Department
of Applied Mathematics

Faculty of Electrical Engineering and Computer Science
VŠB-Technical University of Ostrava

DOCTORAL THESIS

Marta Jarošová
2010

Effective implementation of some algorithms for solving quadratic programming problems

Marta Jarošová née Domorádová

Ostrava, 2010

Field of study
Computer Science and Applied Mathematics

Supervisor
Prof. RNDr. Zdeněk Dostál, DSc.

"Science . . . never solves a problem without creating ten more."

George Bernard Shaw

Acknowledgements

First of all I would like to express my deep and sincere gratitude to my supervisor Prof. Zdeněk Dostál for his motivation, patience, enthusiasm, and immense knowledge. His guidance and lots of very interesting discussions, not only about math, helped me in all the time of research and writing of this thesis.

My sincere thanks also go to Prof. Axel Klawonn and Dr. Oliver Rheinbach from University of Duisburg-Essen for offering me an internship opportunity in their group and giving me insight into their work about the transformation of basis.

I am indeed grateful to Dr. Alex Markopoulos for helping me with the implementation of the codes to the MatSol library.

Last but not the least, I would like to thank to my family: to my parents for everything they have done for me and to my husband for supporting me at all a times.

This research was supported by the grants GA CR 201/07/0294, the Ministry of Education of the Czech Republic No. MSM6198910027 and GA CR 103/09/H078.

Abstract

In this thesis we consider preconditioning strategies of algorithms for the solution of linear elasticity contact problems. We are interested especially in the preconditioning strategies which result in improved bounds on the rate of convergence. Let us point out that this goal cannot be achieved by a variant of preconditioning in face, since such preconditioning affects only the linear steps of the algorithm, but not the nonlinear steps.

We consider two strategies exploiting the edge averages for FETI-DP (Dual-Primal Finite Element Tearing and Interconnecting) methods. The first is preconditioning by conjugate projector. In the case, when it is combined with FETI-DP method, the Lagrange multipliers corresponding to the variables of the coinciding edges are aggregated.

The second one an explicit transformation of basis uses certain edge averages, which are introduced as new, additional primal variables.

For a special case, it is shown that both methods iterate in the same space and thus have the same rate of convergence. This theoretical result is confirmed by the solution of a model boundary variational inequality. At the end of the thesis we show some results of the numerical experiments from 2D linear elasticity, where the improvement of the rate of convergence is illustrated.

Keywords

Transformation of basis, edge averaging, conjugate projector, preconditioning, contact problems, variational inequalities, FETI-DP.

Contents

1 Introduction

The field of our interest is the solution of contact problems. Such problems arise whenever deformable bodies interact by touching each other. Contact problems are relevant to numerous applications such as sheet metal forming or computation of elastic deformation of a tire which interacts with an asphalt surface of the street. It is not surprising that contact problems have always occupied a position of special importance in the mechanics of solids.

In contact problems, the deformation of a system of bodies does not only depend on applied forces which are known a priori, but also on induced contact stresses on the a priori unknown region of contact. Description of the conditions of equilibrium of a system of elastic bodies in mutual contact includes inequality constraints, i.e., non-penetration conditions, which make the solution of the corresponding contact problem strongly nonlinear.

This thesis is motivated by an effort to improve the recently proposed scalable algorithms for the solution of contact problems. Let us recall that an algorithm is numerically scalable if the cost of the solution is nearly proportional to the number of unknowns, and it enjoys parallel scalability if the time required for the solution can be reduced nearly proportionally to the number of available processors or processor cores. Such fully scalable algorithms for a numerical solution of linear problems can be constructed using domain decomposition techniques.

The first scalable domain decomposition method with Lagrange multipliers for linear problems, the FETI-1 method, was introduced in the 90s [18]. It was quite challenging to obtain similar results for variational inequalities. Since the cost of the solution of a linear problem is proportional to the number of variables, a scalable algorithm must identify the active constraints in a sense without additional computational cost.

In this thesis we combine the results on quadratic programming [6, 14], in particular development of the algorithms for the solution of bound and/or equality constrained quadratic programming problems with the rate of convergence in the bounds on the spectrum of the Hessian matrix, with the duality-based methods that can be even more successful for the solution of variational inequalities than for linear

problems. Duality turns the general inequality constraints into bound constraints for free; this is an aspect which is not exploited in the solution of linear problems.

We will rely on results on FETI (Finite Element Tearing and Interconnecting) or a special analysis of FETI-DP (dual-primal FETI) [13, 12]. These methods are based on a decomposition of the original domain into non-overlapping subdomains and the application of the duality. The continuity of the solution across the subdomain interfaces is then enforced by Lagrange multipliers and the primal problem is reduced to a small, better conditioned, bound constrained quadratic programming problem. An important feature of this approach is that the solution of the system of such subproblems may be efficiently parallelized. In general, the main idea behind the non-overlapping domain decomposition methods is the decomposition of the spatial domain into smaller subdomains overlapping only on their interfaces, and then, instead of the large problem formulated on the original domain, we solve many smaller problems formulated on the subdomains. These subproblems are linked together by suitable transmission conditions. The idea of domain decomposition is also quite natural, for instance, when different physical models are needed to be used in different parts of the domain.

We are concerned especially with the FETI-DP method, where, in the simplest version for two dimensional elliptic problems, continuity of the solution along the subdomain interfaces is enforced by Lagrange multipliers except for the subdomain corners, which remain primal variables. We can think of this as a result of incisions in the mesh along the interface leaving only the subdomain corner nodes attached. The FETI-DP method was first introduced by Farhat, Lesoinne, Le Tallec, Pierson, and Rixen [15] and was analyzed for scalar two dimensional problems by Mandel and Tezaur [32]; see also Klawonn, Rheinbach, and Widlund [27], where the analysis is also extended to subdomains with very irregular boundaries. It is sometimes important to replace or enhance the coarse problem of the dual-primal FETI method, especially in three space dimensions. A possibility is to introduce certain edge or face averages or edge first order moments, either additionally or instead of the assembly in a selected number of primal variables; see, e.g., Farhat, Lesoinne, Pierson [16], Klawonn and Widlund [23, 28, 29], Klawonn, Widlund, and Dryja [30], and Klawonn and Rheinbach [25, 26]. Since the elimination of the primal variables is carried out by the Gaussian elimination method, the dual-primal FETI method is also denoted as exact FETI-DP method. For a large number of subdomains and processors, the exact elimination of the primal variables can lead to a deterioration of the parallel scalability. A remedy is given by the inexact FETI-DP methods; see Klawonn and Rheinbach [26], which have been shown to be scalable to more than 65 000 processor cores [22, 39].

A natural way how to improve the rate of convergence of the algorithms is to use preconditioning. This is a challenging task as the standard preconditioners typically transform variables, so that they also transform the bound constraints into more general inequality constraints; recall that there are no algorithms with a rate of convergence expressed in matrix-vector multiplications for the solution of quadratic programming problems with general inequality constraints. Though it is rather straightforward to precondition linear steps, the application of this idea goes back at

least to O'Leary [37], such preconditioning technique (also called preconditioning in face) does not affect the nonlinear steps, and thus does not result in an improvement of the rate of convergence.

The only successful result so far in this direction is the preconditioning by a conjugate projector, proposed for linear systems independently by Marchuk and Kuznetzov [33], Nicolaides [34], and Dostál [10] and later extended for bound constrained problems by Domorádová and Dostál [4]. It was shown that this approach, using projectors which do not affect the constrained variables, does result in improved bounds on the rate of convergence [4].

An unpleasant drawback of the preconditioning by a projector is the cost of the conjugate projector. To improve the efficiency of this approach, Dostál proposed to examine the relation between averaging and the conjugate projector. There relations were found and presented by Jarošová, Klawonn, and Rheinbach [21]. In [21], the FETI-DP method using edge constraints implemented by using a transformation of basis is compared with a FETI-DP method that was combined with a related projector preconditioning approach. It is shown that both methods iterate in the same finite element subspace and have the same rate of convergence. This is an important result, since the explicit construction of the dual matrix in the projector can be replaced by the transformation of basis which works locally and can easily be parallelized.

The thesis is arranged into three parts. The first part deals with linear problems. In Chapter 2 we introduce a linear model problem on which we are able to explain easily the ideas and methods relevant for our research. Chapter 3 is devoted to basic direct methods and the conjugate gradient method (CG), Chapter 4 describes the main idea of preconditioning, preconditioned conjugate gradient method (PCG), some basic preconditioners, and preconditioning by conjugate projectors. In Chapter 5 the Dual-Primal Finite Element Tearing and Interconnecting (FETI-DP) method and special preconditioning strategies - preconditioning by a projector for FETI-DP and the Dirichlet preconditioner are introduced. Chapter 6 describes transformation of basis as a preconditioning strategy introducing edge averages as new, additional primal variables into the FETI-DP system.

The second part deals with nonlinear problems. In Chapter 7 we introduce a model contact problem serving for an easy description of the ideas and the methods relevant for our research. Chapter 8 is devoted to the active set based algorithm MPRGP (modified proportioning with reduced gradient projections) for the solution of bound constrained quadratic programing problems with the rate of convergence in terms of the spectral condition number of the Hessian matrix. Chapter 9 deals with the FETI-DP method for the solution of nonlinear problems. Chapter 10 describes the preconditioning of the nonlinear problems, the preconditioning in face, and variants of preconditioning by a conjugate projector. Chapter 11, transformation of basis for contact problems, contains the proof that the preconditioning by a conjugate projector and the transformation of basis assembling the averages as new primal variables iterate in the same finite element subspace and thus have the same rate of convergence.

The third part shows the results of numerical experiments, Chapter 12 shows the improved bounds on the rate of convergence, when the projector preconditioning is applied to the finite element discretization problem. Chapter 13 shows the same iteration counts for the projector preconditioning and transformation of basis, Chapter 14 illustrates on 2D Hertz problem, implemented in MatSol library, some possibilities of the choice of coarse problem nodes for the transformation of basis, and Chapter 15 shows, only for comparison, the results of the same problem solved by Total FETI (T-FETI) method. The conclusions of the thesis are summed up in Chapter 16.

Notation

Ω	open domain $(0,1) \times (0,1)$ with the boundary $\partial\Omega$
Γ_D	boundary of Ω, where the Dirichlet boundary condition is prescribed
Γ_N	boundary of Ω, where the Neumann boundary condition is prescribed
Γ_c	contact boundary of Ω
Υ	feasible set
K	stiffness matrix
$\mathbf{u}$	displacement variables
$\mathbf{f}$	load vector (righthand side)
ϕ	quadratic functional
$L^2(\Omega)$	space of square integrable functions on Ω
$H^1(\Omega)$	space of functions which are square integrable on Ω as well as their first derivatives in the sense of distributions
$\mathrm{Im}\mathsf{A}$	image of A
$\mathrm{Ker}\mathsf{A}$	kernel of A
O	zero matrix
I	identity matrix
$\mathbf{g}$	gradient vector
$\mathbf{p}$	search direction, conjugate direction
$\lambda_{min}, \lambda_{max}$	extremal eigenvalues
$\kappa(\mathsf{A})$	condition number of A
$\mathsf{P},\ \mathsf{Q} = \mathsf{I} - \mathsf{P}$	conjugate projectors
U	matrix of "aggregations"
Ω_i	subdomain
Γ	subdomains interface
$\mathbf{u}_I$	interior displacement variables
$\mathbf{u}_\Delta$	dual displacement variables
$\mathbf{u}_\Pi$	primal displacement variables

$\mathbf{u}_B$	interior and dual displacement variables, $\mathbf{u}_B = [\mathbf{u}_I, \mathbf{u}_\Delta]$
λ	Lagrange multipliers
L	extending map
B	matrix enforcing the continuity on subdomain interfaces and/or inequality constraints on contact boundary
L_0	Lagrangian function
F	dual matrix
$\mathbf{d}$	dual righthand side
T	transformation matrix
$\mathbf{u}_E$	edge variables for which the change of variables is considered, $\mathbf{u}_E = [\mathbf{u}_A, \mathbf{u}_\Delta]$
$\mathbf{u}_A$	averages (assembled to $\mathbf{u}_\Pi$)
$\mathbf{u}_V$	vertex constrained (assembled to $\mathbf{u}_\Pi$)
$\mathbb{R}^n$	n-dimensional real space

I

Linear problems

2 Model problem

In this section we introduce the model problem, on which we are able to explain easily the ideas and methods relevant for our research. This model problem is used throughout the first part of this thesis.

Let $\Omega = (0,1) \times (0,1)$ be an open domain with the boundary $\partial\Omega$, and by ν we denote the outward normal to $\partial\Omega$. Let us consider a two-dimensional mixed problem depicted in Figure 2.1 with a Dirichlet boundary condition on Γ_D and a Neumann boundary condition elsewhere (on Γ_N), such that

$$\begin{cases} -\Delta u &= f \quad \text{in } \Omega \\ u &= 0 \quad \text{on } \Gamma_D \\ \dfrac{\partial u}{\partial \nu} &= 0 \quad \text{on } \Gamma_N, \end{cases} \tag{2.1}$$

where $\Gamma_D = \{0\} \times [0,1]$ and $\Gamma_N = \partial\Omega \setminus \Gamma_D$ are disjoint subsets of $\partial\Omega$.

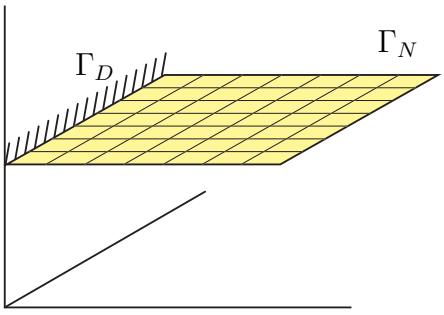


Figure 2.1: Two-dimensional problem with the Dirichlet boundary condition on Γ_D and the homogeneous Neumann boundary condition elsewhere (on Γ_N).

The solution to this problem is shown in Figure 2.2. It can be interpreted as the displacement of the membrane under the traction defined by the density f. The membrane is fixed on Γ_D.

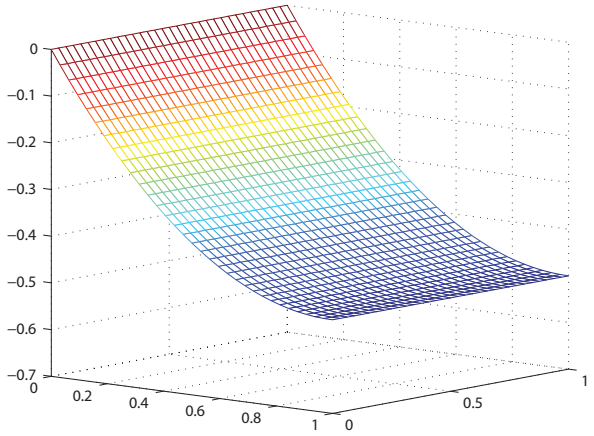


Figure 2.2: The solution to the model problem.

2.1 Weak Formulation

By multiplying the first row in (2.1) by a test function v, integrating by parts over Ω, and discarding the boundary terms, we obtain

$$\int_\Omega \nabla u \cdot \nabla v \, \mathrm{d}x = \int_\Omega f v \, \mathrm{d}x.$$

Let $L^2(\Omega)$ be the space of square integrable functions. It is a Hilbert space with the scalar product

$$(u, v)_{L^2(\Omega)} = \int_\Omega uv \, \mathrm{d}x. \tag{2.2}$$

Let $H^1(\Omega)$ be the space of functions which are square integrable as well as their first derivatives in the sense of distributions. Let us now introduce the Hilbert space

$$V = \{v \in H^1(\Omega) : v = 0 \text{ on } \Gamma_D\}. \tag{2.3}$$

The weak formulation of (2.1) then reads as follows: for a given $f \in L^2(\Omega)$,

$$\text{find } u \in V : \quad a(u, v) = (f, v), \quad \forall v \in V, \tag{2.4}$$

where $(\cdot, \cdot)$ is the scalar product in $L^2(\Omega)$ (2.2) and $a(u, v)$ denotes the bilinear form

$$a(u, v) = \int_\Omega \nabla u \cdot \nabla v \, \mathrm{d}x.$$

The Lax-Milgram lemma guarantees that the problem (2.4) has a unique solution if $a(\cdot, \cdot)$ is symmetric, continuous, and elliptic and $f(\cdot)$ is continuous [44].

2.2 Discretization and minimization

When using the finite element method, a solution of the weak problem (2.4) can be approximated by a solution of the finite dimensional problem obtained by replacing the infinite dimensional function space V, introduced in (2.3), by a finite dimensional subspace V_h. This leads to a following approximate problem:

$$\text{find } u_h \in V_h : \quad a(u_h, v_h) = (f, v_h), \quad \forall v_h \in V_h. \tag{2.5}$$

The space V_h is defined as the space of all functions that are piecewise linear and continuous on Ω_h and that vanish on the boundary Γ_D [40], where h denotes the discretization parameter.

Writing the solution u_h in the basis $\{\varphi_i\}$ as $u_h = \sum_i u_{h,i}\varphi_i$ leads to the system of linear equations

$$\mathsf{K}\mathbf{u} = \mathbf{f}, \tag{2.6}$$

where $[\mathsf{K}]_{ij} = a(\varphi_i, \varphi_j)$ and $[\mathbf{f}]_i = (f, \varphi_i)$. For more details about finite element method we refer to Zienkiewicz et al. [45].

The problem to find the solution of (2.6) is equivalent to the minimization problem

$$\min \phi(\mathbf{u}), \tag{2.7}$$

where

$$\phi(\mathbf{u}) = \frac{1}{2}\mathbf{u}^T\mathsf{K}\mathbf{u} - \mathbf{u}^T\mathbf{f} \tag{2.8}$$

is a quadratic functional with a symmetric positive definite matrix K.

Numerical solution 3

In this chapter we briefly recall the basic direct methods and describe the conjugate gradient method (CG), the iterative method used throught this thesis for solving an (auxiliary) linear systems with symmetric positive definite coefficient matrix.

3.1 Existence of a solution

We want to solve the system of linear equations

$$\mathsf{K}\mathbf{u} = \mathbf{f}. \tag{3.1}$$

We distinguish three situations [40] from the point of view of existence of a solution.

(i) The matrix K is nonsingular. There is a unique solution given by $\mathbf{u} = \mathsf{K}^{-1}\mathbf{f}$.

(ii) The matrix K is singular and $\mathbf{f} \in \mathrm{Im}\mathsf{K}$. Since $\mathbf{f} \in \mathrm{Im}\mathsf{K}$, there is an $\mathbf{u}_0$ such that $\mathsf{K}\mathbf{u}_0 = \mathbf{f}$. Moreover, $\mathbf{u}_0 + \nu$ is also a solution for any $\nu \in \mathrm{Ker}\mathsf{K}$. Since $\mathrm{Ker}\mathsf{K}$ is at least one-dimensional, there are infinitely many solutions.

(iii) The matrix K is singular and $\mathbf{f} \notin \mathrm{Im}\mathsf{K}$. There are no solutions.

3.2 Direct methods

When using direct methods for solving the system of linear equations, the problem to find the solution of the original linear system is transformed into the problem finding the solution of easily solvable linear system(s) with a triangular matrix.

Matrix factorization

The symmetric square matrix K can be decomposed as

$$\mathsf{K} = \mathsf{L}\mathsf{U},$$

where L and U are a lower triangular and an upper triangular matrix, respectively. This decomposition is called LU factorization. The solution $\mathbf{u}$ to the system (3.1) then can be evaluated from

$$\mathsf{L}\mathbf{z} = \mathbf{f} \quad \text{and} \quad \mathsf{U}\mathbf{u} = \mathbf{z}.$$

The symmetric positive definite matrix K can be decomposed into the product $\mathsf{L}\mathsf{L}^T$. This decomposition is called a Cholesky factorization.

When solving the system $\mathsf{L}\mathbf{z} = \mathbf{f}$ with the lower triangular matrix L, we can find the first unknown easily from the first equation, since there is only one. Then we can substitute it to other equations. By repeating this process, we can find all unknowns.

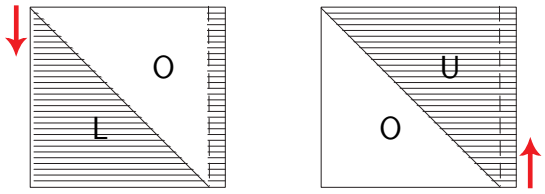


Figure 3.1: The process for solving linear systems with triangular matrices.

The same process can be used also in the case when solving the system $\mathsf{U}\mathbf{u} = \mathbf{z}$ with the upper triangular matrix U. We have to start with the last equation. The process for solving the systems with the triangular matrices is illustrated in Figure 3.1.

Gaussian elimination

The main idea of a Gaussian elimination method is to transform an augmented matrix $[\mathsf{K}|\mathbf{f}]$ to $[\mathsf{U}|\bar{\mathbf{f}}]$,

$$[\mathsf{K}|\mathbf{f}] \to [\mathsf{K}^{(1)}|\mathbf{f}^{(1)}] \to [\mathsf{K}^{(2)}|\mathbf{f}^{(2)}] \to \cdots \to [\mathsf{U}|\bar{\mathbf{f}}],$$

using elementary operations. The matrix $[\mathsf{K}^{(i)}|\mathbf{f}^{(i)}]$ is transformed to $[\mathsf{K}^{(i+1)}|\mathbf{f}^{(i+1)}]$ in such a way that the elements of the i-th column below the diagonal of the matrix $[\mathsf{K}^{(i+1)}|\mathbf{f}^{(i+1)}]$ are set to zero. This part, called a forward elimination, is depicted in Figure 3.2. The solution is then obtained from the system $\mathsf{U}\mathbf{u} = \bar{\mathbf{f}}$ in at most n steps in the process called a backward substitution. The algorithm requires $O(n^3)$ operations.

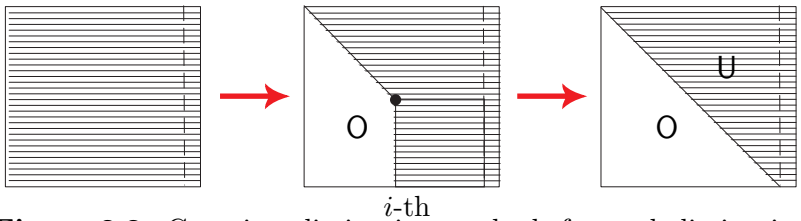


Figure 3.2: Gaussian elimination method: forward elimination.

Gauss-Jordan elimination

There is another way to do the second part: to transform the augmented matrix $[\mathsf{U}|\bar{\mathbf{f}}]$ to $[\,\mathsf{D}\,|\bar{\mathbf{u}}]$, where D is some diagonal matrix. The matrix $[\mathsf{U}^{(i)}|\bar{\mathbf{f}}^{(i)}]$ is transformed to $[\mathsf{U}^{(i+1)}|\bar{\mathbf{f}}^{(i+1)}]$ in such a way that the elements of the j-th row ($j = n - i$) on the righthand side of the diagonal of the matrix $\mathsf{U}^{(i+1)}$ are set to zero. The solution $\mathbf{u}$ is obtained from $[\,\mathsf{D}\,|\bar{\mathbf{u}}]$ by dividing rows by the corresponding diagonal element, so we obtain $[\,\mathsf{I}\,|\mathbf{u}]$, where I denotes identity matrix. The process described here, called a Gauss-Jordan elimination, is depicted in Figure 3.3.

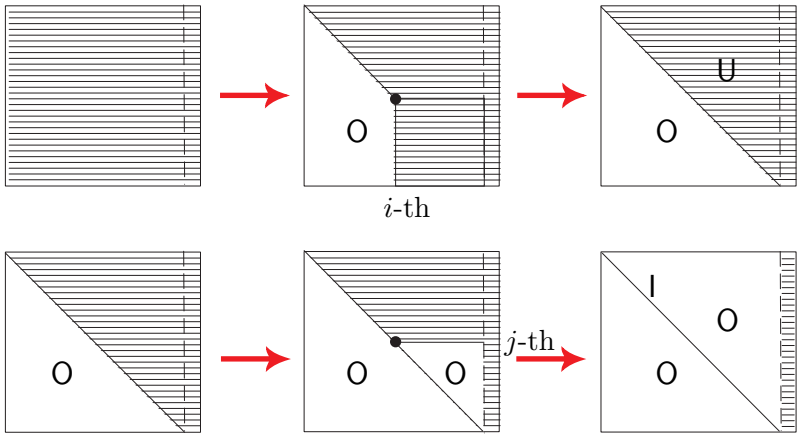


Figure 3.3: Gauss-Jordan elimination.

3.3 Conjugate gradient method

To solve an (auxiliary) linear systems throught this thesis we use the conjugate gradient method. It is an iterative method for solving linear systems with a symmetric positive definite coefficient matrix. The method was introduced in 1952 by Hestenes and Stiefel [19], but came into wide use in the mid-70's.

As was mentioned in Section 2.2, solving the linear system $\mathsf{K}\mathbf{u} = \mathbf{f}$ is equivalent to the minimization problem

$$\min \phi(\mathbf{u}), \tag{3.2}$$

where ϕ is the quadratic functional defined by (2.8).

The gradient of ϕ equals to the residual of the linear system and has the form

$$\nabla\phi(\mathbf{u}) = \mathsf{K}\mathbf{u} - \mathbf{f} = \mathbf{g}.$$

Let us express the vector $\mathbf{u}_{i+1}$ as

$$\mathbf{u}_{i+1} = \mathbf{u}_i - \alpha\mathbf{p}_i$$

with the steplength α and the search direction $\mathbf{p}_i$, i. e. the direction in which the minimization is sought. The coefficient α, the value for which $\phi(\mathbf{u}_{i+1})$ is minimal,

can be expressed by solving the problem

$$\min_{\alpha} \phi(\mathbf{u}_{i+1}).$$

Since

$$\begin{aligned} \phi(\mathbf{u}_{i+1}) = \phi(\mathbf{u}_i - \alpha\mathbf{p_i}) &= \frac{1}{2}\big\langle \mathsf{K}(\mathbf{u}_i - \alpha\mathbf{p_i}), (\mathbf{u}_i - \alpha\mathbf{p_i})\big\rangle - \big\langle \mathbf{f}, (\mathbf{u}_i - \alpha\mathbf{p_i})\big\rangle \\ &= \frac{1}{2}\langle \mathsf{K}\mathbf{u}_i, \mathbf{u}_i\rangle - \alpha\langle \mathsf{K}\mathbf{u}_i, \mathbf{p}_i\rangle + \frac{1}{2}\alpha^2\langle \mathsf{K}\mathbf{p}_i, \mathbf{p}_i\rangle \qquad (3.3) \\ &- \langle \mathbf{f}, \mathbf{u}_i\rangle + \alpha\langle \mathbf{f}, \mathbf{p_i}\rangle \end{aligned}$$

and

$$\frac{\partial\phi(\mathbf{u}_{i+1})}{\partial\alpha} = -\langle \mathsf{K}\mathbf{u}_i, \mathbf{p}_i\rangle + \alpha\langle \mathsf{K}\mathbf{p}_i, \mathbf{p}_i\rangle + \langle \mathbf{f}, \mathbf{p_i}\rangle = 0,$$

we can write

$$\alpha = \frac{\langle \mathsf{K}\mathbf{u}_i - \mathbf{f}, \mathbf{p}_i\rangle}{\langle \mathsf{K}\mathbf{p}_i, \mathbf{p}_i\rangle} = \frac{\langle \mathbf{g}_i, \mathbf{p}_i\rangle}{\langle \mathsf{K}\mathbf{p}_i, \mathbf{p}_i\rangle}.$$

The search directions $\mathbf{p}_i$ are chosen so that they are mutually conjugate with respect to the scalar product defined by

$$\langle \mathsf{K}\mathbf{p}_i, \mathbf{p}_{i+1}\rangle = 0,$$

where the vector $\mathbf{p}_{i+1}$ can be written as

$$\mathbf{p}_{i+1} = \mathbf{g}_{i+1} - \beta\mathbf{p}_i.$$

Using the last two formulas, we can write

$$\langle \mathsf{K}\mathbf{p}_i, \mathbf{g}_{i+1} - \beta\mathbf{p}_i\rangle = \langle \mathsf{K}\mathbf{p}_i, \mathbf{g}_{i+1}\rangle - \beta\langle \mathsf{K}\mathbf{p}_i, \mathbf{p}_i\rangle = 0$$

and

$$\beta = \frac{\langle \mathbf{g}_{i+1}, \mathsf{K}\mathbf{p}_i\rangle}{\langle \mathsf{K}\mathbf{p}_i, \mathbf{p}_i\rangle}.$$

The vector $\mathbf{g}_{i+1}$ can be written in the form

$$\mathbf{g}_{i+1} = \mathbf{g}_i - \alpha\mathsf{K}\mathbf{p}_i,$$

since

$$\mathbf{g}_{i+1} = \mathsf{K}\mathbf{u}_{i+1} - \mathbf{f} = \mathsf{K}\big(\mathbf{u}_i - \alpha\mathbf{p}_i\big) - \mathbf{f} = \big(\mathsf{K}\mathbf{u}_i - \mathbf{f}\big) - \alpha\mathsf{K}\mathbf{p}_i = \mathbf{g}_i - \alpha\mathsf{K}\mathbf{p}_i.$$

Lemma 1. *Let $\mathbf{u}_0 \in \mathbb{R}^n$ and let $\{\mathbf{p}_i\}$ be an arbitrary set of the conjugate directions. Then*

$$\langle \mathbf{g}_i, \mathbf{p}_j\rangle = 0, \quad j = 0, \ldots, i-1,$$

and $\mathbf{u}_i$ is the minimizer of $\phi(\mathbf{u})$ over the space

$$\mathbf{u}_0 + \operatorname{span}\{\mathbf{p}_0, \mathbf{p}_1, \ldots, \mathbf{p}_{i-1}\}.$$

Consequently, the sequence $\{\mathbf{u}_i\}$ converges to the solution $\mathbf{u}^*$ of (3.2) in at most n steps.

Algorithm 2. Conjugate gradient method

Given an spd matrix $\mathsf{K} \in \mathbb{R}^{n\times n}$, *n-vectors* $\mathbf{f}$ *and* $\mathbf{u}_0$.

Step 0. {*Initialization.*}
Compute $\mathbf{g}_0 = \mathsf{K}\mathbf{u}_0 - \mathbf{f}$ *and set* $\mathbf{p}_1 = \mathbf{g}_0$ *and* $i = 0$.
Step 1. {*Conjugate gradient loop.*}
while $\|\mathbf{g}_i\|$ *is not small*
$\alpha = \langle \mathbf{g}_i, \mathbf{p}_i\rangle / \langle \mathsf{K}\mathbf{p}_i, \mathbf{p}_i\rangle = \langle \mathbf{g}_i, \mathbf{g}_i\rangle / \langle \mathsf{K}\mathbf{p}_i, \mathbf{p}_i\rangle$
$\mathbf{u}_{i+1} = \mathbf{u}_i - \alpha\mathbf{p}_i$
$\mathbf{g}_{i+1} = \mathbf{g}_i - \alpha\mathsf{K}\mathbf{p}_i$
$\beta = \langle \mathbf{g}_i, \mathsf{K}\mathbf{p}_i\rangle / \langle \mathsf{K}\mathbf{p}_i, \mathbf{p}_i\rangle = -\langle \mathbf{g}_{i+1}, \mathbf{g}_{i+1}\rangle / \langle \mathbf{g}_i, \mathbf{g}_i\rangle$
$\mathbf{p}_{i+1} = \mathbf{g}_i - \beta\mathbf{p}_i$
$i = i + 1$
end while
Step 2. {*Return the solution.*}
$\hat{\mathbf{u}} = \mathbf{u}_i$

Let us sum up the properties of the conjugate gradient method. First, the gradients $\mathbf{g}_j$ are mutually orthogonal

$$\langle \mathbf{g}_i, \mathbf{g}_j\rangle = 0, \quad j = 0, \ldots, i-1,$$

and the conjugate directions $\mathbf{p}_j$ are K-conjugate

$$\langle \mathbf{p}_i, \mathsf{K}\mathbf{p}_j\rangle = 0, \quad j = 0, \ldots, i-1.$$

Second, each conjugate direction $\mathbf{p}_i$ and gradient $\mathbf{g}_i$ is contained in the Krylov subspace of degree i for $\mathbf{g}_0$, defined as

$$\mathcal{K}(\mathbf{g}_0; i) = \operatorname{span}\{\mathbf{g}_0, \mathsf{K}\mathbf{g}_0, \ldots, \mathsf{K}^i\mathbf{g}_0\} = \operatorname{span}\{\mathbf{p}_0, \mathbf{p}_1, \ldots, \mathbf{p}_i\}.$$

For the conjugate gradient method, shown in Algorithm 2, we obtain the following convergence result.

Lemma 3. *Let* K *be symmetric positive definite and let* $\mathbf{u}^*$ *denotes the solution of (3.2). Then the conjugate gradient method satisfies the error bound*

$$\|\mathbf{u}_{i+1} - \mathbf{u}^*\|_{\mathsf{K}}^2 \le \eta_{\mathsf{K}}^2 \|\mathbf{u}_i - \mathbf{u}^*\|_{\mathsf{K}}^2, \tag{3.4}$$

where the convergence factor is bounded by

$$\eta_{\mathsf{K}} = \frac{\sqrt{\kappa(\mathsf{K})} - 1}{\sqrt{\kappa(\mathsf{K})} + 1},$$

where $\kappa(\mathsf{K}) = \lambda_{max}/\lambda_{min}$ *is a condition number of the matrix* K.

More details about the conjugate gradient method can be found, e.g., in Nocedal and Wright [35, Chapter 5] or in Templates by Dongarra et al. [2].

Preconditioning for linear problems 4

Preconditioning is a key tool that we shall use to improve the convergence of iterative methods such as the conjugate gradients. In this chapter we describe the Preconditioned conjugate gradients, some basic preconditioners, and the preconditioning by conjugate projectors.

The main idea of the preconditioning is to transform the original linear system

$$\mathsf{K}\mathbf{u} = \mathbf{f} \tag{4.1}$$

into the preconditioned system

$$\mathsf{MK}\mathbf{u} = \mathsf{M}\mathbf{f}, \tag{4.2}$$

where the nonsingular matrix M is a preconditioner.

When solving (4.2) by Conjugate gradient (CG) algorithm, the convergence depends on the properties of MK instead on those of K. To ensure better properties of MK, M should be in some sense close to K, for example $\mathsf{M} \sim \mathsf{K}^{-1}$. If the preconditioner M is well chosen, i. e., if the condition number of MK is close to one, (4.2) may be solved much more rapidly then (4.1).

4.1 Basic preconditioners

Here we show some basic preconditioning techniques. Let us assume the splitting

$$\mathsf{K} = \mathsf{D} + \mathsf{E} + \mathsf{E}^T,$$

where D is diagonal of K and E is its strict lower triangular part. This splitting is depicted in Figure 4.1.

The easiest preconditioner is of the form

$$\mathsf{M}_J = \mathsf{D}^{-1}$$

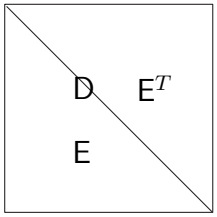


Figure 4.1: The splitting $\mathsf{K} = \mathsf{D} + \mathsf{E} + \mathsf{E}^T$.

and is called a Jacobi preconditioner. Another preconditioner derived from this splitting is the SSOR preconditioner [2]

$$\mathsf{M}_{SSOR}(\omega) = \frac{1}{2-\omega}\Big(\frac{1}{\omega}\mathsf{D} + \mathsf{E}\Big)^{-T}\Big(\frac{1}{\omega}\mathsf{D}\Big)\Big(\frac{1}{\omega}\mathsf{D} + \mathsf{E}\Big)^{-1}.$$

Taking $\omega = 1$ leads to the Symmetric Gauss-Seidel preconditioner

$$\mathsf{M}_{SGS} = (\mathsf{D} + \mathsf{E})^{-T}\mathsf{D}(\mathsf{D} + \mathsf{E})^{-1}.$$

We can use a lot of another preconditioning techniques, e.g., ILU, multigrid based preconditioner, and many others. More details can be found, e.g., in Saad [40].

4.2 Preconditioned conjugate gradient method

The preconditioned CG algorithm can be derived from the CG algorithm by using M inner product, see, e.g., Saad [40, Section 9.2]. The Preconditioned CG method is shown in Algorithm 4.

Algorithm 4. Preconditioned conjugate gradient method

Given an spd matrix $\mathsf{K} \in \mathbb{R}^{n\times n}$*, spd preconditioner* $\mathsf{M} \in \mathbb{R}^{n\times n}$*, n-vectors* $\mathbf{f}$ *and* $\mathbf{u}_0$.

Step 0. {*Initialization.*}
Compute $\mathbf{z}_0 = \mathsf{M}\mathbf{g}_0$ *and set* $\mathbf{p}^0 = \mathbf{z}^0$, $i = 0$.
Step 1. {*PCG loop.*}
while $\|\mathbf{g}_i\|$ *is not small*
$\alpha = \langle \mathbf{z}_i, \mathbf{g}_i\rangle / \langle \mathbf{p}_i, \mathsf{K}\mathbf{p}_i\rangle$
$\mathbf{u}_{i+1} = \mathbf{u}_i + \alpha\mathbf{p}_i$
$\mathbf{g}_{i+1} = \mathbf{g}_i - \alpha\mathsf{K}\mathbf{p}_i$
$\mathbf{z}_{i+1} = \mathsf{M}\mathbf{g}_{i+1}$
$\beta = \langle \mathbf{z}_{i+1}, \mathbf{g}_{i+1}\rangle / \langle \mathbf{z}_i, \mathbf{g}_i\rangle$
$\mathbf{p}_{i+1} = \mathbf{z}_{i+1} + \beta\mathbf{p}_i$
$i = i + 1$
end while
Step 2. {*Return solution.*}
$\hat{\mathbf{u}} = \mathbf{u}_i$

4.3 Conjugate projectors

The main goal of the investigation is to modify the classical conjugate gradient method in such a way that in dependence an the initial approximation the number of iterations becomes as small as possible. To this end the classical method is reformulated in such a way that the approximations belong to a predetermined affine subspace. The preconditioning by the conjugate projector considered here, was proposed for linear systems independently by Marchuk and Kuznetsov [33], Nicolaides [34], and Dostál [10]. First, we give an overview of the properties of the projectors, see, e.g., [11].

Let us describe two important subspaces associated with each mapping. Each matrix $\mathsf{A} \in \mathbb{R}^{m\times n}$ defines the mapping, which assigns to each $\mathbf{v} \in \mathbb{R}^n$ the vector $\mathsf{A}\mathbf{v} \in \mathbb{R}^m$. A *range* or *image space* of A is defined as

$$\mathrm{Im}\mathsf{A} = \{\mathsf{A}\mathbf{v} : \mathbf{v} \in \mathbb{R}^n\}$$

and a *kernel* or *null space* as

$$\mathrm{Ker}\mathsf{A} = \{\mathbf{v} \in \mathbb{R}^n : \mathsf{A}\mathbf{v} = \mathbf{o}\}.$$

A *projector* is a square matrix P that satisfies

$$\mathsf{P}^2 = \mathsf{P}.$$

If P is projector, then $\mathsf{Q} = \mathsf{I} - \mathsf{P}$ and P^T are also projectors as

$$(\mathsf{I} - \mathsf{P})^2 = \mathsf{I} - 2\mathsf{P} + \mathsf{P}^2 = \mathsf{I} - \mathsf{P} \quad \text{and} \quad (\mathsf{P}^T)^2 = (\mathsf{P}^2)^T = \mathsf{P}^T.$$

A vector $\mathbf{v} \in \mathrm{Im}\mathsf{P}$, i.e. $\mathbf{v} = \mathsf{P}\mathbf{v}$, iff there is $\mathbf{x} \in \mathbb{R}^n$ such that $\mathbf{v} = \mathsf{P}\mathbf{x}$, so that

$$\mathsf{P}\mathbf{v} = \mathsf{P}(\mathsf{P}\mathbf{x}) = \mathsf{P}\mathbf{x} = \mathbf{v}.$$

Since for any $\mathbf{v} \in \mathbb{R}^n$

$$\mathbf{v} = \mathsf{P}\mathbf{v} + (\mathsf{I} - \mathsf{P})\mathbf{v},$$

it simply follows that $\mathrm{Im}\mathsf{Q} = \mathrm{Ker}\mathsf{P}$ and $\mathrm{Im}\mathsf{P} = \mathrm{Ker}\mathsf{Q}$,

$$\mathbb{R}^n = \mathrm{Im}\mathsf{P} \oplus \mathrm{Ker}\mathsf{P}, \quad \text{and} \quad \mathrm{Ker}\mathsf{P} \cap \mathrm{Im}\mathsf{P} = \{0\}.$$

We say that P is a projector onto $\mathcal{U} = \mathrm{Im}\mathsf{P}$ along $\mathcal{V} = \mathrm{Ker}\mathsf{P}$ and Q is a complementary projector onto $\mathcal{V}$ along $\mathcal{U}$. It is also easy to see that $\mathsf{PQ} = \mathsf{QP} = \mathsf{O}$, since

$$\mathsf{PQ} = \mathsf{P}(\mathsf{I} - \mathsf{P}) = \mathsf{P} - \mathsf{P}^2 = \mathsf{O} \quad \text{and} \quad \mathsf{QP} = (\mathsf{I} - \mathsf{P})\mathsf{P} = \mathsf{P} - \mathsf{P}^2 = \mathsf{O}.$$

Let $\mathsf{K} \in \mathbb{R}^{n\times n}$ be a symmetric positive definite matrix. A projector P is an K-*conjugate projector* or briefly a *conjugate projector* if $\mathrm{Im}\mathsf{P}$ is K-conjugate to $\mathrm{Ker}\mathsf{P}$, or equivalently

$$\mathsf{P}^T\mathsf{K}(\mathsf{I} - \mathsf{P}) = \mathsf{P}^T\mathsf{K} - \mathsf{P}^T\mathsf{K}\mathsf{P} = \mathsf{O}.$$

It follows that $\mathsf{Q} = \mathsf{I} - \mathsf{P}$ is also a conjugate projector,

$$\mathsf{P}^T\mathsf{K} = \mathsf{K}\mathsf{P} = \mathsf{P}^T\mathsf{K}\mathsf{P}, \quad \text{and} \quad \mathsf{Q}^{\mathrm{T}}\mathsf{K} = \mathsf{K}\mathsf{Q} = \mathsf{Q}^{\mathrm{T}}\mathsf{K}\mathsf{Q}. \tag{4.3}$$

If $\mathbf{u} \in \mathsf{K}\mathcal{V}$, then

$$\mathsf{Q}^T\mathsf{K}\mathsf{Q}\mathbf{u} = \mathsf{K}\mathsf{Q}\mathbf{u},$$

which implies

$$\mathsf{Q}^T\mathsf{K}\mathsf{Q}(\mathsf{K}\mathcal{V}) \subseteq \mathsf{K}\mathcal{V}. \tag{4.4}$$

Thus $\mathsf{K}\mathcal{V}$ is an invariant subspace of $\mathsf{Q}^T\mathsf{K}\mathsf{Q}$.

The following lemma shows that the mapping which assigns to each $\mathbf{u} \in \mathsf{K}\mathcal{V}$ the vector $\mathsf{Q}\mathbf{u} \in \mathcal{V}$ is expansive.

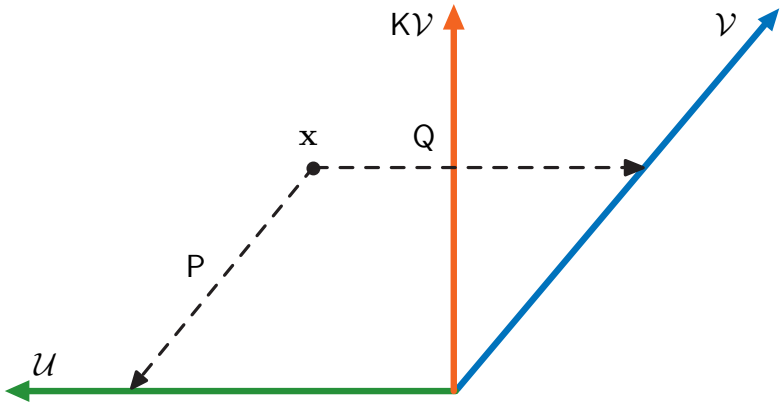


Figure 4.2: Geometric illustration of the conjugate projectors: $\mathcal{V} = \mathrm{Im}\mathsf{Q}$ and $\mathcal{U} = \mathrm{Im}\mathsf{P}$.

Lemma 5. *Let* Q *denote a conjugate projector on* $\mathcal{V}$*. Then for any* $\mathbf{u} \in \mathsf{K}\mathcal{V}$

$$\|\mathsf{Q}\mathbf{u}\| \geq \|\mathbf{u}\| \tag{4.5}$$

and

$$\mathcal{V} = \mathsf{Q}(\mathsf{K}\mathcal{V}). \tag{4.6}$$

Proof. Let us assume that $\mathbf{u} \in \mathsf{K}\mathcal{V}$, so there is an $\mathbf{y} \in \mathbb{R}^n$ such that $\mathbf{u} = \mathsf{K}\mathsf{Q}\mathbf{y}$. It follows that

$$\mathsf{Q}^T\mathbf{u} = \mathsf{Q}^T\mathsf{K}\mathsf{Q}\mathbf{y} = \mathsf{K}\mathsf{Q}\mathbf{y} = \mathbf{u}$$

and $\mathbf{u}^T\mathsf{Q}\mathbf{u} = \mathbf{u}^T\mathsf{Q}^T\mathbf{u} = \|\mathbf{u}\|^2$, which implies

$$\|\mathsf{Q}\mathbf{u}\|^2 = \mathbf{u}^T\mathsf{Q}^T\mathsf{Q}\mathbf{u} = \mathbf{u}^T\left((\mathsf{Q}^T - \mathsf{I}) + \mathsf{I}\right)((\mathsf{Q} - \mathsf{I}) + \mathsf{I})\,\mathbf{u} = \|(\mathsf{Q} - \mathsf{I})\mathbf{u}\|^2 + \|\mathbf{u}\|^2.$$

This proves (4.5). To prove (4.6) observe that $\mathcal{V} = \mathrm{Im}\mathsf{Q}$, such that $\mathcal{V} = \mathsf{Q}(\mathbb{R}^n) \supseteq \mathsf{Q}(\mathsf{K}\mathcal{V})$. Since K is nonsingular and the mapping which assigns to each $\mathbf{u} \in \mathsf{K}\mathcal{V}$ the vector $\mathsf{Q}\mathbf{u}$ is injective, it is enough to use a dimension argument to finish the proof. □

If $\mathcal{U}$ is the subspace spanned by the columns of a full column rank matrix $\mathsf{U} \in \mathbb{R}^{n\times p}$, then

$$\mathsf{P} = \mathsf{U}(\mathsf{U}^T\mathsf{K}\mathsf{U})^{-1}\mathsf{U}^T\mathsf{K} \tag{4.7}$$

is a conjugate projector onto $\mathcal{U}$.

Using the projector P it is possible to solve the auxiliary problem

$$\min_{\mathbf{u}\in\mathcal{U}} \phi(\mathbf{u}) = \min_{\mathbf{y}\in\mathbb{R}^p} \phi(\mathsf{U}\mathbf{y}) = \min_{\mathbf{y}\in\mathbb{R}^p} \frac{1}{2}\mathbf{y}^T\mathsf{U}^T\mathsf{K}\mathsf{U}\mathbf{y} - \mathbf{f}^T\mathsf{U}\mathbf{y}.$$

By the gradient argument, we get that the minimizer $\mathbf{u}^0 = \mathsf{U}\mathbf{y}^0$ of ϕ over $\mathcal{U}$ is defined by

$$\mathsf{U}^T\mathsf{K}\mathsf{U}\mathbf{y} = \mathsf{U}^T\mathbf{f}, \tag{4.8}$$

hence

$$\mathbf{u}^0 = \mathsf{U}(\mathsf{U}^T\mathsf{K}\mathsf{U})^{-1}\mathsf{U}^T\mathbf{f} = \mathsf{P}\mathsf{K}^{-1}\mathbf{f}. \tag{4.9}$$

Thus we can find the minimum of ϕ over $\mathcal{U}$ effectively whenever we are able to solve (4.8).

We shall use the conjugate projectors P and $\mathsf{Q} = \mathsf{I} - \mathsf{P}$ to decompose our minimization problem (3.2) into the minimization on $\mathcal{U}$ and the minimization on $\mathcal{V} = \mathrm{Im}\mathsf{Q}$. In particular, we shall use three observations. First, using Lemma 5, we get that the mapping which assigns to each $\mathbf{u} \in \mathsf{K}\mathcal{V}$ a vector $\mathsf{Q}\mathbf{u} \in \mathcal{V}$ is an isomorphism. Second, using (4.9), we get

$$\mathbf{g}^0 = \mathsf{K}\mathbf{u}^0 - \mathbf{f} = \mathsf{K}\mathsf{P}\mathsf{K}^{-1}\mathbf{f} - \mathbf{f} = \mathsf{P}^T\mathbf{f} - \mathbf{f} = -\mathsf{Q}^T\mathbf{f}. \tag{4.10}$$

Since

$$\mathrm{Im}\mathsf{Q}^{\mathrm{T}} = \mathrm{Im}(\mathsf{Q}^{\mathrm{T}}\mathsf{K}) = \mathrm{Im}(\mathsf{K}\mathsf{Q}) = \mathsf{K}\mathcal{V} \tag{4.11}$$

and $\mathbf{g}^0 \in \mathrm{Im}\mathsf{Q}^{\mathrm{T}}$ by (4.10), we get that $\mathbf{g}^0 \in \mathsf{K}\mathcal{V}$. Finally, observe that if $\mathbf{o} \neq \mathbf{u} \in \mathsf{K}\mathcal{V}$, then by Lemma 5 $\mathsf{Q}\mathbf{u} \neq \mathbf{o}$, such that $\mathbf{u}^T\mathsf{Q}^T\mathsf{K}\mathsf{Q}\mathbf{u} > 0$. Thus the restriction $\mathsf{Q}^T\mathsf{K}\mathsf{Q}|\mathsf{K}\mathcal{V}$ is positive definite.

We write

$$\begin{aligned}
\min \phi(\mathbf{u}) &= \min_{\mathbf{x}\in\mathcal{U},\mathbf{y}\in\mathcal{V}} \phi(\mathbf{x}+\mathbf{y}) = \min_{\mathbf{x}\in\mathcal{U}} \phi(\mathbf{x}) + \min_{\mathbf{y}\in\mathcal{V}} \phi(\mathbf{y}) \\
&= \phi(\mathbf{u}^0) + \min_{\mathbf{y}\in\mathcal{V}} \phi(\mathbf{y}) = \phi(\mathbf{u}^0) + \min_{\mathbf{y}\in\mathsf{K}\mathcal{V}} \frac{1}{2}\mathbf{y}^T\mathsf{Q}^T\mathsf{K}\mathsf{Q}\mathbf{y} - \mathbf{f}^T\mathsf{Q}\mathbf{y} \\
&= \phi(\mathbf{u}^0) + \min_{\mathbf{y}\in\mathsf{K}\mathcal{V}} \frac{1}{2}\mathbf{y}^T\mathsf{Q}^T\mathsf{K}\mathsf{Q}\mathbf{y} + \mathbf{y}^T\mathbf{g}^0,
\end{aligned}$$

where $\mathbf{u}^0$ is defined by (4.9) and $\mathbf{g}^0$ by (4.10). Then the solution $\hat{\mathbf{u}}$ can be expressed as

$$\hat{\mathbf{u}} = \mathbf{u}^0 + \mathsf{Q}\hat{\mathbf{y}},$$

where $\hat{\mathbf{y}}$ is the solution on $\mathsf{K}\mathcal{V}$.

The Conjugate gradient method with projector preconditioning is shown in Algorithm 6. For a convergence results we refer the interested reader to Dostál [10].

Algorithm 6. Conjugate gradients with projector preconditioning

Given an spd matrix $\mathsf{K} \in \mathbb{R}^{n\times n}$*, a full column rank matrix* $\mathsf{U} \in \mathbb{R}^{n\times p}$*, n-vector* $\mathbf{f}$*, projectors* P *defined by (10.12) and* $\mathsf{Q} = \mathsf{I} - \mathsf{P}$*.*

Step 0. {*Initialization.*}
Compute $\mathbf{u}_0 = \mathsf{P}\mathsf{K}^{-1}\mathbf{f} = \mathsf{U}(\mathsf{U}^T\mathsf{K}\mathsf{U})^{-1}\mathsf{U}\mathbf{f}$,
$\mathbf{g}_0 = \mathsf{K}\mathbf{u}_0 - \mathbf{f}$, $\mathbf{z}_0 = \mathsf{Q}\mathbf{g}_0$, $\mathbf{p}_0 = \mathbf{z}_0$ *and set* $i = 0$.
Step 1. {*Conjugate gradient loop.* }
while $\|\mathbf{g}_i\|$ *is not small*
$\alpha = \langle \mathbf{z}_i, \mathbf{g}_i\rangle / \langle \mathbf{p}_i, \mathsf{K}\mathbf{p}_i\rangle$
$\mathbf{u}_{i+1} = \mathbf{u}_i + \alpha\mathbf{p}_i$
$\mathbf{g}_{i+1} = \mathbf{g}_i - \alpha\mathsf{K}\mathbf{p}_i$
$\mathbf{z}_{i+1} = \mathsf{Q}\mathbf{g}_{i+1}$
$\beta = \langle \mathbf{g}_{i+1}, \mathbf{z}_{i+1}\rangle / \langle \mathbf{g}_i, \mathbf{z}_i\rangle$
$\mathbf{p}_{i+1} = \mathbf{z}_{i+1} + \beta\mathbf{p}_i$
$i = i + 1$
end while
Step 2. {*Return solution.*}
$\widehat{\mathbf{u}} = \mathbf{u}_i$

Projector defined by aggregations

The matrix U, used in projector P (4.7), can be defined, e.g. by the elements of the aggregation bases such as those depicted in Figure 4.3. Each element of such basis can be represented by the column $\mathbf{u}_k \in \mathbb{R}^n$, $k = 1, \ldots, p$, with all the entries equal to zero except the entries which correspond to the aggregated variables and are equal to one. For example, if $\mathbf{u}_k$ corresponds to the element of the aggregation basis depicted in Figure 4.3, then $u_{ik} = 1$ and $u_{jk} = 0$, thus the matrix U is of the form

$$\mathsf{U} = \begin{array}{c} \left[\begin{array}{ccc} & & \\ & & \\ \hline & 1 & \\ \hline & 0 & \\ \hline & & \\ & & \\ & & \end{array}\right] \begin{array}{l} \\ \\ i-th \\ j-th \\ \\ \\ \\ \end{array} \\ k-th \end{array} . \tag{4.12}$$

Projector defined by the traces of linear functions

Another possibility is to define matrix U by the traces of linear functions on the coarse grid such as that depicted in Figure 4.4. Each element of such basis can be represented by the column $\mathbf{u}_m \in \mathbb{R}^n$, $m = 1, \ldots, p$, with all the entries equal to zero except the entries which correspond to the support. For example, if $\mathbf{u}_m$ corresponds

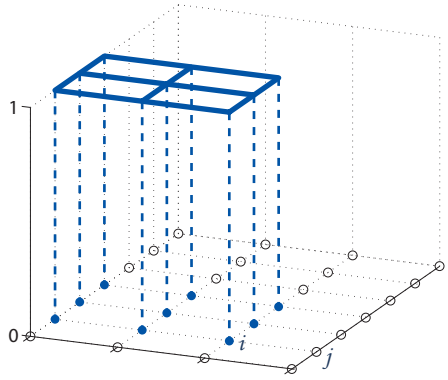


Figure 4.3: Aggregation basis function.

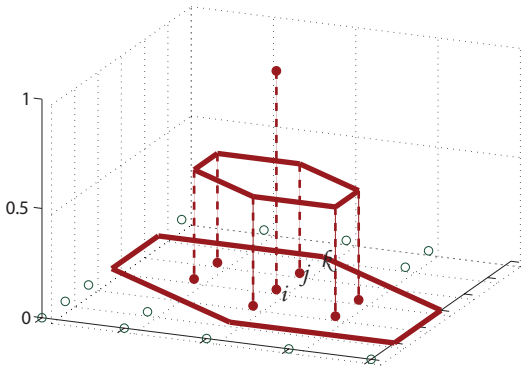


Figure 4.4: Coarse linear basis function.

to the coarse space basis function depicted in Figure 4.4, then $u_{im} = 1$, $u_{jm} = 0.5$, and $u_{km} = 0$, thus the matrix U is of the form

$$\mathsf{U} = \begin{bmatrix} & & \\ & 1 & \\ & 1/2 & \\ & 0 & \\ & & \end{bmatrix} \begin{matrix} \\ i-th \\ j-th \\ k-th \\ \\ \end{matrix} . \qquad (4.13)$$

$m - th$

Dual-Primal FETI Method

5

In this chapter we consider the Dual-Primal Finite Element Tearing and Interconnecting (FETI-DP) method, originally introduced by Farhat, Lesoinne, Le Tallec, Pierson, and Rixen [15], followed by Mandel and Tezaur [32] with theory for two dimensional second and fourth order problems, and later extended to three dimensional problems by Farhat, Lesoinne, and Pierson [16].

In this method the original domain Ω is decomposed into several nonoverlapping subdomains Ω_i. The continuity of the primal solution is implemented directly into the formulation of the primal problem, so the subdomains leaving connected in the nodes called vertices or corners, see Figure 5.1. The continuity of the variables across the rest of the subdomains interface is enforced by the Lagrange multipliers.

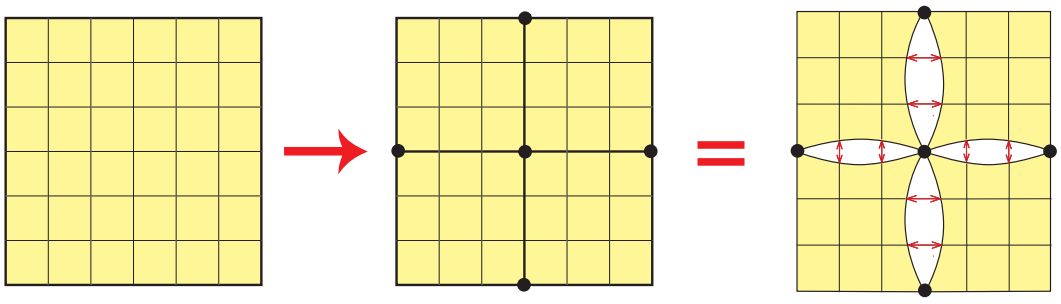


Figure 5.1: Domain Ω is decomposed into four subdomains, which are connected in the vertices (black nodes). To illustrate the Lagrange multipliers (red arrows) curved edges of the subdomains are used.

5.1 Notation

Let us start with a detailed description of the variables used in the FETI-DP method depicted in Figure 5.2. We use the notation as in [25, 29, 24]. *Primal displacement variables* or coarse problem nodes are those nodes, where the subdomains are connected. Since the FETI-DP method was introduced with the primal displacement

variables in the corners of the subdomains, these nodes are also called corners or vertices. In this work we prefer to use the notation primal displacement variables or coarse problem nodes, since we assumed also situations in which the primal displacement variables are not situated in the corners of the subdomains. *Dual displacement variables* are the nodes situated on the subdomains interface, where the continuity is enforced by *Lagrange multipliers* also called *dual variables*. *Interior displacement variables* are the nodes inside the subdomains.

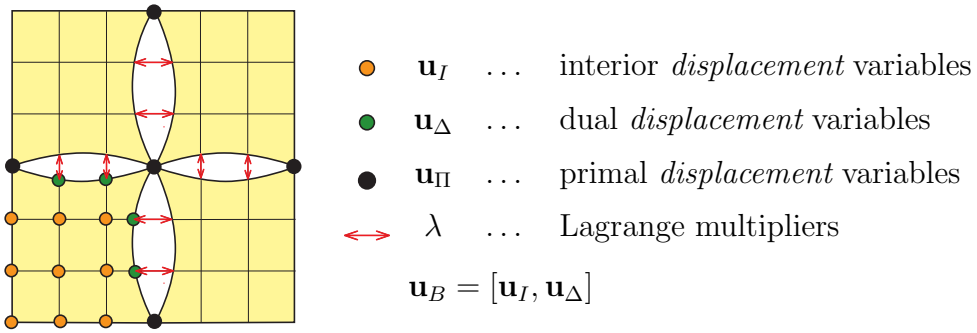


Figure 5.2: Types of variables used in FETI-DP method.

Using this notation we obtain the stiffness matrix K in the form

$$\mathsf{K} = \left[\begin{array}{cccccc|c} \mathsf{K}_{II}^{(1)} & \mathsf{K}_{\Delta I}^{(1)T} & & & & & \widetilde{\mathsf{K}}_{\Pi I}^{(1)T} \\ \mathsf{K}_{\Delta I}^{(1)} & \mathsf{K}_{\Delta\Delta}^{(1)} & & & & & \widetilde{\mathsf{K}}_{\Pi\Delta}^{(1)T} \\ & & \ddots & & & & \vdots \\ & & & \mathsf{K}_{II}^{(s)} & \mathsf{K}_{\Delta I}^{(s)T} & & \widetilde{\mathsf{K}}_{\Pi I}^{(s)T} \\ & & & \mathsf{K}_{\Delta I}^{(s)} & \mathsf{K}_{\Delta\Delta}^{(s)} & & \widetilde{\mathsf{K}}_{\Pi\Delta}^{(s)T} \\ \hline \widetilde{\mathsf{K}}_{\Pi I}^{(1)} & \widetilde{\mathsf{K}}_{\Pi\Delta}^{(1)} & \dots & \widetilde{\mathsf{K}}_{\Pi I}^{(s)} & \widetilde{\mathsf{K}}_{\Pi\Delta}^{(s)} & & \widetilde{\mathsf{K}}_{\Pi\Pi} \end{array}\right] = \left[\begin{array}{cc|c} \mathsf{K}_{II} & \mathsf{K}_{\Delta I}^T & \widetilde{\mathsf{K}}_{\Pi I}^T \\ \mathsf{K}_{\Delta I} & \mathsf{K}_{\Delta\Delta} & \widetilde{\mathsf{K}}_{\Pi\Delta}^T \\ \hline \widetilde{\mathsf{K}}_{\Pi I} & \widetilde{\mathsf{K}}_{\Pi\Delta} & \widetilde{\mathsf{K}}_{\Pi\Pi} \end{array}\right], \qquad (5.1)$$

where s denotes the number of subdomains. Using the notation

$$\mathsf{K}_{BB}^{(i)} = \begin{bmatrix} \mathsf{K}_{II}^{(i)} & \mathsf{K}_{\Delta I}^{(i)T} \\ \mathsf{K}_{\Delta I}^{(i)} & \mathsf{K}_{\Delta\Delta}^{(i)} \end{bmatrix} \quad \text{and} \quad \widetilde{\mathsf{K}}_{\Pi B}^{(i)} = \begin{bmatrix} \widetilde{\mathsf{K}}_{\Pi I}^{(i)} & \widetilde{\mathsf{K}}_{\Pi\Delta}^{(i)} \end{bmatrix},$$

we rewrite (5.1) as

$$\mathsf{K} = \begin{bmatrix} \mathsf{K}_{BB}^{(i)} & & & \widetilde{\mathsf{K}}_{\Pi B}^{(1)T} \\ & \ddots & & \\ & & \mathsf{K}_{BB}^{(s)} & \widetilde{\mathsf{K}}_{\Pi B}^{(s)T} \\ \widetilde{\mathsf{K}}_{\Pi B}^{(1)} & \dots & \widetilde{\mathsf{K}}_{\Pi B}^{(s)} & \widetilde{\mathsf{K}}_{\Pi\Pi} \end{bmatrix} = \begin{bmatrix} \mathsf{K}_{BB} & \widetilde{\mathsf{K}}_{\Pi B}^T \\ \widetilde{\mathsf{K}}_{\Pi B} & \widetilde{\mathsf{K}}_{\Pi\Pi} \end{bmatrix}.$$

The load vector $\mathbf{f}$ and the solution vector $\mathbf{u}$ of the nodal values can be written in a similar way

$$\mathbf{f} = \left[\begin{array}{c} \mathbf{f}_I \\ \hline \mathbf{f}_\Delta \\ \hline \mathbf{f}_\Pi \end{array}\right] = \begin{bmatrix} \mathbf{f}_B \\ \mathbf{f}_\Pi \end{bmatrix} \quad \text{and} \quad \mathbf{u} = \left[\begin{array}{c} \mathbf{u}_I \\ \hline \mathbf{u}_\Delta \\ \hline \mathbf{u}_\Pi \end{array}\right] = \begin{bmatrix} \mathbf{u}_B \\ \mathbf{u}_\Pi \end{bmatrix} = \begin{bmatrix} \mathbf{u}_B \\ \mathsf{L}\tilde{\mathbf{u}}_\Pi \end{bmatrix}. \qquad (5.2)$$

To simplify the implementation we introduce a global vector of degrees of freedom $\tilde{\mathbf{u}}_\Pi$ and an extending map L with one nonzero entry per line equal to 1, and we require that $\mathbf{u}_\Pi = \mathsf{L}\tilde{\mathbf{u}}_\Pi$.

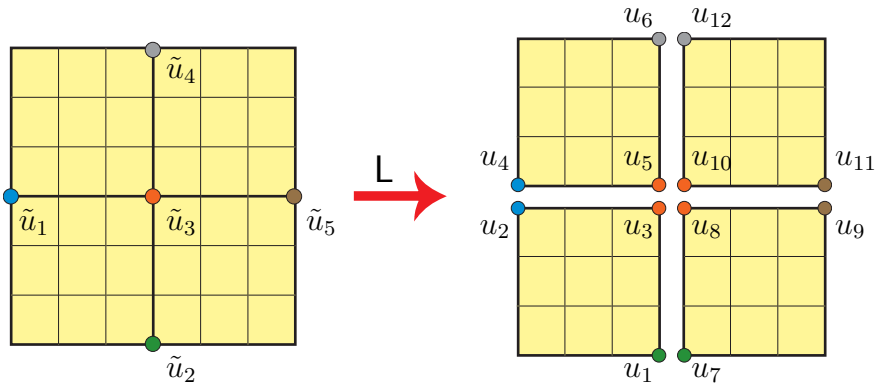


Figure 5.3: The extending map L is introduced to guarantee continuity at the primal displacement variables.

The matrix L for the problem depicted in Figure 5.3 has the form

$$\mathsf{L} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}. \tag{5.3}$$

Now it is easy to see that

$$\mathsf{L}\begin{bmatrix} \tilde{u}_1 \\ \tilde{u}_2 \\ \tilde{u}_3 \\ \tilde{u}_4 \\ \tilde{u}_5 \end{bmatrix} = \begin{bmatrix} \tilde{u}_2 \\ \tilde{u}_1 \\ \tilde{u}_3 \\ \tilde{u}_1 \\ \tilde{u}_3 \\ \tilde{u}_4 \\ \tilde{u}_2 \\ \tilde{u}_3 \\ \tilde{u}_5 \\ \tilde{u}_3 \\ \tilde{u}_5 \\ \tilde{u}_4 \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_8 \\ u_9 \\ u_{10} \\ u_{11} \\ u_{12} \end{bmatrix}. \tag{5.4}$$

In terms of the stiffness matrix we obtain

$$\widetilde{\mathsf{K}} = \begin{bmatrix} \mathsf{K}_{BB} & \widetilde{\mathsf{K}}_{\Pi B}^T \\ \widetilde{\mathsf{K}}_{\Pi B} & \widetilde{\mathsf{K}}_{\Pi\Pi} \end{bmatrix} = \begin{bmatrix} \mathsf{I}_B & \mathsf{O} \\ \mathsf{O} & \mathsf{L}^T \end{bmatrix} \begin{bmatrix} \mathsf{K}_{BB} & \mathsf{K}_{\Pi B}^T \\ \mathsf{K}_{\Pi B} & \mathsf{K}_{\Pi\Pi} \end{bmatrix} \begin{bmatrix} \mathsf{I}_B & \mathsf{O} \\ \mathsf{O} & \mathsf{L} \end{bmatrix} \tag{5.5}$$

where I_B is the identity matrix. $\widetilde{\mathsf{K}}$ is coupled in the primal variables but it still has a block structure in the K_{BB} block. The corresponding partially assembled righthand side is

$$\tilde{\mathbf{f}} = \begin{bmatrix} \mathbf{f}_B \\ \tilde{\mathbf{f}}_\Pi \end{bmatrix} = \begin{bmatrix} \mathsf{I}_B & \mathsf{O} \\ \mathsf{O} & \mathsf{L}^T \end{bmatrix} \begin{bmatrix} \mathbf{f}_B \\ \mathbf{f}_\Pi \end{bmatrix}. \tag{5.6}$$

Now, we can consider the primal problem, equivalent to (2.7),

$$\min_{\mathbf{u}\in\Upsilon_E} \phi(\mathbf{u}), \quad \Upsilon_E = \{\mathbf{u} \in \mathbb{R}^n : \mathsf{B}_\mathcal{E}\mathbf{u} = \mathbf{o}\}, \tag{5.7}$$

where $\mathsf{B}_\mathcal{E}\mathbf{u} = \mathbf{o}$ are "gluing" conditions arising from the domain decomposition.

5.2 Lagrangian function

Let us now explain the derivation of the dual problem. The Lagrangian function associated with (5.7) if of the form

$$L_0(\mathbf{u}, \lambda) = \frac{1}{2}\mathbf{u}^T\widetilde{\mathsf{K}}\mathbf{u} - \mathbf{u}^T\tilde{\mathbf{f}} + \mathbf{u}^T\mathsf{B}_\mathcal{E}^T\lambda, \tag{5.8}$$

where λ is a vector of the Lagrange multipliers. The corresponding saddle point system then can be written as

$$\begin{bmatrix} \widetilde{\mathsf{K}} & \mathsf{B}_\mathcal{E}^T \\ \mathsf{B}_\mathcal{E} & \mathsf{O} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \lambda \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{f}} \\ \mathbf{o} \end{bmatrix}. \tag{5.9}$$

Using the notation

$$\mathsf{B}_\mathcal{E} = \begin{bmatrix} \mathsf{B}_B & \mathsf{O} \end{bmatrix} \quad \Big(= \begin{bmatrix} \mathsf{B}_B & \mathsf{B}_\Pi \end{bmatrix} \Big), \tag{5.10}$$

we rewrite the saddle point problem (5.9) as

$$\begin{bmatrix} \mathsf{K}_{BB} & \mathsf{K}_{\Pi B}^T\mathsf{L} & \mathsf{B}_B^T \\ \mathsf{L}^T\mathsf{K}_{\Pi B} & \mathsf{L}^T\mathsf{K}_{\Pi\Pi}\mathsf{L} & \mathsf{O} \\ \mathsf{B}_B & \mathsf{O} & \mathsf{O} \end{bmatrix} \begin{bmatrix} \mathbf{u}_B \\ \tilde{\mathbf{u}}_\Pi \\ \lambda \end{bmatrix} = \begin{bmatrix} \mathbf{f}_B \\ \mathsf{L}^T\mathbf{f}_\Pi \\ \mathbf{o} \end{bmatrix}, \tag{5.11}$$

where B_B is the so called jump operator enforcing the continuity at the dual displacement variables, which is constructed using values $\{-1, 0, 1\}$ in such a way, that the values of the solution $\mathbf{u}_B$ associated with more than one subdomain coincide when

$\mathsf{B}_B\mathbf{u}_B = 0$; the interior variables $\mathbf{u}_I$ remain unchanged and thus the corresponding entries in B_B are zero [25]. We can rewrite also (5.8) as

$$\begin{aligned} L_0(\mathbf{u}_B, \tilde{\mathbf{u}}_\Pi, \lambda) &= \frac{1}{2}\mathbf{u}_B^T\mathsf{K}_{BB}\mathbf{u}_B + \mathbf{u}_B^T\mathsf{K}_{\Pi B}^T\mathsf{L}\tilde{\mathbf{u}}_\Pi + \frac{1}{2}\tilde{\mathbf{u}}_\Pi^T\mathsf{L}^T\mathsf{K}_{\Pi\Pi}\mathsf{L}\tilde{\mathbf{u}}_\Pi - \mathbf{f}_B^T\mathbf{u}_B \\ &- \mathbf{f}_\Pi^T\mathsf{L}\tilde{\mathbf{u}}_\Pi + \mathbf{u}_B^T\mathsf{B}_B^T\lambda \\ &= \frac{1}{2}\mathbf{u}_B^T\mathsf{K}_{BB}\mathbf{u}_B - \mathbf{u}_B^T(\mathbf{f}_B - \mathsf{K}_{\Pi B}^T\mathsf{L}\tilde{\mathbf{u}}_\Pi - \mathsf{B}_B^T\lambda) \\ &+ \frac{1}{2}\tilde{\mathbf{u}}_\Pi^T\mathsf{L}^T\mathsf{K}_{\Pi\Pi}\mathsf{L}\tilde{\mathbf{u}}_\Pi - \mathbf{f}_\Pi^T\mathsf{L}\tilde{\mathbf{u}}_\Pi. \end{aligned} \tag{5.12}$$

To minimize $L_0(\mathbf{u}_B, \tilde{\mathbf{u}}_\Pi, \lambda)$ over $\mathbf{u}_B$, we consider

$$\frac{\partial L_0}{\partial \mathbf{u}_B} = \mathsf{K}_{BB}\mathbf{u}_B - \Big(\mathbf{f}_B - \mathsf{K}_{\Pi B}^T\mathsf{L}\tilde{\mathbf{u}}_\Pi - \mathsf{B}_B^T\lambda\Big) = 0$$

which implies

$$\mathbf{u}_B = \mathsf{K}_{BB}^{-1}\Big(\mathbf{f}_B - \mathsf{K}_{\Pi B}^T\mathsf{L}\tilde{\mathbf{u}}_\Pi - \mathsf{B}_B^T\lambda\Big).$$

By substituting this result into (5.12), we obtain

$$\begin{aligned} L_0(\tilde{\mathbf{u}}_\Pi, \lambda) &= \frac{1}{2}\tilde{\mathbf{u}}_\Pi^T\mathsf{L}^T\Big(\mathsf{K}_{\Pi\Pi} - \mathsf{K}_{\Pi B}\mathsf{K}_{BB}^{-1}\mathsf{K}_{\Pi B}^T\Big)\mathsf{L}\tilde{\mathbf{u}}_\Pi \\ &- \tilde{\mathbf{u}}_\Pi^T\mathsf{L}^T\Big(\mathbf{f}_\Pi - \mathsf{K}_{\Pi B}\mathsf{K}_{BB}^{-1}\mathbf{f}_B + \mathsf{K}_{\Pi B}\mathsf{K}_{BB}^{-1}\mathsf{B}_B^T\lambda\Big) \\ &- \frac{1}{2}(\mathbf{f}_B - \mathsf{B}_B^T\lambda)^T\mathsf{K}_{BB}^{-1}(\mathbf{f}_B - \mathsf{B}_B^T\lambda) \\ &= \frac{1}{2}\tilde{\mathbf{u}}_\Pi^T\tilde{\mathsf{S}}_{\Pi\Pi}\tilde{\mathbf{u}}_\Pi - \tilde{\mathbf{u}}_\Pi^T(\widehat{\mathbf{f}}_\Pi - \widehat{\mathsf{K}}_{\Pi B}\lambda) \\ &- \frac{1}{2}(\mathbf{f}_B - \mathsf{B}_B^T\lambda)^T\mathsf{K}_{BB}^{-1}(\mathbf{f}_B - \mathsf{B}_B^T\lambda), \end{aligned} \tag{5.13}$$

where

$$\begin{aligned} \tilde{\mathsf{S}}_{\Pi\Pi} &= \mathsf{L}^T\Big(\mathsf{K}_{\Pi\Pi} - \mathsf{K}_{\Pi B}\mathsf{K}_{BB}^{-1}\mathsf{K}_{\Pi B}^T\Big)\mathsf{L}, \\ \widehat{\mathbf{f}}_\Pi &= \mathsf{L}^T\Big(\mathbf{f}_\Pi - \mathsf{K}_{\Pi B}\mathsf{K}_{BB}^{-1}\mathbf{f}_B\Big), \\ \widehat{\mathsf{K}}_{\Pi B}^T &= -\mathsf{B}_B\mathsf{K}_{BB}^{-1}\mathsf{K}_{\Pi B}^T\mathsf{L}. \end{aligned} \tag{5.14}$$

In order to minimize (5.13) over $\tilde{\mathbf{u}}_\Pi$, we consider

$$\frac{\partial L_0}{\partial \tilde{\mathbf{u}}_\Pi} = \tilde{\mathsf{S}}_{\Pi\Pi}\tilde{\mathbf{u}}_\Pi - \Big(\widehat{\mathbf{f}}_\Pi - \widehat{\mathsf{K}}_{\Pi B}\lambda\Big) = 0$$

and obtain

$$\tilde{\mathbf{u}}_\Pi = \tilde{\mathsf{S}}_{\Pi\Pi}^{-1}\Big(\widehat{\mathbf{f}}_\Pi - \widehat{\mathsf{K}}_{\Pi B}\lambda\Big), \quad \mathbf{u}_\Pi = \mathsf{L}\tilde{\mathbf{u}}_\Pi.$$

Using this result, we rewrite (5.13) in the form

$$\begin{aligned} L_0(\lambda) &= \frac{1}{2}\lambda^T\Big(\widehat{\mathsf{K}}_{BB} - \widehat{\mathsf{K}}_{\Pi B}^T\tilde{\mathsf{S}}_{\Pi\Pi}^{-1}\widehat{\mathsf{K}}_{\Pi B}\Big)\lambda - \lambda^T\Big(\widehat{\mathbf{f}}_B - \widehat{\mathsf{K}}_{\Pi B}^T\tilde{\mathsf{S}}_{\Pi\Pi}^{-1}\widehat{\mathbf{f}}_\Pi\Big) \\ &- \frac{1}{2}\mathbf{f}_B^T\mathsf{K}_{BB}^{-1}\mathbf{f}_B - \frac{1}{2}\widehat{\mathbf{f}}_\Pi^T\tilde{\mathsf{S}}_{\Pi\Pi}^{-1}\widehat{\mathbf{f}}_\Pi, \end{aligned} \tag{5.15}$$

where $\widehat{\mathsf{K}}_{BB} = -\mathsf{B}_B\mathsf{K}_{BB}^{-1}\mathsf{B}_B^T$ and $\widehat{\mathbf{f}}_B = -\mathsf{B}_B\mathsf{K}_{BB}^{-1}\mathbf{f}_B$.

Finnally, using the notation of theory of duality, we can rewrite (5.15) as the dual problem

$$\min \Theta(\lambda), \quad \Theta(\lambda) = \frac{1}{2}\lambda^T\mathsf{F}\lambda - \lambda^T\mathbf{d}, \tag{5.16}$$

where $\mathsf{F} = \widehat{\mathsf{K}}_{BB} - \widehat{\mathsf{K}}_{\Pi B}^T\tilde{\mathsf{S}}_{\Pi\Pi}^{-1}\widehat{\mathsf{K}}_{\Pi B}$ and $\mathbf{d} = \widehat{\mathbf{f}}_B - \widehat{\mathsf{K}}_{\Pi B}^T\tilde{\mathsf{S}}_{\Pi\Pi}^{-1}\widehat{\mathbf{f}}_\Pi$. Minimizing $\Theta(\lambda)$ is equivalent to solving problem (2.7).

Remarks

- Note that the set of primal variables Π can be formed by some "optional" nodes from the interface, not necessarily by vertices as was assumed in this section. The effort is to ensure a small size of the dual problem, on the other hand we have to choose enough primal variables in order to control the rigid body motions.

5.3 Projector preconditioning for FETI-DP method

The combination of the projector preconditioning with the FETI-DP method was introduced by Jarošová, Klawonn, Rheinbach [21].

Let $\mathsf{F} \in \mathbb{R}^{m\times m}$ be a symmetric positive definite matrix. A projector P is an F-*conjugate projector* or briefly a *conjugate projector* if $\mathrm{Im}\mathsf{P}$ is F-conjugate to $\mathrm{Ker}\mathsf{P}$, or equivalently

$$\mathsf{P}^T\mathsf{F}(\mathsf{I}-\mathsf{P}) = \mathsf{P}^T\mathsf{F} - \mathsf{P}^T\mathsf{F}\mathsf{P} = \mathsf{O}.$$

If $\mathcal{U}$ is the subspace spanned by the columns of a full column rank matrix $\mathsf{U} \in \mathbb{R}^{m\times p}$, then

$$\mathsf{P} = \mathsf{U}(\mathsf{U}^T\mathsf{F}\mathsf{U})^{-1}\mathsf{U}^T\mathsf{F} \tag{5.17}$$

is a conjugate projector onto $\mathcal{U}$. We use the conjugate projectors P and $\mathsf{Q} = \mathsf{I}-\mathsf{P}$ to decompose our dual minimization problem (5.16) into the minimization on $\mathcal{U}$ and the minimization on $\mathcal{V} = \mathrm{Im}\mathsf{Q}$, we can write

$$\begin{aligned}
\min\Theta(\lambda) &= \min_{\eta\in\mathcal{U},\mu\in\mathcal{V}}\Theta(\eta+\mu) = \min_{\eta\in\mathcal{U}}\Theta(\eta) + \min_{\mu\in\mathcal{V}}\Theta(\mu)\\
&= \Theta(\lambda^0) + \min_{\mu\in\mathcal{V}}\Theta(\mu) = \Theta(\lambda^0) + \min_{\mu\in\mathsf{F}\mathcal{V}}\frac{1}{2}\mu^T\mathsf{Q}^T\mathsf{F}\mathsf{Q}\mu - \mathbf{d}^T\mathsf{Q}\mu\\
&= \Theta(\lambda^0) + \min_{\mu\in\mathsf{F}\mathcal{V}}\frac{1}{2}\mu^T\mathsf{Q}^T\mathsf{F}\mathsf{Q}\mu + \mu^T\mathbf{g}^0,
\end{aligned}$$

where $\lambda^0 = \mathsf{P}\mathsf{F}^{-1}\mathbf{d}$ and $\mathbf{g}^0 = -\mathsf{Q}^T\mathbf{d}$. The solution $\widehat{\lambda}$ of the dual problem (5.16) can then be expressed by $\widehat{\lambda} = \lambda^0 + \mathsf{Q}\widehat{\mu}$, where $\widehat{\mu}$ is the solution on $\mathsf{F}\mathcal{V}$.

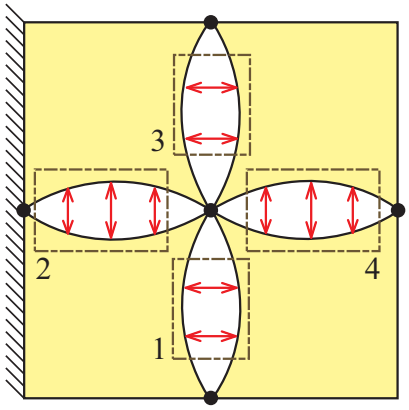


Figure 5.4: Aggregated Lagrange multipliers.

In this approach the matrix U is defined by the elements of the aggregation bases such as those depicted in Figure 5.4. The aggregated variables are the Lagrange multipliers that enforce the continuity conditions of the primal displacement variables of two adjoining subdomains. The matrix U for the problem depicted in Figure 5.4 is of the form

$$\mathsf{U} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{5.18}$$

5.4 Dirichlet preconditioner

In the standard CG accelerated FETI-DP methods we often use the theoretically almost optimal Dirichlet preconditioner M_D is often used [15, 17]. Let us first define the multiplicity of a node as the number of subdomains it belongs to. Now we can define the scaled operator $\mathsf{B}_{B,D}$. The matrix $\mathsf{B}_{B,D}$ is a scaled variant of the operator B_B, where the contribution from and to each interface node is scaled by the inverse of the multiplicity of the node. This scaling is therefore refered to in the literature as multiplicity scaling. In the simplest 2D case where all vertex unknows are primal, we have $\mathsf{B}_{B,D} = \mathsf{D}\mathsf{B}_B$ with $\mathsf{D} = \frac{1}{2}\mathsf{I}$.

To define the preconditioner we also need a restriction matrix $[\mathsf{O}\ \mathsf{I}_\Delta]$ which restricts the nonprimal variables $\mathbf{u}_B$ to the dual part $\mathbf{u}_\Delta$, i.e., it is zero on $\mathbf{u}_I$ and the identity on $\mathbf{u}_\Delta$. Then the Dirichlet preconditioner is the Schur complement

$$\mathsf{M}_D^{-1} = \mathsf{B}_{B,D}[\mathsf{O}\ \mathsf{I}_\Delta]^T(\mathsf{K}_{\Delta\Delta} - \mathsf{K}_{\Delta I}(\mathsf{K}_{II})^{-1}\mathsf{K}_{\Delta I}^T)[\mathsf{O}\ \mathsf{I}_\Delta]\mathsf{B}_{B,D}^T.$$

For heterogeneous problems a different scaling is necessary; see, e.g., [29]. A comparison of different scalings for ragged subdomain interfaces has been considered by Klawonn, Rheinbach, and Widlund in [27].

Transformation of basis 6

In this chapter we describe the method exploiting the transformation of basis to replace or enhance the coarse problem of the dual-primal FETI method.

This method uses certain edge or face averages which are introduced either in addition to or instead of the assembly in a selected number of primal variables; see, e.g., Farhat, Lesoinne, Pierson [16], Klawonn and Widlund [29], Klawonn, Widlund, and Dryja [30], and Klawonn and Rheinbach [25].

Let us consider the transformation of basis described by Klawonn and Widlund [29], Klawonn and Rheinbach [25], and Li and Widlund [31]. In this approach, the averages are introduced as new primal variables into the FETI-DP system and then subassembled in these degrees of freedom as shown in Figure 6.1. Our approach uses an explicitly the change of basis. As a result, the finite element functions associated with dual displacement vectors will have zero edge averages over the coinciding edges.

6.1 Change of variables

Let us now describe the main idea of the transformation of basis on the situation depicted in Figure 6.1 *(left)*. Obviously, in the solution the equality conditions

$$\begin{aligned} u_1 &= u_4 \\ u_2 &= u_5 \\ u_3 &= u_6 \end{aligned} \tag{6.1}$$

have to be satisfied. Let us now replace an arbitrary equation, for example, the last one, in the form of "average" as

$$u_1 + u_2 + u_3 = u_4 + u_5 + u_6. \tag{6.2}$$

We consider the equality conditions in the form

$$\begin{aligned} u_1 &= u_4 \\ u_2 &= u_5 \\ u_1 + u_2 + u_3 &= u_4 + u_5 + u_6. \end{aligned} \tag{6.3}$$

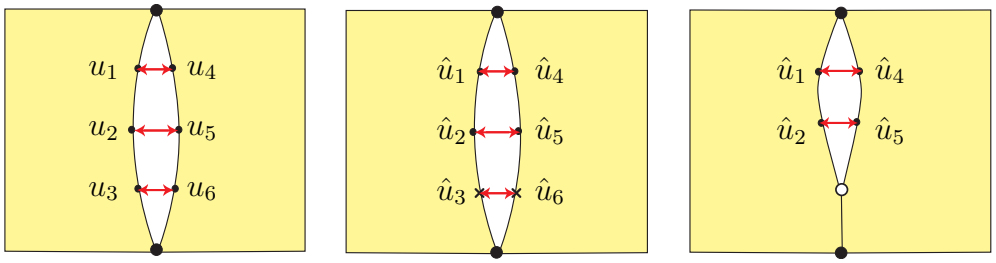


Figure 6.1: Vertex constraint, transformation of basis, and assembly of averages.

It is easy to see that also the conditions (6.3) have to be satisfied in the solution and are equivalent to (6.1). Now, we introduce new variables $\hat{\mathbf{u}}$, see Figure 6.1 *(middle)*, and let us rewrite (6.3) as

$$\begin{aligned} \hat{u}_1 &= \hat{u}_4 \\ \hat{u}_2 &= \hat{u}_5 \\ \hat{u}_3 &= \hat{u}_6. \end{aligned} \tag{6.4}$$

The change of the edge variables from $\mathbf{u}$ to $\hat{\mathbf{u}}$ and back can be described for the first subdomain in the form

$$\begin{bmatrix} \hat{u}_1 \\ \hat{u}_2 \\ \hat{u}_3 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix}}_{\mathsf{T}^{-1}} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -1 & -1 & 1 \end{bmatrix}}_{\mathsf{T}} \begin{bmatrix} \hat{u}_1 \\ \hat{u}_2 \\ \hat{u}_3 \end{bmatrix}, \tag{6.5}$$

where the transformation matrix T describes the change of variables from the new basis to the original one. To extend the information from the whole edge efficiently over the interface, we replace the Lagrangian multiplier between the averages by the new primal variable. This situation is depicted in Figure 6.1 *(right)*. Below, we introduce a different form of the transformation matrix T, which turns out to be more suitable for our research, see Remarks in Chapter 11.

6.2 Two subdomains

Let us consider the model problem introduced in Chapter 2. Let us decompose the domain Ω into two subdomains, Ω_1 and Ω_2, and denote by Γ their interface, as shown in Figure 6.2. In this the case of two subdomains, no vertices are introduced instead of vertex constraints, the interface average is selected as the sole primal variable.

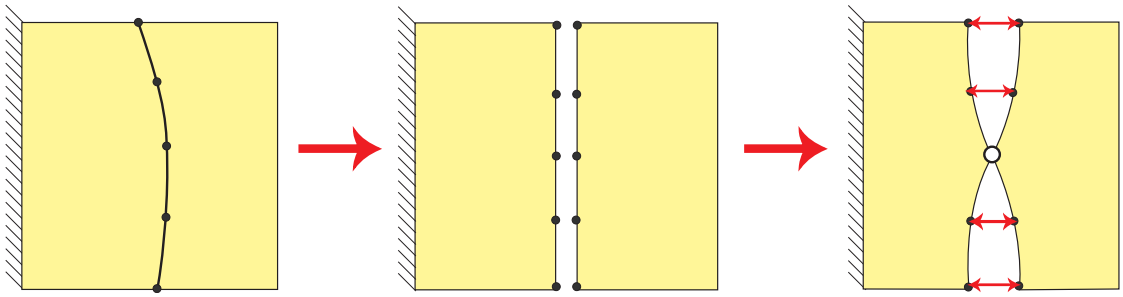


Figure 6.2: Partition into two subdomains in the absence of a vertex constraint.

First let us show, as Li and Widlund [31], how to change the variables to make the edge average degree of freedom explicit. For each subdomain Ω_i, we denote the variables corresponding to the nodes on the interface Γ by $(u_1^{(i)}, \dots, u_m^{(i)}, \dots, u_l^{(i)})$, where the node m can be any node on the edge. The rest of the unknowns will be denoted by $\mathbf{u}_I^{(i)}$.

A linear system for subdomain Ω_i can be written as

$$\mathsf{K}^{(i)}\mathbf{u}^{(i)} = \begin{bmatrix} \mathsf{K}_{II}^{(i)} & \mathsf{K}_{1I}^{(i)T} & \cdots & \mathsf{K}_{mI}^{(i)T} & \cdots & \mathsf{K}_{lI}^{(i)T} \\ \mathsf{K}_{1I}^{(i)} & k_{11}^{(i)} & \cdots & k_{1m}^{(i)} & \cdots & k_{1l}^{(i)} \\ \vdots & \vdots & \ddots & \vdots & & \vdots \\ \mathsf{K}_{mI}^{(i)} & k_{m1}^{(i)} & \cdots & k_{mm}^{(i)} & \cdots & k_{ml}^{(i)} \\ \vdots & \vdots & & \vdots & \ddots & \vdots \\ \mathsf{K}_{lI}^{(i)} & k_{l1}^{(i)} & \cdots & k_{lm}^{(i)} & \cdots & k_{ll}^{(i)} \end{bmatrix} \begin{bmatrix} \mathbf{u}_I^{(i)} \\ u_1^{(i)} \\ \vdots \\ u_m^{(i)} \\ \vdots \\ u_l^{(i)} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_I^{(i)} \\ f_1^{(i)} \\ \vdots \\ f_m^{(i)} \\ \vdots \\ f_l^{(i)} \end{bmatrix}.$$

Let us now consider $l \times l$ matrix T_E in the form

$$\mathsf{T}_E = \begin{bmatrix} 1 & & 1 & & \\ & \ddots & \vdots & & \\ -1 & \cdots & 1 & \cdots & -1 \\ & & \vdots & \ddots & \\ & & 1 & & 1 \end{bmatrix}$$

with nonzero elements on the main diagonal, in the m-th row, and in the m-th column. The matrix T_E is the transformation matrix with columns representing the new basis of the space of edge variables.

Using this transformation matrix T_E, the interface variables of both subdomains can be changed. We denote the interface variables in the new basis by $(\hat{u}_1^{(i)}, \dots, \hat{u}_m^{(i)}, \dots, \hat{u}_l^{(i)})$, so we write

$$\begin{bmatrix} u_1^{(i)} \\ \vdots \\ u_m^{(i)} \\ \vdots \\ u_l^{(i)} \end{bmatrix} = \mathsf{T}_E \begin{bmatrix} \hat{u}_1^{(i)} \\ \vdots \\ \hat{u}_m^{(i)} \\ \vdots \\ \hat{u}_l^{(i)} \end{bmatrix} = \begin{bmatrix} 1 & & 1 & & \\ & \ddots & \vdots & & \\ -1 & \cdots & 1 & \cdots & -1 \\ & & \vdots & \ddots & \\ & & 1 & & 1 \end{bmatrix} \begin{bmatrix} \hat{u}_1^{(i)} \\ \vdots \\ \hat{u}_m^{(i)} \\ \vdots \\ \hat{u}_l^{(i)} \end{bmatrix}.$$

The original interface variables are now separated into two parts

$$\begin{bmatrix} u_1^{(i)} \\ \vdots \\ u_m^{(i)} \\ \vdots \\ u_l^{(i)} \end{bmatrix} = \begin{bmatrix} 1 \\ \vdots \\ 1 \\ \vdots \\ 1 \end{bmatrix} \hat{u}_m^{(i)} + \begin{bmatrix} \hat{u}_1^{(i)} \\ \vdots \\ -\hat{u}_1^{(i)} - \cdots - \hat{u}_{m-1}^{(i)} - \hat{u}_{m+1}^{(i)} - \cdots - \hat{u}_l^{(i)} \\ \vdots \\ \hat{u}_l^{(i)} \end{bmatrix},$$

the first part corresponds to the basis function which is constant on the edge and has the value $\hat{u}_m^{(i)}$ for the subdomain Ω_i; while the second corresponds to the functions with zero edge average. Figure 6.3 depicts the nodal basis in 1D for the edge nodes from Figure 6.2. The basis includes the average in analogy to [38].

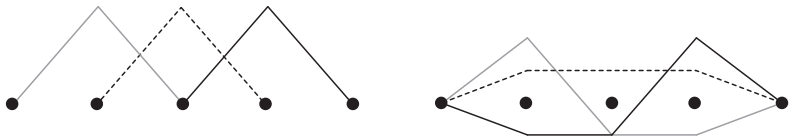


Figure 6.3: (*left*): Nodal basis consisting of 3 nodal basis functions. (*right*): Basis consisting of an average basis function (corresponding to the middle point) and a 2 nodal basis functions.

Let us now introduce the matrix $\mathsf{T}^{(i)}$, the transformation matrix for all variables of one subdomain. Since the transformation affects only the interface variables, the matrix $\mathsf{T}^{(i)}$ is a block diagonal of the form

$$\mathsf{T}^{(i)} = \begin{bmatrix} \mathsf{I}_I & \\ & \mathsf{T}_E \end{bmatrix},$$

where I_I is an identity matrix on the positions of the interior variables. The transformed subdomain problem can now be written as

$$\mathsf{T}^{(i)T} \begin{bmatrix} \mathsf{K}_{II}^{(i)} & \mathsf{K}_{1I}^{(i)T} & \cdots & \mathsf{K}_{mI}^{(i)T} & \cdots & \mathsf{K}_{lI}^{(i)T} \\ \mathsf{K}_{1I}^{(i)} & k_{11}^{(i)} & \cdots & k_{1m}^{(i)} & \cdots & k_{1l}^{(i)} \\ \vdots & \vdots & \ddots & \vdots & & \vdots \\ \mathsf{K}_{mI}^{(i)} & k_{m1}^{(i)} & \cdots & k_{mm}^{(i)} & \cdots & k_{ml}^{(i)} \\ \vdots & \vdots & & \vdots & \ddots & \vdots \\ \mathsf{K}_{lI}^{(i)} & k_{l1}^{(i)} & \cdots & k_{lm}^{(i)} & \cdots & k_{ll}^{(i)} \end{bmatrix} \mathsf{T}^{(i)} \begin{bmatrix} \mathbf{u}_I^{(i)} \\ \hat{u}_1^{(i)} \\ \vdots \\ \hat{u}_m^{(i)} \\ \vdots \\ \hat{u}_l^{(i)} \end{bmatrix} = \mathsf{T}^{(i)T} \begin{bmatrix} \mathbf{f}_I^{(i)} \\ f_1^{(i)} \\ \vdots \\ f_m^{(i)} \\ \vdots \\ f_l^{(i)} \end{bmatrix}.$$

The subdomain edge average variables, $\hat{u}_m^{(1)}$ and $\hat{u}_m^{(2)}$, are required to have a common value throughout the FETI-DP iteration. Let us denote this common variables by $\hat{\mathbf{u}}_\Pi$, as a primal variable. The other interface variables, the dual displacement variables, are denoted by $\hat{\mathbf{u}}_\Delta^{(1)}$ and $\hat{\mathbf{u}}_\Delta^{(2)}$, with their own values at the same interface nodes. The global system of the example can then be written as

$$\begin{bmatrix} \bar{\mathsf{K}}_{BB}^{(1)} & & \bar{\mathsf{K}}_{\Pi B}^{(1)T} & \mathsf{B}_B^{(1)T} \\ & \bar{\mathsf{K}}_{BB}^{(2)} & \bar{\mathsf{K}}_{\Pi B}^{(2)T} & \mathsf{B}_B^{(2)T} \\ \bar{\mathsf{K}}_{\Pi B}^{(1)} & \bar{\mathsf{K}}_{\Pi B}^{(2)} & \bar{\mathsf{K}}_{\Pi\Pi}^{(1)} + \bar{\mathsf{K}}_{\Pi\Pi}^{(2)} & \\ \mathsf{B}_B^{(1)} & \mathsf{B}_B^{(2)} & & \end{bmatrix} \begin{bmatrix} \hat{\mathbf{u}}_B^{(1)} \\ \hat{\mathbf{u}}_B^{(2)} \\ \hat{\mathbf{u}}_\Pi \\ \lambda \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{f}}_B^{(1)} \\ \hat{\mathbf{f}}_B^{(2)} \\ \hat{\mathbf{f}}_\Pi^{(1)} + \hat{\mathbf{f}}_\Pi^{(2)} \\ \mathbf{o} \end{bmatrix}, \tag{6.6}$$

where $\hat{\mathbf{u}}_B^{(i)} = [\hat{\mathbf{u}}_I^{(i)T}, \hat{\mathbf{u}}_\Delta^{(i)T}]$. System (6.6) is of the same form as system (5.11) and can be solved in the same way, see Chapter 5. To obtain the primal solution we need to use the backward transformation in the form of $\mathbf{u}_E = \mathsf{T}_E \hat{\mathbf{u}}_E$.

To see that $\hat{u}_m^{(i)}$ represents the edge average indeed, it is enought to express this variable using the relation $\hat{\mathbf{u}}_E = \mathsf{T}_E^{-1}\mathbf{u}_E$. To save the simplicity, let us consider only 3 interface variables, so we have

$$\begin{bmatrix} \hat{\mathbf{u}}_1 \\ \hat{\mathbf{u}}_2 \\ \hat{\mathbf{u}}_3 \end{bmatrix} = \frac{1}{3} \begin{bmatrix} 2 & -1 & -1 \\ 1 & 1 & 1 \\ -1 & -1 & 2 \end{bmatrix} \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \mathbf{u}_3 \end{bmatrix}.$$

It is easy to see that, in this case, the edge average is represented by

$$\hat{\mathbf{u}}_3 = \frac{1}{3}(\mathbf{u}_1 + \mathbf{u}_2 + \mathbf{u}_3).$$

We note that the average can be placed to any interface position.

6.3 Many subdomains

Let us again consider the model problem (2.1). Now, the problem will be decomposed into many subdomains. Let $\hat{\mathbf{u}}_E$ denote the edge unknowns in the new basis, then

$$\mathbf{u}_E = \mathsf{T}_E \hat{\mathbf{u}}_E, \tag{6.7}$$

where T_E is the transformation matrix with columns representing the new basis. This matrix performs the desired change of basis from the new basis to the original nodal basis.

Ordering the average last, T_E can be written as

$$\mathsf{T}_E = \begin{bmatrix} 1 & \dots & 0 & 1 \\ & \ddots & & \vdots \\ 0 & & 1 & 1 \\ -1 & \dots & -1 & 1 \end{bmatrix}. \tag{6.8}$$

Such transformation matrix can be constructed separately for each edge. The resulting transformation matrix $\mathsf{T}_E^{(i)}$, which operates on all relevant edges of Ω_i, is a direct sum of the relevant transformation matrices T_E associated with the edges of the subdomain Ω_i. $\mathsf{T}_E^{(i)}$ is a block diagonal, where each block represents the transformation of the variables of one edge.

Since in this case we also assume the vertex constraints, the transformation matrix for all variables of one subdomain Ω_i is of the form

$$\mathsf{T}^{(i)} = \begin{bmatrix} \mathsf{I}_I^{(i)} & \mathsf{O} & \mathsf{O} \\ \mathsf{O} & \mathsf{I}_V^{(i)} & \mathsf{O} \\ \mathsf{O} & \mathsf{O} & \mathsf{T}_E^{(i)} \end{bmatrix}, \tag{6.9}$$

where the subscripts I, V, E denote interior, vertex, and edge nodes, respectively. $\mathsf{I}_I^{(i)}$ and $\mathsf{I}_V^{(i)}$ denote identity matrices.

The transformed subdomain problem is of the form

$$\mathsf{T}^{(i)T}\mathsf{K}^{(i)}\mathsf{T}^{(i)}\hat{\mathbf{u}} = \mathsf{T}^{(i)T}\mathbf{f}. \tag{6.10}$$

Using the same decomposition as in (6.9) also for the local stiffness matrix $\mathsf{K}^{(i)}$, we have

$$\mathsf{K}^{(i)} = \left[\begin{array}{cc|c} \mathsf{K}_{II}^{(i)} & \mathsf{K}_{VI}^{(i)T} & \mathsf{K}_{EI}^{(i)T} \\ \mathsf{K}_{VI}^{(i)} & \mathsf{K}_{VV}^{(i)} & \mathsf{K}_{EV}^{(i)T} \\ \hline \mathsf{K}_{EI}^{(i)} & \mathsf{K}_{EV}^{(i)} & \mathsf{K}_{EE}^{(i)} \end{array}\right].$$

The matrix of the transformed system (6.10) than has the form

$$\begin{aligned} \mathsf{T}^{(i)T}\mathsf{K}^{(i)}\mathsf{T}^{(i)} &= \left[\begin{array}{cc|c} \mathsf{K}_{II}^{(i)} & \mathsf{K}_{VI}^{(i)T} & \mathsf{K}_{EI}^{(i)T}\mathsf{T}_E^{(i)} \\ \mathsf{K}_{VI}^{(i)} & \mathsf{K}_{VV}^{(i)} & \mathsf{K}_{EV}^{(i)T}\mathsf{T}_E^{(i)} \\ \hline \mathsf{T}_E^{(i)T}\mathsf{K}_{EI}^{(i)} & \mathsf{T}_E^{(i)T}\mathsf{K}_{EV}^{(i)} & \mathsf{T}_E^{(i)T}\mathsf{K}_{EE}^{(i)}\mathsf{T}_E^{(i)} \end{array}\right] \\ &= \left[\begin{array}{cc|c} \mathsf{K}_{II}^{(i)} & \mathsf{K}_{VI}^{(i)T} & \bar{\mathsf{K}}_{EI}^{(i)T} \\ \mathsf{K}_{VI}^{(i)} & \mathsf{K}_{VV}^{(i)} & \bar{\mathsf{K}}_{EV}^{(i)T} \\ \hline \bar{\mathsf{K}}_{EI}^{(i)} & \bar{\mathsf{K}}_{EV}^{(i)} & \bar{\mathsf{K}}_{EE}^{(i)} \end{array}\right]. \end{aligned} \tag{6.11}$$

The edge variables are now split into two parts: the dual variables and the averages, so that $\hat{\mathbf{u}}_E = [\hat{\mathbf{u}}_\Delta, \hat{\mathbf{u}}_A]$. Using this notation, we rewrite the matrix $\bar{\mathsf{K}}_{EE}^{(i)}$ in the form

$$\bar{\mathsf{K}}_{EE}^{(i)} = \begin{bmatrix} \bar{\mathsf{K}}_{\Delta\Delta}^{(i)} & \bar{\mathsf{K}}_{A\Delta}^{(i)T} \\ \bar{\mathsf{K}}_{A\Delta}^{(i)} & \bar{\mathsf{K}}_{AA}^{(i)} \end{bmatrix}.$$

Putting it into the previous matrix, we obtain

$$\mathsf{T}^{(i)T}\mathsf{K}^{(i)}\mathsf{T}^{(i)} = \left[\begin{array}{cc|cc} \mathsf{K}_{II}^{(i)} & \mathsf{K}_{VI}^{(i)T} & \bar{\mathsf{K}}_{\Delta I}^{(i)T} & \bar{\mathsf{K}}_{AI}^{(i)T} \\ \mathsf{K}_{VI}^{(i)} & \mathsf{K}_{VV}^{(i)} & \bar{\mathsf{K}}_{\Delta V}^{(i)T} & \bar{\mathsf{K}}_{AV}^{(i)T} \\ \hline \bar{\mathsf{K}}_{\Delta I}^{(i)} & \bar{\mathsf{K}}_{\Delta V}^{(i)} & \bar{\mathsf{K}}_{\Delta\Delta}^{(i)} & \bar{\mathsf{K}}_{A\Delta}^{(i)T} \\ \bar{\mathsf{K}}_{AI}^{(i)} & \bar{\mathsf{K}}_{AV}^{(i)} & \bar{\mathsf{K}}_{A\Delta}^{(i)} & \bar{\mathsf{K}}_{AA}^{(i)} \end{array}\right].$$

Ordering the primal variables last, we obtain

$$\mathsf{T}^{(i)T}\mathsf{K}^{(i)}\mathsf{T}^{(i)} = \left[\begin{array}{cc|cc} \mathsf{K}_{II}^{(i)} & \bar{\mathsf{K}}_{\Delta I}^{(i)T} & \mathsf{K}_{VI}^{(i)T} & \bar{\mathsf{K}}_{AI}^{(i)T} \\ \bar{\mathsf{K}}_{\Delta I}^{(i)} & \bar{\mathsf{K}}_{\Delta\Delta}^{(i)} & \bar{\mathsf{K}}_{\Delta V}^{(i)} & \bar{\mathsf{K}}_{A\Delta}^{(i)T} \\ \hline \mathsf{K}_{VI}^{(i)} & \bar{\mathsf{K}}_{\Delta V}^{(i)T} & \mathsf{K}_{VV}^{(i)} & \bar{\mathsf{K}}_{AV}^{(i)T} \\ \bar{\mathsf{K}}_{AI}^{(i)} & \bar{\mathsf{K}}_{A\Delta}^{(i)} & \bar{\mathsf{K}}_{AV}^{(i)} & \bar{\mathsf{K}}_{AA}^{(i)} \end{array}\right],$$

and using the substitution $[\hat{\mathbf{u}}_V, \hat{\mathbf{u}}_A] = \hat{\mathbf{u}}_\Pi$, we have

$$\begin{bmatrix} \mathsf{K}_{II}^{(i)} & \bar{\mathsf{K}}_{\Delta I}^{(i)T} & \hat{\mathsf{K}}_{\Pi I}^{(i)T} \\ \bar{\mathsf{K}}_{\Delta I}^{(i)} & \bar{\mathsf{K}}_{\Delta\Delta}^{(i)} & \hat{\mathsf{K}}_{\Pi\Delta}^{(i)T} \\ \hat{\mathsf{K}}_{\Pi I}^{(i)} & \hat{\mathsf{K}}_{\Pi\Delta}^{(i)} & \hat{\mathsf{K}}_{\Pi\Pi}^{(i)} \end{bmatrix}.$$

Assembling the primal contributions of each transformed $\mathsf{K}^{(i)}$ to $\tilde{\mathsf{K}}_{\Pi\Pi}$, we obtain the transformed stiffness matrix $\tilde{\mathsf{K}}$ in the form

$$\tilde{\mathsf{K}} = \left[\begin{array}{ccccc|c} \mathsf{K}_{II}^{(1)} & \bar{\mathsf{K}}_{\Delta I}^{(1)T} & & & & \tilde{\mathsf{K}}_{\Pi I}^{(1)T} \\ \bar{\mathsf{K}}_{\Delta I}^{(1)} & \bar{\mathsf{K}}_{\Delta\Delta}^{(1)} & & & & \tilde{\mathsf{K}}_{\Pi\Delta}^{(1)T} \\ & & \ddots & & & \vdots \\ & & & \mathsf{K}_{II}^{(s)} & \bar{\mathsf{K}}_{\Delta I}^{s)T} & \tilde{\mathsf{K}}_{\Pi I}^{(s)T} \\ & & & \bar{\mathsf{K}}_{\Delta I}^{(s)} & \bar{\mathsf{K}}_{\Delta\Delta}^{(s)} & \tilde{\mathsf{K}}_{\Pi\Delta}^{(s)T} \\ \hline \tilde{\mathsf{K}}_{\Pi I}^{(1)} & \tilde{\mathsf{K}}_{\Pi\Delta}^{(1)} & \dots & \tilde{\mathsf{K}}_{\Pi I}^{(s)} & \tilde{\mathsf{K}}_{\Pi\Delta}^{(s)} & \tilde{\mathsf{K}}_{\Pi\Pi} \end{array}\right].$$

Now we rewrite problem (3.2)

$$\min \frac{1}{2}\mathbf{u}^T\mathsf{K}\mathbf{u} - \mathbf{f}^T\mathbf{u}$$

as

$$\min \frac{1}{2}\hat{\mathbf{u}}^T\mathsf{T}^T\mathsf{K}\mathsf{T}\hat{\mathbf{u}} - \mathbf{f}^T\mathsf{T}\hat{\mathbf{u}} = \min \frac{1}{2}\hat{\mathbf{u}}^T\tilde{\mathsf{K}}\hat{\mathbf{u}} - \hat{\mathbf{u}}^T\hat{\mathbf{f}}, \tag{6.12}$$

where $\hat{\mathbf{u}}$, $\hat{\mathbf{f}}$ denote the vector of unknowns and the load vector in the new basis, respectively. Using the process described in Chapter 5 and PCG (Algorithm 4) with Dirichlet preconditioner (Section 5.4), we obtain the solution to this problem. To obtain the primal solution we need to use the backward transformation in the form of (6.7).

Remarks

- The transformation of basis changes the sparsity pattern of the transformed matrices $\mathsf{T}^{(i)T}\mathsf{K}^{(i)}\mathsf{T}^{(i)}$ compared to that of the original local stiffness matrices $\mathsf{K}^{(i)}$, but only the matrix blocks related to the edge variables are affected, see (6.11). Thus, the transformation of basis only slightly affects the sparsity pattern.

- If the transformation of basis is used, then the edge averages constraints can be treated algoritmically exactly in the same way as primal vertices. After the change of basis has been carried out, we can always use the same implementation of FETI-DP as the description of the algorithm in Chapter 5 does not depend on a specific choice of the primal and dual displacement variables. We note that the local problems as well as the Schur complement $\tilde{\mathsf{S}}_{\Pi\Pi}$ remain symmetric positive definite.

- In the papers [29, 25, 31] the transformation of basis for linear problems is applied in combination with Dirichlet preconditioner introduced in Section 5.4.
- The set V can be formed by another well choosen primal nodes, not only by the vertices, as was presented in this chapter. We also allow the set V to be empty. This situation is depicted in Figure 6.4.
- The averages can be situated to arbitrary nodes on the edge.

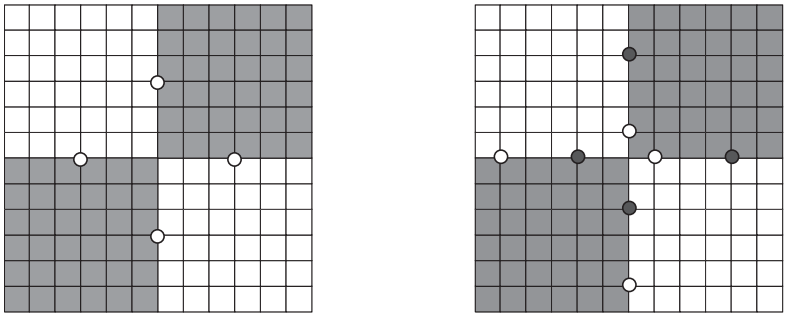


Figure 6.4: Illustration of the nodes of set V (black nodes) and the averages (white nodes). *(left)* Empty set V and averages in the middle of the edges. *(right)* Set V formed by some well chosen primal nodes and averages in arbitrary node of the edge.

- For the solution of 2D and 3D elasticity problems we introduce the transformation matrix in the form

$$\mathsf{T}_E = \begin{bmatrix} \mathsf{I}_d & \dots & \mathsf{O} & \mathsf{I}_d \\ & \ddots & & \vdots \\ \mathsf{O} & & \mathsf{I}_d & \mathsf{I}_d \\ -\mathsf{I}_d & \dots & -\mathsf{I}_d & \mathsf{I}_d \end{bmatrix}, \tag{6.13}$$

  where the I_d is an identity matrix of order 2 or 3.

II

Nonlinear problems

Model variational inequality problem 7

In this section we describe the model contact problem used throughout this part of the thesis.

Let $\Omega = (0,1) \times (0,1)$ be an open domain with the boundary $\partial\Omega$, and by ν we denote the outward normal to $\partial\Omega$. Let us consider a two-dimensional mixed problem depicted in Figure 7.1, with an obstacle ℓ under the contact boundary Γ_c, a Dirichlet boundary condition on Γ_D, and a Neumann boundary condition on Γ_N, so that

$$\left\{\begin{array}{rcll} -\Delta u & = & f & \text{in } \Omega \\ u & = & 0 & \text{on } \Gamma_D \\ \dfrac{\partial u}{\partial \nu} & = & 0 & \text{on } \Gamma_N, \\ u & \geq & \ell & \text{on } \Gamma_c, \end{array}\right. \tag{7.1}$$

where

$$\begin{array}{rcl} \Gamma_c & = & \{1\} \times [0,1], \\ \Gamma_D & = & \{0\} \times [0,1], \\ \Gamma_N & = & \{[0,1] \times \{0\}\} \cup \{[0,1] \times \{1\}\} \end{array}$$

are disjoint subsets of $\partial\Omega$.

The solution to this problem is shown in Figure 7.2. It can be interpreted as the displacement of the membrane under the traction defined by the density f. The membrane is fixed on Γ_D and it is not allowed to penetrate the obstacle on Γ_c.

Our contact model problem (7.1) can also be discretized by the finite element method in the similar way as the linear model problem (2.1).

After the discretization we write the problem (7.1) in the form

$$\min_{\mathbf{u} \in \Upsilon_B} \phi(\mathbf{u}), \quad \Upsilon_B = \{\mathbf{u} \in \mathbb{R}^n : \ \mathbf{u}_{\mathcal{I}} \geq \ell_{\mathcal{I}}\}, \quad \mathcal{I} = \{n-k, \ldots, n-1, n\}, \tag{7.2}$$

where $\phi(\mathbf{u}) = \frac{1}{2}\mathbf{u}^T \mathsf{K}\mathbf{u} - \mathbf{u}^T\mathbf{f}$ is a quadratic function. Since the feasible set Υ_B is described by the bound constraints on some variables, the problems of this type are called partially bound constrained problems.

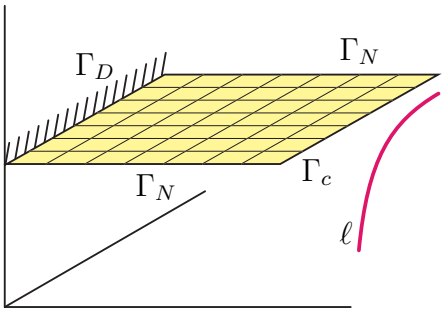


Figure 7.1: Two-dimensional problem with the Dirichlet boundary condition on Γ_D and the homogeneous Neumann boundary condition elsewhere (on Γ_N).

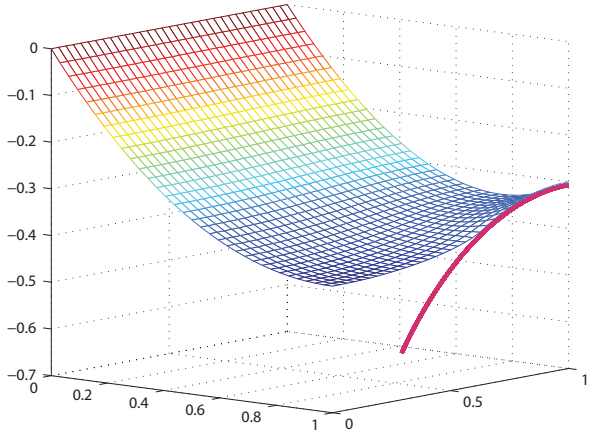


Figure 7.2: The solution to the model problem.

Let us now consider an inequality constrained problem, where the feasible region is described by linear inequalities. So, we want to find

$$\min_{\mathbf{u}\in\Upsilon_I} \phi(\mathbf{u}), \quad \Upsilon_I = \{\mathbf{u} \in \mathbb{R}^n : \mathsf{B}_{\mathcal{I}}\mathbf{u} \leq \mathbf{c}\}. \tag{7.3}$$

Denoting $\mathsf{B}_{\mathcal{I}} = [\mathsf{O}, -\mathsf{I}]$ and $\mathbf{c} = [\mathbf{o}^T, -\ell_{\mathcal{I}}^T]^T$ we can observe that

$$\Upsilon_I = \{\mathbf{u} \in \mathbb{R}^n : -\mathsf{I}\mathbf{u}_{\mathcal{I}} \leq -\ell_{\mathcal{I}}\} = \{\mathbf{u} \in \mathbb{R}^n : \ \mathbf{u}_{\mathcal{I}} \geq \ell_{\mathcal{I}}\} = \Upsilon_B,$$

so the problem (7.2) is a special case of the problem (7.3). Using theory of duality the problem (7.3) can be transformed to problem (7.2).

Numerical solution 8

In this chapter we describe the MPRGP (modified proportioning with reduced gradient projections) algorithm for the solution of bound constrained quadratic programming problems (7.2). This algorithm, based on active set strategy, was introduced by Dostál and Schöberl [14]. They proved the rate of convergence of this algorithm in terms of the spectral condition number of the Hessian matrix.

8.1 Basic terms

It is well known that the solution to the problem (7.2) always exists, and it is necessarily unique [3]. To simplify our notation, let us denote, for any n-vector $\mathbf{u}$, the gradient of ϕ at $\mathbf{u}$ by

$$\mathbf{g} = \mathbf{g}(\mathbf{u}) = \mathsf{K}\mathbf{u} - \mathbf{f}. \tag{8.1}$$

Then the unique solution $\widehat{\mathbf{u}}$ of (7.2) is fully determined by the Karush-Kuhn-Tucker (KKT) optimality conditions [3]. To describe them in more detail, let

$$\mathcal{N} = \{1, 2, \ldots, n\},$$

and let $\mathcal{I}$ denote the set of indices of the constrained variables from problem (7.2). Then the KKT conditions read

$$\begin{aligned} \widehat{u}_i = \ell_i \quad \text{and} \quad i \in \mathcal{I} \qquad &\text{implies} \quad \widehat{g}_i \geq 0, \\ \text{and} \quad \widehat{u}_i > \ell_i \quad \text{or} \quad i \in \mathcal{N} \setminus \mathcal{I} \quad &\text{implies} \quad \widehat{g}_i = 0. \end{aligned} \tag{8.2}$$

The set of all indices $i \in \mathcal{I}$ for which $u_i = \ell_i$ is called an *active set* of $\mathbf{u}$. We denote it by $\mathcal{A}(\mathbf{u})$, i.e.,

$$\mathcal{A}(\mathbf{u}) = \{i \in \mathcal{I} : u_i = \ell_i\}.$$

The complement $\mathcal{F}(\mathbf{u}) = \mathcal{N} \setminus \mathcal{A}(\mathbf{u})$ of $\mathcal{A}(\mathbf{u})$ will be called a *free set* of $\mathbf{u}$.

To enable an alternative reference to the Karush-Kuhn-Tucker conditions (8.2), we introduce a notation for the *free gradient* φ that is defined by

$$\varphi_i(\mathbf{u}) = \begin{cases} g_i(\mathbf{u}) & \text{for} \quad i \in \mathcal{F}(\mathbf{u}) \\ 0 & \text{for} \quad i \in \mathcal{A}(\mathbf{u}) \end{cases} \tag{8.3}$$

and the *chopped gradient* β that is defined by

$$\beta_i(\mathbf{u}) = \begin{cases} 0 & \text{for } i \in \mathcal{F}(\mathbf{u}) \\ g_i^-(\mathbf{u}) & \text{for } i \in \mathcal{A}(\mathbf{u}) \end{cases}, \tag{8.4}$$

where we used the notation $g_i^- = \min\{g_i, 0\}$. Thus the Karush-Kuhn-Tucker conditions (8.2) are satisfied if and only if the *projected gradient* $\mathbf{g}^P(\mathbf{u}) = \varphi(\mathbf{u}) + \beta(\mathbf{u})$ is equal to zero.

The projection P_Ω to the set of feasible vectors is defined for any n-vector $\mathbf{u}$ by

$$P_\Omega(\mathbf{u})_i = \begin{cases} \max\{u_i, \ell_i\} & \text{for } i \in \mathcal{I} \\ u_i & \text{for } i \in \mathcal{N} \setminus \mathcal{I} \end{cases}.$$

8.2 MPRGP algorithm

Let us briefly describe the algorithm [14] for the solution of (7.2) that combines the proportioning algorithm [5] with the gradient projections [41]. We use a given constant $\Gamma > 0$, a test to decide about leaving the face, and three types of steps to generate a sequence of iterates $\{\mathbf{u}^k\}$ that approximate the solution of (7.2).

The *expansion step* is defined by

$$\mathbf{u}^{k+1} = P_\Omega\left(\mathbf{u}^k - \overline{\alpha}\varphi(\mathbf{u}^k)\right) \tag{8.5}$$

with the fixed step length $\overline{\alpha} \in (0, \|\mathsf{K}\|^{-1}]$. This step may expand the current active set. To describe it without P_Ω, let us introduce, for any feasible $\mathbf{u}$, the *reduced free gradient* $\widetilde{\varphi}(\mathbf{u})$ with the entries

$$\widetilde{\varphi}_i = \widetilde{\varphi}_i(\mathbf{u}) = \min\{(u_i - \ell_i)/\overline{\alpha}, \varphi_i\} \quad \text{for } i \in \mathcal{I}, \qquad \widetilde{\varphi}_i = \varphi_i \quad \text{for } i \in \mathcal{N} \setminus \mathcal{I},$$

so that

$$P_\Omega\left(\mathbf{u} - \overline{\alpha}\varphi(\mathbf{u})\right) = \mathbf{u} - \overline{\alpha}\widetilde{\varphi}(\mathbf{u}). \tag{8.6}$$

If the inequality

$$||\beta(\mathbf{u}^k)||^2 \le \Gamma^2 \widetilde{\varphi}(\mathbf{u}^k)^T \varphi(\mathbf{u}^k) \tag{8.7}$$

holds, then we call the iterate $\mathbf{u}^k$ *strictly proportional*. The test (8.7) is used to decide which component of the projected gradient $\mathbf{g}^P(\mathbf{u}^k)$ will be reduced in the next step.

The *proportioning step* is defined by

$$\mathbf{u}^{k+1} = \mathbf{u}^k - \alpha_{cg}\beta(\mathbf{u}^k) \tag{8.8}$$

with the step length α_{cg} that minimizes $f\left(\mathbf{u}^k - \alpha\beta(\mathbf{u}^k)\right)$ with respect to α. It is easy to check [1] that α_{cg} minimizing $f(\mathbf{u} - \alpha\mathbf{d})$ for a given $\mathbf{d}$ and $\mathbf{u}$ may be evaluated by the formula

$$\alpha_{cg} = \alpha_{cg}(\mathbf{d}) = \frac{\mathbf{d}^T\mathbf{g}(\mathbf{u})}{\mathbf{d}^T\mathsf{K}\mathbf{d}}. \tag{8.9}$$

The purpose of the proportioning step is to remove indices from the active set.

Algorithm 7. Modified proportioning with reduced gradient projections (MPRGP)

Given a symmetric positive definite matrix K *of the order n, n-vectors* $\mathbf{f}$, ℓ, $\Omega_B = \{\mathbf{u} \in \mathbb{R}^n : \ \mathbf{u} \geq \ell\}$; *choose* $\mathbf{u}^0 \in \Omega_B$, $\Gamma > 0$, $\overline{\alpha} \in (0, 2\|\mathsf{A}\|^{-1}]$.

Step 0. {*Initialization.*}
Set $k = 0$, $\mathbf{g} = \mathsf{K}\mathbf{u}^0 - \mathbf{f}$, $\mathbf{p} = \varphi(u^0)$
while $\|\mathbf{g}^P(\mathbf{u}^k)\|$ *is not small*
if $\|\beta(\mathbf{u}^k)\|^2 \leq \Gamma^2 \widetilde{\varphi}(\mathbf{u}^k)^T \varphi(\mathbf{u}^k)$
Step 1. {*Proportional* $\mathbf{u}^k$. *Trial conjugate gradient step.*}
$\alpha_{cg} = \mathbf{g}^T\mathbf{p}/\mathbf{p}^T\mathsf{K}\mathbf{p}$, $\mathbf{y} = \mathbf{u}^k - \alpha_{cg}\mathbf{p}$
$\alpha_f = \max\{\alpha : \ \mathbf{u}^k - \alpha\mathbf{p} \in \Omega_B\} = \min\{(u_i^k - \ell_i)/p_i : \ p_i > 0\}$
if $\alpha_{cg} \leq \alpha_f$
Step 2. {*Conjugate gradient step.*}
$\mathbf{u}^{k+1} = \mathbf{y}$, $\mathbf{g} = \mathbf{g} - \alpha_{cg}\mathsf{K}\mathbf{p}$
$\beta = \varphi(\mathbf{y})^T\mathsf{K}\mathbf{p}/\mathbf{p}^T\mathsf{K}\mathbf{p}$, $\mathbf{p} = \varphi(\mathbf{y}) - \beta\mathbf{p}$
else
Step 3. {*Expansion step.*}
$\mathbf{u}^{k+\frac{1}{2}} = \mathbf{u}^k - \alpha_f\mathbf{p}$, $\mathbf{g} = \mathbf{g} - \alpha_f\mathsf{K}\mathbf{p}$
$\mathbf{u}^{k+1} = P_{\Omega_B}\left(\mathbf{u}^{k+\frac{1}{2}} - \overline{\alpha}\varphi(\mathbf{u}^{k+\frac{1}{2}})\right)$
$\mathbf{g} = \mathsf{K}\mathbf{u}^{k+1} - \mathbf{f}$, $\mathbf{p} = \varphi(\mathbf{u}^{k+1})$
end if
else
Step 4. {*Proportioning step.*}
$\mathbf{d} = \beta(\mathbf{u}^k)$, $\alpha_{cg} = \mathbf{g}^T\mathbf{d}/\mathbf{d}^T\mathsf{K}\mathbf{d}$
$\mathbf{u}^{k+1} = \mathbf{u}^k - \alpha_{cg}\mathbf{d}$, $\mathbf{g} = \mathbf{g} - \alpha_{cg}\mathsf{K}\mathbf{d}$, $\mathbf{p} = \varphi(\mathbf{u}^{k+1})$
end if
$k = k + 1$
end while
Step 5. {*Return (possibly inexact) solution.*}
$\widetilde{\mathbf{u}} = \mathbf{u}^k$

The *conjugate gradient step* is defined by

$$\mathbf{u}^{k+1} = \mathbf{u}^k - \alpha_{cg}\mathbf{p}^k \tag{8.10}$$

where $\mathbf{p}^k$ is the conjugate gradient direction [1] which is constructed recurrently. The recurrence starts or restarts with $\mathbf{p}^s = \varphi(\mathbf{u}^s)$ whenever $\mathbf{u}^s$ is generated by the expansion step or the proportioning step. If $\mathbf{p}^k$ is known, then $\mathbf{p}^{k+1}$ is given by the formula [1]

$$\mathbf{p}^{k+1} = \varphi(\mathbf{u}^k) - \gamma\mathbf{p}^k, \quad \gamma = \frac{\varphi(\mathbf{u}^k)^T\mathsf{K}\mathbf{p}^k}{(\mathbf{p}^k)^T\mathsf{K}\mathbf{p}^k}. \tag{8.11}$$

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

$$\mathcal{W}_{\mathcal{J}} = \{\mathbf{u} : u_i = \ell_i \text{ for } i \in \mathcal{J} \ \} \tag{8.12}$$

given by $\mathcal{J} = \mathcal{A}(\mathbf{u}^s)$ efficiently.

The algorithm MPRGP is described in Algorithm 7. More details about implementation of the algorithm can be found in [14]. The basic properties of the algorithm are summed up in the following theorem.

Theorem 8. *Let $\Gamma > 0$ be a given constant, let $\lambda_{\min}$ denote the smallest eigenvalue of K, $\widehat{\Gamma} = \max\{\Gamma, \Gamma^{-1}\}$, let $\widehat{\mathbf{u}}$ denote the unique solution of (7.2), and let $\{\mathbf{u}^k\}$ denote the sequence generated by Algorithm 7 with $\overline{\alpha} \in (0, \|\mathsf{K}\|^{-1}]$. Then the following statements hold:*

(i) The rate of convergence in the energy norm defined by $\|\mathbf{u}\|_{\mathsf{K}}^2 = \mathbf{u}^T\mathsf{K}\mathbf{u}$ is given by

$$\|\mathbf{u}^k - \widehat{\mathbf{u}}\|_{\mathsf{K}}^2 \leq 2\eta^k \left(\phi(\mathbf{u}^0) - \phi(\widehat{\mathbf{u}})\right), \tag{8.13}$$

where

$$\eta = 1 - \frac{\overline{\alpha}\lambda_{\min}}{2 + 2\widehat{\Gamma}^2}. \tag{8.14}$$

(ii) If the solution $\widehat{\mathbf{u}}$ satisfies the strict complementarity conditions, i.e., $\widehat{u}_i = 0$ implies $g_i(\widehat{\mathbf{u}}) > 0$, then there is $k \geq 0$ such that $\mathbf{u}^k = \widehat{\mathbf{u}}$.

(iii) If Γ and the spectral condition number $\kappa(\mathsf{K})$ of K satisfy

$$\Gamma \geq 2\left(\sqrt{\kappa(\mathsf{K})} + 1\right), \tag{8.15}$$

then there is $k \geq 0$ such that $\mathbf{u}^k = \widehat{\mathbf{u}}$.

Proof. See [14]. □

Dual-Primal FETI methods

9

In this chapter we describe FETI-DP method for the solution of problems described by variational inequalities. Even though this method was originally developed for the solution of linear problems [15], it has been observed that domain decomposition methods, based on theory of duality, may also be successful for the solution of variational inequalities. Since the duality transforms more general inequality constraints to bound constraints, the dual problem can be solved much more efficiently than the primal problem. Results concerning application of basic FETI-DP to the solution of variational inequalities can be found in [12, 13].

FETI-DP for variational inequalities

We use the notation introduced in Section 5.1. Using the theory of duality [11] let us now derive the dual problem to reduce (7.2) to the subdmain interface Γ and contact boundary Γ_c. Let us denote the Lagrange multipliers associated with the inequality and equality constraints by $\lambda_{\mathcal{I}}$ and $\lambda_{\mathcal{E}}$, respectively. The situation is depicted in Figure 9.1.

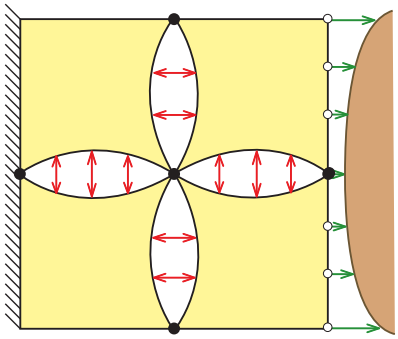


Figure 9.1: Lagrange multipliers associated with the inequality constraints $\lambda_{\mathcal{I}}$ (green arrows) and with the equality constraints $\lambda_{\mathcal{E}}$ (red arrows).

The Lagrangian function [20, 12] associated with (7.2) is of the form

$$L_0(\mathbf{u},\lambda) = \frac{1}{2}\mathbf{u}^T\widetilde{\mathsf{K}}\mathbf{u} - \mathbf{u}^T\tilde{\mathbf{f}} + (\mathsf{B}\mathbf{u}-\mathbf{c})^T\lambda, \tag{9.1}$$

where

$$\mathbf{c} = \begin{bmatrix} \mathbf{o} \\ -\ell_{\mathcal{I}} \end{bmatrix}, \qquad \lambda = \begin{bmatrix} \lambda_{\mathcal{E}} \\ \lambda_{\mathcal{I}} \end{bmatrix} \tag{9.2}$$

and

$$\mathsf{B} = \begin{bmatrix} \mathsf{B}_{\mathcal{E}B} & \mathsf{O} \\ \mathsf{B}_{\mathcal{I}B} & \mathsf{O} \end{bmatrix} = \begin{bmatrix} \mathsf{B}_B & \mathsf{O} \end{bmatrix} \quad \Big(= \begin{bmatrix} \mathsf{B}_B & \mathsf{B}_\Pi \end{bmatrix} \Big). \tag{9.3}$$

The continuity at the dual displacement variables is enforced by the jump operator $\mathsf{B}_{\mathcal{E}B}$, which is constructed from $\{-1,0,1\}$, in such a way that the values of the solution $\mathbf{u}_B$, associated with more than one subdomain, coincide when $\mathsf{B}_{\mathcal{E}B}\mathbf{u}_B = 0$; the interior variables $\mathbf{u}_I$ remain unchanged and thus the corresponding entries in $\mathsf{B}_{\mathcal{E}B}$ are zero.

Matrix $\mathsf{B}_{\mathcal{I}B}$, which enforces inequality constraints, has entries corresponding to nodal values of the solution on a contact boundary equal to -1; the rest of the variables remains unchanged and thus the corresponding entries in $\mathsf{B}_{\mathcal{I}B}$ are zero.

Using the notation described in Chapter 5 and (9.2), we rewrite (9.1) as

$$\begin{aligned} L_0(\mathbf{u}_B,\tilde{\mathbf{u}}_\Pi,\lambda) &= \frac{1}{2}\mathbf{u}_B^T\mathsf{K}_{BB}\mathbf{u}_B + \mathbf{u}_B^T\mathsf{K}_{\Pi B}^T\mathsf{L}\tilde{\mathbf{u}}_\Pi + \frac{1}{2}\tilde{\mathbf{u}}_\Pi^T\mathsf{L}^T\mathsf{K}_{\Pi\Pi}\mathsf{L}\tilde{\mathbf{u}}_\Pi - \mathbf{f}_B^T\mathbf{u}_B \\ &- \mathbf{f}_\Pi^T\mathsf{L}\tilde{\mathbf{u}}_\Pi + \mathbf{u}_B^T\mathsf{B}_B^T\lambda - \mathbf{c}^T\lambda \\ &= \frac{1}{2}\mathbf{u}_B^T\mathsf{K}_{BB}\mathbf{u}_B - \mathbf{u}_B^T(\mathbf{f}_B - \mathsf{K}_{\Pi B}^T\mathsf{L}\tilde{\mathbf{u}}_\Pi - \mathsf{B}_B^T\lambda) \\ &+ \frac{1}{2}\tilde{\mathbf{u}}_\Pi^T\mathsf{L}^T\mathsf{K}_{\Pi\Pi}\mathsf{L}\tilde{\mathbf{u}}_\Pi - \mathbf{f}_\Pi^T\mathsf{L}\tilde{\mathbf{u}}_\Pi - \mathbf{c}^T\lambda. \end{aligned} \tag{9.4}$$

To minimize $L_0(\mathbf{u}_B,\tilde{\mathbf{u}}_\Pi,\lambda)$ over $\mathbf{u}_B$, we consider

$$\frac{\partial L_0}{\partial \mathbf{u}_B} = \mathsf{K}_{BB}\mathbf{u}_B - \Big(\mathbf{f}_B - \mathsf{K}_{\Pi B}^T\mathsf{L}\tilde{\mathbf{u}}_\Pi - \mathsf{B}_B^T\lambda\Big) = 0$$

which implies

$$\mathbf{u}_B = \mathsf{K}_{BB}^{-1}\Big(\mathbf{f}_B - \mathsf{K}_{\Pi B}^T\mathsf{L}\tilde{\mathbf{u}}_\Pi - \mathsf{B}_B^T\lambda\Big).$$

By inserting this result into (9.4), we obtain

$$\begin{aligned} L_0(\tilde{\mathbf{u}}_\Pi,\lambda) &= \frac{1}{2}\tilde{\mathbf{u}}_\Pi^T\mathsf{L}^T\Big(\mathsf{K}_{\Pi\Pi} - \mathsf{K}_{\Pi B}\mathsf{K}_{BB}^{-1}\mathsf{K}_{\Pi B}^T\Big)\mathsf{L}\tilde{\mathbf{u}}_\Pi \\ &- \tilde{\mathbf{u}}_\Pi^T\mathsf{L}^T\Big(\mathbf{f}_\Pi - \mathsf{K}_{\Pi B}\mathsf{K}_{BB}^{-1}\mathbf{f}_B + \mathsf{K}_{\Pi B}\mathsf{K}_{BB}^{-1}\mathsf{B}_B^T\lambda\Big) \\ &- \frac{1}{2}(\mathbf{f}_B - \mathsf{B}_B^T\lambda)^T\mathsf{K}_{BB}^{-1}(\mathbf{f}_B - \mathsf{B}_B^T\lambda) - \mathbf{c}^T\lambda \\ &= \frac{1}{2}\tilde{\mathbf{u}}_\Pi^T\tilde{\mathsf{S}}_{\Pi\Pi}\tilde{\mathbf{u}}_\Pi - \tilde{\mathbf{u}}_\Pi^T(\widehat{\mathbf{f}}_\Pi - \widehat{\mathsf{K}}_{\Pi B}\lambda) \\ &- \frac{1}{2}(\mathbf{f}_B - \mathsf{B}_B^T\lambda)^T\mathsf{K}_{BB}^{-1}(\mathbf{f}_B - \mathsf{B}_B^T\lambda) - \mathbf{c}^T\lambda, \end{aligned} \tag{9.5}$$

where

$$\begin{aligned}\tilde{\mathsf{S}}_{\Pi\Pi} &= \mathsf{L}^T\Big(\mathsf{K}_{\Pi\Pi} - \mathsf{K}_{\Pi B}\mathsf{K}_{BB}^{-1}\mathsf{K}_{\Pi B}^T\Big)\mathsf{L},\\ \widehat{\mathbf{f}}_{\Pi} &= \mathsf{L}^T\Big(\mathbf{f}_{\Pi} - \mathsf{K}_{\Pi B}\mathsf{K}_{BB}^{-1}\mathbf{f}_B\Big),\\ \widehat{\mathsf{K}}_{\Pi B}^T &= -\mathsf{B}_B\mathsf{K}_{BB}^{-1}\mathsf{K}_{\Pi B}^T\mathsf{L}. \end{aligned} \tag{9.6}$$

In order to minimize (9.5) over $\tilde{\mathbf{u}}_{\Pi}$, we consider

$$\frac{\partial L_0}{\partial \tilde{\mathbf{u}}_{\Pi}} = \tilde{\mathsf{S}}_{\Pi\Pi}\tilde{\mathbf{u}}_{\Pi} - \Big(\widehat{\mathbf{f}}_{\Pi} - \widehat{\mathsf{K}}_{\Pi B}\lambda\Big) = 0$$

and obtain

$$\tilde{\mathbf{u}}_{\Pi} = \tilde{\mathsf{S}}_{\Pi\Pi}^{-1}\Big(\widehat{\mathbf{f}}_{\Pi} - \widehat{\mathsf{K}}_{\Pi B}\lambda\Big), \quad \mathbf{u}_{\Pi} = \mathsf{L}\tilde{\mathbf{u}}_{\Pi}.$$

Using this result, we can rewrite (9.5) in the form

$$\begin{aligned} L_0(\lambda) &= \frac{1}{2}\lambda^T\Big(\widehat{\mathsf{K}}_{BB} - \widehat{\mathsf{K}}_{\Pi B}^T\tilde{\mathsf{S}}_{\Pi\Pi}^{-1}\widehat{\mathsf{K}}_{\Pi B}\Big)\lambda - \lambda^T\Big(\widehat{\mathbf{f}}_B - \widehat{\mathsf{K}}_{\Pi B}^T\tilde{\mathsf{S}}_{\Pi\Pi}^{-1}\widehat{\mathbf{f}}_{\Pi} + \mathbf{c}\Big)\\ &- \frac{1}{2}\mathbf{f}_B^T\mathsf{K}_{BB}^{-1}\mathbf{f}_B - \frac{1}{2}\widehat{\mathbf{f}}_{\Pi}^T\tilde{\mathsf{S}}_{\Pi\Pi}^{-1}\widehat{\mathbf{f}}_{\Pi}, \end{aligned} \tag{9.7}$$

where $\widehat{\mathsf{K}}_{BB} = -\mathsf{B}_B\mathsf{K}_{BB}^{-1}\mathsf{B}_B^T$ and $\widehat{\mathbf{f}}_B = -\mathsf{B}_B\mathsf{K}_{BB}^{-1}\mathbf{f}_B$.

Using the notation of theory of duality, we rewrite (9.7) as the dual problem

$$\min_{\lambda\geq 0}\Theta(\lambda), \quad \Theta(\lambda) = \frac{1}{2}\lambda^T\mathsf{F}\lambda - \lambda^T\mathbf{d}, \tag{9.8}$$

where $\mathsf{F} = \widehat{\mathsf{K}}_{BB} - \widehat{\mathsf{K}}_{\Pi B}^T\tilde{\mathsf{S}}_{\Pi\Pi}^{-1}\widehat{\mathsf{K}}_{\Pi B}$ and $\mathbf{d} = \widehat{\mathbf{f}}_B - \widehat{\mathsf{K}}_{\Pi B}^T\tilde{\mathsf{S}}_{\Pi\Pi}^{-1}\widehat{\mathbf{f}}_{\Pi} + \mathbf{c}$. Minimizing $\Theta(\lambda)$ over $\lambda \geq 0$ is equivalent to solving problem (7.2).

Remarks

- If the primal displacement variables are used on the contact boundary it is necessary to carry out some modifications introduced by Horák in [20]. Let B_c be the matrix made of the columns of the matrix B corresponding to primal variables, including primal variables on the contact boundary and let L_c denote the global to local map enforcing the continuity at the primal variables, including primal variables on the contact boundary. Then, using a modification of (9.6) in the form

  $$\widehat{\mathsf{K}}_{\Pi B}^T = \widehat{\mathsf{K}}_{\Pi B}^T - \mathsf{B}_c\mathsf{L}_c, \tag{9.9}$$

  we can allow the primal variables also on the contact boundary [20] without further changes.

- Let us introduce two parameters describing the discretization in FETI methods. The discretization parameter h describes the size of the element, and the decomposition parameter H describes the size of the subdomain as depicted in Figure 9.2.

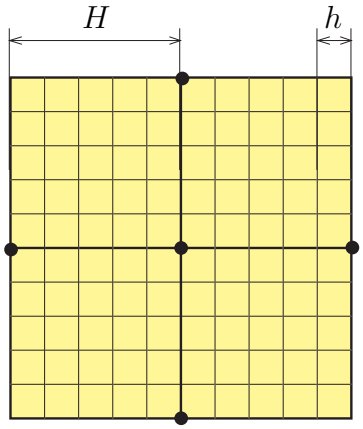


Figure 9.2: Parameters h and H in FETI methods.

Proposition 9. *Let $\mathsf{F}_{H,h}$ denote the Hessian of the dual function Θ of (9.8) defined by the decomposition parameter H and discretization parameter h. Then there are constants $C_1 > 0$ and $C_2 > 0$ independent of h and H such that*

$$C_1 \leq \lambda_{min}(\mathsf{F}_{H,h}) \quad \text{and} \quad \lambda_{max}(\mathsf{F}_{H,h}) = \|\mathsf{F}_{H,h}\| \leq C_2 \left(\frac{H}{h}\right)^2 .$$

Proof. See [12]. □

10 Preconditioning for nonlinear problems

Even if the preconditioning changes variables and transforms the bound constraints into more general inequality constraints, we can use some kind of preconditioning also for bound constraints problems.

10.1 Preconditioning in face

Though the performance of the algorithm can be improved considerably by the preconditioners described in this section. This type of preconditioning does not result in improved bounds on the rate of convergence, since this preconditioner affects only the CG step, leaving the expansion and the proportioning steps without any preconditioning.

Preconditioning out of contact boundary

The first idea is to use preconditioner, which doesn't affect the variables on the contact boundary. Figure 10.1 depicts this type of preconditioner. The nodes corresponding to affected variables are in the red rectangle, while the nodes, which are not affected, corresponding to variables on the contact boundary, are in the green rectagle. Let us call this preconditioning as *preconditioning out of contact boundary* and denote it by M^{out}.

Let us now describe the construction of the preconditioner $\mathsf{M}^{out} \in \mathbb{R}^{n\times n}$. First, we need to define an auxiliary logical vector $\mathbf{j}$ of the length n, with entries equal to zero on the positions corresponding to variables on the contact boundary and equal to one on the positions corresponding to variables away from the contact boundary, so that

$$j_i = \begin{cases} 0, & \text{for the } i-\text{th node on the contact boundary,} \\ 1, & \text{for the } i-\text{th node out of the contact boundary.} \end{cases} \tag{10.1}$$

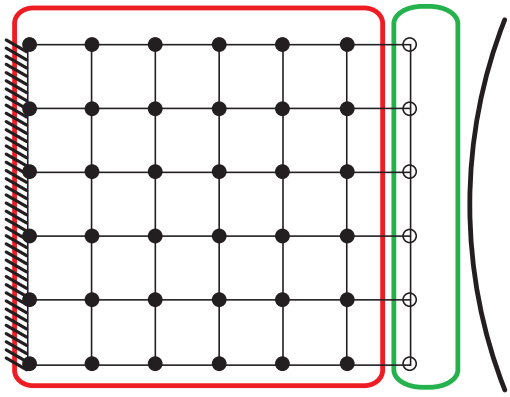


Figure 10.1: Preconditioning out of contact boundary. The nodes corresponding to affected variables are in the red rectangle, while the nodes, which are not affected, corresponding to variables on the contact boundary, are in the green rectangle.

Let $\mathsf{M} \in \mathbb{R}^{n \times n}$ denote preconditioning matrix generated, e.g., by any of the methods described in Section 4.1. The preconditioner M^{out} is constructed from M in the following way. Let M^{out} be initialized to a zero matrix

$$\mathsf{M}^{out} = \mathsf{O}.$$

Using the vector $\mathbf{j}$ (10.1) and the matrix M we set the entries of M^{out} corresponding to ones in $\mathbf{j}$ to the values at the same positions in matrix M, so that

$$\mathsf{M}^{out}(\mathbf{j}, \mathbf{j}) = \mathsf{M}(\mathbf{j}, \mathbf{j}).$$

Preconditioner M^{out} affects all variables out of the contact boundary, leaving variables on contact boundary without any change.

Preconditioning in face

Obviously preconditioning out of contact boundary can be improved, applying the preconditioner also for the free variables on the contact boundary. Let us now define an auxiliary logical vector $\mathbf{j}$ in the following way,

$$j_i = \begin{cases} 0, & \text{for the } i\text{−th node } \in \mathcal{A}, \\ 1, & \text{for the } i\text{−th node } \in \mathcal{F}, \end{cases} \tag{10.2}$$

where $\mathcal{A}$ and $\mathcal{F}$ ate the active and the free set, respectively, defined in Section 8.1. This preconditioner is changing, as well as the free and active sets. Let us denote it by M^{face}. Figure 10.2 depicted this type of preconditioner. Green nodes on the contact boundary are in the contact with the obstacle, so they are in the active set $\mathcal{A}$ denoted by the green rectangle. The white nodes on the contact boundary are not at the moment in contact, so they are, as well as black nodes inside the domain, in the free set $\mathcal{F}$ depicted by the red set.

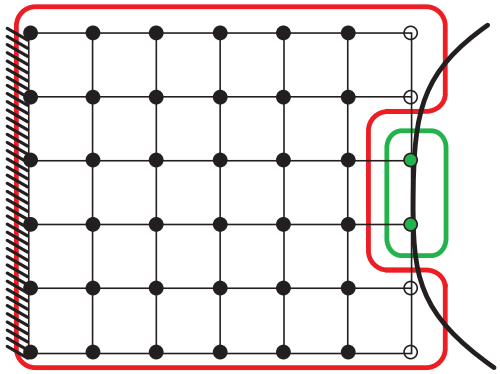


Figure 10.2: Preconditioning in face. Green nodes on the contact boundary are in the contact with the obstacle, so they are in active set $\mathcal{A}$ denoted by green rectangle. The white nodes on the contact boundary are not at the moment in contact, so they are, as well as black nodes inside the domain, in free set $\mathcal{F}$ depicted by red set.

The construction of the preconditioner M^{face} from matrix M is almost same as described in the previous part for M^{out}. Let M^{face} be initialized again to a zero matrix

$$\mathsf{M}^{face} = \mathsf{O}.$$

Using vector $\mathbf{j}$ defined by (10.2) and matrix M we can set the entries corresponding to free variables in M^{face} to the values at the same positions in matrix M, so that

$$\mathsf{M}^{face}(\mathbf{j},\mathbf{j}) = \mathsf{M}(\mathbf{j},\mathbf{j}).$$

We need to know the actual information about the free and the active sets, so the vector $\mathbf{j}$ is up-dated after every change of these sets.

Evidently, the preconditioning out of contact boundary is a special case of preconditioning in face.

Notes to implementation

Both types of preconditioning described in this section can be implemented in the way shown in Algorithm 10.

The preconditioner M^{out} can be established at the start of the algorithm, so that the matrix-vector multiplication $\mathbf{z} = \mathsf{M}(\mathbf{j},\mathbf{j})\,\mathbf{g}$ can be replaced by $\mathbf{z} = \mathsf{M}^{out}\,\mathbf{g}$.

Using preconditioning in face the matrix-vector multiplication $\mathbf{z} = \mathsf{M}(\mathbf{j},\mathbf{j})\,\mathbf{g}$ is in the fact implemented as `z = j.*(M*(j.*g))`, where `.*` denotes element-by-element multiplication.

Algorithm 10. MPRGP with preconditioning in face

Given a symmetric positive definite matrix K *of the order n, n-vectors* $\mathbf{f}$, ℓ, $\Upsilon_I = \{\mathbf{u} \in \mathbb{R}^n : \ \mathbf{u} \geq \ell\}$; *choose* $\mathbf{u}^0 \in \Upsilon_I$, $\Gamma > 0$, $\overline{\alpha} \in (0, 2\|\mathsf{A}\|^{-1}]$, *and logical vector* $\mathbf{j}$ *defined by 10.1 and 10.2, respectively.*

Step 0. {*Initialization.*}
  Set $k = 0$, $\mathbf{g} = \mathsf{K}\mathbf{u}^0 - \mathbf{b}$, $\mathbf{p} = \mathsf{M}(\mathbf{j},\mathbf{j})\,\mathbf{g}$
  while $\|\mathbf{g}^P(\mathbf{u}^k)\|$ *is not small*
    if $\|\beta(\mathbf{u}^k)\|^2 \leq \Gamma^2 \widetilde{\varphi}(\mathbf{u}^k)^T \varphi(\mathbf{u}^k)$
Step 1. {*Proportional* $\mathbf{u}^k$. *Trial conjugate gradient step.*}
      $\alpha_{cg} = \mathbf{z}^T\mathbf{g}/\mathbf{p}^T\mathsf{K}\mathbf{p}$, $\mathbf{y} = \mathbf{u}^k - \alpha_{cg}\mathbf{p}$
      $\alpha_f = \max\{\alpha : \ \mathbf{u}^k - \alpha\mathbf{p} \in \Omega_B\} = \min\{(x_i^k - \ell_i)/p_i : \ p_i > 0\}$
      if $\alpha_{cg} \leq \alpha_f$
Step 2. {*Conjugate gradient step.*}
        $\mathbf{u}^{k+1} = \mathbf{y}$, $\mathbf{g} = \mathbf{g} - \alpha_{cg}\mathsf{K}\mathbf{p}$, $\mathbf{z} = \mathsf{M}(\mathbf{j},\mathbf{j})\,\mathbf{g}$
        $\beta = \mathbf{z}^T\mathsf{K}\mathbf{p}/\mathbf{p}^T\mathsf{K}\mathbf{p}$, $\mathbf{p} = \mathbf{z} - \beta\mathbf{p}$
      else
Step 3. {*Expansion step.*}
        $\mathbf{u}^{k+\frac{1}{2}} = \mathbf{u}^k - \alpha_f\mathbf{p}$, $\mathbf{g} = \mathbf{g} - \alpha_f\mathsf{K}\mathbf{p}$
        $\mathbf{u}^{k+1} = P_{\Upsilon_I}\left(\mathbf{u}^{k+\frac{1}{2}} - \overline{\alpha}\varphi(\mathbf{u}^{k+\frac{1}{2}})\right)$
        $\mathbf{g} = \mathsf{K}\mathbf{u}^{k+1} - \mathbf{b}$, $\mathbf{p} = \mathsf{M}(\mathbf{j},\mathbf{j})\,\mathbf{g}$
      end if
    else
Step 4. {*Proportioning step.*}
      $\mathbf{d} = \beta(\mathbf{u}^k)$, $\alpha_{cg} = \mathbf{g}^T\mathbf{d}/\mathbf{d}^T\mathsf{K}\mathbf{d}$
      $\mathbf{u}^{k+1} = \mathbf{u}^k - \alpha_{cg}\mathbf{d}$, $\mathbf{g} = \mathbf{g} - \alpha_{cg}\mathsf{K}\mathbf{d}$, $\mathbf{p} = \mathsf{M}(\mathbf{j},\mathbf{j})\,\mathbf{g}$
    end if
    $k = k + 1$
  end while
Step 5. {*Return (possibly inexact) solution.*}
  $\widetilde{\mathbf{x}} = \mathbf{u}^k$

10.2 Preconditioning by conjugate projector

In this section we consider the preconditioning by the conjugate projector for the solution of (7.2). Using this approach which does not affect the constrained variables, it is possible to give an improvement bound on the rate of convergence of the preconditioned problem [4]. We use notation from Section 4.3.

Let us assume that $\mathcal{U}$ is the subspace spanned by the full column rank matrix $\mathsf{U} \in \mathbb{R}^{n\times p}$,

$$\mathsf{U} = \begin{bmatrix} \mathsf{V} \\ \mathsf{O} \end{bmatrix}, \quad \mathsf{V} \in \mathbb{R}^{(n-m)\times p}.$$

We shall decompose our partially constrained problem (7.2) by means of the conju-

gate projectors

$$\mathsf{P} = \mathsf{U}(\mathsf{U}^T\mathsf{K}\mathsf{U})^{-1}\mathsf{U}^T\mathsf{K} \tag{10.3}$$

and $\mathsf{Q} = \mathsf{I} - \mathsf{P}$ onto $\mathcal{U}$ and $\mathcal{V} = \mathrm{Im}\mathsf{Q}$, respectively. Due to our special choice of U, we get that for any $\mathbf{u} \in \mathbb{R}^n$ is

$$[\mathsf{Q}\mathbf{u}]_{\mathcal{I}} = \mathbf{u}_{\mathcal{I}}, \tag{10.4}$$

and that for any $\mathbf{y} \in \mathcal{U}$ and $\mathbf{z} \in \mathcal{V}$ is $\mathbf{y} + \mathbf{z} \in \Upsilon_I$ if and only if $\mathbf{z} \in \Upsilon_I$. Using (4.9), (4.10), and (10.4), we get

$$\begin{aligned}
\min_{\mathbf{u}\in\Upsilon_I} \phi(\mathbf{u}) &= \min_{\substack{\mathbf{y}\in\mathcal{U},\mathbf{z}\in\mathcal{V}\\ \mathbf{y}+\mathbf{z}\in\Upsilon_I}} \phi(\mathbf{y}+\mathbf{z}) = \min_{\mathbf{y}\in\mathcal{U}} \phi(\mathbf{y}) + \min_{\mathbf{z}\in\mathcal{V}\cap\Upsilon_I} \phi(\mathbf{z}) \\
&= \phi(\mathbf{u}^0) + \min_{\mathbf{z}\in\mathcal{V}\cap\Upsilon_I} \phi(\mathbf{z}) = \phi(\mathbf{u}^0) + \min_{\substack{\mathbf{z}\in\mathsf{K}\mathcal{V}\\ \mathbf{z}_{\mathcal{I}}\geq\ell_{\mathcal{I}}}} \frac{1}{2}\mathbf{z}^T\mathsf{Q}^T\mathsf{K}\mathsf{Q}\mathbf{z} - \mathbf{f}^T\mathsf{Q}\mathbf{z} \\
&= \phi(\mathbf{u}^0) + \min_{\substack{\mathbf{z}\in\mathsf{K}\mathcal{V}\\ \mathbf{z}_{\mathcal{I}}\geq\ell_{\mathcal{I}}}} \frac{1}{2}\mathbf{z}^T\mathsf{Q}^T\mathsf{K}\mathsf{Q}\mathbf{z} + \left(\mathbf{g}^0\right)^T\mathbf{z},
\end{aligned}$$

where $\mathbf{u}^0 = \mathsf{P}\mathsf{K}^{-1}\mathbf{f}$ and $\mathbf{g}^0 = -\mathsf{Q}^T\mathbf{f}$. Thus we have reduced our bound constrained problem (7.2) to the problem

$$\min_{\substack{\mathbf{z}\in\mathsf{K}\mathcal{V}\\ \mathbf{z}_{\mathcal{I}}\geq\ell_{\mathcal{I}}}} \frac{1}{2}\mathbf{z}^T\mathsf{Q}^T\mathsf{K}\mathsf{Q}\mathbf{z} + \left(\mathbf{g}^0\right)^T\mathbf{z}. \tag{10.5}$$

The following lemma shows that the above problem can be solved by the MPRGP algorithm without any change.

Lemma 11. *Let* $\mathbf{z}^1, \mathbf{z}^2, \ldots$ *be generated by the MPRGP algorithm for the problem*

$$\min_{\mathbf{z}_{\mathcal{I}}\geq\ell_{\mathcal{I}}} \frac{1}{2}\mathbf{z}^T\mathsf{Q}^T\mathsf{K}\mathsf{Q}\mathbf{z} + \left(\mathbf{g}^0\right)^T\mathbf{z} \tag{10.6}$$

starting from $\mathbf{z}^0 = P_{\Upsilon_I}(\mathbf{g}^0)$. *Then* $\mathbf{z}^k \in \mathsf{K}\mathcal{V},\ k = 0, 1, 2, \ldots$.

Proof. First observe that since $\mathsf{K}\mathcal{V}$ is orthogonal to $\mathcal{U}$ and $\dim \mathsf{K}\mathcal{V} = \dim \mathcal{V}$, it follows that $\mathsf{K}\mathcal{V}$ is the orthogonal complement of $\mathcal{U}$. Thus $\mathsf{K}\mathcal{V}$ is not only an invariant subspace of Q, but it is also an invariant subspace of P_{Υ_I}. Moreover, it also follows that $\mathsf{K}\mathcal{V}$ contains the set $\mathcal{V}_0 \subseteq \mathbb{R}^n$ of all the vectors of $\mathbb{R}^m$ padded with zeros,

$$\mathcal{V}_0 = \{\mathbf{u} \in \mathbb{R}^n :\ \mathbf{u}_{\mathcal{J}} = \mathbf{o},\ \mathcal{J} = \{m+1, \ldots, n\}\}.$$

More formally,

$$P_{\Upsilon_I}(\mathsf{K}\mathcal{V}) \subseteq \mathsf{K}\mathcal{V} \quad \text{and} \quad \mathcal{V}_0 \subseteq \mathsf{K}\mathcal{V}. \tag{10.7}$$

Let us now recall that by (4.10) $\mathbf{g}^0 \in \mathrm{Im}\mathsf{Q}^{\mathrm{T}}$ and by (4.11) $\mathrm{Im}\mathsf{Q}^{\mathrm{T}} = \mathsf{K}\mathcal{V}$, so that $\mathbf{g}^0 \in \mathsf{K}\mathcal{V}$. Using the definition of $\mathbf{z}^0$ and (10.7), we have $\mathbf{z}^0 \in \mathsf{K}\mathcal{V}$.

To finish the proof by induction, let us assume that $\mathbf{z}^k \in \mathsf{K}\mathcal{V}$. Since

$$\mathbf{g}^k = \mathsf{Q}^T\mathsf{K}\mathsf{Q}\mathbf{z} + \mathbf{g}^0 = \mathsf{K}\mathsf{Q}\mathbf{z} + \mathbf{g}^0,$$

we have $\mathbf{g}^k \in \mathsf{K}\mathcal{V}$. We shall use this observation to examine separately each of the three possible steps of the MPRGP algorithm of the previous section that can be used to generate $\mathbf{z}^{k+1}$.

First, let us assume that $\mathbf{z}^{k+1}$ is generated by the proportioning step. Then

$$\mathbf{z}^{k+1} = \mathbf{z}^k + \alpha_{cg}\beta(\mathbf{z}^k).$$

Using the definition of the chopped gradient, it is rather easy to check that $\beta(\mathbf{z}^k) \in \mathcal{V}_0$. Since $\mathsf{K}\mathcal{V}$ is a subspace of $\mathbb{R}^n$ and we assume that $\mathbf{z}^k \in \mathsf{K}\mathcal{V}$, this proves that $\mathbf{z}^{k+1} \in \mathsf{K}\mathcal{V}$ whenever $\mathbf{z}^{k+1}$ is generated by the proportioning step.

Before examining the other two steps, observe that $\varphi(\mathbf{z}^k) - \mathbf{g}^k \in \mathcal{V}_0$, so that

$$\varphi(\mathbf{z}^k) = \left(\varphi(\mathbf{z}^k) - \mathbf{g}^k\right) + \mathbf{g}^k \in \mathsf{K}\mathcal{V}.$$

Thus

$$\mathbf{z}^k - \alpha\varphi(\mathbf{z}^k) \in \mathsf{K}\mathcal{V}$$

for any $\alpha \in \mathbb{R}$. Using the first inclusion of (10.7), we get that

$$P_\Upsilon\left(\mathbf{z}^k - \overline{\alpha}\varphi(\mathbf{z}^k)\right) \in \mathsf{K}\mathcal{V}$$

for any $\overline{\alpha}$ of Algorithm 7. This proves that $\mathbf{z}^{k+1} \in \mathsf{K}\mathcal{V}$ for $\mathbf{z}^{k+1}$ generated by the expansion step. To finish the proof, observe that the conjugate direction $\mathbf{p}^k$ is either equal to $\varphi(\mathbf{z}^k)$ or is defined by the recurrence $\mathbf{p}^{k+1} = \varphi(\mathbf{u}^k) - \gamma\mathbf{p}^k$ starting from the restart $\mathbf{p}^s = \varphi(\mathbf{z}^s)$. In either case, $\mathbf{p}^k \in \mathsf{K}\mathcal{V}$. Since we assume that $\mathbf{z}^k \in \mathsf{K}\mathcal{V}$ and the iterate $\mathbf{z}^{k+1}$ generated by the conjugate gradient step is a linear combination of $\mathbf{z}^k$ and $\mathbf{p}^k$, this completes the proof. □

It follows that we obtain the correction $\widehat{\mathbf{z}}$ which solves the auxiliary problem by the standard MPRGP algorithm. Since the iterations are reduced to the subspace $\mathsf{K}\mathcal{V}$, the *projector preconditions all three types of steps* and we can give an improved bound on the rate of convergence. The solution $\widehat{\mathbf{x}}$ of the bound constrained problem (7.2) can then be expressed by $\widehat{\mathbf{x}} = \mathbf{u}^0 + \mathsf{Q}\widehat{\mathbf{z}}$. The complete algorithm for the solution of the preconditioned problem (10.6) is Algorithm 12. In our implementation, we enhanced a feasible halfstep introduced in [14].

Algorithm 12. MPRGP with conjugate projector

Given a symmetric positive definite matrix K *of the order* n*, a full column rank matrix* $\mathsf{U} \in \mathbb{R}^{n\times p}$*, the* K*-conjugate projector* P *defined by (10.3),* $\mathsf{Q} = \mathsf{I} - \mathsf{P}$*;* n*-vectors* $\mathbf{g}^0$*,* ℓ*,* $\mathbf{u}^0 = \mathsf{P}\mathsf{K}^{-1}\mathbf{f}$*,* $\mathbf{z}^0 = P_{\Upsilon_I}(\mathbf{g}^0)$*;*
$\Upsilon = \{\mathbf{z} : \mathbf{z}_{\mathcal{I}} \geq \ell_{\mathcal{I}}\}$*,* $\Gamma > 0$*,* $\overline{\alpha} \in (0, \|\mathsf{K}\mathsf{Q}\|^{-1}]$*, and* $\epsilon > 0$*.*
Step 0. {*Initialization.*}
Set $k = 0$*,* $\mathbf{g} = \mathsf{K}\mathsf{Q}\mathbf{z}^0 + \mathbf{g}^0$*,* $\mathbf{p} = \varphi(\mathbf{z}^0)$
while $\|\mathbf{g}^P(\mathbf{z}^k)\| > \epsilon$
if $\|\beta(\mathbf{z}^k)\|^2 \leq \Gamma\ \widetilde{\varphi}(\mathbf{z}^k)^T\varphi(\mathbf{z}^k)$
Step 1. {*Proportional* $\mathbf{z}^k$*. Trial conjugate gradient step.* }
$\alpha_{cg} = \mathbf{g}^T\mathbf{p}/\mathbf{p}^T\mathsf{K}\mathsf{Q}\mathbf{p}$*,* $\mathbf{y} = \mathbf{z}^k - \alpha_{cg}\mathbf{p}$
$\alpha_f = \max\{\alpha : \mathbf{z}^k - \alpha\mathbf{p} \in \Upsilon\} = \min\{(\ell_i - z_i^k)/p_i : p_i < 0\}$
if $\alpha_{cg} \leq \alpha_f$
Step 2. {*Conjugate gradient step.*}
$\mathbf{z}^{k+1} = \mathbf{y}$*,* $\mathbf{g} = \mathbf{g} - \alpha_{cg}\mathsf{K}\mathsf{Q}\mathbf{p}$
$\gamma = \varphi(\mathbf{y})^T\mathsf{K}\mathsf{Q}\mathbf{p}/\mathbf{p}^T\mathsf{K}\mathsf{Q}\mathbf{p}$*,* $\mathbf{p} = \varphi(\mathbf{y}) - \gamma\mathbf{p}$
end *if*
else
Step 3. {*Expansion step.*}
$\mathbf{z}^{k+\frac{1}{2}} = \mathbf{z}^k - \alpha_f\mathbf{p}$*,* $\mathbf{g} = \mathbf{g} - \alpha_f\mathsf{K}\mathsf{Q}\mathbf{p}$
$\mathbf{z}^{k+1} = P_{\Upsilon_I}(\mathbf{z}^{k+\frac{1}{2}} - \overline{\alpha}\varphi(\mathbf{z}^{k+\frac{1}{2}}))$
$\mathbf{g} = \mathsf{K}\mathsf{Q}\mathbf{z}^{k+1} + \mathbf{g}^0$*,* $\mathbf{p} = \varphi(\mathbf{z}^{k+1})$
end else
end if
else
Step 4. {*Proportioning step.*}
$\mathbf{d} = \beta(\mathbf{z}^k)$*,* $\alpha_{cg} = \mathbf{g}^T\mathbf{d}/\mathbf{d}^T\mathsf{K}\mathsf{Q}\mathbf{d}$
$\mathbf{z}^{k+1} = \mathbf{z}^k - \alpha_{cg}\mathbf{d}$*,* $\mathbf{g} = \mathbf{g} - \alpha_{cg}\mathsf{K}\mathsf{Q}\mathbf{d}$*,* $\mathbf{p} = \varphi(\mathbf{z}^{k+1})$
end else
$k = k + 1$
end while
Step 5. {*Return solution.*}
$\widehat{\mathbf{x}} = \mathbf{u}^0 + \mathsf{Q}\mathbf{z}^k$

Preconditioning Effect

As we have seen above, the iterations of Algorithm 12 may be considered as the conjugate gradient iterations for minimization of

$$f_{0,\mathsf{Q}}(\mathbf{z}) = \frac{1}{2}\mathbf{z}^T\mathsf{Q}^T\mathsf{K}\mathsf{Q}\mathbf{z} - (\mathbf{g}^0)^T\mathbf{z}$$

that generate iterations

$$\mathbf{z}^k \in \mathcal{K}^k(\mathsf{Q}^T\mathsf{K}\mathsf{Q}, \mathbf{g}^0) \subseteq \mathsf{K}\mathcal{V}.$$

Thus only the positive definite restriction $\mathsf{Q}^T\mathsf{K}\mathsf{Q}|\mathsf{K}\mathcal{V}$ of $\mathsf{Q}^T\mathsf{K}\mathsf{Q}$ to $\mathsf{K}\mathcal{V}$ takes part in the process of solution, and the rate of convergence can be estimated by the spectral

condition number $\kappa(\mathsf{Q}^T\mathsf{KQ}|\mathsf{K}\mathcal{V})$ of $\mathsf{Q}^T\mathsf{KQ}|\mathsf{K}\mathcal{V}$. For convenience of the reader, first we rehearse the estimate that were used in [10].

It is rather easy to see that

$$\kappa(\mathsf{Q}^T\mathsf{KQ}|\mathsf{K}\mathcal{V}) \le \kappa(\mathsf{K}).$$

Indeed, denoting the eigenvalues of K by $\lambda_1 \ge \cdots \ge \lambda_n$, we can observe that if $\mathbf{u} \in \mathsf{K}\mathcal{V}$, $\|\mathbf{u}\| = 1$, then by Lemma 5

$$\mathbf{u}^T\mathsf{Q}^T\mathsf{KQ}\mathbf{u} \ge (\mathsf{Q}\mathbf{u})^T\mathsf{K}(\mathsf{Q}\mathbf{u})/\|\mathsf{Q}\mathbf{u}\|^2 \ge \lambda_n$$

and

$$\mathbf{u}^T\mathsf{Q}^T\mathsf{KQ}\mathbf{u} \le \mathbf{u}^T\mathsf{Q}^T\mathsf{KQ}\mathbf{u} + \mathbf{u}^T\mathsf{P}^T\mathsf{KP}\mathbf{u} = \mathbf{u}^T\mathsf{K}\mathbf{u} \le \lambda_1. \tag{10.8}$$

To see the preconditioning effect of the algorithm in more detail, let us denote by $\mathcal{E}$ the p-dimensional subspace spanned by the eigenvectors corresponding to the p smallest eigenvalues $\lambda_{n-p+1} \ge \cdots \ge \lambda_n$, and let $\mathsf{R}_{\mathsf{K}\mathcal{U}}$ and $\mathsf{R}_{\mathcal{E}}$ denote the orthogonal projectors on $\mathsf{K}\mathcal{U}$ and $\mathcal{E}$, respectively. Let

$$\overline{\gamma} = \|\mathsf{R}_{\mathsf{K}\mathcal{U}} - \mathsf{R}_{\mathcal{E}}\|$$

denote the *gap* between $\mathsf{K}\mathcal{U}$ and $\mathcal{E}$. It can be evaluated effectively provided we have matrices U and E whose columns form orthogonal bases of $\mathsf{K}\mathcal{U}$ and $\mathcal{E}$, respectively. It is known [42] that if σ is the least singular value of $\mathsf{U}^T\mathsf{E}$, then

$$\overline{\gamma} = \sqrt{1-\sigma^2} \le 1.$$

Theorem 13. *Let $\mathcal{U}$ and Q be those of Algorithm 12, $\mathcal{V} = \mathrm{Im}\mathsf{Q}$. Then*

$$\kappa(\mathsf{Q}^T\mathsf{KQ}|\mathsf{K}\mathcal{V}) \le \frac{\lambda_1}{\sqrt{(1-\overline{\gamma}^2)\lambda_{n-m}^2 + \overline{\gamma}^2\lambda_n^2}}.$$

Proof. See [10]. □

The above theorem suggests that $\mathcal{U}$ should approximate the subspace spanned by the eigenvectors which correspond to the smallest eigenvalues of K. If $\mathsf{U}^T\mathsf{E}$ is nonsingular and $\lambda_n < \lambda_{n-m}$, then $\overline{\gamma} < 1$ and

$$\kappa(\mathsf{Q}^T\mathsf{KQ}|\mathsf{K}\mathcal{V}) < \kappa(\mathsf{K}).$$

Now we are ready to prove the following theorem.

Theorem 14. *Let $\Gamma > 0$ be a given constant, let $\overline{\alpha} \in (0, \|\mathsf{KQ}\|^{-1}]$, and let $\{\mathbf{z}^k\}$ denote the sequence generated by Algorithm 12 for the problem (10.6) starting from $\mathbf{z}^0 = P_\Upsilon(\mathbf{g}^0)$. Let us denote*

$$f_{0,\mathsf{Q}}(\mathbf{z}) = \frac{1}{2}\mathbf{z}^T\mathsf{Q}^T\mathsf{KQ}\mathbf{z} - (\mathbf{g}^0)^T\mathbf{z}.$$

Then

$$f_{0,\mathsf{Q}}(\mathbf{z}^{k+1}) - f_{0,\mathsf{Q}}(\widehat{\mathbf{z}}) \le \overline{\eta}_\Gamma \left(f_{0,\mathsf{Q}}(\mathbf{z}^k) - f_{0,\mathsf{Q}}(\widehat{\mathbf{z}})\right), \tag{10.9}$$

where $\widehat{\mathbf{z}}$ *denotes the unique solution of* (7.2) *and*

$$\overline{\eta}_\Gamma = 1 - \frac{\overline{\alpha}\overline{\lambda}_{\min}}{2 + 2\widehat{\Gamma}^2} \tag{10.10}$$

with $\widehat{\Gamma} = \max\{\Gamma, \Gamma^{-1}\}$. *Moreover, if* $\mathsf{U}^T\mathsf{E}$ *is nonsingular and* $\lambda_n < \lambda_{n-m}$, *then*

$$\overline{\eta}_\Gamma = 1 - \frac{\overline{\alpha}\overline{\lambda}_{\min}}{2 + 2\widehat{\Gamma}^2} < 1 - \frac{\overline{\alpha}\lambda_{\min}}{2 + 2\widehat{\Gamma}^2} = \eta_\Gamma, \tag{10.11}$$

where $\lambda_{\min}$ *denotes the smallest eigenvalue of* K.

Proof. It is enough to apply Theorem 8 with the given bounds above on the spectrum of $\mathsf{Q}^T\mathsf{K}\mathsf{Q}|\mathsf{K}\mathcal{V}$. □

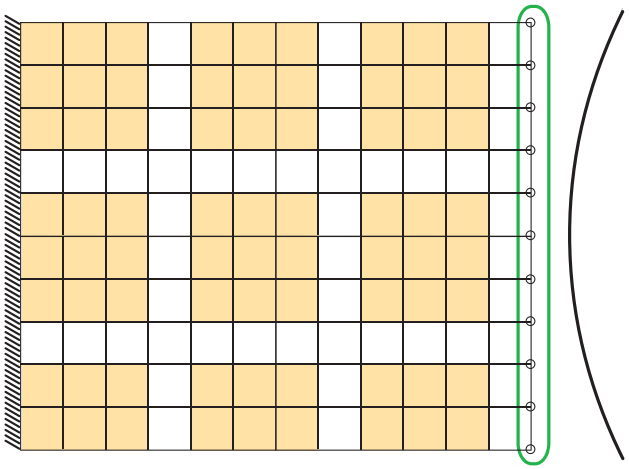


Figure 10.3: The basis functions are only used away from the contact boundary.

10.3 Projector in combination with FETI-DP method

We use an approach introduced in [21]. Let $\mathsf{F} \in \mathbb{R}^{m\times m}$ be a symmetric positive definite matrix. A projector P is an F-*conjugate projector* or briefly a *conjugate projector* if $\mathrm{Im}\mathsf{P}$ is F-conjugate to $\mathrm{Ker}\mathsf{P}$, or equivalently

$$\mathsf{P}^T\mathsf{F}(\mathsf{I} - \mathsf{P}) = \mathsf{P}^T\mathsf{F} - \mathsf{P}^T\mathsf{F}\mathsf{P} = \mathsf{O}.$$

If $\mathcal{U}$ is the subspace spanned by the columns of a full column rank matrix $\mathsf{U} \in \mathbb{R}^{m\times p}$, then

$$\mathsf{P} = \mathsf{U}(\mathsf{U}^T\mathsf{F}\mathsf{U})^{-1}\mathsf{U}^T\mathsf{F} \tag{10.12}$$

is a conjugate projector onto $\mathcal{U}$. Let $\Upsilon_0 = \{\lambda \in \mathbb{R}^m : \lambda \geq \mathbf{o}\}$. We use the conjugate projectors P and $\mathsf{Q} = \mathsf{I} - \mathsf{P}$ to decompose the dual minimization problem (9.8) into

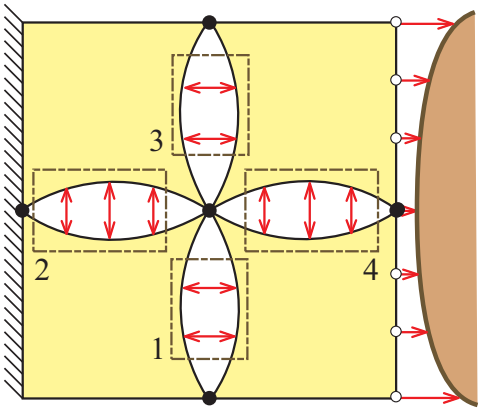


Figure 10.4: Aggregated Lagrange multipliers.

the minimization on $\mathcal{U}$ and the minimization on $\mathcal{V} \cap \Upsilon_0$, $\mathcal{V} = \mathrm{Im}\mathsf{Q}$, we write

$$\begin{aligned}
\min_{\lambda \geq 0} \Theta(\lambda) &= \min_{\substack{\eta \in \mathcal{U}, \mu \in \mathcal{V} \\ \eta + \mu \in \Upsilon_0}} \Theta(\eta + \mu) = \min_{\eta \in \mathcal{U}} \Theta(\eta) + \min_{\mu \in \mathcal{V} \cap \Upsilon_0} \Theta(\mu) \\
&= \Theta(\lambda^0) + \min_{\mu \in \mathcal{V} \cap \Upsilon_0} \Theta(\mu) = \Theta(\lambda^0) + \min_{\substack{\mu \in \mathsf{A}\mathcal{V} \\ \mu \geq 0}} \frac{1}{2} \mu^T \mathsf{Q}^T \mathsf{F} \mathsf{Q} \mu - \mathbf{d}^T \mathsf{Q} \mu \\
&= \Theta(\lambda^0) + \min_{\substack{\mu \in \mathsf{A}\mathcal{V} \\ \mu \geq 0}} \frac{1}{2} \mu^T \mathsf{Q}^T \mathsf{F} \mathsf{Q} \mu + \mu^T \mathbf{g}^0,
\end{aligned}$$

where $\lambda^0 = \mathsf{F}\mathsf{P}^{-1}\mathbf{d}$ and $\mathbf{g}^0 = -\mathsf{Q}^T\mathbf{d}$. The solution $\widehat{\lambda}$ of the dual problem (9.8) can then be expressed as $\widehat{\lambda} = \lambda^0 + \mathsf{Q}\widehat{\mu}$.

In this approach the matrix U is defined by the elements of the aggregation bases such as those depicted in Figure 10.4. Aggregated variables are Lagrange multipliers that enforce continuity conditions of primal displacement variables of two adjoining subdomains. The matrix U for the problem depicted in Figure 10.4 has the form

$$\mathsf{U} = \left[\begin{array}{cccc}
1 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 \\
0 & 0 & 0 & 1 \\
0 & 0 & 0 & 1 \\
\hline
 & \vdots & & \\
0 & 0 & 0 & 0 \\
 & \vdots & &
\end{array}\right]. \tag{10.13}$$

11 Transformation of basis

Since the application of transformation of basis to contact problems is straightforward, we only introduce the theoretical estimate comparing the transformation of basis with the preconditioning by conjugate projector. This result was presented in [21].

Theoretical estimate

Theorem 15. *The iterates generated by projector preconditioning with aggregated Lagrange multipliers are in the same space as the iterates generated by the algorithm using an explicit change of basis.*

Proof. The dual problem of (7.2) in the form that is convenient for our analysis reads as follows

$$\min_{\lambda \geq 0} \Theta(\lambda), \quad \Theta(\lambda) = \frac{1}{2}\lambda^T \mathsf{F}\lambda - \lambda^T \mathbf{d}, \tag{11.1}$$

where $\mathsf{F} = -\mathsf{B}\widetilde{\mathsf{K}}^{-1}\mathsf{B}^T$ and $\mathbf{d} = -\mathsf{B}\widetilde{\mathsf{K}}^{-1}\mathbf{f} + \mathbf{c}$ and the primal solution $\mathbf{u}$ has the form

$$\mathbf{u} = \widetilde{\mathsf{K}}^{-1}(\mathbf{f} - \mathsf{B}^T\lambda).$$

Using preconditioning by conjugate projector to solve the dual problem (11.1), the solution splits into

$$\lambda = \lambda^0 + \mathsf{Q}\mu,$$

where $\lambda^0 = \mathsf{P}\mathsf{F}^{-1}\mathbf{d}$.

Let us compute the primal solution related to λ^0, thus

$$\begin{aligned} \mathbf{u}_0 &= \widetilde{\mathsf{K}}^{-1}(\mathbf{f} - \mathsf{B}^T\lambda^0) = \widetilde{\mathsf{K}}^{-1}\Big(\mathbf{f} - \mathsf{B}^T\mathsf{P}\mathsf{F}^{-1}(-\mathsf{B}\widetilde{\mathsf{K}}^{-1}\mathbf{f} + \mathbf{c})\Big) \\ &= \widetilde{\mathsf{K}}^{-1}\mathbf{f} + \widetilde{\mathsf{K}}^{-1}\mathsf{B}^T\mathsf{P}\mathsf{F}^{-1}\mathsf{B}\widetilde{\mathsf{K}}^{-1}\mathbf{f} - \widetilde{\mathsf{K}}^{-1}\mathsf{B}^T\mathsf{P}\mathsf{F}^{-1}\mathbf{c} \\ &= \Big(\mathsf{I} + \widetilde{\mathsf{K}}^{-1}\mathsf{B}^T\mathsf{P}\mathsf{F}^{-1}\mathsf{B}\Big)\widetilde{\mathsf{K}}^{-1}\mathbf{f} - \widetilde{\mathsf{K}}^{-1}\mathsf{B}^T\mathsf{P}\mathsf{F}^{-1}\mathbf{c}. \end{aligned} \tag{11.2}$$

Using the definitions of the projector $\mathsf{P} = \mathsf{U}(\mathsf{U}^T\mathsf{F}\mathsf{U})^{-1}\mathsf{U}^T\mathsf{F}$, the obstacle $\mathbf{c}$ in (9.2) and the matrix U in (10.13), respectively, we have get

$$\begin{aligned}
\mathsf{U}^T\mathsf{B}\mathbf{u}_0 &= \mathsf{U}^T\mathsf{B}\Big(\mathsf{I} + \widetilde{\mathsf{K}}^{-1}\mathsf{B}^T\mathsf{P}\mathsf{F}^{-1}\mathsf{B}\Big)\widetilde{\mathsf{K}}^{-1}\mathbf{f} - \mathsf{U}^T\mathsf{B}\widetilde{\mathsf{K}}^{-1}\mathsf{B}^T\mathsf{P}\mathsf{F}^{-1}\mathbf{c} \\
&= \Big(\mathsf{U}^T\mathsf{B} + \mathsf{U}^T\underbrace{\mathsf{B}\widetilde{\mathsf{K}}^{-1}\mathsf{B}^T}_{-\mathsf{F}}\mathsf{P}\mathsf{F}^{-1}\mathsf{B}\Big)\widetilde{\mathsf{K}}^{-1}\mathbf{f} - \mathsf{U}^T\underbrace{\mathsf{B}\widetilde{\mathsf{K}}^{-1}\mathsf{B}^T}_{-\mathsf{F}}\mathsf{P}\mathsf{F}^{-1}\mathbf{c} \\
&= \Big(\mathsf{U}^T\mathsf{B} - \mathsf{U}^T\mathsf{B}\Big)\widetilde{\mathsf{K}}^{-1}\mathbf{f} + \mathsf{U}^T\mathbf{c} = \mathsf{U}^T\mathbf{c} = \mathsf{U}_E^T\mathbf{o} - \mathsf{O}_{\mathcal{I}}^T\ell_{\mathcal{I}} = 0. \qquad (11.3)
\end{aligned}$$

Let us now compute the primal solution related to an iterate λ

$$\mathbf{u} = \widetilde{\mathsf{K}}^{-1}(\mathbf{f} - \mathsf{B}^T\lambda) = \widetilde{\mathsf{K}}^{-1}\Big(\mathbf{f} - \mathsf{B}^T\lambda^0 - \mathsf{B}^T\mathsf{Q}\mu\Big) = \mathbf{u}_0 - \widetilde{\mathsf{K}}^{-1}\mathsf{B}^T\mathsf{Q}\mu$$

Furthermore we get

$$\begin{aligned}
-\mathsf{U}^T\underbrace{\mathsf{B}\widetilde{\mathsf{K}}^{-1}\mathsf{B}^T}_{-\mathsf{F}}\mathsf{Q}\mu &= \mathsf{U}^T\mathsf{F}\mathsf{Q}\mu = \mathsf{U}^T(\mathsf{F} - \mathsf{F}\mathsf{P})\mu \\
&= \mathsf{U}^T\mathsf{F}\mu - \mathsf{U}^T\mathsf{F}\mathsf{U}(\mathsf{U}^T\mathsf{F}\mathsf{U})^{-1}\mathsf{U}^T\mathsf{F}\mu = 0. \qquad (11.4)
\end{aligned}$$

From (11.3) and (11.4) we have for $\mathbf{u}$ related to λ that

$$\mathsf{U}^T\mathsf{B}\mathbf{u} = \mathsf{U}^T\mathsf{B}\mathbf{u}_0 - \mathsf{U}^T\mathsf{B}\widetilde{\mathsf{K}}^{-1}\mathsf{B}^T\mathsf{Q}\mu = 0 - 0 = 0.$$

Thus, the primal solution $\mathbf{u}^{\{k\}}$ related to the k-th iterate λ^k has continuous averages across the interface described by $\mathsf{U}^T\mathsf{B}\mathbf{u}^{\{k\}}$.

This shows that all iterates of FETI-DP with projector preconditioning are indeed in the space $\widetilde{W}$ (coarse problem subspace) [29, 44].

□

Corollary 16. *FETI-DP algorithm using average constraints and a transformation of basis has the same bounds on the rate of convergence as FETI-DP algorithm with preconditioning by conjugate projector introduced in Theorem 14.*

Remarks

- In the case of contact problems we used orthogonalized transformation matrix, obtained from (6.8) by QR factorization.
- Let us compare two variants, see (6.5) and (6.8), of the construction of the transformation matrix. For illustration we consider two 5×5 matrices, representing both strategies. Let us denote them by A_i and B_i, respectively, where the index i denotes the position of average, i. e. for $i = 5$ we have

$$\mathsf{A}_5 = \begin{bmatrix} 1 & & & & 1 \\ & 1 & & & 1 \\ & & 1 & & 1 \\ & & & 1 & 1 \\ -1 & -1 & -1 & -1 & 1 \end{bmatrix} \quad \text{and} \quad \mathsf{B}_5 = \begin{bmatrix} 1 & & & & \\ & 1 & & & \\ & & 1 & & \\ & & & 1 & \\ -1 & -1 & -1 & -1 & 1 \end{bmatrix}.$$

Let us denote by A_i^O and B_i^O, respectively, their orthogonalized versions obtained by QR factorization. Now, we are prepared to compare matrices A_1^O, A_3^O and A_5^O, where

$$\mathsf{A}_1^O = \begin{bmatrix} -0.447 & 0.707 & 0.408 & 0.289 & -0.224 \\ -0.447 & -0.707 & 0.408 & 0.289 & -0.224 \\ -0.447 & 0 & -0.816 & 0.289 & -0.224 \\ -0.447 & 0 & 0 & -0.866 & -0.224 \\ -0.447 & 0 & 0 & 0 & 0.894 \end{bmatrix}$$

$$\mathsf{A}_3^O = \begin{bmatrix} -0.707 & 0.408 & -0.447 & 0.289 & -0.224 \\ 0 & -0.816 & -0.447 & 0.289 & -0.224 \\ 0.707 & 0.408 & -0.447 & 0.289 & -0.224 \\ 0 & 0 & -0.447 & -0.866 & -0.224 \\ 0 & 0 & -0.447 & 0 & 0.894 \end{bmatrix}$$

$$\mathsf{A}_5^O = \begin{bmatrix} -0.707 & 0.408 & 0.289 & 0.224 & 0.447 \\ 0 & -0.816 & 0.289 & 0.224 & 0.447 \\ 0 & 0 & -0.866 & 0.224 & 0.447 \\ 0 & 0 & 0 & -0.894 & 0.447 \\ 0.707 & 0.408 & 0.289 & 0.224 & 0.447 \end{bmatrix}$$

Obviously, we can write

$$\mathsf{A}_1^O = \mathsf{P}_i \mathsf{A}_3^O \mathsf{P}_i \sim \mathsf{P}_j \mathsf{A}_5^O \mathsf{P}_j,$$

where P_i denotes a suitable permutation matrix and $\sim$ admits some differences in the signs. We also can compare B_1^O, B_3^O and B_3^O, where

$$\mathsf{B}_1^O = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathsf{B}_3^O = \begin{bmatrix} -0.707 & 0.408 & 0.577 & 0 & 0 \\ 0 & -0.816 & 0.577 & 0 & 0 \\ 0.707 & 0.408 & 0.577 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathsf{B}_5^O = \begin{bmatrix} -0.707 & 0.408 & 0.289 & 0.224 & 0.447 \\ 0 & -0.816 & 0.289 & 0.224 & 0.447 \\ 0 & 0 & -0.866 & 0.224 & 0.447 \\ 0 & 0 & 0 & -0.894 & 0.447 \\ 0.707 & 0.408 & 0.289 & 0.224 & 0.447 \end{bmatrix}$$

We observe that $\mathsf{B}_1^O = \mathsf{I}$. In the case of B_3^O we see that there is no average for all 5 variables. The "average" in the 3th column is only for the first 3 variables. And for the matrix B_5^O we have $\mathsf{A}_5^O = \mathsf{B}_5^O$.

These observations illustrate that in combination with orthogonalization it is better to use transformation matrix based on the form A_i. It seems, that it does not matter where the average is situated. So, in our experiments we use transformation matrix $\mathsf{T}_E = \mathsf{A}_i$ with the average in arbitrary edge node.

III

Numerical experiments

Projector preconditioning 12

First, we show the action of the projector on the string and then, on the model problem from the second part, we illustrated the improvement of the rate of convergence, when the projector preconditioning is applied to the finite element discretization problem.

12.1 One dimensional problem

To demonstrate the work of the conjugate projector visually, let us consider one dimensional problem, where no obstacle is considered. This can be interpreted as a displacement of the string fixed on both ends.

Figure 12.1, where we have aggregated 8 neighbouring nodes, illustrate approximation of the solution $\mathbf{u}_0$ in the subspace $\mathcal{U}$, approximation of the solution $\mathsf{Q}\mathbf{u}$ in the subspace $\mathcal{V}$ and the solution $\mathbf{u}$.

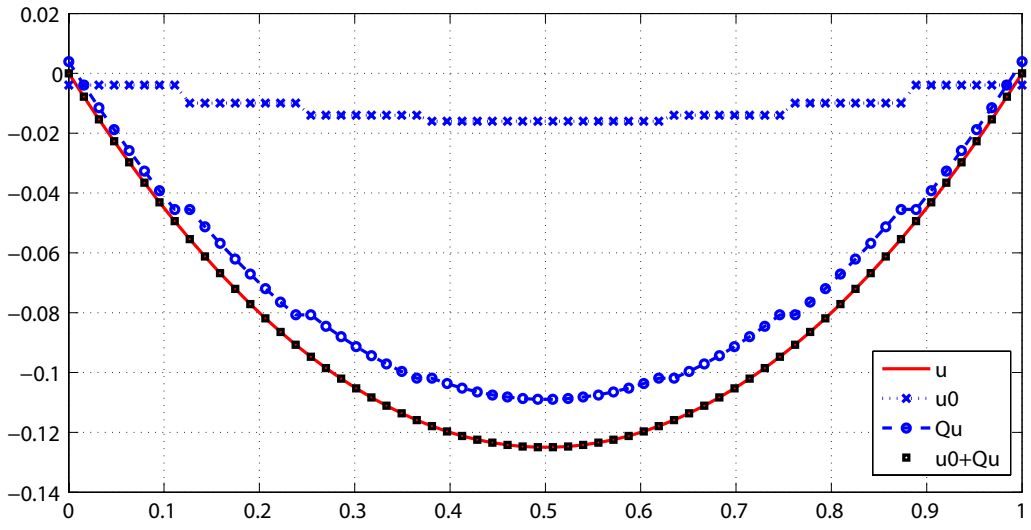


Figure 12.1: Splitting of the solution $\mathbf{u}$: $\mathbf{u}_0 \in \mathcal{U}$ and $\mathsf{Q}\mathbf{u} \in \mathcal{V}$.

We have solved the problem with changing number of unknowns (*dof*) and the fixed aggregation size equal to 8. The results are in Table 12.1, where *iter in* $\mathbb{R}^n$ denotes the number of iteration of the CG algorithm, *numA* denotes the number of aggregations and *iter in* $\mathcal{V}$ denotes the number of conjugate gradient iterations with projector preconditioning (CGCP). We can see that the number of CGCP is constant. The algorithm has a numerical scalability.

n	iter in $\mathbb{R}^n$	numA	iter in $\mathcal{V}$
32	15	4	14
64	31	8	14
128	63	16	14
256	127	32	14
512	255	64	14
1024	511	128	14

Table 12.1: Scalability of aggregations with fixed agr-size = 8.

12.2 Displacement of membrane

The model problem considered here was introduced in Chapter 7, and is depicted in Figure 12.2.

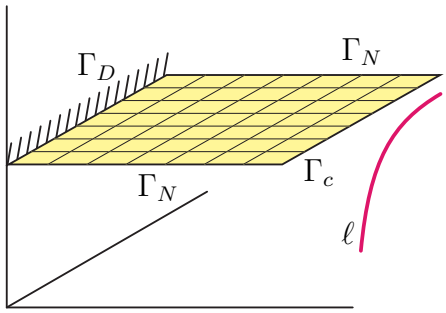


Figure 12.2: Two-dimensional problem with the Dirichlet boundary condition on Γ_D and the homogeneous Neumann boundary condition elsewhere (on Γ_N).

The obstacle l on the contact boundary Γ_c was defined by the upper part of the circle with the radius $R = 1$ and the center $S = (1; 0.5; -1.3)$. We have discretized this problem by the piece-wise finite elements using regular grids defined by the discretization parameter h. After the discretization, we got the problem of the form (7.2). The computations were performed with MPRGP algorithm (Algorithm 7) and MPRGP-CP (Algorithm 12) the parameters $\overline{\alpha} = \|\mathsf{K}\|^{-1}$ and $\Gamma = 1$. The stopping criterion was $\|\mathbf{g}^P(x)\| \leq 10^{-8}\|\mathbf{f}\|$.

Projector defined by aggregations

In the first set of our numerical experiments, we have used the projectors defined by aggregations as depicted in Figure 12.3, see also Section 10.2 and 4.3.

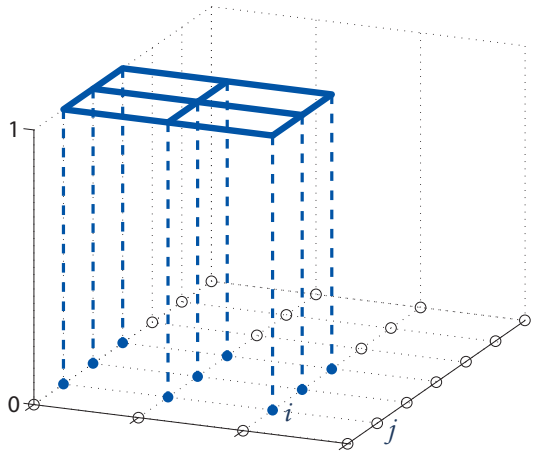


Figure 12.3: Aggregation basis function.

We have carried out the computations with the number of unknows $dof \in \{323, 115, 4355, 16899, 66563\}$ and a fixed size of aggregations $sizeA = 8$, see Figure 12.4, and with changing parameter $sizeA \in \{8, 16, 32, 64\}$ and fixed $dof = 16899$. The results are in Table 12.2, where *contact* denotes the number of the constrained variables. After that follows the numbers of iterations and the time in seconds that was necessary to get the solution, and in Table 12.3, respectively.

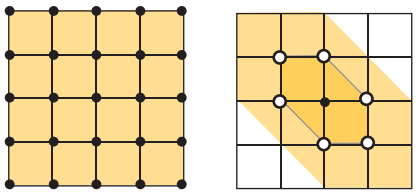


Figure 12.4: Illustration of the parameter sizeA = 4.

		No projector		Projector	
dof	contact	iter	sec	iter	time
323	17	130	0.026	80	0.041
1155	33	236	0.056	105	0.089
4355	65	511	0.327	134	0.246
16899	129	997	2.379	180	1.309
66563	257	2471	26.921	275	9.398

Table 12.2: Projectors defined by aggregations. Counts of iterations for increasing number of variables and fixed $sizeA = 8$.

Figure 12.5 illustrates that number of iterations varies moderately with increasing dimension of the problem, so we observe a numerical scalability in the computations. In spite of poor approximation properties of the aggregation bases, our results show that it can considerably reduce both the number of iterations and the time of the solution.

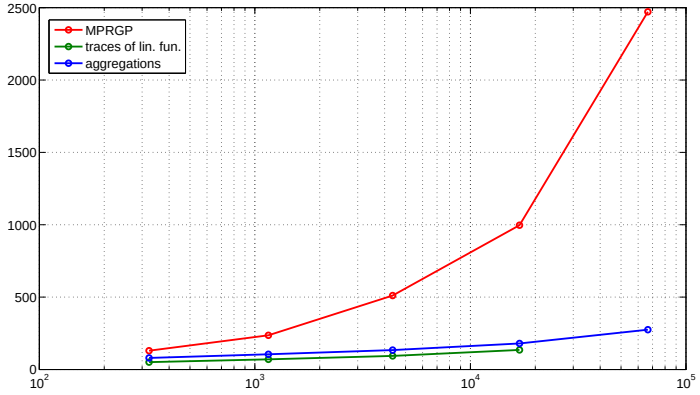


Figure 12.5: Scalability of aggregations and piece-wise linear functions with fixed $sizeA = 8$.

	Traces of lin. fun.		Aggregations	
sizeA	iter	time	iter	time
8			180	8.317
16	148	7.001	242	10.730
32	184	8.369	351	15.911
64	277	13.198	614	28.315
128	433	20.153		

Table 12.3: Iteration counts for changing sizeA and fixed $dof = 16899$.

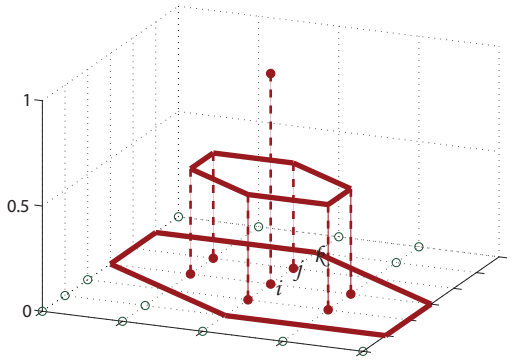


Figure 12.6: Coarse linear basis function.

Projector defined by the traces of linear functions on a coarse grid

We used also the projectors defined by the traces of linear functions on the coarse grids such as depicted in Figure 12.6, see Section 10.2 and 4.3.

The results of numerical experiments are in Table 12.4, where the notation has the same meaning as in Table 12.2. In this case, we fixed the size of basis functions $sizeA = 8$, see Figure 12.4. The results for changing parameter $sizeA$ are in Table 12.3. We see that the number of iterations is in general smaller than that for the aggregations, but the iterations are more costly. The picture can change in different implementation.

		No projector		Projector	
dof	contact	iter	sec	iter	time
323	17	130	0.137	51	0.171
1155	33	236	0.440	70	0.288
4355	65	511	3.176	94	1.311
16899	129	997	20.303	135	6.422

Table 12.4: Projectors defined by the traces of linear functions. Iteration counts for increasing number of variables and fixed $sizeA = 8$.

13 Transformation of basis vs. projector preconditioning

In this chapter we compare two methods exploiting the edge averages to improve the performace: projector preconditioning and transformation of basis.

Let us consider the model problem from Chapter 7. The obstacle $\mathbf{c}$ on the contact boundary Γ_c was defined by the upper part of the circle with the radius $R = 1$ and the center $S = (1; 1; -1.3)$.

The problem was solved with piece-wise linear finite elements and established in the form using primal displacement variables also on the contact boundary described in [20]. It was solved by the MPRGP algorithm (Algorithm 7) and its modification MPRGP-CP (Algorithm 12), where preconditioning by conjugate projector was used. We note, that for the numerical experiments with a transformation of basis we used the orthogonal transformation matrix, see Remarks in Chapter 11.

Computational results, presented by Jarošová, Klawonn, and Rheinbach in [21], are shown in Tables 13.1 and 13.2. Iteration counts for changing number of subdomains and H/h = 8 are shown in Table 13.2 and iteration counts for 4×4 subdomains and changing H/h are shown in Table 13.1.

	FETI-DP		
H/h	*no preconditioned*	*proj. preconditioning*	*trans. of basis*
4	32	22	22
8	51	30	31
16	82	41	42
32	118	58	61

Table 13.1: Iteration counts of dual problem for 4×4 subdomains and changing H/h.

	FETI-DP		
num. of sub.	*no preconditioned*	*proj. preconditioning*	*trans. of basis*
4×4	51	30	31
8×8	79	34	35
12×12	91	46	45
16×16	101	52	51
20×20	118	58	57

Table 13.2: Iteration counts of dual problem for changing number of subdomains and H/h = 8.

We have the same results in terms of iteration counts for a) FETI-DP using edge constraints implemented by using a transformation of basis and b) FETI-DP enforcing edge constraints by projector preconditioning.

FETI-DP averages for linear elasticity contact problems

14

The benchmark considered here is a variant of the 2D Hertz problem [9, 43] of pressure distribution between an elastic cylinder and an elastic half-space in mutual contact. The geometry of the problem is in Figure 14.1.

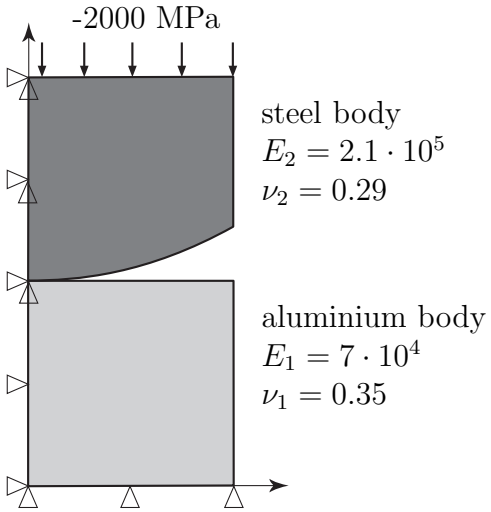


Figure 14.1: Geometry and traction.

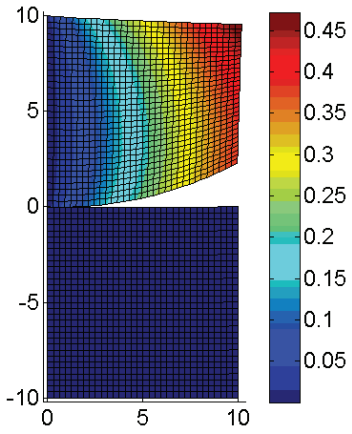


Figure 14.2: Solution.

The upper body is loaded by traction -2000 [Mpa] along the top edge. The material constants are defined by the Young modulus $E_1 = 7 \cdot 10^4$ [MPa] and Poisson's ratio $\nu_1 = 0.35$ for the lower aluminium body and $E_2 = 2.1 \cdot 10^5$ [MPa] and $\nu_2 = 0.29$ for the upper steel body. The lower body is fixed in horizontal direction along the vertical boundary and in vertical direction along the bottom. The upper body is fixed in both directions along the vertical boundary. The nonpenetration condition was imposed between the bodies and the plain strain was assumed.

	sub	coarse	primal	dual	iter	time
left	3	48	2916	393	46	2.278
middle	3	48	2916	393	46	2.200
right	3	48	2916	393	46	2.184

Table 14.1: The computational results for the averages placed to arbitrary edge nodes. Notation *left, middle, right* corresponds to Figure 14.3.

In Table 14.1 we show the results for the averages placed to arbitrary edge nodes. We can compare the same results for all three variants depicted in Figure 14.3.

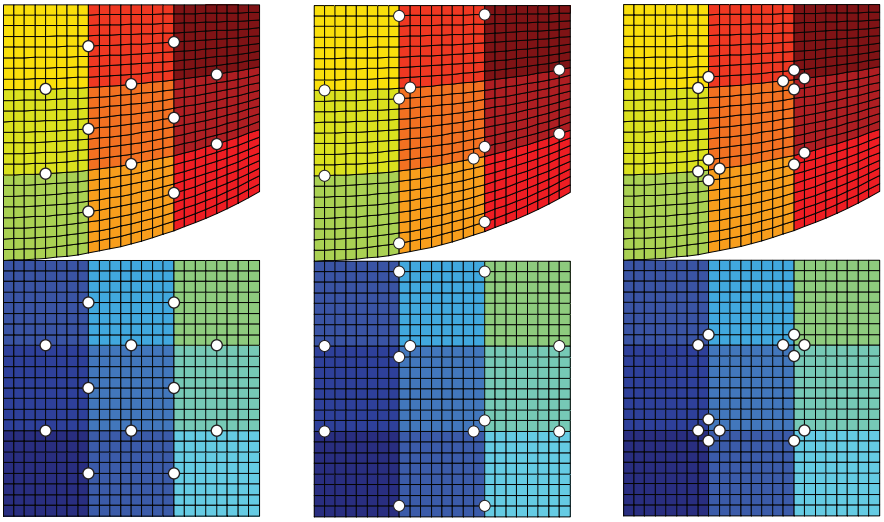
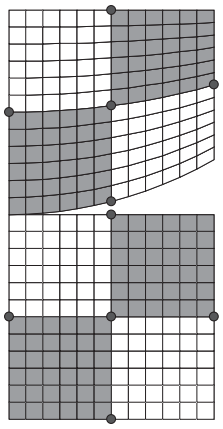


Figure 14.3: The averages (white nodes) placed to arbitrary edge nodes.

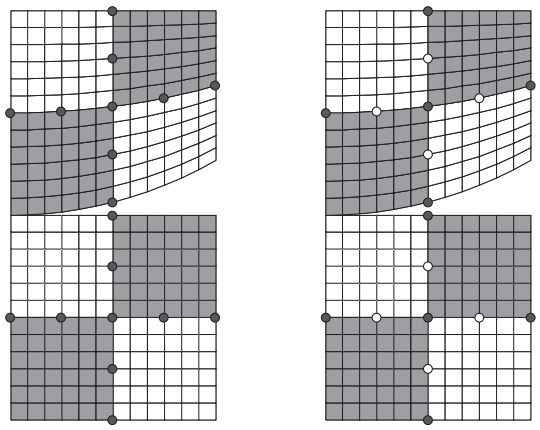
We have carried out the computations with the decomposition parameter $H \in \{1/2, 1/4, 1/8, 1/12, 1/16\}$ and with the fixed discretization parameter $h = 1/16$. The stopping criterion was $\|\mathbf{g}^P(x)\| \leq 10^{-6}\|\mathbf{f}\|$. The results, for different positions of primal displacement variables (coarse problem nodes), are shown in Tables 14.2–14.5 along with the figures illustrating the tested strategies. In the tables, *1/H* denotes the number of subdomains in one direction of one body, *coarse* denotes the number of coarse problem nodes, *primal* denotes the total number of degrees of freedom, *dual* denotes the number of dual displacement variables, then there is the number of iterations and time in seconds that was necessary to get the solution.



Strategy A

1/H	coarse	primal	dual	iter/time
2	20	4624	273	50/2
4	84	18496	1505	82/9
8	308	73984	6849	119/104
12	660	166464	16033	144/837
16	1140	295936	29057	169/3246

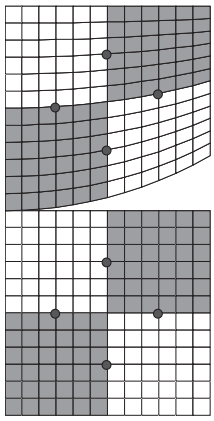
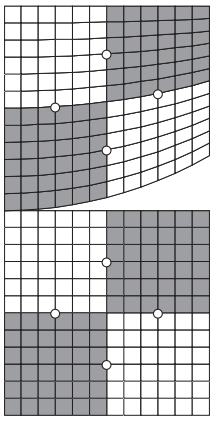
Table 14.2: The coarse problem nodes (black nodes) were introduced in the vertices. This can be assumed to be the standard FETI-DP method.



Strategy B **Strategy C**

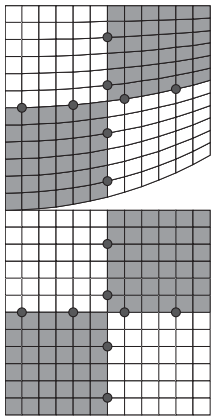
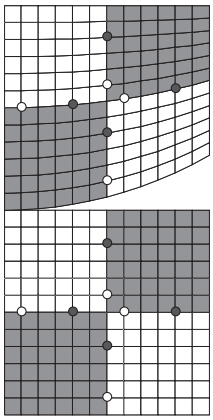
1/H	coarse	primal	dual	iter/time	iter/time
2	36	4624	257	35/1	24/1
4	180	18496	1409	44/5	28/4
8	756	73984	6401	60/59	36/40
12	1716	166464	14977	83/544	37/262
16	3060	295936	27137	84/1701	40/924

Table 14.3: The coarse problem nodes (black and white nodes) were introduced in the vertices as well as in the middle of the edges. The averages (white nodes) were placed in the middle of the edges when the transformation of basis was assumed.

				Strategy D	**Strategy E**
1/H	coarse	primal	dual	iter/time	iter/time
2	16	4624	285	70/1	47/2
4	96	18496	1565	111/10	67/8
8	448	73984	7101	289/204	136/125
12	1056	166464	16605	466/2213	158/842
16	1920	295936	30077	659/9719	253/4462

Table 14.4: The coarse problem nodes (black and white nodes) were introduced in the middle of the edges, as well as the averages (white nodes) were placed when the transformation of basis was assumed.

				Strategy F	**Strategy G**
1/H	coarse	primal	dual	iter/time	iter/time
2	32	4624	269	49/1	42/1
4	192	18496	1469	54/7	48/7
8	896	73984	6653	75/67	64/80
12	2112	166464	15549	88/538	68/385
16	3840	295936	28157	93/1878	86/1961

Table 14.5: Two coarse problem nodes (black and white nodes) per edge were assumed. The averages (white nodes) were placed in one of them when the transformation of basis was assumed.

Total FETI

15

Just to compare the results of numerical experiments we briefly describe the variant of classical FETI method called Total FETI (TFETI), introduced independently by Dostál, Horák, and Kučera [7] and Of (all floating FETI) [36].

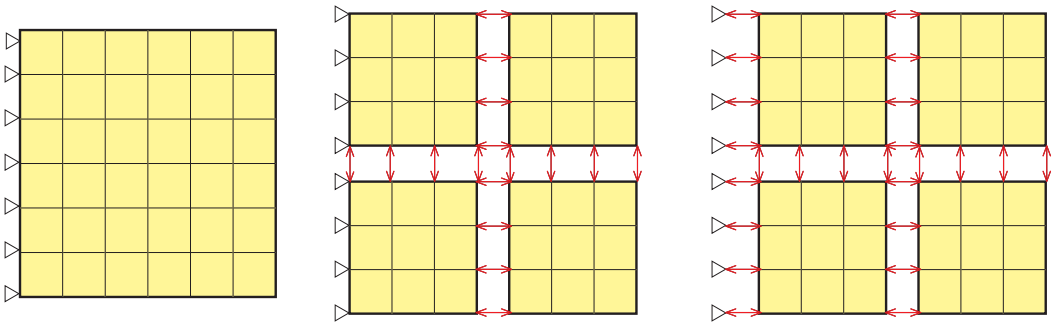


Figure 15.1: Original domain is decomposed for FETI (middle) and TFETI (right) method.

We use the description of Dostál [11]. The TFETI method differs from the original FETI method in the way which is used to implement the Dirichlet boundary conditions. While the FETI method assumes that the subdomains inherit the Dirichlet boundary conditions from the original problem, TFETI uses the Lagrange multipliers to "glue" the subdomains to the boundary whenever the Dirichlet boundary conditions are prescribed, see Figure 15.1. Such approach simplifies the implementation as all the stiffness matrices of the subdomains have typically a priori known kernels and can be treated in the same way. For more details about TFETI we refer to [7, 9, 8]. This method is also implemented in our MatSol library.

1/H	primal	dual	iter	time
4	18496	1921	57	6
8	73984	8065	76	61
12	166464	18433	103	185
16	295936	33025	129	396

Table 15.1: TFETI method: Iteration counts for 2D Hertz problem.

The results of numerical experiments on 2D Hertz problem, described in Chapter 14, are in Table 15.1, where *1/H* denotes number of subdomains in one direction of one body, *primal* denotes total number of degrees of freedom, *dual* denotes number of dual displacement variables, then there is number of iteration and time in seconds that was necessary to get the solution.

Let us compare the numerical experiments for FETI-DP and T-FETI. Since the FETI-DP need some time to preprocessing steps, we can see the differences in the time requirements. On other hand, the iteration counts for FETI-DP (e.g. Strategy F and G) are smaller than for T-FETI. It could be interesting to try to combine both approaches.

Conclusions 16

In this thesis two preconditioning strategies, which result in the improved bounds on the rate of convergence, were considered. The first one was a preconditioning by conjugate projector, where in combination with FETI-DP, the Lagrange multipliers corresponding to the variables of the coinciding edges were aggregated. The second one was an explicit transformation of basis, where certain edge averages were introduced as new, additional primal variables.

For a special case, it was shown that both methods iterate in the same space and thus have the same rate of convergence. This is an important result, since the explicit construction of the projector can be replaced by the transformation of basis which works locally and can be easily parallelized. Let us recall that until recently, there were no results on the rate of convergence for the problems with inequality constraints in the terms of bounds on the spectrum of the Hessian. Even the classical result by O'Leary [37] on preconditioning in face does not guarantee any improvement.

The future development comprises two parts. The first one is the convergence analysis of FETI-DP with variants of averaging, but without preconditioning. It will comprise of analysis of the linear and nonlinear model elasticity problems with the goal to find the theoretical support for the results without preconditioning. The second one, numerical experiments, will comprise development and implementation of the algorithms for the problems from 2D and 3D linear elasticity, application of standard preconditioners for linear iterations, formulation and implementation of academic benchmarks for contact problems with focus on scalability, and demonstration of the performance of the algorithms on real world problems.

Bibliography

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

[2] R. Barrett, M. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. Van der Vorst. *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods, 2nd Edition.* SIAM, Philadelphia, PA, 1994.

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

[4] M. Domorádová and Z. Dostál. Projector preconditioning for partially bound-constrained quadratic optimization. *Numerical Linear Algebra with Applications*, 14:791–806, 2007.

[5] Z. Dostál. Box constrained quadratic programming with proportioning and projections. *SIAM Journal on Optimization*, 7:871–887, 1997.

[6] Z. Dostál. An optimal algorithm for bound and equality constrained quadratic programming problems with bounded spectrum. *Computing (Springer)*, 78:311–328, 2006.

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

[8] Z. Dostál, T. Kozubek, V. Vondrák, T. Brzobohatý, and A. Markopoulos. Scalable TFETI algorithm for the solution of multibody contact problems of elasticity. *International Journal for Numerical Methods in Engineering.*

[9] Z. Dostál, T. Kozubek, V. Vondrák, T. Brzobohatý, and A. Markopoulos. A scalable TFETI based algorithm for 2d and 3d frictionless contact problems. *In Proceedings of LSSC 7. Ed. S. Margenov, Berlin: Springer, LNCS*, 5910:94–102, 2010.

[10] Zdeněk Dostál. Conjugate gradient method with preconditioning by projector. *Intern. J. Computer Math.*, 23:315–323, 1988.

[11] Zdeněk Dostál. *Optimal Quadratic Programming Algorithms, with Applications to Variational Inequalities. 1st edition.* Springer US, New York, 2009.

[12] Zdeněk Dostál, David Horák, and Dan Stefanica. A scalable FETI-DP algorithm for a coercive variational inequality. *Appl. Numer. Math.*, 54(3–4):378–390, 2005.

[13] Zdeněk Dostál, David Horák, and Dan Stefanica. A scalable feti-dp algorithm for semi-coercive variational inequalities. *Computer Methods in Applied Mechanics and Engineering*, 196:1369–1379, 2007.

[14] Zdeněk Dostál and Joachim Schöberl. Minimizing quadratic functions subject to bound constraints with the rate of convergence and finite termination. *Comput. Optim. Appl.*, 30(1):23–43, 2005.

[15] Charbel Farhat, Michel Lesoinne, Patrick LeTallec, Kendall Pierson, and Daniel Rixen. FETI-DP: A dual-primal unified FETI method - part i: A faster alternative to the two-level FETI method. *Internat. J. Numer. Methods Engrg.*, 50:1523–1544, 2001.

[16] Charbel Farhat, Michel Lesoinne, and Kendall Pierson. A scalable dual-primal domain decomposition method. *Numer. Lin. Alg. Appl.*, 7:687–714, 2000.

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

[18] Charbel Farhat and Francois-Xavier Roux. A method of Finite Element Tearing and Interconnecting and its parallel solution algorithm. *Int. J. Numer. Meth. Engrg.*, 32:1205–1227, 1991.

[19] Magnus R. Hestenes and Eduard Stiefel. Methods of conjugate gradients for solving linear systems. *Journal of Research of the National Bureau of Standards*, 49(6):409–436, December 1952.

[20] David Horák. *FETI based domain decomposition methods for variational inequalities*. PhD thesis, VŠB-TU Ostrava, Czech Republic, 2007.

[21] M. Jarošová, A. Klawonn, and O. Rheinbach. Projector preconditioning and transformation of basis in FETI-DP algorithms for contact problems. *Mathematics and Computers in Simulations*, 2009. Accepted for publication.

[22] A. Klawonn and O. Rheinbach. Highly scalable parallel domain decomposition methods with an application to biomechanics. *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift fr Angewandte Mathematik und Mechanik*, 90:5–32, 2009.

[23] Axel Klawonn and Olof B.Widlund. Dual and dual-primal FETI methods for elliptic problems with discontinuous coefficients in three dimensions. In *Twelfth International Conference on Domain Decomposition Methods, Chiba, Japan*, 2001.

[24] Axel Klawonn, Luca F. Pavarino, and Oliver Rheinbach. Spectral element FETI-DP and BDDC preconditioners with multi-element subdomains. *Comput. Meth. Appl. Mech. Engrg.*, 198(3–4):511–523, 2008.

[25] Axel Klawonn and Oliver Rheinbach. A parallel implementation of Dual-Primal FETI methods for three dimensional linear elasticity using a transformation of basis. *SIAM J. Sci. Comput.*, 28(5):1886–1906, 2006.

[26] Axel Klawonn and Oliver Rheinbach. Robust FETI-DP methods for heterogeneous three dimensional elasticity problems. *Comput. Methods Appl. Mech. Engrg.*, 196(8):1400–1414, 2007.

[27] Axel Klawonn, Oliver Rheinbach, and Olof B. Widlund. An analysis of a FETI-DP algorithm on irregular subdomains in the plane. *SIAM J. Numer. Anal.*, 46(5):2484–2504, 2008.

[28] Axel Klawonn and Olof B. Widlund. Selecting constraints in dual-primal FETI methods for elasticity in three dimensions. In Ralf Kornhuber, Ronald H. W. Hoppe, Jacques Périaux, Olivier Pironneau, Olof B. Widlund, and Jinchao Xu, editors, *Proceedings of the 15th international domain decomposition conference*, pages 67–81. Springer, 2003.

[29] Axel Klawonn and Olof B. Widlund. Dual-primal FETI methods for linear elasticity. *Communications on pure and applied mathematics*, 59(11):1523–1572, 2006.

[30] Axel Klawonn, Olof B. Widlund, and Maksymilian Dryja. Dual-Primal FETI methods for three-dimensional elliptic problems with heterogeneous coefficients. *SIAM J.Numer. Anal.*, 40:159–179, 2002.

[31] Jing Li and Olof B. Widlund. FETI-DP, BDDC, and block cholesky methods. *International journal for numerical methods in engineering*, 66:250–271, 2006.

[32] Jan Mandel and Radek Tezaur. On the convergence of a dual-primal substructuring method. *Numer. Math.*, 88:543–558, 2001.

[33] G.I. Marchuk and Yu.A. Kuznetsov. Theory and applications of the generalized conjugate gradient method. *Adv. Math., Suppl. Stud.*, 10:153–167, 1986.

[34] R. A. Nicolaides. Deflation of conjugate gradients with applications to boundary value problems. *SIAM J. Numer. Anal.*, 24(2):355–365, 1987.

[35] Jorge Nocedal and Stephen Wright. *Numerical Optimization.* Springer Series in Operations Research. Springer, 2nd edition, 2006.

[36] G. Of. *BETI - Gebietszerlegungsmethoden mit schnellen Randelementverfahren und Anwendungen.* PhD thesis, University of Stuttgart, Germany, 2006.

[37] Dianne P. O'Leary. A generalised conjugate gradient algorithm for solving a class of quadratic programming problems. *Linear Algebra and its Applications*, 34:371–399, 1980.

[38] Oliver Rheinbach. *Parallel Scalable Iterative Substructuring: Robust Exact and Inexact FETI-DP Methods with Applications to Elasticity.* PhD thesis, Department of Mathematics, University of Duisburg-Essen, Essen, Germany, 2006.

[39] Oliver Rheinbach. Parallel iterative substructuring in structural mechanics. *Journal Archives of Computational Methods in Engineering*, 16(4):425–463, 2009.

[40] Yousef Saad. *Iterative Methods for Sparse Linear Systems.* Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2nd edition, 2003.

[41] J. Schoberl. Solving the signorini problem on the basis of domain decomposition techniques. *Computing*, 60:323–344, 1998.

[42] G. W. Stewart. Error and perturbation bounds for subspace associated with certain eigenvalue problems. *SIAM Review*, 15:727–763, 1973.

[43] S.P. Timoshenko and J.N. Goodier. *Theory of Elasticity.* McGraw-Hill, New York, 1970.

[44] Andrea Toselli and Olof Widlund. *Domain Decomposition Methods - Algorithms and Theory*, volume 34 of *Springer Series in Computational Mathematics.* Springer, 2004.

[45] O.C. Zienkiewicz, R.L. Taylor, and J.Z. Zhu. *The finite element method: its basis and fundamentals; 6th ed.* Elsevier, Amsterdam, 2005.