Domain Decomposition with Subdomain CCG for Material Jump Elliptic Problems

* Institute of Numerical Mathematics
  Russian Academy of Sciences
  Gubkina st, 8
  Moscow 117333
  Russia

Preprint SC 97–36 (October 1997)
Abstract

A combination of the cascadic conjugate gradient (CCG) method for homogeneous problems with a non-overlapping domain decomposition (DD) method is studied. Mortar finite elements on interfaces are applied to permit non-matching grids in neighboring subdomains. For material jump problems, the method is designed as an alternative to the cascadic methods.
1 Introduction

In this paper we consider linear elliptic problems

$$\nabla(a \nabla u) + cu = f$$

on general domains with space dimension $p$ equal to 2 or 3, where typically the coefficient $a$ is strongly varying. This type of problems arises whenever different materials are combined. Standard (multiplicative) multigrid methods [13] or additive multilevel methods such as KASKADE with BPX preconditioners [12, 20, 9] deal quite efficiently with such a situation - apart from certain pathological examples in 3D. However, the recently developed cascadic multigrid methods such as CCG [11, 8, 7, 6], which are extremely fast for homogeneous problems, tend to exhibit some slow-down whenever the material jumps are “too strong”. Moreover, adaptive gridding in such a problem may lead to unnecessarily fine grid regions on both sides of interfaces between “fine grid” and “coarse grid” subdomains. In order to overcome the described undesirable effects, the present paper studies the combination of CCG on homogeneous subdomains with non-overlapping domain decomposition (DD) methods. Mortar finite elements for interfaces are applied to permit non-matching grids in neighboring subdomains.

For the thus arising saddle point problem, we follow an approach proposed by Kuznetsov in [16, 15] where a theory of constructing block diagonal preconditioners has been elaborated. The convergence rate of a two-level iterative scheme proposed by this author is independent of grid steps, the subdomain diameters and the jumps in coefficients. Up to now, results have been obtained for the case of linear preconditioners. We will generalize these results for the case of nonlinear preconditioners.

The outline of the paper is as follows. In Section 2 we pose the problem and give its non-conforming finite element discretization [5, 4, 3, 1, 16]. In Section 3 we develop the two types of necessary preconditioners, one for the subdomains (the CCG part) and one for the interfaces (the DD part). In Section 4 we formulate a two-term gradient method with the variable-step preconditioner $H$. The preconditioner $H$ is considered as an arbitrary continuous mapping $v \rightarrow H^{-1}[v]$. The convergence of the outer iterative process holds, if the mapping satisfies the coercivity assumption [2]. In our theoretical analysis, we use the CCG method for the subdomain problems and a preconditioned two-term gradient method for the interface problem. For the inner iterations, the termination strategy due to [11, 8] is selected. The optimal arithmetical complexity of the two-level iterative algorithm for the case of quasi-uniform grids immediately follows from results of [16] and [8] provided that a sufficient regularity of subdomain problems is assured (Section 5.1). In addition, we derive an a–posteriori termination strategy for the adaptive algorithm (Section 5.2). This strategy utilizes spectral properties of the preconditioned problem and a heuristic assumption on the family of adaptive triangulations. Finally, in Section 6, comparative computations on a material jump test problem from the literature are given, which involve further outer iterative schemes like PCG, GMRES, and PCR.
2 Mortar finite element method

Let $\Omega \in \mathbb{R}^p$, $p=2,3$, be a polygonal Lipschitz domain. We consider an elliptic Dirichlet problem: find $u \in H_0^1(\Omega)$ such that
\begin{equation}
\int_{\Omega} (a \nabla u \nabla v + cuv) \, d\Omega = \int_{\Omega} f v \, d\Omega \quad \forall v \in H_0^1(\Omega),
\end{equation}
where $f \in H^{-1}(\Omega)$, $a$ is a uniformly positive bounded function, $c$ is a nonnegative bounded function. In this paper both functions are assumed to be piecewise smooth ones.

We specify a non-overlapping partitioning $\Omega$ into simply connected subdomains $\Omega_k$ with characteristic diameters $d_k$:
\[ \bar{\Omega} = \bigcup_{k=1}^m \bar{\Omega}_k. \]

Let $\Omega_{kh}$ be a triangular ($p=2$) or tetrahedral ($p=3$) regular covering of $\Omega_k$ with a discretization parameter $h_k$. Without loss of generality we assume that $\Omega_{kh} \equiv \Omega_k$. For methods of overcoming the difficulties associated with curvilinear boundaries see, e.g. [10].

Let $\Gamma_{kl}$ be the interface between subdomains $\Omega_k$ and $\Omega_l$, $\Gamma_{k0} = \partial \Omega_k \cap \partial \Omega$, and $\Gamma_{klh}$ be the trace of $\Omega_{kh}$ at $\Gamma_{kl}$. We assume that $\Gamma_{kl}$ is a simply connected set. Note that the traces of grids $\Omega_{kh}$ and $\Omega_{lh}$ at the interface $\Gamma_{kl}$, generally speaking, do not coincide. These grids are called non-matching grids.

We denote by $V_{kh}$ the piecewise linear finite element subspace of $H_1^1(\Omega_k)$ defined by the grid $\Omega_{kh}$ and by $W_{klh}$ the trace of $V_{kh}$ at each of the interfaces $\Gamma_{kl}$, $k, l = 0, \ldots, m$.

**Definition 2.1** The space $\Lambda_{klh}$ will be called the mortar finite element space associated with $\Gamma_{kl}$, $l < k$, if it satisfies the two requirements:
\begin{align*}
(a) & \quad \inf_{\lambda_h \in \Lambda_{klh}} \| \mu - \lambda_h \|_{H^{-1/2}(\Gamma_{kl})} \leq \hat{c}_1 h_k \| \mu \|_{H^{1/2}(\Gamma_{kl})} \quad \forall \mu \in H^{1/2}(\Gamma_{kl}), \\
(b) & \quad \inf_{\lambda_h \in \Lambda_{klh}} \sup_{v_h \in W_{kh} \cap H^1_0(\Gamma_{kl})} \left( \frac{\| \lambda_h \|_{L^2(\Gamma_{kl})} \| v_h \|_{L^2(\Gamma_{kl})}}{\| \lambda_h \|_{L^2(\Gamma_{kl})}} \right) \geq \hat{c}_2,
\end{align*}
with positive constants $\hat{c}_1$ and $\hat{c}_2$ independent of $h_k$.

The first requirement means that the mortar space is a consistent approximation of $H^{-1/2}(\Gamma_{kl})$. The second requirement means that $\Lambda_{klh}$ must be small compared to the space $W_{kh}$. In practice, the mortar space is constructed as a subspace of $W_{kh}$ as follows. The space $W_{kh}$ is decomposed into a direct sum
\[ W_{kh} = W^{(i)}_{kh} \oplus W^{(b)}_{kh}, \]
where functions in $W^{(b)}_{kh}$ vanish at inner nodes of $\Gamma_{kh}$. Let $\{ \varphi_k^{(i)} \}_{k=1}^{n_{kl}^{(i)}}$ and $\{ \varphi_k^{(b)} \}_{k=1}^{n_{kl}^{(b)}}$ be the nodal basis functions of $W^{(i)}_{kh}$ and $W^{(b)}_{kh}$, respectively. Then the mortar finite element space $\Lambda_{klh}$ is defined as the linear span of the functions
\[ \psi_k^{(i)} = \varphi_k^{(i)} + \sum_{l=1}^{n_{kl}^{(b)}} \alpha_{kl} \varphi_l^{(b)}, \quad k = 1, \ldots, n_{kl}^{(i)}, \]
where \( \{\alpha_{kl}\} \) are coefficients. The choice of the coefficients \( \{\alpha_{kl}\} \) is specified by the two requirements [16]:

\[
\begin{align*}
(a) & \ 1 \in \Lambda_{klh}; \\
(b) & \ \lambda_h^{(i)} \geq (\lambda_h^{(i)}, \lambda_h^{(i)})
\end{align*}
\]

for any function \( \lambda_h \in \Lambda_{klh} \), \( \lambda_h = \lambda_h^{(i)} + \lambda_h^{(b)} \), where \( \lambda_h^{(i)} \in W_{klh}^{(i)} \) and \( \lambda_h^{(b)} \in W_{klh}^{(b)} \).

The first requirement is again an approximation requirement. The second requirement gives the inf-sup condition from Definition 2.1 with \( \hat{c}_2 = 0.5 \). The difference between basis functions of \( \Lambda_{klh} \) and \( \Lambda_{kl} \) is illustrated by Fig. 2.1. The construction of the mortar finite element space in the three-dimensional case is not so obvious and we refer to paper [16] for details.

![Figure 2.1. The nodal basis functions of \( \Lambda_{klh} \) (left) and \( W_{klh} \) (right).](image)

Using the above notations, we consider the finite element problem with Lagrange multipliers at the interfaces \( \Gamma_{kl} \), \( l < k, k = 1, \ldots, m \): find \( (u_h, \lambda_h) \in X_h \) such that

\[
\begin{align*}
& a(u_h, v_h) + b(\lambda_h, v_h) = f(v_h), \\
& b(\mu_h, u_h) = 0
\end{align*}
\]

for all \( (v_h, \mu_h) \in X_h \). Here

\[
\begin{align*}
& a(u, v) = \sum_{k=1}^{m} a_k(u, v), \quad a_k(u, v) = \int_{\Omega_k} [a \nabla u \nabla v + cuv] \, d\Omega, \\
& b(\lambda, v) = \sum_{\{k \neq l\} / |\Gamma_{kl}| \neq 0} b_{kl}(\lambda, v), \quad b_{kl}(\lambda, v) = \int_{\Gamma_{kl}} \lambda_{kl} v_k \, ds, \\
& f(v) = \sum_{k=1}^{m} f_k(v), \quad f_k(v) = \int_{\Omega_k} f v \, d\Omega,
\end{align*}
\]

where \( \lambda_{kl} = -\lambda_{lk} \) at \( l > k \),

\[
V_h = \prod_{k=1}^{m} V_{kh}, \quad \Lambda_h = \prod_{l < k \atop |\Gamma_{kl}| \neq 0} \Lambda_{klh} \quad \text{and} \quad X_h = V_h \times \Lambda_h.
\]

The error analysis of the finite element problem was done in [17, 3, 5, 4] for the case of quasi-uniform grids. Assume that the exact solution \( u \) is a sufficiently smooth function, e.g. \( u \in H^1_0(\Omega) \cap \prod_{k=1}^{m} H^2(\Omega_k) \). Then the optimal error estimate holds [17] in spite of non-conformity:

\[
\sum_{k=1}^{m} \| u - u_h \|_{H^1(\Omega_k)} \leq \hat{c} \sum_{k=1}^{m} h_k \left( \| u \|_{H^2(\Omega_k)}^2 + \sum_{l \neq k} \| \partial u / \partial n \|_{H^{1/2}(\Gamma_{kl})}^2 \right)^{1/2}.
\]
The finite element problem (2.2) results in a system of linear algebraic equations in the saddle point form:

\[
\begin{bmatrix} A_1 & B_1^T \\ \vdots & \vdots \\ A_m & B_m^T \\ B_1 & \ldots & B_m \end{bmatrix} \begin{bmatrix} u_1 \\ \vdots \\ u_m \\ \lambda \end{bmatrix} = \begin{bmatrix} f_1 \\ \vdots \\ f_m \\ 0 \end{bmatrix},
\]

where \( A_k \) is the symmetric \( n_k \times n_k \) positive semidefinite matrix, and \( B_k^T \) is the \( n_k \times n_\lambda \) mass matrix. Let \( N = n_\lambda + \sum_{k=1}^m n_k \) denote the problem size.

From the construction of \( \Lambda_{klh} \) we conclude that \([B_1 \ldots B_m]\) is a full rank matrix. Therefore problem (2.3) has a unique solution even if some matrices \( A_k \) are singular [16].

### 3  Block diagonal preconditioner

Without loss of generality we assume that functions \( a \) and \( c \) are piecewise constant, i.e. \( a = a_k > 0 \) and \( c = c_k > 0 \) in \( \Omega_k \). When the restriction \( c \) on \( \Omega_k \) is a zero function, the positive coefficient \( c_k \) may be introduced artificially. Clearly this coefficient must be sufficiently small so that it does not significantly affect the accuracy of finite element solutions. Finally we assume that \( c_k \leq \hat{c} a_k / d_k^2, \ k \geq 1 \), where constant \( \hat{c} \) is independent of \( a_k, c_k, d_k \) and \( h_k \).

Let us define a block diagonal matrix

\[
H = \begin{bmatrix} H_1 & \cdots & \cdots & H_m \\ \vdots & \ddots & \ddots & \vdots \\ \vdots & \cdots & \ddots & \vdots \\ \vdots & \cdots & \cdots & H_\lambda \end{bmatrix},
\]

**Lemma 3.1.** [14] Suppose that \( A_k, H_k, k = 1, \ldots, m, \) and \( H_\lambda \) are symmetric positive definite matrices. If there exist positive constants \( \eta_1, \eta_2, \theta_1 \) and \( \theta_2 \) independent of \( a_k, c_k, d_k \) and \( h_k \) such that

\[
\eta_1(H_kv_k, v_k) \leq (A_kv_k, v_k) \leq \eta_2(H_kv_k, v_k) \quad \forall v_k \in \mathbb{R}^{n_k}, \ k \geq 1,
\]

and

\[
\theta_1(H_\lambda \lambda, \lambda) \leq (\sum_{k=1}^m B_k A_k^{-1} B_k^T \lambda, \lambda) \leq \theta_2(H_\lambda \lambda, \lambda) \quad \forall \lambda \in \mathbb{R}^{n_\lambda},
\]

\]
then

\[ \gamma_1(Hx, x) \leq |(Ax, x)| \leq \gamma_2(Hx, x) \quad \forall x \in \mathbb{R}^N, \]

where

\[ \gamma_1 = \min \left\{ \eta_1, \frac{\sqrt{\eta_1^2 + 4\eta_1\theta_1} - \eta_1}{2} \right\} \quad \text{and} \quad \gamma_2 = \frac{\sqrt{\eta_1^2 + 4\eta_2\theta_2} + \eta_2}{2}. \]

An approach for constructing linear preconditioners \( H_k, k \geq 1 \), and \( H_\lambda \) was proposed in [16]. The scheme of discussions used by the author is easily generalized for the case of nonlinear preconditioners. We shall follow this scheme in the next subsections.

### 3.1 Subdomain preconditioners

We denote by \( \hat{A}_k \) the matrix \( A_k \) for \( a_k = 1 \), \( c_k = 0 \) and by \( M_k \) the matrix \( A_k \) for \( a_k = 0 \), \( c_k = 1 \). It is obvious that \( \hat{A}_k \) and \( M_k \) are stiffness and mass matrices corresponding to the Laplace operator. Let us consider an eigenvalue problem

\[ \hat{A}_k w = \nu M_k w \]

with eigenvalues

\[ 0 = \nu_{k1} < \nu_{k2} \leq \ldots \leq \nu_{kn_k} \]

and \( M_k \)-orthonormalized eigenvectors \( w_{k1}, \ldots, w_{kn_k} \).

The linear preconditioner \( H_k \) for the matrix \( A_k \) is defined by

\[ H_k^{-1} = \frac{1}{a_k} \left( \hat{A}_k + \frac{1}{d_k} M_k \right)^{-1} + \frac{1}{c_k} P_k \quad (3.1) \]

where

\[ P_k = w_{k1} w_{k1}^T \]

is the \( M_k \)-orthogonal projector on the kernel of \( \hat{A}_k \). In [16, 15] was shown that eigenvalues of the matrix \( H_k^{-1} A_k \) belong to the segment \([\eta_1, \eta_2]\), where \( \eta_1 \) and \( \eta_2 \) are positive constants independent of \( a_k, c_k, d_k \) and \( h_k, k \geq 1 \).

Assume that a linear system

\[ \hat{A}_k u_k \equiv (\hat{A}_k + \frac{1}{d_k^2} M_k) u_k = v_k \]

is solved approximately such that the relative error in the \( H_k^{-1} \)-norm between the exact \( H_k^{-1} v_k \) and approximate \( \hat{H}_k^{-1} v_k \) solutions is less than \( \varepsilon \), i.e.

\[ \|H_k^{-1} v_k - \hat{H}_k^{-1} v_k\|_{H_k} \leq \varepsilon \|H_k^{-1} v_k\|_{H_k} \quad \forall v_k \in \mathbb{R}^{n_k}. \quad (3.2) \]
Hereafter the notation $H_k^{-1}[v_k]$ means an arbitrary mapping $v_k \rightarrow H_k^{-1}[v_k]$ satisfying (3.2) with a sufficiently small positive constant $\varepsilon$. The particular choice of the mapping will be discussed in Section 4.2. Let a mapping $H_k^{-1}[v_k]$ be defined by

$$H_k^{-1}[v_k] \equiv \frac{1}{a_k} \circ H_k^{-1}[v_k] + \frac{1}{c_k} P_k v_k.$$ 

It is easily to check that the exact $H_k^{-1}v_k$ and approximate $H_k^{-1}[v_k]$ solutions satisfy an inequality analogous to (3.2) with the same constant $\varepsilon$:

$$\|H_k^{-1}v_k - H_k^{-1}[v_k]\|_{H_k} \leq \frac{1}{\sqrt{a_k}} \|H_k^{-1}v_k - H_k^{-1}[v_k]\|_{H_k} \leq \varepsilon \|H_k^{-1}v_k\|_{H_k}. \quad (3.3)$$

### 3.2 Interface preconditioner

We divide the grid nodes in each subdomain $\Omega_{kh}$ into two groups so that the first group contains the inner nodes of $\Omega_{kh}$, and the second group contains the nodes from $\partial \Omega_{kh}$. If the first group is equipped with the subscript $I$ and the second group with the subscript $\Gamma$, the matrices $A_k$ and $B_k$ can be represented in a more detailed form:

$$A_k = \begin{bmatrix} A_{Ik} & A_{I\Gamma_k} \\ A_{\Gamma I_k} & A_{\Gamma k} \end{bmatrix}, \quad B_k^T = \begin{bmatrix} 0 \\ B_{\Gamma k}^T \end{bmatrix}. \quad (3.4)$$

According to block partitioning (3.4) we have

$$S_\lambda \equiv \sum_{k=1}^m B_k A_k^{-1} B_k^T = \sum_{k=1}^m B_{\Gamma k} S_{\Gamma k}^{-1} B_{\Gamma k}^T$$

where

$$S_{\Gamma k} = A_{\Gamma k} - A_{\Gamma I_k} A_{I\Gamma_k}^{-1} A_{\Gamma k}, \quad k \geq 1.$$ 

We denote by $\hat{S}_{\Gamma k}$ the matrix $S_{\Gamma k}$ for $a_k = 1$, $c_k = 0$ and by $M_{\Gamma k}$ the corresponding mass matrix associated with a bilinear form

$$m_{\Gamma k}(u_k, v_k) = \int_{\partial \Omega_k} u_k v_k \, ds.$$ 

Let us introduce an auxiliary matrix

$$R_\lambda = \sum_{k=1}^m B_{\Gamma k} \left[ \frac{1}{a_k} (\hat{S}_{\Gamma k} + \frac{1}{d_k} M_{\Gamma k})^{-1} + \frac{1}{c_k d_k} P_{\Gamma k} \right] B_{\Gamma k}^T,$$

where $P_{\Gamma k}$ is the $M_{\Gamma k}$-orthogonal projector on the kernel of $\hat{S}_{\Gamma k}$. The matrix $R_\lambda$ is spectrally equivalent to $S_\lambda$ [16, 15]. Assume that a matrix $L_k$ is spectrally equivalent to the matrix
\( \hat{H}_k \) and has the block representation analogous to (3.4). Then the Schur complement 
\( \hat{S}_{\Gamma k} \equiv L_{\Gamma k} - L_{\Gamma k} L_{\Gamma k}^{-1} L_{\Gamma k} \) is spectrally equivalent to the matrix 
\( \hat{S}_{\Gamma k} + \frac{1}{d_k} M_{\Gamma k} \). We use this fact to construct an interface preconditioner

\[
H_\lambda = \sum_{k=1}^{m} B_{\Gamma k} \left[ \frac{1}{a_k} \hat{S}_{\Gamma k}^{-1} + \frac{1}{c_k d_k} P_{\Gamma k} \right] B_{\Gamma k}^T. \tag{3.5}
\]

In [16, 15] was shown that eigenvalues of the matrix \( H_\lambda^{-1} S_{\lambda} \) belong to the segment \([\theta_1, \theta_2]\), where \( \theta_1 \) and \( \theta_2 \) are positive constants independent of \( a_k, c_k, d_k \) and \( h_k \), \( k \geq 1 \). Since all assumptions of Lemma 3.1 are satisfied, matrices \( A \) and \( H \) are spectrally equivalent. In our method only the product \( H_\lambda \lambda \) has to be calculated. The procedure of finding the product \( y_{\Gamma k} = \hat{S}_{\Gamma k}^{-1} g_{\Gamma k} \) is reduced to a partial problem

\[
L_k \begin{bmatrix} y_{\Gamma k} \\ y_{\Gamma k} \end{bmatrix} = \begin{bmatrix} g_{\Gamma k} \\ 0 \end{bmatrix},
\]

where the nonzero entries of the right hand side and the required solution correspond to boundary nodes of \( \Omega_{kh} \). Therefore the multilevel methods (see, for example, [9, 21] and references therein) are the most appropriate choice for constructing \( L_k \). The arithmetical complexity of these methods is estimated as \( O(n_k^{(p-1)/p} \ln n_k) \).

Let \( \lambda \rightarrow H_\lambda^{-1}[\lambda] \) be an arbitrary mapping such that

\[
\|H_\lambda^{-1} \lambda - H_{\lambda_2}^{-1}[\lambda]\|_{H_\lambda} \leq \varepsilon \|H_\lambda^{-1} \lambda\|_{H_\lambda} \tag{3.6}
\]

is valid for \( \forall \lambda \in \mathbb{R}^{n_\lambda} \) and some sufficiently small positive constant \( \varepsilon \). The particular choice of this mapping will be discussed in Section 4.2.

## 4 Combination of DD and CCG

In this section, we study the combination of domain decomposition for the outer iteration and the cascadic conjugate gradient method for the inner iteration.

### 4.1 Outer iterative process: two–term gradient method

We are now in the position to define a mapping

\[
x \rightarrow H^{-1}[x], \quad x = (v_1, \ldots, v_m, \lambda)^T,
\]

as the direct sum of mappings \( H_k^{-1}[v_k], \) \( k \geq 1 \), and \( H_{\lambda}^{-1}[\lambda] \) defined above. The accuracy assumptions (3.3) and (3.6) give us the following estimate:

\[
\|H^{-1}[x] - H^{-1}x\|_H \leq \varepsilon \|H^{-1}x\|_H \quad \forall x \in \mathbb{R}^N. \tag{4.1}
\]
We again apply Lemma 3.1 and accuracy estimate (4.1) to obtain the coercivity:

\[ x_0 = 0, \]
\[ x_i = x_{i-1} - \tau_{i-1}H^{-1}[AH^{-1}[r_{i-1}]], \quad i = 1, 2, \ldots \]  
(4.2)

\[ \tau_i = \frac{(r_i, H^{-1}[r_i])}{(AH^{-1}[r_i], H^{-1}[AH^{-1}[r_i]])}, \]

where \( r_i = Ax_i - g \). This algorithm belongs to a class of methods with variable step preconditioners (see, for example, [2] and references therein). The peculiarity of our formulation is that the preconditioner \( H \) is given in the implicit form (3.1), (3.5).

**Lemma 4.2.** Let \( \varepsilon \) be a sufficiently small positive constant. Then the mapping \( x \to H^{-1}[AH^{-1}[Ax]] \) is coercive,

\[ (x, H^{-1}[AH^{-1}[Ax]])_H \geq \sigma_1(x, x)_H \quad \forall x \in \mathbb{R}^N, \]

and continuous,

\[ \|H^{-1}[AH^{-1}[Ax]]\|_H \leq \sigma_2\|x\|_H \quad \forall x \in \mathbb{R}^N, \]

with positive constants \( \sigma_1 \) and \( \sigma_2 \) independent of \( a_k, c_k, d_k \) and \( h_k, k = 1, \ldots, m \).

**Proof.** The triangle inequality and (4.1) give

\[ (1 - \varepsilon)\|H^{-1}x\|_H \leq \|H^{-1}[Ax]\|_H \leq (1 + \varepsilon)\|H^{-1}x\|_H. \]

We utilize the spectral equivalence of matrices \( A \) and \( H \) to obtain

\[ \|H^{-1}[Ax]\|_H \leq (1 + \varepsilon)\|H^{-1}Ax\|_H \leq \gamma_2(1 + \varepsilon)\|x\|_H. \]

By induction we have

\[ \|H^{-1}[AH^{-1}[Ax]]\|_H \leq \gamma_2(1 + \varepsilon)\|H^{-1}[Ax]\|_H \leq \gamma_2^2(1 + \varepsilon)^2\|x\|_H. \]

Therefore, the continuity of the mapping takes place with the constant \( \sigma_2 = \gamma_2^2(1 + \varepsilon)^2 \).

We again apply Lemma 3.1 and accuracy estimate (4.1) to obtain the coercivity:

\[ (x, H^{-1}[AH^{-1}[Ax]])_H = (x, H^{-1}AH^{-1}Ax)_H \]
\[ + (x, H^{-1}[AH^{-1}[Ax]] - H^{-1}AH^{-1}Ax)_H \]
\[ \geq \gamma_1^2\|x\|_H^2 - \varepsilon\|H^{-1}AH^{-1}[Ax]\|_H\|x\|_H \]
\[ \geq \gamma_1^2\|x\|_H^2 - \varepsilon\gamma_2\|H^{-1}[Ax]\|_H\|x\|_H \]
\[ \geq \gamma_1^2\|x\|_H^2 - \varepsilon\gamma_2^2(1 + \varepsilon)\|x\|_H^2. \]

Thus for a sufficiently small \( \varepsilon \), constants \( \sigma_1 = \gamma_1^2 - \varepsilon(1 + \varepsilon)\gamma_2^2 \) and \( \sigma_2 \) are positive and independent of \( a_k, c_k, d_k \) and \( h_k, k = 1, \ldots, m \). \( \square \)
Let us return back to the iterative process (4.2). It is well known [19] that the optimal choice of the iterative parameter is given by
\[ \tau^*_i = \left( H e_i, H^{-1}[AH^{-1}[Ae_i]] \right) \]
where both the preconditioner \( H \) and the iterative error \( e_i = x_i - x \) cannot be computed. The optimal choice \( \tau^*_i \) yields the steepest minimization of \( \| e_{i+1} \|_H \) at each step of the iterative process:
\[ \| e_{i+1} \|_H^2 = \| e_i \|_H^2 - \tau_i (2\tau^*_i - \tau_i) \| H^{-1}[AH^{-1}[Ae_i]] \|_H^2. \]
The arguments used in the proof of Lemma 4.2 enable us to evaluate the ratio of \( \tau_i \) and \( \tau^*_i \) as follows:
\[ \frac{1 - \varepsilon}{1 + \varepsilon} \leq \frac{\tau^*_i}{\tau_i} \leq \frac{\gamma_i^2 + \varepsilon \gamma_2^2}{\gamma_i^2 - \varepsilon \gamma_2^2} \frac{1 + \varepsilon}{1 - \varepsilon} \quad \text{for} \quad \varepsilon \leq \frac{1}{3}. \]
We apply the last inequalities and Lemma 3.2 to derive the following estimate
\[ \| x_i - x \|_H \leq \left[ 1 - \left( \frac{\sigma_1}{\sigma_2} \right)^2 \right]^{i/2} \| x_0 - x \|_H, \quad (4.3) \]
where \( \sigma_1 \) and \( \sigma_2 \) are the first order corrections to \( \sigma^*_1 \) and \( \sigma^*_2 \), respectively:
\[ \sigma_1 = \sigma^*_1 (1 - \varepsilon)(1 - \varepsilon \gamma_2^2 / \gamma_1^2), \quad \sigma_2 = \sigma^*_2 (1 + \varepsilon)(1 + \varepsilon \gamma_2^2 / \gamma_1^2). \]
There are several ways for improving the convergence rate of outer iterations. The first approach we shall follow in numerical experiments is to use the well known acceleration strategy [19]. The second way is to replace the two–term algorithm by either the generalized conjugate gradient method [2] or the generalized minimal residual method [18] or the preconditioned conjugate residual method. To complete the discussion of method (4.2), we give an estimate similar to (4.3) which holds in the case of exact inner iterations:
\[ \| x_i - x \|_H \leq \left[ 1 - \left( \frac{\gamma_1}{\gamma_2} \right)^4 \right]^{i/2} \| x_0 - x \|_H. \]

4.2 Inner iteration: CCG

In this subsection we apply the cascadic conjugate gradient method [11, 8] for constructing the mapping \( g_k \rightarrow H_k^{-1}[g_k] \). The subdomain problem to be solved can be given in a weak formulation: find \( u_h^k \in V_{kh} \) such that
\[ \int_{\Omega_k} \left[ \nabla u_h^k \nabla v + \frac{1}{d_k^2} u_h^k v \right] \, d\Omega = \int_{\Omega_k} g_h^k v \, d\Omega \quad \forall v \in V_{kh}. \]
We assume that the subdomain \( \Omega_k \) is convex and \( g_h^k \in L^2(\Omega_k) \), i.e. the elliptic problem is at least \( H^2 \)-regular. Suppose that the triangulation \( \Omega_{kh} \) induces a nested family of finite element spaces:

\[
V_{k0} \subset V_{k1} \subset \ldots \subset V_{kL} \equiv V_{kh}, \quad L \geq 1.
\]

For each \( j = 0, \ldots, L \) we introduce the finite element approximations \( u_{kj}^h \in V_{kj} \) as solutions of

\[
\int_{\Omega_k} [\nabla u_{kj}^h \nabla v + \frac{1}{d_k^2} u_{kj}^h v] \, d\Omega = \int_{\Omega_k} g_h^k v \, d\Omega \quad \forall v \in V_{kj}.
\tag{4.4}
\]

The cascadic conjugate gradient method deals with a sequence of linear systems of equations associated with finite element problems (4.4). On the level \( j = 0 \), which is assumed to have only a small number of degrees of freedom, the linear system is solved directly. On the next levels \( j \geq 1 \), linear systems are solved iteratively by the conjugate gradient method. Starting point for the CG iterations on the level \( j \) is the finite element solution from the previous level. Denoting this starting point by \( u_{kj}^* \), the CCG method can be formalized as follows

\[
u_{kj}^* = \mathcal{I}_{j,m_j} u_{kj-1}^*
\tag{4.5}
\]

where \( \mathcal{I}_{j,m_j} \) denotes \( m_j \) steps of the CG method on the level \( j \) and \( u_{k0}^* \equiv u_{k0}^h \).

Let \( \| \cdot \|_a \) denote the energy norm induced by the bilinear form in the left hand side of (4.4). The iterative error of the CCG method is estimated [8] by

\[
\|u_{kj}^h - u_{kj}^*\|_a \leq \hat{c} \frac{m_j}{m_{j-1}} \|u_{kj}^h - u_{kj-1}^h\|_a + \|u_{kj-1}^h - u_{kj-1}^*\|_a,
\tag{4.6}
\]

where the positive constant \( \hat{c} \) depends on the shape regularity of the triangulation \( \Omega_{kh} \). We get by induction

\[
\|u_{kL}^h - u_{kL}^*\|_a \leq \hat{c} \sum_{j=1}^L \frac{1}{m_j} \|u_{kj}^h - u_{kj-1}^h\|_a \leq \hat{c} \sum_{j=1}^L \frac{1}{m_j} \|u_{kL}^h\|_a.
\]

Thus, assumption (3.2) will be satisfied if parameters \( m_j \) belong to a geometrical progression. In order to evaluate the arithmetical complexity of the CCG method we assume additionally that numbers \( n_{kj} \) of degrees of freedom of \( V_{kj} \), \( j = 1, \ldots, L \), also belong to a geometrical progression:

\[
n_{k,j+1} \geq \beta_1 n_{kj}, \quad \beta_1 > 1.
\]

With the choice of the iteration number \( m_j \) on the level \( j \) as

\[
m_j = \left[ m_L \beta_2^{L-j} \right], \quad \beta_1 > \beta_2 > 1,
\]

we get the final error

\[
\|u_{kL}^h - u_{kL}^*\|_a \leq \hat{c} \frac{1}{m_L} \frac{1}{1 - 1/\beta_2} \|u_{kL}^h\|_a
\tag{4.7}
\]
and the arithmetical complexity

\[
\sum_{j=1}^{L} m_j n_{kj} \leq m_L n_{kL} \frac{1}{1 - \beta_2 / \beta_1}.
\]

In order to keep the convergence rate of outer iterations (4.2) below a reasonable value, the final error should be significantly small. The reasonable value of \( \varepsilon \) is implicitly defined by Lemma 3.2 and is independent of \( a_k, c_k, d_k \) and \( h_k \). Since \( \varepsilon \sim m_k^{-1} \), we have the optimal arithmetical complexity for the mapping \( H_k^{-1} \).

Without having a priori estimates for the spectral boundaries \( \gamma_1 \) and \( \gamma_2 \), the explicit value of \( \varepsilon \) is defined during the iterative process. We just have to control the sign of both numerator and denominator of the iterative parameter \( \tau_i \) and make \( \varepsilon \) smaller if one of them is negative. This makes algorithm (4.2) a “black box” solver.

In actual computations, the CG iterations can be accurate enough much earlier than stated by theory. The implementable control strategy of the relative error is derived by repeating arguments of [11, 8]. Let

\[
\delta_j = \| u_{kj}^h - u_{kj}^* \|_a^2
\]

denote the iterative error on the level \( j \). The energy norm of finite element solutions is monotonically increased during the running of nested iterations

\[
\| u_{kj}^h \|_a \leq \| u_{kj+1}^h \|_a \leq \| u_{kL}^h \|_a.
\]

Therefore the iterative termination criterion in the spirit of [11] is

\[
\delta_j \leq \rho \varepsilon^2 \| u_{kj-1}^* \|_a^2
\]

(4.8)

where \( \rho \) has to be understood as a safety factor. The energy norm of \( u_{kj-1}^* \) can be easily computed when CG iterations are initialized. The safety factor reflects both the disturbance introduced by using \( u_{kj-1}^* \) instead of \( u_{kj}^h \) and an error in evaluating the \( \delta_j \). In numerical experiments we used an empirical choice of \( \rho = 0.5 \).

The termination strategy similar to (4.8) has to be used if the mapping \( H_\lambda^{-1}[g_\lambda] \) is defined by the preconditioned CG iterations:

\[
\begin{align*}
p_0 &= g_\lambda - H_\lambda \lambda_0, \\
\lambda_i &= \lambda_{i-1} + \alpha_i p_{i-1}, \\
p_i &= Q_\lambda^{-1} r_i + \beta_i p_{i-1}, \quad i = 1, 2, \ldots,
\end{align*}
\]

(4.9)

where \( r_i = g_\lambda - H_\lambda \lambda_i \) and

\[
\alpha_i = \frac{(Q_\lambda^{-1} r_i, r_i)}{(p_i, H_\lambda p_i)}, \quad \beta_i = \frac{(Q_\lambda^{-1} r_i, r_i)}{(Q_\lambda^{-1} r_{i-1}, r_{i-1})}.
\]

The correct choice of the matrix \( Q_\lambda \) is a very important problem. At first, this matrix should be inverted easily. At second, the condition number of the matrix \( Q_\lambda^{-1} H_\lambda \) should be
independent of the jump in coefficients. The matrix $Q\lambda$ satisfying the above requirements was proposed in [16, 15]:

$$Q\lambda = D\lambda + \sum_{k=1}^{m} \frac{1}{c_k d_k} B_{\Gamma k} P_{\Gamma k} B_{\Gamma k}^T$$

where $D\lambda$ is a diagonal matrix spectrally equivalent to the matrix $\sum_{k=1}^{m} \frac{1}{a_k} B_{\Gamma k} B_{\Gamma k}^T$ and can be introduced by the lumping procedure. In the case of quasi-uniform grids, the eigenvalues of the matrix $Q^{-1}_\lambda H_\lambda$ belong to the segment $[\tilde{\theta}_1, \tilde{\theta}_2 \max_k d_k h_k]$, where $\tilde{\theta}_1$ and $\tilde{\theta}_2$ are positive constants independent of $a_k$, $c_k$, $d_k$ and $h_k$ [16]. Since the error in the energy norm of the PCG method is majorized by

$$\|\lambda - \lambda_i\|_{H_\lambda} = \min_{q \in P_i} \|q(Q^{-1}_\lambda H_\lambda)(\lambda - \lambda_0)\|_{H_\lambda}$$

where $P_i$ denotes the set of polynomials $q$ with degree less or equal to $i$ satisfying $q(0) = 1$, the theory of Chebyshev polynomials can be used for evaluating the number $m_\lambda$ of PCG iterations such that accuracy assumption (3.6) holds. This theory gives $m_\lambda \sim \max_k \sqrt{d_k h_k} \varepsilon$.

As we already noted the constant $\varepsilon$ can be chosen to be independent of $a_k$, $c_k$, $d_k$ and $h_k$. Thus, the arithmetical complexity of the mapping $H^{-1}_\lambda [g_\lambda]$ is estimated as

$$\hat{c}_1 \max_k n_k^{\frac{1}{2p}} \sum_{k=1}^{m} n_k^{\frac{p-1}{2p}} \ln n_k \leq \hat{c}_2 N.$$ 

Note that the last estimate is too stringent. Usually, the arithmetical complexity of CCG iterations significantly preponderates the amount of work needed for PCG iterations (4.9).

In the case of regular grids, the last estimate was justified in all numerical experiments (see Section 6).

We are still left with the decision of how to control the iterative error $\delta_j$. Since the fundamental properties of methods (4.5) and (4.9) are the same, it is sufficient to discuss the evaluation of the iterative error only for one of them. Let now

$$\delta_i = \|\lambda - \lambda_i\|_{H_\lambda}^2 \quad \text{and} \quad \epsilon_i = \|\lambda_{i+1} - \lambda_i\|_{H_\lambda}^2$$

denote the iterative and correction errors of (4.9). We assume that $\lambda_0 = 0$. The orthogonality of the PCG iterative corrections gives

$$\delta_{i+1} = \delta_i - \epsilon_i \quad \text{and} \quad \sum_{i=1}^{i} \epsilon_i \leq \|\lambda\|_{H_\lambda}^2.$$

We utilize the superlinear convergence of the PCG method to define a contraction factor $\Theta$, $\Theta < 1$, such that

$$\epsilon_i \leq \Theta \epsilon_{i-1} \quad \text{for} \quad i > i_0$$

where $i_0$ is a threshold index. For $i > i_0$ we can evaluate the iterative error as follows [11]:

$$\delta_i \leq \frac{\epsilon_i}{1 - \Theta}.$$
Finally, the desired termination criteria for cascadic iterations (4.5) and PCG method (4.9) are
\[
\frac{\epsilon_i}{1 - \Theta} \leq \rho \varepsilon^2 \|u^i_{k,j-1}\|_a^2 \quad \text{and} \quad \frac{\epsilon_i}{1 - \Theta} \leq \rho \varepsilon^2 \sum_{i=1}^t \epsilon_i, \tag{4.10}
\]
where the same notations \(\epsilon_i\) and \(\Theta\) are used for clarity to denote the correction error and the contraction factor for both methods.

5 Adaptive cascadic method

In this section we develop an adaptive control for the cascadic method on non-matching grids.

5.1 Adaptive mesh construction

Let us consider a nested family of finite element spaces generated by the adaptive method:
\[
X_{0h} \subset X_{1h} \subset \ldots \subset X_{Lh}, \quad L \geq 1,
\]
and the corresponding systems of linear algebraic equations
\[
A_j x_j = f_j, \quad j = 1, \ldots, L, \tag{5.1}
\]
where \(A_j\) is the symmetric \(N_j \times N_j\) matrix in the saddle point form and \(x_j = (u_j, \lambda_j)^T\). For each \(j = 1, \ldots, L\) we use results of previous sections to introduce a matrix \(H_j\) as the preconditioner for \(A_j\).

Let us make the following assumption on the family of triangulations:
\[
\sum_{k=1}^m \int_{\Omega_k} a_k |\nabla (u - u^h_j)|^2 + c_k |u - u^h_j|^2 d\Omega \leq \hat{c} N_j^{-2/p} \tag{5.2}
\]
where the constant \(\hat{c}\) depends on the shape regularity of triangulations and the \(L^2\)-norm of the right hand side \(f\). This assumption is a statement of optimal accuracy which can be justified by results of nonlinear approximation theory.

Let \(x^*_j\) be an approximate solution to (5.1) obtained after \(m_j\) iterations of (4.2). The cascadic principle states that on the level \(j = 0\) the linear system is solved directly, i.e. \(x^*_0 \equiv x_0\). On the finer level \(j \geq 1\), the linear system is solved iteratively by algorithm (4.2). Starting point of iterations is the finite element solution from the previous level. From the algebraic viewpoint this initial guess is defined as
\[
Q_j x^*_j - 1
\]
where \(Q_j\) is an interpolation operator, \(Q_j : \mathbb{R}^{N_j-1} \to \mathbb{R}^{N_j}\). The spectral equivalence of matrices \(A_j\) and \(H_j\) leads to a simple estimate of the energy norm of \(Q_j\):
\[
\|Q_j v\|_{H_j}^2 \leq \frac{1}{\gamma_1} |(Q_j A_j Q_j v, v)| = \frac{1}{\gamma_1} |(A_{j-1} v, v)| \leq \frac{\gamma_2}{\gamma_1} \|v\|_{H_{j-1}}^2, \quad \forall v \in \mathbb{R}^{N_j-1}. \tag{5.3}
\]
Let $\Theta$ be an contraction factor, $\Theta < 1$, such that

$$\|x_j - x_j^s\|_{H_j} \leq \Theta^{m_j}\|x_j - Q_jx_{j-1}\|_{H_j}.$$  

According to convergence rate estimate (4.3) and Lemma 3.2 the contraction factor is independent of $a_k$, $c_k$, $d_k$, $k = 1, \ldots, m$, and $j$. The triangle inequality and estimate (5.3) give

$$\|x_j - x_j^s\|_{H_j} \leq \Theta^{m_j}\|x_j - Q_jx_{j-1}\|_{H_j} + \Theta^{m_j}\frac{\gamma_2}{\gamma_1}\|x_{j-1} - x^*_{j-1}\|_{H_{j-1}}. \tag{5.4}$$

Let us show that assumption (5.2) can be used to estimate the first term in the right hand side. Since $x_j$ is the exact solution of (5.1), we have

$$\|x_j - Q_jx_{j-1}\|_{H_j} \leq \frac{1}{\gamma_1} |(A_j(x_j - Q_jx_{j-1}), x_j - Q_jx_{j-1})|$$

$$= \frac{1}{\gamma_1^2} \sum_{k=1}^{m} \|u_j - Q^{(u)}_j u_{j-1}\|_{A_k}^2 \leq \hat{c} N_j^{-2/p},$$

where $Q^{(u)}_j$ is the restriction of $Q_j$ to the subvector $u_j$. With the choice of iteration numbers $m_j$ as

$$\frac{\gamma_2}{\gamma_1} \left(\frac{N_j}{N_{j-1}}\right)^{1/p} \Theta^{m_j} \leq \frac{1}{2}, \tag{5.5}$$

the iterative errors introduced on coarser levels are rapidly damped on finer levels, so that the iterative approximation $x^*_L$ achieves the discretization error accuracy. Indeed, using assumption (5.2), we get by induction

$$\|x_L - x^*_L\|_{H_L} \leq \frac{\hat{c}}{2N_L^{1/p}} + \frac{1}{2} \left(\frac{N_{j-1}}{N_j}\right)^{1/p} \|x_L - x^*_L\|_{H_{L-1}} \leq \frac{\hat{c}}{N_L^{1/p}} \sum_{j=1}^{L} \frac{1}{2^j} = \frac{\hat{c}}{N_L^{1/p}}.$$  

In practice, the termination criterion (5.5) has to be replaced by an implementable termination strategy. Let $x_{ji}$ be the approximation to $x_j$ on the iteration $i$ and

$$\delta_i = \|x_j - x_{ji}\|_{H_j}, \quad \epsilon_i = \tau_i(2\tau_i^* - \tau_i)\|H_j^{-1}[A_j H_j^{-1}[r_i]]\|^2_{H_j}$$

denote different iterative errors. Since both errors are not available in the iterative process, we introduce a calculable approximation of $\epsilon_i$:

$$\hat{\epsilon}_i = \tau_i^2(AH_j^{-1}[r_i], H_j^{-1}[A_j H_j^{-1}[r_i]]).$$

Using the above notations, we replace termination criterion (5.5) by

$$\left(\frac{\gamma_2}{\gamma_1}\right)^2 \left(\frac{N_j}{N_{j-1}}\right)^{2/p} \delta_i \leq \frac{1}{4} \delta_0. \tag{5.6}$$

Let us evaluate both sides of the last inequality. The ratio $\gamma_2/\gamma_1$ is bounded above by $\sqrt{\sigma_2/\sigma_1}$ and consequently can be approximated as $(1 - \Theta^2)^{-1/4}$. We apply results of Section 4.1 and 4.2 to estimate the correction error $\epsilon_i$:

$$(1 - \varepsilon) \left(\frac{2\tau_i^*}{\tau_i} - 1\right) \leq \frac{\epsilon_i}{\hat{\epsilon}_i} \leq \left(\frac{2\tau_i^*}{\tau_i} - 1\right) \frac{1 - \varepsilon}{1 - 2\varepsilon}.$$
The initial error $\delta_0$ can be approximated by the sum of iterative corrections $\epsilon_i$. The statement of the optimal convergence rate gives

$$\delta_{i+1} = \delta_i - \epsilon_i \leq \Theta^2 \delta_i$$

and consequently the above estimate for $\delta_i$. Summing the above speculations, we can replace (5.6) by

$$\hat{\epsilon}_i \left(1 - \Theta^2\right) \left(\frac{N_j}{N_{j-1}}\right)^{2/\rho} \leq \frac{\rho}{4} \sum_{i=1}^{i} \hat{\epsilon}_i$$

(5.7)

where $\rho$ again is a safety factor, $\rho < 1$. The safety factor accumulates errors in evaluating of $\Theta$ and $\delta_i$ as well as the difference between $\epsilon_i$ and $\hat{\epsilon}_i$. In numerical experiments we used an empirical value $\rho = 10^{-2}$. The evaluation of the contraction factor requires a heuristic assumption. Note that in the case of exact inner iterations, the contraction factors $\Theta_{ji} = \sqrt{\delta_i/\delta_{i-1}}$ are bounded above by $\Theta$ and monotonically increased. Therefore, in the absence of better information an estimate of the kind

$$\Theta = \max_{j,i} \Theta_{ji}, \quad \Theta^2_{ji} = \frac{\delta_i}{\delta_{i-1}} \approx \frac{\hat{\epsilon}_i}{\hat{\epsilon}_{i-1}}$$

will be applied. The termination criterion does not involve any user given tolerance. Consequently, the unnecessary levels of the refinement are possible. To avoid the additional amount of computational work, we modify the termination criterion as follows (see [11] for details):

$$\hat{\epsilon}_i \left(1 - \Theta^2\right) \leq \rho \text{TOL}^2,$$

(5.8)

where $\text{TOL}$ is the tolerance given by the user.

### 5.2 Discretization error estimation

First, we describe briefly a discretization error estimation based on an extension of the linear finite element space $V_{kh}$ by a space $V_{kQ}$ of piecewise quadratic finite element functions associated with edges of $\Omega_{kh}$ [12]. Each quadratic finite element function vanishes at the vertices of $\Omega_{kh}$ and can be characterized by the midpoint value on the corresponding edge. Here $\Omega_{kh}$ denotes the triangulation obtained after several refinement levels of the adaptive method. Let

$$V_Q = \prod_{k=1}^{m} V_{kQ} \quad \text{and} \quad X_Q = V_Q \times \Lambda_h.$$

The finite element problem with Lagrange multipliers at the interfaces $\Gamma_{kl}$, $l < k$, posed in $X_Q$ results in a system of algebraic equations in the saddle point form:

$$\begin{bmatrix} A_{LL} & A_{LQ} & B_L^T \\ A_{QL} & A_{QQ} & B_Q^T \\ B_L & B_Q & 0 \end{bmatrix} \begin{bmatrix} u_L \\ u_Q \\ \lambda \end{bmatrix} = \begin{bmatrix} f_L \\ f_Q \\ 0 \end{bmatrix}.$$
Let $x^*_L = (u^*_L, \lambda^*)^T$ be an approximate solution obtained in the two-level algorithm. Let us define two iterative errors $d_L = u_L - u^*_L$, $d_\lambda = \lambda - \lambda^*$ and a discretization error $d_Q = u_Q$. They satisfy

$$
\begin{bmatrix}
A_{LL} & A_{LQ} & B_L^T \\
A_{QL} & A_{QQ} & B_Q^T \\
B_L & B_Q & 0
\end{bmatrix}
\begin{bmatrix}
d_L \\
d_Q \\
d_\lambda
\end{bmatrix} =
\begin{bmatrix}
r_L \\
r_Q \\
r_\lambda
\end{bmatrix},
$$

(5.9)

with

$$
r_L = f_L - A_{LL}u_L - B_L^T\lambda^*, \\
r_Q = f_Q - A_{QL}u_L - B_Q^T\lambda^* \\
r_\lambda = -B_Lu_L^*.$$

The exact solution of (5.9) is too expensive. In order to estimate the discretization error $d_Q$, equation (5.9) is replaced by a much simpler system

$$
\begin{bmatrix}
A_{LL} & 0 & 0 \\
0 & D_{QQ} & 0 \\
0 & 0 & S_Q
\end{bmatrix}
\begin{bmatrix}
\tilde{d}_L \\
\tilde{d}_Q \\
\tilde{d}_\lambda
\end{bmatrix} =
\begin{bmatrix}
r_L \\
r_Q \\
r_\lambda
\end{bmatrix}.
$$

(5.10)

Here $D_{QQ}$ is the diagonal part of $A_{QQ}$ and

$$S_Q = [B_L B_Q] \left[ \begin{array}{cc} A_{LL} & A_{LQ} \\ A_{QL} & A_{QQ} \end{array} \right]^{-1} \left[ \begin{array}{c} B_L^T \\ B_Q^T \end{array} \right].$$

Using a local element analysis, we can easily show that a diagonal matrix $\text{diag}\{A_{LL}, D_{QQ}\}$ is spectrally equivalent to the corresponding $2 \times 2$ block matrix in (5.9). Lemma 3.1 states that stiffness matrices in (5.9) and (5.10) are spectrally equivalent. Therefore an energy norm $\|d_Q\|_E$ of the discretization error can be estimated by

$$\|d_Q\|_E \approx \|\tilde{d}_Q\|_{D_{QQ}}.$$

Since the global discretization error is the sum of local errors distributed over edges of triangulations $\Omega_{kh}$, the following mesh refinement strategy is applied. Let $\eta_T$ be a local error corresponding to a triangle $T$ of $\Omega_{kh}$. It is calculated as a sum of errors associated with edges of $T$. Then those triangles are marked for the refinement that satisfy

$$\eta_T \geq \frac{1}{4} \max_T \eta_T.$$

6 Numerical experiments

In this section, we now compare the above described DD/CCG algorithm with the (unpreconditioned) CCG method [6, 11] and with KASKADE [12] preconditioned by BPX [9] when applied to the undecomposed problem. In the DD context, we furthermore compare the theoretically justified two-term gradient method as outer iteration for the numerical
saddle point problem with other iterative solvers like the flexible GMRES (fGMRES), the preconditioned conjugate gradient method (PCG), and the three-term preconditioned conjugate residual (PCR) method. For illustration purposes, we chose a relatively simple test problem from the literature [12] — adding, however, a small perturbation term.

As a material jump model problem, let $\Omega = [0, 1]^2$ and consider

\[-\nabla a(x) \nabla u + 10^{-4} u = 100 \quad \text{in } \Omega, \]

\[u = 0 \quad \text{on } \partial \Omega\]

with

\[a(x) = \begin{cases} 
1 & \text{if } x \in [0.25, 0.75] \setminus [0.375, 0.625]^2, \\
10^6 & \text{otherwise.}
\end{cases}\]

With the strong material jump, this model problem is dominated by $L_2$ rather than $H^1$ — which will cause trouble for the CCG algorithm. For DD, the unit square is decomposed into 9 subdomains as shown by thick lines in Fig. 6.1. In Table 6.1 we give the number of inner iterations on level $j$, the problem size $N$, the accumulated CPU time and the energy norm of the final iterative solution $u_j^*$. As expected from [6, 7, 8], the $L_2$-domination of the model problem leads to the fact that the CCG method fails to converge to the correct solution. (This behavior would also occur, if diagonal preconditioning were used.) The situation is significantly improved as soon as an efficient preconditioner is included (see KASKADE/BPX).

In Fig. 6.2 we compare the grids arising in the DD/CCG and in the KASKADE/BPX methods. As can be seen, the matching grids from KASKADE/BPX require excess refinements near interfaces between “fine grid” and “coarse grid” subdomains — an effect, which will be more marked in 3D problems. The non-matching grids from DD/CCG, however, lead to a more flexible and parallel adaptive algorithm. This desirable feature is paid for by the appearance of additional degrees of freedom associated with the Lagrange multipliers and by the saddle point structure of the numerical problem.

Table 6.1: Comparison of three methods: $\varepsilon = 0.06$, $TOL = 0.02\|x_L\|_{H^1}$.
Figure 6.1: Initial grid decomposition and iterative solution.

Figure 6.2: Adaptive grids in the two-term gradient DD/CCG method (row above) and the KASKADE/BPX method (row below).
In a second set of experiments, we replaced the simple two-term gradient outer iterative process by fGMRES, PCG, or PCR, respectively – see Table 6.2. Each iteration of fGMRES and PCR requires one matrix-vector multiplication $A x$ and one computation of the mapping $H^{-1}[x]$. Hence, if the number of iterations is sufficiently small or if the cost of a matrix-vector operation $A H^{-1}[x]$ is sufficiently big, one fGMRES iteration will be roughly twice as cheap as one two-term gradient iteration. On the other hand, one PCG iteration requires two pairs of matrix-vector operations $A x$ and $H^{-1}[x]$. Hence, one PCG iteration will approximately cost the same as one two-term gradient iteration. In the macro-hybrid formulation, the nested iterations considered above do not keep the energy norm of the iterative error below the discretization error reached on coarser levels. In other words: a finite element solution obtained on level $j$ is not the optimal starting point for further iterations on the next level. As a consequence, we need much more iterations per level in the DD/CCG method compared to the KASKADE/BPX method. This explains the observation that the fastest among our DD/CCG methods turns out to be a factor of two slower than the KASKADE/BPX method.

Table 6.2: Comparison of four outer iterative methods ($\varepsilon = 0.06$, $TOL = 0.02\|x_L\|_{H_L}$).
Conclusion

The cascadic conjugate gradient method (CCG) tends to be inefficient when applied to material jump problems with sufficiently strong jumps. Therefore, in this paper, we worked out a combination of domain decomposition (DD) with CCG applied only within homogeneous subdomains. In order to allow for non–matching grids at interfaces, mortar finite elements were additionally suggested. In an illustrative model problem in 2D, the best version of our derived algorithm turns out to be roughly a factor of two slower in comparison with the preconditioned KASKADE/BPX applied to the problem as a whole. However, the extremely nice non-matching grid feature may compensate for this loss in computing time. Further studies with the aim of developing an improved DD/CCG algorithm are needed to come to a final conclusion as to the relative merits of this method.

Figure 6.3: Arithmetical complexity per iteration (left) and total computing time (right) versus problem size $N$. 
References


