyadic refinement need to be implemented and that in an adaptive scheme that selects the right subset of the Haar system on this mesh-structure, the overall approximation rate measured in terms of dimensions of the resulting computational subspaces is even better.

References

[Dah97]W. Dahmen. Wavelet and multiscale methods for operator equations. *Acta Numerica*, 6:55–228, 1997.

[DD97]S. Dahlke and R. A. DeVore. Besov regularity for elliptic boundary value problems. *Commun. Partial Diff. Eqns.*, 22:1–16, 1997.

[DeV98]R. A. DeVore. Nonlinear approximation. *Acta Numerica*, 7:51–150, 1998.

[DJP92]R. A. DeVore, B. Jawerth, and V. Popov. Compression of wavelet decompositions. *Amer. J. Math.*, 114:737–785, 1992.

[GOS99]M. Griebel, P. Oswald, and T. Schiekofer. Sparse grids for boundary integral equations. *Numer. Math.*, 83:279–312, 1999.

[HMS97]N. Heuer, M. Maischak, and E. P. Stephan. The hp-version of the boundary element method for screen problems. Technical Report ifam6, IfAM, University Hannover, September 1997.

[Osw90]Peter Oswald. On the degree of nonlinear spline approximation in besov-sobolev spaces. *J. Appr. Theory*, 61:131–157, 1990.

[Osw99]P. Oswald. On N-term approximation from the Haar system in H^s-norms. In S. M. Nikolskij, B. S. Kashin, and A. D. Izaak, editors, *Metric Function Theory and Related Problems of Analysis*, pages 137–163. AFC, Moscow, 1999. in Russian.

[Ste96]E. P. Stephan. The h-p version of the boundary element method for solving 2- and 3-dimensional problems. *Comp. Meth. Appl. Mech. Eng.*, 1996:183–208, 1996.

[vP89]T. von Petersdorff. *Randwertprobleme der Elastizitätstheorie für Polyeder-Singularitäten und Approximation mit Randelementmethoden.* PhD thesis, TH Darmstadt, 1989. in German.

[vPa90]T. von Petersdorff and. Regularity of mixed boundary value problems in $\mathbf{R}^3$ and boundary element methods on graded meshes. *Math. Meth. Appl. Sci.*, 12:229–249, 1990.

[vPS97]T. von Petersdorff and C. Schwab. Fully discrete multiscale Galerkin BEM. In W. Dahmen, A. J. Kurdila, and P. Oswald, editors, *Multiscale Wavelet Methods for Partial Differential Equations*, chapter 6, pages 287–346. Academic Press, San Diego, 1997. Wavelet Analysis and Its Applications.

12th International Conference on Domain Decomposition Methods
Editors: Tony Chan, Takashi Kako, Hideo Kawarada, Olivier Pironneau, ©2001 DDM.org

47. Shape Optimization for an Acoustic Problem

H. Suito [1], H. Kawarada [2]

Introduction

In this paper, an optimal shape design for an interfacial boundary between different media, through which sound propagates, is discussed. For example, designs for soundproof walls along high-speed train routes or highways, walls of concert halls, etc are included in the same category.

For the above-mentioned problems, an algorithm to search for an optimal shape was proposed and tested numerically in the three-dimensional problems in [KS99], in which Fuzzy Optimization Method (FOM)[KS97] was used effectively.

Originally, FOM was invented as a local minimizer search algorithm. In order to look for a global minimizer, Multi-start Fuzzy Optimization Method (MS-FOM), which is a hybrid algorithm with FOM and Genetic Algorithms (GAs), has been developed on the basis of FOM[KOPS98].

An application of MS-FOM to such optimization problems makes it possible not only to look for a global minimizer but also to clarify the structure of the manifold of the cost functional defined in the parameter space. This fact depends mainly upon the functions of MS-FOM, one of which is counting-up of all local minimizers.

Here, the algorithm to search for an optimal shape by use of MSFOM is briefly stated and numerical results using it are presented. An observation of such results indicates the rather precise structure of the cost manifold, i.e., the distribution of local maximizers and minimizers in the parameter space. Such observation may be impossible by an application of other global minimizer searching algorithms, for example, Genetic Algorithms.

Finally, physical meanings of a set of local maximizers will be discussed from the view point of resonance phenomena corresponding to the variations of eigenfrequencies of coupled media based on the shape change of the interfacial boundary. Through the discussion mentioned above, shape optimization for an acoustic problem arouses careful treatment to look for a global minimizer.

Shape optimization problem

Configuration

- Γ_{top} and Γ_{bottom} are rigid boundaries, i.e., the density of these walls is infinity. Hence, a sound wave is completely reflected at these boundaries.

- Γ_{in} is a vibrating plate which generates a sound wave.

[1] Department of Urban Environment Systems, Chiba University, 1-33 Yayoicho, Inage-ku, Chiba, 263-8522, Japan. suito@tu.chiba-u.ac.jp

[2] Department of Urban Environment Systems, Chiba University, 1-33 Yayoicho, Inage-ku, Chiba, 263-8522, Japan. kawarada@tu.chiba-u.ac.jp

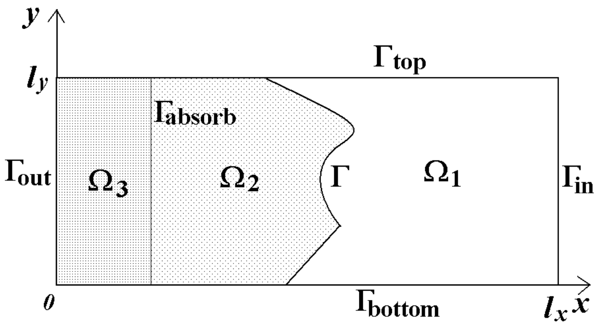


Figure 1: Geometry

- Ω_1 is occupied by water.
- Ω_2 is assumed to be made of pine timber, the role of which is to absorb the sound wave coming through Ω_1.
- Γ is the boundary between Ω_1 and Ω_2. We will try to optimize its shape to transmit the sound wave into Ω_2 as much as possible.
- Ω_3 is a so-called Fictitious Domain, i.e., artificial domain to approximate the boundary condition at infinity. In this domain, Helmholtz eq. with complex wave number is assumed, which is derived from Navier-Stokes eq. including the viscosity term. A sound wave transmitted from Ω_2 is almost completely damped in this domain and is not reflected into Ω_2.
- Γ_{absorb}, on which the amplitude of an absorbed sound wave in Ω_2 is computed.
- Γ_{out}, on which no sound waves exist because of the damping effect in the domain Ω_3.
- $\Omega = \Omega_1 \cup \Gamma \cup \Omega_2 \cup \Omega_3 = (0, l_x) \times (0, l_y)$
- $u^{(i)}(x, y)$ $(i = 1, 2, 3)$: Complex sound pressure.
- k_i $(i = 1, 2, 3)$: Wave number.
- ω : Angular velocity of the incident wave.
- ρ_i $(i = 1, 2, 3)$: Density of medium.
- $\mathbf{n}$: Outward normal vector on the boundaries.
- Γ : Interfacial boundary between Ω_1 and Ω_2.
- α : An incident angle of plane wave.

where $i = 1$ means water, $i = 2$ means pine and $i = 3$ means the fictitious domain.

Parameterization of the interfacial boundary

In order to parameterize the shape of the interfacial boundary, a scaling function for wavelet is introduced as follows;
Let

$$\eta_0(x) = \begin{cases} 1 & x \in [0,1], \\ 0 & else, \end{cases} \tag{1}$$

$$\begin{aligned} f_0(x) &= (-0.585x^2 + 1.867x)\eta_0(x), & (2)\\ f_1(x) &= (1.170x^2 - 2.734x + 1.282)\eta_0(x), & (3)\\ f_2(x) &= (0.585x^2 + 0.867x - 0.282)\eta_0(x). & (4) \end{aligned}$$

and

$$\phi(x) = f_0(x) + f_1(x-1) + f_2(x-2). \tag{5}$$

We define

$$\phi_{L,m}(x) = \sqrt{N_L} \cdot \phi(N_L x - m) \ \ (m \in Z) \tag{6}$$

where $N_L = 2^L$. Then $\{\phi_{L,m}\}$ constitutes an orthonormal set, i.e.,

$$\int_R \phi_{L,m}\phi_{L,m'}dx = \delta_{m,m'}. \tag{7}$$

By using these scaling functions, we parameterize the interfacial boundary by means of a superposition of $\phi_{L,m}(y)$, i.e.,

$$\Gamma(y) = \sum_m \gamma_m \cdot \phi_{L,m}(y). \tag{8}$$

Admissible set for the deformation of the interfacial boundary is defined by

$$\mathcal{A}_1 = \{\gamma_m \in R | \, |\gamma_m| \le K \ (m = 1, 2, 3, \cdots, M_1)\}. \tag{9}$$

Definition of optimization problem

Define the state equation;

$$\begin{cases} (\triangle + k_i^2)\, u^{(i)}(\Gamma, a) = 0 & \text{in } \Omega_i, \ \ (i = 1, 2, 3), \\ u^{(1)}(\Gamma, a) = u^{(2)}(\Gamma, a) = a & \text{on } \Gamma, \\ \dfrac{\partial u^{(i)}(\Gamma, a)}{\partial n} = 0 & \text{on } \Gamma_{top} \cup \Gamma_{bottom} \ (i = 1, 2, 3), \\ u^{(1)}(\Gamma, a) = e^{ik_1 \cos\alpha l_x} e^{ik_1 \sin\alpha y} & \text{on } \Gamma_{in}, \\ u^{(2)}(\Gamma, a) = 0 & \text{on } \Gamma_{out}, \end{cases} \tag{10}$$

and the cost function;

$$J_c(\Gamma, a) = -\int\limits_{\Gamma_{absorb}} \left|u^{(2)}(\Gamma, a)\right|^2 d\Gamma$$
$$+\frac{1}{\varepsilon}\int\limits_{\Gamma}\left|\frac{1}{\rho_1}\frac{\partial u^{(1)}(\Gamma, a)}{\partial n} - \frac{1}{\rho_2}\frac{\partial u^{(2)}(\Gamma, a)}{\partial n}\right|^2 d\Gamma. \tag{11}$$

In the definition of the cost function, the constraint caused by the transmission condition is included as a penalty term with a small positive parameter ε.

The Dirichlet datum a is defined on Γ by

$$a = \sum_m a_m \cos(\frac{\pi m}{l_y} y). \tag{12}$$

Admissible set for a is represented by

$$\mathcal{A}_2 = \{a_{mm'} \in \mathbf{C} \;\; (m, m' = 0, 1, 2, \cdots, M_2) | |a_{mm'}| \leq L\}. \tag{13}$$

Therefore, our minimization problem is;

$[P_r]$: Minimize $J_c(\Gamma, a)$ for $(\Gamma, a) \in \mathcal{A} = \mathcal{A}_1 \times \mathcal{A}_2$.

Numerical solution of Helmholtz equation

In order to compute the sound field in the domain bounded by a complicated interfacial boundary, the coordinate transformation is used as follows;

1. Generate mesh system in the deformed domain, which is the transformation from physical domain to computational one;

$$x = x(\xi, \eta), \; y = y(\xi, \eta). \tag{14}$$

2. Transform differential operators by use of (14).
3. Transform Helmholtz eq. by use of (14).

Transformed Helmholtz equation is discretized by use of finite difference method. Discretized Helmholtz eq. constitutes a large-scale system of equations. In order to solve this system of equations, GPBi-CG method[Zha97] is adopted.

Hybridized algorithm by FOM and GAs

In this section, Multi-start Fuzzy Optimization Method, which is a hybridized algorithm by Fuzzy Optimization Method (FOM) and Genetic Algorithms (GAs), is briefly summarized. Let us define operators F, M and R as follows.

- F : Algorithm due to Fuzzy Optimization Method. This procedure is a down-hill process on the cost manifold. (Refer to [KS97] for the detailed implementations.)

- M : Mountain crossing algorithm. This procedure is a up-hill process on the cost manifold. (Refer to [KOPS98] for the detailed implementations.)
- R : Rearrangement algorithm by GAs. In this procedure, starting points for the next down-hill process are rearranged by use of GAs.

Solution algorithm of (P_r)

The algorithm of Multi-start FOM is stated in the following way;

Step 1 Give an initial population W^0 (the set of searchers).

Step 2 Compute $U^n := FW^n$ (the set of local minimizers obtained).

Step 3 Compute $V^n := MU^n$ (the set of quasi-local maximizers obtained).

Step 4 Compute $W^n := RV^n$ (the set of rearranged searchers).

Step 5 Increase generation number $n := n + 1$ and repeat steps from **2** to **4** until the generation number n is beyond the preset one.

It should be noted that the operation R is applied in order to obtain a good viewing point, which is taken by the surviving searchers through the fitness selection rule. It is observed through our numerical experiments that the viewpoints for restarting initial points are rather effective.

Results and discussions

Local minima, local maxima and a global minimum

As mentioned in the previous section, MS-FOM makes it possible to discover a set of local minimizers. In fact, MS-FOM found at least four local minimizers A, B, C and D. However, MS-FOM does not guarantee non-existence of local minimizers apart from them. The values of the cost function of these local minimizers are 0.01039, 0.02800, 0.01758 and 0.02042, respectively. The local minimum A is the smallest among them and is concluded to be the global minimum. Figures from 2 to 5 show the sound fields corresponding to these local minimizers, respectively. Obviously, they correspond to the different shapes of the interfacial boundary.

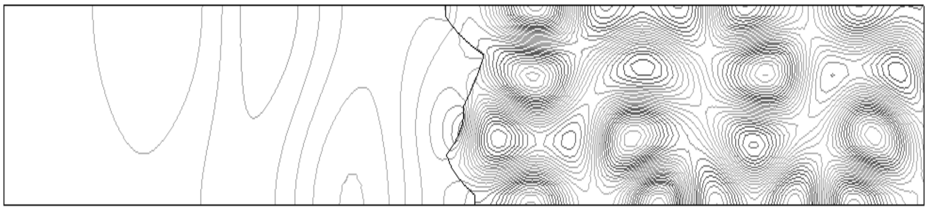


Figure 2: Real part of sound pressure corresponding to local minimizer A

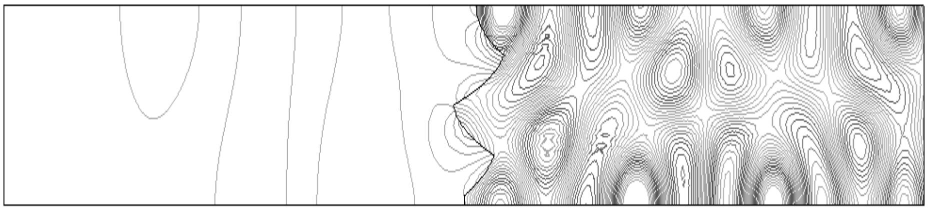


Figure 3: Real part of sound pressure corresponding to local minimizer B

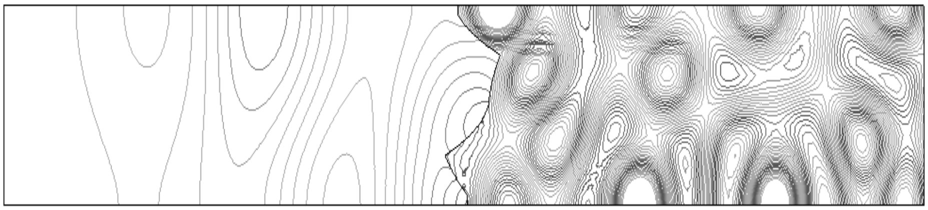


Figure 4: Real part of sound pressure corresponding to local minimizer C

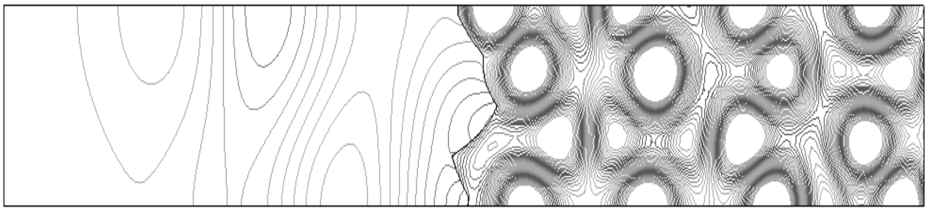


Figure 5: Real part of sound pressure corresponding to local minimizer D

Perspective of the cost manifold

In order to characterize the cost manifold, we draw some one-dimensional cross-sections of the cost manifold. Each one-dimensional cross section is a straight line in the 18-dimensional parameter space connecting two local minimizers. Concretely, figure 6 shows the values of the cost function on a straight line connecting local minimizers A and B, where A is the global minimizer. In this figure, 0 and 1 on the horizontal axis correspond to the local minimizers A and B, respectively. Figure 7 and 8 show the values of the cost function on straight lines connecting B and C, and B and D, respectively. We can see from these figures that the cost manifold has a lot of local minimizers and maximizers. Furthermore, we conjecture from these figures that the cost manifold originally forms convex envelopes and expect that the global minimizer concluded in our computations seems to be a reliable global minimizer. Then, what physical meanings do local maximizers have?

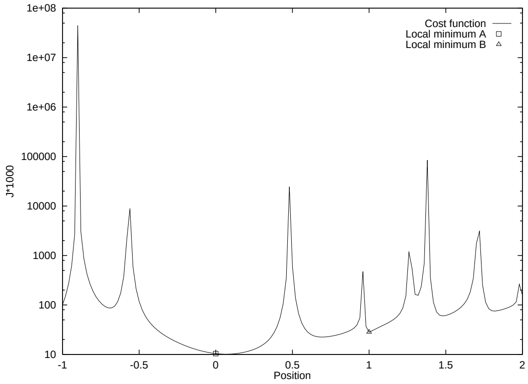


Figure 6: A cross section of the cost manifold connecting local minimizers A and B

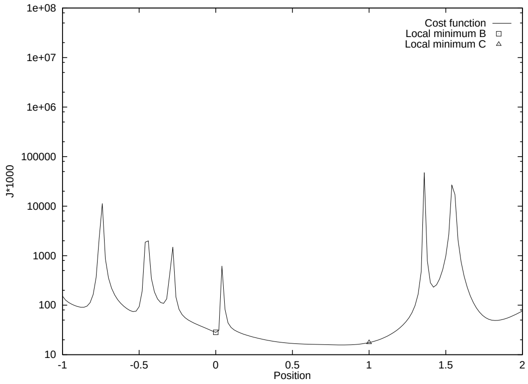


Figure 7: A cross section of the cost manifold connecting local minimizers B and C

A reason of the existence of several local maximizers shown in figures 6, 7 and 8 may be that each local maximizer corresponds to the resonance frequencies of sound

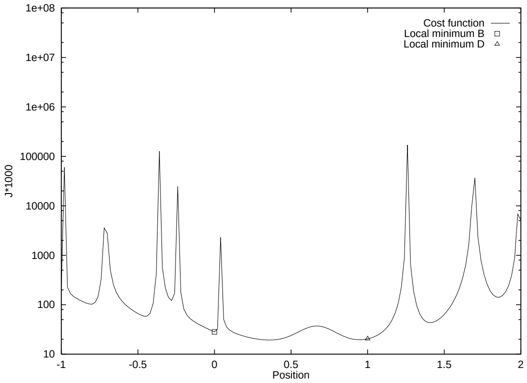


Figure 8: A cross section of the cost manifold connecting local minimizers B and D

propagation in coupled media. The eigenfrequencies of coupled media are sensitive to the interfacial boundary between them. We checked similar phenomena in such a case with a simpler geometry through numerical experiments, in which numerical eigenfrequencies coincided with theoretical ones. In order to provide evidence for such a conjecture, computations of eigenfrequencies to the domain with the related interfacial boundary remain.

Finally, it should be emphasized that MS-FOM has not found out all local minimizers but some of them, however, it counted up the local minimizers very efficiently and it was able to find out the reliable global minimizer. This fact shows the usefulness of our search strategy such as *repeating up-down procedures* and *rearrangement of starting points* mentioned in the previous section, which makes it possible to investigate the perspective of the cost manifold.

Conclusions

The shape design of the interfacial boundary in order to minimize the amplitude of a reflected wave was discussed by use of an algorithm based on MS-FOM. Since the optimization problem with respect to sound propagation includes resonance structure, the cost manifold has very complicated shapes. The numerical results show that the algorithm works well for such problems by avoiding the influence of resonance phenomena.

References

[KOPS98]H. Kawarada, T. Ohtomo, J. Periaux, and H. Suito. Multi-start fuzzy optimization method. *GAKUTO International Series, Mathematical Sciences and Applications*, 11:338–345, 1998.

[KS97]H. Kawarada and H. Suito. Fuzzy optimization method. In *Computational Science for the 21st Century*, pages 642–651. John Wiley & Sons, 1997.

[KS99]Hideo Kawarada and Hiroshi Suito. Optimal shape of pine for sound absorption in water. In C.-H. Lai, P.E. Bjorstad, M. Cross, and O. Widlund, editors, *Proceedings of 11th International Conference on Domain Decomposition Methods*, pages 270–281. DDM.org, 1999.

[Zha97]S.-L. Zhang. GPBi-CG : Generalized product-type methods based on Bi-CG for solving nonsymmetric linear systems. *SIAM Journal on Scientific Computing*, 18(2):537–551, 1997.