Chapter 5
Iterative Solution Methods

5.1 Condition Numbers and Multigrid

5.1.1 Conditioning of the Stiffness Matrix

For a triangulation \( \mathcal{T}_h \) of a Lipschitz domain \( \Omega \subset \mathbb{R}^d \), we let \( A_h \in \mathbb{R}^{n \times n} \) denote the finite element stiffness matrix related to the finite element space \( \mathcal{S}_1^0(\mathcal{T}_h) \), i.e.,

\[
(A_h)_{ij} = \int_{\Omega} \nabla \varphi_{z_i} \cdot \nabla \varphi_{z_j} \, dx
\]

for \( i, j = 1, 2, \ldots, n \), and with the nodal basis functions \( (\varphi_{z_j} : j = 1, 2, \ldots, n) \) for an enumeration \( z_1, z_2, \ldots, z_n \in \mathcal{N}_h \cap \Omega \) of the free nodes. We assume that \( \mathcal{T}_h \) is quasiuniform, i.e., that \( c_{\text{unif}} h \leq \min_{T \in \mathcal{T}_h} h_T \leq h \), so that we have the inverse estimate

\[
\| \nabla v_h \|_{L^2(\Omega)} \leq c_{\text{inv}} h^{-1} \| v_h \|_{L^2(\Omega)}
\]

and the norm equivalence

\[
c_{\text{eq}}^{-2} \| v_h \|^2_{L^2(\Omega)} \leq h^d \sum_{z \in \mathcal{N}_h} |v_h(z)|^2 \leq c_{\text{eq}}^2 \| v_h \|^2_{L^2(\Omega)}
\]

for all \( v_h \in \mathcal{S}_1^0(\mathcal{T}_h) \), cf. Lemmas 3.8 and 3.7. For the iterative solution of the linear system of equations, e.g., with the conjugate gradient method, the spectral condition number is an important quantity.

**Theorem 5.1 (Spectral Conditioning)** We have \( \lambda_{\text{max}}(A_h) \leq c_{\text{max}} h^{d-2} \) and

\[
\text{cond}_2(A_h) = \frac{\lambda_{\text{max}}(A_h)}{\lambda_{\text{min}}(A_h)} \leq c_{\text{cond}} h^{-2}.
\]
Proof We use the characterization of $\lambda_{\text{max}}$ and $\lambda_{\text{min}}$ by Rayleigh-quotients, i.e.,

\[
\lambda_{\text{max}} = \sup_{y \in \mathbb{R}^n \setminus \{0\}} \frac{A_h y \cdot y}{|y|^2}, \quad \lambda_{\text{min}} = \inf_{y \in \mathbb{R}^n \setminus \{0\}} \frac{A_h y \cdot y}{|y|^2}.
\]

We identify a vector $y \in \mathbb{R}^n$ with a function $v_h \in \mathcal{S}_0^1(\mathcal{T}_h)$ via

\[
v_h = \sum_{j=1}^n y_j \varphi_j.
\]

We have

\[
A_h y \cdot y = \int_{\Omega} \nabla v_h \cdot \nabla v_h \, dx
\]

\[
= \left\| \nabla v_h \right\|_{L^2(\Omega)}^2
\]

\[
\leq c_{\text{inv}}^2 h^{-2} \left\| v_h \right\|_{L^2(\Omega)}^2
\]

\[
\leq c_{\text{inv}}^2 c_{\text{eq}}^{-2} h^{d-2} |y|^2,
\]

which proves that $\lambda_{\text{max}}(A_h) \leq ch^{d-2}$. To bound the smallest eigenvalue from below, we use Poincaré’s inequality to verify that

\[
|y|^2 \leq c_{\text{eq}}^2 h^{-d} \left\| v_h \right\|_{L^2(\Omega)}^2 \leq c_{\text{eq}}^2 c_{\text{P}}^{-2} \left\| \nabla v_h \right\|_{L^2(\Omega)}^2.
\]

This implies that $\lambda_{\text{min}}(A_h) \geq h^d c_{\text{eq}}^{-2} c_{\text{P}}^{-2}$. \hfill \square

Remark 5.1 The estimate of the theorem is optimal.

Although the condition number of the stiffness matrix with respect to the Euclidean norm is large for small mesh-sizes, the discrete Poisson problem is not ill-conditioned. In particular, there exist natural norms for which the condition number is optimal.

Proposition 5.1 (Optimal Conditioning) Let $f, \tilde{f} \in L^2(\Omega)$ and let $u_h, \tilde{u}_h \in \mathcal{S}_0^1(\mathcal{T}_h)$ be the corresponding Galerkin approximations of the Poisson problem. Assume that $u_h \neq 0$. We then have

\[
\frac{\left\| \nabla (u_h - \tilde{u}_h) \right\|_{L^2(\Omega)}}{\left\| \nabla u_h \right\|_{L^2(\Omega)}} \leq \frac{\left\| f - \tilde{f} \right\|_{\mathcal{S}_0^1(\mathcal{T}_h)'}}{\left\| f \right\|_{\mathcal{S}_0^1(\mathcal{T}_h)'}}.
\]

where for $g \in L^2(\Omega)$,

\[
\left\| g \right\|_{\mathcal{S}_0^1(\mathcal{T}_h)'} = \sup_{v_h \in \mathcal{S}_0^1(\mathcal{T}_h) \setminus \{0\}} \frac{\int_{\Omega} g v_h \, dx}{\left\| \nabla v_h \right\|_{L^2(\Omega)}}.
\]
Proof For the difference $u_h - \tilde{u}_h \in \mathcal{S}_0^1(\mathcal{T}_h)$ and every $v_h \in \mathcal{S}_0^1(\mathcal{T}_h)$, we have that
\[
\int_\Omega \nabla (u_h - \tilde{u}_h) \cdot \nabla v_h \, dx = \int_\Omega (f - \tilde{f}) v_h \, dx.
\]
The choice $v_h = u_h - \tilde{u}_h$ implies that
\[
\| \nabla (u_h - \tilde{u}_h) \|^2_{L^2(\Omega)} = \int_\Omega (f - \tilde{f}) (u_h - \tilde{u}_h) \, dx \\
\leq \| f - \tilde{f} \|_{\mathcal{S}_0^1(\mathcal{T}_h)'} \| \nabla (u_h - \tilde{u}_h) \|_{L^2(\Omega)}.
\]
Combining this with the bound
\[
\| f \|_{\mathcal{S}_0^1(\mathcal{T}_h)'} = \sup_{v_h \in \mathcal{S}_0^1(\mathcal{T}_h) \setminus \{0\}} \frac{\int_\Omega f v_h \, dx}{\| \nabla v_h \|_{L^2(\Omega)}} \leq \| \nabla u_h \|_{L^2(\Omega)}
\]
implies the estimate. \(\square\)

Because of the large condition number $\text{cond}_2(A_h)$ for $0 < h \ll 1$, stationary iteration methods like the Richardson-, the Jacobi-, or the Gauss–Seidel and the conjugate gradient method will typically converge slowly. Figure 5.1 displays the errors $u_h^k - u_h$ in a one-dimensional Poisson problem of the Richardson iteration
\[
x^{k+1} = x^k - \omega_h (A_h x^k - b)
\]
after $k = 1, 10, 100$ iterations, with $\omega_h = h^{2-d} \sim 1/\lambda_{\max}(A_h)$ and randomly defined initial configuration $u_0^h$. It is important to observe that rapid oscillations are removed within a few iterations, while the maximal iteration error decays slowly.

For analytical considerations it is helpful to interpret the matrix $A_h$ not as an operator on $\mathbb{R}^n$ but on $\mathcal{S}_0^1(\mathcal{T}_h)$. In particular, we consider the linear operator
\[
\tilde{A}_h : \mathcal{S}_0^1(\mathcal{T}_h) \rightarrow \mathcal{S}_0^1(\mathcal{T}_h)' \cong \mathcal{S}_0^1(\mathcal{T}_h)
\]

Fig. 5.1 Iteration error $u_h^k - u_h$ after $k = 1, 10, 100$ Richardson iterations in the discretized Poisson problem $-u'' = 1$ in $(0, 1)$, $u(0) = u(1) = 0$, with $h = 2^{-4}$
defined by $\xi_h = \hat{A}_h v_h$, such that for all $w_h \in \mathcal{S}_0^1(\mathcal{T}_h)$, we have

$$\int_{\Omega} \xi_h w_h \, dx = A_h V_h \cdot W_h,$$

where $V_h, W_h \in \mathbb{R}^n$ are the coefficient vectors for $v_h, w_h \in \mathcal{S}_0^1(\mathcal{T}_h)$, respectively. Note that for all $v_h, w_h \in \mathcal{S}_0^1(\mathcal{T}_h)$, we have

$$\int_{\Omega} (\hat{A}_h v_h) w_h \, dx = A_h V_h \cdot W_h = \int_{\Omega} \nabla v_h \cdot \nabla w_h \, dx.$$

The definition of $\hat{A}_h$ motivates using the $L^2$-norm and the corresponding inner product on the vector space $\mathcal{S}_0^1(\mathcal{T}_h)$. We then have

$$\lambda_{\text{max}}(\hat{A}_h) = \sup_{v_h \in \mathcal{S}_0^1(\mathcal{T}_h) \setminus \{0\}} \frac{\int_{\Omega} (\hat{A}_h v_h) v_h \, dx}{\|v_h\|_{L^2(\Omega)}^2} \leq c_{\text{inv}} h^{-2}.$$ 

and

$$\lambda_{\text{min}}(\hat{A}_h) = \inf_{v_h \in \mathcal{S}_0^1(\mathcal{T}_h) \setminus \{0\}} \frac{\int_{\Omega} (\hat{A}_h v_h) v_h \, dx}{\|v_h\|_{L^2(\Omega)}^2} \geq c_{p}.$$ 

The operator $\hat{A}_h : \mathcal{S}_0^1(\mathcal{T}_h) \to \mathcal{S}_0^1(\mathcal{T}_h)$ may be regarded as a discrete version of the negative Laplace operator, i.e., often one denotes $\hat{A}_h = -\Delta_h$. The representing matrix differs from $A_h$ by the inverse of the mass matrix.

### 5.1.2 Two-Grid Iteration

The observation that the error $e_h^k = u_h^k - u_h$ in the Richardson iteration is smooth after a few iterations implies that the function $e_h^k$ can be resolved accurately on a coarser grid with mesh-size $2h$, on which the iteration is less expensive. The following lemma quantifies the smoothing property of the Richardson iteration. Given a scalar product $\langle \cdot, \cdot \rangle$ on an $n$-dimensional space $V$, and a symmetric and positive definite operator $\hat{A} : V \to V$, i.e., we have $\langle \hat{A} v, w \rangle = \langle v, \hat{A} w \rangle$ and $\langle \hat{A} v, v \rangle > 0$ for all $v, w \in V \setminus \{0\}$, we define the (generalized) maximal eigenvalue of $\hat{A}$ via

$$\lambda_{\text{max}} = \sup_{v \in V \setminus \{0\}} \frac{\langle \hat{A} v, v \rangle}{\langle v, v \rangle}.$$ 

Throughout what follows we adopt arguments from [14].
**Lemma 5.1 (Smoothing Property)**  Let $\hat{A} : V \rightarrow V$ be a positive definite and symmetric operator. With the Richardson iteration operator

$$ S = I - \lambda^{-1}_{\text{max}} \hat{A}, $$

we have for every $k \in \mathbb{N}$ that

$$ \|\hat{A}S^k\| \leq k^{-1}\lambda_{\text{max}}, $$

where $\| \cdot \|$ is the operator norm induced by the scalar product on $V$.

**Proof** We let $(v_1, v_2, \ldots, v_n)$ be an orthonormal basis of eigenvectors for $\hat{A}$, associated with the generalized eigenvalues $0 < \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n = \lambda_{\text{max}}$ of $\hat{A}$, i.e., $\langle \hat{A}v_j, w \rangle = \lambda_j \langle v_j, w \rangle$ for $j = 1, 2, \ldots, n$ and all $w \in V$. For an arbitrary element

$$ v = \sum_{j=1}^{n} \alpha_j v_j, $$

we have

$$ \langle \hat{A}(I - \lambda^{-1}_{\text{max}} \hat{A})^k v, w \rangle = \sum_{j=1}^{n} \lambda_j (1 - \lambda_j/\lambda_n)^k \alpha_j \langle v_j, w \rangle. $$

This shows that

$$ \|\hat{A}S^k v\| \leq \lambda_n \max_{j=1, \ldots, n} \frac{\lambda_j}{\lambda_n} \left(1 - \frac{\lambda_j}{\lambda_n}\right)^k \|v\|. $$

An exercise proves that

$$ \max_{t \in [0, 1]} t(1 - t)^k \leq \frac{1}{ek} \leq k^{-1}, $$

and this implies the estimate. \(\square\)

The quantity $\|\hat{A}_h v_h\|$ is a measure of the smoothness of a finite element function $v_h$. Hence we see that a few iterations with the Richardson matrix improve the smoothness of the iteration error. Note however that the factor $\lambda_{\text{max}} \sim h^{-2}$ is large in case of the Poisson problem. To quantify the idea of representing the iteration error on a coarser grid, we have to analyze the corresponding approximation errors.

**Lemma 5.2 (Approximation Property)**  Assume that the Poisson problem is $H^2$-regular, let $\mathcal{T}_h$ and $\mathcal{T}_H$ be triangulations of $\Omega$ with mesh-sizes $h = \gamma H$ with $0 < \gamma < 1$. Let $v_h \in \mathcal{S}_0^1(\mathcal{T}_h)$ and $v_H \in \mathcal{S}_0^1(\mathcal{T}_H)$ be the Galerkin approximations of the
Poisson problem with $r \in L^2(\Omega)$ on the right-hand side. We then have that

$$\|c_h - c_H\|_{L^2(\Omega)} \leq c_{\text{ap}} h^2 \|r\|_{L^2(\Omega)}.$$  

**Proof** If $z \in H^1_0(\Omega) \cap H^2(\Omega)$ is the exact solution of the Poisson problem $-\Delta z = r$, then we have that

$$\|z - c_h\|_{L^2(\Omega)} \leq ch^2 \|r\|_{L^2(\Omega)},$$

$$\|z - c_H\|_{L^2(\Omega)} \leq cH^2 \|r\|_{L^2(\Omega)}.$$  

The triangle inequality and $H^2 = y^{-2} h^2$ imply the estimate. □

Assume that we have carried out $k$ iterations of the Richardson scheme. This defines an approximation $u^k_h \in \mathcal{S}_1^0(\mathcal{T}_h)$ and a residual $r_h \in \mathcal{S}_1^0(\mathcal{T}_h)$ via

$$r_h = b_h - \tilde{A}_h u^k_h = \tilde{A}_h (u_h - u^k_h).$$

Here $b_h \in \mathcal{S}_0^1(\mathcal{T}_h)$ is defined by

$$\int \Omega b_h v_h \, dx = \int \Omega f v_h \, dx$$

for all $v_h \in \mathcal{S}_1^0(\mathcal{T}_h)$. To obtain the exact discrete solution $u_h$ from the approximation $u^k_h$, we have to compute the correction $c_h = u_h - u^k_h \in \mathcal{S}_0^1(\mathcal{T}_h)$ which satisfies

$$\int \Omega \nabla c_h \cdot \nabla v_h \, dx = \int \Omega r_h v_h \, dx$$

for all $v_h \in \mathcal{S}_1^0(\mathcal{T}_h)$. We approximate the solution $c_h$ on the coarser grid $\mathcal{T}_H$.

**Theorem 5.2 (Two-Grid Iteration)** Let $u^k_h \in \mathcal{S}_0^1(\mathcal{T}_h)$ be obtained from $k \geq 1$ Richardson iterations with an initial $u^0_h \in \mathcal{S}_0^1(\mathcal{T}_h)$. Assume that $\mathcal{S}_0^1(\mathcal{T}_H) \subset \mathcal{S}_0^1(\mathcal{T}_h)$ and let $c_H \in \mathcal{S}_0^1(\mathcal{T}_H)$ solve

$$\int \Omega \nabla c_H \cdot \nabla v_H \, dx = \int \Omega r_h v_H \, dx = \int \Omega f v_H \, dx - \int \Omega \nabla u^k_h \cdot \nabla v_H \, dx$$

for all $v_H \in \mathcal{S}_0^1(\mathcal{T}_H)$. For the function $u^{k+\text{corr}}_h = u^k_h + c_H \in \mathcal{S}_0^1(\mathcal{T}_h)$ and the Galerkin approximation $u_h \in \mathcal{S}_0^1(\mathcal{T}_h)$ of the $H^2$-regular Poisson problem with $f \in L^2(\Omega)$ on the right-hand side, we have

$$\|u_h - u^{k+\text{corr}}_h\|_{L^2(\Omega)} \leq c_H k^{-1} \|u_h - u^0_h\|_{L^2(\Omega)}.$$
Proof

(i) The Richardson iteration computes $u_h^k$ recursively via
\[ u_h^k = u_h^{k-1} - \lambda^{-1}_\text{max}(\hat{A}_h)u_h^{k-1} - b_h. \]

Noting $b_h = \hat{A}_h u_h$, with the Richardson operator $S = (I - \lambda^{-1}_\text{max}\hat{A}_h)$ we have that
\[ u_h - u_h^k = u_h - u_h^{k-1} - \lambda^{-1}_\text{max}\hat{A}_h(u_h - u_h^{k-1}) = S(u_h - u_h^{k-1}). \]

By induction, we see that $u_h - u_h^k = S^k(u_h - u_h^0)$, and with the smoothing property of Lemma 5.1 we find that
\[ \|\hat{A}_h(u_h - u_h^k)\|_{L^2(\Omega)} = \|\hat{A}_h S^k(u_h - u_h^0)\|_{L^2(\Omega)} \leq k^{-1} \lambda_{\text{max}}\|u_h - u_h^0\|_{L^2(\Omega)}. \]

(ii) Defining $r_h = b_h - \hat{A}_h u_h^k$, and letting $c_h \in \mathcal{S}_0^1(\mathcal{T}_h)$ be the Galerkin approximation of the Poisson problem with $r_h$ on the right-hand side, we have that
\[ u_h = u_h^k + c_h. \]

With $c_H \in \mathcal{S}_1^1(\mathcal{T}_H)$ that defines $u_h^{k+\text{corr}} = u_h^k + c_H$, the approximation property of Lemma 5.2 leads to
\[ \|u_h - u_h^{k+\text{corr}}\|_{L^2(\Omega)} = \|c_h - c_H\|_{L^2(\Omega)} \leq c_{ap} h^2 \|r_h\|_{L^2(\Omega)} = c_{ap} h^2 \|\hat{A}_h u_h^k - b_h\|_{L^2(\Omega)} = c_{ap} h^2 \|\hat{A}_h(u_h^k - u_h)\|_{L^2(\Omega)}. \]

(iii) Combining the previous estimates and noting $\lambda_{\text{max}}(\hat{A}_h) \leq ch^{-2}$ prove the estimate. \qed

Remark 5.2 We have proved the convergence of an iterative scheme for solving the discretized Poisson problem with an $h$-independent reduction factor. To achieve an iteration error, which is comparable to the discretization error $O(h^2)$, approximately $k \sim h^{-2}$ iterations are required for the two-level iteration defined by the theorem.
5.1.3 Multigrid Algorithm

Computing the coarse-grid correction \( c_H \) in the two-grid iteration requires solving a discrete Poisson problem. This can again be done approximately via Richardson iteration and another coarse grid correction. Repeating this idea with a hierarchy of refined triangulations

\[ \mathcal{T}_1 \rightarrow \mathcal{T}_2 \rightarrow \cdots \rightarrow \mathcal{T}_L \]

leads to a multigrid algorithm. Assuming that \( h_L = \gamma^{-L} h_1 \), for a mesh-size reduction factor \( 0 < \gamma < 1 \) and \( h_1 = \mathcal{O}(1) \), only a finite number of iterations is necessary to achieve an iteration error that is comparable to the discretization error with a complexity \( \mathcal{O}(n) \). To formulate the idea in an implementable way, we specify matrices that realize the grid transfer.

**Lemma 5.3 (Prolongation Operator)** Let \( \mathcal{T}_h \) be a refinement of the triangulation \( \mathcal{T}_H \) in the sense that \( S_1^0(\mathcal{T}_h) \subset S_1^0(\mathcal{T}_H) \). There exists a uniquely defined matrix \( P_h^H \in \mathbb{R}^{n \times n} \) such that for every \( v_H \in S_1^0(\mathcal{T}_H) \) with coefficient vector \( V_H \in \mathbb{R}^N \), the vector \( V_h = P_h^H V_H \in \mathbb{R}^n \) is the coefficient vector of \( v_H \) with respect to the nodal basis of \( S_1^0(\mathcal{T}_h) \).

**Proof** Exercise. \( \square \)

The matrix \( P_h^H \) embeds the nodal values of \( v_H \) in a larger vector. Its transpose \( (P_h^H)^\top \in \mathbb{R}^{N \times n} \) can be regarded as a mapping from \( S_0^1(\mathcal{T}_h) \) to \( S_0^1(\mathcal{T}_H) \).

**Definition 5.1** The restriction operator \( R_h^H : S_0^1(\mathcal{T}_h) \rightarrow S_0^1(\mathcal{T}_H) \) is, for \( v_h \in S_0^1(\mathcal{T}_h) \) with coefficient vector \( V_h \in \mathbb{R}^n \), defined as the function \( v_H \in S_0^1(\mathcal{T}_H) \) with coefficient vector \( V_H = R_h^H V_h \in \mathbb{R}^N \) for \( R_h^H = (P_h^H)^\top \).

Note that \( R_h^H v_h \neq v_h \) and \( R_h^H P_h^H v_H \neq v_H \) for \( v_h \in S_0^1(\mathcal{T}_h) \) and \( v_H \in S_0^1(\mathcal{T}_H) \) in general, cf. Fig 5.2.

**Fig. 5.2** Restriction of a fine-grid function to a coarser grid, and prolongation of a coarse-grid function onto the finer grid.
5.1 Condition Numbers and Multigrid

Remarks 5.3

(i) The restriction can alternatively be defined by the $L^2$-projection onto $\mathcal{S}_0^1(\mathcal{T}_H)$, i.e., $v_H = R^h_H v_h$ is defined by

$$\int_{\Omega} v_H w_H \, dx = \int_{\Omega} v_h w_H \, dx$$

for all $w_H \in \mathcal{S}_0^1(\mathcal{T}_H)$. This definition is compatible with Theorem 5.2, but is more expensive to realize.

(ii) An important aspect of the prolongation is that the finite element stiffness matrix only has to be computed on the finest grid, i.e., we have

$$V_H^T A_H W_H = \int_{\Omega} \nabla v_H \cdot \nabla w_H \, dx = (P^h_H v_H)^T A_h (P^h_H W_H)$$

for functions $v_H, w_H \in \mathcal{S}_0^1(\mathcal{T}_H)$ with coefficient vectors $V_H, W_H \in \mathbb{R}^N$.

The multigrid algorithm consists in carrying out a specified number of Richardson iterations, called the pre-smoothing procedure, computing the coarse-grid correction by restricting the residual to a coarser grid, and carrying out additional Richardson iterations called post-smoothing. Since the computation of the coarse-grid correction is also done with this idea, unless we have reached the coarsest level, we obtain a recursive algorithm, which is illustrated in Fig. 5.3.

Algorithm 5.1 (Multigrid) Let $\mathcal{T}_{\ell_0} < \cdots < \mathcal{T}_L$ be a sequence of refined triangulations, with stiffness matrices $A_{\ell}$, $\ell_0 \leq \ell \leq L$, let $b_L \in \mathcal{S}_0^1(\mathcal{T}_L)$ be the right-hand side, and let $v_{\text{pre}}, v_{\text{post}} \in \mathbb{N}$. The approximation $u_{\text{mg}}^L \in \mathcal{S}_0^1(\mathcal{T}_L)$ is defined by

$$u_{\text{mg}}^L = MG(A_L, b_L, L),$$

with the recursive function $MG : \mathbb{R}^{n \times n} \times \mathcal{S}_0^1(\mathcal{T}) \times \{\ell_0, \ldots, L\} \to \mathcal{S}_0^1(\mathcal{T})$

$$MG[A_{\ell}, b_{\ell}, \ell] = \begin{cases} T^\text{post}_{R,\ell} MG[P_{\ell}^T A_{\ell} P_{\ell}, R_{\ell}(b_{\ell} - A_{\ell} T^\text{pre}_{R,\ell} 0), \ell - 1], & \ell > \ell_0, \\ A_{\ell}^{-1} b_{\ell}, & \ell = \ell_0. \end{cases}$$

![Fig. 5.3 Illustration of the recursive multigrid strategy related to a sequence of refined triangulations](image-url)
Here $T_{R_\ell}^w w_\ell^0$ denotes the application of $v$ Richardson iterations on the $\ell$-th level with starting value $w_\ell^0$, and $P_\ell$ and $R_\ell$ are the transfer operators between the levels $\ell - 1$ and $\ell$.

Figure 5.4 shows a MATLAB realization of the algorithm. Since the initial triangulation has no interior nodes, we use an exact solution of the linear system of equations on the second level, i.e., we use $\ell_0 = 2$. Figure 5.5 shows the decay of the iteration error for different choices of the smoothing parameters $\nu_{\text{pre}}$ and $\nu_{\text{post}}$.

```matlab
function multigrid(d_tmp,L)
    global P d; d = d_tmp;
    [c4n,n4e,Db,Nb] = triang_cube(d); Db = [Db;Nb]; Nb = [];
    nC = size(c4n,1); fNodes_prev = setdiff(1:nC,unique(Db));
    for ell = 1:L
        [c4n,n4e,Db,Nb,P1] = red_refine(c4n,n4e,Db,Nb);
        nC = size(c4n,1); fNodes = setdiff(1:nC,unique(Db));
        P{ell} = P1(fNodes,fNodes_prev);
        fNodes_prev = fNodes;
    end
    u = zeros(nC,1);
    [s,m] = fe_matrices(c4n,n4e);
    b = m(fNodes,:)*f(c4n);
    A = s(fNodes,fNodes);
    u(fNodes) = MG(A,b,L);
    show_p1(c4n,n4e,Db,Nb,u)
end

function u = MG(A,b,ell)
    global P;
    nu_pre = 2; nu_post = 2; ell_0 = 2;
    if ell == ell_0
        u = A\b;
    else
        u_ini = zeros(size(b,1),1);
        u = richardson(A,b,u_ini,nu_pre,ell);
        A_coarse = P{ell}'*A*P{ell};
        r_coarse = P{ell}'*(b-A*u);
        c = MG(A_coarse,r_coarse,ell-1);
        u = u+P{ell}*c;
        u = richardson(A,b,u,nu_post,ell);
    end
end

function u = richardson(A,b,u,nu,ell)
    global d;
    h = 2^(-ell); omega = h^(d-2)/10;
    for k = 1:nu
        u = u-omega*(A*u-b);
    end
end

function val = f(x)
    val = ones(size(x,1),1);
end
```

**Fig. 5.4** MATLAB implementation of the multigrid algorithm
Fig. 5.5 Solution error $\| \nabla (u_h - u_{\text{mg}}) \|_{L^2(\Omega)}$ for different pre- and postsmoothing strategies in the multigrid solution of a two-dimensional Poisson problem with 3969 degrees of freedom.

Remark 5.4 On every level we expect that the error be decreased by the factor $v_{\text{pre}}^{-1}$. Assuming that $L = \log_2(h_L^{-1})$, e.g., $\gamma = 1/2$ and $h_1 = 1/2$, we deduce that we have a total reduction by

$$v_{\text{pre}}^{-L} = 2^{\log_2(v_{\text{pre}}) \log_2(h_L)} = h_L^{\log_2(v_{\text{pre}})}.$$  

Hence, for $v_{\text{pre}} = 4$, we expect an iteration error $O(h_L^2)$. Noting that $n_\ell = h_{\ell}^{-d} \sim 2^{\ell d} = n_L 2^{-(\ell - L)d}$, and that the computational complexity is of order $O(n_\ell)$ on the $\ell$-th level, we see that the total computational complexity is of the order

$$n_L \sum_{\ell=1}^{L} 2^{-(\ell - L)d} = O(n_L).$$

This heuristic argument can be made rigorous, and shows the optimality of the multigrid method, i.e., that it provides an approximation with an iteration error that matches the discretization error with linear computational complexity.

5.2 Domain Decomposition Methods

5.2.1 Transmission Conditions

To avoid large matrices in solving partial differential equations, it is desirable to decompose a domain into subdomains and solve local problems in those
subdomains. We thus consider a bounded Lipschitz domain $\Omega \subset \mathbb{R}^d$, and assume that we are given subsets $\Omega_1, \Omega_2 \subset \Omega$ which are Lipschitz domains with

$$
\Omega = \overline{\Omega}_1 \cup \overline{\Omega}_2, \quad \overline{\Omega}_1 \cap \overline{\Omega}_2 = \emptyset.
$$

Accordingly we partition the boundary of $\Omega$ and define the interface $\gamma$ between the subdomains, i.e.,

$$
\gamma = \partial \Omega_1 \cap \partial \Omega_2, \quad \Gamma_j = \partial \Omega \cap \partial \Omega_j
$$

for $j = 1, 2$, cf. Fig. 5.6. Throughout this section we follow [5].

To reduce the solution of a partial differential equation to problems on the subdomains $\Omega_1$ and $\Omega_2$, certain weak continuity conditions on the interface have to be satisfied. We assume for simplicity that the Poisson problem in $\Omega$ is $H^2$-regular.

**Proposition 5.2 (Transmission Conditions)** Let $f \in L^2(\Omega)$ and $u \in H^2(\Omega) \cap H^1_0(\Omega)$. We have

$$
-\Delta u = f \text{ in } \Omega, \quad u|_{\partial \Omega} = 0,
$$

if and only if the functions $u_j = u|_{\Omega_j}, j = 1, 2$, solve

$$
-\Delta u_j = f \text{ in } \Omega_j, \quad u_j|_{\Gamma_j} = 0,
$$

for $j = 1, 2$, and satisfy the transmission conditions

$$
u_1 = u_2, \quad \partial_{n_1} u_1 = -\partial_{n_2} u_2,
$$

on $\gamma$, where $\partial_{n_j} u_j = \nabla u_j \cdot n_j$ on $\gamma$ with the outer unit normal $n_j$ to $\partial \Omega_j$, i.e., $n_2 = -n_1$.

**Proof** Exercise.

**Remarks 5.5**

(i) If $d = 1$ and $\gamma = \{a\}$, then the conditions require that $u_1(a) = u_2(a)$ and $u_1'(a) = u_2'(a)$. Both conditions are needed, cf. Fig. 5.7.

(ii) Note that at every point on the boundary of a domain we can only impose one condition within a second-order elliptic partial differential equation.
5.2 Domain Decomposition Methods

5.2.2 Dirichlet–Neumann Method

The transmission condition couples the local problems in the subdomains. We will alternately solve the problems with appropriately chosen boundary conditions on the interface to satisfy the transmission conditions.

**Algorithm 5.2 (Dirichlet–Neumann Method)**

Choose $\lambda^0 \in C(\gamma)$ with $\lambda^0|_{\gamma \cap \Omega} = 0$, parameters $\theta > 0$, $\epsilon_{\text{stop}} > 0$, and set $k = 0$.

1. Determine $u_1^{k+1} \in H^1(\Omega_1)$ such that

   $-\Delta u_1^{k+1} = f$ in $\Omega_1, \quad u_1^{k+1}|_{\Gamma_1} = 0, \quad u_1^{k+1}|_{\gamma} = \lambda^k$.

2. Determine $u_2^{k+1} \in H^1(\Omega_2)$ such that

   $-\Delta u_2^{k+1} = f$ in $\Omega_2, \quad u_2^{k+1}|_{\Gamma_2} = 0, \quad \partial_n u_2^{k+1}|_{\gamma} = -\partial_n u_1^{k+1}|_{\gamma}$.

3. Stop if $\|u_1^{k+1} - u_2^{k+1}\|_{L^2(\gamma)} \leq \epsilon_{\text{stop}}$; otherwise, set

   $\lambda^{k+1} = \theta u_2^{k+1} + (1 - \theta)\lambda^k$,

   set $k \rightarrow k + 1$ and continue with (1).

A simple implementation of the Dirichlet–Neumann method is shown in Fig. 5.8, where we used the stronger stopping criterion $\|u_1^{k+1} - u_2^{k+1}\|_{L^\infty(\gamma)} \leq \epsilon_{\text{stop}}$. If the iteration becomes stationary, then the functions $u_1$ and $u_2$ coincide in the subdomains with the global solution of the Poisson problem. Figure 5.9 shows some iterates of the method in a two-dimensional numerical experiment.

**Lemma 5.4 (Consistency)** Every stationary pair $(u_1, u_2)$ for the Dirichlet–Neumann method coincides with the solution $u \in H^1_0(\Omega)$ of the Poisson problem with $f$ on the right-hand side, i.e., $u_j = u|_{\Omega_j}$ for $j = 1, 2$.

**Proof** Exercise.

The damping parameter $\theta$ has to be sufficiently small to guarantee convergence.

**Example 5.1** If $\Omega = (0, 1), f = 0$, and $\Omega_1 = (0, a), \Omega_2 = (a, 1)$ for $0 < a < 1/2$, then Algorithm 5.2 converges if and only if $\theta < 1$. 

---

Fig. 5.7 The transmission conditions are not satisfied at the interface $\gamma = \{a\}$ (left and middle); both transmission conditions are satisfied (right)
function dirichlet_neumann(red)
c4n{1} = [0 0;1 0;0 1;1 1];
c4n{2} = [1 0;2 0;1 2;1 1];
n4e{1} = [1 2 4;1 4 3];
n4e{2} = [1 2 4;1 4 3];
Db{1} = [1 2;4 3;3 1];
Db{2} = [1 2;2 4;4 3];
for j = 1:red
    [c4n{1},n4e{1},Db{1}] = red_refine(c4n{1},n4e{1},Db{1},[]);
    [c4n{2},n4e{2},Db{2}] = red_refine(c4n{2},n4e{2},Db{2},[]);
end
for j = 1:2
    nC{j} = size(c4n{j},1);
dNodes{j} = unique(Db{j});
    [s{j},m{j}] = fe_matrices(c4n{j},n4e{j});
    b{j} = m{j}*f(c4n{j});
end
 [~,gamma{1},gamma{2}] = intersect(c4n{1},c4n{2},'rows');
dNodes{1} = union(dNodes{1},gamma{1});
for j = 1:2
    fNodes{j} = setdiff(1:nC{j},dNodes{j});
end
lambda = zeros(length(gamma{1}),1);
theta = 1/4;
eps_stop = 1e-3; diff = 1;
while diff > eps_stop
    %% Initialize
    u{1} = zeros(nC{1},1); u{2} = zeros(nC{2},1);
    %% Step (1)
    u{1}(gamma{1}) = lambda;
    b1 = b{1}-s{1}*u{1};
    u{1}(fNodes{1}) = s{1}(fNodes{1},fNodes{1})\b1(fNodes{1});
    %% Step (2)
    b2 = b{2};
    normal_trans = b{1}-s{1}*u{1};
    b2(gamma{2}) = b2(gamma{2})+normal_trans(gamma{1});
    u{2}(fNodes{2}) = s{2}(fNodes{2},fNodes{2})\b2(fNodes{2});
    %% Step (3)
    lambda = theta*u{2}(gamma{2})+(1-theta)*lambda;
    diff = max(abs((u{1}(gamma{1})-u{2}(gamma{2}))))
    %% Visualize
    show_p1(c4n{1},n4e{1},Db{1},[],u{1}); hold on;
    show_p1(c4n{2},n4e{2},Db{2},[],u{2}); hold off;
    pause(1);
end

function val = f(x); val = ones(size(x,1),1);

Fig. 5.8  MATLAB implementation of the Dirichlet–Neumann-method
Various other aspects have to be taken into account in the practical realization of the Dirichlet–Neumann-method.

**Remarks 5.6**

(i) If we replace the boundary condition \( \partial_{n_1} u_1^{k+1} = -\partial_{n_2} u_2^{k+1} \) on \( \gamma \) in Step (2) by \( \partial_{n_1} u_1^{k+1} = -\partial_{n_2} u_2^{k} \), then the problems in Steps (1) and (2) can be carried out in parallel.

(ii) Notice that we need \( \Gamma_2 = \partial \Omega_2 \cap \partial \Omega \neq \emptyset \) in order to have a well-posed problem in Step (2), i.e., \( \Omega_2 \) should be a nonfloating domain.

(iii) The Dirichlet–Neumann-method can be generalized to partitions with more than two nonfloating, nonoverlapping subdomains.

(iv) The normal derivative of \( u_1^{k+1} \) on \( \gamma \) does not have to be computed explicitly in Step (2), since for every \( v \in H^1(\Omega_2) \) with \( v|_{\Gamma_2} = 0 \) and an arbitrary extension \( \tilde{v} \in H^1_0(\Omega) \), i.e., \( \tilde{v}|_{\Omega_2} = v \), we have

\[
\int_{\gamma} \partial_{n_1} u_1^{k+1} v \, ds = \int_{\Omega_1} \nabla u_1^{k+1} \cdot \nabla \tilde{v} \, dx - \int_{\Omega_1} f \tilde{v} \, dx.
\]
To prove convergence of the Dirichlet–Neumann iteration, we derive a single equation on the interface. For this we use the space

\[ H^{1/2}_{00}(\gamma) = \{ v|_\gamma : v \in H^1_0(\Omega) \} \]

which consists of all functions \( \psi \in L^1(\gamma) \) whose trivial extensions to \( \partial \Omega_j \) coincide with the trace of a function in \( H^1(\Omega_j) \) for \( j = 1, 2 \). In general it is strictly included in the space \( H^{1/2}_{0j}(\Omega_j) = \{ v|_{\Gamma_j} : v \in H^1(\Omega_j), v|_{\partial\Gamma_j} = 0 \} \). We also define for \( j = 1, 2 \), the spaces

\[ H^{1/2}_{1j}(\Omega_j) = \{ v \in H^1(\Omega_j) : v|_{\Gamma_j} = 0 \}. \]

For \( \psi \in H^{1/2}_{00}(\gamma) \), we let for \( j = 1, 2 \) the function \( v_j \in H^{1/2}_{1j}(\Omega_j) \) be the unique weak solution of

\[ -\Delta v_j = 0 \text{ in } \Omega_j, \quad v_j|_{\Gamma_j} = 0, \quad v_j|_\gamma = \psi. \]

This defines linear operators \( H_j : H^{1/2}_{00}(\gamma) \to H^{1/2}_{1j}(\Omega_j) \) by \( \psi \mapsto H_j \psi = v_j \) called the harmonic extensions of \( \psi \) to \( \Omega_j \).

**Lemma 5.5 (Norm Equivalence)** The expressions \( \| \psi \|_j = \| \nabla H_j \psi \|_{L^2(\Omega_j)} \), \( j = 1, 2 \), define equivalent norms on \( H^{1/2}_{00}(\gamma) \), i.e., there exists \( c_0 \geq 1 \), such that for all \( \psi \in H^{1/2}_{00}(\gamma) \) we have

\[ c_0^{-1} \| \psi \|_2 \leq \| \psi \|_1 \leq c_0 \| \psi \|_2. \]

**Proof** Exercise. \( \square \)

To simplify notation, we define

\[ a_j(v, w) = \int_{\Omega_j} \nabla v \cdot \nabla w \, dx, \quad b_j(w) = \int_{\Omega_j} fw \, dx. \]

We then have that \( u^{k+1}_1 \in H^1_{11}(\Omega_1) \) satisfies

\[ a_1(u^{k+1}_1, v_1) = b_1(v_1), \quad u^{k+1}_1|_\gamma = \lambda^k, \]

for all \( v_1 \in H^1_0(\Omega_1) \). Moreover, we have that \( u^{k+1}_2 \in H^1_{12}(\Omega_2) \) satisfies

\[ a_2(u^{k+1}_2, v_2) = b_2(v_2) - a_1(u^{k+1}_2, H_1 v_2|_\gamma) + b_1(H_1 v_2|_\gamma) \]

for all \( v_2 \in H^1_{12}(\Omega_2) \), where we used the harmonic extension of \( v_2|_{\gamma} \) to \( \Omega_1 \) to incorporate the Neumann condition on \( \gamma \). We let \( u \in H^1_0(\Omega) \) be the solution of the
Poisson problem with \( f \) on the right-hand side, i.e., \( u \) satisfies

\[
a_2(u, v_2) = b_1(H_1 v_2|_\gamma) + b_2(v_2) - a_1(u, H_1 v_2|_\gamma)
\]

for all \( v_2 \in H^1_{H_2}(\Omega_2) \) with harmonic extension \( H_1 v_2|_\gamma \in H^1_{H_1}(\Omega_1) \) to \( \Omega_1 \).

**Lemma 5.6 (Interface Equation)** Define \( T : H^{1/2}_{00}(\gamma) \to H^{1/2}_{00}(\gamma) \) by \( T\psi = w_2|_\gamma \), where \( w_2 \in H^1_{H_2}(\Omega_2) \) solves

\[
-\Delta w_2 = 0 \text{ in } \Omega_2, \quad w_2|_{\Gamma_2} = 0, \quad \partial_{n_2} w_2|_\gamma = -\partial_{n_1} H_1 \psi|_\gamma.
\]

For the interface error \( \delta^k = \lambda^k - u|_\gamma \) and \( k = 0, 1, \ldots \), we then have that

\[
\delta^{k+1} = \theta T \delta^k + (1 - \theta) \delta^k.
\]

**Proof** For \( j = 1, 2 \) set \( e_j^{k+1} = u_j^{k+1} - u|_{\Omega_2} \). We then have that \( a_1(e_1^{k+1}, v_1) = 0 \) for all \( v_1 \in H^1_0(\Omega_1) \), i.e.,

\[
e_1^{k+1} = H_1 \delta^k.
\]

Moreover, for all \( v_2 \in H^1_{H_2}(\Omega_2) \) we have that

\[
a_2(e_2^{k+1}, v_2) = -a_1(e_1^{k+1}, H_1 v_2|_\gamma).
\]

The combination of identities yields

\[
a_2(e_2^{k+1}, v_2) = -a_1(H_1 \delta^k, H_1 v_2|_\gamma)
\]

for all \( v_2 \in H^1_{H_2}(\Omega_2) \), which is equivalent to

\[
T \delta^k = e_2^{k+1}|_\gamma.
\]

Together with the error equation on the interface, which follows from subtracting \( u|_\gamma \) from the equation for \( \lambda^{k+1} \) in Algorithm 5.2,

\[
\delta^{k+1} = \theta e_2^{k+1}|_\gamma + (1 - \theta) \delta^k,
\]

we deduce the asserted identity. \( \square \)

Note that \( w_2 \in H^1_{H_2}(\Omega_2) \) coincides with the harmonic extension of its own boundary data \( w_2 \) on \( \gamma \), which by definition of \( T \) is the same as \( T\psi \), so that

\[
a_2(H_2 T\psi, v_2) = -a_1(H_1 \psi, H_1 v_2|_\gamma)
\]

for all \( v_2 \in H^1_{H_2}(\Omega_2) \). For sufficiently small \( \theta \), the equation on the interface defines a contraction.
Theorem 5.3 (Contraction) There exists $\theta^* > 0$ such that, for $0 < \theta < \theta^*$, the linear operator $T_\theta = \theta T + (1 - \theta) \text{id} : H^{1/2}_0(\gamma) \to H^{1/2}_0(\gamma)$ is a contraction.

Proof From the preceding identity for $a_2(H_2 T \psi, v_2)$ with $v_2 = H_2 T \psi$, we find that

$$a_1(H_1 \psi, H_1 T \psi) = -a_2(H_2 T \psi, H_2 T \psi) = -\|\nabla H_2 T \psi\|^2_{L^2(\Omega_2)} = -\|T \psi\|^2,$$

where we used that $H_2 T \psi|_y = T \psi$ and hence $H_1(H_2 T \psi)|_y = H_1 T \psi$. Using the equivalence of $\| \cdot \|_1$ and $\| \cdot \|_2$, we deduce that

$$\|T \psi\|^2_2 = -a_1(H_1 \psi, H_1 T \psi) \leq \|\nabla H_1 \psi\|_{L^2(\Omega_1)} \|\nabla H_1 T \psi\|_{L^2(\Omega_1)}$$

$$= \|\psi\|_1 \|T \psi\|_1 \leq c_0 \|\psi\|_1 \|T \psi\|_2,$$

which implies that

$$\|T \psi\|_2 \leq c_0 \|\psi\|_1$$

and

$$\|T \psi\|_1 \leq c_0 \|T \psi\|_2 \leq c_0^2 \|\psi\|_1.$$ 

Similarly, with the identity for $a_2(H_2 T \psi, v_2)$ we find that

$$\|\psi\|^2_1 = a_1(H_1 \psi, H_1 \psi) = -a_2(H_2 T \psi, H_2 \psi)$$

$$\leq \|\nabla H_2 T \psi\|_{L^2(\Omega_2)} \|\nabla H_2 \psi\|_{L^2(\Omega_2)}$$

$$= \|T \psi\|_2 \|\psi\|_2 \leq c_0 \|T \psi\|_2 \|\psi\|_1,$$

i.e.,

$$\|T \psi\|_2 \geq c_0^{-1} \|\psi\|_1.$$ 

We can now estimate the norm of the operator $T_\theta$. We have that

$$\|T_\theta \psi\|^2_1 = \theta^2 \|T \psi\|^2_1 + (1 - \theta)^2 \|\psi\|^2_1 + 2\theta(1 - \theta)a_1(H_1 T \psi, H_1 \psi)$$

$$= \theta^2 \|T \psi\|^2_1 + (1 - \theta)^2 \|\psi\|^2_1 - 2\theta(1 - \theta)\|T \psi\|^2_2$$

$$\leq \theta^2 \|T \psi\|^2_1 + (1 - \theta)^2 \|\psi\|^2_1 - 2c_0^{-2} \theta(1 - \theta)\|\psi\|^2_1$$

$$\leq c_0^4 \theta^2 \|\psi\|^2_1 + (1 - \theta)^2 \|\psi\|^2_1 - 2c_0^{-2} \theta(1 - \theta)\|\psi\|^2_1$$

$$= (c_0^4 \theta^2 + (1 - \theta)^2 - 2c_0^{-2} \theta(1 - \theta)) \|\psi\|^2_1$$

$$= K_0^2 \|\psi\|^2_1.$$
A straightforward calculation shows that $K_\theta < 1$ if
\[
\theta < \theta^* = \frac{2(1 + c_0^{-2})}{1 + c_0^2 + 2c_0^{-2}},
\]
which proves the contraction property. \[\square\]

The theorem implies that the functions $u_1^k \mid \gamma = \lambda^{k-1}$ converge in $H^{1/2}_0(\gamma)$. It remains to show that also the functions $u_1^k$ and $u_2^k$ converge. Then it follows from Lemma 5.4 that the limits $u_1$ and $u_2$ coincide with the solution restricted to $\Omega_1$ and $\Omega_2$, respectively.

**Lemma 5.7 (Convergence)** Suppose that the sequence $(u_1^k \mid \gamma)_{k \in \mathbb{N}}$ converges in $H^{1/2}_0(\gamma)$. Then for $j = 1, 2$, the sequence $(u_j^k)_{k \in \mathbb{N}}$ converges in $H^1(\Omega_j)$ to $u \mid \Omega_j$ as $k \to \infty$.

**Proof** Since $u_1^k - u_1^j = H_1(u_1^k - u_1^j) \mid \gamma$ we have that
\[
\|\nabla(u_1^k - u_1^j)\|_{L^2(\Omega_1)} = \|\nabla H_1(u_1^k - u_1^j) \mid \gamma\|_{L^2(\Omega_1)} = \|u_1^k - u_1^j\|_\gamma,
\]
and hence, $(u_1^k)_{k \in \mathbb{N}}$ is a Cauchy sequence in $H^1(\Omega_1)$. The identities $u_1^{k+1} \mid \gamma = \lambda^k$ and $\lambda^k = \theta u_2^k + (1 - \theta)\lambda^{k-1}$ imply that
\[
\theta u_2^k \mid \gamma = (u_1^{k+1} - u_1^k) \mid \gamma + \theta u_1^k \mid \gamma,
\]
so that the sequence $(u_2^k \mid \gamma)_{k \in \mathbb{N}}$ has the same limit as $(u_1^k \mid \gamma)_{k \in \mathbb{N}}$. Using that
\[
\|\nabla(u_2^k - u_2^j)\|^2_{L^2(\Omega_2)} = -a_1(u_1^k - u_1^j, H_1(u_2^k - u_2^j) \mid \gamma)
\leq \|\nabla(u_1^k - u_1^j)\|_{L^2(\Omega_1)} \|u_2^k - u_2^j\|_\gamma,
\]
we find that $(u_2^k)_{k \in \mathbb{N}}$ is a Cauchy sequence in $H^1(\Omega_2)$. \[\square\]

### 5.2.3 Overlapping Schwarz Method

The overlapping Schwarz method was introduced to establish the existence of solutions for partial differential equations on domains that are representable as unions of simple domains such as disks and rectangles on which the equation can be solved analytically. We assume that
\[
\Omega = \Omega_1 \cup \Omega_2
\]
such that the overlap region $\Omega_{12} = \Omega_1 \cap \Omega_2$ is a Lipschitz domain, and set
\[
\gamma_j = \partial \Omega_j \cap \partial \Omega_{12}, \quad I_j = \partial \Omega_j \cap \partial \Omega
\]
for $j = 1, 2$, cf. Fig. 5.10.
The function \( u \in H^1_0(\Omega) \) solves the Poisson problem with \( f \in L^2(\Omega) \) on the right-hand side if and only if the functions \( u_j = u|_{\Omega_j} \in H^1(\Omega_j) \) solve

\[-\Delta u_j = f \text{ in } \Omega_j, \quad u_j|_{\Gamma_j} = 0\]

for \( j = 1, 2 \) and satisfy \( u_1 = u_2 \) in \( \Omega_{12} \). In an iterative algorithm, we use the trace of the solution in \( \Omega_2 \) on \( \gamma_1 \) to define the Dirichlet data for problems in \( \Omega_1 \) and vice versa.

**Algorithm 5.3 (Alternating Schwarz Method)** Choose \( u^0 \in H^1_0(\Omega) \), define \( u_j^0 = u^0|_{\Omega_j} \) for \( j = 1, 2 \), let \( \varepsilon_{\text{stop}} > 0 \), and set \( k = 0 \).

1. Determine \( u_1^{k+1} \in H^1(\Omega_1) \) with
   \[-\Delta u_1^{k+1} = f \text{ in } \Omega_1, \quad u_1^{k+1}|_{\Gamma_1} = 0, \quad u_1^{k+1}|_{\gamma_1} = u_2^{k+1}|_{\gamma_1}.
   
2. Determine \( u_2^{k+1} \in H^1(\Omega_2) \) with
   \[-\Delta u_2^{k+1} = f \text{ in } \Omega_2, \quad u_2^{k+1}|_{\Gamma_2} = 0, \quad u_2^{k+1}|_{\gamma_2} = u_1^{k+1}|_{\gamma_2}.

3. Stop if \( \|\nabla (u_1^{k+1} - u_2^{k+1})\|_{L^2(\Omega_{12})} \leq \varepsilon_{\text{stop}} \); set \( k = k + 1 \) and continue with Step (1).

No numerical parameter is needed in the algorithm but the speed of convergence depends on the radius of the inner circle of the overlap region, cf. Fig. 5.11.

![Fig. 5.10 Overlapping partition of \( \Omega \) with overlap region \( \Omega_{12} \)](image)

![Fig. 5.11 The speed of convergence depends on the diameter of the overlap region \( \Omega_{12} \)](image)
Remark 5.7 If Steps (1) and (2) are decoupled by using the boundary condition $u_2^{k+1}|_{j_2} = u_1^k|_{j_2}$ in Step (2), then the algorithm can be parallelized.

To analyze the algorithm, we transform the boundary conditions and rewrite the problems in weak form. For this we use the spaces

$$V = H^1_0(\Omega), \quad V_j = \{ v \in V : v = 0 \text{ on } \Omega \setminus \Omega_j \},$$

i.e., the functions in $H^1_0(\Omega_j)$ that are extended trivially to the whole domain $\Omega$ for $j = 1, 2$. We can then consider the bilinear form $a$ and the linear form $b$ defined on $V \times V$ and $V$ by

$$a(v, w) = \int_{\Omega} \nabla v \cdot \nabla w \, dx, \quad b(v) = \int_{\Omega} f v \, dx.$$  

Instead of the functions $u_1$ and $u_2$, we compute $w_1$ and $w_2$ with homogeneous boundary data, i.e., given some $u_0 \in V$, we set $k = 0$ and compute $w_1^k \in V_1$ such that

$$a(w_1^k, v_1) = b(v_1) - a(u_1^k, v_1)$$

for all $v_1 \in V_1$. We then set $u^{k+1/2} = u^k + w_1^k$ and determine $w_2^k \in V_2$ such that

$$a(w_2^k, v_2) = b(v_2) - a(u^{k+1/2}, v_2)$$

for all $v_2 \in V_2$. The new iterate is $u^{k+1} = u^{k+1/2} + w_2^k$. These two steps are iterated until we have $\| \nabla (u^k - u^{k+1}) \|_{L^2(\Omega)} \leq \varepsilon_{\text{stop}}$.

**Lemma 5.8 (Iterated Projection)** For $j = 1, 2$, let $\mathcal{P}_j : V \to V_j$ denote the orthogonal projection onto $V_j$ with respect to the scalar product $a(\cdot, \cdot)$. We then have that

$$u - u^{k+1} = (I - \mathcal{P}_2)(I - \mathcal{P}_1)(u - u^k).$$

**Proof** Using that $a(u, v_1) = b(v_1)$ for all $v_1 \in V_1$, we have

$$a(u^{k+1/2} - u^k, v_1) = a(w_1^k, v_1) = a(u - u^k, v_1),$$

which is equivalent to $u^{k+1/2} - u^k = \mathcal{P}_1(u - u^k)$. Similarly, the identity

$$a(u^{k+1} - u^{k+1/2}, v_2) = a(w_2^k, v_2) = a(u - u^{k+1/2}, v_2)$$

shows that $u^{k+1} - u^{k+1/2} = \mathcal{P}_2(u - u^{k+1/2})$. We thus have that

$$u - u^{k+1/2} = u - u^k + u^k - u^{k+1/2} = (I - \mathcal{P}_1)(u - u^k)$$
and

\[ u - u^{k+1} = u - u^{k+1/2} + u^{k+1/2} - u^k = (I - \mathcal{P}_2)(u - u^{k+1/2}). \]

The combination of the two identities proves the result. \( \square \)

**Remark 5.8** The lemma explains the terminology *multiplicative iteration*. For the decoupled version of the algorithm one can prove that

\[ u - u^{k+1} = (I - \mathcal{P}_2 - \mathcal{P}_1)(u - u^k), \]

which motivates the terminology *additive iteration*.

To prove convergence of the alternating Schwarz method, we use the following result.

**Lemma 5.9** There exists \( c_1 \geq 1 \) such that for all \( v \in V \), we have

\[ \| \nabla v \| \leq c_1 \left( \| \nabla \mathcal{P}_1 v \|^2 + \| \nabla \mathcal{P}_2 v \|^2 \right)^{1/2}. \]

**Proof** For \( v \in V \) let \( H_{12}v|_{\gamma_2} \) denote the harmonic extension of \( v|_{\gamma_2} \) to \( \Omega_{12} \), such that \( H_{12}v \) vanishes on \( \gamma_1 \). Let \( v_1 \in V_1 \) be defined by \( v_1 = v \) in \( \Omega_1 \setminus \Omega_{12} \) and \( v_1 = H_{12}v|_{\gamma_2} \) in \( \Omega_{12} \). Let \( v_2 \in V_2 \) be defined by \( v_2 = v \) in \( \Omega_2 \setminus \Omega_{12} \) and \( v_2 = v - H_{12}v|_{\gamma_2} \) in \( \Omega_{12} \). Then \( v = v_1 + v_2 \) and

\[ \| \nabla v_1 \|_{L^2(\Omega_1)} + \| \nabla v_2 \|_{L^2(\Omega_2)} \leq c_1 \| \nabla v \|_{L^2(\Omega)}. \]

The Cauchy–Schwarz inequality implies that

\[
\begin{align*}
\| \nabla v \|^2_{L^2(\Omega)} &= a(v_1, v) + a(v_2, v) \\
&= a(v_1, \mathcal{P}_1 v) + a(v_2, \mathcal{P}_2 v) \\
&\leq \| \nabla v_1 \|_{L^2(\Omega_1)} \| \nabla \mathcal{P}_1 v \|_{L^2(\Omega_1)} + \| \nabla v_2 \|_{L^2(\Omega_2)} \| \nabla \mathcal{P}_2 v \|_{L^2(\Omega_2)} \\
&\leq \left( \| \nabla v_1 \|^2_{L^2(\Omega_1)} + \| \nabla v_2 \|^2_{L^2(\Omega_2)} \right)^{1/2} \left( \| \nabla \mathcal{P}_1 v \|^2_{L^2(\Omega_1)} + \| \nabla \mathcal{P}_2 v \|^2_{L^2(\Omega_2)} \right)^{1/2} \\
&\leq c_1 \| \nabla v \|_{L^2(\Omega)} \left( \| \nabla \mathcal{P}_1 v \|^2_{L^2(\Omega_1)} + \| \nabla \mathcal{P}_2 v \|^2_{L^2(\Omega_2)} \right)^{1/2}
\end{align*}
\]

and this proves the estimate. \( \square \)

We are now in position to verify convergence of the alternating Schwarz method.

**Theorem 5.4 (Convergence)** The operator \((I - \mathcal{P}_2)(I - \mathcal{P}_1)\) is a contraction, i.e., the alternating Schwarz method converges.
5.3 Preconditioning

**Proof** Since \( \mathcal{P}_1(I - \mathcal{P}_1)w = 0 \) for all \( w \in V \), from the previous lemma for \( v = (I - \mathcal{P}_1)w \) we deduce that

\[
\| \nabla (I - \mathcal{P}_1)w \|_{L^2(\Omega_1)} \leq c_1 \| \nabla \mathcal{P}_2(I - \mathcal{P}_1)w \|_{L^2(\Omega_2)}.
\]

Since \( \mathcal{P}_2 v \) and \((I - \mathcal{P}_2)v\) are orthogonal with respect to \( a(\cdot, \cdot) \), we have that

\[
\| \nabla (I - \mathcal{P}_1)w \|_{L^2(\Omega_1)}^2 = \| \nabla (I - \mathcal{P}_2)(I - \mathcal{P}_1)w \|_{L^2(\Omega_2)}^2 + \| \nabla \mathcal{P}_2(I - \mathcal{P}_1)w \|_{L^2(\Omega_2)}^2
\]

\[
\geq \| \nabla (I - \mathcal{P}_2)(I - \mathcal{P}_1)w \|_{L^2(\Omega_2)}^2 + c_1^{-2} \| \nabla (I - \mathcal{P}_1)w \|_{L^2(\Omega_1)}^2,
\]

i.e.,

\[
\| \nabla (I - \mathcal{P}_2)(I - \mathcal{P}_1)w \|_{L^2(\Omega_2)}^2 \leq (1 - c_1^{-2}) \| \nabla (I - \mathcal{P}_1)w \|_{L^2(\Omega_1)}^2
\]

\[
\leq (1 - c_1^{-2}) \| \nabla w \|_{L^2(\Omega_1)}^2.
\]

Noting \( c_1 \geq 1 \), this proves the contraction property. \( \square \)

### 5.3 Preconditioning

#### 5.3.1 Preconditioned CG Algorithm

The solution of a linear system \( Ax = b \) with the conjugate gradient method leads to the convergence result

\[
\| x^k - x \|_A \leq 2 \left( \frac{\kappa^{1/2} - 1}{\kappa^{1/2} + 1} \right)^k \| x^0 - x \|_A
\]

with \( \kappa = \text{cond}_2(A) \), provided that \( A \) is symmetric and positive definite. For the finite element stiffness matrix of the Poisson problem, we have \( \kappa \sim h^{-2} \), and a large number of iterations is required to guarantee \( \| x^j - x \|_A \leq \varepsilon \) for some given tolerance \( \varepsilon > 0 \). If we could construct an invertible matrix \( C \) such that \( \text{cond}_2(CA) \ll \text{cond}_2(A) \), then we could consider the equivalent linear system \( CAx = Cb \). The best possible choice is \( C = A^{-1} \), but then the multiplication by \( C \) would be as expensive as the direct solution of the original system. Thus a good compromise between reducing the condition number and the cost of matrix-vector multiplications has to be achieved.

**Definition 5.2** A regular matrix \( C \in \mathbb{R}^{n \times n} \) is a *preconditioner* for the regular matrix \( A \in \mathbb{R}^{n \times n} \) if the condition number of \( CA \) is asymptotically smaller than the condition number of \( A \), and the computation of matrix-vector products \( r \mapsto Cr \) is cheaper than the direct solution of \( Ax = b \).

We apply the conjugate gradient algorithm to the equation \( CAx = Cb \).
Algorithm 5.4 (PCG Algorithm) Let $A, C \in \mathbb{R}^{n \times n}$ and $b \in \mathbb{R}^n$. Let $x^0 \in \mathbb{R}^n$, $\varepsilon_{\text{stop}} > 0$, and set $r^0 = b - Ax^0$, $d^0 = z^0 = Cr^0$, $k = 0$.

1. Set $x^{k+1} = x^k + \alpha_k d^k$, $r^{k+1} = r^k - \alpha_k Ad^k$, $z^{k+1} = Cr^{k+1}$, and $d^{k+1} = z^{k+1} + \beta_k d^k$, where

$$\alpha_k = \frac{r^k \cdot z^k}{Ad^k \cdot d^k}, \quad \beta_k = \frac{r^{k+1} \cdot z^{k+1}}{r^k \cdot z^k}.$$

2. Stop if $|r^{k+1}| \leq \varepsilon_{\text{stop}}$; set $k \rightarrow k + 1$ and continue with (1) otherwise.

The standard convergence result requires $CA$ to be positive definite and symmetric, which is often difficult to guarantee. Instead, one ensures a factorization $C = KK^T$, e.g., a Cholesky factorization, and considers the equivalent linear system

$$(K^TAK)^{-1}x = K^Tb.$$ 

The factorization $C = KK^T$ is irrelevant for formulating the preconditioned CG algorithm, but could be of interest for an efficient computation of matrix-vector products.

Examples 5.2

(i) Let $D \in \mathbb{R}^{n \times n}$ denote the diagonal part of $A$ and assume that $D$ is invertible. Then $C = D^{-1}$ is called the diagonal or Jacobi preconditioner.

(ii) Define the diagonal matrix $D \in \mathbb{R}^{n \times n}$ by $d_{ii} = \sum_{j=1}^n |a_{ij}|$ for $i = 1, 2, \ldots, n$. Then $\text{cond}_\infty(D^{-1}A) \leq \text{cond}_\infty(A)$ and $C = D^{-1}$ is called an equilibration preconditioner.

(iii) If $x^{k+1} = x^k - M(Ax^k - b)$ is a convergent iteration, then often $C = M$ defines a preconditioner, e.g., if $M$ is the inverse of the lower triangular part of $A$, then $M$ is called the Gauss–Seidel preconditioner. A symmetric factorization is achieved by considering

$$C = [(D + L)D^{-1}(D + L)^T]^{-1},$$

where $D$ and $L$ denote the diagonal and lower triangular parts of $A$, and the inversion is understood in the sense of successive elimination.

(iv) Let $A = LL^T$ be the Cholesky factorization of $A$ and define $\widetilde{L} \in \mathbb{R}^{n \times n}$ such that the population pattern of $A$ is preserved, i.e.,

$$\widetilde{L}_{ij} = \begin{cases} L_{ij} & \text{if } A_{ij} \neq 0, \\ 0 & \text{else.} \end{cases}$$

Then $A = \widetilde{L}\widetilde{L}^T + E$ with an error term $E \in \mathbb{R}^{n \times n}$ and $C = (\widetilde{L}\widetilde{L}^T)^{-1}$ is called the incomplete Cholesky preconditioner.
5.3 Preconditioning

These blackbox preconditioners often lead to an improvement in the performance of the conjugate gradient algorithm. Whether they are preconditioners in the sense of Definition 5.2 depends on the specific properties of the problem under consideration.

Remark 5.9 The construction of preconditioners is closely related to certain norm equivalences, i.e., if $c_1\|x\|_{C^{-1}} \leq \|x\|_{A} \leq c_2\|x\|_{C^{-1}}$ for all $x \in \mathbb{R}^n$, where $\|x\|_B^2 = Bx \cdot x$, then we have $\text{cond}_2(CA) \leq c_2/c_1$.

5.3.2 Abstract Multilevel Preconditioner

We follow [5] and assume that we are given a sequence of nested finite-dimensional spaces

$$V_0 \subset V_1 \subset \cdots \subset V_L = V$$

with increasing dimensions $n_\ell = \dim(V_\ell)$ and $n = \dim(V)$. We assume that for $\ell = 0, 1, \ldots, L$, we are given injective linear operators

$$P_\ell : V_\ell \to V.$$

Moreover, we assume that for each space $V_\ell$ we are given a basis, and accordingly identify operators $T : V_\ell \to V_j$ with matrices $T \in \mathbb{R}^{n_j \times n_\ell}$.

Definition 5.3 For symmetric, positive definite matrices $B_\ell \in \mathbb{R}^{n_\ell \times n_\ell}$, $\ell = 0, 1, \ldots, L$, a multilevel preconditioner $C \in \mathbb{R}^{n \times n}$ is defined by

$$C = \sum_{\ell=0}^{L} P_\ell B_\ell^{-1} P_\ell^T.$$

To justify $C$ as a preconditioner, we first show that it is symmetric and positive definite.

Lemma 5.10 (Symmetry and Definiteness) The multilevel preconditioner is symmetric and positive definite.

Proof The symmetry of $C$ is a direct consequence of its definition. The definiteness of $B_\ell^{-1}$, $\ell = 0, \ldots, L$, implies that for every $v \in V$, we have

$$v \cdot Cv = \sum_{\ell=0}^{L} (P_\ell^T v) \cdot (B_\ell^{-1} P_\ell^T v) \geq 0,$$

i.e., $C$ is positive semidefinite. Since $P_L$ is invertible, we find that $C$ is positive definite. \qed

The inverse of $C$ can be represented in terms of the matrices $B_\ell$, $\ell = 0, 1, \ldots, L$. 

Lemma 5.11 (Inverse Matrix) For every \( v \in V \) we have

\[
C^{-1}v \cdot v = \min_{P \in \mathbb{P}} \sum_{\ell=0}^{L} B_{\ell} v_{\ell} \cdot v_{\ell},
\]

where the minimum is over all tuples \((v_{\ell}) = (v_{0}, v_{1}, \ldots, v_{L}) \in \prod_{\ell=0}^{L} V_{\ell}\) with \( v = \sum_{\ell=0}^{L} P_{\ell} v_{\ell} \).

Proof Since \( B_{\ell}^{-1} \) is symmetric and positive definite, the Cauchy–Schwarz inequality implies that for \( y_{\ell}, z_{\ell} \in V_{\ell} \), we have

\[
y_{\ell} \cdot B_{\ell}^{-1} z_{\ell} \leq (y_{\ell} \cdot B_{\ell}^{-1} y_{\ell})^{1/2} (z_{\ell} \cdot B_{\ell}^{-1} z_{\ell})^{1/2}.
\]

For \( v = \sum_{\ell=0}^{L} P_{\ell} v_{\ell} \) with \( v_{\ell} \in V_{\ell} \), we thus have that

\[
C^{-1}v \cdot v = C^{-1}v \cdot \sum_{\ell=0}^{L} P_{\ell} v_{\ell} = \sum_{\ell=0}^{L} P_{\ell}^{T} C^{-1}v \cdot (B_{\ell}^{-1} B_{\ell} v_{\ell}) \\
\leq \sum_{\ell=0}^{L} \left[ (P_{\ell}^{T} C^{-1}v) \cdot B_{\ell}^{-1} (P_{\ell}^{T} C^{-1}v) \right]^{1/2} \left[ B_{\ell} v_{\ell} \cdot B_{\ell}^{-1} B_{\ell} v_{\ell} \right]^{1/2}.
\]

The Cauchy–Schwarz inequality in \( \mathbb{R}^{L+1} \) and the definition of \( C \) lead to

\[
C^{-1}v \cdot v \leq \left[ \sum_{\ell=0}^{L} (P_{\ell}^{T} C^{-1}v) \cdot B_{\ell}^{-1} (P_{\ell}^{T} C^{-1}v) \right]^{1/2} \left[ \sum_{\ell=0}^{L} B_{\ell} v_{\ell} \cdot v_{\ell} \right]^{1/2} \\
= \left[ C^{-1}v \cdot \left( \sum_{\ell=0}^{L} P_{\ell} B_{\ell}^{-1} P_{\ell}^{T} C^{-1}v \right) \right]^{1/2} \left[ \sum_{\ell=0}^{L} B_{\ell} v_{\ell} \cdot v_{\ell} \right]^{1/2} \\
= (C^{-1}v \cdot v)^{1/2} \left[ \sum_{\ell=0}^{L} B_{\ell} v_{\ell} \cdot v_{\ell} \right]^{1/2}
\]

so that

\[
C^{-1}v \cdot v \leq \sum_{\ell=0}^{L} B_{\ell} v_{\ell} \cdot v_{\ell}.
\]

To prove the assertion, we have to show that the minimum on the right-hand side equals the value on the left-hand side. For this we define for \( \ell = 0, 1, \ldots, L, \)

\[
v_{\ell} = B_{\ell}^{-1} P_{\ell}^{T} C^{-1}v.
\]
By definition of $C$, we have $v = \sum_{\ell=0}^{L} P_{\ell} v_{\ell}$. Moreover, by using the definitions of $v_{\ell}$ and $C$, we verify that
\[
\sum_{\ell=0}^{L} B_{\ell} v_{\ell} \cdot v_{\ell} = \sum_{\ell=0}^{L} C^{-1} v \cdot (P_{\ell} B_{\ell}^{-1} P_{\ell}^{T} C^{-1} v) = C^{-1} v \cdot v,
\]
which concludes the proof.

Together with the lemmas, we obtain useful characterizations of the extremal eigenvalues of a matrix product $CA$.

**Theorem 5.5 (Eigenvalues of $CA$)** Let $A \in \mathbb{R}^{n \times n}$ be symmetric and positive definite. The minimal and maximal eigenvalue of $CA$ are the extrema of the mapping
\[
v \mapsto \frac{A v \cdot v}{\min_{P_{\ell} v_{\ell} = v} \sum_{\ell=0}^{L} B_{\ell} v_{\ell} \cdot v_{\ell}}
\]
in the set of all $v \in V \setminus \{0\}$.

**Proof** The bilinear form $\langle v, w \rangle_{C^{-1}} = C^{-1} v \cdot w$ defines a scalar product on $\mathbb{R}^{n}$, and the matrix $CA$ is symmetric and positive definite with respect to this scalar product, i.e., $\langle CA v, w \rangle_{C^{-1}} = \langle v, CAw \rangle_{C^{-1}}$ and $\langle CA v, v \rangle_{C^{-1}} > 0$ for all $v, w \in V \setminus \{0\}$. Hence, the eigenvalues of $CA$ are given as the extrema of the Rayleigh quotient
\[
R(v) = \frac{\langle CA v, v \rangle_{C^{-1}}}{\langle v, v \rangle_{C^{-1}}}
\]
in the set $v \in V \setminus \{0\}$. The characterization of $\langle v, v \rangle_{C^{-1}} = C^{-1} v \cdot v$ in Lemma 5.11 implies the assertion.

**Remark 5.10** Instead of assuming that the operators $P_{\ell} : V_{\ell} \to V$ are injective, it is sufficient to assume that for every $v \in V$, there exists $(v_{\ell})_{\ell=0}^{L} \in \prod_{\ell=0}^{L} V_{\ell}$ with $v = \sum_{\ell=0}^{L} P_{\ell} v_{\ell}$.

### 5.3.3 BPX Preconditioner

We apply the abstract framework to a sequence of refined, quasiuniform triangulations $\mathcal{T}_{0}, \mathcal{T}_{1}, \ldots, \mathcal{T}_{L}$ and $\mathcal{T}_{h} = \mathcal{T}_{L}$ with associated finite element spaces
\[
\mathcal{S}_{0}^{1}(\mathcal{T}_{0}) \subset \mathcal{S}_{0}^{1}(\mathcal{T}_{1}) \subset \cdots \subset \mathcal{S}_{0}^{1}(\mathcal{T}_{L}) = \mathcal{S}_{0}^{1}(\mathcal{T}_{h}),
\]
with maximal mesh-sizes $h_{\ell} = \gamma^{\ell} h_{0}, \ell = 0, 1, \ldots, L$, for a mesh-size reduction factor $0 < \gamma < 1$, cf. Fig. 5.12. We follow [17].
It is our goal to efficiently solve the discrete Poisson problem on the finest triangulation \( \mathcal{T}_h \), i.e., to approximately determine \( u_h \in \mathcal{S}_1^0 (\mathcal{T}_h) \) with

\[
\int_\Omega \nabla u_h \cdot \nabla v_h \, dx = \int_\Omega f v_h \, dx
\]

for all \( v_h \in \mathcal{S}_1^0 (\mathcal{T}_h) \). This is equivalent to a regular linear system of equations \( Ax = b \) with a positive definite and symmetric matrix \( A \in \mathbb{R}^{n \times n} \). We assume that the Poisson problem in \( \Omega \) is \( H^2 \)-regular.

**Lemma 5.12 (Stable Decomposition)** For every \( v_h \in \mathcal{S}_1^0 (\mathcal{T}_h) \), there exists \( w_\ell \in \mathcal{S}_1^0 (\mathcal{T}_\ell) \), \( \ell = 0, 1, \ldots, L \), such that \( v_h = \sum_{\ell=0}^{L} w_\ell \) and

\[
\sum_{\ell=0}^{L} h_\ell^{-2} \| w_\ell \|_{L^2(\Omega)}^2 \leq c_{\text{dec}} \| \nabla v_h \|_{L^2(\Omega)}^2.
\]

**Proof**

(i) By the Lax–Milgram lemma there exists for every \( \ell = 0, 1, \ldots, L \), a uniquely defined function \( v_\ell = Q_\ell v_h \in \mathcal{S}_1^0 (\mathcal{T}_\ell) \) such that

\[
\int_\Omega \nabla v_\ell \cdot \nabla r_\ell \, dx = \int_\Omega \nabla v_h \cdot \nabla r_\ell \, dx
\]

for all \( r_\ell \in \mathcal{S}_1^0 (\mathcal{T}_\ell) \). Letting \( z \in H^2(\Omega) \cap H_0^1(\Omega) \) be the weak solution of the Poisson problem

\[-\Delta z = v_\ell - v_h,
\]

we find that

\[
\| v_\ell - v_h \|_{L^2(\Omega)}^2 = \int_\Omega \nabla (z - \mathcal{I}_\ell z) \cdot \nabla (v_\ell - v_h) \, dx
\]

\[
\leq \| \nabla (z - \mathcal{I}_\ell z) \|_{L^2(\Omega)} \| \nabla (v_\ell - v_h) \|_{L^2(\Omega)},
\]

where \( \mathcal{I}_\ell : H^2(\Omega) \cap H_0^1(\Omega) \to \mathcal{S}_1^0 (\mathcal{T}_\ell) \) is the nodal interpolation operator on \( \mathcal{T}_\ell \). Using interpolation estimates and the assumed \( H^2 \)-regularity

\[
\| D^2 z \|_{L^2(\Omega)} \leq c_2 \| v_\ell - v_h \|_{L^2(\Omega)}
\]
yield that
\[ \| (Q_{\ell} - \text{id}) v_h \|_{L^2(\Omega)} = \| v_\ell - v_h \|_{L^2(\Omega)} \leq c \gamma c_2 h_\ell \| \nabla (v_\ell - v_h) \|_{L^2(\Omega)}. \]

(ii) We formally define \( Q_{-1} = 0 \), and note that \( Q_L = \text{id} \) on \( S_{0}^{1}(T_L) \). Therefore, by a telescope argument, we have
\[ v_h = \sum_{\ell=0}^{L} (Q_{\ell} - Q_{\ell-1}) v_h, \]
and we define \( w_\ell = (Q_{\ell} - Q_{\ell-1}) v_h \in S_{0}^{1}(T_{\ell}) \). Noting that \( Q_{\ell} \circ Q_{j} = Q_{\ell} \) for \( \ell \leq j \), we have the operator identity
\[ Q_{\ell} - Q_{\ell-1} = (\text{id} - Q_{\ell-1}) (Q_{\ell} - Q_{\ell-1}), \]
which allows us to verify that
\[ \sum_{\ell=0}^{L} h_\ell^{-2} \| (Q_{\ell} - Q_{\ell-1}) v_h \|_{L^2(\Omega)}^2 \leq c \gamma c_2^2 \gamma^{-2} \sum_{\ell=0}^{L} \| \nabla (Q_{\ell} - Q_{\ell-1}) v_h \|_{L^2(\Omega)}^2. \]

For \( 0 \leq \ell, m \leq L \) with \( \ell \neq m \), we have that
\[ \int_{\Omega} \left[ \nabla (Q_{\ell} - Q_{\ell-1}) v_h \right] \cdot \left[ \nabla (Q_{m} - Q_{m-1}) v_h \right] \, dx = 0, \]
which implies that
\[ \| \nabla \sum_{\ell=0}^{L} (Q_{\ell} - Q_{\ell-1}) v_h \|_{L^2(\Omega)}^2 = \sum_{\ell=0}^{L} \| \nabla (Q_{\ell} - Q_{\ell-1}) v_h \|_{L^2(\Omega)}^2. \]

A combination of the estimates proves the lemma. \( \square \)

The second important ingredient for the analysis of multilevel preconditioners is a variant of the Cauchy–Schwarz inequality.

**Lemma 5.13 (Strengthened Cauchy–Schwarz Inequality)** Let \( v_\ell \in V_{\ell} \) and \( w_m \in V_m \) for \( \ell \geq m \) and assume that \( h_\ell \leq \gamma^{\ell-m} h_m \). We then have
\[ \int_{\Omega} \nabla v_\ell \cdot \nabla w_m \, dx \leq c_{\text{scs}} \gamma^{(\ell-m)/2} h_\ell^{-1} \| v_\ell \|_{L^2(\Omega)} \| \nabla w_m \|_{L^2(\Omega)}. \]

**Proof** Let \( T \in T_m \) and note that \( \Delta w_m |_{\partial T} = 0 \). An integration-by-parts yields that
\[ \int_{T} \nabla v_\ell \cdot \nabla w_m \, dx = \int_{\partial T} v_\ell (\nabla w_m \cdot n) \, ds \leq \| v_\ell \|_{L^2(\partial T)} \| \nabla w_m \|_{L^2(\partial T)}. \]
Since $\nabla w_m|_T$ is constant, we have

$$\|\nabla w_m\|_{L^2(\partial T)} \leq ch_m^{-1/2}\|\nabla w_m\|_{L^2(T)}.$$  

There exist sides $S_i$, $i = 1, 2, \ldots, I$, in the triangulation $\mathcal{T}_\ell$, such that $S_i \cup \cdots \cup S_I = \partial T$. With elements $T_i \in \mathcal{T}_\ell$ such that $S_i \subset \partial T_i$ for $i = 1, 2, \ldots, I$, cf. Fig. 5.13, and the trace inequality, combined with an inverse estimate,

$$\|v_\ell\|_{L^2(S_i)} \leq c(h_\ell^{-1/2}\|v_\ell\|_{L^2(T_i)} + h_\ell^{1/2}\|\nabla v_\ell\|_{L^2(T_i)}) \leq ch_\ell^{-1/2}\|v_\ell\|_{L^2(T_i)},$$

we find that

$$\|v_\ell\|_{L^2(\partial T)} \leq ch_\ell^{-1/2}\|v_\ell\|_{L^2(T)}.$$  

A combination of the estimates yields that

$$\int_T \nabla v_\ell \cdot \nabla w_m \, dx \leq c(h_\ell/h_m)^{1/2}\|\nabla w_m\|_{L^2(T)}h_\ell^{-1}\|v_\ell\|_{L^2(T)}.$$  

Summing this estimate over all $T \in \mathcal{T}_m$, using the Cauchy–Schwarz inequality, and incorporating $h_\ell = \gamma^{\ell-m}h_m$ proves the estimate.

The following multilevel preconditioner realizes a simultaneous step of a Richardson iteration on all levels.

**Definition 5.4** The *Bramble–Pasciak–Xu* or *BPX preconditioner* is defined by

$$C_v = \sum_{\ell=0}^L P_\ell B_\ell^{-1} P_\ell^T v,$$

where $P_\ell : \mathcal{S}_\ell^1(\mathcal{T}_\ell) \rightarrow \mathcal{S}_0^1(\mathcal{T}_h)$ is the canonical embedding operator, i.e., $P_\ell v_\ell = v_\ell$, and $B_\ell = h_\ell^{\ell-2}\text{id}_\ell$ with the identity map $\text{id}_\ell : \mathcal{S}_0^1(\mathcal{T}_\ell) \rightarrow \mathcal{S}_0^1(\mathcal{T}_\ell)$.

The BPX preconditioner is optimal in the sense of the following theorem.
Theorem 5.6 (Optimality of the BPX Preconditioner) For the BPX preconditioner $C$ of the finite element stiffness matrix $A \in \mathbb{R}^{n \times n}$, we have that
\[
\text{cond}_2(CA) \leq c_{\text{bp}}.
\]

Proof

(i) Let $v_h \in \mathcal{P}^1(\mathcal{T}_h)$ and let $v_h = \sum_{\ell=0}^L w_\ell$ be its decomposition with $w_\ell \in \mathcal{P}^1(\mathcal{T}_\ell)$ according to Lemma 5.12. A norm equivalence implies that for $\ell = 0, 1, \ldots, L$, we have
\[
c_{\text{eq}}^2 \|w_\ell\|_{L^2(\Omega)}^2 \leq \sum_{z \in \mathcal{N}_\ell} h_\ell^d |w_\ell(z)|^2 \leq c_{\text{eq}}^2 \|w_\ell\|_{L^2(\Omega)}^2.
\]
In combination with the estimate of Lemma 5.12, we find that
\[
\sum_{\ell=0}^L B_\ell w_\ell \cdot w_\ell = \sum_{\ell=0}^L h_\ell^{d-2} \sum_{z \in \mathcal{N}_\ell} |w_\ell(z)|^2 \leq c \|\nabla v_h\|_{L^2(\Omega)}^2.
\]
With the characterization of $\lambda_{\text{min}}(CA)$ of Theorem 5.5, and the identity $A v_h \cdot v_h = \|\nabla v_h\|_{L^2(\Omega)}^2$, we have
\[
\lambda_{\text{min}}(CA) = \min_{v_h \in \mathcal{P}^1(\mathcal{T}_h) \setminus \{0\}} \frac{A v_h \cdot v_h \min_{p_\ell v_\ell = v_h} \sum_{\ell=0}^L B_\ell v_\ell \cdot v_\ell}{\min_{p_\ell v_\ell = v_h} \sum_{\ell=0}^L B_\ell v_\ell \cdot v_\ell} \geq c.
\]

(ii) To derive an upper bound for the maximal eigenvalue $\lambda_{\text{max}}(CA)$, let $v_h \in \mathcal{P}^1(\mathcal{T}_h)$ and let $v_h = \sum_{\ell=0}^L v_\ell$ be an arbitrary decomposition with $v_\ell \in \mathcal{P}^1(\mathcal{T}_\ell)$ for $\ell = 0, 1, \ldots, L$. With the strengthened Cauchy–Schwarz inequality, we have that
\[
A v_h \cdot v_h \leq 2 \sum_{m=0}^L \sum_{\ell=m}^L \int_\Omega \nabla v_\ell \cdot \nabla v_m \, dx
\]
\[
\leq 2c_{\text{scs}} \sum_{m=0}^L \sum_{\ell=m}^L \gamma^{(\ell-m)/2} h_\ell^{-1} \|v_\ell\|_{L^2(\Omega)} \|\nabla v_m\|_{L^2(\Omega)}
\]
\[
\leq 2c_{\text{scs}} \sum_{\ell,m=0}^L \gamma^{(\ell-m)/2} h_\ell^{-1} \|v_\ell\|_{L^2(\Omega)} \|\nabla v_m\|_{L^2(\Omega)}.
\]
Defining $\Gamma \in \mathbb{R}^{(L+1) \times (L+1)}$ by $\Gamma_{\ell m} = \gamma^{(\ell-m)/2}$ for $\ell, m = 0, 1, \ldots, L$, we have for vectors $\alpha, \beta \in \mathbb{R}^{L+1}$, that
\[
\sum_{\ell,m=0}^L \gamma^{(\ell-m)} \alpha_\ell \beta_m = (\Gamma \alpha) \cdot \beta \leq \varrho(\Gamma) \|\alpha\| \|\beta\| \leq \frac{1}{1 - \gamma^{1/2} \|\alpha\| \|\beta\|}.
\]
where \( \rho(\Gamma') \) is the spectral norm of \( \Gamma' \). This implies that

\[
A v_h \cdot v_h \leq \frac{2c_{s_{\text{scs}}}}{1 - \gamma^{1/2}} \left( \sum_{\ell=0}^{L} h_{\ell}^{-2} \| u_{\ell} \|_{L^2(\Omega)}^2 \right)^{1/2} \left( \sum_{m=0}^{L} \| \nabla u_m \|_{L^2(\Omega)}^2 \right)^{1/2}.
\]

Incorporating the inverse estimates \( \| \nabla v_m \|_{L^2(\Omega)} \leq c h_m^{-1} \| v_m \|_{L^2(\Omega)} \), \( m = 0, 1, \ldots, L \), we deduce that

\[
A v_h \cdot v_h \leq c \sum_{\ell=0}^{L} h_{\ell}^{-2} \| u_{\ell} \|_{L^2(\Omega)}^2.
\]

We use the norm equivalence to verify that

\[
A v_h \cdot v_h \leq c \sum_{\ell=0}^{L} h_{\ell}^{-2} \sum_{z \in \mathcal{N}_\ell} | v_{\ell}(z) |^2 = c \sum_{\ell=0}^{L} B_{\ell} v_{\ell} \cdot v_{\ell}.
\]

The characterization of \( \lambda_{\text{max}}(CA) \) of Theorem 5.5 leads to

\[
\lambda_{\text{max}}(CA) = \max_{v_h \in \mathcal{S}_0^1(\mathcal{T}_h) \setminus \{0\}} \frac{A v_h \cdot v_h}{\min_{\ell \in \mathcal{L}} v_{\ell} = v_h \sum_{\ell=0}^{L} h_{\ell}^{-2} \sum_{z \in \mathcal{N}_\ell} | v_{\ell}(z) |^2} \leq c,
\]

which implies the estimate for the condition number. \( \square \)

The injection operators \( P_{\ell} : \mathcal{S}_0^1(\mathcal{T}) \to \mathcal{S}_0^1(\mathcal{T}_h) \) are in terms of the nodal bases given by the mapping

\[
\left( v_{\ell}(z) \right)_{z \in \mathcal{N}_\ell} \mapsto \left( v_{\ell}(z') \right)_{z' \in \mathcal{N}_\ell}.
\]

The representing matrices for the transfer between successive triangulations are provided by the MATLAB routine \texttt{red_refine.m}. Hence the operators \( P_{\ell} \) can be obtained by appropriate matrix products. Figure 5.14 displays a MATLAB implementation of the preconditioned CG algorithm with BPX preconditioner. In Tables 5.1 and 5.2 we displayed corresponding iteration numbers and compared them to those of the CG algorithm without preconditioning. We see that the BPX preconditioning leads to a termination within finitely many iterations for a sequence of uniformly refined triangulations and a fixed stopping criterion. Although the iteration numbers in the unpreconditioned case grow rapidly, the total CPU times of the two solution methods are comparable for all meshes. To benefit fully from the good properties of the BPX preconditioner, a more efficient implementation than the one shown in Fig. 5.14 is required. An alternative is using hierarchical preconditioners which avoid certain redundancies.
function bpx_precond_cg(d_tmp,L_tmp)
global h d L P_full; d = d_tmp; L = L_tmp;
[c4n,n4e,Db,Nb] = triang_cube(d); Db = [Db,Nb]; Nb = [];
nc = size(c4n,1); fNodes_prev = setdiff(1:nc,unique(Db));
h = zeros(L,1);
for ell = 1:L
    [c4n,n4e,Db,Nb,~,P1] = red_refine(c4n,n4e,Db,Nb);
    nc = size(c4n,1); fNodes = setdiff(1:nc,unique(Db));
    P{ell} = P1(fNodes,fNodes_prev);
    fNodes_prev = fNodes; h(ell) = 2^(-ell);
end
nfNodes = size(fNodes,2);
P_full{L} = spye(nfNodes);
for ell = L-1:-1:1
    P_full{ell} = P_full{ell+1}*P{ell+1};
end
[s,m] = fe_matrices(c4n,n4e);
A = s(fNodes,fNodes);
b = m(fNodes,:)\*f(c4n);
u = zeros(nc,1);
u(fNodes) = cg_precond(u(fNodes),A,b);
show_p1(c4n,n4e,Db,Nb,u)

function x = cg_precond(x,A,b)
r = b-A*x; z = apply_bpx(r); d = z;
rz_old = r'*z; eps = 1e-4;
while sqrt(r'*r) > eps
    alpha = rz_old/(d'*A*d);
    x = x+alpha*d;
    r = r-alpha*A*d;
    z = apply_bpx(r);
    rz_new = r'*z;
    beta = rz_new/rz_old;
    d = z+beta*d;
    rz_old = rz_new;
end

function Cr = apply_bpx(r)
global h d L P_full;
Cr = zeros(size(r));
for ell = 1:L
    Cr = Cr+h(ell)^(2-d)*P_full{ell}*(P_full{ell}'\*r);
end

function val = f(x); val = ones(size(x,1),1);

Fig. 5.14 MATLAB implementation of the preconditioned conjugate gradient method with BPX preconditioner.
Table 5.1 Iteration numbers and total CPU times in seconds for the conjugate gradient algorithm (CG) and its preconditioned version with the BPX preconditioner (BPX) in a two-dimensional Poisson problem

<table>
<thead>
<tr>
<th>$d = 2$, $#N_h$</th>
<th>9</th>
<th>49</th>
<th>225</th>
<th>961</th>
<th>3969</th>
<th>16129</th>
<th>65025</th>
<th>261121</th>
</tr>
</thead>
<tbody>
<tr>
<td>CG, $N_{iter}$</td>
<td>3</td>
<td>8</td>
<td>17</td>
<td>34</td>
<td>67</td>
<td>129</td>
<td>251</td>
<td>480</td>
</tr>
<tr>
<td>BPX, $N_{iter}$</td>
<td>4</td>
<td>9</td>
<td>11</td>
<td>12</td>
<td>12</td>
<td>13</td>
<td>13</td>
<td>13</td>
</tr>
<tr>
<td>CG, $T_{cpu}$ [s]</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.2</td>
<td>0.8</td>
<td>3.0</td>
<td>12.6</td>
<td>55.2</td>
</tr>
<tr>
<td>BPX, $T_{cpu}$ [s]</td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.2</td>
<td>0.7</td>
<td>3.1</td>
<td>12.2</td>
<td>49.5</td>
</tr>
</tbody>
</table>

Table 5.2 Iteration numbers and total CPU times in seconds for the conjugate gradient algorithm (CG) and its preconditioned version with the BPX preconditioner (BPX) in a three-dimensional Poisson problem

<table>
<thead>
<tr>
<th>$d = 3$, $#N_h$</th>
<th>27</th>
<th>343</th>
<th>3375</th>
<th>26791</th>
<th>250047</th>
</tr>
</thead>
<tbody>
<tr>
<td>CG, $N_{iter}$</td>
<td>5</td>
<td>12</td>
<td>28</td>
<td>62</td>
<td>137</td>
</tr>
<tr>
<td>BPX, $N_{iter}$</td>
<td>5</td>
<td>10</td>
<td>14</td>
<td>18</td>
<td>23</td>
</tr>
<tr>
<td>CG, $T_{cpu}$ [s]</td>
<td>0.1</td>
<td>0.4</td>
<td>2.9</td>
<td>23.8</td>
<td>192.3</td>
</tr>
<tr>
<td>BPX, $T_{cpu}$ [s]</td>
<td>0.1</td>
<td>0.4</td>
<td>2.9</td>
<td>29.1</td>
<td>188.0</td>
</tr>
</tbody>
</table>

Fig. 5.15 Coarse triangulation $\mathcal{T}_H$, fine triangulation $\mathcal{T}_h$, and compatible overlapping partition of $\Omega$

5.3.4 Two-Level Preconditioning

Let $\mathcal{T}_h$ be a triangulation of $\Omega$ and let $(\Omega_j)_{j=1,\ldots,J}$ be an overlapping partition of $\Omega$ into Lipschitz domains $\Omega_j \subset \Omega$ whose boundaries $\partial \Omega_j$ are matched by edges in $\mathcal{T}_h$. Let $\mathcal{T}_H$ be another triangulation of $\Omega$ such that $\mathcal{T}_h$ is a uniform refinement of $\mathcal{T}_H$, cf. Fig. 5.15.

For $j = 1,\ldots,J$, let $\mathcal{T}_h^j$ be the induced triangulation $\mathcal{T}_h|_{\Omega_j}$ and let $\mathcal{S}_0^1(\mathcal{T}_h^j)$ be the space of finite element functions in $\mathcal{S}_0^1(\mathcal{T}_h^j)$ that are extended by zero to $\Omega$. We let

$$P_j : \mathcal{S}_0^1(\mathcal{T}_h^j) \rightarrow \mathcal{S}_0^1(\mathcal{T}_h),$$

$j = 1,\ldots,J$, and let

$$P_H : \mathcal{S}_0^1(\mathcal{T}_H) \rightarrow \mathcal{S}_0^1(\mathcal{T}_h)$$
denote the embedding of functions into $\mathcal{X}_0^1(\mathcal{T}_h)$, i.e., the identity operator. Letting $A_h$, $A_H$, and $A_j$, $j = 1, \ldots, J$, denote the finite element stiffness matrices related to the spaces $\mathcal{X}_0^1(\mathcal{T}_h)$, $\mathcal{X}_0^1(\mathcal{T}_H)$, and $\mathcal{X}_0^1(\mathcal{T}_j^h)$, respectively, we define the two-level additive Schwarz preconditioner $C_2\ell \in \mathbb{R}^{n \times n}$ by

$$ C_2\ell = P_H A_H^{-1} P_H^T + \sum_{j=1}^J P_j A_j^{-1} P_j^T. $$

For this preconditioner one can show $\text{cond}_2(C_2\ell A_h) \leq c_2\ell (1 + H/\delta)$, where $\delta$ is the minimal overlap diameter. In the situation depicted in Fig. 5.15 we have $\delta \sim h$.

**References**

Fundamental contributions to the development of multigrid methods, preconditioning of finite element matrices, and domain decomposition methods are the articles [2–4, 6, 9, 12, 18]. Specialized textbooks on the subjects are the references [7, 10, 11, 13, 15, 16]. The historical development of domain decomposition methods is recapitulated in [8], and the survey article [17] discusses various aspects of preconditioning techniques. Chapters on iterative solution methods are contained in the textbooks [1, 5, 14].


