VŠB-Technical University Ostrava

Faculty of Electrical Engineering and Computer Science

Department of applied mathematics

FETI based domain decomposition methods for variational inequalities

Ph.D. Thesis

January 2007 Ing. David Horák

Acknowledgments

I would like to thank first of all to my supervisor Prof. Zdeněk Dostál, he has entrusted theme of domain decomposition methods for variational inequalities to me, for his help during the solution of this problem and for everything he has taugth me during all my studies and I hope he will still teach.

The successful realization was possible thanks to Prof. Charbel Farhat, Michel Lesoinne, Dan Stefanica, Radek Tezaur and not least to my colleagues from my department and friends, especially to Doc. Nina Častová, who has attracted me with my supervisor for applied mathematics, and to Vít Vondrák, whom I had opportunity to work on implementations with.

My work and research during my Ph.D. studies was also supported by many grants: NSF - Kontakt ME 641, GACR - 101/04/1145 and HPC-Europa project (RII3-CT-2003-506079) with support of the European Community - Research Infrastructure Action under FP6.

I thank very much to my fiancee Ivanka for her love and help - because of her the depression (described in section Introduction) was uknown for me during all my work, to my mother and father that they have educated me as they have done, to my sisters and grandparents, to the whole Czech and Slovak family for their love and support.

Rychvald, 13th January 2007

Contents

List of Figures

List of Tables

Notation

symbol	meaning
$a(u,v)$	symmetric bilinear form
$l(v)$	linear form
u	solution - displacements
u_c	global vector of corners
f	density of forces
λ	vector of Lagrange multipliers
K	stiffness matrix
B	matrix with constraints
F	dual operator
F^{-1}	dual preconditioner
R	null space of matrix K
g	gradient
Ω^i	domain or subdomain
$\Gamma^{i,j}$	boundary
$\mathbb{R}^n$	n-th dimensional real space
$H^1(\Omega)$	Sobolev space
$J(u)$	primal energy functional
$\theta(\lambda)$	dual energy functional
$\|z\|_{l^2}$	Euclidian norm of vector z
C	generic constant
H	subdomain, decomposition parameter
h	mesh, discretization parameter

Applied abbreviations:

QP	quadratic programming
CG	conjugate gradient
FEM	finite element method
PDEs	partial differential equations
DD	domain decomposition
s.t.	subject to
dof	degree of freedom

Part I

Chapter 1

Introduction

"Konáme-li poprvé větší výpočty, jaké se vyskytují zvláště v astronomii, vyšší geodesii, mechanice těles nebeských i jinde, docházíme zpravidla k chybným výsledkům, poněvadž nemáme cviku v numerickém počítání. Opakujíce takový výpočet dopouštíme se často nejen týchž omylů, nýbrž i nových, neboť duševní deprese, která se dostavuje v důsledku učiněných omylů, stává se zdrojem omylů nových. Omyly toho druhu možno omeziti delším cvikem v numerickém počítání, jednak užíváním početních tabulek a strojů, které se staly v novější době nezbytnou pomůckou počtářskou. Další zdroje početních omylů jsou: neuspořádanost výpočtů, nedbalé psaní číslic, přílišný spěch a malá pozornost při provádění výpočtů. Jest proto zapotřebí, abychom si navykli počítati zvolna, pozorně, psáti číslice výrazně a náležitě pod sebou. Dobrý počtář napíše asi 40 číslic za minutu, čili názorněji řečeno, násobení obvyklým způsobem pětimístného čísla pětimístným číslem trvá průměrně jednu minutu. Známý počtář Dase potřeboval k vypočítání součinu dvou stomístných čísel skorem 9 hodin. Počítání strojem jest pro rutinovaného počtáře ovšem značně rychlejší a pohodlnější a průměrně nevyžaduje více než 1/5 až 1/4 času potřebného k témuž výpočtu obyčejným způsobem."

"If we perform large calculations occuring especially in astronomy, geonics, mechanics and elsewhere, we get as a rule bad results, because we are not appropriate practised in numerical reckonning. Sources of arithmetical faults are: confusion of calculations, negligent writing of figures, large haste and small heed during execution the calculations. That's why it is necessary to get used to reckon steadily, carefully, to write figures conspicously and one below another. A good reckonner writes about 40 figures per minute, otherwise the multiplication of 5-figured number by 5-figured one takes him about 1 minute in usually way. The famous reckonner Dase needed for multiplication of two 100-figured numbers about 9 hours. The calculation by a machine is for skilled reckonner faster and more comfortable and takes him about 1/4 or 1/5 of time needed for the same calculation in common way."

This is a part of Introduction of the charming Czech book Theory and Practice of Numerical Reckonning [1] from 1927. Since then mathematics and computing made enormous progress that would not be possible without our ancestors' fundamentals. This development is never-ending process, although each age has its charm, positives and negatives. Mathematics and computer science stimulate each other and their interaction produces synergistic effect.

For the numerical solution of partial differential equations (PDEs) that are encountered in science and engineering, it is necessary to discretize them - this is the first step. The second one is to solve the resulting system of linear equations, that with an effort to approximate the reality reaches the huge dimensions. These large scale problems can be solved by means of powerful computers. Development of parallel computers started boom of algorithms and set also new criteria for assesment of their performance. An algoritm which is efficient on sequential computer can be inefficient in parallel case and opposite. An effort to implement the solution of these equations on parallel computers leads to the domain decomposition.

Domain decomposition methods are powerful iterative methods for solving systems of algebraic equations arising from the discretization. The main idea is that the problem is divided into smaller subproblems corresponding to subdomains arising from decomposition of computational domain, each of processors then works with data corresponding to local subdomains with the minimum of interprocessor communication. The speed up due to parallel implementation is nearly proportional to the number of processors. This nice feature - called parallel scalability - can be eliminated by decreasing of discretization parameter's size resulting in increase of number of iterations. Thus we face a new challenge to design and to develop algorithms computing problems independently of this discretization parameter keeping the ratio of the subdomain and discretization parameteres constant. This feature is called as numerical scalability, the cost of solution is proportional to the number of nodal parameters. Let us recall that an algorithm is called scalable for a given class of problems if it has both, parallel and numerical scalabilities.

The FETI (Finite Element Tearing and Interconnecting) method turned out to be one of the most successful algorithms for parallel solution of problems described by elliptic PDEs. FETI methods are based on the decomposition of the spatial domain into non-overlapping subdomains glued by Lagrange multipliers. Theoretical results and numerical experiments establish scalability of variants of FETI algorithm for linear problems. This thesis concerns efficient implementations of the scalable FETI-based methods for variational inequalities.

Ph.D. thesis is devided in two parts. Part I (Chapters 1 - 6) is devoted to an overview of standard methods, discretization techniques, etc. and is organized as follows:

- Chapter 2 describes two model contact problems. I use them in my thesis to simplify the explanation and to test the performance of our algorithms in numerical experiments (coercive and semicoercive).

- Chapter 3 presents continuous formulation of the model problem.
- Chapter 4 explains discretization, domain decomposition, interface and boundary condition techniques, and primal problem formulation.
- Chapter 5 describes the algorithms used for the solution of constraint QP problems.
- Chapter 6 contains an overview of FETI methods and recalls their principles.

Part II (Chapters 7 - 14) contains new author's results and remarks, and the results on which the author participated as a member of teams. These results including the parallel implementation and many numerical experiments were published in 13 impact papers and in 33 conference papers. Many of these results were obtained in cooperation of our Department of Applied Mathematics with Prof. Charbel Farhat - the inventor of FETI methods, Radek Tezaur (both Stanford University California), Prof. Michel Lesoinne, Prof. Jan Mandel (Boulder University of Colorado, Department of aerospace engineering), Prof. Dan Stefanica (City University of New York and MIT), Prof. Ulrich Langer (Johannes Kepler University Linz), Ediburgh Parallel Computing Centre (EPCC), etc.

Following chapters summarize parallel implementations of algorithms, numerical experiments and their importance for developping the theoretical results:

- Chapter 7 deals with optimality of the dual penalty method and its importance for numerical scalability of algorithms. Numerical experiments gave an impulse to start the research of the scalability of FETI based algorithms for variational inequalities and to develop the theory.
- Chapter 10 describes the implementation of new TFETI-1 (Total FETI-1) method designed by Prof. Dostál. Its nice properties were confirmed in my numerical experiments in 2D and in joint implementation in 3D elasticity with Vít Vondrák and Prof. Farhat's group during our stay at Stanford University.
- Chapter 12 describes the important parts of implementations of algorithms, especially the parallel implementation carried out during my stay in Edinburgh Parallel Computing Centre.
- Chapter 13 presents many numerical experiments for all presented methods for contact problems computed on parallel computers in Ostrava, Linz, Boulder, Stanford and Edinburgh.
- Chapter 14 concludes all the work and gives ideas for the future work.

The most important theoretical results are in Chapters 8,9 and 11:

- Chapter 8 describes modification of FETI-DP for corners on the contact zone for both coercive and semicoercive cases developed during my stay at Boulder University, Department of Aerospace Engineering. As mentioned, Prof. Farhat, father of FETI methods, avoided corners on the contact zone in his many papers because of jumps and penetration in these corners. I have derived the additional condition that preserves the nonpenetration in Lagrange multipliers and enables the usage of corners on contact interface.

- Chapter 9 introduces my important observation, which is the key ingredient for proof of numerical scalability of FETI-DP method for contact problems with mortars. Norm of each row of mortar constraint matrix is of the order of the corresponding mesh size, while every row of the nodal constraint matrix has norm of order 1. It is important to use constrained matrices with normalized rows in algorithms to get significantly smaller upper bound on the condition number of the dual operator matrix. This is the joint work with Dan Stefanica from City University of New York.

- Chapter 11 introduces new generation FETI method for linear and contact extreme large scale problems - so called MFETI-DP (Multilevel FETI-DP) - proposed by Prof. Farhat. The size of coarse problem increases due to increasing number of subdomains, increasing degree of parrallelism, reducing memory requirements, reducing CPU for factors and solution of coarse problem by forward/backward substitution. The basic idea is a natural application of multigrid methods, i.e. recursive application of FETI-DP to the coarse problem. The implementation of FETI-DP algorithm for variational inequalities is complicated, but much more complicated is that of MFETI-DP. The implementation of this method was carried out during my stay in Boulder, including the numerical experiments with comparison of finite and iterative solution of coarse problems and inexact solution of coarse problems on the second level with CG method in simple V-cycle for both, linear and contact problems.

Chapter 2

Model contact problems and domain decomposition

In this chapter two variational inequality model problems are introduced, which shall simplify the exposition and represent problems for numerical experiments (see Figure 2.1).

These problems arising from variational formulation of problem with inequality boundary conditions read: find sufficiently smooth $u(x,y)$ so that

$$-\triangle u = f \quad \text{in} \quad \Omega = \Omega^1 \cup \Omega^2 \tag{2.1}$$

$$u^i = 0 \quad \text{on} \quad \Gamma_u^i,\ i = 1,2 \tag{2.2}$$

$$\frac{\partial u^i}{\partial n_i} = 0 \quad \text{on} \quad \Gamma_f^i,\ i = 1,2 \tag{2.3}$$

$$u^2 - u^1 \geq 0 \quad \text{on} \quad \Gamma_c = \Gamma_c^1 = \Gamma_c^2 \tag{2.4}$$

$$\frac{\partial u^2}{\partial n_2} \geq 0 \quad \text{on} \quad \Gamma_c \tag{2.5}$$

$$\frac{\partial u^2}{\partial n_2}(u^2 - u^1) = 0 \quad \text{on} \quad \Gamma_c \tag{2.6}$$

$$\frac{\partial u^1}{\partial n_1} + \frac{\partial u^2}{\partial n_2} = 0 \quad \text{on} \quad \Gamma_c \tag{2.7}$$

here $\Omega^1 = (0,1) \times (0,1), \Omega^2 = (1,2) \times (0,1)$, denote open domains with boundaries Γ^1, Γ^2 and their parts $\Gamma_u^1, \Gamma_f^i, \Gamma_c^i$, formed by the sides of $\Omega^i, i = 1,2$, and $n_k(x,y)$ denote components of the outer unit normal at $(x,y) \in \Gamma^i$. For the semicoercive problem $\Gamma_u^2 = \emptyset$.

The solution $u(x,y)$ can be interpreted as a vertical displacement of two membranes stretched by normalized horizontal forces and pressed together by vertical forces with density $f(x,y)$. Relation (2.1) is elliptic equation, (2.2) is boundary condition of Dirichlet type, (2.3) is boundary condition of Neumann type, (2.4) describes the nonpenetration of the adjacent edges of the membranes, with the edge of the right membrane

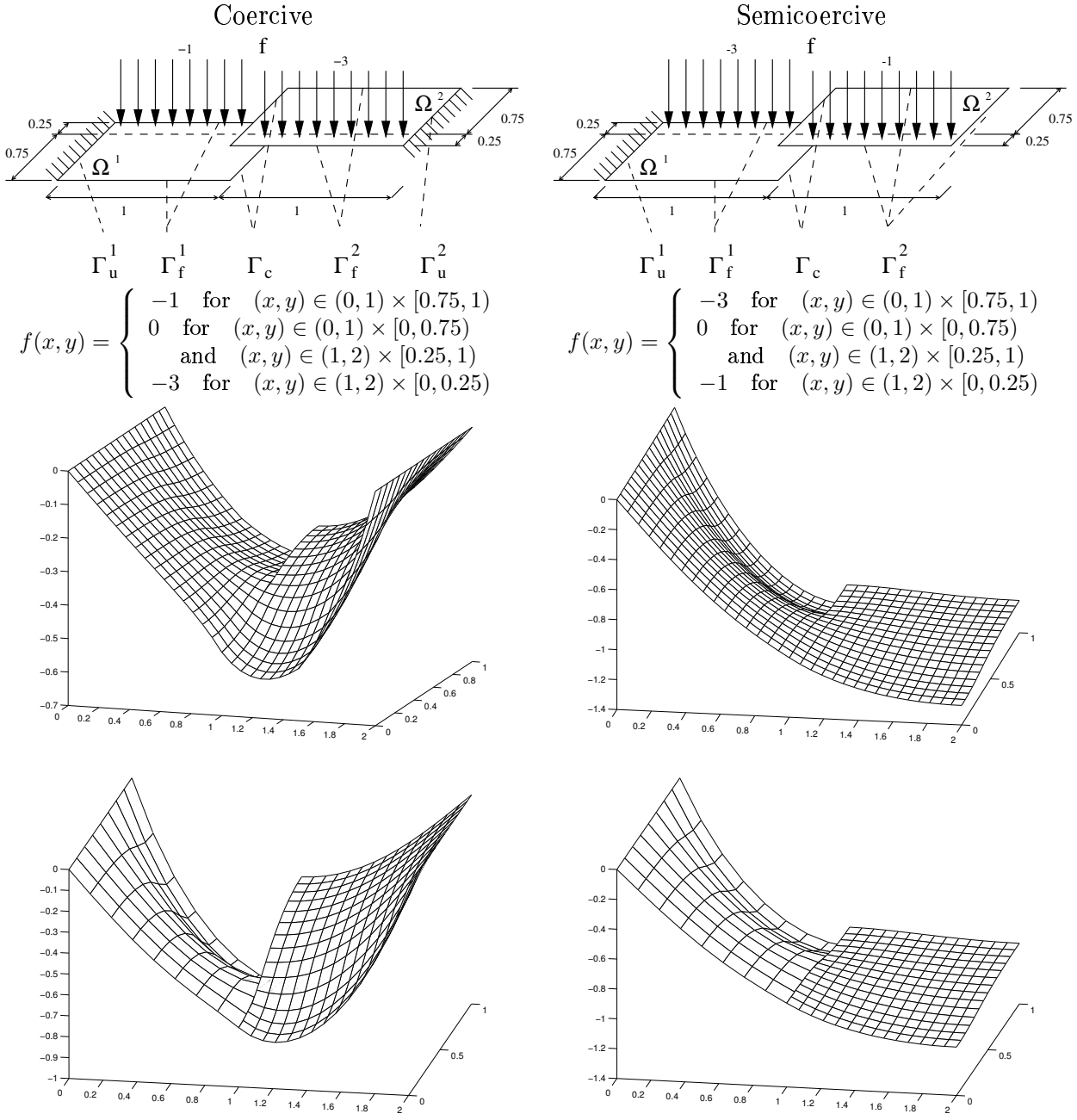


Figure 2.1: Two model problems - coercive and semicoercive, solutions obtained by conforming and nonconforming FE discretizations

above the edge of the left membrane, (2.5) expresses, that right membrane can press the left one down at the contact points, (2.6)-(2.7) define points, that are in contact - if there is no contact at $(x, y) \in \Gamma_c$, i.e. $u^2(x, y) > u^1(x, y)$, then the membranes are stretched by the horizontal force in the same way as at $(x, y) \in \Gamma_f^i$.

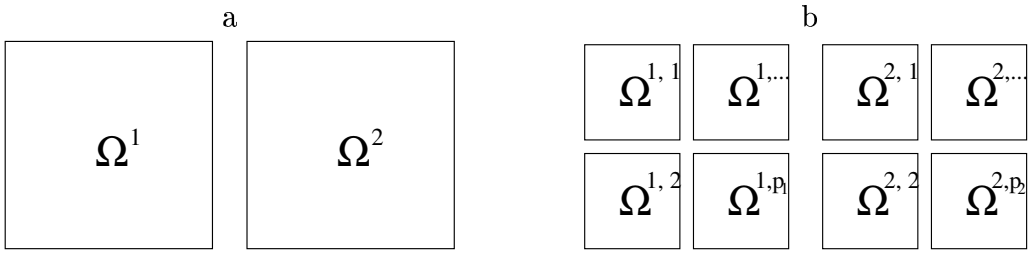


Figure 2.2: a) Model problem of two bodies, b) the decomposed problem into subdomains

As mentioned, the most effective method for solution is based on an auxiliary decomposition of these two domains into subdomains, i.e. decomposition of each domain Ω^i into p_i subdomains $\Omega^{i,j}, j = 1, ..., p_i$, looking for the vertical displacements of subdomains satisfying above described conditions including additional gluing conditions on auxiliary interfaces, as illustrated in Figure 2.2. The decomposed solutions are in Figure 2.3 and more details are described in following chapters. The efficiency lies on the fact, that each processor of the parallel computer deals with data associated with separated subdomains with minimum of interprocessor communication, more details are discussed in Section 12.2.

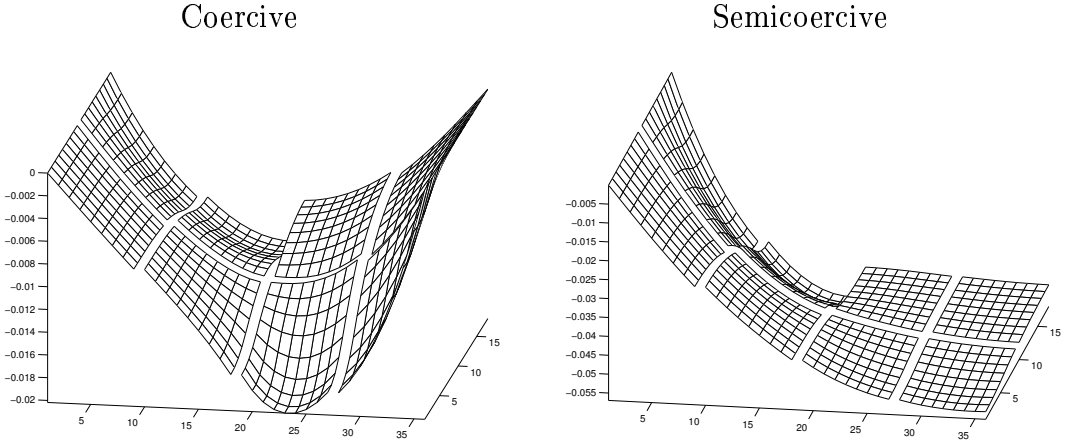


Figure 2.3: The solutions of decomposed model problems using conforming FE

Chapter 3

Continuous formulations

3.1 Continuous formulation of model problem

To derive the variational inequality, whose smooth solution satisfies (2.1)-(2.7), let $H^1(\Omega^i)$ denote Sobolev space of first order on the space $L^2(\Omega^i)$ of the functions on Ω^i, whose squares are integrable in the Lebesgue sence. Then $f \in H^1(\Omega^i)$ iff both f and its first derivates belong to $L^2(\Omega^i)$.

Lets define following requisities: $V^i = \{v \in H^1(\Omega^i) : v^i = 0 \text{ on } \Gamma_u^i\} \subseteq H^1(\Omega^i),\ i = 1, 2$ denotes the closed subspace of $H^1(\Omega^i)$, $V = V^1 \times V^2$ denotes the closed subspace and $\mathcal{K} = \{(v^1, v^2) \in V : v^2 - v^1 \geq 0 \text{ on } \Gamma_c\}$ closed convex subset of $\mathcal{H} = H^1(\Omega^1) \times H^1(\Omega^2)$. On $\mathcal{H}$ lets define a symmetric bilinear form

$$a(u, v) = \sum_{i=1}^{2} \int_{\Omega^i} \left(\frac{\partial u^i}{\partial x} \frac{\partial v^i}{\partial x} + \frac{\partial u^i}{\partial y} \frac{\partial v^i}{\partial y} \right) d\Omega$$

and a linear form

$$\ell(v) = \sum_{i=1}^{2} \int_{\Omega^i} f^i v^i \, d\Omega.$$

Let u denote a smooth solution of (2.1)-(2.7). After multiplication of (2.1) by $v \in V$ and application of the Green theorem and using relations (2.2)-(2.3) we obtain

$$a(u, v) - \ell(v) = \int_{\Gamma_c} \left\{ \frac{\partial u^1}{\partial n_1} v^1 + \frac{\partial u^2}{\partial n_2} v^2 \right\} d\Gamma$$

and for $v = w - u$, $w \in \mathcal{K}$

$$a(u, w - u) - \ell(w - u) = \int_{\Gamma_c} \left\{ \frac{\partial u^1}{\partial n_1} (w^1 - u^1) + \frac{\partial u^2}{\partial n_2} (w^2 - u^2) \right\} d\Gamma. \tag{3.1}$$

At the points of Γ_c with $u^1 < u^2$ there is , due to (2.6)-(2.7)

$$\frac{\partial u^1}{\partial n_1} = \frac{\partial u^2}{\partial n_2} = 0, \tag{3.2}$$

so that the integrand in (3.1) vanishes at such points. At the points of Γ_c with $u^1 = u^2$ there is , due to (2.5) and (2.7)

$$\frac{\partial u^1}{\partial n_1}(w^1 - u^1) + \frac{\partial u^2}{\partial n_2}(w^2 - u^2) = \frac{\partial u^1}{\partial n_1}w^1 + \frac{\partial u^2}{\partial n_2}w^2 = \frac{\partial u^2}{\partial n_2}(w^2 - w^1) \geq 0. \tag{3.3}$$

The integral in (3.1) is nonnegative for any $w \in \mathcal{K}$ and the solution u of problem defined by relations (2.1)-(2.7) solves also problem defined in following way

$$\text{find } u \in \mathcal{K} \text{ so that } a(u, w-u) - \ell(w-u) \geq 0 \ , \forall w \in \mathcal{K} \tag{3.4}$$

and opposite [3].

The expression on the left of inequality in relation (3.4) is gradient of the energy functional

$$J(v) = \frac{1}{2}a(v,v) - \ell(v)$$

at u, and that's why the problem given by (3.4) is equivalent to the minimization problem

$$\min \ J(v) \ \text{ s.t. } \ v \in \mathcal{K}. \tag{3.5}$$

It can be proved, that functional $J(v)$ is convex and coercive or semicoercive on $\mathcal{K}$, and theorem for existence and uniqueness [4] of minimum for such functionals then guarantees, that problems (3.4)-(3.5) have unique solution if f satisfies $\int_{\Omega^2} f \, d\Omega < 0$, what is assumed in all following expositions.

3.2 Continuos formulation of decomposed model problem

To derive the variational inequality now, we follow analogously previous approach, where $H^1(\Omega^{i,j})$ denotes Sobolev space of first order on the space $L^2(\Omega^{i,j})$, $\Gamma^{i,j} = \partial\Omega^{i,j}$, $V^{i,j} = \{v \in H^1(\Omega^{i,j}) : v^i = 0 \text{ on } \Gamma_u^i \cap \Gamma^{i,j}\} \subseteq H^1(\Omega^{i,j})$, $i = 1,2$, $j = 1, ..., p_i$, $V^i = V^{i,1} \times ... \times V^{i,p_i}$, $i = 1,2$, $V = V^1 \times V^2$ and $\mathcal{K}_E = \left\{(v^1, v^2) \in V : v^{i,k} - v^{i,j} = 0 \text{ on } \Gamma^{i,k} \cap \Gamma^{i,j}\right\}$, $\mathcal{K}_I = \left\{(v^1, v^2) \in V : v^{2,k} - v^{1,j} \geq 0 \text{ on } \Gamma_c \cap \Gamma^{i,k} \cap \Gamma^{i,j}\right\}$, $f^{i,j} = f \mid_{\Omega^{i,j}}$, $u^{i,j} = u \mid_{\Omega^{i,j}}$, $i = 1,2$, $j,k = 1, ..., p_i$, $\mathcal{K} = \mathcal{K}_I \cap \mathcal{K}_E$ closed convex subset of $\mathcal{H} = H^1(\Omega^{1,1}) \times ... \times H^1(\Omega^{1,p_1}) \times H^1(\Omega^{2,1}) \times ... \times H^1(\Omega^{2,p_2})$. On $\mathcal{H}$ lets define a symmetric bilinear form

$$a(u,v) = \sum_{i=1}^{2} \sum_{j=1}^{p_i} \int_{\Omega^{i,j}} \left(\frac{\partial u^{i,j}}{\partial x}\frac{\partial v^{i,j}}{\partial x} + \frac{\partial u^{i,j}}{\partial y}\frac{\partial v^{i,j}}{\partial y} \right) d\Omega$$

and a linear form

$$\ell(v) = \sum_{i=1}^{2} \sum_{j=1}^{p_i} \int_{\Omega^{i,j}} f^{i,j} v^{i,j} \, d\Omega.$$

Following the same procedure as in previous section we get in the end the minimization problem

$$\min \ J(v) \ \text{ s.t. } \ v \in \mathcal{K}, \ \text{ with } J(v) = \frac{1}{2}a(v,v) - \ell(v).$$

Chapter 4

Discretization, boundary and interface conditions

4.1 Discretization

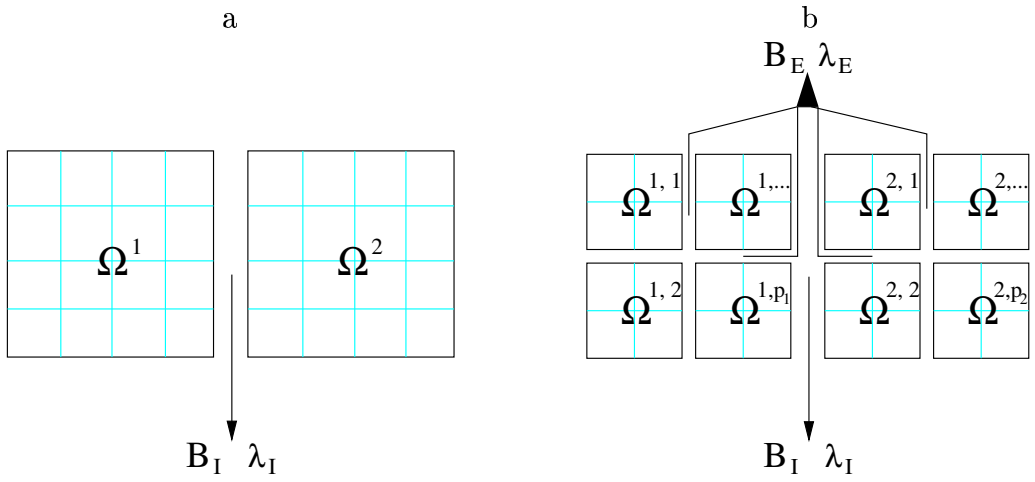


Figure 4.1: a) Discretized model problem of two bodies, b) the decomposed problem into subdomains

4.1.1 Discretization of model problem

For the numerical processing of this problem it is necessary to execute the linearization and finite element discretization. To this purpose let's define (η_h, τ_h) a partitioning of Ω into triangles $T_j \in \tau_h$, with suitable numbering of vertices at $N_k \in \eta_h$.

For $i = 1, 2$ (see Figure 4.1a) let P_h^i denote the piecewise linear finite element subspaces of $H^1(\Omega^i)$, let $V_h^i = P_h^i \cap V^i$, and define $V_h = V_h^1 \times V_h^2$ and $\mathcal{K}_h = \mathcal{K} \cap V_h$. The

problem (3.5) is then aproximated by the finite element problem

$$\min\ J(v_h)\ \text{ s.t. }\ v_h \in \mathcal{K}_h. \tag{4.1}$$

The functions $p_h^i \in P_h^i$ are fully determined by values $u_k^i = p_h^i(N_k^i)$ at nodes $N_k^i \in \overline{\Omega^i}$, while $\overline{\Omega^i}$ denotes a closure of Ω^i. Assuming the independent indexing of nodes of $\Omega^i \setminus \Gamma_u$ by indices $1, 2, ..., s_i$ and denoting of standard basis functions of V_k^i by e_k^i, so that $e_k^i(N_j^i) = \delta_{kj}$ (Kronecker symbol), it is possible to write any $v_h^i \in V_h^i$ in following form

$$v_h^i = \sum_{k=1}^{s_i} u_k^i e_k^i. \tag{4.2}$$

Substituting (4.2) into expression for functional $J(v)$ gives

$$J(v_h) = \frac{1}{2} u^T K u - f^T u,$$

where

$$K = \begin{bmatrix} K^1 & 0 \\ 0 & K^2 \end{bmatrix},\ f = \begin{bmatrix} f^1 \\ f^2 \end{bmatrix},\ u = \begin{bmatrix} u^1 \\ u^2 \end{bmatrix},$$

K is symmetric positive semidefinite block-diagonal stiffness matrix of order n (symmetric means that $a_{ij} = a_{ji}$ for $i, j = 1, ..., n$, while positive semidefinite means that $u^T K u \geq 0$ for all vectors u), f vector of nodal forces of size n and u vector of nodal unknowns of size n (note: in previous sections the symbols u, f denote continuous functions, from this place on, because of simplicity, u, f represent the vectors), with $K^i = [a_{jk}^i]$, $a_{jk}^i = a(e_j^i, e_k^i)$, $f^i = [f_j^i]$, $f_j^i = \ell(e_j^i)$ and $u^i = [u_j^i]$.

4.1.2 Discretization of decomposed model problem

Above presented exposition takes into account just the decomposition in two subdomains Ω^1 and Ω^2, but it is possible to decompose auxiliary each domain Ω^i, $i = 1, 2$ into p_i subdomains $\Omega^{i,j}, j = 1, ..., p_i$ (see Figure 4.1b) , so that each of these subdomains is partitioned by a subset of (η_h, τ_h). After using the finite element discretization of problem (3.5) with the basis functions, that are zero extensions of $P_h^{i,j} \subset H^1(\Omega^{i,j})$ for $i = 1, 2$ and $j = 1, ..., p_i$, the functional has for all piecewise linear functions v_h continuous in subdomains $\Omega^{i,j}$ the form

$$J(v_h) = \frac{1}{2} u^T K u - f^T u,$$

where

$$K = \begin{bmatrix} K^{1,1} & 0 & \dots & \dots & \dots & 0 \\ 0 & \ddots & 0 & \dots & \dots & 0 \\ 0 & 0 & K^{1,p_1} & 0 & \dots & 0 \\ 0 & \dots & 0 & K^{2,1} & 0 & 0 \\ 0 & \dots & \dots & 0 & \ddots & 0 \\ 0 & \dots & \dots & \dots & 0 & K^{2,p_2} \end{bmatrix},\ f = \begin{bmatrix} f^{1,1} \\ \vdots \\ f^{1,p_1} \\ f^{2,1} \\ \vdots \\ f^{2,p_2} \end{bmatrix},\ u = \begin{bmatrix} u^{1,1} \\ \vdots \\ u^{1,p_1} \\ u^{2,1} \\ \vdots \\ u^{2,p_2} \end{bmatrix}$$

have the same interpretation as above.

4.2 Interface conditions

For completition of discretizaton it is necessary to prescribe conditions on u_k^i. Let full rank matrices B_I and B_E describe the discretized inequality (nonpenetration) and equality (gluing) conditions, respectively.

4.2.1 Conforming-nodal FE

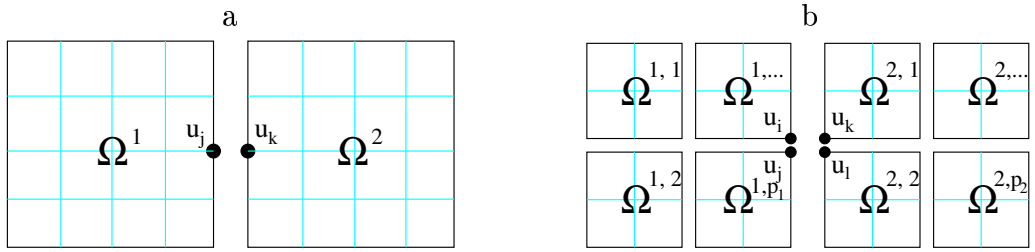


Figure 4.2: Nonpenetration conditions for a) couple of nodes b) four corner nodes

Nodal (conforming) nonpenetration conditions

$$u_j - u_k \leq 0 \text{ or } u_i + u_j - u_k - u_l \leq 0$$

can be substituted by exprression $B_I u \leq 0$, with full rank matrix B_I consisting of orthogonal rows

$$\left[\cdots \quad 1 \quad \cdots \quad -1 \quad \cdots \right] \text{ or } \left[\cdots \quad 1 \quad \cdots \quad 1 \quad \cdots \quad -1 \quad \cdots \quad -1 \quad \cdots \right].$$

Continuity conditions are enforced in similar way as inequality conditions replacing inequality with equality sign $B_E u = 0$. More details about construction of B-matrices and mapping Lagrange multipliers on the subdomains (non-redundant, redundant, orthonormal) are presented in [12].

4.2.2 Nonconforming-mortar FE

Another approach of prescribtion of constraints is usage of mortar finite elements. Mortar finite elements are non-conforming finite elements that allow geometrically nonconforming decomposition of domain into subdomains and their independent discretizations and have advantages as flexible mesh generation, easy local refinement, good parallelization properties. Mortars were first introduced by Bernardi, Maday and Patera in [32] for low order and spectral finite elements, 3D version was developed by Ben Belgacem and Maday [33]. Family of biorthogonal mortar elements was designed by Wohlmuth [34], [35], [36]. The mortar wavelet method was introduced by Bertoluzza, Perrier [37]. Cai, Dryja, and Sarkis [41, 39] extended mortars to overlapping decompositions. Achdou,

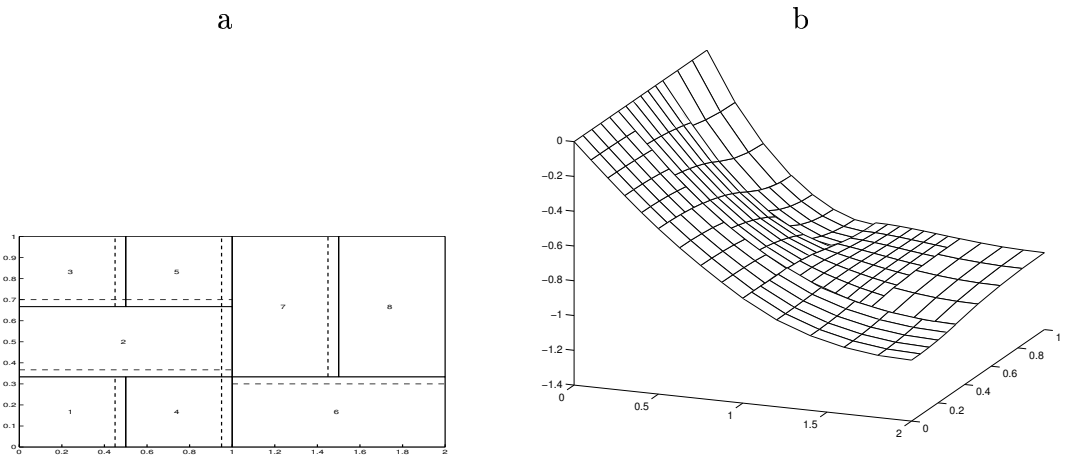


Figure 4.3: a) Example of nonconforming DD b) Solution of semicoecive problem for nonconforming meshes $f(x,y) = -2$, for $(x,y) \in (0,1) \times [0.75,1)$

Maday, Widlund [40] demonstrated similar behaviour of iterative substructuring methods with mortars as those with conforming finite elements, Dryja [41] and Le Tallec, Sassi and Vidrascu [42] for Neumann-Neumann algorithms, and Stefanica, Klawonn [43], [38] for FETI methods.

In 2D case let suppose decomposition in subdomains $\Omega^{i,j}, i = 1,2,\ j = 1,...,p_i$, Γ let denote interface among subdomains defined as the set of points that belong to the boundaries of at least two subdomains. The mortar finite element space V_h contains mortar functions v that vanish at all nodes on Γ_u, and restriction of v to $\Omega^{i,j}$ is a P_1 or Q_1 finite element function. It is neccessary to decompose the subdomain interface Γ into disjoint set of open edges called nonmortar sides (γ_m, dashed line in Figure 4.3a). The edges forming Γ and not belonging to nonmortars, are called mortars ($\widetilde{\gamma_m}$). We impose weak continuity and nonpenetration for v across each nonmortar side γ_m given by the mortar conditions

$$\int_{\gamma_m} \left(v \mid_{\gamma_m} - v \mid_{\bigcup \widetilde{\gamma_m}} \right) \psi \, ds \leq 0, \quad v \in V_h, \psi \in \Psi_h(\gamma_m), \tag{4.3}$$

where $\Psi_h(\gamma_m)$ is Lagrange multiplier space of test functions having the same dimension as the number of interior nodes on γ_m, it is subspace of $V_h(\gamma_m)$, that is restriction of V_h to γ_m, $v \mid_{\gamma_m}$is restriction of v to γ_m and $v \mid_{\bigcup \widetilde{\gamma_m}}$ is restriction of v to $\zeta_m = \bigcup \widetilde{\gamma_m}$, the union of parts of the mortars $\widetilde{\gamma_m}$ that coincides geometrically with γ_m.

Figure 4.4 shows two types of dual basis test functions $\psi \in \Psi_h(\gamma_m)$ - classical (Bernardi, Maday, Patera - continuous, piecewise linear) and biorthogonal (Wohlmuth - discountinuous, piecewise linear, based on observation $V_h(\Omega^i) \subset H^1(\Omega^i)$, $v \mid_{\gamma_m} \in H^{1/2}(\gamma_m)$, so $\Psi_h(\gamma_m)$ may be embedded in the dual space of $H^{1/2}(\gamma_m)$ with respect to the L^2 inner product , and therefore $\Psi_h(\gamma_m) \subset H^{-1/2}(\gamma_m)$, the advantage is nicer matrix structure).

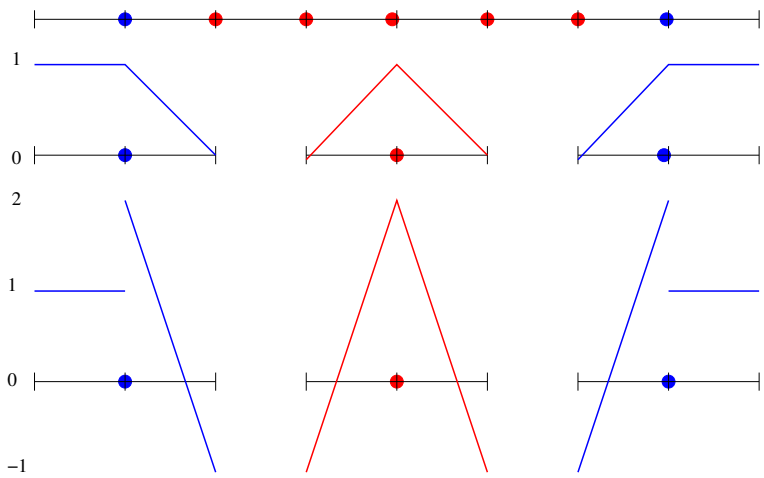


Figure 4.4: 2D interface and test functions - classical mortars and new biorthogonal mortars

Similarly to presented formulations, we obtain saddle point problem: find $(u, \lambda) \in V_h \times \Psi_h$ so that

$$\begin{array}{ccc} a(u,v) + b(v,\lambda) & = \ell(v) & v \in V_h, \\ b(u,\psi) & \leq 0 & \psi \in \Psi_h, \end{array}$$

with $b(v,\psi) = \sum_{m=1}^{M} \int_{\gamma_m} \left(v\,|_{\gamma_m} - v\,|_{\bigcup \widetilde{\gamma_m}} \right) \psi \, ds$ being bilinear form for mortar formulation, $b(u,\psi) \leq 0$ guarantees weak continuity or nonpenetration of u. This problem is equivalent to: find $v \in \mathcal{K}_h = \{v \in V_h : \, b(v,\psi) \leq 0, \, \psi \in \Psi_h\}$ so that

$$a(u,v) = \ell(v),$$

where $a(u,v)$ is symmetric bilinear form uniformly elliptic on $\mathcal{K}_h \times \mathcal{K}_h$. Algebraic formulation of saddle point problem is:

$$\begin{array}{ccc} Ku + B^T\lambda & = f \\ Bu & \leq 0 \end{array}.$$

Discretized problem is the same as in conforming case, so the same algorithms can be applied without changes. B_I and B_E matrices consist of horizontal blocks,

$$B_{\gamma_m} = \left[... M_{\gamma_m} ... - N_{\gamma_m} ... \right],$$

for each nonmortar γ_m enforcing the mortar conditions

$$M_{\gamma_m} u_{\gamma_m} - N_{\gamma_m} u_{\bigcup \widetilde{\gamma_m}} \leq 0$$

across γ_m instead of pointwise conditions in conforming case.

Using orthogonal mortars (classical, new biorthogonal), B_I and B_E matrices consist of horizontal blocks,

$$B_{\gamma_m} = \left[... I_{\gamma_m} ... - P_{\gamma_m} ... \right],$$

for each nonmortar γ_m enforcing the mortar conditions

$$u_{\gamma_m} - P_{\gamma_m} u_{\bigcup \widetilde{\gamma_m}} \leq 0$$

with I_{γ_m} denoting the identity matrix and $P_{\gamma_m} = M_{\gamma_m}^{-1} N_{\gamma_m}$.

For the 3D case, the mortars and nonmortars are open faces forming disjoint partition of the interface Γ. Figure 4.5 shows biorthogonal dual basis test functions developed by Wohlmuth for 3D. Another system of test functions was designed by Kim, Lazarov, Pasciak, Vasilevski [45].

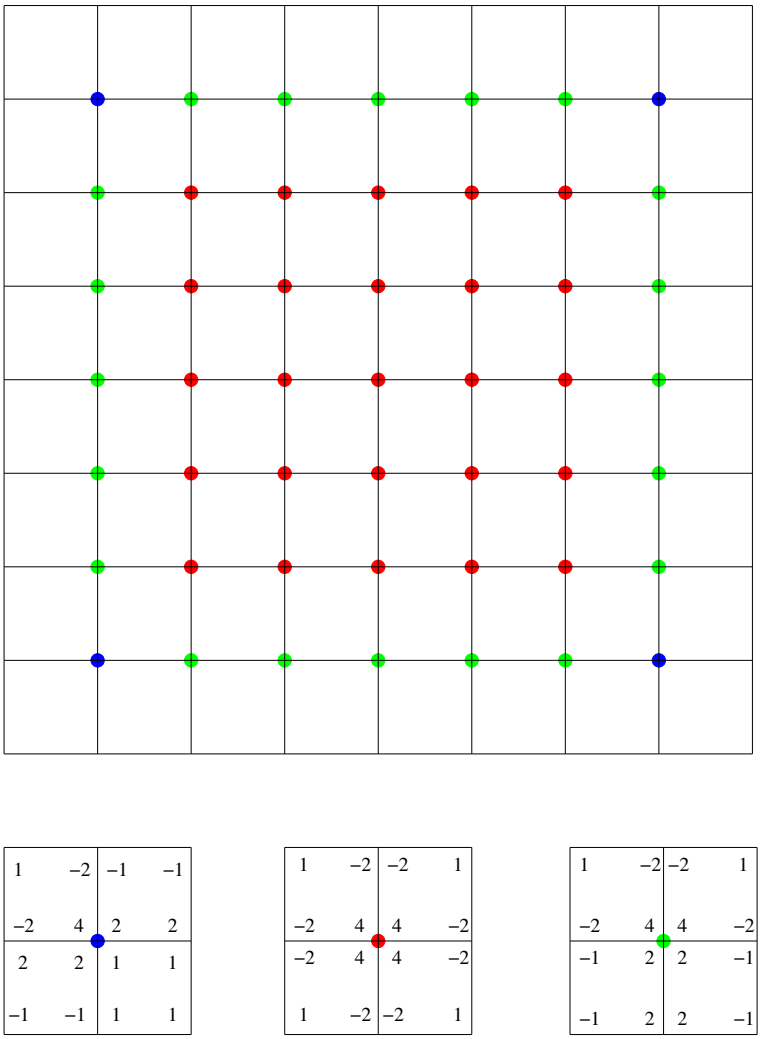


Figure 4.5: 3D interface and piecewise bilinear dual basis test functions - new biorthogonal mortars

4.3 Boundary conditions

The solution is given also by the imposed boundary conditions. There are two possibilities how to prescribe Dirichlet boundary conditions.

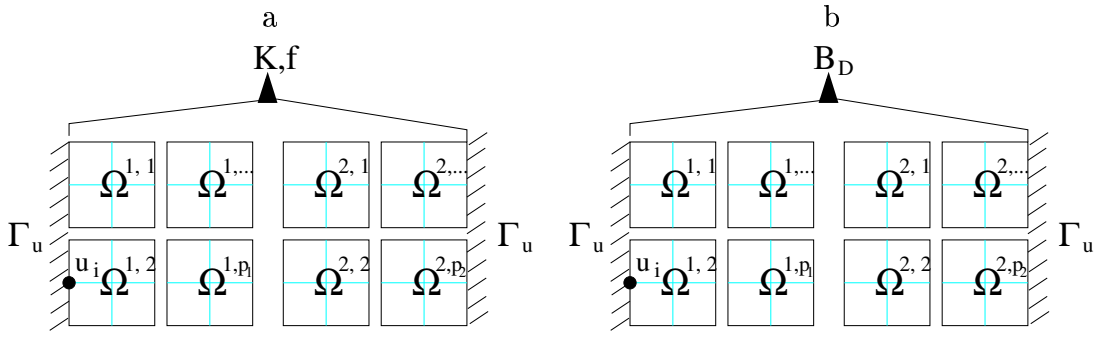


Figure 4.6: Dirichlet boundary conditions for coercive problem incorporated in a) stiffness matrix K and vector of forces f, b) matrix with equality constraints B_D

Dirichlet boundary conditions for nodes u_i on Γ_u

$$u_i = 0$$

can be incorporated into the stiffness matrix K and the vector of nodal forces f in such way, that we zero i-th row and column of K, set diagonal element $K_{i,i} = 1$ and i-th component $f_i = 0$, i.e.

$$K = \begin{bmatrix} & 0 & & & \\ 0 & 1 & 0 & \dots & 0 \\ & 0 & & & \\ & \vdots & & & \\ & 0 & & & \end{bmatrix}, \quad f = \begin{bmatrix} 0 \\ \\ \\ \end{bmatrix},$$

or can be substituted by expression $B_D u = 0$, with full rank matrix B_D consisting of orthogonal rows

$$\begin{bmatrix} \cdots & 1 & \cdots \end{bmatrix}.$$

This matrix becomes a part of matrix B_E enforcing equality contraints. This approach results in dual formulation in Total FETI methods described in Chapter 10, where each row of B_D is associated with one component of Lagrange multipliers λ_D.

4.4 Primal QP problem

The discretized version of problem (3.5) then reads

$$\min \ \frac{1}{2}u^T K u - f^T u \ \text{ s.t. } \ B_I u \leq 0 \ \text{ and } \ B_E u = 0. \tag{4.4}$$

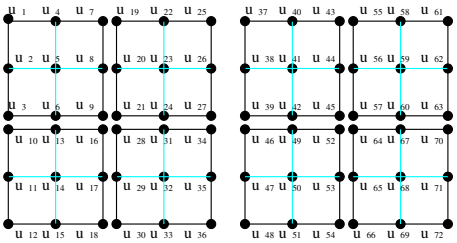


Figure 4.7: Primal QP problem

Although (4.4) is a typical QP problem, it is not very suitable for numerical computation especially because of ill conditioning, singularity of stiffness matrix K or complicated feasible set of displacements $\mathcal{K}_h = \{u : B_I u \leq 0 \text{ and } B_E u = 0\}$. How to avoid above introduced complications is to apply duality theory of convex programming.

Chapter 5

Algorithms for QP with simple bounds and equality constraints

This chapter focuses to the short description of three algorithms used for implementation and numerical experiments. Let us suppose the general QP problem with simple bound and equality constraints

$$\min \ \frac{1}{2}x^T Ax - b^T x \ \text{ s.t. } \ x_I \geq -\widetilde{x}_I \ \text{ and } \ Cx = 0.$$

These algorithms applied for dual problems are:

1. QPMPGP (Quadratic Programming with Modified Proportioning and Gradient Projection) is an efficient algorithm for the solution of convex quadratic programming problems with simple bounds, while equality constraints are enforced by penalty. The basic version was proposed independently by Dostál and Friedlander and Martínez and can be considered as a modification of the Polyak algorithm. Dostál and Schöberl in [47] combine the proportioning algorithm [9] with the gradient projections [44], they use the constant $\Gamma > 0$, the test to decide about leaving the face, and three types of steps to generate the sequence of iterates x^k that approximate the solution.

2. ALAPC (Augmented Lagrangian with Adaptive Precision Control) is a variant of algorithm proposed by Conn, Gould and Toint [8] for identification of stacionary points and supplied by the adaptive precision control of auxiliary problems in Step 1, so that the algorithm approximates new Lagrange multipliers for equality constraints in the outer loop, while quadratic programming problems with simple bounds are solved in the inner loop, e.g. by QPMPGP. The algorithm treats each type of constraints separately, so that efficient algorithm using projections and adaptive precision control in the active set strategy [9] may be used for the bound constrained quadratic programming problems.

3. SMALBE (Semi-Monotonic Augmented Lagrangian with Bound and Equality) is a modification of previous ALAPC algorithm with different criterium for update

of penalty based on value of augmented Lagrangian, with proved optimality. It is based on the observation, that better penalty approximation leads to increase of Lagrangian, so it increases penalty until increasing Lagrangian's values are generated. For more details see [52].

Main ideas of algorithms are in following schemes.

5.1 QPMPGP

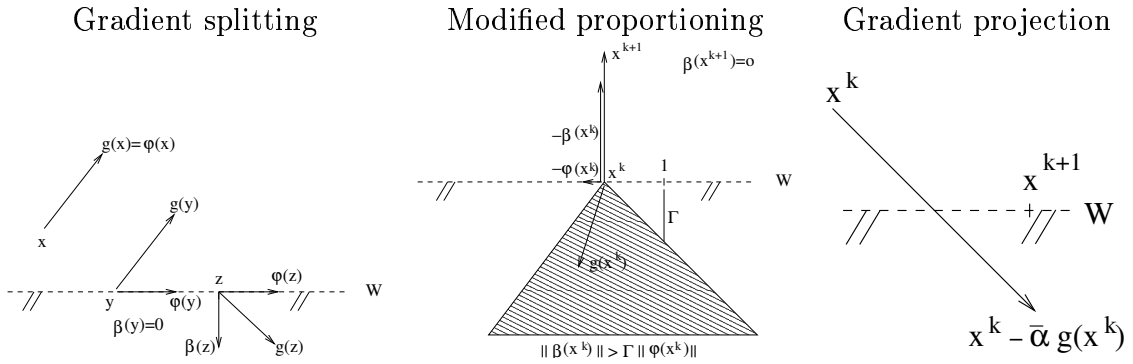


Figure 5.1: Main ideas of QPMPGP

Step 0 {Initialization of parameters}
$x^0 \in \Omega$, $\overline{\alpha} \in \left(0, \|A\|^{-1}\right]$, $\Gamma > 0$

Step 1 {Proportioning step - removes indices from $\mathcal{A}(x^k)$}
If x^k is not proportional then define $x^{k+1} = x^k - \alpha_{CG}\beta(x^k)$ by proportioning, $\alpha_{CG} = \frac{\beta^T(x^k)g(x^k)}{\beta^T(x^k)A\beta(x^k)}$minimizes $\theta(x^k - \alpha\beta(x^k))$ with respect to α

Step 2 {CG step}
If x^k is proportional then generate $x^{k+1} = x^k - \alpha_{CG}p^k$ by trial cg step, $p^{k+1} = \varphi(x^k) - \gamma p^k$, $\gamma = \frac{\varphi^T(x^k)Ap^k}{(p^k)^TAp^k}$

Step 3 {Expansion step - expands $\mathcal{A}(x^k)$}
If $x^k \in \Omega$ then accept it else generate $x^{k+1} = P_+(x^k - \overline{\alpha}\varphi(x^k)) = x^k - \overline{\alpha}\widetilde{\varphi}(x^k)$ by projection to feasible set

Let $g = g(x) = Ax - b$ be the gradient and projected gradient $g^P = g^P(x)$ of $\theta(x) = \frac{1}{2}x^TAx - x^Tb$ at x is given componentwise by

$$g_i^P = \begin{cases} g_i & \text{for} \quad x_i > -\widetilde{x}_i \quad \text{or} \quad i \notin I \\ \min(g_i, 0) & \text{for} \quad x_i = -\widetilde{x}_i \quad \text{and} \quad i \in I \end{cases}.$$

Let I and E denote the sets of indices corresponding to the entries of vectors x_I and x_E and let $\mathcal{A}(x)$ and $\mathcal{F}(x)$ define the *active set* and *free set* of indices of x

$$\mathcal{A}(x) = \{i \in I : x_i = -\widetilde{x}_i\} \qquad \text{and} \qquad \mathcal{F}(x) = \{i : x_i > -\widetilde{x}_i \text{ or } i \in E\},$$

determining the *chopped gradient* $g^C = \beta(x)$, the *inner* or *free gradient* $g^F = \varphi(x)$ and *reduced-free gradient* $g^R = \widetilde{\varphi}(x)$ of $\theta(x)$ in following way

$$\begin{gathered} g_i^C = 0 \quad \text{for} \quad i \in \mathcal{F}(x) \quad \text{and} \quad g_i^C = \min(g_i, 0) \quad \text{for} \quad i \in \mathcal{A}(x), \\ g_i^F = g_i \quad \text{for} \quad i \in \mathcal{F}(x) \quad \text{and} \quad g_i^F = 0 \quad \text{for} \quad i \in \mathcal{A}(x), \\ g_i^R = g_i^F \quad \text{for} \quad i \in E \quad \text{and} \quad g_i^R = \min(x_i/\overline{\alpha}, g_i^F) \quad \text{for} \quad i \in I. \end{gathered}$$

The unique solution $\overline{x} = \overline{x}(\mu, \rho)$ of auxiliary problems satisfies the Karush-Kuhn-Tucker conditions expressed by the relation

$$g^P(\overline{x}) = 0,$$

hence the solution of problem

$$\min \theta(x) \quad \text{s.t.} \quad x_I \geq -\widetilde{x}_I$$

satisfies the Karush-Kuhn-Tucker conditions if the following relation is valid

$$g^P(x^k) = \nu(x^k) = g^F(x^k) + g^C(x^k) = \varphi(x^k) + \beta(x^k).$$

The precision of the solution of auxiliary problems is controlled by norm of violation of Karush-Kuhn-Tucker condition in each inner iterate x^k by

$$\Gamma^2 \widetilde{\varphi}^T(x^k)\varphi(x^k) \geq \left\|\beta(x^k)\right\|^2,$$

while $\Gamma > 0$ and x^k satisfying this inequality is called as *proportional*. If $x \in \Omega$, i.e. $x_I \geq -\widetilde{x}_I$, we call x *feasible* and P_+ is the projection to set of feasible vectors defined as

$$P_+(x)_i = \max\{x_i, -\widetilde{x}_i\} \text{ for } i \in I, \quad P_+(x)_i = x_i \text{ for } i \in E.$$

The conjugate gradient steps are used to carry out the minimization in the face

$$W_J = \{x : x_i = -\widetilde{x}_i \quad \text{for} \quad i \in J\}$$

with a given active set $J = \mathcal{A}(x^k)$ efficiently, recurrence starts or restarts whenever x^k is generated by expansion or proportioning step. To touch up the performance of conjugate gradient method it is possible through the preconditioning, consisting in finding a suitable matrix that approximates in our case matrix A^{-1}, so that modified system is more easy to solve than the original one. More details about implementation of the algorithm may be found in [47], [51].

5.2 ALAPC

Step 0 {Initialization of parameters}
$0 < \alpha < 1$ (10^{-1}), $\beta > 1$ (10), $M > 0$ (10^0),
$\rho_0 > 0$ (10^1), $\eta_0 > 0$ (10^{-2}), $\mu^0 > 0$ (0), k (0)
Step 1 Find x^k so that $\left\|g^P(x^k, \mu^k, \rho_k)\right\| \leq M \left\|Cx^k\right\|$...(QPMPGP)
Step 2 If $\left\|g^P(x^k, \mu^k, \rho_k)\right\|$ and $M \left\|Cx^k\right\|$ are suffic. small then x^k is the solution
Step 3 $\mu^{k+1} = \mu^k + \rho_k C x^k$
Step 4 If $\left\|Cx^k\right\| \leq \eta_k$
then $\rho_{k+1} = \rho_k, \quad \eta_{k+1} = \alpha\eta_k$
else $\rho_{k+1} = \beta\rho_k, \quad \eta_{k+1} = \eta_k$
Step 5 Update $k = k + 1$ and return to Step 1

Let $g(x, \mu, \rho) = Ax - b + C^T(\mu + \rho Cx)$ denote the gradient of the augmented Lagrangian

$$L(x, \mu, \rho) = \frac{1}{2}x^T Ax - x^T b + \mu^T Cx + \frac{1}{2}\rho \left\|Cx\right\|^2,$$

and projected gradient $g^P = g^P(x, \mu, \rho)$ of L at x. The salient feature of this algorithm is that it deals with each type of constraint completely separately and that it accepts inexact solutions for the auxiliary box consrained problems in Step 1. The algorithm has been proved to converge for any combination of parameters satisfying given conditions, while only parameters ρ_0 and M are problem dependent (ρ_0 should be larger than M - the algorithm is designed so that it warrants the adaption of ρ to all parameters including M, typical values of parameters are in brackets). Step 1 can be realized by the minimization of the augmented Lagrangian L subject to $x_I \geq -\widetilde{x_I}$ by QPMPGP - for fixed μ and ρ, we are allowed to write $\theta(x) = L(x, \mu, \rho)$.

5.3 SMALBE

Step 0 {Initialization of parameters}
$\beta > 1$ (10), $M > 0$ (10^0), $\rho_0 > 0$ (10^1), $\eta_0 > 0$ (10^{-2}), $\mu^0 > 0$ (0), k (0)
Step 1 Find x^k so that $\left\|g^P(x^k, \mu^k, \rho_k)\right\| \leq M \left\|Cx^k\right\|$...(QPMPGP)
Step 2 If $\left\|g^P(x^k, \mu^k, \rho_k)\right\|$ and $M \left\|Cx^k\right\|$ are suffic. small then x^k is the solution
Step 3 $\mu^{k+1} = \mu^k + \rho_k C x^k$
Step 4 If $L(x^k, \mu^k, \rho_k) < L(x^{k-1}, \mu^{k-1}, \rho_{k-1}) + \frac{1}{2}\rho \left\|Cx^k\right\|^2$
then $\rho_{k+1} = \beta\rho_k$
else $\rho_{k+1} = \rho_k$
Step 5 Update $k = k + 1$ and return to Step 1

5.4 Theory of numerical scalability

Presented algorithms solving QP problems with bound and equality constraints have known rate of convergence given in terms of the spectral condition number of the quadratic problem. Convergence bounds that guarantee the scalability of algorithms and more details including lemmas and theorems for FETI-1, TFETI-1 and FETI-DP with pointwise and mortars conditions can be found in [47], [14], [15], [13], the key ingredient of scalability is optimality of dual penalty described in Chapter 7.

Chapter 6

Overview of FETI methods

The FETI (Finite Element Tearing and Interconnecting) method proposed by Farhat and Roux turned out to be one of the most successful algorithms for parallel solution of problems described by elliptic PDEs. The FETI-1 method is based on the decomposition of the spatial domain into non-overlapping subdomains that are "glued" by Lagrange multipliers. Efficiency of the FETI-1 method was further improved by introducing special projectors and preconditioners.

By projecting the Lagrange multipliers in each iteration onto an auxiliary space to enforce continuity of the primal solutions at the crosspoints, Farhat, Mandel and Tezaur obtained a faster converging FETI method for plate and shell problems - FETI-2.

Similar effect was achieved by a variant called the Dual-Primal FETI method FETI-DP, introduced by Farhat et al. The continuity of the primal solution at crosspoints is implemented directly into the formulation of the primal problem so that one degree of freedom is considered at each crosspoint shared by two and more adjacent subdomains. The continuity of the primal variables across the rest of the subdomain interfaces is once again enforced by Lagrange multipliers. After eliminating the primal variables, the problem reduces to a small, relatively well conditioned strictly convex QP problem that is again solved iteratively. In spite of the success of FETI for the solution of linear problems, it was demonstrated that the method is even more successful for the solution of variational inequalities. The reason is that duality transforms the general inequality constraints into the nonnegativity constraints so that efficient algorithms that exploit cheap projections and other tools may be exploited. Experimental scalability results based on application of special solvers [9, 55] were documented by Dostál, Gomes and Santos [2, 48] and Dostál and Horák [13, 69]. Scalability was later proved for an algorithm that combined FETI with optimal dual penalty Dostál and Horák [16, 69]. The same is true for adaptation of FETI–DP to the solution of variational inequalities.

A new Lagrange multipliers FETI–DP algorithm, FETI–C, based on active set strategies with additional planning steps, was introduced by Farhat et al. and its scalability was established only experimentally. Numerical scalability for FETI–DP algorithm for coercive problems was proven in [14]. Later, the result was extended to include mortar disctretization [15] and for semicoercive problems.

Implementation of the FETI-1 and FETI-2 method into general purpose packages requires an effective method for automatic identification of the kernels of the stiffness matrices of the subdomains as these kernels are used both in elimination of the primal variables and in definition of the natural coarse grid projectors. An effective method based on combination of the Cholesky factorization and the singular value decomposition was proposed by Farhat and Gérardin [26]. However, it turns out that it is still quite difficult to determine the kernels reliably in the presence of rounding errors. This was one of the motivations that led to development of the FETI-DP, where the stiffness matrices of the subdomains are invertible. However, even though FETI-DP may be efficiently preconditioned so that it scales better than the original FETI for plates and shells, the coarse grid defined by the corners without additional preconditioning is less efficient [14, 25] than that defined by the rigid body motions, which is important for some applications [23, 14, 25], and the FETI-DP method is more difficult to implement as it requires special treatment of the corners.

The new variant of the FETI-1 method TFETI-1 (Total-FETI-1) was proposed by Dostál. It is easier to implement and it preserves efficiency of the coarse grid of the classical FETI-1. The basic idea is to simplify inversion of the stiffness matrices of subdomains by using Lagrange multipliers not only for gluing of the subdomains along the auxiliary interfaces, but also for implementation of the Dirichlet boundary conditions. Heuristic arguments and results of numerical experiments indicate that the new method may be not only easier to implement, but also more efficient then the original FETI-1.

Another improvement lies in so called MFETI-DP (Multilevel-FETI-DP) applicable for problems with large coarse problem dimension. The size of coarse problem increases due to increasing number of subdomains increasing degree of parrallelism, reducing memory requirements, reducing CPU for factors and solution of coarse problem by forward/backward substitution. FETI-DP is applied recursively to the coarse problem. Numerical experiments compare finite and iterative solution of coarse problems including inexact solution with CG in simple V-cycle for both linear and contact problems.

The rate of convergence of QP algorithms can be bounded in terms of the bounds on the spectrum of the regular part of the Hessian matrix, and therefore the scalability of the resulting algorithm can be established. Such estimates are in terms of the decomposition parameter H and the discretization parameter h. An easy consequence of this result is a bound on the number of conjugate iterations required for finding the solution of the discretized variational inequality to a given precision. This bound is independent of both the decomposition of the computational domain and the discretization, provided that we keep the ratio H/h fixed. Numerical results are in agreement with the theory and confirm the numerical scalability of our algorithms.

The effort to develop scalable solvers for variational inequalities was not restricted to FETI-type methods. Analogue ideas were used in BETI (Boundary Element Tearing and Interconnecting) based method etc.

6.1 FETI-1 for coercive and semicoercive contact problems

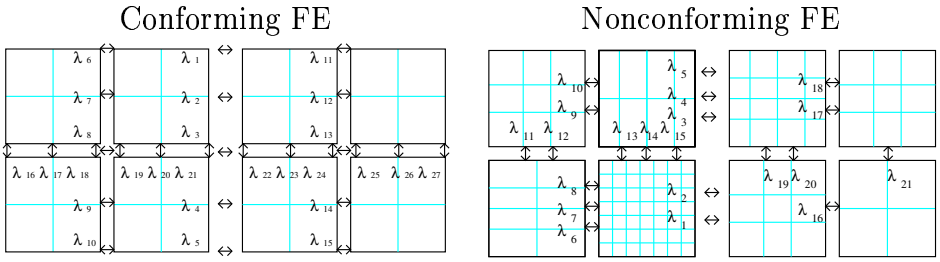


Figure 6.1: FETI-1 method with conforming and nonconforming FE

The FETI methods are characterized, that computational domain is partitioned into non-overlapping subdomains, the equilibrium equations are written at subdomain level and Lagrange multipliers are introduced at subdomain interfaces to enforce the compatibility of displacement field. The main idea of dual formulation is that large problem with primal variable u, representing displacement in subdomains' nodes, is transformed to the significantly smaller problem with dual variable λ representing the "gluing" forces on interfaces.

The Lagrangian of problem (4.4) looks

$$L(u, \lambda_I, \lambda_E) = \frac{1}{2} u^T K u - f^T u + \lambda_I^T B_I u + \lambda_E^T B_E u, \tag{6.1}$$

where λ_I and λ_E denote Lagrange multipliers associated with inequalities and equalities. Using notation $\lambda = \begin{bmatrix} \lambda_I \\ \lambda_E \end{bmatrix}$ and $B = \begin{bmatrix} B_I \\ B_E \end{bmatrix}$ the Lagrangian gets simplified form

$$L(u, \lambda) = \frac{1}{2} u^T K u - f^T u + \lambda^T B u.$$

The problem (6.1) is then equivalent to the saddle point problem

$$\text{find} \quad (\overline{u}, \overline{\lambda}) \quad \text{so that} \quad L(\overline{u}, \overline{\lambda}) = \sup_{\lambda_I \geq 0} \inf_{u} L(u, \lambda). \tag{6.2}$$

For fixed λ, the Lagrange function is convex in the first variable and the minimizer u satisfies equation

$$Ku = f - B^T \lambda, \tag{6.3}$$

which expresses, from a physical viewpoint, the subdomain equation of equilibrium with Neumann boundary conditions. Equation (6.3) has solution iff

$$f - B^T \lambda \in \mathrm{Im} K, \tag{6.4}$$

which can be expressed by means of a matrix R, whose columns span the null space of K and represent rigid body or zero energy modes of subdomains

$$R^T(f - B^T\lambda) = 0. \tag{6.5}$$

Matrix R can be formed directly so that each floating subdomain is assigned to a row of R with ones in positions of nodes that belong to the subdomain and zeros elsewhere. The solution u of (6.3) with assumption that λ satisfies (6.4) can be evaluated by formula

$$u = K^\dagger(f - B^T\lambda) + R\alpha. \tag{6.6}$$

Here $K^\dagger$ denotes matrix satisfying $KK^\dagger K = K$ such as generalized inverse or Moore-Penrose pseudoinverse.

Let's establish following notation

$$F = BK^\dagger B^T,\ \widetilde{G} = R^T B^T,\ \widetilde{d} = BK^\dagger f,\ \widetilde{e} = R^T f.$$

Substitution of (6.6) into problem (6.2) and some manipulations lead to minimization problem

$$\min\ \frac{1}{2}\lambda^T F\lambda - \lambda^T\widetilde{d}\ \text{ s.t. }\ \lambda_I \geq 0\ \text{ and }\ \widetilde{G}\lambda = \widetilde{e} \tag{6.7}$$

whose Hessian is positive definite and conforms with one from FETI basic method by Farhat and Roux [5], having relatively favorable distributed spectrum for application of the conjugate gradient method.

Even though the problem (6.7) is much more suitable for computation than (4.4), further improvement can be achieved by adapting observations and results of Farhat, Mandel and Roux [6], consisting in formulation of equivalent problem with augmented Lagrangian, that has spectral properties guaranting optimal convergence of conjugate gradient method. Before incorporation of these observations let's establish following notation

$$G = T\widetilde{G}, \quad e = T\widetilde{e}$$

and let T denote a nonsingular matrix, that defines the orthonormalization of the rows of $\widetilde{G}$. The problem of minimization on the subset of the affine space is transformed to the problem on subset of vector space by means of arbitrary $\widetilde{\lambda}$ which satisfies $G\widetilde{\lambda} = e$ while the solution is looked for in the form $\lambda = \widehat{\lambda} + \widetilde{\lambda}$. Because of relation

$$\frac{1}{2}\lambda^T F\lambda - \lambda^T\widetilde{d} = \frac{1}{2}\widehat{\lambda}^T F\widehat{\lambda} - \widehat{\lambda}^T(\widetilde{d} - F\widetilde{\lambda}) + \frac{1}{2}\widetilde{\lambda}^T F\widetilde{\lambda} - \widetilde{\lambda}^T\widetilde{d},$$

using old notation and denoting $d = \widetilde{d} - F\widetilde{\lambda}$, the problem (6.7) is equivalent to

$$\min\ \frac{1}{2}\lambda^T F\lambda - \lambda^T d\ \text{ s.t. }\ \lambda_I \geq -\widetilde{\lambda}_I\ \text{ and }\ G\lambda = 0. \tag{6.8}$$

Further improvement is based on the observation, that the augmented Lagrangian for problem (6.8) can be decomposed by orthogonal projectors

$$Q = G^T G \qquad \text{and} \qquad P = I - Q$$

on the kernel of G and on the image space of G^T: $\mathrm{Im}Q = \mathrm{Ker}G$ and $\mathrm{Im}P = \mathrm{Im}G^T$.

The modified dual formulation of problem (6.8) has then the form

$$\min \frac{1}{2}\lambda^T PFP\lambda - \lambda^T Pd \text{ s.t. } \lambda_I \geq -\widetilde{\lambda_I} \text{ and } G\lambda = 0. \tag{6.9}$$

Further

$$\alpha = -(R^T\widetilde{B}^T\widetilde{B}R)^{-1}R^T\widetilde{B}^T\widetilde{B}K^{\dagger}(f - B^T\overline{\lambda})$$

denotes vector of amplitudes that specifies the contribution of the null space R to the solution u determined by relation $\widetilde{B}K^{\dagger}(f - B^T\overline{\lambda}) + \widetilde{B}R\alpha = 0$, with $\widetilde{B} = \begin{bmatrix} \widetilde{B_I} \\ B_E \end{bmatrix}$ and $\widetilde{B_I}$ formed by rows of B_I, that correspond to active constraints latter characterized by $\overline{\lambda_i} \neq 0$.

Let's investigate for a comparison the distributions of the spectra of Hessians

$$H_1 = F + \rho\widetilde{G}^T\widetilde{G}, \quad H_2 = F + \rho G^T G \quad \text{and} \quad H_3 = PFP + \rho Q$$

of the augmented Lagrangians corresponding to the problems (6.7), (6.8) and (6.9), with the assumption that the eigenvalues of F are in the interval $[a, b]$, the eigenvalues of $\widetilde{G}^T\widetilde{G}$ are in $[\gamma, \delta]$ and $\sigma(A)$ denotes a spectrum of square matrix A. Using the analysis of [7] it is possible to obtain following estimates

$$\sigma(H_1) \subseteq [a, b] \cup [a + \rho\gamma, b + \rho\delta], \quad \sigma(H_2) \subseteq [a, b] \cup [a + \rho, b + \rho] \quad \text{and}$$
$$\sigma(H_3) \subseteq [a_P, b_P] \cup \{\rho\},$$

where $[a_P, b_P] \subseteq [a, b]$ denotes the interval of non-zero eigenvalues of PFP (see Figure 6.2). If ρ is sufficiently large and $\gamma < \rho$, then the spectrum of H_1 is distributed in two intervals with the larger one on the right, while the rate of convergence of conjugate gradients for minimization of quadratic function with H_1 depends on the penalization parameter ρ. In the second case the situation is much more favorable for H_2, because the spectrum is always distributed in two intervals of the same length and the rate of convergence is governed by the effective condition number $\overline{\kappa}(H_2) = \frac{4b}{a}$, so that the number k_2 of conjugate gradient iterations for reducing the gradient of the augmented Lagrangian $L_2(\lambda, \mu, \rho) = \frac{1}{2}\lambda^T(F + \rho Q)\lambda - \lambda^T d + \mu^T G\lambda$ for (6.8) by ϵ satisfies

$$k_2 \leq \frac{1}{2}\mathrm{int}\left(\sqrt{\frac{4b}{a}}\ln\left(\frac{2}{\epsilon}\right) + 1\right)$$

and does not depend on the penalization parameter ρ. In the third case the Hessian H_3 of the augmented Lagrangian $L_3(\lambda, \mu, \rho) = \frac{1}{2}\lambda^T(PFP + \rho Q)\lambda - \lambda^T Pd + \mu^T G\lambda$ is decomposed by projectors P and Q whose image spaces are invariant subspaces of H_3 and the number k_3 of conjugate gradient iterations for reducing the gradient of $L_3(\lambda, \mu, \rho)$ for (6.9) by ϵ satisfies

$$k_3 \leq \frac{1}{2}\mathrm{int}\left(\sqrt{\frac{b_P}{a_P}}\ln\left(\frac{2}{\epsilon}\right) + 3\right),$$

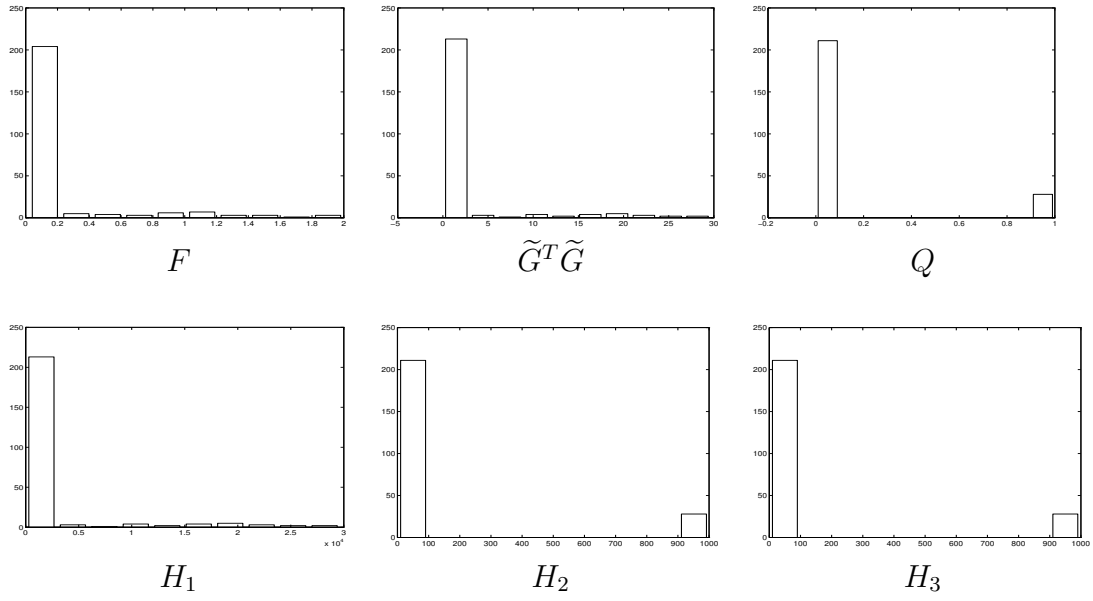


Figure 6.2: Example of spectrum distributions of Hessians ($h = 1/8, H = 1/2, \rho = 10^3$)

while according to the analysis of the FETI method by Farhat, Mandel and Roux following relation is valid

$$\frac{a_P}{b_P} \leq C\frac{H}{h}$$

with h denoting the mesh and H subdomain diameter (see Figure 6.3).

The FETI-1 method was recently combined with the dual penalty method by Dostál, Horák [23],[16] to obtain a theoretically supported scalable algorithm for solution of coercive and semicoercive variational inequalities. Thus the rate of convergence does not depend on penalization parameter ρ or discretization parameter h, but it is bounded by the constant given by the ratio $\frac{H}{h}$: $\kappa(PFP \mid \mathrm{Im}P) \leq C\frac{H}{h}$.

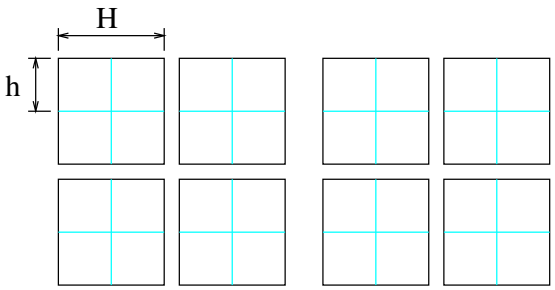


Figure 6.3: Mesh and subdomain parameters

6.2 FETI-DP for coercive contact problems

During the joint work with Dan Stefanica we developed an optimal scalable algorithm for the numerical solution of coercive variational inequalities, by combining FETI algorithms of dual-primal type with recent results for bound constrained QP problems, that is described in this section.

A variant of the FETI method called the Dual-Primal FETI method (FETI–DP) was introduced by Farhat et al. [27]. The continuity of the primal solution at crosspoints is implemented directly into the formulation of the primal problem so that one degree of freedom is considered at each crosspoint shared by two and more adjacent subdomains. The continuity of the primal variables across the rest of the subdomain interfaces is once again enforced by Lagrange multipliers. After eliminating the primal variables, the problem reduces to a small, unconstrained, relatively well conditioned strictly convex QP problem that is again solved iteratively. An attractive feature of FETI–DP is that the local problems that are solved to eliminate the primal variables are non–singular. Moreover, the resulting QP problem is unconstrained and its conditioning may be further improved by preconditioning.

The FETI–DP methodology is first applied to the discretized elliptic variational inequality to obtain a strictly convex QP problem with nonnegativity constraints. We also obtain a bound on the number of conjugate iterations required for finding the solution of the discretized variational inequality to a given precision. This bound is independent of both the decomposition of the computational domain and the discretization, provided that we keep the ratio H/h fixed.

For each crosspoint, i.e. a corner that belongs to two and more subdomains, a single global degree of freedom is considered, while two degrees of freedom at all the other matching nodes across subdomain edges. None of the nodes on contact interface should be defined as corner, see C. Farhat et al. in [46], page 2411.

We partition the nodal values of u into boundary corner nodal values, denoted by u_{bc}, and remainder nodal values, denoted by u_r. Note that u_r can be further split into interior nodal values u_i and remainder boundary nodal values u_{br}.

The continuity conditions at the subdomain corners are enforced by using a global vector of degrees of freedom $u_c = \left[\begin{array}{c} u_c^1, ..., u_c^{N_c} \end{array} \right]^T$ and a global to local map B_c with one nonzero entry per line equal to 1, and by requiring that $u_{bc} = B_c u_c$.

Therefore,

$$u = \left[\begin{array}{c} u_i \\ u_b \end{array} \right] = \left[\begin{array}{c} u_i \\ u_{br} \\ u_{bc} \end{array} \right] = \left[\begin{array}{c} u_r \\ u_{bc} \end{array} \right] = \left[\begin{array}{c} u_r \\ B_c u_c \end{array} \right].$$

Problem can be written as a constrained minimization problem as follows:

$$\min \frac{1}{2} u^T K u - u^T f \quad \text{s.t.} \quad B_I u \leq 0, B_E u = 0, \quad \text{and} \quad u_{bc} = B_c u_c. \tag{6.10}$$

Let f_{bc} and f_r be the parts of the right hand side f corresponding to the boundary corner and remainder nodes, respectively. The Lagrangian associated with problem

(6.10) can be expressed using Lagrange multipliers λ_E and λ_I to enforce equality and inequality constraints as follows:

$$L(u_r, u_c, \lambda_E, \lambda_I) = \frac{1}{2}[u_r^T \ (B_c u_c)^T] K \begin{bmatrix} u_r \\ B_c u_c \end{bmatrix} - [u_r^T \ u_c^T] \begin{bmatrix} f_r \\ B_c^T f_{bc} \end{bmatrix} + u^T B_E^T \lambda_E + u^T B_I^T \lambda_I. \tag{6.11}$$

Let $B_{I,r}$ and $B_{I,bc}$ be the matrices made of the columns of B_I corresponding to u_r and u_{bc}, respectively; define $B_{E,r}$ and $B_{E,bc}$ similarly. Then $B_I = [B_{I,r} \ B_{I,bc}]$ and $B_E = [B_{E,r} \ B_{E,bc}]$. We can also group the parts of the Lagrange multiplier matrices B_I and B_E together with respect to the boundary corner and reminder nodes as follows:

$$B_r = \begin{bmatrix} B_{I,r} \\ B_{E,r} \end{bmatrix} \quad \text{and} \quad B_{bc} = \begin{bmatrix} B_{I,bc} \\ B_{E,bc} \end{bmatrix}.$$

Then $B_r u_r = \pm u_{br}$. Let

$$\lambda = \begin{bmatrix} \lambda_I \\ \lambda_E \end{bmatrix}.$$

Then

$$u^T B_E^T \lambda_E + u^T B_I^T \lambda_I = u_r^T B_r^T \lambda + u_c^T B_c^T B_{bc}^T \lambda.$$

Let K_{rr}, K_{rc}, and K_{cc} be the blocks of K corresponding to the decomposition of u into u_r and u_{bc}, so that

$$K = \begin{bmatrix} K_{rr} & K_{rc} \\ K_{rc}^T & K_{cc} \end{bmatrix}.$$

To minimize $L(u_r, u_c, \lambda_E, \lambda_I)$ over u_r, we rewrite (6.11) as

$$L(u_r, u_c, \lambda_E, \lambda_I) = \frac{1}{2}\left(u_r^T K_{rr} u_r + 2u_r^T K_{rc} B_c u_c + u_c^T B_c^T K_{cc} B_c u_c\right) - u_r^T f_r - u_c^T B_c^T f_c + u_r^T B_r^T \lambda + u_c^T B_c^T B_{bc}^T \lambda$$

and obtain that u_r is a solution of

$$K_{rr} u_r + K_{rc} B_c u_c - f_r + B_r^T \lambda = 0.$$

Let $S_{cc} = K_{cc} - K_{rc}^T K_{rr}^{-1} K_{rc}$ be a Schur complement type matrix. We end up with the following Lagrangian to minimize over u_c:

$$L_c(u_c, \lambda_E, \lambda_I) = \frac{1}{2} u_c^T B_c^T S_{cc} B_c u_c - u_c^T B_c^T f_c + u_c^T B_c^T B_{bc}^T \lambda - \frac{1}{2}\left(f_r - K_{rc} B_c u_c - B_r^T \lambda\right)^T K_{rr}^{-1} \left(f_r - K_{rc} B_c u_c - B_r^T \lambda\right).$$

We use the following notations borrowed from [27]:

$$F_{I_{rr}} = B_r K_{rr}^{-1} B_r^T;$$
$$F_{I_{rc}} = B_r K_{rr}^{-1} K_{rc} B_c$$
$$K_{cc}^* = B_c^T (K_{cc} - K_{rc}^T K_{rr}^{-1} K_{rc}) B_c.$$

The solution to the minimization of $L_c(u_c, \lambda_E, \lambda_I)$ over u_c must satisfy

$$K_{cc}^* u_c - F_{I_{rc}}^T \lambda - f_c^* = 0,$$

where $f_c^* = B_c^T f_{bc} - B_c^T K_{rc}^T K_{rr}^{-1} f_r$. This problem is solvable since, for a coercive problem, K_{cc}^* is a non-singular matrix. The corresponding minimal value of $L_c(u_c, \lambda_E, \lambda_I)$ is

$$L_\lambda(\lambda_E, \lambda_I) = -\frac{1}{2}\left(f_c^* + F_{I_{rc}}^T \lambda\right)^T (K_{cc}^*)^{-1} \left(f_c^* + F_{I_{rc}}^T \lambda\right) - \frac{1}{2}\left(f_r - B_r^T \lambda\right)^T K_{rr}^{-1} \left(f_r - B_r^T \lambda\right).$$

Thus, maximizing L_λ over $\lambda_I \geq 0$ is equivalent to finding

$$\min \frac{1}{2}\lambda^T F \lambda - \lambda^T d \quad \text{s.t.} \quad \lambda_I \geq 0, \tag{6.12}$$

where

$$F = F_{I_{rr}} + F_{I_{rc}} (K_{cc}^*)^{-1} F_{I_{rc}}^T; \qquad d = B_r K_{rr}^{-1} f_r - F_{I_{rc}} (K_{cc}^*)^{-1} f_c^*. \tag{6.13}$$

Having the solution of dual problem $\overline{\lambda}$, primal solution can be reconstructed according to formulae

$$u_c = (K_{cc}^*)^{-1} \left(F_{I_{rc}}^T \overline{\lambda} + f_c^*\right), u_r = K_{rr}^{-1} \left(f_r - B_r^T \overline{\lambda} - K_{rc} B_c u_c\right), u_{bc} = B_c u_c.$$

The convergence bounds guarantee the scalability of the algorithm, this is confirmed by numerical experiments in section 13.2. The effect of different choices of nonmortar sides on the numerical performance of our method is investigated.

6.3 FETI-DP for semicoercive contact problems

Continuing in joint work with Dan Stefanica, following similar approach as for coercive FETI-DP, we used the FETI–DP method to develop a scalable algorithm for the numerical solution of a semicoercive variational inequality. We recall that the variational inequality describing the equilibrium of a system of bodies in contact is semicoercive if floating bodies exist. In this case, the application of FETI–DP methodology results in a convex QP problem with a positive semidefinite Hessian matrix and bound and equality constraints. For contact problems, the kernel of the Hessian matrix is spanned by the rigid body motions of the floating bodies. We solve this problem by using SMALBE. The rate of convergence of this algorithm can be estimated in terms of the bounds on the spectrum of the regular part of the Hessian of the quadratic cost function. We derive such bounds in terms of the decomposition parameter H and of the discretization parameter h, therefore establishing the scalability of our algorithm. We also obtain a bound on the number of conjugate iterations required for finding the solution of the discretized variational inequality to a given precision. This bound is independent of both the decomposition of the computational domain and the discretization, provided that the ratio H/h is kept fixed. Reported numerical results are in agreement with the theory and confirm the numerical scalability of the algorithm.

Several extensions of our method are possible and work on these problems is currently in progress. For example, a faster algorithm might be obtained by using a preconditioner in the conjugate gradient step; see recent work by Dostál and Lesoinne [58]. Another possibility is to use a mortar finite element discretization of the computational domain [66]; see also, e.g., [38, 57]. To minimize the inherent extra computational effort required to compute the mortar conditions, every subdomain would be discretized with continuous elements. The mesh would be unstructured only across the contact interface, where mortar conditions will be required.

We note that the effort to develop scalable solvers for variational inequalities was not restricted to FETI–type methods. For example, multigrid ideas were used early on by Mandel [71]. Kornhuber, Krause and Wohlmuth [60, 61, 67] introduced an algorithm based on monotone multigrid with scalable solution of auxiliary linear problems. Combining multigrid ideas and approximate projections, Schöberl [44, 56] introduced an algorithm for which linear complexity was established.

We follow the notation and relations introduced in previous section concerning FETI-DP for coercive contact problems that was proven to be numerically scalable. The discretized version of semicoercive problem has the form as coercive problem (6.10), where $K = \mathrm{diag}(K^1, K^2)$ is the block diagonal positive semidefinite stiffness matrix. The block K^1 corresponding to Ω^1 is nonsingular, due to the Dirichlet boundary conditions enforced on Γ_u^1. The block K^2 corresponding to Ω^2 is singular and its kernel is made of a vector v with all entries equal to 1. Therefore, the kernel of K is spanned by the matrix R defined by

$$R \;=\; \begin{bmatrix} 0 \\ v \end{bmatrix}. \tag{6.14}$$

Even though R is a column vector for our model problem, we will regard R as a matrix whose columns span the kernel of K. Moreover, we assume that the nodes in Ω^1 and Ω^2 are numbered contiguously, so that K^i, $i = 1, 2$ are block diagonal matrices.

We reorder also the matrix R to comply with the above splitting of the nodal values of u. The Lagrangian associated with problem can be expressed using the above splitting of u and Lagrange multipliers λ_E and λ_I to enforce the equality and inequality constraints as in coercive case.

The solution $\overline{u}_c$ to the minimization of $L_c(u_c, \lambda_E, \lambda_I)$ over u_c must satisfy

$$K_{cc}^* \overline{u}_c \;-\; F_{I_{rc}}^T \lambda \;-\; f_c^* \;=\; 0. \tag{6.15}$$

Note that K_{cc}^* is singular so that problem (6.15) is solvable if and only if $F_{I_{rc}}^T \lambda + f_c^*$ belongs to the range of K_{cc}^*. Since K_{cc}^* is symmetric, this is equivalent to requiring that

$$F_{I_{rc}}^T \lambda + f_c^* \;\perp\; \mathrm{Ker} K_{cc}^*.$$

Let R_c be the matrix whose columns span the kernel of the Schur complement matrix $S_{cc} = K_{cc} - K_{rc}^T K_{rr}^{-1} K_{rc}$. Note that R_c and R, the matrix spanning the kernel of the stiffness matrix K, cf. (6.14), are connected by

$$R \;=\; \begin{bmatrix} -K_{rr}^{-1} K_{rc} \\ I_c \end{bmatrix} R_c,$$

where I_c is an identity matrix. Then, the kernel of K^*_{cc} is spanned by

$$\widetilde{R}_c = (B_c^T B_c)^{-1} B_c^T R_c.$$

The condition for the solvability of (6.15) can be written as

$$\widetilde{R}_c^T (F_{I_{rc}}^T \lambda \ + \ f_c^*) = 0. \tag{6.16}$$

If (6.16) is satisfied, the solution $\overline{u}_c$ of (6.15) has the form

$$\overline{u}_c \ = \ (K^*_{cc})^\dagger (F_{I_{rc}}^T \lambda \ + \ f_c^*) + \widetilde{R}_c \alpha,$$

where $(K^*_{cc})^\dagger$ denotes a pseudoinverse of K^*_{cc}, i.e., a matrix satisfying $K^*_{cc}(K^*_{cc})^\dagger K^*_{cc} = K^*_{cc}$, and α is an arbitrary vector. The corresponding minimal value of $L_c(\overline{u}_c, \lambda_E, \lambda_I)$ is

$$L_\lambda(\lambda_E, \lambda_I) \ = \ -\frac{1}{2} \left(f_c^* + F_{I_{rc}}^T \lambda \right)^T (K^*_{cc})^\dagger \left(f_c^* + F_{I_{rc}}^T \lambda \right) - \frac{1}{2} \left(f_r - B_r^T \lambda \right)^T K_{rr}^{-1} \left(f_r - B_r^T \lambda \right).$$

Maximizing L_λ over $\lambda_I \geq 0$ is equivalent to finding

$$\min \frac{1}{2} \lambda^T F \lambda - \lambda^T \widetilde{d} \quad \text{s.t.} \quad \lambda_I \geq 0 \ \text{ and } \ \widetilde{R}_c^T (F_{I_{rc}}^T \lambda \ + \ f_c^*) = 0, \tag{6.17}$$

where

$$F = F_{I_{rr}} + F_{I_{rc}} (K^*_{cc})^\dagger F_{I_{rc}}^T; \qquad \widetilde{d} = B_r K_{rr}^{-1} f_r - F_{I_{rc}} (K^*_{cc})^\dagger f_c^*. \tag{6.18}$$

Even though problem (6.17) is much more suitable for computations than (6.10) and was used for solving discretized variational inequalities efficiently [54], further improvement may be achieved by modifications as follows. Let $\widetilde{G} = \widetilde{R}_c^T \ F_{I_{rc}}^T$, $\widetilde{e} = -\widetilde{R}_c^T f_c^*$ and denote by T a nonsingular matrix that defines the orthonormalization of the rows of $\widetilde{G}$ such that the matrix $G = T\widetilde{G}$ has orthonormal rows. Let $e = T\widetilde{e}$. Then, problem (6.17) reads

$$\min \quad \frac{1}{2} \lambda^T F \lambda - \lambda^T \widetilde{d} \quad \text{s.t.} \quad \lambda_I \geq 0 \quad \text{and} \quad G\lambda = e. \tag{6.19}$$

Next, the problem of minimization is transformed on the subset of the affine space to a minimization problem on the subset of a vector space. Let $\widetilde{\lambda}$ be an arbitrary feasible vector such that $G\widetilde{\lambda} = e$. We look for the solution λ of (6.17) in the form $\lambda = \mu + \widetilde{\lambda}$. Since

$$\frac{1}{2} \lambda^T F \lambda \ - \ \lambda^T \widetilde{d} \ = \ \frac{1}{2} \mu^T F \mu - \mu^T (\widetilde{d} - F\widetilde{\lambda}) \ + \ \frac{1}{2} \widetilde{\lambda}^T F \widetilde{\lambda} \ - \ \widetilde{\lambda}^T \widetilde{d},$$

problem (6.19) is, after returning to the old notation by replacing μ by λ, equivalent to

$$\min \quad \frac{1}{2} \lambda^T F \lambda - d^T \lambda \quad \text{s.t.} \quad \lambda_I \geq -\widetilde{\lambda}_I \quad \text{and} \quad G\lambda = 0, \tag{6.20}$$

with $d = \widetilde{d} - F\widetilde{\lambda}$. Our final step is based on the observation that the augmented Lagrangian for problem (6.20) may be decomposed by the orthogonal projectors

$$Q = G^T G \qquad \text{and} \qquad P = I - Q$$

on the image space of G^T and on the kernel of G, respectively. Since $P\lambda = \lambda$ for any feasible λ, problem (6.20) is equivalent to

$$\min \quad \frac{1}{2}\lambda^T PFP\lambda - \lambda^T Pd \quad \text{s.t.} \quad \lambda_I \geq -\widetilde{\lambda}_I \quad \text{and} \quad G\lambda = 0. \tag{6.21}$$

The Hessian $H_{AL} = PFP + \rho Q$ of the augmented Lagrangian

$$L(\lambda, \mu, \rho) = \frac{1}{2}\lambda^T (PFP + \rho Q)\lambda - \lambda^T Pd + \lambda^T G^T \mu \tag{6.22}$$

is decomposed by the projectors P and Q whose image spaces are invariant subspaces of H_{AL}. Let $a, b > 0$ such that the non-zero eigenvalues of F restricted to $\mathrm{Im}P$ are located in the interval $[a, b]$. Then,

$$\sigma(H_{AL}) \subseteq [a, b] \cup \{\rho\}.$$

By the analysis of Axelsson [7], the number k of conjugate gradient iterations necessary to reduce the gradient of the augmented Lagrangian (6.22) by ϵ satisfies

$$k \leq \frac{1}{2}\mathrm{int}\left(\sqrt{\frac{b}{a}}\ln\left(\frac{2}{\epsilon}\right) + 3\right).$$

The proof of the numerical scalability of the SMALBE algorithm is based in part on spectral bounds for the operator F restricted to $\mathrm{Im}P$. The bounds on a and b are in terms of the decomposition and the discretization parameters H and h that guarantee the optimality of our algorithm, see Chapter 9.

6.4 Augmented FETI methods: FETI-2, FETI-DP2 for coercive and semicoercive contact problems

Structural mechanics problems can be subdivided into second-order and fourth-order problems. Second-order problems are typically modeled by bar, plane stress/strain, and solid elements, and fourth-order problems by beam, plate, and shell elements. The condition number of a generalized symmetric stiffness matrix K arising from FE discretization of second-order problems grows aymptotically with the mesh size h as

$$\kappa(K) \leq C\frac{1}{h^2},$$

and that of a generalized symmetric stiffness matrix arising from the FE discretization of a fourth-order problems grows aymptotically with the mesh size h as

$$\kappa(K) \leq C\frac{1}{h^4}.$$

Material and discretization heterogeneities as well as bad element aspect ratios, all of which are common in real-world FE models, worsen further the above condition numbers.

The FETI-1 method incorporates a relatively small-size auxiliary problem which is based on the rigid body modes of the floating subdomains. This coarse problem accelerates convergence by propagating the error globally. For second-order elasticity problems, the condition number of the interface problem associated with FETI-1 equipped with the Dirichlet preconditioner grows at most polylogarithmically with the number of elements per subdomain as

$$\kappa(F_D^{-1}PFP) \leq C\left(1+\lg^2\frac{H}{h}\right). \tag{6.23}$$

This improved estimate establishes the numerical scalability.

For fourth-order problems, preserving this quasi-optimal condition number estimate requires enriching the coarse problem of FETI-1 by the subdomain corner modes. This enrichment transforms the original FETI-1 method into algorithm known as the FETI-2 method. With domain decomposition using local Neumann problems, the jumps of local solution fields at interface crosspoints can be discontinuous. For high-order problems the singularity entails dependence of the condition number upon the local mesh size. So it is of great importance to get rid of these singularities. The solution consists in constraining the Lagrange multiplier to generate local displacement fields that are continuous at interface crosspoints. Enforcing this constraint induces another level of preconditioning for the FETI method.

In FETI-2 method we solve

$$\min\ \frac{1}{2}\lambda^T P_C PFP\lambda - \lambda^T P_C Pd \ \text{ s.t. } \ \lambda_I \geq -\widetilde{\lambda}_I \ \text{ and } \ G\lambda = 0, \tag{6.24}$$

using standard FETI-1 notation, projector $P_C = I - PFPC^T(CPFPC^T)^{-1}C$ playing the role of preconditioner, and C is chosen matrix, we can see that

$$Cr = CP(d - F\lambda) = CBu = 0.$$

Notice that Bu is the jump of solution accross the subdomain interfaces. By $CP(d - F\lambda) = 0$ we make this jump orthogonal to the rows of C. If we choose C such that each row correspons to a subdomain corner mode, then the solution is always continuous at these corners. The corner modes and the associated global operator C are defined from corner motions in the same way as rigid body modes and operator G are defined from rigid body motions. So the enriched coarse grid matrix is $[C\ G] = \left[B_{bc}\ R^T\right] B^T$, with B_{bc}, R, B defined above.

Unfornately, enriching the coarse problem of FETI-1 by the subdomain corner modes increases its computational complexity to a point, where the overall scalability is diminished on a very large number of processors, $N_{procs} > 1000$. For this reason, the basic principles of FETI-2 were revisited to construct more efficient FETI-DP method

having the same quasi-optimal condition number estimate (6.23) for both second- and fourth-order problems, but employs more economical coarse problem than FETI-2.

As FETI-1 saddle point problem

$$Ku = f - B^T\lambda,\ Bu \leq 0,\ GBu \leq 0$$

results after augmentation in FETI-2 saddle point problem

$$Ku = f - B^T\lambda,\ Bu \leq 0,\ GBu \leq 0,\ CBu \leq 0,$$

in similar way, supposing splitting of nodes into boundary remainder, boundary corner, remainder and interior, FETI-DP saddle point problem

$$Ku = f - B^T\lambda,\ Bu \leq 0$$

results after augmentation in FETI-DP2 saddle point problem

$$Ku = f - B^T\lambda,\ Bu \leq 0,\ CBu \leq 0.$$

How the FETI-DP coarse problem could be extended to improve convergence? This can be accomplished again by forcing the residual to be orthogonal to a chosen set of vectors at each iteration of FETI-DP algorithm. Let C be a matrix of arbitrarily chosen vectors, r the residual, then we enforce the following equation to enhance convergence

$$Cr = CBu = 0. \tag{6.25}$$

We add these equations by introducing new Lagrange multipliers, to enforce constraints associated with (6.25). The resulting FETI-DP operator has the same form. We obtain augmented FETI-DP coarse grid, coined as FETI-DP2, that can be viewed as preconditioned FETI-DP. These C matrices can be chosen to be average x, y or z jump along a subdomain edge resulting in an edge by edge sparsity pattern for the augmented set of equations. The augmented version is more robust and scale better for more complicated high-order problems. Numerical experiments comparing the performance of FETI-1, FETI-2 and FETI-DP methods for coercive contact model problem are summarized in Section 13.2.5.

Part II

Chapter 7

Optimality of dual penalty and numerical scalability

It is neccessary to mention so called optimality of dual penalty method - Dostál, Horák [16] - QPMPGP algorithm for FETI-1 problem with dual penalty to enforce equality contraints. This is theoretically supported scalable algorithm for the solution of coercive and semicoercive variational inequalities.

The penalty method is optimal in the sence that a given bound on the relative error of violation of the equality contraints may be achieved with the value of the penalty parameter independently of the discretization parameter

$$\|G\lambda\| \leq C\frac{1+\epsilon}{\rho}\|Pd\|\,.$$

My numerical experiments presented in section 13.1 and illustrated in Figure 13.1c lead to this important theoretical feature resulting in scalability of our algorithms. Primal penalty method formulated in terms of displacements can experience various numerical difficulties. For more details see Theorems 6.1, 6.2 and 6.4 and Lemma 6.3 of [16], which are based on the estimate by Mandel and Tezaur, Lemma 3.11 of [20].

Chapter 8

FETI-DP with corners on contact interface

8.1 Corners on contact interface

As mentioned by C. Farhat et al. in [46], page 2411, he avoided the corners on the contact zone: "As stated earlier, the FETI-DP strategy requires partitioning the perfect interface into corner and remainder dofs. However, all dofs existing on the potential contact interface must be included in the remainder partition, and none of them can be defined as a corner dof. This is because by definition, the displacement field at a corner node is defined only at the global level." Usage of corners on the contact interface without any modification results into jumps and penetration in these corners as shown in Figure 8.1.

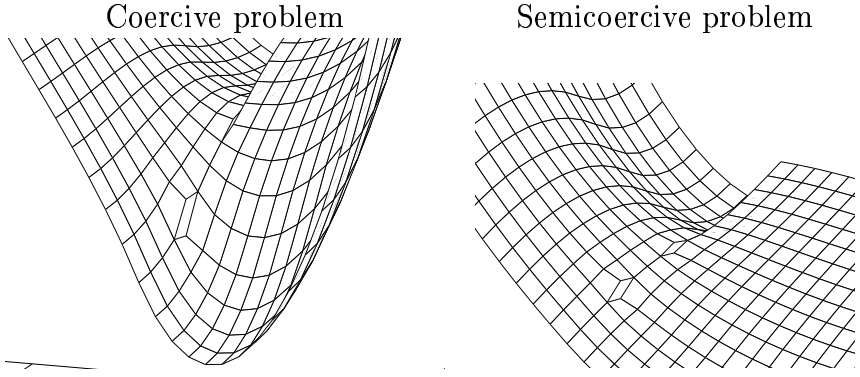


Figure 8.1: Incorrect FETI-DP for coercive and semicoercive problem with corners on the contact zone - zoomed jumps in solution

Three solutions appeared - two of them follow Farhat's advise and the third one consists in modification of method for corners on contact zone:

- corner node belongs to 4 subdomains as in Figure 8.2a

- corner node belongs to 4 and 2 subdomains except the contact zone as in Figure 8.2b
- corner node belongs to 4 and 2 subdomains including contact zone as in Figure 8.2c

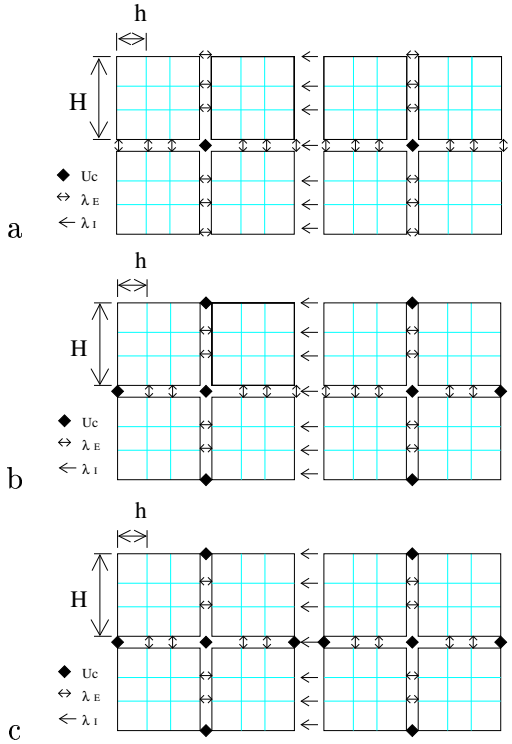


Figure 8.2: FETI-DP with changing meshes of corners and Lagrange multipliers

Special consideration earns my research result in this area - the third case - developed during my stay at Boulder University of Colorado, Department of Aerospace Engineering.

The usage of corners on contact interface is possible through an additional condition that preserves the nonpenetration in Lagrange multipliers. This condition comes from the natural condition $Bu \leq 0$. According to C. Farhat, all dofs existing on the potential contact interface must be included in the remainder partition, so we have to ensure then only $B_r u_r \leq 0$. This is not true in case with corners on contact zone anymore and so term

$$B_r u_r + B_{bc} u_{bc} \leq 0$$

is essential in our formulation, which proves finally in $\widehat{F}_{I_{rc}} = F_{I_{rc}} - B_{bc} B_c$ instead of $F_{I_{rc}}$.

Let us start with algebraic formulation of the saddle point problem with no corners on the contact zone

$$\begin{bmatrix} K_{rr} & K_{rc} \\ K_{rc}^T & K_{cc} \end{bmatrix} \begin{bmatrix} u_r \\ u_{bc} \end{bmatrix} = \begin{bmatrix} f_r \\ f_{bc} \end{bmatrix} - \begin{bmatrix} B_r^T \\ 0 \end{bmatrix} \lambda, \;\; B_r u_r \leq 0, \;\; \lambda_I \geq 0.$$

The modified saddle point problem allowing the use of corners on the contact interface reads

$$\begin{bmatrix} K_{rr} & K_{rc} \\ K_{rc}^T & K_{cc} \end{bmatrix} \begin{bmatrix} u_r \\ u_{bc} \end{bmatrix} = \begin{bmatrix} f_r \\ f_{bc} \end{bmatrix} - \begin{bmatrix} B_r^T \\ B_{bc}^T \end{bmatrix} \lambda, \;\; B_r u_r + B_{bc} u_{bc} \leq 0, \;\; \lambda_I \geq 0$$

that is equivalent to

$$K_{rr} u_r + K_{rc} u_{bc} = f_r - B_r^T \lambda, \tag{8.1}$$

$$K_{rc}^T u_r + K_{cc} u_{bc} = f_{bc} - B_{bc}^T \lambda, \tag{8.2}$$

$$B_r u_r + B_{bc} u_{bc} \leq 0. \tag{8.3}$$

From (8.1) $u_r = K_{rr}^{-1}(f_r - B_r^T \lambda - K_{rc} u_{bc})$, we substitute u_r into (8.2) and (8.3) and multiply (8.2) by B_c^T from the left. Using standard FETI-DP notation introduced in Section 6.2 and 6.3 we obtain

$$\begin{array}{ccccccc} -F_{I_{rc}}^T \lambda & + & K_{cc}^* u_c & = & f_c^* & - & B_c^T B_{bc}^T \lambda \\ F_{I_{rr}} \lambda & + & F_{I_{rc}} u_c & \geq & B_r K_{rr}^{-1} f_r & + & B_{bc} B_c u_c \end{array}, \quad \lambda_I \geq 0$$

and after some manipulations problem

$$\begin{array}{ccccccc} F_{I_{rr}} \lambda & + & (F_{I_{rc}} - B_{bc} B_c) u_c & \geq & B_r K_{rr}^{-1} f_r \\ (F_{I_{rc}} - B_{bc} B_c)^T \lambda & - & K_{cc}^* u_c & = & -f_c^* \end{array}, \;\; \lambda_I \geq 0$$

that can be written in matrix form

$$\begin{bmatrix} F_{I_{rr}} & F_{I_{rc}} - B_{bc} B_c \\ F_{I_{rc}}^T - B_c^T B_{bc}^T & -K_{cc}^* \end{bmatrix} \begin{bmatrix} \lambda \\ u_c \end{bmatrix} \begin{array}{c} \geq \\ = \end{array} \begin{bmatrix} B_r K_{rr}^{-1} f_r \\ -f_c^* \end{bmatrix}, \;\; \lambda_I \geq 0. \tag{8.4}$$

Thus we can state the following Proposition 8.1.

Proposition 8.1: *Let us suppose standard FETI-DP notation. If the contact interface contains corner nodes then* $\widehat{F}_{I_{rc}} = F_{I_{rc}} - B_{bc} B_c$ *has to be used instead of* $F_{I_{rc}}$.

The modification is valid also for mortar FE used on the contact interface as in Figure 8.3.

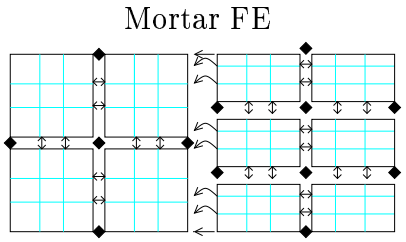


Figure 8.3: Modification of FETI-DP with corners on contact zone for mortar FE

8.2 FETI-DP for coercive contact problems

Final FETI-DP problem having corners on the contact zone after the modification of matrix $\widehat{F}_{I_{rc}}$ and elimination u_c from (8.4) reads

$$\min \frac{1}{2}\lambda^T F\lambda - \lambda^T d \quad \text{s.t.} \quad \lambda_I \geq 0,$$

$$F = F_{I_{rr}} + \widehat{F}_{I_{rc}} K_{cc}^{*-1} \widehat{F}_{I_{rc}}^T; \qquad d = B_r K_{rr}^{-1} f_r - \widehat{F}_{I_{rc}} K_{cc}^{*-1} f_c^*.$$

This problem is then solved efficiently by QPMPGP. The rate of convergence of this algorithm can be bounded in terms of the spectral condition number of Hessian of the QP problem, and therefore the scalability of the resulting algorithm can be established, see Dostál, Horák, Stefanica [14], [15]. Numerical experiments are presented in Section 13.2, including those with changing density of Lagrange multipliers and corners on the contact zone for coercive contact problem.

8.3 FETI-DP for semicoercive contact problems

Following similar approach as for coercive FETI-DP, including the modification of matrix $\widehat{F}_{I_{rc}}$, u_c must satisfy $K_{cc}^* u_c - \widehat{F}_{I_{rc}}^T \lambda - f_c^* = 0 \Leftrightarrow \widehat{F}_{I_{rc}}^T \lambda + f_c^* \in \operatorname{Im} K_{cc}^* \Leftrightarrow \widehat{F}_{I_{rc}}^T \lambda + f_c^* \perp \operatorname{Ker} K_{cc}^*$ with solvability condition $\widetilde{R_c}^T \left(\widehat{F}_{I_{rc}}^T \lambda + f_c^* \right) = 0$, Then $u_c = K_{cc}^{*\dagger} \left(\widehat{F}_{I_{rc}}^T \lambda + f_c^* \right) + \widetilde{R_c} \alpha$.

FETI-DP problem then reads

$$\min \frac{1}{2}\lambda^T F\lambda - \lambda^T d \text{ s.t. } \lambda_I \geq 0 \text{ and } \widetilde{R_c}^T \left(\widehat{F}_{I_{rc}}^T \lambda + f_c^* \right) = 0.$$

After modification: $\widetilde{G} = \widetilde{R_c}^T \widehat{F}_{I_{rc}}^T, \widetilde{e} = -\widetilde{R_c}^T f_c^*, G = T\widetilde{G}, e = T\widetilde{e}$ (T defines orthonormalization of rows of $\widetilde{G}$) and homogenization - choice of particular solution $\widetilde{\lambda}$ satisfying $\widetilde{\lambda}_I \geq 0$ and $G\widetilde{\lambda} = e$ and denoting $d = \widetilde{d} - F\widetilde{\lambda}$, it results in

$$\min \frac{1}{2}\lambda^T F\lambda - \lambda^T d \text{ s.t. } \lambda_I \geq -\widetilde{\lambda}_I \text{ and } G\lambda = 0.$$

Orthogonal projectors $Q = G^T G$ and $P = I - Q$ on $\operatorname{Im} G^T$ and $\operatorname{Ker} G$ improve convergence, so that final problem with incorporated matrix $\widehat{F}_{I_{rc}}$ is

$$\min \frac{1}{2}\lambda^T PFP\lambda - \lambda^T Pd \text{ s.t. } \lambda_I \geq -\widetilde{\lambda}_I \text{ and } G\lambda = 0.$$

Numerical experiments are presented in Section 13.2.

Chapter 9

Normalization of mortar conditions and bounds on spectra

In this chapter I would like to emphasize my important observation which is the key ingredient of numerical scalability of FETI based methods with mortars. Norm of each mortar row $B_{\gamma_m} = [...M_{\gamma_m}... - N_{\gamma_m}...]$ is of the order of the mesh size h_{γ_m} on γ_m, while every row of nodal condition has norm of order 1. It is important to use in our algorithms matrices with normalized rows to have unit norm.

Lemmas, theorems, proofs follow and more details may be found in [15], [14], [25].

Theorem 9.1: *If F and P denote the matrices of the problem (6.9, 10.5) (generated either by FETI-1 or TFETI-1), then the following spectral bounds hold:*

$$C \leq \lambda_{\min}(PFP|\mathrm{Im}P) \leq \lambda_{\max}(PFP|\mathrm{Im}P) \leq ||F|| \leq C\frac{H}{h}; \sigma(PFP|\mathrm{Im}P) \leq C\frac{H}{h}.$$

If F denotes the matrix of the problem (6.12) generated by FETI–DP for the coercive problem, then the following spectral bounds hold:

$$C \leq \lambda_{\min}(F) \leq \lambda_{\max}(F) \leq ||F|| \leq C\left(\frac{H}{h}\right)^2; \sigma(F) \leq C\left(\frac{H}{h}\right)^2.$$

If F and P denote the matrices of the problem (6.21) generated by FETI–DP for the semicoercive problem, then the following spectral bounds hold:

$$C \leq \lambda_{\min}(PFP|\mathrm{Im}P) \leq \lambda_{\max}(PFP|\mathrm{Im}P) \leq ||F|| \leq C\left(\frac{H}{h}\right)^2; \sigma(PFP|\mathrm{Im}P) \leq C\left(\frac{H}{h}\right)^2.$$

Proof: See [28, 23, 14, 25]. □

We showed theoretically here in Theorem 9.3 and experimentally in Section 13.4 that the FETI–DP algorithm using the nonnormalized matrix $B_I^{\#}$ instead of normalized B_I results in a nonscalable algorithm.

Let W be a discrete space constructed as follows: each domain Ω^i, $i = 1, 2$, is partitioned on a rectangular grid into subdomains $\Omega^{i,1}, \ldots, \Omega^{i,p_i}$, with $p_i > 1$. Let H be the diameter of the partitions. The subdomain grids do not necessarily match across the potential contact interface Γ_c. The restrictions of W to Ω^1 and Ω^2 are Q_1 finite element spaces corresponding to the rectangular grids in Ω^1 and Ω^2. We assume that these two Q_1 elements have comparable mesh sizes, on the order of h. We call a crosspoint either a corner that belongs to four subdomains, or a corner that belongs to two subdomains. Since the mortar conditions are only related to interior nodes on the nonmortar edges, two extra conditions are necessary to enforce nonpenetration at the first and last nodes on Γ_c.

We derive bounds on the spectrum of F that will be used for the convergence analysis of the algorithm required to solve the bound constrained QP problem (6.12).

Let $\overline{B} = [B_r \; 0]$ and let $\overline{K}$ be the stiffness matrix corresponding to the model problem on a finite element discretization where continuity is required at the corners but no other continuity is required across the subdomain edges. From inverse inequalities and Poincaré's inequality, it follows that

$$\frac{C}{H^2}||w||^2_{L^2(\Omega)} \;\leq\; \langle \overline{K}w, w\rangle \;\leq\; \frac{C}{h^2}||w||^2_{L^2(\Omega)}, \quad \forall w \in W. \tag{9.1}$$

C is a generic constant independent of h, H, and the number of subdomains in the partitions of Ω^1 and Ω^2.

Note that $F = \overline{B}\;\overline{K}^{-1}\;\overline{B}^T$ and therefore we find as in Lemma 4.3 of [20] that

$$\langle F\lambda, \lambda\rangle \;=\; \sup_{w\in W} \frac{\langle \overline{B}^T\lambda, \lambda\rangle^2}{\langle \overline{K}w, w\rangle}. \tag{9.2}$$

Let $||w||_{l^2}$ and $||\lambda||_{l^2}$ be the Euclidean norms of the primal and dual variables w and λ, respectively. Since w is a finite element function,

$$||w||^2_{L^2(\Omega)} \;\approx\; h^2||w||^2_{l^2}, \tag{9.3}$$

and, in particular,

$$||\overline{B}^T\lambda||^2_{L^2(\Omega)} \approx h^2||\overline{B}^T\lambda||^2_{l^2}. \tag{9.4}$$

From (9.1), (9.2), and (9.3), we find that

$$\begin{aligned}\langle F\lambda, \lambda\rangle \;&\leq\; CH^2 \sup_{w\in W} \frac{\langle \overline{B}^T\lambda, w\rangle^2}{||w||^2_{L^2(\Omega)}} \;\leq\; CH^2||\overline{B}^T\lambda||^2_{l^2} \sup_{w\in W} \frac{||w||^2_{l^2}}{||w||^2_{L^2(\Omega)}} \\ &\leq\; C\left(\frac{H}{h}\right)^2 ||\overline{B}^T\lambda||^2_{l^2}.\end{aligned} \tag{9.5}$$

Let $w_0 = \overline{B}^T\lambda$. From (9.1) and (9.2), and using (9.4), it follows that

$$\begin{aligned}\langle F\lambda, \lambda\rangle \;&\geq\; Ch^2 \sup_{w\in W} \frac{\langle \overline{B}^T\lambda, w\rangle^2}{||w||^2_{L^2(\Omega)}} \;\geq\; Ch^2 \frac{\langle \overline{B}^T\lambda, w_0\rangle^2}{||w_0||^2_{L^2(\Omega)}} \;=\; Ch^2 \frac{||\overline{B}^T\lambda||^4_{l^2}}{||\overline{B}^T\lambda||^2_{L^2(\Omega)}} \\ &\approx\; C||\overline{B}^T\lambda||^2_{l^2}.\end{aligned} \tag{9.6}$$

Recall that

$$\overline{B} \;=\; [B_r\ 0] \;=\; \begin{bmatrix} B_{I,r} & 0 \\ B_{E,r} & 0 \end{bmatrix}.$$

To analyze the importance of normalization for the performance of our algorithm, let us denote by $\overline{B}^{\#}$ the matrix similar to $\overline{B}$ without normalizing the rows of B_I, i.e., using $B_I^{\#}$:

$$\overline{B}^{\#} \;=\; \begin{bmatrix} B_{I,r}^{\#} & 0 \\ B_{E,r} & 0 \end{bmatrix}.$$

Lemma 9.2: *If the rows of B_I are normalized, then*

$$||\overline{B}^T\lambda||_{l^2}^2 \;\approx\; ||\lambda||_{l^2}^2. \tag{9.7}$$

If the rows of B_I are not normalized, then

$$Ch^2||\lambda||_{l^2}^2 \;\leq\; ||(\overline{B}^{\#})^T\lambda||_{l^2}^2 \;\leq\; C||\lambda||_{l^2}^2. \tag{9.8}$$

Proof: Note that

$$\overline{B}^T\lambda \;=\; \begin{bmatrix} B_{I,r}^T\lambda_I \\ 0 \end{bmatrix} \;+\; \begin{bmatrix} B_{E,r}^T\lambda_E \\ 0 \end{bmatrix}.$$

Since the nonzero primal variables corresponding to $B_{I,r}^T\lambda_I$ are all on Γ_c, while those corresponding to $B_{E,r}^T\lambda_E$ are all on the interface of the partitions of Ω^1 and Ω^2, we obtain that

$$||\overline{B}^T\lambda||_{l^2}^2 \;=\; ||B_{I,r}^T\lambda_I||_{l^2}^2 \;+\; ||B_{E,r}^T\lambda_E||_{l^2}^2. \tag{9.9}$$

The entries in each row of $B_{E,r}^T$ are 0, except for two entries which are either 1 or -1. Therefore, $||B_{E,r}^T\lambda_E||_{l^2}^2 \approx ||\lambda_E||_{l^2}^2$ and

$$||\overline{B}^T\lambda||_{l^2}^2 \;\approx\; ||B_{I,r}^T\lambda_I||_{l^2}^2 \;+\; ||\lambda_E||_{l^2}^2. \tag{9.10}$$

Let N_{nm} be the number of nonmortar sides, and let $\gamma(j)$, $j = 1, ..., N_{nm}$, be the nonmortar sides on Γ_c. Let $B_{\gamma(j)}$ be the normalized mortar conditions matrix corresponding to $\gamma(j)$, and let $\lambda_{I,j}$ be the Lagrange multipliers corresponding to $\gamma(j)$. We denote by $B_{\gamma(j),r}$ the part of $B_{\gamma(j)}$ corresponding to the remainder nodes. Then

$$B_{I,r} \;=\; \begin{bmatrix} B_{\gamma(1),r} \\ \cdots \\ B_{\gamma(N_{nm}),r} \end{bmatrix}, \quad \text{and} \quad B_{I,r}^T\,\lambda_I \;=\; \sum_{j=1}^{N_{nm}} B_{\gamma(j),r}^T\,\lambda_{I,j}.$$

For every node on Γ_c, there are at most two vectors with nonzero entries at that nodes out of the vectors $\{B_{\gamma(j),r}^T\,\lambda_{I,j}\}_{j=1,...,N_{nm}}$. Therefore,

$$||B_{I,r}^T\,\lambda_I||_{l^2}^2 \;\leq\; 2\sum_{j=1}^{N_{nm}} ||B_{\gamma(j),r}^T\,\lambda_{I,j}||_{l^2}^2. \tag{9.11}$$

Recall that $B_{\gamma(j),r}$ is the normalized version of $B^{\#}_{\gamma(j),r} = [0 \ M_{\gamma(j)} \ 0 \ -N_{\gamma(j)}]$ corresponding to the remainder nodes. For orthogonal mortars,

$$B_{\gamma(j),r} \ = \ [0 \ \ I_{I,j} \ \ 0 \ \ -P_{\gamma(j)}],$$

where $I_{I,j}$ is the identity matrix of size equal to the number of Lagrange multipliers corresponding to $\gamma(j)$, i.e. the length of $\lambda_{I,j}$, and $P_{\gamma(j)}$ is the mortar projection matrix corresponding to $\gamma(j)$. Thus,

$$B^T_{\gamma(j),r} \ \lambda_{I,j} \ = \ \begin{bmatrix} 0 \\ \lambda_{I,j} \\ 0 \\ -P^T_{\gamma(j)} \ \lambda_{I,j} \end{bmatrix}$$

and therefore

$$||B^T_{\gamma(j),r} \ \lambda_{I,j}||^2_{l^2} \ = \ ||\lambda_{I,j}||^2_{l^2} \ + \ ||P^T_{\gamma(j)} \ \lambda_{I,j}||^2_{l^2}. \tag{9.12}$$

Also, note that

$$||B^T_{I,r} \ \lambda_I||^2_{l^2} \ \geq \ \sum_{j=1}^{N_{nm}} ||\lambda_{I,j}||^2_{l^2} \ = \ ||\lambda_I||^2_{l^2}. \tag{9.13}$$

To estimate the norm of $P^T_{\gamma(j)} \ \lambda_{I,j}$, we use the L^2–stability of the mortar projection. It is easy to see that

$$\begin{aligned} ||P^T_{\gamma(j)} \ \lambda_{I,j}||^2_{l^2} \ &= \ \sup_{\psi \neq 0} \frac{\langle P^T_{\gamma(j)} \ \lambda_{I,j}, \psi \rangle^2}{||\psi||^2_{l^2}} \ = \ \sup_{\psi \neq 0} \frac{\langle \lambda_{I,j}, P_{\gamma(j)} \ \psi \rangle^2}{||\psi||^2_{l^2}} \\ &\leq \ ||\lambda_{I,j}||^2_{l^2} \ \sup_{\psi \neq 0} \frac{||P_{\gamma(j)} \ \psi||^2_{l^2}}{||\psi||^2_{l^2}}. \end{aligned} \tag{9.14}$$

Let $\zeta(j)$ be the union of mortar sides from Γ^2_c opposite $\gamma(j)$. Then ψ corresponds to a vector of nodal values on $\zeta(j)$ and $P_{\gamma(j)}\psi$ is the vector of nodal values of the mortar projection of ψ on $\gamma(j)$. From the stability of the mortar projection we find that

$$||P_{\gamma(j)} \ \psi||^2_{L^2(\gamma(j))} \ \leq \ C||\psi||^2_{L^2(\zeta(j))}.$$

Let $h_{\gamma(j)}$ and $h_{\zeta(j)}$ be the mesh sizes on $\gamma(j)$ and on $\zeta(j)$, respectively. We recall that the meshes across any nonmortar side were assumed to be of order h, i.e. $h_{\zeta(j)}/h_{\gamma(j)}$ is uniformly bounded. Using the fact that ψ and $P_{\gamma(j)} \ \psi$ are traces of first order finite element functions, we obtain

$$||P_{\gamma(j)} \ \psi||^2_{l^2} \ \leq \ C\frac{h_{\zeta(j)}}{h_{\gamma(j)}} \ ||\psi||^2_{l^2} \ \leq \ C||\lambda_{I,j}||^2_{l^2},$$

and therefore, from (9.14), it follows that

$$||P^T_{\gamma(j)} \ \lambda_{I,j}||^2_{l^2} \ \leq \ C||\lambda_{I,j}||^2_{l^2}.$$

Using (9.12), we find that

$$||\lambda_{I,j}||_{l^2}^2 \;\leq\; ||P_{\gamma(j)}^T\;\lambda_{I,j}||_{l^2}^2 \;\leq\; C||\lambda_{I,j}||_{l^2}^2.$$

A bound for the norm of $B_{I,r}^T\lambda_I$ can now be established using (9.11) and (9.13):

$$||\lambda_I||_{l^2}^2 \;\leq\; ||B_{I,r}^T\;\lambda_I||_{l^2}^2 \;\leq\; C||\lambda_I||_{l^2}^2, \tag{9.15}$$

and, using (9.10), we establish (9.7):

$$||\lambda||_{l^2}^2 \;=\; ||\lambda_I||_{l^2}^2 + ||\lambda_E||_{l^2}^2 \;\approx\; ||\overline{B}^T\lambda||_{l^2}^2$$

For the case when nonnormalized mortar conditions are used across the contact interface, i.e. when $B_I^{\#}$ and $\overline{B}^{\#}$ are used instead of B^I and $\overline{B}$, the only difference is in the scaling of the rows of $B_I^{\#}$ by h. In other words, we obtain, instead of (9.15), that

$$Ch^2||\lambda_I||_{l^2}^2 \;\leq\; ||(B_{I,r}^{\#})^T\;\lambda_I||_{l^2}^2 \;\leq\; Ch^2||\lambda_I||_{l^2}^2.$$

Since (9.10) also holds for this case, i.e.

$$||(\overline{B}^{\#})^T\lambda||_{l^2}^2 \;\approx\; ||(B_{I,r}^{\#})^T\lambda_I||_{l^2}^2 \;+\; ||\lambda_E||_{l^2}^2,$$

we conclude that (9.8) is established. □

Theorem 9.3: *Let $F^{\#} = \overline{B}^{\#}\;\overline{K}^{-1}\;(\overline{B}^{\#})^T$, the operator corresponding to F generated by FETI–DP for the coercive problem, if the nonnormalized mortar inequality matrix $B_I^{\#}$ is used in our algorithm instead of B_I. Then the condition number of $F^{\#}$ deteriorates as follows:*

$$Ch^2 \leq \lambda_{min}(F^{\#}) \leq \lambda_{max}(F^{\#}) \leq ||F^{\#}|| \leq C\left(\frac{H}{h}\right)^2;\;\; \kappa(F^{\#}) \leq \frac{C}{h^2}\left(\frac{H}{h}\right)^2. \tag{9.16}$$

Proof: From (9.6) and (9.5) we find that

$$C\frac{||\overline{B}^T\lambda||_{l^2}^2}{||\lambda||_{l^2}^2} \;\leq\; \lambda_{min}(F) \;\leq\; \lambda_{max}(F) \;\leq\; C\left(\frac{H}{h}\right)^2\frac{||\overline{B}^T\lambda||_{l^2}^2}{||\lambda||_{l^2}^2}. \tag{9.17}$$

Then corresponding part of Theorem 9.1 follows from (9.7) of Lemma 9.2. A similar inequality to (9.17) also holds for $F^{\#}$, and (9.16) follows as before from (9.8) of Lemma 9.2. □

Chapter 10

TFETI-1 - an easier implementable variant of the FETI-1 method

Implementation of the FETI-1 method into general purpose packages requires an effective method for automatic identification of the kernels of the stiffness matrices of the subdomains as these kernels are used both in elimination of the primal variables and in definition of the natural coarse grid projectors. An effective method based on combination of the Cholesky factorization and the singular value decomposition was proposed by Farhat and Gérardin [26]. However, it turns out that it is still quite difficult to determine the kernels reliably in the presence of rounding errors. This was one of the motivations that led to development of the FETI-DP method, which enforces the continuity of displacements at the corners on primal level so that the stiffness matrices of the subdomains of the FETI-DP method are invertible. However, even though FETI-DP may be efficiently preconditioned so that it scales better than the original FETI-1 for plates and shells, the coarse grid defined by the corners without additional preconditioning is less efficient [14, 25] than that defined by the rigid body motions, which is important for some applications [23, 14, 25], and the FETI-DP method is more difficult to implement as it requires special treatment of the corners. More discussions concerning the FETI-1 and FETI-DP methods may be found e.g. in [31].

TFETI-1 - a new variant of the FETI method for numerical solution of elliptic PDE is presented in joint work [21]. The basic idea is to simplify inversion of the stiffness matrices of subdomains by using Lagrange multipliers not only for gluing the subdomains along the auxiliary interfaces, but also for implementation of the Dirichlet boundary conditions. Results of numerical experiments are presented and indicate that the new method may be even more efficient than the original FETI-1.

10.1 TFETI-1 and model linear problem in 2D

Let us consider linear 2D model boundary value problem of membrane pressed down by force with density -1 as shown in Figure 10.1. It is assumed that the boundary Γ is

decomposed into two disjoint parts Γ_u and Γ_f, $\Gamma = \overline{\Gamma}_u \cup \overline{\Gamma}_f$ and that there are prescribed Dirichlet boundary conditions on Γ_u and Neumann boundary conditions on Γ_f as in Figure 10.1a. The Dirichlet boundary conditions represent the prescribed displacements and the Neumann conditions represent the surface tractions.

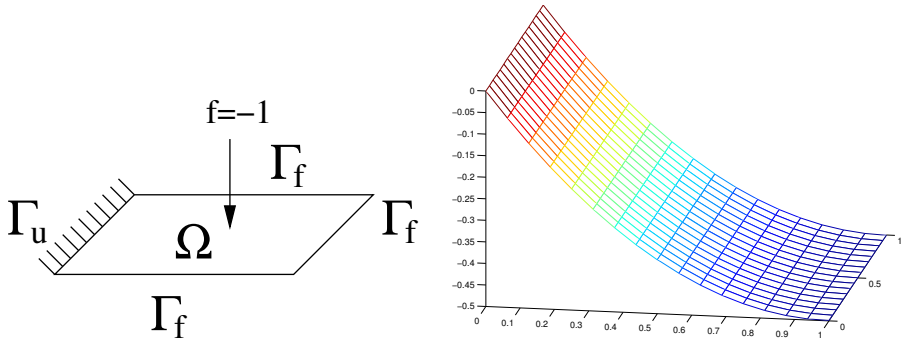


Figure 10.1: a) 2D model problem b) solution of this problem

To apply the FETI-1 based domain decomposition let us partition Ω into N_s subdomains Ω^s as in Figure 10.2 and we denote by K^s, f^s, u^s and B^s, respectively the subdomain stiffness matrix, the subdomain force and displacement vectors and the signed matrix with entries -1, 0, 1 describing the subdomain interconnectivity. We shall get the discretized problem

$$\min \frac{1}{2} u^T K u - u^T f \quad \text{s. t.} \quad Bu = 0 \tag{10.1}$$

with the bock–diagonal stiffness matrix

$$K = \begin{bmatrix} K^1 & & \\ & \ddots & \\ & & K^{N_s} \end{bmatrix}, \quad f = \begin{bmatrix} f^1 \\ \vdots \\ f^{N_s} \end{bmatrix}, u = \begin{bmatrix} u^1 \\ \vdots \\ u^{N_s} \end{bmatrix}, \quad B = [B^1, \ldots, B^{N_s}]. \tag{10.2}$$

The original FETI-1 method assumes that the boundary subdomains inherit the Dirichlet conditions from the original problem as in Figure 10.3a, so that the defect of the stiffness matrices K^s may vary from zero corresponding to the boundary subdomains with sufficient Dirichlet data to the maximum corresponding to the interior floating subdomains.

The basic idea of TFETI-1 (Total-FETI-1) is to keep all the subdomain stiffness matrices K^s as if there were no prescribed displacements and to enhance the prescribed displacements into the matrix of constraints B. To enhance the boundary conditions like $u_i = 0$, just append the row b with all the entries equal to zero except $b_i = 1$. The prescribed displacements will be enforced by the Lagrange multipliers which may be interpreted as forces as in Figure 10.3b. An immediate result of this procedure is that all the subdomain stiffness matrices will have known and typically the same defect. For

example, the defect of each K^s for 2D and 3D elasticity will be equal to three and six, respectively. The remaining procedure is exactly the same as described for FETI-1, the key point is that the kernels R^s of the locall stiffness matrices K^s are known and can be formed directly. For example , if the subdomain Ω^s of a 2D elasticity problem is discretized by means of n_s nodes with the coordinates $(x_i, y_i),\ i = 1, \ldots, n_s$, then

$$R^{s^T} = [(R^{s^1})^T, \ldots, (R^{s^{n_s}})^T]^T, \quad R^{s^i} = \begin{bmatrix} 1 & 0 & -y_i \\ 0 & 1 & x_i \end{bmatrix},\ i = 1, \ldots, n_s. \tag{10.3}$$

Using R^s, we can easily assemble the block–diagonal basis R of the kernel of K as

$$R = \begin{bmatrix} R^1 & & \\ & \ddots & \\ & & R^{N_s} \end{bmatrix}. \tag{10.4}$$

The critical point of evaluation of $K^\dagger x$, the determination of the ranks of the subdomain stiffness matrices K^s, is trivial when the TFETI-1 procedure is applied. The final problem reads

$$\min \quad \frac{1}{2}\lambda^T PFP\lambda - \lambda^T Pd \quad \text{s.t} \quad G\lambda = 0, \tag{10.5}$$

with the same notation as in FETI-1 and may be solved effectively by the conjugate gradient method as the proof of the classical estimate by Farhat, Mandel and Roux ([28], Theorem 3.2)

$$\kappa(PFP|\mathrm{Im}P) \leq C\frac{H}{h} \tag{10.6}$$

of the spectral condition number κ of the restriction of PFP to the range of P by the ratio of the decomposition parameter H and the discretization parameter h remains valid for TFETI-1.

Closer inspection of the proof [28] of (10.6) reveals that the constants in the bound (10.6) may be in many cases more favorable for TFETI-1 than for classical FETI-1. The reason is that the proof of (10.6) is based on estimates of bounds of the eigenvalues of the regular parts of the subdomain matrices K^s that will be in many cases more favorable than those generated by FETI-1. For example, if Ω is decomposed into identical subdomains, introduction of the Dirichlet boundary conditions into the boundary subdomains can only increase these local estimates. From a different point of view, the slightly faster rate of convergence of TFETI-1 than that of FETI-1 should not be surprising as TFETI-1 generates larger kernel of the global stiffness matrix K that have similar role as the coarse grid in multigrid methods. A certain drawback of the procedure is that the dimension of the coarse problem is larger, so that it may happen that the number of the iterations may be slightly larger than that required by the classical FETI-1.

Similarly to the classical FETI-1 method, the performance of the method may be improved by standard FETI preconditioners [30, 31]. The results are useful also for solving contact problems by FETI based methods [23].

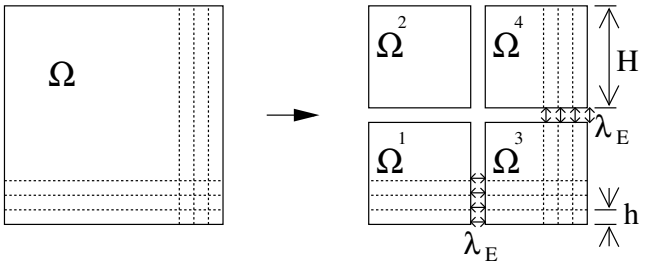


Figure 10.2: Decomposition and discretization of the 2D model problem

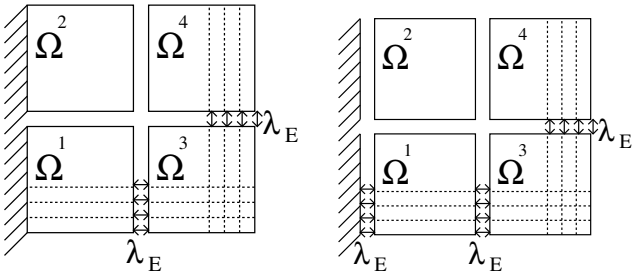


Figure 10.3: a) The idea of original FETI-1 b) the idea of TFETI-1

10.2 TFETI-1 and model contact problem in 2D

The above described procedure for linear problem adapted to contact problems results in

$$\min \frac{1}{2}\lambda^T PFP\lambda - \lambda^T Pd \text{ s.t. } \lambda_I \geq -\widetilde{\lambda}_I \text{ and } G\lambda = 0.$$

In this section and next one 2D and 3D model contact problems are introduced, that demonstrate cooperation of our department with Stanford University in California, USA and Boulder University in Colorado, USA. The described TFETI-1 method and algorithms were incorporated during my and Vít Vondrák's stay in Stanford into software developed by group of Prof. Farhat and used for computation of more realistic problems.

The 2D problem involves 6 rectangles in mutual contact as it is depicted on the Figure 10.4. The left rectangles are fixed on the left side (blue arrows) while the right ones are free and they are loaded (red arrows) such a way that the problem has unique solution. Numerical experiments are depicted in Section 13.3.2.

10.3 TFETI-1 and model contact problem in 3D

The 3D model problem consists of 2 bricks in mutual contact, see Figure 10.5. The bottom brick is fixed in all degrees of freedom while the upper one is fixed only in such a way, that only vertical rigid body movement is allowed. We have analyzed 2 cases. The

first one, with matching grid on the contact interface prescribes node-to-node contact conditions. The second one allows nonmatching grids and the mortar elements were used for assembling of contact conditions. Numerical experiments are in Section 13.3.2.

Conforming decomp., nodal discret. 2D problem: stress distribution

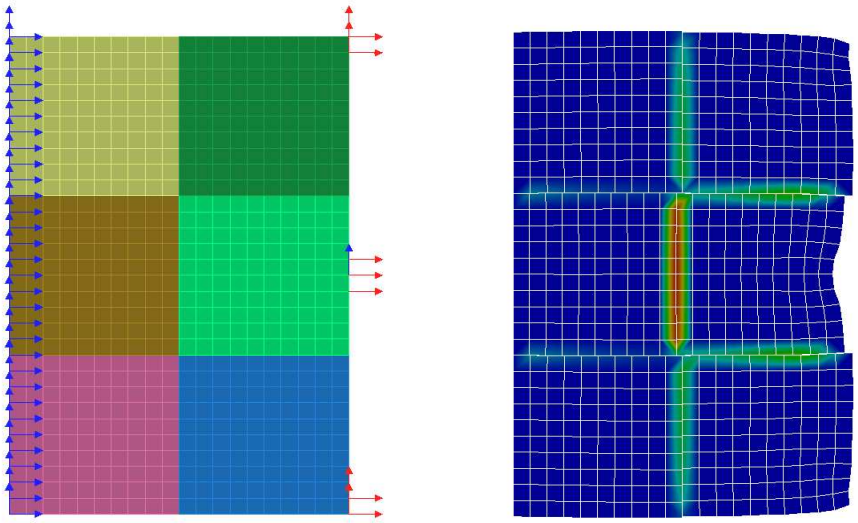


Figure 10.4: 2D model contact problem of 6 rectangles - conforming decompositions and nodal discretization, its solution

Conforming decomp., nodal discret. Nonconforming decomp., mortar discret.

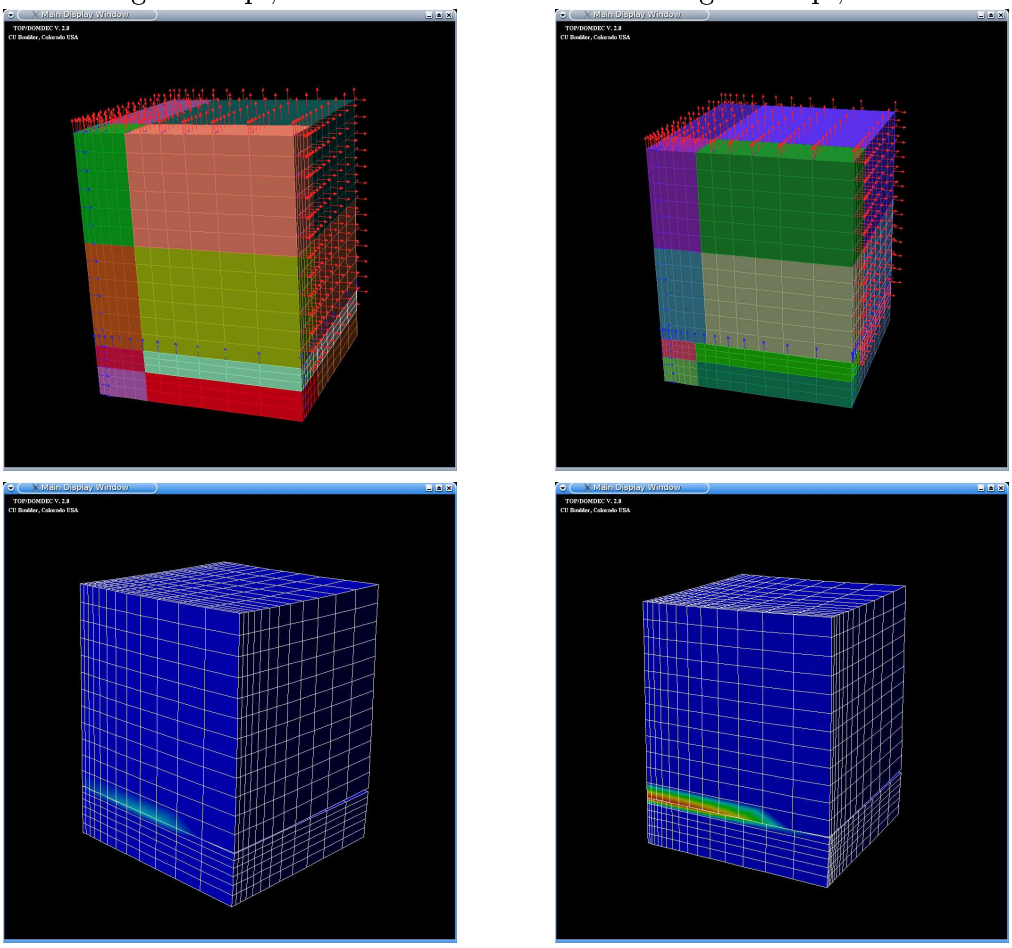


Figure 10.5: 3D model problem of 2 bricks - conforming and nonconforming decompositions, nodal and mortar discretizations, and solutions

Chapter 11

MFETI-DP - multilevel variant of the FETI-DP method

During my stay at the Department of aerospace engineering in Boulder in Colorado I carried out implementation of Prof. Farhat's exciting idea called MFETI-DP - Multilevel-FETI-DP of new generation. The general FETI-DP coarse problem

$$K_{cc}^{*}y = z \quad \text{i.e.} \quad B_c^T(K_{cc} - K_{rc}^T K_{rr}^{-1} K_{rc})B_c y = z, \tag{11.1}$$

which is significantly smaller in comparison to the size of dual problem, can be for very complicated huge problems very large. The factorization and subsequent forward/backward substitutions of this coarse problem becomes the dominant factor in solving the global problem as the number of subdomains becomes large ($N_s > 1000$). Increasing number of subdomains increases degree of parallelism, reduces memory requirements for K^s and factors, reduces CPU for factors and solution of coarse problem by forward/backward substitution, but it increases size of coarse problem.

So an effort to use iterative method instead of direct one arised. Iterative method can be attractive in theory because it reduces the storage requirements, but increases the robustness issue as it introduces an additional convergence criterion at subdomain level, furthermore local solver is embedded in iterative loop in Lagrange multipliers - therefore computational efficiency requires optimizing an iterative local solver for succesive right hand sides.

Why not use the FETI-DP recursively? All assumptions for application of FETI-DP for coarse problem are satisfied. The FETI-DP coarse problem appears in the solution of coercive model linear and contact problem by QPMPGP at three places:

- in the computation of dual right hand side $d = B_r K_{rr}^{-1} f_r - F_{I_{rc}} K_{cc}^{*-1} f_c^*$,
- in each matrix-vector multiplication Fv in QPMPGP, where the dual operator $F = F_{I_{rr}} + F_{I_{rc}} K_{cc}^{*-1} F_{I_{rc}}^T$,
- in the reconstruction formula $u_c = K_{cc}^{*-1}(f_c^* + F_{I_{rc}}^T \lambda)$.

For next explanation let us consider the last case having also nice graphic interpretation, see Figures 11.1, 11.3, 11.4, i.e.

$$K^*_{cc}u_c = f^*_c + F^T_{I_{rc}}\lambda.$$

The corner is the node belonging to four subdomains. It is possible to define corner as the node shared by two and more subdomains, for both linear and contact problems. In case of contact problems we have to use $\widehat{F}_{I_{rc}}$ instead of $F_{I_{rc}}$, see Chapter 8.

11.1 MFETI-DP for coercive linear problems

The coercive linear model problem is similar to coercive contact model problem. We impose continuity conditions instead of nonpenetration on Γ_c. Let

$$\min \frac{1}{2}\lambda_l^T F_l \lambda_l - \lambda_l^T d_l$$

$$F_l = F_{l,I_{rr}} + F_{l,I_{rc}} K^{*-1}_{l,cc} F^T_{l,I_{rc}}; \qquad d_l = B_{l,r} K^{-1}_{l,rr} f_{l,r} - F_{l,I_{rc}} K^{*-1}_{l,cc} f^*_{l,c}$$

denote the FETI-DP problem on the l-th level and

$$K^*_{l,cc}u_{l,c} = f^*_{l,c} + F^T_{l,I_{rc}}\lambda_l$$

its coarse problem of the third type on the l-th level, $l = 1, ..., N_{level}$. For $l = 1, ..., N_{level} - 1$, we process this coarse problem in the same way as $Ku = f$, i.e. by application of FETI-DP method, but we have to decompose it first. The coarse problem on the last level $l = N_{level}$ is solved by the factorization.

The decomposition is defined by mapping of $u_{l,c}$ to u_{l+1}. We split the decomposed nodes on the $l+1$-th level, according to given decomposition, into boundary corner nodes, denoted by bc, remainder nodes, denoted by r, these can be further split into interior nodes, denoted by i, and boundary remainder nodes, denoted by br. The continuity conditions at the new subdomains' corners are enforced by using a global vector of degrees of freedom $u_{l+1,c}$ in primal variable, and at new boundary remainders by using vector of Lagrange mulripliers λ_{l+1} in dual variable. We define B_{l+1},$B_{l+1,c}$, $K^*_{l+1,cc}$, $f^*_{l+1,c}$, etc., and we apply this procedure recursively.

Special consideration is required to the decomposition of the problem and assembling following objects

$$K_{l+1} = \begin{bmatrix} K^{*,1,1}_{l,cc} & 0 & \dots & \dots & \dots & 0 \\ 0 & \ddots & 0 & \dots & \dots & 0 \\ 0 & 0 & K^{*,1,p_{l,1}}_{l,cc} & 0 & \dots & 0 \\ 0 & \dots & 0 & K^{*,2,1}_{l,cc} & 0 & 0 \\ 0 & \dots & \dots & 0 & \ddots & 0 \\ 0 & \dots & \dots & \dots & 0 & K^{*,2,p_{l,2}}_{l,cc} \end{bmatrix}, u_{l+1} = \begin{bmatrix} u^{1,1}_{l,c} \\ \vdots \\ u^{1,p_{l,1}}_{l,c} \\ u^{2,1}_{l,c} \\ \vdots \\ u^{2,p_{l,2}}_{l,c} \end{bmatrix},$$

$$f_{l+1} = \begin{bmatrix} f_{l,c}^{*,1,1} + F_{l,I_{rc}}^{1,1,T} \lambda_l^{1,1} \\ \vdots \\ f_{l,c}^{*,1,p_{l,1}} + F_{l,I_{rc}}^{1,p_{l,1},T} \lambda_l^{1,p_{l,1}} \\ f_{l,c}^{*,2,1} + F_{l,I_{rc}}^{2,1,T} \lambda_l^{2,1} \\ \vdots \\ f_{l,c}^{*,2,p_{l,2}} + F_{l,I_{rc}}^{2,p_{l,2},T} \lambda_l^{2,p_{l,2}} \end{bmatrix},$$

$l = 1, ..., N_{level} - 1$ for the decomposed coarse problem

$$\min \ \frac{1}{2} u_{l+1}^T K_{l+1} u_{l+1} - f_{l+1}^T u_{l+1} \ \text{ s.t. } \ B_{l+1,E} u_{l+1} = 0 \ \text{ and } \ B_{l+1,c} u_{l+1,c} = u_{l+1,bc},$$

playing the role of primal problem and being prepared to the application of FETI-DP resulting in dual problem

$$\min \frac{1}{2} \lambda_{l+1}^T F_{l+1} \lambda_{l+1} - \lambda_{l+1}^T d_{l+1}.$$

Let $K_1 = K$, $f_1 = f$, $u_1 = u$, $B_{1,E} = B_E$, $B_{1,c} = B_c$ and assemble following objects according to formulae:

$$K_{l,cc}^{*,j,k} = \sum_{s \in S_l^{j,k}} B_{l,c}^{j,s,T} (K_{l,cc}^{j,s} - K_{l,rc}^{j,s,T} K_{l,rr}^{j,s,-1} K_{l,rc}^{j,s}) B_{l,c}^{j,s},$$

$$f_{l,c}^{*,j,k} = \sum_{s \in S_l^{j,k}} B_{l,c}^{j,s,T} (f_{l,bc}^{j,s} - K_{l,rc}^{j,s,T} K_{l,rr}^{j,s,-1} f_{l,r}^{j,s}),$$

$$F_{l,I_{rr}}^{j,k} = \sum_{s \in S_l^{j,k}} B_{l,r}^{j,s} K_{l,rr}^{j,s,-1} B_{l,r}^{j,s,T},$$

$$F_{l,I_{rc}}^{j,k} = \sum_{s \in S_l^{j,k}} B_{l,r}^{j,s} K_{l,rr}^{j,s,-1} K_{l,rc}^{j,s} B_{l,c}^{j,s}, \ j = 1, 2,$$

where $S_l^{j,k}$ contains indices s of subdomains $\Omega_l^{j,s}$ of j-th domain Ω_l^j on $l-$th level, which contribute to matrices and vectors of new subdomain $\Omega_{l+1}^{j,k}$ on $l+1-$th level.

Once the problem is solved, we map the solution $u_{l+1}$to $u_{l,c}$. As in multigrid methods we can perform various courses through the levels - V,W cycles and their combinations - as illustrated in Figure 11.2. These mappings can be viewed as operators working on the hierarchy of spaces W_l of P1 or Q1 finite element functions with characteristic mesh size h_l and subdomain size H_l.

I implemented MFETI-DP and run numerical experiments comparing finite and iterative solution of coarse problems of the third type including inexact solution with specified number of CG iterations in simple V-cycle on two levels, see Section 13.4.1.

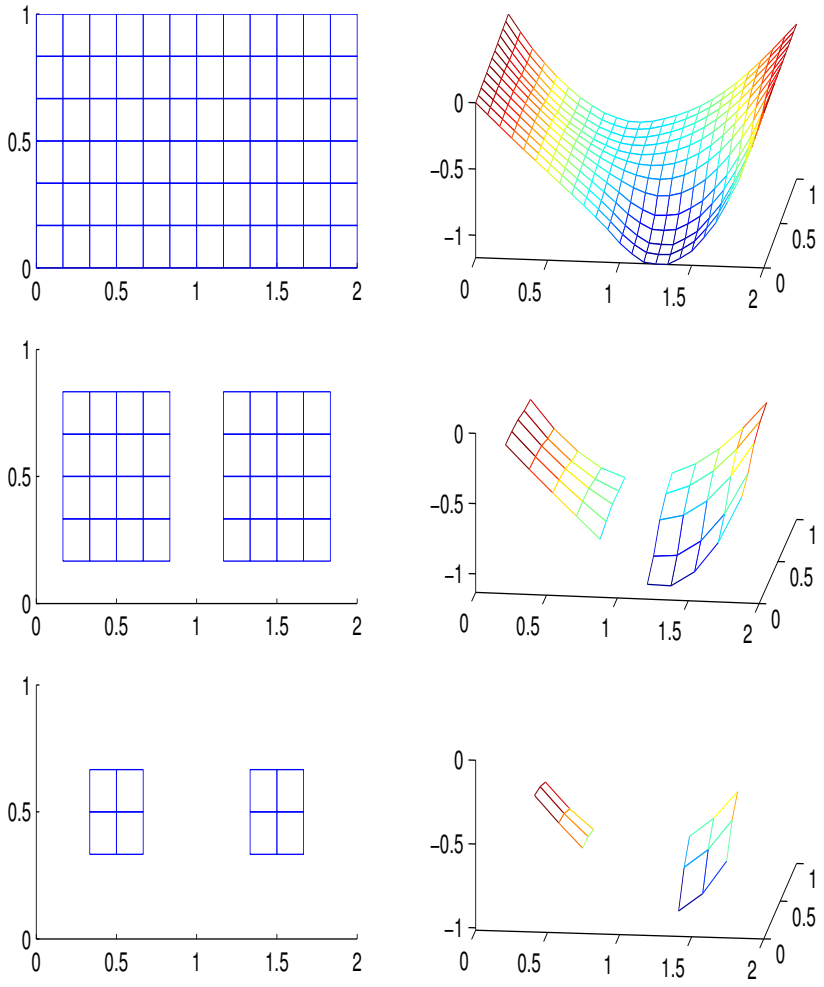


Figure 11.1: Illustration of MFETI-DP method - subdomains, solutions on three levels

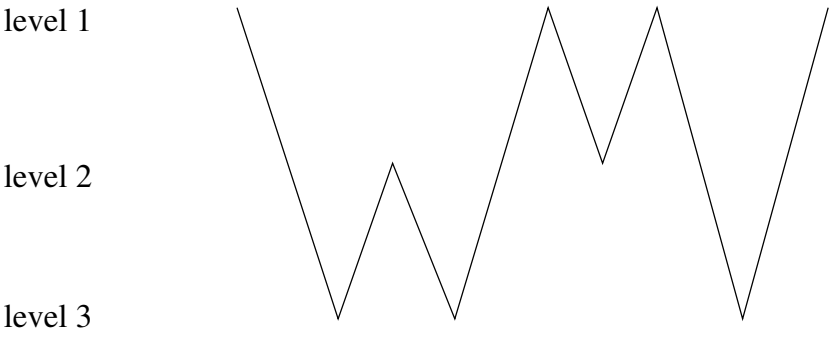


Figure 11.2: Example of course through three levels

11.2 MFETI-DP for coercive contact problems

For coercive contact problems is the situation practically the same. The dual problem on the first level is only bound constrained

$$\min \frac{1}{2}\lambda_1^T F_1 \lambda_1 - \lambda_1^T d_1 \text{ s.t. } \lambda_{1,I} \geq 0,$$

problems on all other levels $l = 2, ..., N_{level}$ are linear, i.e. unconstrained

$$\min \frac{1}{2}\lambda_l^T F_l \lambda_l - \lambda_l^T d_l,$$

and are solved as described in previous section.

I implemented MFETI-DP for coercive contact problems and run numerical experiments on two levels, see Section 13.4.2.

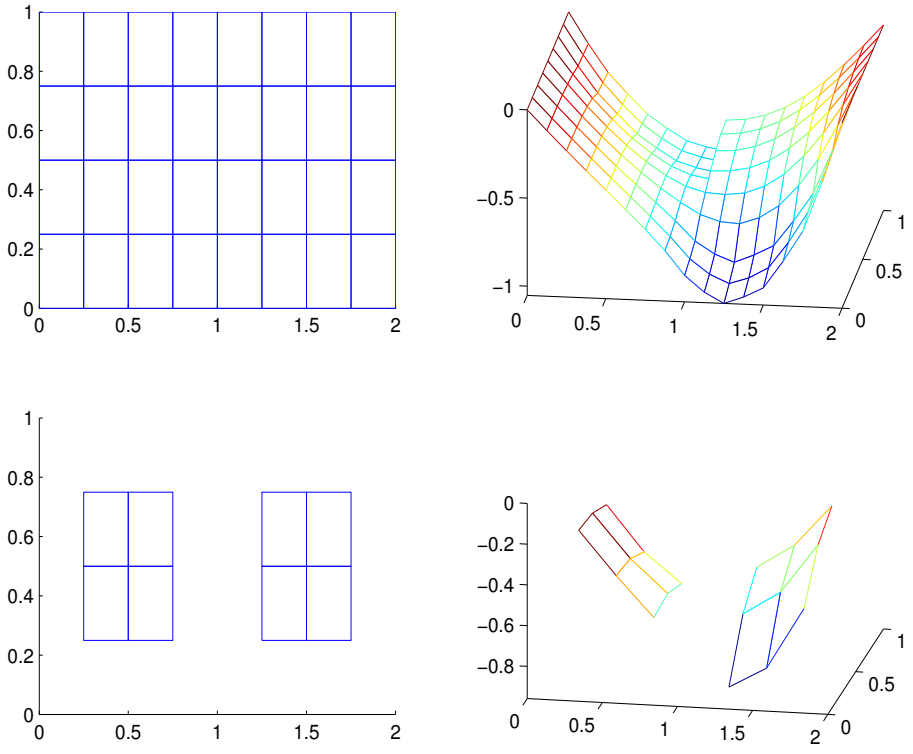


Figure 11.3: Illustration of MFETI-DP for contact coercive model problem - subdomains, solutions on both levels, $H_1 = 1/4, h_1 = 1/8, H_2 = 1/2, h_2 = 1/2$

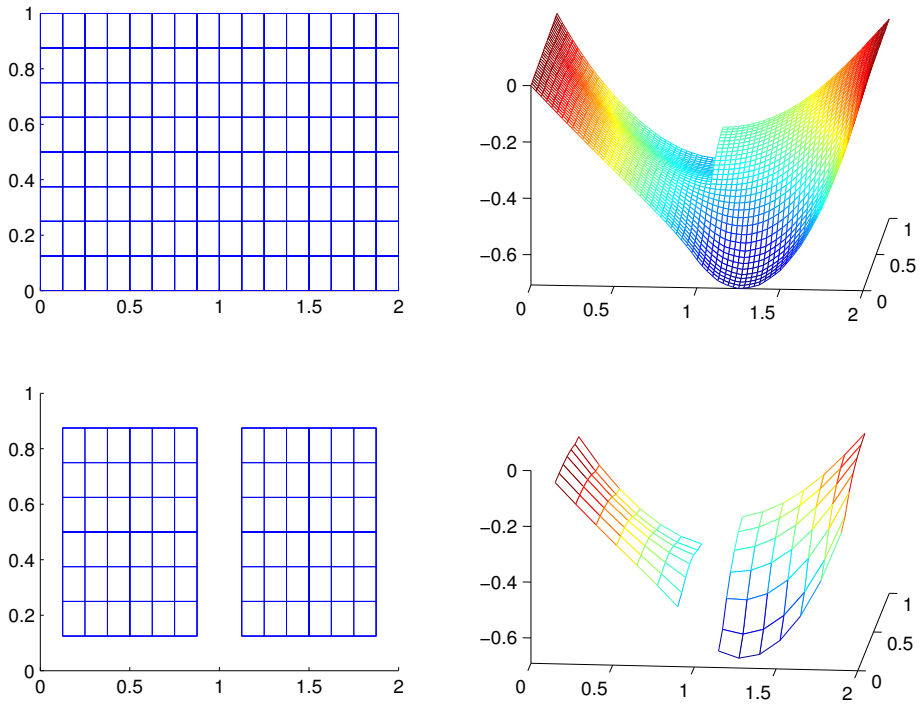


Figure 11.4: Illustration of MFETI-DP for contact coercive model problem - subdomains, solutions on both levels, $H_1 = 1/8, h_1 = 1/32, H_2 = 1/6, h_2 = 1/6$

Chapter 12

Implementations of algorithms

Presented methods were implemented for model problems in Matlab 7. To demonstrate parallel scalability and to run huge numerical experiments it was necessary to implement the code in parallel using PETSc 2.1.22, that was realized during my stays in EPCC (Edinburgh Parallel Computing Centre) in 2001 (TRACS programme) and 2005 (HPC-Europa programme) and at Johannes Kepler University in Linz in 2001 (group of Prof. Ulrich Langer). Following chapter gives a simplified overview of the most relevant phases of the FETI-1 and TFETI-1 algorithm and the description of their parallel implementations. To solve more realistic problems in industry I have participated in cooperation with Colorado University in Boulder in 2004 and Stanford University in California in 2006 (both in groups of father of FETI methods, Prof. Charbel Farhat).

12.1 Matlab's implementation of algorithms

Here is Matlab's code SMALBE algorithm, that is very similar to ALAPC and uses QPMPGP in Step 1:

```
% SMALBE
% Step 0
epsr = 1.e-6*norm(b);
Gama = 1;
ncg = 0; ne = 0; nout = 0; ncg_max = 100;
u = max(B*f/2,c);               % Initial lambda
c(niq) = -Inf*ones(length(niq),1);
lAl = rho;                      % lAl is the norm of the matrix
alfa=1/lAl;                     % alpha_bar...can be computed more precis.by Reigh.quot.
J = (u > c);
M=1; betarho=10; ny=norm(P*d); mi=zeros(size(G,1),1);
Lag=0.5*u'*A*u-b'*u;
gp=ones(length(u));
Gu=G*u;
% Step 1,2
while norm(gp)>epsr
 lAl = rho;
```

```
 alfa=1/rho;
 % Gradient splitting of g=-r, gf=gradient free(fi), gr=gradient free
 % reduced (fired), gc=gradient chopped or cut (beta), gp=gf+gc=projected grad.
 g = A*u-b; gf = J.*g; gc = min((~J).*g,0); gr = min(lAl*J.*(u-c), gf); gp = gf + gc;
 p=gf;

 while norm(gp)>min(M*norm(Gu),ny)
   if gc'*gc <= Gama*gr'*gf
      % Proportional iteration. Trial conjugate gradient step.
      Ap=A*p; rtp=g'*p; pAp=p'*Ap; acg=rtp/pAp;  yy=u-acg*p; ncg=ncg+1;
      if all(yy>=c)
         %Conjugate gradient step
         u=yy; g=g-acg*Ap;
         gf = J.*g; gc = min((~J).*g,0); gr = min(lAl*J.*(u-c), gf); gp = gf + gc;
         beta=gf'*Ap/pAp;
         p=gf-beta*p;
      else
         %partial conjugate gradient step
         a=1/min((J.*p)./(J.*(u-c)+(~J)));
         u=max(u-a*p,c); J=(u>c); g=g-a*Ap;
         gf = J.*g; gr = min(lAl*J.*(u-c), gf);
         %expansion step
         u=max(u-1.01*alfa*gr,c); J=(u>c); g=A*u-b;
         gf = J.*g; gc = min((~J).*g,0); gr = min(lAl*J.*(u-c), gf); gp = gf + gc;
         p=gf; ne=ne+1;
      end
   else
      %Proportioning step
      Ap=A*gc; acg=(gc'*g)/(gc'*Ap);
      u=u-acg*gc; J=(u>c); g=g-acg*Ap;
      gf = J.*g; gc = min((~J).*g,0); gr = min(lAl*J.*(u-c), gf); gp = gf + gc;
      p=gf; ncg=ncg+1;
   end
   if ncg >= ncg_max   break;   end
 end
 % Step 3
 Gu=G*u;
 mi=mi+rho*Gu;
 % Step 4
 Lag_old=Lag;
 Lag=0.5*u'*A*u-b'*u;
 if (Lag < Lag_old + 0.5*rho*(Gu'*Gu))
     rho = betarho * rho;
 end;
 % Step 5
 A=P*F*P+rho*Q;
 b=b0-G'*mi;
 nout = nout + 1;
 if ncg >= ncg_max  break;   end
end
```

12.2 Parallel implementation

12.2.1 PETSc - tool for parallel implementation

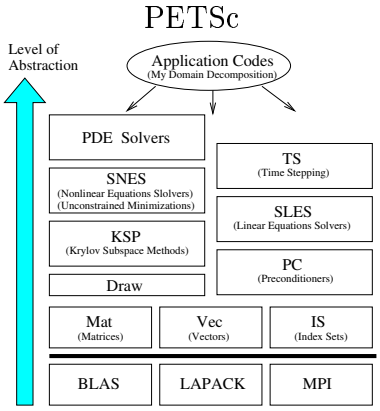


Figure 12.1: Organization of the PETSc library

Before the description of the parallel implementation I would like to introduce first a tool, that was instrumental in all the realization - its name is PETSc. What does the word "PETSc" mean? The name PETSc is the abbreviation formed by the initial letters of the Portable Extensible Toolkit for Scientific Computation, developed by Argonne National Laboratory by group formed by Balay, Gropp, McInnes and Smith [10].

PETSc is a suite of data structures and routines that provide the building blocks for the implementation of large-scale application codes on parallel and sequential computers especially used for the numerical solution of PDEs and related problems on high-performance computers. A number of parallel linear and nonlinear equation solvers and unconstrained minimization modules using modern programming paradigms enables development of large scientific codes written in C, C++ or Fortran. PETSc uses the MPI standard for all message-passing communication and routines from BLAS, LAPACK, MINPACK, SPARSPAK and BlockSolve95. Technique of object oriented programming provides enormous flexibility and code reuse. The library is hierarchically organized according to level of abstraction (see Figure 12.1). PETSc consists of a variety of components, which manipulate a particular family of objects. These components are index sets, vectors, matrices (sparse and dense), distributed arrays (useful for parallelizing regular grid-based problems), Krylov subspace methods, preconditioners, nonlinear solvers, unconstrained minimization, timesteppers for solving time dependent PDEs, graphics devices, etc.

Each of these components consists of an abstract interface and one or more implementations using particular data structures. Thus PETSc provides clean and effective codes for various phases of solving PDEs, with a uniform approach for each class of problems, as well as a rich environment for modeling scientific applications and for algorithm design and prototyping.

12.2.2 Data structure for FETI-1 and TFETI-1

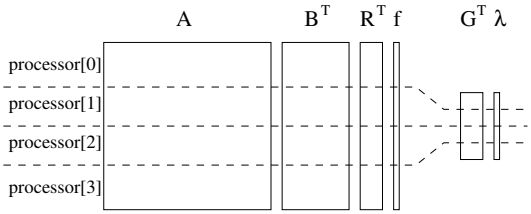


Figure 12.2: Example of data distribution over 4 processors

The program consists of two parts namely generator of input objects and the solver of model problem given by these input objects.

Each processor works with local part associated with its subdomains. For simplification let $X^{[rank]}$ denote this local part or the restriction of object X on current processor specified by $rank$ ($rank = 0, ..., size-1$, with $size$ denoting number of processors being at disposal in communicator for our computation).

Most of computations appearing in this program are purely local and therefore parallelizable, but some operations require data transfers. I distinguish two kinds of these data transfers in my program, namely:

- data transfer that picks up the contributions from all processors and then distributes the complete result to each processor (see Figure 12.3, transfered data are indicated by black color) - such transfer is denoted in description of parallel scheme by $\lceil\cdot\rceil$ (e.g. computation of dot products of parallel distributed vectors, computation of the norm of parallel vector, matrix by vector multiplication $G\lambda$ etc.)

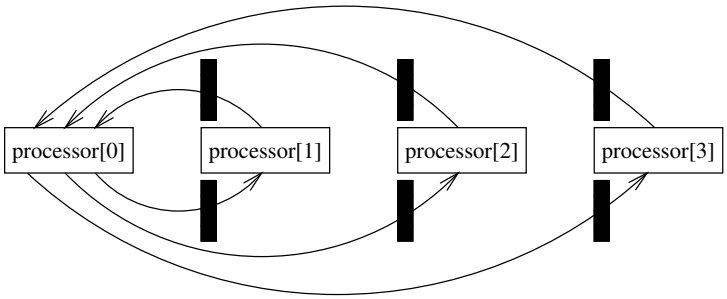


Figure 12.3: Interprocessor communication with data transfer of type $\lceil\cdot\rceil$

- data transfer that picks up the contributions from all processors and then distributes the corresponding parts of result to processors (see Figure 12.4, transfered data are indicated by black color) - such transfer is denoted in description of

parallel scheme by $\lfloor\cdot\rfloor$ (e.g. assembling of sequential vector on one of processors from particular results located on other processors and conversion of this sequential vector by means of scatter functions into parallel vector - matrix by vector multiplication Bf etc.)

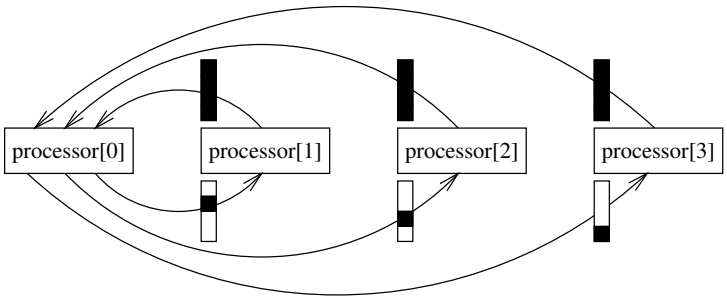


Figure 12.4: Interprocessor communication with data transfer of type $\lfloor\cdot\rfloor$

12.2.3 Objects defining model problem

I would like first to describe a part generating input objects i.e. stiffness matrix K, matrix B describing the interconnectivity of subdomains, vector of forces f and matrix R representing the null space of K. These objects are distributed over the processors, so that their locally stored portions of the same size correspond to relevant subdomains. The data distribution of various types of objects over the processors showing the local portions is then presented in Figure 12.2. Matrices and vectors are simply divided into segments of equal *size*. Let N denote the primal dimension (number of primal variables) and N_{dual} denote the dual dimension (number of dual variables, size of vector of Lagrange multipliers).

The allocation of memory needed for storage of these matrices including distribution is realized by PETSc function

```
MatCreateMPIAIJ(PETSC_COMM_WORLD,int m,int n,int M,int N,
                int dnz,int *dnz,int onz,int *onz,Mat *mat),
```

where `PETSC_COMM_WORLD` is a communicator comprising all processors being at disposal for computation, then follows numbers of local rows `m`, local columns `n`, global rows `M` and global columns `N`, parameters `dnz,*dnz,onz,*onz` deal with a control of dynamic allocation of matrix memory space. Specification `MPI` reminds the fact, that `mat` is parallel matrix, and `AIJ` indicates its sparse format (all the matrices used in program are of this format). If the communicator consists only of one processor then following function is used

```
MatCreateSeqAIJ(PETSC_COMM_SELF,int M,int N,
                int nz,int *nz,Mat *mat)
```

only with indications of global dimensions and other parameters of the same meaning as above.

For vectors the situation is analogous - it is possible to use two functions according to size of communicator, namely

```
VecCreateMPI(PETSC_COMM_WORLD,int n,int N,Vec *vec),
VecCreateSeq(PETSC_COMM_SELF,int N,Vec *vec).
```

So the components defining the model problem include:

- stiffness matrix K - global number of rows N, local number of rows stored on processor $N/size$, global number of columns N, local number of columns stored on processor N,
- matrix B^T for interconnectivity of subdomains - global number of rows N, local number of rows stored on processor $N/size$, global number of columns N_{dual}, local number of columns stored on processor N_{dual},
- matrix R of null space of K - global number of rows N, local number of rows stored on processor $N/size$, global number of columns N_e, local number of columns stored on processor N_e, with N_e denoting the number of floating subdomains,
- vector of forces f - global number of elements N, local number of elements stored on processor $N/size$.

12.2.4 Implementation of dual formulation and modifications

Objects presented in previous section are input parameters for domain decomposition solver. Before the start of algorithm it is necessary to execute preparatory phase consisting of dual formulation and modifications. The descriptions and comments of the realizations follow:

- Computation of matrix
$$F = BK^{\dagger}B^T$$
- the principle of efficient programming is to avoid time-consuming operations including the matrix by matrix multiplication. Fortunately, only matrix by vector multiplication appears in this algorithm and it is performed as $Fvec_d = B(K^{\dagger}(B^T vec_d))$. The action of $K^{\dagger}$ can be evaluated by means of Cholesky decomposition of 2 stiffness submatrices on each processor (factorization of 2 blocks is sufficient in our case, because there are only 2 types of subdomains - fixed and floating), or 1 stiffness matrix in TFETI-1 case, respectively, and the storage requirements of these matrices are also acceptable. These two submatrices can be obtained on each processor by means of PETSc function

```
MatGetSubMatrices(Mat mat,int n,IS *irow,IS *icol,
                  MatGetSubMatrixCall scall,Mat *submat),
```

which exploits arrays of index sets (`IS` is data type containing indices) to extract `n` submatrices from a matrix `mat`, `submat` then points to an array of current matrices. In this case two `IS` are created on each processor - `IS[0]` defines global indices of rows and columns of stiffness submatrix of fixed subdomain, and `IS[1]` defines global indices of rows and columns of stiffness submatrix of floating subdomain (last row and last column of this submatrix are cut off). For purpose of factorization PETSc provides functions

```
MatCholeskyFactor(Mat mat,IS perm,double fill),
```

performing in-place Cholesky factorization of a symmetric matrix, while index sets define permutations of possible orderings and `fill`>1 is the predicted fill in expected in factored matrix, as a ratio to the original fill in. Having factored matrices `L[k]` (such that $\texttt{K}^{\texttt{i}}=\texttt{L[k]L[k]}^{\mathrm{T}}$, where `k=0` if `i` indicates fixed subdomain and `k=1` if `i` indicates floating subdomain), one can then solve system $\texttt{K}^{\texttt{i}}\texttt{x[k]=b[k]}$ by means of the function

```
MatSolve(Mat L[k],Vec b[k],Vec x[k]).
```

If it is necessary to compute product $vec_f = K^{\dagger}vec_f$ (where $vec_f = B^T vec_d$ denotes vector of the same type as vector f, and vec_d vector of the same type as vector d), the procedure is following: on each processor an array `is` of index sets `IS` is created (for one subdomain one index set of global indices of `vecf` of the same size as corresponding `L[k]`), `is[i]` defines the part from vector `vecf` for scatter into vector `b[k]` at positions defined by index set `isb[k]` (`k=0` if `i` indicates fixed subdomain and `k=1` if `i` indicates floating subdomain), while the following functions are used

```
VecScatterCreate(Vec vecf,Vec b[k],IS is[i],IS isb[k],
                 VecScatter *ctx),
VecScatterBegin(Vec vecf,Vec b[k],INSERT_VALUES,
                SCATTER_FORWARD,VecScatter ctx),
VecScatterEnd(Vec vecf,Vec b[k],INSERT_VALUES,
              SCATTER_FORWARD,VecScatter ctx),
```

followed by solution of system `MatSolve(Mat L[k],Vec b[k],Vec x[k])`. Acquired solution `x[k]` is then put back into `vecf` at positions defined by index set `is[i]` using calls of scatter functions with changed arguments

```
VecScatterBegin(Vec x[k],Vec vecf,INSERT_VALUES,
                SCATTER_REVERSE,VecScatter ctx),
```

```
VecScatterEnd(Vec x[k],Vec vecf,INSERT_VALUES,
              SCATTER_REVERSE,VecScatter ctx),
VecScatterDestroy(VecScatter ctx),
```

while the last function destroys a scatter context created by `VecScatterCreate()`. In the case of floating subdomain, we have to set the position in vector `vecf` corresponding to cut off row and column to value 0.0. All these operations concerning $K^{\dagger}$ are performed on each processor with own local portions without any communication. The communication is then required for matrix by vector multiplications $vec_f^{[rank]} = B^{T[rank]} \left[vec_d^{[rank]} \right]$ and $vec_d^{[rank]} = \left\lfloor B^{[rank]} vec_f^{[rank]} \right\rfloor$. The product $Fvec_d$ is computed at least once per CG-iteration. Although it is efficiently performed in parallel way, it dominates the overall time because of large primal dimension.

- Computation of matrix
$$\widetilde{G} = R^T B^T$$
- since the size of the dual problem can be still considerable large, I have decided to parallelize the vector of Lagrange multipliers λ (and of course vectors of the same type denoted as vec_d) and matrix $\widetilde{G}^T$ in such way that each of processors owns nearly the same local portion $\widetilde{G}^{T[rank]}$, namely for vector λ - global number of elements N_{dual}, local number of elements stored on processor $N_{dual}/size$, for matrix $\widetilde{G}^T$- global number of rows N_{dual}, local number of rows stored on processor $N_{dual}/size$, global number of columns N_e, local number of columns stored on processor N_e. Although a matrix by matrix multiplication, as was mentioned above, does not belong to efficient operations, there is no way how to avoid it. Also PETSc provides no function for this multiplication and so nothing else is left, as tranform matrix by matrix multiplication $\widetilde{G}^T = BR$ to matrix by vector multiplication $Bvec_f$, while vec_f is vector created by i-th column of matrix R and obtained by function

```
MatGetColumnVector(Mat R,Vec vecf,int i),
```

the result then forms i-th column of matrix $\widetilde{G}^T$. Fortunately number of columns of matrix R is small and equal to number of floating subdomains N_e. All the computation is done in parallel, the communication is required only during the distribution of these N_e result vectors over the processors in communicator. For more details see the function `ComputeMatG()`.
Parallel scheme:
$$\widetilde{G}_{icol}^{T[rank]} = \left\lfloor B^{[rank]} R_{icol}^{[rank]} \right\rfloor .$$

- Computation of vector
$$\widetilde{d} = BK^{\dagger} f$$

- matrix by vector multiplication is realized according to scheme $K^{\dagger}vec_f$ introduced in item concerning matrix F. Matrix by vector multiplication $Bvec_f$ is performed in parallel but the communication is necessary to convert the result into the parallel vector $\widetilde{d}$ - for more details see function `BLLvec()`.
Parallel scheme:

$$\widetilde{d}^{[rank]} = \left\lfloor B^{[rank]} \left(K^{\dagger [rank]} f^{[rank]} \right) \right\rfloor .$$

- Computation of vector

$$\widetilde{e} = R^T f$$

- in line with the formats of R and f, this matrix by vector multiplication is also parallelized and vector $\widetilde{e}$ is distributed to each processor.
Parallel scheme:

$$\widetilde{e} = \left\lceil R^{T[rank]} f^{[rank]} \right\rceil .$$

All presented operations have to be done in case of basic, projected and projected-preconditioned version. However, for last two named versions it is necessary to execute following modifications:

- Computation of matrix

$$G = T\widetilde{G}$$

with T denoting a nonsingular matrix that defines the orthonormalization of the rows of $\widetilde{G}$. Further acceleration of computation reached via projectors built thanks to matrix G is paid by orthonormalization of columns of matrix $\widetilde{G}^T$. For this purpose the classical Gram-Schmidt algorithm was chosen, that appears more suitable for parallelization of this problem than modified or iterated classical Gram-Schmidt [11] (classical Gram-Schmidt requires half the number of floating-point operations, on parallel computers it can be much faster than modified Gram-Schmidt, and its parallel efficiency equals that of iterated classical Gram-Schmidt). The columns of matrix $\widetilde{G}^T$ are copied into the array $g[\,]$ of vectors of type vec_d (local size $N_{dual}/size$, global size N_{dual}) and process of orthonormalization is performed according to

$$g[i] = g[i] - \sum_{j=0}^{i-1} (g[i]^T g[j]) g[j], \;\; g[i] = \frac{g[i]}{\|g[i]\|}, i = 0, ..., N_e - 1.$$

The obtained vectors form columns of required matrix G^T. For more details see the function `OrthogMatG()`.
Parallel scheme:

$$\begin{array}{l} g[i]^{[rank]} = g[i]^{[rank]} - \sum_{j=0}^{i-1} \left\lceil g[i]^{T[rank]} g[j]^{[rank]} \right\rceil g[j]^{[rank]}, \\ g[i]^{[rank]} = \frac{g[i]^{[rank]}}{\left\lceil \left\| g[i]^{[rank]} \right\| \right\rceil}, i = 0, ..., N_e - 1. \end{array}$$

- Computation of vector

$$e = T\widetilde{e}$$

- problem of finding vector e transforms after some manipulations to problem to solve the small system of equations $\widetilde{G}G^T e = \widetilde{e}$, while the product $\widetilde{G}G^T$ is computed in similar way as product BR - see the function `ComputeMatGGort()`.

- Computation of vector $\widetilde{\lambda}$ that satisfies relation $G\widetilde{\lambda} = e$, but being aware of features of matrix G, the vector $\widetilde{\lambda}$ can be easy obtained as

$$\widetilde{\lambda} = G^T e.$$

Parallel scheme:

$$\widetilde{\lambda}^{[rank]} = G^{T[rank]} e.$$

- Computation of vector

$$d = \widetilde{d} - F\widetilde{\lambda}$$

- vector d is given by difference of vector $\widetilde{d}$ and vector acquired as the above explained product $F\widetilde{\lambda}$.
Parallel scheme:

$$d^{[rank]} = \widetilde{d}^{[rank]} - \left\lfloor B^{[rank]} \left(K^{\dagger[rank]} \left(B^{T[rank]} \left\lceil \widetilde{\lambda}^{[rank]} \right\rceil \right) \right) \right\rfloor .$$

- Creation of active set $\mathcal{A}(\lambda)$ and free set $\mathcal{F}(\lambda)$ - vector J is used for this purpose having ones at positions i belonging to $\mathcal{F}(\lambda)$, where $\lambda_i > c_i$, with $c_i = -\widetilde{\lambda}_i$ for $i \in I$ and $c_i = -\infty$ for $i \in E$, and zeros at positions belonging to $\mathcal{A}(\lambda)$.
Parallel scheme:

$$J_i^{[rank]} = \begin{cases} 0 & \text{for} \quad \lambda_i^{[rank]} \leq c_i^{[rank]} \\ 1 & \text{for} \quad \lambda_i^{[rank]} > c_i^{[rank]} \end{cases} .$$

The course of computation is appreciably influenced by initial approximation of vector λ^0 (comparison of various aproximations is shown in Table 13.6). In my program I use following initial approximations :

1. $\lambda_i^{0,\mathrm{I}} = \max\left(0, -\widetilde{\lambda}_i\right)$,
2. $\lambda_i^{0,\mathrm{II}} = -\widetilde{\lambda}_i$,
3. $\lambda_i^{0,\mathrm{III}} = \max\left(-\widetilde{\lambda}_i, \underline{\lambda_i}\right)$.

We can consider $-\widetilde{\lambda}_I$ as an obstacle. I have observed that

$$\underline{\lambda} = \frac{1}{2} Bf$$

computed according to parallel scheme

$$\underline{\lambda}^{[rank]} = \frac{1}{2} \left\lfloor B^{[rank]} f^{[rank]} \right\rfloor$$

is optimal estimate of resulting vector λ, taking into consideration distribution of forces f on interfaces, relation $B^T\lambda = f$, and orthogonality $BB^T = 2I_B$ with I_B denoting square identity matrix. This third initial approximation leads to significant decrease of CG iterations in comparison to other initial vectors for all problems.

12.3 Implementation to FEM

As was mentioned, with cooperation with Prof. Farhat's group during our stays in Boulder and Stanford arised C++ implementation of algorithms QPMPGP and SMALBE into their software

```
GenFetiDPSolver<double>::solveQP(GenDistrVector<double> &f,
                                 GenDistrVector<double> &u),
GenFetiDPSolver<double>::solveOSTRAVA(GenDistrVector<double> &f,
                                      GenDistrVector<double> &u).
```

These class methods allow to solve 2D and 3D contact FETI-1, FETI-DP and TFETI-1 multibodies problems. Each body performs subdomain for FETI-1 or TFETI-1 method and at the same time it performs domain being decomposed in subdomains by FETI-DP method. So it combines both FETI-1 or TFETI-1 with FETI-DP.

The algorithms had to be modified for the solution of dual problems corresponding to the primal problem

$$\min \ \frac{1}{2}u^T K u - f^T u \ \text{ s.t. } \ B_I u \geq 0 \ \text{ and } \ B_E u = 0$$

differing from that described in this thesis in opposite inequality sign, recall our condition $B_I u \leq 0$.

FETI-1, TFETI-1 and FETI-DP methods result in maximization problem of opposite dual functional with opposite bound

$$\max - \left(\frac{1}{2} x^T A x - b^T x \right) \ \text{ s.t. } \ x_I \leq -\widetilde{x}_I \ \text{ and } \ Cx = 0$$

with $A, C, b, \widetilde{x}$ being defined according to FETI type method. This problem is equivalent to problem introduced in Chapter 5.

Let us briefly mention another differences - there is no G matrix arising by orthonormalization of rows of $\widetilde{G}$, because of its less sparsity. Then $\widetilde{\lambda} = G^T e$ can be reached as $\widetilde{\lambda} = \widetilde{G}^T \left(\widetilde{G}\widetilde{G}^T \right)^{-1} e$.

Dual linear problem is solved by projected conjugate gradient method (PCG) with the initial approximation $\lambda^0 = \widetilde{\lambda}$. Then $\widetilde{G}\lambda^0 = e$ and $\lambda - \lambda^0 \in \operatorname{Ker} \widetilde{G}$; $\widetilde{G}\lambda = e$ is satisfied if all increments $\lambda^k - \lambda^{k-1} \in \operatorname{Ker} \widetilde{G}$. In contrast to the classical CG method, in each iteration of PCG algorithm, the residual and the search directions are projected onto $\operatorname{Ker} \widetilde{G}$. This projection step plays role of coarse problem which is solved in each iteration.

Futhermore for contact problems FETI-DPC was proposed by C. Farhat et al. This algorithm is based on FETI-DP domain decomposition method and uses the Newton-like method which solves the equilibrium equation in Lagrange multipliers in the inner loop, while feasibility of each step is ensured in the outer loop by the primal and dual planning steps. Primal planning step reconstruct primal solution to correct the penetration of bodies, although in our algorithm this is authomatically satisfied by simple bound in dual variable. Dual planning step looks for the search direction to the obstacle of dual variable. The algorithm exploits standard FETI preconditoners, namely the Schur and lumped ones. The additional speedup of convergence is achieved by application Krylov type acceleration scheme. The algorithm exploits a globalization strategy in order to achieve monotonic global convergence.

The efficiency was tested on realistic 2D and 3D contact problems including those with mortars, see Section 13.3.2.

Chapter 13

Numerical experiments

13.1 FETI-1

13.1.1 Semicoercive problem + conforming FE + QPMPGP: Numerical, parallel scalability and optimality of dual penalty

To illustrate the performance of the algorithms, in particular their numerical and parallel scalabilities and optimality of dual penalty, I have implemented algorithms in C exploiting the package PETSc [10]. The experiments were run on SGI Origin 38000 computer courtesy of Johannes Kepler University of Linz: shared memory (MIMD), 128 processors R12000, 400 MHz, 48128 MB RAM, 500 GB disk space, DLT 7000 stacker 70 GB, net: 3x ATM 155 Mb/s, FDDI 1Gb/2, ethetnet 10 Mb/s, and Lomond 52-processor Sun Ultra SPARC-III based system with 900 MHz, 52 GB of shared memory, nominal peak performance 93.6 GFlops, 64 kB level 1 and 8 MB level 2 cache in EPCC Edinburgh. Each processor works with local portion of data associated with its subdomains.

Each domain was first decomposed into identical rectangles Ω^{ij} with the sides H that were discretized by regular grids defined by the stepsize h. Stopping criteria: $\left\|g^P(\lambda,\mu,\rho)\right\| \leq 10^{-4}\,\|d\|$ and parameters $M = 1, \Gamma = 1, \lambda^0 = \frac{1}{2}Bf$ were used for all calculations.

The selected results are summarized in Tables 13.1-13.3 and figures 13.1a-13.1c.

Table 13.1: Parallel scalability for semicoer. problem with prim. dim 540800, dual dim.14975, 2 out. iters, 53 cg iters, 128 subdomains using Lomond, $\rho = 10^3$

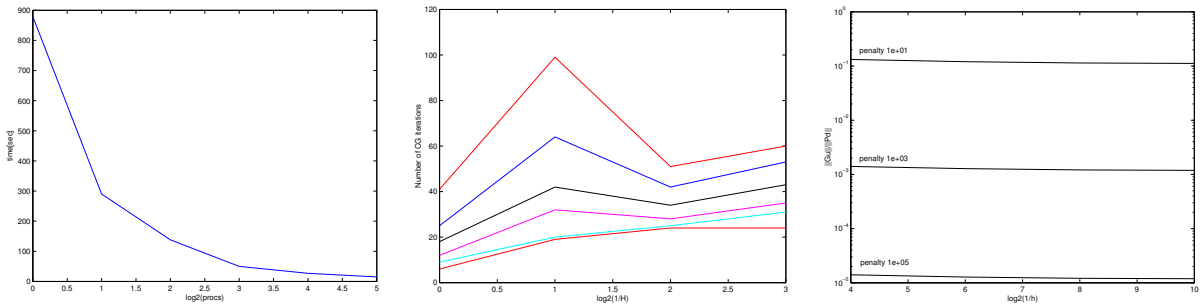
processors	1	2	4	8	16	32
time [sec]	879	290	138	50	27	15

Table 13.2: Numerical scalability QPMPGP - upper row gives prim. dim./dual dim., $\rho = 10^3$, number in lower row gives CG iters. for given H/h

H	1	1/2	1/4	1/8
$H/h = 128$	33282/129 41	133128/1287 99	532512/6687 51	2130048/29823 60
$H/h = 64$	8450/65 25	33800/647 64	135200/3359 42	540800/14975 53
$H/h = 32$	2178/33 18	8712/327 42	34848/1695 34	139392/7551 43
$H/h = 16$	578/17 12	2312/167 32	9248/863 28	36992/3839 35
$H/h = 8$	162/9 9	648/87 20	2592/447 25	10365/1983 31
$H/h = 4$	50/5 6	200/47 19	800/239 24	3200/1055 24

Table 13.3: Optimality of dual penalty

prim.dim./dual.dim.		1152/591	10368/1983	139392/7551	2130048/29823
$\|G\lambda\| \,/\, \|Pd\|$	ρ =1.0e+01	0.489e-01	0.455e-01	0.430e-01	0.422e-01
	ρ =1.0e+03	0.508e-03	0.470e-03	0.445e-03	0.437e-03
	ρ =1.0e+05	0.508e-05	0.470e-05	0.445e-05	0.438e-05



a b c

Figure 13.1: a) Parallel scalability, b) numerical scalability, c) optimality of dual penalty

13.1.2 Semicoercive problem + conforming FE + ALAPC: Numerical scalability and highlights

Stopping criteria $\left\|g^P(\lambda,\mu,\rho)\right\| \leq 10^{-4}\,\|d\|$ and $\|G\lambda\| \leq 10^{-4}\,\|e\|$ were used.

Table 13.4: Large problems using SGI Origin by ALAPC

h	H	prim. dim.	dual. dim.	num. of subdom.	procs	cg. iter.	time [sec]
1/1024	1/8	2130048	29823	128	32	47	167
1/2048	1/8	8454272	59519	128	64	65	1281

Table 13.5: Numerical scalability for ALAPC- upper row gives prim. dim./dual dim., $\rho_0 = 10$, number in lower row gives CG iters.

H	1	1/2	1/4	1/8
$H/h = 128$	33282/129 28	133128/1287 59	532512/6687 36	2130048/29823 47
$H/h = 64$	8450/65 22	33800/647 47	135200/3359 33	540800/14975 43
$H/h = 32$	2178/33 17	8712/327 33	34848/1695 30	139392/7551 37
$H/h = 16$	578/17 13	2312/167 29	9248/863 26	36992/3839 32
$H/h = 8$	162/9 10	648/87 20	2592/447 23	10365/1983 27
$H/h = 4$	50/5 7	200/47 19	800/239 22	3200/1055 25

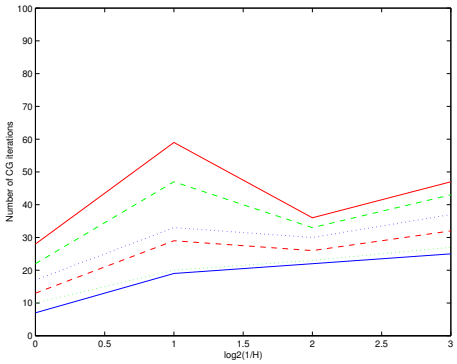


Figure 13.2: Numerical scalability

13.1.3 Semicoercive problem + conforming FE + ALAPC: Comparison of initial approximations $\lambda^{0,\mathbf{I}}$, $\lambda^{0,\mathbf{II}}$, $\lambda^{0,\mathbf{III}}$

Table 13.6: Comparison of $\lambda^{0,\mathrm{I}}$, $\lambda^{0,\mathrm{II}}$, $\lambda^{0,\mathrm{III}}$ and impact of ρ_0 and M; $h = 1/128$, $H = 1/4$, prim. dim. 34848, dual dim. 1695

M	ρ_0	$\lambda^{0,\mathrm{I}}$		$\lambda^{0,\mathrm{II}}$		$\lambda^{0,\mathrm{III}}$	
		out.iter.	cg iter.	out.iter.	cg iter.	out.iter.	cg iter.
10^0	10^0	4	22	5	41	5	35
	10^1	2	24	2	45	2	30
	10^2	1	27	1	56	1	30
10^1	10^1	5	61	3	59	3	30
	10^2	2	27	2	59	2	30
	10^3	1	32	1	69	1	34
10^2	10^2	5	116	3	71	3	34
	10^3	1	32	2	69	2	34
	10^4	1	35	1	155	1	38
10^3	10^3	7	185	3	155	3	114
	10^4	2	35	2	94	2	70
	10^5	1	38	1	247	1	40

13.1.4 Semicoercive problem + conforming FE + ALAPC: The use of lumped preconditioner $F^{-1} = PBKB^TP + Q$

Table 13.7: Comparison of projected, projected-preconditioned version for regular decomposition $H = H_x = H_y$ with parameters $M = 10, \rho_0 = 10^3, \Gamma = 1$

h	H	primar	dual	proj.		proj.-prec.	
		dim.	dim.	out.i.	cg.i.	out.i.	cg.i.
1/64	1	8450	65	2	45	2	36
	1/2	8712	327	2	89	2	74
	1/4	9248	863	2	122	2	75
1/128	1	33282	129	2	64	2	47
	1/2	33800	647	2	131	2	104
	1/4	34848	1695	2	151	2	128

13.2 FETI-DP

13.2.1 Coercive problem + conforming FE + QPMPGP: Changing density of meshes of corner nodes and Lagrange multipliers

An upper row gives primal/dual/corner dimension, $\rho_0 = 10^2$, number in lower row gives CG iterations - it decreases with growing density of corner nodes because of better propagation of error through global corners.

Table 13.8: Numerical scalability: Discretization corresponding to Figure 7a

H	1/2	1/4	1/8
$H/h = 16$	2312/161/2 45	9248/809/18 61	36992/3545/98 72
$H/h = 8$	648/81/2 31	2592/393/18 49	10365/1689/98 55
$H/h = 4$	200/41/2 26	800/185/18 36	3200/761/98 40

Table 13.9: Numerical scalability: Discretization corresponding to Figure 7b

H	1/2	1/4	1/8
$H/h = 16$	2312/155/8 33	9248/791/36 54	36992/3503/140 59
$H/h = 8$	648/75/8 28	2592/375/36 41	10365/1647/140 49
$H/h = 4$	200/35/8 20	800/167/36 31	3200/719/140 27

Table 13.10: Numerical scalability: Discretization corresponding to Figure 7c, coercive problem + conforming FE

H	1/2	1/4	1/8
$H/h = 16$	2312/153/10 27	9248/785/42 48	36992/3489/154 51
$H/h = 8$	648/73/10 22	2592/369/42 36	10365/1633/154 38
$H/h = 4$	200/33/10 17	800/161/42 21	3200/705/154 27

13.2.2 Coercive, semicoercive problem + conforming, mortar FE + QPMPGP: Numerical scalability

This section reports some results for the numerical solution of the model coercive and semicoercive contact problem to illustrate the performance of FETI–DP algorithm, mainly its numerical scalability. The computations were performed using parameter values $\rho_0 = 100, \overline{\alpha} = \rho^{-1}$, $\Gamma = 1$, and $\lambda^0 = 0$. The stopping criterion in the CG iteration was $||g^P(\lambda^k)|| < 10^{-5}||d||$. All experiments were performed in MATLAB.

Selected results of the computations for varying values of H and H/h are given in Tables 13.10-13.13. The primal dimension/dual dimension/number of corners are recorded in the upper row in each field of the tables, while the number of the CG iterations required for the convergence of the solution to the given precision is recorded in the lower row. The dependence of the number of iterations on the number of subdomains is shown in Figures 13.3, 13.4, where each line corresponds to a fixed value for H/h, i.e. to one row of table. The key point is that the number of the CG iterations for a fixed ratio H/h varies very moderately with the increasing number of subdomains.

Table 13.11: Numerical scalability: semicoercive problem + conforming FE

H	1/2	1/4	1/8
$H/h = 16$	2312/153/10 41	9248/785/42 57	36992/3489/154 63
$H/h = 8$	648/73/10 27	2592/369/42 39	10365/1633/154 46
$H/h = 4$	200/33/10 24	800/161/42 24	3200/705/154 31

Table 13.12: Numerical scalability: coercive problem + mortar FE

N_1 N_2	1x2 1x3	2x4 2x5	4x8 4x11
$H_1/h_1 = 16$	2606/95/6	9072/524/25	38992/2654/97
$H_2/h_2 = 25$	33	57	78
$H_1/h_1 = 8$	750/47/6	2608/256/25	11216/1298/97
$H_2/h_2 = 13$	29	49	59
$H_1/h_1 = 4$	242/23/6	840/122/25	3616/620/97
$H_2/h_2 = 7$	15	34	49

Table 13.13: Numerical scalability: semicoercive problem + mortar FE, nonmortars on Γ_c^1

N_1 N_2	1x2 1x3	2x4 2x5	4x8 4x11
$H_1/h_1 = 16$	2606/95/6	9072/524/25	38992/2654/97
$H_2/h_2 = 25$	39	65	84
$H_1/h_1 = 8$	750/47/6	2608/256/25	11216/1298/97
$H_2/h_2 = 13$	36	56	70
$H_1/h_1 = 4$	242/23/6	840/122/25	3616/620/97
$H_2/h_2 = 7$	20	37	52

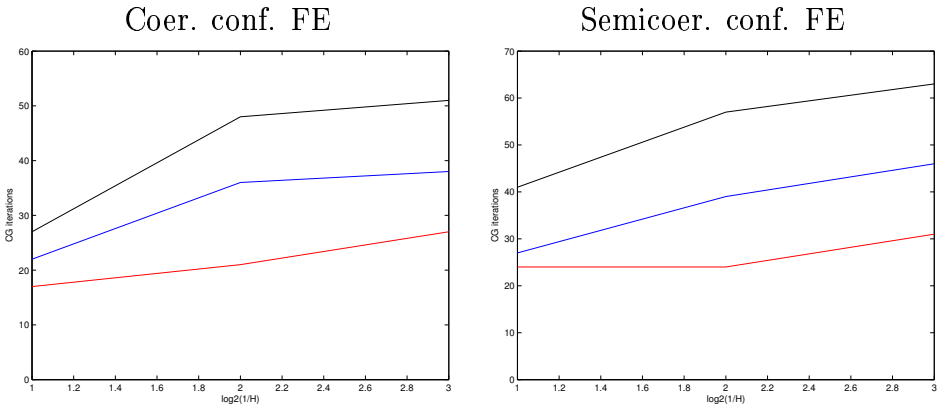


Figure 13.3: Numerical scalability for FETI-DP - conforming FE

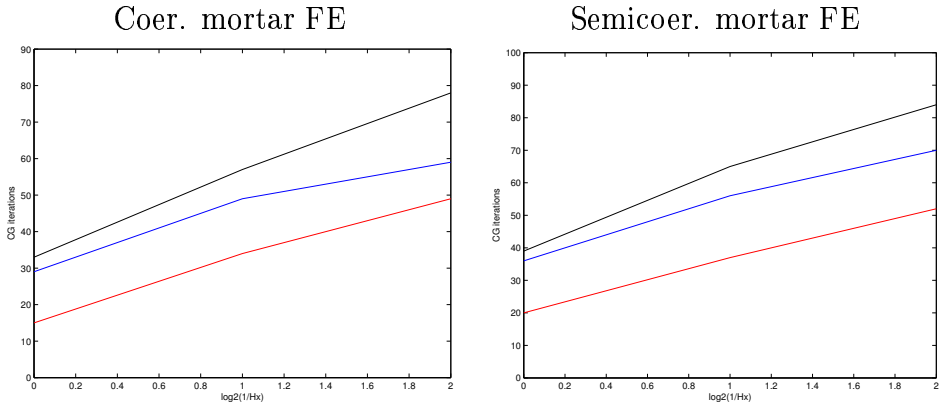


Figure 13.4: Numerical scalability for FETI-DP - mortar FE

13.2.3 Coercive problem + mortar FE + QPMPGP: Nonnormalized vs. normalized B_I

This section presents that the performance of our FETI–DP method deteriorates unless the rows have norms of similar order, see Theorem 9.3 for an explanation of this phenomenon.

Since the mortar conditions are only related to interior nodes on the nonmortar edges, two extra conditions are necessary to enforce nonpenetration at the first and last nodes on Γ_c. All these inequality constraints can be written in matrix formulation as

$$B_I u \ \leq \ 0,$$

where B_I has one normalized horizontal block B_{γ_m} for each nonmortar side γ_m, and two more rows for the nonpenetration conditions at the endpoints of Γ_c. Also, let $B_I^{\#}$ be the matrix having one nonnormalized block $B_{\gamma_m}^{\#}$ for each nonmortar γ_m, together with the two extra rows as before. We showed theoretically in Chapter 9 and experimentally in in this section that the FETI–DP algorithm using the matrix $B_I^{\#}$ instead of B_I results in a nonscalable algorithm, what is presented in table, where the convergence results for the case of matching subdomains across Γ_c for the case when B_I is not normalized, for the case when the nonmortar sides were chosen to be on Γ_c^1, and the convergence results for the algorithm with normalized B_I. These results are consistent with the condition number estimate from Theorems 9.1 and 9.3.

An explanation of this rather surprising result is that there is no direct scaling of multiplications by the matrix B in our algorithm, as it was the case for other FETI algorithms, where a factor of the form $(BB^T)^{-1}$ appeared in the definition of the preconditioners.

An upper row of following table gives primal/dual/corner dimension, number in lower row gives CG iterations for normalized vs. nonnormalized B_I

Table 13.14: Numerical scalability for coercive mortar FE - normalized vs. nonnormalized B_I

N_1 N_2	1x1 1x1	2x2 2x2	4x4 4x4
$H_1/h_1 = 16$	965/17/0	3860/188/10	15440/998/42
$H_2/h_2 = 25$	16 x 25	31 x 268	60 x 743
$H_1/h_1 = 8$	277/9/0	1108/92/10	4432/486/42
$H_2/h_2 = 13$	11 x14	26 x 118	46 x 263
$H_1/h_1 = 4$	89/5/0	356/44/10	1424/230/42
$H_2/h_2 = 7$	6 x 8	20 x 48	28 x 106

13.2.4 Coercive, semicoercive problem + mortar FE + QPMPGP: Nonmortars on Γ_c^1 vs. Γ_c^2

In this section, we report results for the numerical solution of the model coercive and semicoercive contact problem to illustrate the performance of our FETI–DP algorithm. The goals of our experiments were as follows:

- to establish numerical evidence for the scalability of the algorithm;
- to compare the performance of the method for the cases when the subdomain partitions of Ω^1 and Ω^2 match, or do not match, across Γ_c, the potential contact interface;
- to investigate the dependence of the convergence on the choice of nonmortar sides either on Γ_c^1 or on Γ_c^2.

For the case when the subdomain partitions across Γ_c match, we partitioned both Ω^1 and Ω^2 into 1×1, 2×2, and 4×4 squares, respectively, corresponding to $H_1 = H_2 \in \{1, 1/2, 1/4\}$. To avoid perfectly matching meshes, the number of nodes on each side of the square subdomains was chosen to be $H_1/h_1 \in \{4, 8, 16\}$, in Ω^1, corresponding to $H_2/h_2 \in \{7, 13, 25\}$, in Ω^2. In Table 13.15, we report the iteration counts for coercive and semicoercive problem, i.e. the number of the CG iterations required for the convergence of the solution of the problem to the given precision, as well as the size of the primal problem, of the dual problem, and the number of global corner degrees of freedom, i.e. the size of the coarse problem corresponding to solving a linear system for K_{cc}^*, for each partition described above.

The algorithm converged after a reasonably small number of iterations, for all partitions considered. For fixed number of nodes per subdomain edge, i.e. for H_1/h_1 and H_2/h_2 simultaneously fixed, the number of iterations increased moderately when the number of subdomains quadrupled. Thus, numerical scalability of our method was observed for practical applications, and we may infer that the unspecified constants in theorems were not very large.

Table 13.15: Convergence results: matching subdomain partitions across Γ_c, coercive/semicoercive problem

				Nonmortars on Γ_c^1				Nonmortars on Γ_c^2			
N_1	N_2	$\frac{H_1}{h_1}$	$\frac{H_2}{h_2}$	Iter	prim.	dual	cor.	Iter	prim.	dual	cor.
1×1	1×1	4	7	6/-	89	5	0	13/-	89	8	0
		8	13	11/-	277	9	0	19/-	277	14	0
		16	25	16/-	965	17	0	25/-	965	26	0
2×2	2×2	4	7	20/24	356	44	9	30/41	356	50	9
		8	13	26/32	1108	92	9	36/54	1108	102	9
		16	25	31/44	3860	188	9	51/54	3860	206	9
4×4	4×4	4	7	28/39	1424	230	39	42/45	1424	242	39
		8	13	46/43	4432	486	39	63/62	4432	506	39
		16	25	60/59	15440	998	39	74/85	15440	1034	39

The scalability of the method was observed for both when the nonmortars were chosen on Γ_c^1, and on Γ_c^2. The difference between the two methods is given by the number of mortar conditions, and therefore of Lagrange multipliers λ_I and the size of the dual problem. All these numbers are larger when the nonmortars are chosen on the edges with finer local mesh, i.e. on Γ_c^2. The number of iterations in this case was larger by about 50% than in the case when the nonmortars were chosen on the coarser local mesh, i.e. on Γ_c^1. This was due in part to the fact that the nonpenetration conditions had more of a local nature in the case of a finer local mesh, and therefore little was gained by having more such conditions. This holds true with, possibly, the exception of a too coarse mesh on the nonmortar sides, i.e. H_1=1 with $H_1/h_1 = 2$ or $H_1/h_1 = 3$. I this case, penetration may even occur due to the lack of nonpenetration conditions. We choose discontinuous test functions corresponding to the biorthogonal mortars.

For the case when the partitions across Γ_c do not match, we partitioned Ω^1 into 1×2, 2×4, and 4×8 rectangles, corresponding to partitions of Ω^2 into 1×3, 2×5, and 4×11 rectangles, respectively. The number of nodes on each side of the square subdomains was chosen to be, alternatively, in the set $\{4, 8, 16\} \times \{7, 13, 25\}$; see Table 13.16 for more details for coercive and semicoercive problem. As before, the iteration count, and the sizes of the primal and dual problems, and of the coarse problem are reported.

For fixed number of nodes per subdomain edge, i.e. for H_1/h_1 and H_2/h_2 simultaneously fixed, the number of iterations increased moderately when the number of subdomains roughly quadrupled. Thus, numerical scalability of our method was once again observed, independent of whether the nonmortar sides were chosen on Γ_c^1, and on Γ_c^2. The number of iterations was once again larger when more nonpenetration conditions were required, i.e. when the number of nodes on the nonmortars was larger. This was due to the fact that, for a mesh that is fine enough, some of the nonpenetration conditions become less relevant. In the experiments presented above, the rows of the matrix B_I were normalized as discussed in Chapter 9.

Table 13.16: Convergence results: nonmatching subdomain partitions across Γ_c, coercive/semicoercive problem

				Nonmortars on Γ_c^1				Nonmortars on Γ_c^2			
N_1	N_2	$\frac{H_1}{h_1}$	$\frac{H_2}{h_2}$	Iter	prim.	dual	cor.	Iter	prim.	dual	cor.
1×2	1×3	4	7	15/20	242	23	5	41/45	242	35	5
		7	4	28/29	203	26	5	23/28	203	23	5
		8	13	29/36	750	47	5	48/54	750	69	5
		13	8	37/44	635	52	5	36/38	635	49	5
		16	25	33/39	2606	95	5	60/82	2606	137	5
		25	16	41/47	2219	104	5	41/48	2219	101	5
2×4	2×5	4	7	34/37	840	122	22	52/60	840	140	22
		7	4	43/56	762	125	22	52/44	762	116	22
		8	13	49/56	2608	256	22	68/76	2608	288	22
		13	8	58/68	2378	261	22	51/56	2378	248	22
		16	25	57/65	9072	524	22	57/100	9072	584	22
		25	16	67/91	8298	533	22	63/68	8298	512	22
4×8	4×11	4	7	49/52	3616	620	90	65/80	3616	662	90
		7	4	56/60	3148	581	90	50/53	3148	566	90
		8	13	59/70	11216	1298	90	88/112	11216	1374	90
		13	8	71/86	9836	1233	90	65/71	9836	1214	90
		16	25	78/-	38992	2654	90	125/-	38992	2798	90
		25	16	94/-	34348	2537	90	90/-	34348	2510	90

13.2.5 Coercive problem + conforming FE + QPMPGP: Comparison of FETI-1, FETI-2 and FETI-DP

Table 13.17: Number of CG iters. of QPMPGP for FETI-1, FETI-2, FETI-DP for fixed $1/H = 6$ and decreasing h

$1/H$	6	6	6	6
$1/h$	4	8	12	16
P/D/C	1800/575/0	5832/1079/0	12168/1583/0	20808/2087/0
FETI-1: CG	38	50	63	67
P/D/C	1800/575/190	5832/1079/190	12168/1583/190	20808/2087/190
FETI-2: CG	24	45	60	67
P/D/C	1800/385/90	5832/889/90	12168/1393/90	20808/1897/90
FETI-DP: CG	23	37	43	51

13.3 TFETI-1

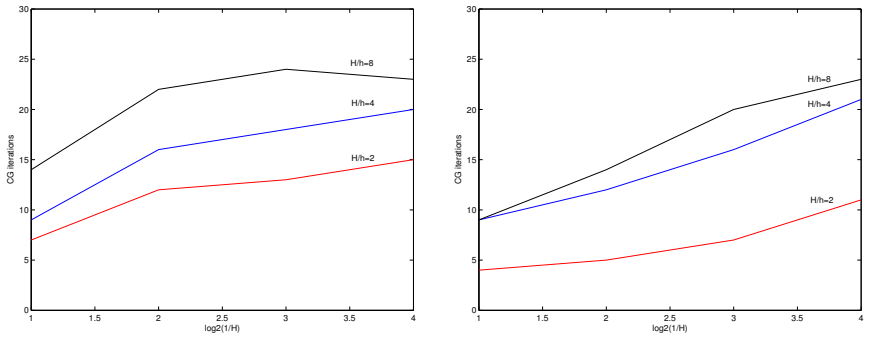
13.3.1 Model linear problem in 2D

I have implemented FETI-1 and TFETI-1 in Matlab 7 and applied both methods to the solution of simple 2D model problem described by square $\Omega = (0,1) \times (0,1)$ and its solution is illustrated in Figure 10.1 a), b), its decomposition and discretization is drawn near in Figure 10.2 and Figure 10.3 demonstrates two approaches of prescribtion of Dirichlet boundary conditions - a) the classical one (original FETI-1), b) the new one by Lagrange multipliers (TFETI-1).

To this end, we have implemented algorithm to solve the basic dual problem (10.5) so that we could plug in the projectors to the natural coarse space.

Table 13.18: Numerical scalability for the linear 2D model problem, QPMPGP

H/h	H	h	prim.	dual FETI-1	dual TFETI-1	CG FETI-1	CG TFETI-1
2	1/2	1/4	36	11	17	7	4
2	1/4	1/8	144	63	75	12	5
2	1/8	1/16	576	287	311	13	7
2	1/16	1/32	2304	1215	1263	15	11
4	1/2	1/8	100	19	29	9	9
4	1/4	1/16	400	111	131	16	12
4	1/8	1/32	1600	511	551	18	16
4	1/16	1/64	6400	2175	2255	20	21
8	1/2	1/16	324	35	53	14	9
8	1/4	1/32	1296	207	243	22	14
8	1/8	1/64	5184	959	1031	24	20
8	1/16	1/128	20736	4095	4239	23	23



a b

Figure 13.5: Numerical scalabilities for FETI-1 and TFETI-1

13.3.2 Model contact problem in 2D and 3D: Comparison of FETI-DPC solved by Newton-like method and TFETI-1 by SMALBE

As mentioned, I participated in implementation of TFETI-1 and FETI-DP for 2D and 3D contact problems as in Figure 10.4 and 10.5. We implemented two FETI based algorithms for the solution of contact problems, using the research software which is being developed in Stanford.

The first one, FETI-DPC, is based on FETI-DP domain decomposition method. The algorithm uses the Newton-like method which solves the equilibrium equation in Lagrange multipliers in the inner loop, while feasibility of each step is ensured in the outer loop by the primal and dual planning steps. The algorithm exploits standard FETI preconditoners, namely the Schur and lumped ones. The additional speedup of convergence is achieved by application Krylov type acceleration scheme. The algorithm exploits a globalization strategy in order to achieve monotonic global convergence.

The second algorithm is based on the TFETI-1 domain decomposition, a variant of the FETI-1, which treats all the boundary conditions by Lagrange multipliers, so that all the subdomains are floating, and their kernels are known a priori and can be used in construction of the natural coarse grid. It exploits recently proposed SMALBE algorithm.

The 2D problem involves 6 rectangles in mutual contact as it is depicted on the Figure 13.6a. The left rectangles are fixed on the left side (blue arrows) while the right ones are free and they are loaded (red arrows) such a way that the problem has unique solution. Each rectangle were further decomposed to the 4 subrectangles and therefore the original problem were decomposed to 24 subdomains, see Figure 13.6b). The performance of the algorithms FETI-DPC by Newton and TFETI-1 by SMALBE is compared in Table 13.19. Outer iterations are used only in the case of SMALBE method while the number of subiterations is used only in methods FETI-DP. The number of dual plannings and primal plannings of FETI-DPC methods corresponds to the number of expansion and proportioning steps in the case of SMALBE method. Therefore they share the same column for each methods. The numbers on the left side of the slashes represent number of iteration for the problem with 6 subdomains and the numbers on the right sides represent the number of iterations for 24 subdomains problem. The resulting deformation with distribution of the stresses are depicted in the Figure 10.4.

Table 13.19: Algorithms performance for 2D semicoercive problem with 6 and 24 subdomains.

	Outer iter.	Main iter.	subiter.	Primal plan. (Exp. step)	Dual plan. (Proport.)
FETI-DPC	-	17/32	0/0	2/2	0/0
TFETI-1	1/21	9/68	-	0/18	1/3

a) 2D problem with 6 subdomains b) 2D problem with 24 subdomains

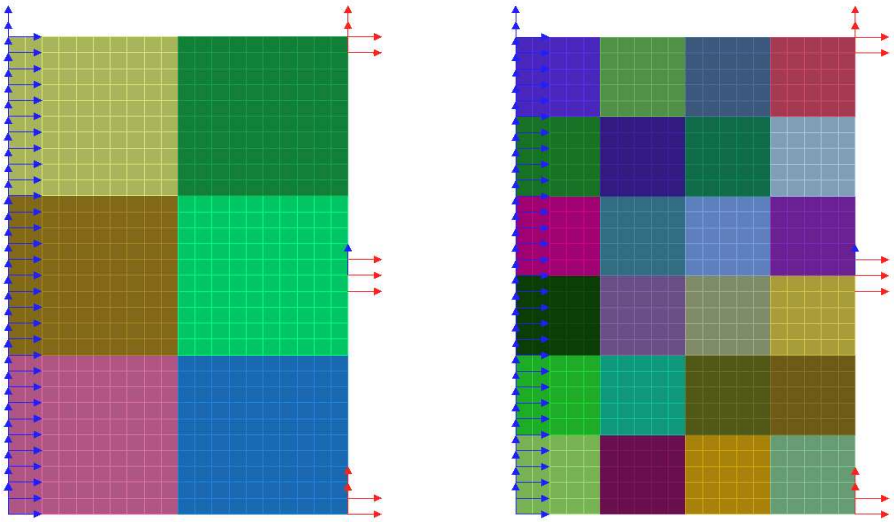


Figure 13.6: 2D model contact problem - conforming decompositions and nodal discretizations

a) 3D problem with matching grids b) 3D problem with nonmatching grids

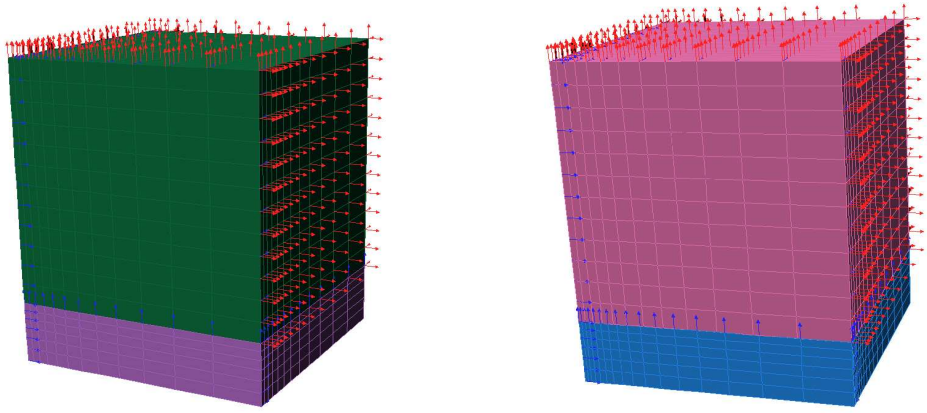


Figure 13.7: 3D model contact problem - conforming decompositions and nodal discretizations

The second 3D model problem consists of 2 bricks in mutual contact. The bottom brick is fixed in all degrees of freedom while the upper one is fixed only in such a way, that only vertical rigid body movement is allowed. The situation is depicted in the Figure 13.7. We have analyzed 2 cases. The first one, with matching grid on the contact interface prescribes node-to-node contact conditions. The second one allows nonmatching grids and the mortar elements were used for assembling of contact conditions. The resulting performance of algorithms is collected in the Table 13.20. Columns in this table have the same meaning as in 2D case.

Table 13.20: Algorithms performance for 3D problem with matching and nonmatching grid on contact interface.

	Outer iter.	Main iter.	subiter.	Primal plan. (Exp. step)	Dual plan. (Proport.)
FETI-DPC	-	24/26	11/10	7/8	0/0
TFETI-1	13/10	29/29	-	20/20	0/0

13.4 MFETI-DP

13.4.1 Model linear problem

Here I present preliminary numerical experiments for coercive linear model problem comparing finite and iterative solution of coarse problems of the third type including inexact solution with CG in simple V-cycle - following tables give overview of CG iterations for 2-level method generated by parameters H_1, h_1 on the first level and H_2, h_2 on the second level. Other rows give primal/dual/corner (P/D/C) dimensions, number of CG iters. for clasical FETI-DP (finite) and then numbers of CG iters on first/second level solved iteratively to given precision or by given number of iterations on second level, when we perform always one CG iteration on the first level by QPMPGP. This coarse problem of the third type plays also the role in monitoring the primal error. Following tables with preliminary results illustrate behaviour of MFETI-DP. Table 13.21 shows the dependence of number of CG iterations for QPMPGP to decreasing discretization parameter h_1, i.e. increasing $1/h_1$, keeping decomposition parameter H_1 fixed. Table 13.22 depicts numbers of CG iterations for QPMPGP for problems with nearly same primal dimension, for all decompositions: $\frac{1}{H_1} \cdot \frac{1}{h_1} \simeq 100$, these numbers varies moderately. Tables 13.23 and 13.24 present number of CG for fixed $1/h_1 = 4$ or $1/h_1 = 8$ and decreasing H_1, i.e. increasing $1/H_1$, in both cases is the ratio $\frac{H_1}{h_1}$ close to 1, so that the numerical scalability of MFETI-DP is demonstrated. The results are drawn in Figures 13.8 and 13.9 - each line corresponds to one row of CG iterations, excluding rows when the problem was solved to the precision 10^{-10}.

Table 13.21: Increasing number of CG for fixed $1/H_1 = 6$ and decreasing h_1

$1/H_1$	6	6	6	6
$1/h_1$	4	8	12	16
$1/H_2$	4	4	4	4
$1/h_2$	4	4	4	4
P/D/C	1800/430/50	5832/934/50	12168/1438/50	20808/1942/50
clasical: 1e-6	44	70	89	107
1:1e-6/2:1e-6	51/769	79/1187	100/1512	123/1867
1:1e-6/2:5cg	74/370	121/605	174/870	210/1050
1:1e-10/2:5cg	124/620	199/995	282/1410	343/1715

Table 13.22: Number of CG for $\frac{1}{H_1} \cdot \frac{1}{h_1} \simeq 100$

$1/H_1$	4	5	6	7	8
$1/h_1$	25	20	17	13	12
$1/H_2$	2	3	4	5	6
$1/h_2$	2	3	4	5	6
P/D/C	21632/1280/18	22050/1657/32	23328/2068/50	19208/2162/72	21632/2624/98
clasical: 1e-6	115	117	114	98	100
1:1e-6/2:1e-6	134/1577	134/1847	128/1925	110/1707	109/1730
1:1e-6/2:5cg	208/1040	238/1190	212/1060	194/970	155/775
1:1e-10/2:5cg	333/1665	403/2015	343/1715	326/1630	252/1260

Table 13.23: Number of CG for fixed $1/h_1 = 4$ and decreasing H_1, $\frac{H_1}{h_1} \in \left[\frac{4}{7}, 1\right]$

$1/H_1$	4	5	6	7
$1/h_1$	4	4	4	4
$1/H_2$	2	3	4	5
$1/h_2$	2	3	4	5
P/D/C	800/188/18	1250/297/32	1800/430/50	2450/587/72
clasical: 1e-6	38	43	44	47
1:1e-6/2:1e-6	43/511	49/661	51/769	53/824
1:1e-6/2:5cg	60/300	76/380	74/370	79/395
1:1e-10/2:5cg	97/485	128/640	124/620	132/660

Table 13.24: Number of CG for fixed $1/h_1 = 8$ and decreasing H_1, $\frac{H_1}{h_1} \in \left[\frac{8}{11}, 1\right]$

$1/H_1$	8	9	10	11
$1/h_1$	8	8	8	8
$1/H_2$	6	7	8	9
$1/h_2$	6	7	8	9
P/D/C	10368/1696/98	13122/2161/128	16200/2682/162	19602/3259/200
clasical: 1e-6	76	81	82	83
1:1e-6/2:1e-6	85/1357	87/1404	89/1457	91/1484
1:1e-6/2:5cg	139/695	125/625	137/685	142/710
1:1e-10/2:5cg	223/1115	213/1065	225/1125	230/1150

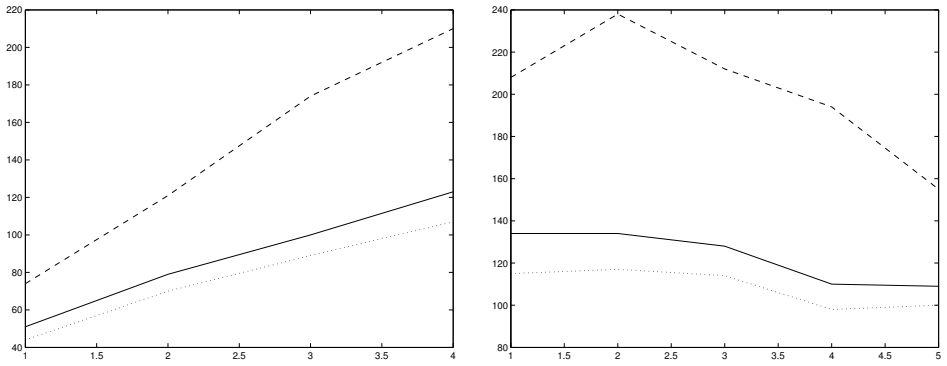


Figure 13.8: Numerical scalability of MFETI-DP corresponding to Tables 13.21, 13.22

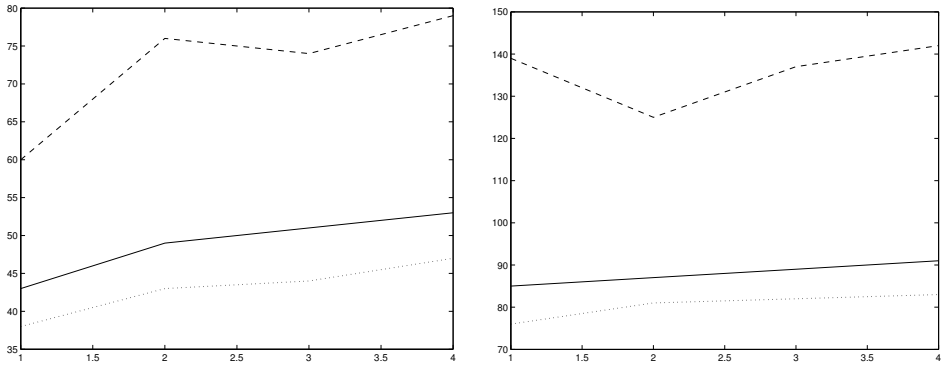


Figure 13.9: Numerical scalability of MFETI-DP corresponding to Tables 13.23, 13.24

13.4.2 Model contact problem

Numerical experiments for coercive contact model problem are presented in this section. I used FETI-DP recursively for coarse problem arising at recontruction of u_c. Table 13.25 gives overview of CG iterations for 2-level method generated by parameters H_1, h_1 on the first level and H_2, h_2 on the second level. Other rows give primal/dual/corner (P/D/C) dimensions, number of CG iterations on the 1st/number of CG iterations on the 2nd levels, both solved once to the relative precision 10^{-6}, for QPMPGP with fixed ratio $\frac{H_1}{h_1} = \frac{1}{2}$, these numbers varies moderately, this indicates numerical scalability.

Table 13.25: Number of CG for fixed $\frac{H_1}{h_1} = \frac{1}{2}$

$1/H_1$	4	8	12
$1/h_1$	2	4	6
$1/H_2$	2	6	10
$1/h_2$	2	6	10
1:P/D/C	288/84/18	3200/768/98	14112/2801/242
2:P/D/C	32/8/2	288/40/50	800/63/162
1:1e-6/2:1e-6	25/4	36/13	56/14

Chapter 14

Conclusion

This dissertation develops new author's results and remarks, and the results on which the author participated as a member of teams in the branch of the FETI-based domain decomposition methods for the solution of coercive and semicoercive variational inequalities. These results including the parallel implementation and many numerical experiments (concerning FETI-1, FETI-2, FETI-DP, TFETI-1, MFETI-DP, optimality of dual penalty, etc.) were published in 13 impact papers and in 33 conference papers and they contributed to developing many theoretical results.

Special consideration belongs to the most important theoretical results:

- modification of FETI-DP for corners on the contact zone for both coercive and semicoercive cases, I have derived the additional condition that preserves the non-penetration in Lagrange multipliers and enables the usage of corners on contact interface,
- normalization of rows of mortar constraint matrix leads to significantly smaller upper bound on the condition number of the dual operator matrix, this my observation is the key ingredient for proof of numerical scalability of FETI-DP method for contact problems with mortars,
- Multilevel FETI-DP method for linear and contact extreme large scale problems - the basic idea is a natural application of multigrid methods, i.e. recursive application of FETI-DP to the coarse problem, I carried out the complicated implementation of this method during my stay in Boulder, including the numerical experiments with comparison of finite and iterative solution of coarse problems and inexact solution of coarse problems on the second level with CG method for both, linear and contact problems.

Presented results of solutions of model variational inequalities indicate both - high numerical and parallel scalability of these algorithms. Numerical experiments with the model variational inequality discretized by up to more than eight million of nodal variables indicate that the algorithm may be very efficient and they are in agreement

with the theory of its optimality. All the reasoning may be exploited also to solution of contact problems of elasticity with the Coulomb friction, but this is a theme of next work.

I have just received an email - the parallel computer HPCx in EPCC I have acces to has been upgrated to masively parallel computer having 2560 processors and the peak performance 12 TFlops. I will start running new larger experiments. 80 years have passed since 1927, the year book I started this work was printed. The progress and improvement made in numerical mathematics and computer science is amazing. What development in these branches can we look forward to in future 80 years? We will see...

Bibliography

[1] V. Láska, V. Hruška, Theory and Practice of Numerical Reckonning, in Czech, Prague, 1927.

[2] Z. Dostál, F. A. M. Gomes Neto, S. A. Santos, Duality based domain decomposition with natural coarse space for variational inequalities, Journal of Computational and Applied Mathematics, 126, 1-2, 2000, 397-415.

[3] G. Duvant, J. L. Lions, Inequalities in Mechanics and Physics, Springer Verlag, Berlin, 1976.

[4] I. Hlaváček, J. Haslinger, J. Nečas, J. Lovíšek, Solution of Variational Inequalities in Mechanics, Springer Verlag, Berlin, 1988.

[5] C. Farhat, P. Chen, F. Risler, F. X. Roux, Simple and Unified Framework for Accelarating the Convergence of Iterative Substructuring Methods with Lagrange Multipliers, SIAM J. Sci. Stat. Comput.

[6] C. Farhat, J. Mandel, F. X. Roux, Optimal convergence properties of the FETI domain decomposition method, Comput. Methods Appl. Mech. Eng. 115, 1994, 365-385.

[7] O. Axelson, A class of iterative methods for finite element equations, Comp. Meth. in. Appl. Mech. and Eng., 9, 1976, 127-137.

[8] A. R. Conn, N. I. M. Gould, Ph. L. Toint, A globally convergent augmented Lagrangian algorithm for optimization with general constraints and simple bounds, SIAM J. Num. Anal. 28, 1991, 545-572.

[9] Z. Dostál, Box constrained quadratic programming with proportioning and projections, SIAM J. Opt.. 7, 1997, 871-887.

[10] S. Balay, W. Gropp, L. C. McInnes, B. Smith, PETSc 2.0 Users Manual, http://www.mcs.anl.gov/petsc/, Argonne National Laboratory.

[11] F. J. Lingen, Efficient Gram-Schmidt orthonormalization on parallel computers, Research report, Department of Aerospace Engineering, Delft University of Technology, 1999.

[12] Y. Fragakis, M. Papadrakakis, A unified framework for formulating Domain Decomposition Methods in Structural Mechanics, technical report at National Technical University Athens, 2002.

[13] Z. Dostál, D. Horák, Scalability and FETI based algorithm for large discretized variational inequalities, Mathematics and Computers in Simulation 61, (3-6), 2003, 347-357.

[14] Z. Dostál, D. Horák and D. Stefanica, A scalable FETI-DP algorithm for a coercive variational inequalities, IMACS J. Appl. Numer. Math., 2005.

[15] Z. Dostál, D. Horák and D. Stefanica, A scalable FETI-DP algorithm with non-penetration mortar conditions on contact interface, 2004.

[16] Z. Dostál, D. Horák, Scalable FETI with Optimal Dual Penalty for a Variational Inequality, Numerical Linear Algebra and Applications 11, 5-6, 2004, 455 - 472.

[17] D. Dureisseix, C. Farhat, A numerically scalable domain decomposition method for solution of frictionless contact problems, Int. J. Numer. Meth. Engng. 50, No.12, 2643-2666, 2001.

[18] C. Farhat, J. Mandel, F.-X. Roux, Optimal convergence properties of the FETI domain decomposition method, Comput. Methods Appl. Mech. Engrg. 115, 1994, 365-385.

[19] C. Farhat, F. X. Roux, An unconventional domain decomposition method for an efficient parallel solution of large-scale finite element systems, SIAM J. Sci. Statist. Comput. 13, 1992, 379-396.

[20] J. Mandel, R. Tezaur, Convergence of substructuring method with Lagrange multipliers, Numerische Mathematik, 73, 1996, 473-487.

[21] Z. Dostál, D. Horák and R. Kučera, Total FETI - an easier implementable variant of the FETI method for numerical solution of elliptic PDE, Commun. in Numerical Methods in Engineering 22, 2006, 1155-1162.

[22] D. P. Bertsekas, Nonlinear Optimization, Athena Scientific, Belmont, 1999.

[23] Z. Dostál and D. Horák, Scalable FETI for numerical solution of variational inequalities, SINUM.

[24] Z. Dostál, D. Horák and D. Stefanica, An Overview of Scalable FETI-DP Algorithms for Variational Inequalities, Lecture Notes in Comput. Science and Engineering 55, Proceedings from the 16th conference on DDM, New York, Springer, 2006, 223-230.

[25] Z. Dostál, D. Horák and D. Stefanica, A Scalable FETI–DP Algorithm for Semicoercive Variational Inequality, Computer Methods in Applied Mechanics and Engineering, Vol. 196, 8, ISSN 0045-7825, 2007, 1369-1379.

[26] C. Farhat, M. Gérardin, On the general solution by a direct method of a large scale singular system of linear equations: application to the analysis of floating structures, International Journal for Numerical Methods in Engineering, 41, 4, 1998 675–696.

[27] C. Farhat, M. Lesoinne, P. LeTallec, K. Pierson and D. Rixen, FETI-DP: A dual–prime unified FETI method. I: A faster alternative to the two–level FETI method, Int. J. Numer. Methods Eng. 50, No.7, 2001, 1523–1544.

[28] C. Farhat, J. Mandel, F.X. Roux, Optimal convergence properties of the FETI domain decomposition method, Computer Methods in Applied Mechanics and Engineering, 115, 1994, 365-385.

[29] C. Farhat, F.X. Roux, An unconventional domain decomposition method for an efficient parallel solution of large-scale finite element systems, SIAM Journal on Scientific Computing 13, 1992, 379-396.

[30] A. Klawonn and O. B. Widlund, FETI and Neumann-Neumann iterative substructuring mathods: connections and New results, Communications on Pure and Applied Mathematics, Vol. LIV, 2001, 57-90.

[31] A. Toselli, O. B. Widlund, Domain Decomposition Methods-Algorithms and Theory, Springer Series on Computational Mathematics 34, Springer-Verlag, Berlin 2005.

[32] C. Bernardi, Y. Maday and A. T. Patera, A new nonconforming approach to domain decomposition: The mortar element method, in College de France Seminar, H. Brezis and J.-L. Lions, eds., Longman Scientific and Technical, UK, 1994, 13-51.

[33] F. Ben Belgacem and Y. Maday, The mortar element method for three dimensional finite elements, RAIRO Model. Math. Anal. Numer., 31, 1997, 289-302.

[34] B. Wohlmuth, A Mortar Finite Element Method Using Dual Spaces for the Lagrange Multiplier, SIAM J. Numer. Anal., Vol 38, No. 3, 2000, 989-1012.

[35] B. Wohlmuth, Discretization Methods and Iterative Solvers Based on Domain Decomposition, Habilitationsschrift Mathematisch-Naturwissenschaftliche Fakultät, Universität Augsburg, Germany, 1999.

[36] R. H. Krause, B. Wohlmuth, A Dirichlet-Neumann Type Algorithm for Contact Problems with Friction.

[37] S. Bertoluzza, V. Perrier, The Mortar Wavelet Method, 1999.

[38] D. Stefanica, A Numerical Study of FETI Algorithms for Mortar Finite Element Methods, SIAM J. Sci. Comput., Vol. 23., No. 4, 2001, 1135-1160.

[39] X.-C. Cai, M. Dryja, and M. Sarkis, Overlapping nonmatching grid mortar element methods for elliptic problems, SIAM J. Numer. Anal., 36, 1999, 581-606.

[40] Y. Achdou, Y. Maday, and O. B. Widlund, Iterative substructuring preconditioners for mortar element methods in two dimensions, SIAM J. Numer. Anal., 36, 1999, 551-580.

[41] M. Dryja, An iterative substructuring method for elliptic mortar finite element problems with a new coarse space, East-West J. Numer. Math., 5, 1997, 79-98.

[42] P. Le Tallec, T. Sassi, and M. Vidrascu, Three-dimensional domain decomposition methods with nonmatching grids and unstructured grid solvers, in Proceedings on the 7th International Conference on DDM in Scientific and Engineering Computing, D. E. Keyes and J. Xu, eds., Contemp. Math. 180, AMS, Providence, RI, 1994, 61-74.

[43] D. Stefanica and A. Klawonn, The FETI method for mortar finite elements, in Proceedings of the 11th International Conference on DDM, C.-H. Lai, P. Bjorstad, M. Cross, O. B. Widlund, eds., Greenwich, UK, 1998, 121-129.

[44] J. Schöberl, Solving the Signorini problem on the basis of domain decomposition techniques, Computing, 1998, 60, 323-344.

[45] C. Kim, R. D. Lazarov, J. E. Pasciak, P. S. Vassilevski, Multiplier spaces for the mortar finite element method in three dimensions, 2000.

[46] P. Avery, G. Rebel, M. Lesoinne, C. Farhat, A numerically scalable dual-primal substructuring method for the solution of contact problems - part I: the frictionless case, Comput. Methods Appl. Mech. Engrg., 193, 2004, 2403-2426.

[47] Z. Dostál, J. Schöberl, Minimizing quadratic functions over non-negative cone with the rate of convergence and finite termination, Computational Optimization and Applications 30, 1, 2005, 23-44.

[48] Z. Dostál, F. A. M. Gomes Neto, S. A. Santos, Solution of contact problems by FETI domain decomposition with natural coarse space projection, Computer Methods in Applied Mechanics and Engineering 190, 13-14, 2000, 1611-1627.

[49] O. Axelsson, Iterative Solution Methods, Cambridge University Press, Cambridge 1994.

[50] Z. Dostál, J. Haslinger, R. Kučera, Implementation of fixed point method for duality based solution of contact problems with friction, Journal of Computational and Applied Mathematics, 140, 2002, 245-256.

[51] Z. Dostál, A proportioning based algorithm for bound constrained quadratic programming with the rate of convergence, Numerical Algorithms 34, 2-4, 2003, 293-302.

[52] Z. Dostál, Inexact semi-monotonic Augmented Lagrangians with optimal feasibility convergence for quadratic programming with simple bounds and equality constraints, SIAM Journal on Numerical Analysis, 2005, to appear.

[53] Z. Dostál, An optimal algorithm for bound and equality constrained quadratic programming problems with bounded spectrum, 2004, submitted.

[54] Z. Dostál, A. Friedlander, S. A. Santos, Solution of contact problems of elasticity by FETI domain decomposition, Contemporary Mathematics 218, 1998, 82-93.

[55] Z. Dostál, A. Friedlander, S. A. Santos, Augmented Lagrangians with adaptive precision control for quadratic programming with simple bounds and equality constraints, SIAM Journal on Optimization 13, 4, 2003, 1120-1140.

[56] J. Schöberl, Efficient contact solvers based on domain decomposition techniques, Comput. Math. Appl. 42, 8-9, 1998, 1217-1228.

[57] D. Stefanica, FETI and FETI-DP Methods for Spectral and Mortar Spectral Elements: A Performance Comparison, J. Sci. Comp., 17, 2002, 629-637.

[58] Z. Dostál, and M. Lesoinne, Accelerating contact area detection in FET-DPC methods, the Proceedings of the 16th International Conference on Domain Decomposition Methods, New York, 2005.

[59] A. Klawonn, O. B. Widlund, M. Dryja, Dual-Primal FETI Methods for Three-dimensional Elliptic Problems with Heterogeneous Coefficients, SIAM J. Numer. Anal., Vol. 40, 2002, 159-179.

[60] R. Kornhuber, Adaptive monotone multigrid methods for nonlinear variational problems, Teubner-Verlag, Stuttgart, 1997.

[61] R. Kornhuber, and R. Krause, Adaptive multigrid methods for Signorini's problem in linear elasticity, Computer Visualization in Science 4, No.1, 2001, 9-20.

[62] A. Friedlander, J. M. Martínez, and M. Raydan, A new method for large scale box constrained quadratic minimization problems, Optimization Methods and Software 5, 1995, 57-74.

[63] S. Fučík, and A. Kufner, Nonlinear Differential Equations, Elsevier, Amsterdam 1980.

[64] D. Rixen, Extended Preconditioners for the FETI Method Applied to Constrained Problems, Int. J. Num. Meth. Eng., 54, 2002, 1-26.

[65] K. F. Traore, C. Farhat, M. Lesoinne, and D. Dureisseix, A domain decomposition method with Lagrange multipliers for the massively parallel solution of large-scale contact problems. In Proceedings of the 5th World Congress on Computational Mechanics (WCCM V), 2002.

[66] B. Wohlmuth, Discretization Methods and Iterative Solvers Based on Domain Decomposition, Springer, Berlin, 2001.

[67] B. Wohlmuth, and R. Krause, Monotone methods on nonmatching grids for nonlinear contact problems, SIAM J. Sci. Comput. 25, 2003, 324-347.

[68] Z. Dostál, and D. Horák, Scalable FETI with Optimal Dual Penalty for Semicoercive Variational Inequalities, Contemporary Mathematics 329, 2003, 79-88.

[69] Z. Dostál, and D. Horák, On scalable algorithms for numerical solution of variational inequalities based on FETI and semi-monotonic augmented Lagrangians, Proceedings of DDM15, Ed. R. Kornhuber et al., R. H. W. Hoppe, J. Périaux, O. Pironneau, O. B. Widlund and J. Xu, Lecture Notes in Computational Science and Engineering, Berlin, 2003, 487-494.

[70] C. Farhat, J. Mandel, The two-level FETI method for static and dynamic plate problems - Part I: An optimal iterative solver for biharmonic systems, Comput. Methods Appl. Mech. Engrg. 155, 1998, 129-152.

[71] J. Mandel, Étude algébrique d'une méthode multigrille pour quelques problémes de frontiére libre (French), Comptes Rendus de l'Academie des Sciences, S.r. I 298, 1984, 469-472.

[72] J. Mandel, C. R. Dohrmann, Convergence of a Balancing Domain Decomposition by Constraints and Energy Minimization, Technical Report, University of Colorado at Denver, 2002.

[73] J. Mandel, C. R. Dohrmann, R. Tezaur, An Algebraic Theory for Primal and Dual Substructuring Methods by Constraints, Applied Numerical Mathematics, 2005.

[74] J. Mandel, R. Tezaur, C. Farhat, A scalable substructuring method by Lagrange multipliers for plate bending problems, SIAM J. Numer. Anal. 36, No.5, 1999, 1370-1391.

[75] J. Mandel, R. Tezaur, On the Convergence of a Dual-Primal Substructuring Method, Numerische Mathematik 88, 2001, 543-558.

[76] G. Rebel, C. Farhat, M. Lesoinne, P. Avery, A scalable Dual-Primal domain decomposition method for the solution of contact problems with friction, 7th U.S. National Congress on Computational Mechanics, 2003.