Ernst P. Stephan

Multigrid and Domain Decomposition Methods

Institut für Angewandte Mathematik
Leibniz Universität Hannover

Compact Course for Trujillo (Peru) and Cochabamba (Bolivia)
October 10, 2019

Lecture Notes

These lecture notes of my compact courses in Cochabamba and Trujillo deal with a) Multigrid Methods (MG) (Chapter 3) and b) Domain Decomposition (Schwarz) Methods (Chapter 4). The notes are completed by sections on linear iteration methods (Chapter 2) and the Conjugate Gradient Method (Chapter 5). A short introduction into MG is given in Chapter 1 and references to further literature for FEM and BEM are given at various places of the manuscript.

Contents

1

Introduction

Multigrid-methods are used to accelerate the solution of large systems of linear equations. Let us consider a simple example. We want to solve approximatively the following 1-dimensional differential equation:

$$\boxed{\begin{aligned} -u''(t) &= f(t), \quad \forall\, t \in (0,1) \\ u(0) &= u(1) = 0. \end{aligned}} \tag{1.1}$$

We take a uniform grid on $[0,1]$ with meshsize h and get $n+2$ points $x_0, \ldots, x_{n+1}$ with $x_i := \frac{i}{n+1}$.

We substitute u'' in (1.1) by finite differences and get the following system of n unknowns $u_1, \ldots, u_n$ and n equations

$$\begin{aligned} f_1 := f(x_1) &\overset{!}{=} \tfrac{-u_2+2u_1-u_0}{h^2} \\ f_2 := f(x_2) &\overset{!}{=} \tfrac{-u_3+2u_2-u_1}{h^2} \\ \vdots \quad &\vdots \\ f_n := f(x_n) &\overset{!}{=} \tfrac{-u_{n+1}+2u_n-u_{n-1}}{h^2} \end{aligned}$$

where u_i denotes the approximands of $u(x_i)$. Note that from the boundary conditions we have

$$u_0 = u(0) = u_{n+1} = u(1) = 0.$$

We can describe this problem by a matrix-vector formulation:

$$\boxed{\begin{aligned} &A^h\mathbf{u} = \mathbf{f} \text{ with} \\ &A^h = \tfrac{1}{h^2}\begin{pmatrix} 2 & -1 & & 0 \\ -1 & 2 & \ddots & \\ & \ddots & \ddots & -1 \\ 0 & & -1 & 2 \end{pmatrix} \in \mathbb{R}^{n\times n} \text{ and } \mathbf{f} := (f_1, \ldots, f_n)^T \in \mathbb{R}^n \end{aligned}} \tag{1.2}$$

A^h is a symmetric and positive definite matrix. We use the **Jacobi method** to solve (1.2), i.e.

$$\mathbf{u}^{k+1} = (I - D^{-1}A^h)\mathbf{u}^k + D^{-1}\mathbf{f}, \quad D := \operatorname{diag}(A^h).$$

The eigenvalues of A^h are

$$\lambda_h^j = \frac{4}{h^2} \cdot \sin^2\left(j\pi\frac{h}{2}\right), \qquad j = 1, \ldots, n,$$

with eigenvectors

$$\varphi_h^j = \left(\sin(jl\pi h) \right)_{l=1}^{n}, \qquad j = 1, \ldots, n.$$

Set $R := I - \frac{h^2}{2} A^h$. The eigenvectors of R and A^h are identical and the eigenvalues of R are

$$\lambda_j(\omega) = 1 - 2\sin^2\left(\pi j \frac{h}{2}\right) \overset{h \ll 1}{\approx} 1 - c \cdot h^2, \qquad j = 1, \ldots, n.$$

Now we calculate the error $\mathbf{e}^k$ between the iterative solution $\mathbf{u}^k$ and the exact solution $\mathbf{u}$ of (1.2).

$$\begin{aligned} \mathbf{e}^{k+1} &:= \mathbf{u}^{k+1} - \mathbf{u} = R\mathbf{u}^k - R\mathbf{u} = R\mathbf{e}^k \\ \Rightarrow \|\mathbf{e}^{k+1}\| &\le \|R\| \cdot \|\mathbf{e}^k\| \approx \underbrace{(1 - c \cdot h^2)}_{\to 1 \text{ as } h \to 0} \cdot \|\mathbf{e}^k\| \end{aligned}$$

Note that the smaller h, the smaller is the reduction of the error. This disadvantage motivates the introduction of a simple multigrid algorithm:

2-Level Algorithm

1. Take an iterative solution $\mathbf{x}^h$ of (1.2) (maybe with Jacobi) and calculate the **residual**:
$$\mathbf{r}^h := \mathbf{f} - A^h \mathbf{x}^h.$$
2. Let H^h be the set of all grid functions on a grid with meshsize h and let
$$I_h^{2h} : \begin{cases} H^h \to H^{2h} \\ u^h \to u^{2h} \end{cases}$$
be a projection on a coarse grid (for example $u_i^{2h} = \frac{1}{4}(u_{2i-1}^h + 2u_{2i}^h + u_{2i+1}^h)$). Set
$$\mathbf{r}^{2h} := I_h^{2h} \mathbf{r}^h.$$
3. *Coarse grid correction:* Solve exactly:
$$A^{2h} \mathbf{e}^{2h} = \mathbf{r}^{2h}.$$
4. Let
$$I_{2h}^h : \begin{cases} H^{2h} \to H^h \\ u^{2h} \to u^h \end{cases}$$
be an extension of the coarse grid onto the fine grid (for example $u_{2i}^h = u_i^{2h}$, $u_{2i+1}^h = \frac{1}{2}(u_i^{2h} + u_{i+1}^{2h})$). Set
$$\mathbf{y}^h := \mathbf{x}^h + I_{2h}^h \mathbf{e}^{2h}.$$
5. Go back to step 1 with $\mathbf{y}^h$ as a new starting vector for the calculation of the iterative solution of (1.2).

We will see, that this simple multigrid algorithm converges much faster than the traditional iterative methods.

2

Linear Iteration Method

The general form of a linear iterative process to solve $Au = f$ is

$$\boxed{u^{k+1} = u^k - B(Au^k - f),} \tag{2.1}$$

where $A, B \in \mathbb{R}^{n\times n}$ are symmetric and positive definite . Some examples for B are:

- Richardson method
$$B = \lambda_{\max}^{-1} I,$$
where $\lambda_{\max}$ is the maximum eigenvalue of A.
- Jacobi method
$$B = D^{-1},$$
where $D = \mathrm{diag}(A)$.
- Gauß-Seidel method
$$B = (L + D)^{-1},$$
where $A = L + D + R$, and L denotes the lower triangular part, R the upper triangular part and D the diagonal of A.

Originally, we want to solve

$$Au = f, \tag{2.2}$$

but it might be expensive to calculate A^{-1}. Hence we try to find a matrix B which is cheaper to calculate and calculate

$$BAu = Bf \tag{2.3}$$

instead of (2.2). Then we get $u \sim Bf$. Then, for the error $e^{k+1} = u^{k+1} - u$ we have

$$\begin{aligned} e^{k+1} &= u^k - B(Au^k - f) - u \\ &= u^k - u - B(Au^k - Au) \\ &= e^k - BAe^k \\ &= (I - BA)e^k, \end{aligned}$$

i.e.

$$\|e^{k+1}\| \le |||I - BA|||\, \|e^k\|.$$

Hence, if $|||I - BA||| < 1$ then we get convergence, i.e. $\|e^k\| \to 0$ as $k \to \infty$. Now, we have to introduce the condition number.

Definition 2.1. *The condition κ of a matrix A is given by*

$$\kappa(A) = \frac{\lambda_{\max}}{\lambda_{\min}},$$

where $\lambda_{\max}(\lambda_{\min})$ denotes the maximum (minimum) eigenvalue of A.

Note that if A and B are symmetric and positive definite then BA is symmetric and positive definite and the eigenvalues of BA are positive. Now we can prove the following theorem.

Theorem 2.2. *1. There exists a constant $K \geq \kappa(BA)$ such that*

$$|||I - BA||| \leq \frac{K-1}{1+K} \ . \tag{2.4}$$

2. If the eigenvalues of BA satisfy $\lambda_{\min} + \lambda_{\max} = 2$, then we have $K = \kappa(BA)$.
3. If $\sigma(BA) \subset (0,2)$, where $\sigma(BA)$ denotes the set of all eigenvalues (spectrum) of BA, then the linear iterative method (2.1) is convergent.

Proof. 1. For every $u \in \mathbb{R}^n$ we have

$$\lambda_{\min} \langle u, u\rangle \leq \langle BAu, u\rangle \leq \lambda_{\max} \langle u, u\rangle \, .$$

Let ϕ_i denote the eigenvectors and λ_i the eigenvalues of BA. Then we can write

$$u = \sum_{i=1}^{n} u_i \phi_i .$$

Hence

$$\big\langle ABAu, u\big\rangle = \sum_{i=1}^{n} u_i \big\langle ABA\phi_i, u\big\rangle = \sum_{i=1}^{n} \lambda_i u_i \big\langle A\phi_i, u\big\rangle \begin{cases} \leq \lambda_{\max} \langle Au, u\rangle \\ \geq \lambda_{\min} \langle Au, u\rangle \end{cases},$$

i.e.

$$\lambda_{\min} \langle Au, u\rangle \leq \langle ABAu, u\rangle \leq \lambda_{\max} \langle Au, u\rangle \, .$$

This implies

$$-(\lambda_{\max} - 1) \langle Au, u\rangle \leq \big\langle A(I - BA)u, u\big\rangle \leq (1 - \lambda_{\min}) \langle Au, u\rangle \, . \tag{2.5}$$

We set

$$a := \max\big\{(\lambda_{\max} - 1), (1 - \lambda_{\min})\big\}, \quad \text{and} \quad K := \frac{1+a}{1-a}.$$

Moreover

$$|||I - BA||| := \sup_{u \in \mathbb{R}^n} \left| \frac{\big\langle A(I - BA)u, u\big\rangle}{\big\langle Au, u\big\rangle} \right|,$$

and

$$[u, u] := \big\langle Au, u\big\rangle \quad \forall u \in \mathbb{R}^n,\ A \text{ symmetric and positive definite}\ .$$

Then

$$|||I - BA||| = \sup_{u \in \mathbb{R}^n} \left| \frac{\big[(I - BA)u, u\big]}{[u, u]} \right|.$$

By the definition of a we get from (2.5)

$$-a \langle Au, u\rangle \leq \big\langle A(I - BA)u, u\big\rangle \leq a \langle Au, u\rangle$$

and this implies

$$|||I - BA||| \leq a = \frac{K-1}{K+1} \quad \text{(by definition of } K).$$

which gives 1).

2. If $\lambda_{\max} + \lambda_{\min} = 2$ we have
$$a = \lambda_{\max} - 1 = 1 - \lambda_{\min}, \quad \Longrightarrow \quad K = \frac{1 + \lambda_{\max} - 1}{1 - (1 - \lambda_{\min})} = \kappa(BA).$$
3. BA is symmetric and positive definite with respect to the inner product $\big[\cdot,\cdot\big] = \big\langle A\cdot,\cdot\big\rangle$. Let ϕ_i denote the eigenvectors of the matrix BA such that
$$BA\phi_i = \lambda_i \phi_i, \quad i = 1, \ldots, n$$
and
$$\big[\phi_i, \phi_j\big] = \delta_{ij}, \quad i, j = 1, \ldots, n.$$
Hence we obtain
$$e^k = (I - BA)e^{k-1} = \ldots = (I - BA)^k e^0,$$
and
$$e^0 = \sum_{i=1}^{n} \big[e^0, \phi_i\big]\phi_i.$$
Therefore, since now $e^k = (I - BA)^k \sum_{i=1}^{n} \big[e^0, \phi_i\big]\phi_i$, we have
$$e^k = \sum_{i=1}^{n} \big[e^0, \phi_i\big](1 - \lambda_i)^k \phi_i$$
so that
$$\begin{aligned} e^k \to 0 &\Leftrightarrow |1 - \lambda_i| < 1 \forall i = 1, \ldots, n \\ &\Leftrightarrow \sigma(BA) \subset (0, 2), \end{aligned}$$
which gives 3. □

Example 2.3. The following Hilbert matrix
$$A = \begin{pmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \frac{1}{4} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \frac{1}{6} \\ \frac{1}{4} & \frac{1}{5} & \frac{1}{6} & \frac{1}{7} \end{pmatrix} \tag{2.6}$$
is a classic example of an ill-conditioned matrix:
$$\lambda_{min}(A) = 0.005, \quad \lambda_{max}(A) = 1.5002 \quad \text{and } \kappa(A) = 1.5514 \times 10^4\,.$$

Appying the Jacobi method with $B = D^{-1}$, we have that $\lambda_{min}(BA) = 0.005$ and $\lambda_{max}(BA) = 3.5821$. Due to $\sigma(BA) \not\subseteq (0, 2)$ we can not use the Theorem 2.2 and the convergence of the method can not be guaranteed. However for the Richardson method with $B = \lambda_{max}^{-1} I$, we have $\lambda_{min}(BA) = 0.00006$ and $\lambda_{max}(BA) = 1.0$, and by the Theorem 2.2 we can ensure the convergence of the method since $\sigma(BA) \subset (0, 2)$. Note that in both cases the matrix BA is ill-conditioned, $\kappa(BA) = 9.35 \times 10^3$ and $\kappa(BA) = 16.6 \times 10^3$ respectively.

3

Multigrid Methods (2-Level Methods)

In this section we follow the lecture notes by J. Bramble [2].

Let $\mathcal{M}$ be a N-dimensional vectorspace ($N < \infty$) with inner product $(\cdot, \cdot)$. Let $A : \mathcal{M} \to \mathcal{M}$ be symmetric, positive definite and linear. We define the inner product

$$a(u, v) := (Au, v) \quad \forall u, v \in \mathcal{M}.$$

The general form of a linear iterative process to solve $Au = f$ ($f \in \mathcal{M}$) is

$$\boxed{u^{n+1} = u^n - B(Au^n - f), \quad (u^0 \text{ arbitrary})} \tag{3.1}$$

where $B : \mathcal{M} \to \mathcal{M}$ is a symmetric and positive definite operator, which depends on the method. For example $B := \lambda^{-1}_{\max(A)} I$ (Richardson). For the error $e^{n+1} := u - u^{n+1}$ we get

$$e^{n+1} = (I - BA)e^n \quad \Longrightarrow \|e^{n+1}\| \leq |||I - BA||| \cdot \|e^n\| \ .$$

The behaviour of (3.1) depends essentially on the spectrum of $I - BA$. For convergence of (3.1) it is necessary to find a norm $|||\cdot|||$ with

$$|||I - BA||| \leq \delta < 1 \ .$$

There are many ways of constructing B in (3.1). One possiblility is the multigrid method. Note that for application of such a method an explicit expression for B is not necessary, we only have to know the action of B on every $v \in \mathcal{M}$. Using the Richardson method, we construct now a special case of the multigrid method.

We recall the traditional **Richardson method**:

> Assume that the spectrum of A is well known ($0 < \lambda_1 \leq \ldots \leq \lambda_N$ with corresponding eigenvectors φ_i). Let $\lambda_{\max} = \lambda_N$ be the largest eigenvalue of A. Choose a starting vector x^0 and compute x^{n+1} as follows:
>
> $$\boxed{x^{n+1} = x^n - \lambda^{-1}_{\max}(Ax^n - f).} \tag{3.2}$$

The error of the Richardson method is

$$e^{n+1} = x - x^{n+1} = (I - \lambda^{-1}_{\max} A)(x - x^n) = (I - \lambda^{-1}_{\max} A)e^n \ .$$

There exist $c_1, \ldots, c_N \in \mathbb{R}$ satisfying

$$x - x^0 = \sum_{i=1}^{N} c_i \varphi_i \quad \Longrightarrow \quad x - x^n = \sum_{i=1}^{N} \underbrace{(1 - \lambda^{-1}_{\max}\lambda_i)^n}_{\text{frequency terms}} \ c_i \varphi_i \ .$$

Thus the high frequency terms are reduced even with one Richardson step, i.e. the coefficient of φ_N in $(x - x^1)$ is zero. However, if $\lambda_i \ll \lambda_{\max}$ we get $1 - \lambda_i \lambda_{\max}^{-1} \approx 1$ and the reduction of the lower frequency terms is unfortunately very small.
To improve the reduction of the lower frequency terms consider the linear subspace $\widetilde{\mathcal{M}} \subseteq \mathcal{M}$ spanned by the eigenvectors belonging to the lower frequency terms and project the residual $A(x - x^{n+1})$ onto the subspace $\widetilde{M}$, i.e. solve the equation

$$\tilde{A}q = \tilde{P}_0 A(x - x^{n+1}), \qquad q \in \widetilde{\mathcal{M}},$$

where $\tilde{P}_0$ and $\tilde{A}$ are defined by

$$\begin{aligned} \tilde{P}_0 : \mathcal{M} \to \widetilde{\mathcal{M}}, \qquad & a(\tilde{P}_0 u, v) = \underbrace{a(u,v)}_{=(Au,v)} \; \forall u \in \mathcal{M} \quad \forall v \in \widetilde{\mathcal{M}} \\ \tilde{A} : \widetilde{\mathcal{M}} \to \widetilde{\mathcal{M}}, \qquad & (\tilde{A}u, v) = \underbrace{A(u,v)}_{=(Au,v)} \quad \forall u, v \in \widetilde{\mathcal{M}} \; . \end{aligned}$$

We correct x^{n+1} by

$$y^{n+1} := x^{n+1} + q \; . \tag{3.3}$$

This new aproximation y^{n+1} is not perturbed by the error of the lower frequency terms. Adding this correction to Richardson's method we get a linear iterative process:

$$\begin{aligned} e^{n+1} &= x - y^{n+1} = x - x^{n+1} - \tilde{A}^{-1}\tilde{P}_0 A(x - x^{n+1}) \\ &= (I - \tilde{A}^{-1}\tilde{P}_0 A)(I - \lambda_{\max}^{-1} A)(x - x^n) =: (I - BA)e^n \\ y^{n+1} &= x^n - B(Ax^n - f) \end{aligned} \tag{3.4}$$

Since a symmetric B would simplify the discussion of this method, we apply another **smoothing step**

$$z^{n+1} = y^{n+1} - \lambda_{\max}^{-1}(Ay^{n+1} - f) \; . \tag{3.5}$$

Thus the whole process writes as:

$$\boxed{\begin{aligned} x^{n+1} &= x^n - \lambda_{\max}^{-1}(Ax^n - f) \\ &\text{Find } q \in \widetilde{\mathcal{M}} \text{ satisfying} \\ \tilde{A}q &= \tilde{P}_0 A(x - x^{n+1}) \\ y^{n+1} &:= x^{n+1} + q \\ z^{n+1} &= y^{n+1} - \lambda_{\max}^{-1}(Ay^{n+1} - f) \end{aligned}} \tag{3.6}$$

Setting $\tilde{P} := \tilde{A}^{-1}\tilde{P}_0 A,\; K := I - \lambda_{\max}^{-1} A$ we get:

$$\begin{aligned} e^{n+1} &= x - z^{n+1} \overset{(3.5)}{=} x - y^{n+1} + \lambda_{\max}^{-1}(Ay^{n+1} - f) \\ &= K(x - y^{n+1}) \overset{(3.4)}{=} K(I - \tilde{P})K(x - x^n) =: (I - BA)e^n \\ z^{n+1} &= x^n - B(Ax^n - f) \end{aligned} \tag{3.7}$$

It can be shown $I - BA$ is symmetric with respect to A, because K is symmetric with respect to A.
We may think of (3.7) as a preconditioned iterative process with preconditioner B. To get the action of B on $v \in \mathcal{M}$ it is not necessary to know B itself. If we apply (3.6) with $x^n = 0$ and $f = v$ we get the action of B on v. (3.6) is called the **2-level multigrid cycle**. B is the corresponding operator for the symmetric 2-level method.

Theorem 3.1. *[2] Assume, there exists a* $\delta \in [0,1)$ *satisfying*

$$a((I - BA)u, u) \le \delta a(u,u) \qquad \forall u \in \mathcal{M}. \tag{3.8}$$

Then the symmetric 2-level method is convergent and we may estimate the error by

$$|||e^{n+1}|||^2 := a(e^{n+1}, e^{n+1}) \le \delta^2 |||e^n|||^2.$$

Proof. Using that $I - BA$ is symmetric and positive definite with respect to A, i.e.

$$\begin{aligned} a((I - BA)u, u) &= a(K(I - \tilde{P})Ku, u) = a((I - \tilde{P})Ku, Ku) \\ &= a((I - \tilde{P})Ku, (I - \tilde{P} + \tilde{P})Ku) = a((I - \tilde{P})Ku, (I - \tilde{P})Ku) \\ &\ge 0, \end{aligned}$$

we get

$$\begin{aligned} a(e^{n+1}, e^{n+1}) &= a((I - BA)e^n, (I - BA)e^n) \\ &= a((I - BA)(I - BA)^{\frac{1}{2}}e^n, (I - BA)^{\frac{1}{2}}e^n) \\ &\le \delta a((I - BA)^{\frac{1}{2}}e^n, (I - BA)^{\frac{1}{2}}e^n) \\ &= \delta a((I - BA)e^n, e^n) \le \delta^2 a(e^n, e^n). \end{aligned}$$

□

We can write (3.8) as

$$(1 - \delta) a(u,u) \le a(BAu, u) \le a(u,u).$$

This means, the symmetric 2-level operator B is a preconditioner for A with condition number

$$\kappa(BA) := \frac{\lambda_{\max}}{\lambda_{\min}} \le \frac{1}{1 - \delta},$$

where $\lambda_{\max}$ is the largest and $\lambda_{\min}$ is the smallest eigenvalue of BA with respect to the inner product $a(\cdot,\cdot)$.

3.1 Example

Let $\Omega \subseteq \mathbb{R}^n$ be a domain and let $f \in \mathbf{L}^2(\Omega)$ be given. Find u satisfying

$$\boxed{\begin{array}{l} -\triangle u = f \quad \text{in } \Omega \\ u = 0 \quad \text{on } \partial\Omega \end{array}} \tag{3.9}$$

We consider $\mathcal{M}_1 \subseteq \mathcal{M}_2 \subseteq \ldots \subseteq \mathcal{M}_J = \mathcal{M} \subseteq H_0^1(\Omega)$. $\mathcal{M}_k$ consists of piecewise linear C^0 functions on a regular k-triangulisation of Ω with meshsize h_k. Further these functions must be zero on $\partial\Omega$. Galerkin's method yields:

$$\text{Find } u_{h_k} \in \mathcal{M}_k \text{ with: } \int_\Omega \nabla u_{h_k} \nabla \chi = \int_\Omega f \cdot \chi \qquad \forall \chi \in \mathcal{M}_k, \tag{3.10}$$

where $\int_\Omega \nabla u_{h_k} \nabla \chi =: a(u_{h_k}, \chi)$.

Let the norm $|||\cdot|||$ be defined as

$$|||u|||^2 := \int_\Omega \nabla u \nabla v \approx \|u\|^2_{H_0^1(\Omega)} \qquad \forall u \in \mathcal{M}_k.$$

Theorem 3.2. *[2] There exists a* $\tilde{c} > 0$ *which is independent of* h_k *such that*

$$|||e^{n+1}||| \le \left(1 - \frac{1}{\tilde{c}}\right)^2 |||e^n||| \qquad \forall n \in \mathbb{N}.$$

Proof. By (3.7) we get

$$\begin{aligned} |||e^{n+1}|||^2 &= |||K(I-\tilde{P})Ke^n|||^2 = |||K(I-\tilde{P})^2Ke^n|||^2 \\ &\leq |||K(I-\tilde{P})|||^2\, |||(I-\tilde{P})Ke^n|||^2 \\ &= |||K(I-\tilde{P})|||^2\, \underbrace{a((I-\tilde{P})Ke^n,(I-\tilde{P})Ke^n)}_{=a(K(I-\tilde{P})^2Ke^n,e^n)=a(e^{n+1},e^n)} \\ &\overset{C.S.}{\leq} |||K(I-\tilde{P})|||^2\, |||e^{n+1}|||\, |||e^n||| \\ \Longrightarrow |||e^{n+1}||| &\leq |||K(I-\tilde{P})|||^2\, |||e^n|||. \end{aligned}$$

Hence we have to study $K(I-\tilde{P})$. $\tilde{P}$ satisfies for all $\chi \in \mathcal{M}, \theta \in \mathcal{M}_k$

$$a((I-\tilde{P})\chi,\theta) = a(\chi,\theta) - \underbrace{a(\tilde{P}\chi,\theta)}_{a(\chi,\theta)} = 0 \tag{3.11}$$

We recall the approximation property of FEM,

$$\exists c>0 \quad \forall \chi\in\mathcal{M} \quad \exists \eta\in\mathcal{M}_k: \quad \|\chi-\eta\|_{L^2(\Omega)} \leq ch_k\|\chi\|_{H^1(\Omega)}, \tag{3.12}$$

and the inverse property

$$\exists \bar{c}>0 \quad \forall \chi\in\mathcal{M}_k: \quad a(\chi,\chi)^{\frac{1}{2}} \approx \|\chi\|_{H^1(\Omega)} \leq \bar{c}h_k^{-1}\|\chi\|_{L^2(\Omega)}\ . \tag{3.13}$$

For the largest eigenvalue λ_k of A in $\mathcal{M}_k$ we have

$$\lambda_k = \sup_{u\in\mathcal{M}_k} \frac{a(u,u)}{(u,u)} \overset{(3.13)}{\leq} \bar{c}^2 h_k^{-2}. \tag{3.14}$$

For a $\chi \in \mathcal{M}$ we set $\tilde{\chi} := (I-\tilde{P})\chi$ and take a $\tilde{\eta} \in \mathcal{M}_k$ satisfying (3.12) for $\tilde{\chi}$. We get

$$\begin{aligned} a(\tilde{\chi},\tilde{\chi}) &\overset{(3.11)}{=} a(\tilde{\chi},\tilde{\chi}-\tilde{\eta}) = (A\tilde{\chi},\tilde{\chi}-\tilde{\eta}) \\ &\overset{C.S.}{\leq} \|A\tilde{\chi}\|_{L^2(\Omega)}\|\tilde{\chi}-\tilde{\eta}\|_{L^2(\Omega)} \\ &\overset{(3.12)}{\leq} \|A\tilde{\chi}\|_{L^2(\Omega)} \cdot ch_k\|\tilde{\chi}\|_{H^1(\Omega)} \\ &\leq \hat{c}h_k(A\tilde{\chi},A\tilde{\chi})^{\frac{1}{2}}\, a(\tilde{\chi},\tilde{\chi})^{\frac{1}{2}} \end{aligned}$$

and therefore

$$a(\tilde{\chi},\tilde{\chi}) \leq \hat{c}^2 h_k^2 (A\tilde{\chi},A\tilde{\chi}) \overset{(3.14)}{\leq} \underbrace{\hat{c}^2\bar{c}^2}_{:=\tilde{c}} \lambda_k^{-1}(A\tilde{\chi},A\tilde{\chi}). \tag{3.15}$$

Now we can consider $K\tilde{\chi}$.

$$\begin{aligned} a(K\tilde{\chi},(I-\lambda_k^{-1}A)\tilde{\chi}) &= a(K\tilde{\chi},\tilde{\chi}) - \lambda_k^{-1}a((I-\lambda_k^{-1}A)\tilde{\chi},A\tilde{\chi}) \\ &= a(K\tilde{\chi},\tilde{\chi}) - \lambda_k^{-1}a(\tilde{\chi},A\tilde{\chi}) + \lambda_k^{-2}a(A\tilde{\chi},A\tilde{\chi}) \\ &\overset{(3.14)}{\leq} a(K\tilde{\chi},\tilde{\chi}) - \lambda_k^{-1}(A\tilde{\chi},A\tilde{\chi}) + \lambda_k^{-1}(A\tilde{\chi},A\tilde{\chi}) \\ &= a(K\tilde{\chi},\tilde{\chi}) = a(\tilde{\chi},\tilde{\chi}) - \lambda_k^{-1}a(A\tilde{\chi},\tilde{\chi}) \\ &\overset{(3.15)}{\leq} \left(1-\tfrac{1}{\tilde{c}}\right) a(\tilde{\chi},\tilde{\chi}) \leq \left(1-\tfrac{1}{\tilde{c}}\right) a(\chi,\chi) \end{aligned}$$

And this means that $|||K(I-\tilde{P})|||^2 \leq \left(1-\frac{1}{\tilde{c}}\right)^2$ □

3.2 Implementation

First, we take a starting vector $u^{(0)}$ and define $u^{(l)}$ by

$$u^{(l)} = \underbrace{u^{(l-1)} - B_J(Au^{(l-1)} - f)}_{=Mg_J(u^{(l-1)},f)},$$

where $\mathcal{M} = \mathcal{M}_j$. Note that $B_j g = Mg_j(0, g)$, i.e. Mg_j may be used as a preconditioner for the cg-method, too. Now, if $j = 1$ we want to get

$$Mg_1(w, f_1) = A_1^{-1} f,$$

and for $k > 1$

$$Mg_k(w, f_k) := y^{2m(k)},$$

where $y^{2m(k)}$ is constructed in the following way

	$x^0 := w \in \mathcal{M}_k$
Presmoothing:	$x^i := x^{i-1} + R_k^{(i+m(k))}\left(f_k - A_k x^{i-1}\right), \quad i = 1, \ldots, m(k)$
Restriction:	$r_{k-1} := Q_{k-1}\left(f_k - A_k x^{m(k)}\right)$
Correction:	Solve $\quad A_{k-1} q = r_{k-1} \quad$ approximately by
	p multigrid steps :
	$q^0 := 0, \quad q^i := Mg_{k-1}(q^{i-1}, r_{k-1}), \quad i = 1, \ldots, p$
Prolongation:	$y^{m(k)} := x^{m(k)} + I_k q^p$
Postsmoothing:	$y^i := y^{i-1} + R_k^{(i+m(k))}\left(f_k - A_k y^{i-1}\right),$
	$i = m(k) + 1, \ldots, 2m(k).$

Note that $R_k^{(l)}$ is defined by

$$R_k^{(l)} := \begin{cases} R_k & l \text{ is even} \\ {R_k}^T & l \text{ is odd} \end{cases}.$$

The translation of this algorithm into Fortran is as follows:

```
      subroutine MG(z,r,mlev,mu,mu1.mu2,mds,xm,xn,gal)

      integer mdim
      parameter (mdim = 65536)
      integer i, l, i1, mu, mu1, mu2, mds, m
      real r(0:maxdim*2-1), z(0:masdim*2-1), gal(0:*)
      real h(0:mdim-1)
      integer mlev, mx(0:mlev), xn(0:mlev), mum(0:16)

      m=0
      mum(m) = 1
    1 continue
      if (m .eq. mlev) then
        call subsolve(gal, z(xm(m)), r(xm(m)), xn(m))
      else
         !! Presmoothing
        do  l =1,mu1
          call presmoother(gal, z(xm(m)), r(xm(m)), xn(m), h, mds)
        end do
         !! Residual
        do  i+0,xn(m)-1
```

```
          h(i) = r(xm(m) + i)
        end do
        call sres11(gal, z(xm(m)), h, xn(m))
       !! Projection
        i1 = xm(m+1)
        do  l=0,xn(m+1)-1
          r(i1) = 0.5 * h(2*l) + h(2*l + 1) + 0.5 * h(2*l + 2)
          i1 = i1+1
        end do
        m = m+1
        mum(m) = mu
        do  i = xm(m),xm(m)_xn(m)-1
          z(i) = 0.0
        end do
        goto 1
      endif
    2 continue
      if (m .lt. mlev) then
        !! Prolongation from m+1 to m
        z(m) = z(m) + I_m z(m+1)
        do l =1,mu2
          call postsmoother(...)
        end do
      endif
      mum(m) = max(0,mum(m)-1)
      if (mum(m) .gt. 0) goto 1
      m = m-1
      if (m .ge. 0) goto 2
      end
```

The translation of this algorithm into MatLab is as follows:

```
  function [z cnt error]=zweilevel1d(A,b,m,w,tol,cntmax,method,zexact)
  z=zeros(size(b));
  zold=z;
  cnt=0;
  error(1)=norm(z-zexact);

  nk=(length(z)+1)/2-1;
  Ak=sparse([],[],[],nk,nk,nk+2*(nk-1));

  for i=1:nk
      for j=max(i-1,1):min(i+1,nk)
          Ak(i,j)=[0.5 1 0.5]*A(2*i-1:2*i+1,2*j-1:2*j+1)*[0.5; 1; 0.5];
      end
  end

      switch (method)
        case {'jacobi','JOR'}
            B=1./diag(A);
      end

while cnt<cntmax
```

```
  cnt=cnt+1;
  % Presmoothing
  for i=1:m
    switch (method)
        case {'jacobi'}
            z=jacobi(A,B,b,z);
        case {'JOR'}
            z=dampedJacobi(A,B,b,z,w);
        case {'gaussSeidel'}
            z=gaussSeidel(A,b,z);
        case {'SOR'}
            z=SOR(A,b,z,w);
        otherwise
            disp('Unknown method.')
    end
  end

  % Restriction of residuum
  r=restiction(A,b,z);


  % Correction
  %q=subsolve(A,r);
  q=Ak\r;

  % Prolongation
  z=prolongation(z,q);

  % Postsmoothing
  for i=1:m
    switch (method)
        case {'jacobi'}
            z=jacobi(A,B,b,z);
        case {'JOR'}
            z=dampedJacobi(A,B,b,z,w);
        case {'gaussSeidel'}
            z=gaussSeidel(A,b,z);
        case {'SOR'}
            z=SOR(A,b,z,w);
        otherwise
            disp('Unknown method.')
    end
  end

 % error(cnt+1)=norm(z-zexact);
  if norm(z-zold)<tol*norm(zold)
      break;
  end
  zold=z;

end

end
```

```
function z=jacobi(A,B,b,z)
  z=z+B.*(b-A*z);
end

function z=dampedJacobi(A,B,b,z,w)
  z=z+w*B.*(b-A*z);
end

function z=gaussSeidel(A,b,z)
    n=length(z);
    for i=1:n
       z(i)=1/A(i,i)*(b(i)-A(i,1:i-1)*z(1:i-1)-A(i,i+1:n)*z(i+1:n));
    end
end

function z=SOR(A,b,z,w)
    n=length(z);
    for i=1:n
       z(i)=(1-w)*z(i)+w/A(i,i)*(b(i)-A(i,1:i-1)*z(1:i-1)-A(i,i+1:n)*z(i+1:n));
    end
end

function r=restiction(A,b,z)
  res=b-A*z;
  %r=zeros((length(b)+1)/2-1,1);

  r=0.5*res(2:2:end)+0.25*( res(1:2:end-2) + res(3:2:end) );

%   for i=1:length(r)
%      r(i)=0.25*res(2*i-1)+0.5*res(2*i)+0.25*res(2*i+1);
%   end

end

function z=prolongation(z,q)
  n=length(z);
% nq=length(q);
  Iq=zeros(n,1);
  Iq(1)=0.5*q(1); Iq(end)=0.5*q(end); Iq(2:2:end)=q;
  Iq(3:2:end-2)=0.5*(q(1:end-1)+q(2:end));

%    Iq=zeros(n,1);
%    for i=1:length(q)
%       Iq(2*i-1:2*i+1)=Iq(2*i-1:2*i+1)+[0.5;1;0.5]*q(i);
%    end

  z=z+Iq;
end
```

3.2.1 Model problem

This subsection is taking from the book by W. Hackbusch [3].

$$u''(x) = f \qquad x \in (0,1), \quad u(0) = u(1) = 0 \tag{3.16}$$

$$u''(x_i) \approx \frac{u(x_{i-1}) - 2u(x_i) - u(x_{i+1})}{h^2} \qquad i = 0, \ldots, 2M \tag{3.17}$$

where $x_i = ih$, $2Mh = 1$

Mesh function

$$u_h = (u(h), u(2h), \ldots, u(1-h))^T$$
$$f_h = \big(f(h), f(2h), \ldots, f(1-h)\big)^T$$

Finite difference approximation

$$A_h u_h = f_h \tag{3.18}$$

such that

$$A_h = \frac{1}{h^2}\begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{pmatrix}$$

Jacobi Method

Let $D_h = \operatorname{diag} A_h$ and $P_h = D_h - A_h$.

$$\begin{aligned} \Rightarrow u_h^{(j+1)} &= D_h^{-1}(P_h u_h^{(j)} + f_h) \\ &= D_h^{-1}\big((D_h - A_h)u_h^{(j)} + f_h\big) \\ &= u_h^{(j)} - D_h^{-1}(A_h u_h^{(j)} - f_h) \end{aligned}$$

Damped Jacobi method: ($\bar{\omega} \in (0,1]$)

$$u_h^{(j+1)} = u_h^{(j)} - \bar{\omega} D_h^{-1}(A_h u_h^{(j)} + f_h)$$

As $D_h^{-1} = \frac{h^2}{2} I$ and using $\omega = \frac{\bar{\omega}}{2}$ we get

$$u_h^{(j+1)} = u_h^{(j)} - \omega h^2 (A_h u_h^{(j)} - f_h) \qquad \omega \in (0, \frac{1}{2}) \tag{3.19}$$

Jacobi matrix

$$B_h = B_h(\omega) = I - \omega h^2 A_h \tag{3.20}$$

$$\overset{(3.19)}{\Rightarrow} u_h^{(j+1)} = B_h u_h^{(j)} + g_h \qquad g_h = \omega h^2 f_h \tag{3.21}$$

Let u_h^* be the exact solution of (3.18). Thus there holds

$$u_h^* = B_h u_h^* + g_h$$

The error vector is defined by $e^{(j)} = u_h^{(j)} - u_h^*$

$$e^{(j+1)} = u_h^{(j+1)} - u_h^* = B_h u_h^{(j)} + g_h - (B_h u_h^* + g_h)$$

$$= B_h(u_h^{(j)} - u_h^*) = B_h e^{(j)}$$

The eigenvalues of B_h are known from the exercise lessons as

$$\lambda_{i,h}(\omega) = 1 - 4\omega \sin^2\Big(\frac{i\pi h}{2}\Big), \quad i = 1, \ldots, 2M-1$$

and the eigen vectors are given by

$$\mathbf{w}_h^i = \sqrt{2h}\Big(\sin(i\pi h), sin(2i\pi h), \ldots, \sin\big((2M-1)i\pi h\big)\Big)^T.$$

For the spectral radius there holds

$$\rho_\omega(B_h) = \left|1 - 4\omega \sin^2 \frac{\pi h}{2}\right| = 1 - \omega\pi^2 h^2 + O(h^4).$$

Smoothing Iterations

The convergence of the method is not in all components equally good.

We can express the error with respect to some orthogonal basis using the eigenvectors $\mathbf{w}_h^i$:

$$e_h^{(0)} = u_h^{(0)} - u_h^* = \sum_{i=1}^{2M-1} \alpha_{ih}\mathbf{w}_h^i \tag{3.22}$$

$$e_h^{(j)} = (B_h)^j e_h^{(0)} = \sum_{i=1}^{2M-1} \alpha_{i,h}(B_h)^j \mathbf{w}_h^i = \sum_{i=1}^{2M-1} \alpha_{i,h}\big(\lambda_{i,h}(\omega)\big)^j \mathbf{w}_h^i \tag{3.23}$$

$$= \sum_{i=1}^{2M-1} \beta_{i,j,h}(\omega)\mathbf{w}_h^i \tag{5}$$

where

$$\beta_{i,j,h}(\omega) = \big(\lambda_{i,h}(\omega)\big)^j \alpha_{i,h} = \left(1 - 4\omega \sin^2 \frac{i\pi h}{2}\right)^j \alpha_{i,h}.$$

For low frequencies the holds $\alpha_{i,h} \approx \beta_{i,j,h}$,
whereas for hight frequencies we have $|\beta_{i,j,h}| \ll |\alpha_{i,h}|$.

The eigenvectors associated to the high frequencies are damped after a few iterations, there influence on the error vector decays rapidly. This means an overall smoothing of the error vector.

The Gauß-Seidel and the cg methods possess the same smoothing property.

Two Level Method

Take some iterative method

$$u_h^{(j+1)} = S_h u_h^{(j)} + g_h, \tag{3.24}$$

that obtains the smoothing property discussed before.

We set $S_h = I - \omega h^2 A_h$, $g_h = \omega h^2 f_h$ with $\omega = \frac{1}{4}$.

Define $\mathbf{u}_h^m$ $(m \geq 0)$ to be the mth approximation of (3.18) on the mesh G_h. Starting with u_h^m we calculate a few iterations as defined in (3.24) and obtain a vector $\bar{u}_h^m$ with

$$A_h \bar{u}_h^m = f_h + d_h \tag{3.25}$$

Using the smoothing property of (3.24), we see, that $e_h^m = \bar{u}_h^m - u_h^*$ "smoother" as $u_h^m - u_h^*$.

Substract (3.18) from (3.25) yields

$$A_h e_h^m = d_h. \tag{3.26}$$

Introduce a rougher mesh ($y_i = 2ih$, $i = 0, \dots, M$).
Goal: Solve (on the rougher mesh)

$$A_{2h} e_{2h}^m = d_{2h} \tag{3.27}$$

where

$$A_{2h} = \frac{1}{4h^2} \begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{pmatrix}$$

and

$$e_{2h}^m = \left(e_{2h}^m(2h), e_{2h}^m(4h), \dots, e_{2h}^m(1-2h)\right)^T.$$

We have to define the right hand of (3.27). This is done by a *restriction* of d_h onto the rougher mesh G_{2h}

$$d_{2h} = R d_h \qquad R \in \mathbb{R}^{(M-1)\times(2M-1)} \tag{3.28}$$

1. $d_{2h}(x) = d_h(x)$ $(x \in G_{2h})$

$$\Rightarrow R = \begin{pmatrix} 0\;1\;0 & & & \\ & 0\;1\;0 & & \\ & & \ddots & \\ & & & 0\;1\;0 \end{pmatrix}$$

2. $d_{2h}(x) = \frac{1}{4}\left(d_h(x-h) + 2d_h(x) + d_h(x+h)\right)$ $(x \in G_{2h})$

$$\Rightarrow R = \frac{1}{4} \begin{pmatrix} 1\;2\;1 & & & \\ & 1\;2\;1 & & \\ & & \ddots & \\ & & & 1\;2\;1 \end{pmatrix}$$

Solve on G_{2h}:

$$e_{2h}^m = A_{2h}^{-1} d_{2h} \tag{3.29}$$

This solution has to be *prolongated* onto the fine mesh:

$$\tilde{e}_h^m = P e_{2h}^m \tag{3.30}$$

where $P \in \mathbb{R}^{(2M-1)\times(M-1)}$.

$$\tilde{e}_h^m(x) = \begin{cases} e_{2h}^m(x) & x \in G_{2h} \\ \frac{1}{2}\left(e_{2h}^m(x-h) + e_{2h}^m(x+h)\right) & x \in G_h \setminus G_{2h} \end{cases}$$

$$\Rightarrow P = \frac{1}{2} \begin{pmatrix} 1 & & & \\ 2 & & & \\ 1 & 1 & & \\ & 2 & & \\ & 1 & & \\ & & \ddots & \\ & & & 1 \\ & & & 2 \\ & & & 1 \end{pmatrix}$$

It holds

$$e_{2h}^m \overset{(3.27)}{=} A_{2h}^{-1} d_{2h} \overset{(3.28)}{=} A_{2h}^{-1} R d_h \overset{(3.25)}{=} A_{2h}^{-1} R(A_h \bar{u}_h^m - f) \tag{3.31}$$

Thus $\tilde{e}^m$ is an approximation of $e_h^m = \bar{u}_h^m - u_h^*$ and therefore $\bar{u}_h^m - \tilde{e}_h^m$ is an approximation of the exact solution u_h^*, which we name u_h^{m+1}, i. e. $u_h^{m+1} = \bar{u}_h^m - \tilde{e}_h^m$. Inserting (3.31) gives

$$u_h^{m+1} = \bar{u}_h^m - PA_{2h}^{-1} R(A_h \bar{u}_h^m - f_h)$$

The Two Level Algorithm

Starting with u_h^0 apply the following algorithm:
$m = 0, 1, 2, \dots$

1. ν iterations using the smoother:

$$\begin{aligned} u_h^{m(0)} &= u_h^m \\ u_h^{m(j+1)} &= u_h^{m(j)} - \frac{h^2}{4}(A_h u_h^{m(j)} - f_h) \qquad j = 0, \dots, \nu - 1 \\ u_h^{m(\nu)} &=: \bar{u}_h^m \end{aligned}$$

2. Calculation of the defect $d_h = A_h \bar{u}_h^m - f_h$
3. Restriction of d_h onto G_{2h}: $d_{2h} = Rd_h$
4. Solve $A_{2h} e_{2h}^m = d_{2h}$ on G_{2h}
5. Prolongation of e_{2h}^m onto G_h and calculation of the $(m+1)$-th iterative:

$$u_h^{m+1} = \bar{u}_h^m - Pe_{2h}^m$$

In general

$$u_h^{m+1} = M_h(\nu) u_h^m + N_h(x) f_h$$

using the iteration matrix

$$M_h = (I - PA_{2h}^{-1} R A_h) S_h^\nu$$

Proof. For $j = 0, \dots, \nu - 1$ we calculate

$$u_h^{m(j+1)} = \underbrace{\left(I - \frac{h^2}{4} A_h\right)}_{S_h} u_h^{m(j)} + \frac{h^2}{4} f_h.$$

Applying the complete induction argument, we can show

$$\bar{u}_h^m = u_h^{m(\nu)} = S_h^\nu u_h^m + \frac{h^2}{4} \left(\sum_{i=0}^{\nu-1} S_h^i \right) f_h. \tag{*}$$

As $\rho(S_h) < 1$ $S_h - I$ is not singular. It holds (geometrical series)

$$\sum_{i=1}^{\nu-1} S_h^i = (S_h - I)^{-1}(S_h^\nu - I).$$

Hence (*) becomes

$$\bar{u}_h^m = S_h^\nu u_h^m + \frac{h^2}{4}(S_h - I)^{-1}(S_h^\nu - I) f_h$$

and therefore

$$
\begin{aligned}
u_h^{m+1} &= \bar{u}_h^m - Pe_{2h}^m = \bar{u}_h^m - PA_{2h}^{-1} \underbrace{d_{2h}}_{Rd_h} = \bar{u}_h^m - PA_{2h}^{-1} R(A_h \bar{u}_h^m - f_h) \\
&= (I - PA_{2h}^{-1} RA_h)\bar{u}_h^m + PA_{2h}^{-1} Rf_h \\
&= (I - PA_{2h}^{-1} RA_h)\big(S_h^\nu u_h^m + \tfrac{h^2}{4}(S_h - I)^{-1}(S_h^\nu - I)f_h\big) + PA_{2h}^{-1} Rf_h \\
&= \underbrace{(I - PA_{2h}^{-1} RA_h)S_h^\nu}_{M_h(\nu)} u_h^m \\
&\qquad + \underbrace{\big(\tfrac{h^2}{4}(I - PA_{2h}^{-1} RA_h)(S_h - I)^{-1}(S_h^\nu - I) + PA_{2h}^{-1} R\big)}_{N_h(\nu)} f_h
\end{aligned}
$$

The two level algorithm converges if and only if for all start vectors holds $\rho(M_h(\nu)) < 1$. Thus we can formulate the following theorem

Theorem 3.3. *[3] With the constants ρ_ν, σ_ν (only depending on ν)*

$$
\begin{aligned}
\rho_\nu &:= \max_{0 \le r \le \frac{1}{2}} \big(r(1-r)^\nu + (1-r)r^\nu\big) < 1 \\
\sigma_\nu &:= \max_{0 \le s \le \frac{1}{2}} \Big(2\big(s^2(1-s)^{2\nu} + (1-s^2)s^{2\nu}\big)\Big)^{\frac{1}{2}} < 1,
\end{aligned}
$$

there holds

$$
\begin{aligned}
\rho\big(M_h(\nu)\big) &\le \rho_\nu - |O(h^2)| \\
\|M_h(\nu)\|_2 &\le \sigma_\nu - |O(h^2)|
\end{aligned}
$$

Remark 1 *ρ_ν and σ_ν are independent of h, i. e. the two level algorithm converges for all h.*

Proof. (Sketch) B_h and A_h have the same eigenvectors

$$
w_h^i = \sqrt{2h}[\sin(i\pi h), \sin(2i\pi h), \ldots, \sin((2M-1)i\pi h)]^T
$$

and w_h^i are an orthonormal basis. Define the unitary $(2M-1) \times (2M-1)$- matrices

$$
\begin{aligned}
W_h &= (w_h^1, w_h^{2M-1}, w_h^2, \ldots, w_h^{m+1}, w_h^m) \\
W_{2h} &= (w_{2h}^1, w_{2h}^2, \ldots, w_{2h}^{N-1})
\end{aligned}
$$

where w_{2h}^i are the eigenvectors of A_{2h}. Let

$$
\begin{aligned}
\bar{A}_h &:= {W_h}^T A_h W_h & \bar{P} &:= {W_h}^T P W_{2h} \\
\bar{A}_{2h} &:= {W_{2h}}^T A_{2h} W_{2h} & \bar{R} &:= {W_{2h}}^T R W_h \\
\bar{B}_h &:= {W_h}^T B_h W_h.
\end{aligned}
$$

With $s_\mu := \sin \frac{\mu\pi h}{2}$, $c_\mu := \cos \frac{\mu\pi h}{2}$, the matrices $\bar{A}_h$, $\bar{A}_{2h}^m$ and $\bar{B}_h$ have block diagonal form with blocks $(\mu = 1, \ldots, M-1)$

$$
\begin{aligned}
\bar{B}^{(\mu)} &= \begin{pmatrix} c_\mu^2 & 0 \\ 0 & s_\mu^2 \end{pmatrix} & \bar{B}^{(M)} &= \frac{1}{2} \\
\bar{A}_h^{(\mu)} &= \frac{4}{h^2} \begin{pmatrix} s_\mu^2 & 0 \\ 0 & c_\mu^2 \end{pmatrix} = \frac{4}{h^2}(I - \bar{B}_h^{(\mu)}) & \bar{A}_h^{(M)} &= \frac{2}{h^2} \\
\bar{A}_{2h}^{(\mu)} &= \frac{4}{h^2} s_\mu^2 c_\mu^2,
\end{aligned}
$$

$$\bar{P} = \begin{pmatrix} \bar{P}^{(1)} & & \\ & \ddots & \\ & & \bar{P}^{(M-1)} \\ \cdots & \cdots & \cdots \\ & & 0 \end{pmatrix} \quad \bar{R} = \begin{pmatrix} \bar{R}^{(1)} & & & \vdots \\ & \ddots & & \vdots\, 0 \\ & & \bar{R}^{(M-1)} & \vdots \end{pmatrix}$$

where $\bar{P}^{(\mu)} = \sqrt{2}(c_\mu^2, -s_\mu^2)^T$, $\bar{R}^{(\mu)} = \frac{1}{\sqrt{2}}(c_\mu^2, -s_\mu^2)$.
The block structure can also be seen for

$$\bar{M}_h(\mu) = {W_h}^T M_h(\nu) W_h$$

with blocks ($\mu = 1, \dots, M-1$)

$$\bar{M}_h^{(\mu)}(\nu) = \begin{pmatrix} s_\mu^2 & c_\mu^2 \\ s_\mu^2 & c_\mu^2 \end{pmatrix} \begin{pmatrix} c_\mu^2 & 0 \\ 0 & s_\mu^2 \end{pmatrix}^\nu \qquad \bar{M}_h^M(\nu) = 2^{-\nu}.$$

Hence,

$$\rho\big(M_h(\nu)\big) = \rho\big(\bar{M}_h(\nu)\big) = \max_{\mu=1,\dots,M} \rho\big(\bar{M}_h^{(\mu)}(\nu)\big),$$

$$\|M_h(\nu)\|_2 = \|\bar{M}_h(\nu)\|_2 = \max_{\mu=1,\dots,M} \|\bar{M}_h^{(\mu)}(\nu)\|_2.$$

Calculation of the eigenvalues of the small 2×2 blocks $\bar{M}_h^{(\mu)}(\nu)$ and $[\bar{M}_h^{(\mu)}(\nu)]^T \bar{M}_h^{(\mu)}(\nu)$ yields the assertion.

3.3 V-Cycle with one Smoothing Step per Level

Here and in the next subsection we again follow J. Bramble [2].

Now, we consider the multigrid algorithm with $p = 1$ and $m(k) = 1$, ($k = 1, \dots, J$). We want to provide an interative solution procedure for $Ax = g$. Set $B_1 := A_1^{-1}$.

Presmoothing:	$x^1 = x^0 - R_k(A_k x^0 - g)$	
Correction:	$x^2 = x^1 + B_{k-1} Q_{k-1}(g - A_k x^1)$	(3.32)
Postsmoothing:	$x^3 = x^2 - R_k(A_k x^2 - g)$	
	$B_k g := x^3$ for $x^0 = 0$	

Note that the calculation of B_k is not necessary for computation. We only need B_k for the following (abstract) analysis of the V-cycle. Obviously the V-cycle contains J iterations. For $J = 2$ we get the 2-level method.

$E^2 := I - B_J A$ is called the operator of error propagation. We study this operator in order to analyze the V-cycle.

Lemma 3.4. Recurrence formula for $I - B_k A_k$
For $K_k := I - R_k A_k$ ($k \geq 2$) we have

$$I - B_k A_k = K_k\big(I - B_{k-1} A_{k-1} P_{k-1}\big) K_k.$$

Proof. Let $x \in \mathcal{M}_k$ be an arbitrary element in $\mathcal{M}_k$ and set $g = A_k x$ and $x^0 = 0$ in (3.32). Then

$$\begin{aligned} x - x^3 &= K_k(x - x^2) \qquad \text{and} \qquad x - x^1 = K_k x \\ x - x^2 &= x - x^1 - B_{k-1} Q_{k-1}(A_k x - A_k x^1) \\ &= (I - B_{k-1} Q_{k-1} A_k)(x - x^1) \\ &\overset{4.4}{=} (I - B_{k-1} A_{k-1} P_{k-1})(x - x^1) \end{aligned}$$

Hence

$$\begin{aligned}(I - B_k A_k)x &= x - B_k g = x - x^3 = K_k(x - x^2)\\ &= K_k(I - B_{k-1}A_{k-1}P_{k-1})(x - x^1)\\ &= K_k(I - B_{k-1}A_{k-1}P_{k-1})K_k x\ .\end{aligned}$$

□

For $k = 2, \ldots, J$ set $T_k := R_k A_k P_k$ and $T_1 := P_1$. Remember that

$$A_k : \mathcal{M}_l \to \mathcal{M}_k \quad \langle A_k u, v\rangle = \underbrace{\langle Au, v\rangle}_{=a(u,v)} \quad \forall u, v \in \mathcal{M}_k,$$

and

$$P_k : \mathcal{M} \to \mathcal{M}_k \quad \underbrace{a(P_k u, v)}_{=\langle AP_k u, v\rangle} = \langle Au, v\rangle \quad \forall u \in \mathcal{M},\ v \in \mathcal{M}_k,$$

and that A_k and P_k were supposed to be symmetric. Then we get the following lemma.

Lemma 3.5. *The operator of error propagation consists of a product of operators*

$$E^2 = I - B_J A = (I - T_J) \cdot \ldots \cdot (I - T_2) \cdot (I - P_1)^2 \cdot (I - T_2) \cdot \ldots \cdot (I - T_J).$$

Proof. Note that for all $k = 2, \ldots, J$ we have

1. $K_k P_k = P_k - R_k A_k P_k = P_k - R_k A_k P_k^2 = (I - T_k)P_k = P_k(I - T_k)$,
2. $(I - T_k)(I - P_k)(I - T_k) = (I - T_k)(I - P_k + P_k T_k - T_k) \quad ,$
   $= (I - T_k)(I - P_k) = (I - P_k)$
   because $T_k P_k = P_k T_k = T_k$,
3. $(I - B_{k-1}A_{k-1}P_{k-1})P_k = P_k(I - B_{k-1}A_{k-1}P_{k-1})$
   $= P_k - B_{k-1}A_{k-1}P_{k-1}$.

Hence we get

$$\begin{aligned}I - B_k A_k P_k &= I - P_k + (I - B_k A_k)P_k\\ &\overset{3.4}{=} I - P_k + K_k(I - B_{k-1}A_{k-1}P_{k-1})K_k P_k\\ &\overset{1.}{=} I - P_k + K_k(I - B_{k-1}A_{k-1}P_{k-1})P_k(I - T_k)\\ &\overset{3.}{=} I - P_k + K_k P_k(I - B_{k-1}A_{k-1}P_{k-1})(I - T_k)\\ &\overset{1.}{=} I - P_k + (I - T_k)P_k(I - B_{k-1}A_{k-1}P_{k-1})(I - T_k)\\ &\overset{3.}{=} I - P_k + (I - T_k)(P_k - B_{k-1}A_{k-1}P_{k-1})(I - T_k)\\ &\overset{2.}{=} (I - T_k)(I - P_k)(I - T_k)\\ &\quad +(I - T_k)(P_k - B_{k-1}A_{k-1}P_{k-1})(I - T_k)\\ &= (I - T_k)(I - B_{k-1}A_{k-1}P_{k-1})(I - T_k)\end{aligned}$$

and since $B_1 = {A_1}^{-1}$ the lemma is proven. □

For our following analysis we will need some assumptions:

Assumptions

Let λ_k be the largest eigenvalue of A_k for every $k = 2, \ldots, J$. Then we have for $k = 2, \ldots, J$:

(S1) $\exists c_R \geq 1$:

$$\frac{\|u\|^2}{\lambda_k} \leq c_R \langle R_k u, u\rangle \quad \forall u \in \mathcal{M}_k.$$

(S2) $\exists \theta < 2 \quad \forall k = 1, \ldots, J$:

$$\langle AT_k u, T_k u \rangle \leq \theta a(u,u) \quad \forall u \in \mathcal{M}_k.$$

(B) a) $\exists c_a > 0$:

$$\|(Q_k - Q_{k-1})u\|^2 \leq c_a \lambda_k^{-1} a(u,u) \quad \forall u \in \mathcal{M}_k.$$

b) $\exists c_b > 0$:

$$a(Q_k u, u) \leq c_b a(u,u) \quad \forall u \in \mathcal{M}.$$

Lemma 3.6. *Suppose that the assumptions (S1), (S2) and (B) hold. Then we have for the V-cycle*

$$a(E_J v, E_J v) \leq \left(1 - \frac{1}{c(J-1)}\right) a(v,v) \quad \forall v \in \mathcal{M}_J,$$

where $c > 0$ and with $E_0 := I$ and $E_i := (I - T_i)E_{i-1}$ $i = 1, \ldots, J$.

Proof. The definition of E_i yields

$$E_{i-1} - E_i = T_i E_{i-1}$$

and this implies

$$\sum_{l=1}^{i} T_l E_{l-1} = E_0 - E_i = I - E_i. \tag{3.33}$$

Let $u \in \mathcal{M}_J$, then

$$\begin{aligned}
\langle Au, u \rangle &= \sum_{k=2}^{J} \langle Au, (Q_k - Q_{k-1})u \rangle + \langle Au, Q_1 u \rangle \\
&= \underbrace{\sum_{k=2}^{J} \langle AE_{k-1}u, (Q_k - Q_{k-1})u \rangle}_{:=S_1} \\
&\quad + \underbrace{\sum_{k=2}^{J} \langle A(I - E_{k-1})u, (Q_k - Q_{k-1})u \rangle + \langle Au, Q_1 u \rangle}_{:=S_2}.
\end{aligned}$$

Since $\langle Aw, v \rangle = \langle A_k P_k w, v \rangle \quad \forall w \in \mathcal{M}, v \in \mathcal{M}_k$ (see definition of P_k) we get

$$\begin{aligned}
S_1 &\leq \sum_{k=2}^{J} \|A_k P_k E_{k-1} u\| \cdot \|(Q_k - Q_{k-1})u\| \\
&\overset{(S1)}{\leq} \sum_{k=2}^{J} \Big(c_R \lambda_k \langle \underbrace{R_k A_k P_k}_{=T_k} E_{k-1}u, A_k P_k E_{k-1} u \rangle \|(Q_k - Q_{k-1})u\|^2 \Big)^{\frac{1}{2}} \\
&\overset{(B)}{\leq} \sum_{k=2}^{J} \Big(c_R \lambda_k \langle A_k P_k E_{k-1}u, T_k E_{k-1} u \rangle \, c_a \lambda_k^{-1} \langle Au, u \rangle \Big)^{\frac{1}{2}} \\
&= \sum_{k=2}^{J} \Big(\tilde{c} \langle AT_k E_{k-1}u, E_{k-1}u \rangle \langle Au, u \rangle \Big)^{\frac{1}{2}} \\
&\overset{C.S.}{\leq} \left(\sum_{k=2}^{J} \tilde{c} \langle AT_k E_{k-1}u, E_{k-1}u \rangle \right)^{\frac{1}{2}} \left(\sum_{k=2}^{J} \langle Au, u \rangle \right)^{\frac{1}{2}}
\end{aligned}$$

$$= \langle Au,u\rangle^{\frac{1}{2}} \left(\tilde{c}(J-1)\sum_{k=2}^{J}\langle AT_kE_{k-1}u,E_{k-1}u\rangle\right)^{\frac{1}{2}}$$

Now we estimate S_2. With $T_1 = P_1$ we have

$$\begin{aligned}
S_2 &= \sum_{k=2}^{J}\langle A(I-E_{k-1})u,(Q_k-Q_{k-1})u\rangle + \langle AP_1u,Q_1u\rangle \\
&= \sum_{k=2}^{J}\langle A(I-E_{k-1})u,Q_ku\rangle - \sum_{k=1}^{J-1}\langle A(I-E_k)u,Q_ku\rangle + \langle AT_1u,Q_1u\rangle \\
&= -\sum_{k=2}^{J-1}\langle AT_kE_{k-1}u,Q_ku\rangle + \big\langle A(I-E_{J-1})u,\underbrace{Q_J}_{=I}u\big\rangle \\
&\overset{(3.33)}{=} -\sum_{k=2}^{J-1}\langle AT_kE_{k-1}u,Q_ku\rangle + \sum_{k=1}^{J-1}\langle AT_kE_{k-1}u,u\rangle \\
&\overset{C.S.}{\leq} \left(\sum_{k=2}^{J-1}\langle AT_kE_{k-1}u,T_kE_{k-1}u\rangle\sum_{k=2}^{J-1}\langle AQ_ku,Q_ku\rangle\right)^{\frac{1}{2}} \\
&\quad + \left(\sum_{k=1}^{J-1}\langle AT_kE_{k-1}u,T_kE_{k-1}u\rangle\sum_{k=1}^{J-1}\langle Au,u\rangle\right)^{\frac{1}{2}} \\
&\overset{(B)}{\leq} \left(\sum_{k=1}^{J-1}\langle AT_kE_{k-1}u,T_kE_{k-1}u\rangle\sum_{k=1}^{J-1}c_b\,\langle Au,u\rangle\right)^{\frac{1}{2}} \\
&\quad + \left(\sum_{k=1}^{J-1}\langle AT_kE_{k-1}u,T_kE_{k-1}u\rangle\,(J-1)\,\langle Au,u\rangle\right)^{\frac{1}{2}} \\
&= \left(\sum_{k=1}^{J-1}\langle AT_kE_{k-1}u,T_kE_{k-1}u\rangle\,(J-1)\,\langle Au,u\rangle\right)^{\frac{1}{2}}(1+\sqrt{c_b}) \\
&\overset{(S2)}{\leq} \left((J-1)\,\langle Au,u\rangle\sum_{k=1}^{J}\langle AT_kE_{k-1}u,E_{k-1}u\rangle\right)^{\frac{1}{2}}\sqrt{\theta}\,(1+\sqrt{c_b})
\end{aligned}$$

Hence we get

$$\langle Au,u\rangle \leq c'(J-1)\sum_{k=1}^{J}\langle AT_kE_{k-1}u,E_{k-1}u\rangle\,. \tag{3.34}$$

And with

$$\begin{aligned}
&(2-\theta)\,\langle AT_kE_{k-1}u,E_{k-1}u\rangle \\
&\leq 2\,\langle AT_kE_{k-1}u,E_{k-1}u\rangle - \langle AT_kE_{k-1}u,T_kE_{k-1}u\rangle \\
&= \langle AT_kE_{k-1}u,E_{k-1}u\rangle + \langle AT_kE_{k-1}u,(I-T_k)E_{k-1}u\rangle \\
&= \langle AT_kE_{k-1}u,(E_k+E_{k-1})u\rangle = \langle A(E_{k-1}-E_k)u,(E_k+E_{k-1})u\rangle \\
&= \langle AE_{k-1}u,E_{k-1}u\rangle - \langle AE_ku,E_ku\rangle\,,
\end{aligned}$$

(3.34) can be written as

$$\langle Au,u\rangle \leq c(J-1)\big(\langle Au,u\rangle - \langle AE_Ju,E_Ju\rangle\,.$$

□

3.4 The V-Cycle with constant Smoothing Steps

We consider the multigrid algorithm with constant ($m = m(k)$) smoothing steps and $p = 1$. Again we provide an iterative solution procedure for $Ax = g$. We have m presmoothing steps ($x^1, \ldots, x^m$), one correction (y^m) and m postsmoothing steps ($y^{m+1}, \ldots, y^{2m}$).

V-cycle with constant smoothing steps
Define $B_1 := A_1^{-1}$.

$$\boxed{\begin{array}{ll} \textbf{Presmoothing:} & x^l = x^{l-1} - R_k(A_k x^{l-1} - g) \quad l = 1, \ldots, m \\ \textbf{Correction:} & y^m = x^m + B_{k-1}Q_{k-1}(g - A_k x^m) \\ \textbf{Postsmoothing:} & y^l = y^{l-1} - R_k(A_k y^{l-1} - g) \quad l = m+1, \ldots, 2m \\ & B_k g = y^{2m} \text{ for } x^0 = 0 \end{array}} \tag{3.35}$$

Again it is helpfully to study the operator $E = I - B_J A$ in order to get convergence results.

Lemma 3.7. Recurrence formula for $I - B_k A_k$
Set $K_k := I - R_k A_k$. K_k *satisfies for all* $k \geq 2$

$$I - B_k A_k = {K_k}^m \big(I - B_{k-1} A_{k-1} P_{k-1}\big) {K_k}^m.$$

Proof. The action of $I - B_k A_k$ on $x \in \mathcal{M}_k$ is obtained by (3.35) with $g := A_k x$ and $x^0 = 0$. Note that for $l = 1, \ldots, m$

$$\begin{aligned} x - x^l =& x - x^{l-1} + R_k(A_k x^{l-1} - g) = x - x^{l-1} + R_k A_k (x^{l-1} - x) \\ =& (I - R_k A_k)(x - x^{l-1}) = K_k(x - x^{l-1}). \end{aligned}$$

Thus

$$x - x^m = {K_k}^m x \text{ and } x - y^{2m} = {K_k}^m (x - y^m)$$

and

$$x - y^m = (I - B_{k-1} A_{k-1} P_{k-1})(x - x^m)$$

and as in Lemma 3.4 the assertion follows. □

Now we prove a useful result due to the fact that the Galerkin error is orthogonal to the space of the test functions with respect to the corresponding inner product.

Lemma 3.8. *For all* $u \in \mathcal{M}$ *it is*

$$a((I - P_k)u, (I - P_k)u) = a((I - P_k)u, u), \qquad \forall k = 1, \ldots, J.$$

Proof.

$$\begin{aligned} & a((I - P_k)u, (I - P_k)u) = a((I - P_k)u, u) - a((I - P_k)u, P_k u) \\ & = a((I - P_k)u, u) - a(u, P_k u) + a(P_k u, P_k u) \\ & = a((I - P_k)u, u) - a(u, P_k u) + a(u, P_k u) = a((I - P_k)u, u) \end{aligned}$$

□

Lemma 3.9. $\forall k = 1, \ldots, J \quad \forall u \in \mathcal{M}_k : \qquad a((I - B_k A_k)u, u) \geq 0$

Proof. We will show this assertion by induction:

$k = 1$: Since $B_1 = {A_1}^{-1}$ we get $I - B_1 A_1 = 0$. The assertion follows immediately for $k = 1$.

$k \to k + 1$: Take a $u \in \mathcal{M}_{k+1}$ and set $v := (K_{k+1})^m u$. With Lemma 3.7 we get

$$\begin{aligned} & a((I - B_{k+1} A_{k+1})u, u) = a\big(\big((I - P_k) + (I - B_k A_k) P_k\big) v, v\big) \\ & = a((I - P_k)v, v) + a((I - B_k A_k) P_k v, P_k v) \\ & \geq a((I - P_k)v, v) \overset{3.8}{=} a((I - P_k)v, (I - P_k)v) \geq 0. \end{aligned}$$

□

Assumption: Regularity Assumption

There exists an $\alpha \in (0,1]$ and a constant C_α such that for all $k = 1,\dots,J$ and for all $u \in \mathcal{M}$

(A_1): $\left\langle A^{1-\alpha} Q_k u, Q_k u \right\rangle \le C_\alpha \cdot \left\langle A^{1-\alpha} u, u \right\rangle$
(A_2): $\left\langle A^{1-\alpha} (I - P_k) u, (I - P_k) u \right\rangle \le C_\alpha \lambda_{\max}^{-\alpha} \cdot A \langle u, u \rangle$

($\alpha = 1$ corresponds to full regularity)

In order to prove the following theorem we need the next lemma:

Lemma 3.10. *Let $C \in \mathbb{R}^{N\times N}$ be a symmetric matrix.*

a) If every eigenvalue λ of C satisfies $\lambda \in [a,b]$ and if for

$$p(x) := \sum_{i=1}^{N_p} \hat{p}_i x^{p_i} \ (p_i > 0) \quad \text{and} \quad q(x) := \sum_{i=1}^{N_q} \hat{q}_i x^{q_i} \ (q_i > 0)$$

there holds $p(x) \le q(x) \quad \forall x \in [a,b]$ then we have

$$\left\langle p(C)u, u \right\rangle \le \left\langle q(C)u, u \right\rangle, \qquad \forall u \in \mathbb{R}^N \ .$$

b) Let $\sigma(C)$ denote the set of eigenvalues (spectrum) of C. Then with C and every $p(x)$ as described in a) there holds

$$\min_{\lambda\in\sigma(C)} |p(\lambda)| \le |||p(C)||| \le \max_{\lambda\in\sigma(C)} |p(\lambda)| \ ,$$

where $|||\cdot|||$ denotes the matrix norm

$$|||p(C)||| := \sup_{u\in\mathbb{R}^N} \frac{\langle p(C)u, u\rangle}{\langle u, u\rangle}$$

Theorem 3.11. *[2] Assume that Richardson's method is used as smoother in (3.35),i.e. ($R_k = \lambda_{\max}^{-1}, K_k = I - \lambda_{\max}^{-1} A_k$). Let the regularity assumptions $(A_1), (A_2)$ be fulfilled with $\alpha = 1$, then for all $k = 1,\dots,J$ and for all $u \in \mathcal{M}_k$ it is*

$$0 \le a((I - B_k A_k)u, u) \le \rho \cdot A \langle u, u \rangle \,, \qquad \rho = \frac{C_1}{2m + C_1} \ .$$

Proof. By Lemma 3.9 we have that $a((I - B_k A_k)u, u)$ is not negative. Let us prove the other inequality by induction. $k = 1$: Since $I - B_1 A_1 = 0$ nothing must be shown.

$k \to k+1$: We assume that there exists a k such that the statement is satisfied. We take $u \in \mathcal{M}_{k+1}$ and set $v := (K_{k+1})^m u$. As in Lemma 3.9 we get for $k+1$

$$\begin{aligned} & a((I - B_{k+1} A_{k+1})u, u) = a((I - P_k)v, v) + a((I - B_k A_k) P_k v, P_k v) \\ & \le a((I - P_k)v, v) + \rho A \langle P_k v, P_k v\rangle = a((I - P_k)v, v) + \rho A \langle P_k v, v \rangle \\ & = (1-\rho) a((I - P_k)v, v) + \rho A \langle v, v\rangle \,. \end{aligned}$$

Setting $w := (I - P_k)v$ we get

$$\begin{aligned} & a((I - P_k)v, v) = a((I - P_k)^2 v, v) = \left\langle (I - P_k)w, A_{k+1} v \right\rangle \\ & \le \|(I - P_k)w\| \cdot \|A_{k+1}v\| \overset{(A_2)}{\le} \sqrt{\tfrac{C_1}{\lambda_{\max}}} A \langle w, w\rangle^{\frac{1}{2}} \, \|A_{k+1} v\| \\ & \overset{3.8}{=} \sqrt{\tfrac{C_1}{\lambda_{\max}}} a((I - P_k)v, v)^{\frac{1}{2}} \|A_{k+1} v\| \end{aligned}$$

and thus

$$a((I-P_k)v,v) \leq \tfrac{C_1}{\lambda_{\max}}\|A_{k+1}v\|^2 = \tfrac{C_1}{\lambda_{\max}}\|A_{k+1}(K_{k+1})^m u\|^2$$
$$= \tfrac{C_1}{\lambda_{\max}}\|(K_{k+1})^m A_{k+1}u\|^2 = \tfrac{C_1}{\lambda_{\max}}\big\langle (K_{k+1})^{2m}A_{k+1}u, A_{k+1}u\big\rangle.$$

For all $z \in [0,1]$ we have

$$z^{2m} \leq \frac{1}{2m}\sum_{i=0}^{2m-1} z^i. \tag{3.36}$$

and K_{k+1} is symmetric and positive semidefinite

$$\big\langle K_{k+1}u, u\big\rangle = \langle u,u\rangle - \frac{1}{\lambda_{\max}}A\langle u,u\rangle \geq \langle u,u\rangle - \langle u,u\rangle = 0.$$

Let $\lambda_1,\dots,\lambda_n$ be the eigenvalues and $\varphi_1,\dots,\varphi_n$ be orthonormal eigenvectors of K_{k+1} ($\dim\mathcal{M}_{k+1} = n$). Then every λ_i is bounded by 1:

$$\lambda_i = \langle K_{k+1}\varphi_i, \varphi_i\rangle = \langle \varphi_i,\varphi_i\rangle - \frac{1}{\lambda_{\max}}A\langle \varphi_i,\varphi_i\rangle \leq \langle \varphi_i,\varphi_i\rangle = 1\ .$$

Thus we can apply Lemma 3.10 with (3.36) and K_{k+1}.

$$\begin{aligned}
a((I-P_k)v,v) &\leq \tfrac{C_1}{\lambda_{\max}}\big\langle (K_{k+1})^{2m}A_{k+1}u, A_{k+1}u\big\rangle \\
&\overset{3.10}{\leq} \tfrac{C_1}{\lambda_{\max}2m}\sum_{i=0}^{2m-1}\big\langle (K_{k+1})^{i}A_{k+1}u, A_{k+1}u\big\rangle \\
&= \tfrac{C_1}{2m}\sum_{i=0}^{2m-1}\big\langle \tfrac{A_{k+1}}{\lambda_{\max}}(K_{k+1})^{i}u, A_{k+1}u\big\rangle \\
&= \tfrac{C_1}{2m}a((I-K_{k+1})\sum_{i=0}^{2m-1}(K_{k+1})^{i}u, u) \\
&= \tfrac{C_1}{2m}a\big(\big(I-(K_{k+1})^{2m}\big)u,u\big) \\
&= \tfrac{C_1}{2m}\left(A\langle u,u\rangle - A\langle v,v\rangle\right)
\end{aligned}$$

Note, that ρ must fulfill

$$\rho = (1-\rho)\frac{C_1}{2m} \quad\Longrightarrow\quad \rho = \frac{C_1}{2m+C_1}\ . \tag{3.37}$$

Hence

$$\begin{aligned}
a((I-B_{k+1}A_{k+1})u,u) &\leq (1-\rho)a((I-P_k)v,v) + \rho A\langle v,v\rangle \\
&\leq (1-\rho)\tfrac{C_1}{2m}\left(A\langle u,u\rangle - A\langle v,v\rangle\right) + \rho A\langle v,v\rangle \\
&\overset{(3.37)}{=} \rho\left(A\langle u,u\rangle - A\langle v,v\rangle\right) + \rho A\langle v,v\rangle = \rho A\langle u,u\rangle\ .
\end{aligned}$$

□

3.4.1 Example

As in section 3.1 we consider the problem

$$\boxed{\begin{array}{l} -\triangle u = f \text{ in}\Omega \\ u = 0 \text{ in } \partial\Omega \end{array}} \tag{3.38}$$

Take a quasi-uniform grid on Ω for discretization, i.e.

$$\exists \hat{c}_0,\hat{c}_1 \quad \forall k = 1,\dots,J-1 \qquad \hat{c}_0 h_k \leq h_{k+1} \leq \hat{c}_1 h_k\ . \tag{3.39}$$

We use Galerkin's method to obtain an approximative solution $u_h \in \mathcal{M}_k$ for the exact solution u:

$$\boxed{\begin{array}{l}\text{Find } u_h \in \mathcal{M}_k \subseteq H_0^1(\Omega) \text{ satisfying} \\ a(u_h,\chi) := \int\limits_\Omega \nabla u_h \nabla \chi = \int\limits_\Omega f\chi \quad ,\forall \chi \in \mathcal{M}_k\end{array}} \tag{3.40}$$

$a(\,\cdot\,,\,\cdot\,)$ is continuous, symmetric and positive definite and by application of the lemma of Lax-Milgram we obtain that there exists one and only one u_h that solves (3.40), respectively one and only one u that solves (3.38) and these u_{h_k} satisfy

$$\|u_{h_k} - u\|_{H^1(\Omega)} \le c_1 \cdot \inf_{\chi \in \mathcal{M}_k} \|\chi - u\|_{H^1(\Omega)}. \tag{3.41}$$

If Ω is convex and $f \in \mathrm{L}^2(\Omega)$ we know that the solution u lies in $H^2(\Omega)$ and satisfies the approximation property:

$$\inf_{\chi \in \mathcal{M}_k} \|\chi - u\|_{H^1(\Omega)} \le c_2 \cdot h_k \|u\|_{H^2(\Omega)} \le c_3 \cdot h_k \|\triangle u\|_{L^2(\Omega)} \tag{3.42}$$

To estimate $\|u_{h_k} - u\|_{L^2(\Omega)}$ we use the Aubin-Nitsche trick:
Consider the problem:

$$\boxed{\begin{array}{l}-\triangle v = u_{h_k} - u \text{ in} \Omega \\ v = 0 \text{ in } \partial\Omega\end{array}}$$

All $\chi \in \mathcal{M}_k$ staisfy $a(u_{h_k},\chi) = a(u,\chi)$. Hence:

$$\begin{aligned}
&\|u_{h_k} - u\|^2_{L^2(\Omega)} = \langle u_{h_k} - u, u_{h_k} - u\rangle = \langle u_{h_k} - u, -\triangle v\rangle \\
=\ & \langle \nabla(u_{h_k} - u), \nabla v\rangle = a(u_{h_k} - u, v) \\
=\ & a(u_{h_k} - u, v - \chi) \le a(u_{h_k} - u, u_{h_k} - u)^{\frac{1}{2}} a(v - \chi, v - \chi)^{\frac{1}{2}} \\
=\ & \|u_{h_k} - u\|_{H^1(\Omega)} \|v - \chi\|_{H^1(\Omega)} \overset{(3.42)}{\le} \|u_{h_k} - u\|_{H^1(\Omega)} c_2 \cdot h_k \| - \triangle v\|_{L^2(\Omega)} \\
=\ & \|u_{h_k} - u\|_{H^1(\Omega)} c_2 \cdot h_k \|u_{h_k} - u\|_{L^2(\Omega)}\ . \\
\Longrightarrow\ & \|u_{h_k} - u\|_{L^2(\Omega)} \le c_2 \cdot h_k \|u_{h_k} - u\|_{H^1(\Omega)}\ .
\end{aligned}$$

By definition of P_k we see, that P_k is the Galerkin projection on $\mathcal{M}_{k-1}$. We prove that the regularity assumption (A_2) is satisfied for $\alpha = 1$ (note, that (A_1) is always fulfilled for $\alpha = 1$).

$$\begin{aligned}
\|(I - P_k)u\|^2_{L^2(\Omega)} &= \|u - u_{h_{k-1}}\|^2_{L^2(\Omega)} \\
&\le c_2{}^2 h^2_{k-1} \|u_{h_{k-1}} - u\|^2_{H^1(\Omega)} \\
&\overset{(3.14)\,(3.39)}{\le} c_3 \lambda_k^{-1} \|u_{h_{k-1}} - u\|^2_{H^1(\Omega)} \\
&\overset{(3.41)}{\le} c_4 \lambda_k^{-1} \|u\|^2_{H^1(\Omega)} \le c_5 \lambda_k^{-1} a(u,u)\ .
\end{aligned}$$

Remark 2 *The above analysis is extended to the hypersingular boundary integral equation in [7]. For a different analysis of the multigrid method to the hypersingular integral equation see [6].*

4

Schwarz and Multilevel Methods

4.1 Additive and Multiplicative Schwarz Methods

Schwarz methods (additive and multiplicative) are domain decomposition methods. Again we consider the problem

$$\boxed{\begin{array}{l} -u'' = f \text{ in } (0,1)\ ,\ f \in L^2(0,1) \\ u(0) = u(1) = 0\ ,\ u \in \overset{\circ}{H}{}^1(0,1), \end{array}} \tag{4.1}$$

where $\overset{\circ}{H}{}^1(0,1)$ denotes the usual Sobolev space $\{u, u' \in L^2(0,1), u(0) = u(1) = 0\}$. This problem is equivalent to the weak formulation

$$a(u,v) := \int_0^1 u'\, v' = \int_0^1 f\, v =: f(v) \quad \forall v \in \overset{\circ}{H}{}^1(0,1). \tag{4.2}$$

Now, the Galerkin method is: find $u_h \in \overset{\circ}{S}{}^1_h(0,1) = \{w \in C^0(0,1) : w$ is piecewise linear on an equidistant mesh with meshsize $h, w(0) = w(1) = 0\} \subset \overset{\circ}{H}{}^1(0,1)$, such that

$$\int_0^1 u_h'\, \tilde{v}' = \int_0^1 f\, \tilde{v} \quad \forall v \in \overset{\circ}{S}{}^1_h(0,1). \tag{4.3}$$

Note that for w piecewise linear w' is piecewise constant with jumps of finite height, i.e. w' is in $L^2(0,1)$. This fact ensures that $\overset{\circ}{S}{}^1_h(0,1) \subset \overset{\circ}{H}{}^1(0,1)$. Now, we may write u_h as

$$u_h(x) = \sum_{j=1}^{N-1} c_j \phi_j(x), \tag{4.4}$$

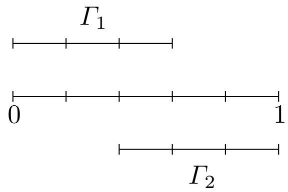
where the sum is performed over all interior nodes and the basis functions ϕ are given by

$$\phi_j(x_k) = \begin{cases} 1\ , k = j \\ 0\ , \mathit{otherwise.} \end{cases}. \tag{4.5}$$

Hence from (4.3) we get a linear system of the form

$$A\mathbf{c} = \mathbf{b},$$

where A is symmetric and positive definite . If the matrix A is too large it seems to be useful to split the problem into several smaller ones, for instance we might split up $\varGamma = (0,1)$ into overlapping intervals $\varGamma_1$ and $\varGamma_2$.



Let $S_1 := \overset{\circ}{S}{}^1_h(\Gamma_1)$ and $S_2 := \overset{\circ}{S}{}^1_h(\Gamma_2)$. Then the Galerkin method (4.3) becomes

$$a(u_h, v) = f(v) \quad \forall v \in \overset{\circ}{S}{}^1_h(\Gamma) = S_1 + S_2. \tag{4.6}$$

Then the alternating multiplicative Schwarz method (MSM) is as follows:
(MSM) For $n = 0, 1, \ldots$:
Find the corrections $\delta_1 u^n \in S_1$ for the approximation u^n of $u_h \in S_1 + S_2$

$$a(\delta_1 u^n, v) = f(v) - a(u^n, v) = a(u_h - u^n, v) \quad \forall v \in S_1.$$

Find the corrections $\delta_2 u^n \in S_2$ for the approximation $u^n + \delta_1 u^n$ of $u_h \in S_1 + S_2$

$$a(\delta_2 u^n, v) = f(v) - a(u^n + \delta_1 u^n, v) = a(u_h - (u^n + \delta_1 u^n), v) \quad \forall v \in S_2.$$

Then

$$u^{n+1} := u^n + \delta_1 u^n + \delta_2 u^n.$$

Now we have that

$$\delta_1 u^n = P_1(u_h - u^n),$$

where $P_j : S := \overset{\circ}{S}{}^1_h(\Gamma) \to S_j$, $j = 1, 2$, denotes the projection with respect to $a(\cdot, \cdot)$. Then

$$\delta_2 u^n = P_2(u_h - u^n - \delta_1 u^n) = P_2(I - P_1)(u_h - u^n)$$

and hence

$$\begin{aligned} u^{n+1} - u_h &= u^n - u_h + \delta_1 u^n + \delta_2 u^n \\ &= u^n - u_h + P_1(u_h - u^n) + P_2(I - P_1)(u_h - u^n) \\ &= (I - P_2)(I - P_1)(u^n - u_h). \end{aligned}$$

Thinking about the question of convergence of u^{n+1} to u_h we have to analyze the norm of $(I - P_2)(I - P_1)$.

4.2 General Frame for Schwarz Methods

Consider the decomposition of V ($\dim V < \infty$) into subspaces

$$V = V_0 + V_1 + \ldots + V_{N-1},$$

and the projections $P_j : V \to V_j$, $j = 0, \ldots, N-1$ defined by

$$a(P_j v, \phi) = a(v, \phi) \quad \forall v \in V,\ \phi \in V_j. \tag{4.7}$$

The general assumption we will make througout this section is that $a(\cdot, \cdot)$ is assumed to be symmetric and positive definite on V.

The multipicative Schwarz operator is defined by

$$P_{MS} := I - E_{N-1}, \quad E_{N-1} := (I - P_{N-1}) \cdots (I - P_1)(I - P_0),$$

where I denotes the identity operator. The additive Schwarz operator is given by

$$P_{AS} := \sum_{j=0}^{N-1} P_j.$$

The difference between the multiplicative and additive Schwarz method (ASM) is that the (ASM) may be solved "in parallel", i.e. for our special case it is given by

(ASM) Solve $Pu_h = \sum_{j=0}^{N-1} P_j u_h = g_h$ with an iterative method, where $g_h = \sum_{j=0}^{N-1} g_j$ with $a(g_j, v) = f(v)$, $v \in S_j$, $j = 0, 1, \ldots, N-1$, i.e.

$$a(\delta_j u^n, v) = f(v) - a(u^n, v) \quad \forall v \in S_j \text{ "in parallel".}$$

Then

$$u^{n+1} := u^n + \theta \sum_{j=1}^{N-1} \delta_j u^n, \quad \theta \in \mathbb{R}.$$

We define the operators

$$E_{-1} := I, \quad E_i := (I - P_i)E_{i-1}, \quad i = 0, \ldots, N-1,$$

such that

$$E_{j-1} - E_j = P_j E_{j-1}$$

and

$$I - E_i = \sum_{j=0}^{i} P_j E_{j-1}, \quad i = 0, \ldots, N-1. \tag{4.8}$$

Lemma 4.1. *For all $v \in V$ we have*

$$\sum_{i=0}^{N-1} a(P_i E_{i-1} v, E_{i-1} v) = a(v, v) - a(E_{N-1} v, E_{N-1} v).$$

Proof. For $i = 0, \ldots, N-1$ we have

$$\begin{aligned} a(E_i v, E_i v) &= a((I - P_i)E_{i-1} v, (I - P_i)E_{i-1} v) \\ &= a(E_{i-1} v, E_{i-1} v) - 2a(P_i E_{i-1} v, E_{i-1} v) + a(P_i E_{i-1} v, P_i E_{i-1} v) \\ &= a(E_{i-1} v, E_{i-1} v) - a(P_i E_{i-1} v, E_{i-1} v), \end{aligned}$$

since P_i is a projection and $P_i E_{i-1} v$ is in V_i. Hence

$$a(P_i E_{i-1}, E_{i-1} v) = a(E_{i-1} v, E_{i-1} v) - a(E_i v, E_i v).$$

Thus summation over i finishes the proof. □

For the following lemma, we need the definition

$$\theta_{i,j} := \sup_{u \in V_i, v \in V_j} \frac{a(u, v)}{a(u, u)^{\frac{1}{2}}\, a(v, v)^{\frac{1}{2}}}, \quad i, j = 1, \ldots, N-1. \tag{4.9}$$

Lemma 4.2. *Let θ be a $(N-1) \times (N-1)-$matrix with entries θ_{ij}, $i, j = 1, \ldots, N-1$ as in (4.9). If there exist constants $C_1, C_2 > 0$ such that*

$$a(P_{AS} v, v) \leq C_2 a(v, v) \quad \forall v \in V$$

and $C_1 := 2 \max\{C_2, ||\theta||_2^2\}$, then we have for all $v \in V$

$$a(P_{AS} v, v) \leq C_1 \sum_{j=0}^{N-1} a(P_j E_{j-1} v, E_{j-1} v).$$

Proof. We have

$$a(P_i v, v) = \underbrace{a(P_i v, E_{i-1} v)}_{=a(P_i v, P_i E_{i-1} v)} + a(P_i v, \underbrace{(I - E_{i-1})}_{\overset{(4.8)}{=} \sum_{j=0}^{i-1} P_j E_{j-1}} v)$$

$$= \sum_{j=0}^{i} a(P_i v, P_j E_{j-1} v).$$

Hence

$$\sum_{i=1}^{N-1} a(P_i v, v) = \underbrace{\sum_{i=1}^{N-1} a(P_i v, P_0 v)}_{=:T_1} + \underbrace{\sum_{i=1}^{N-1} \sum_{j=1}^{i} a(P_i v, P_j E_{j-1} v)}_{=:T_2}. \tag{4.10}$$

With Cauchy-Schwarz we get for the first term

$$T_1 \leq \left(\sum_{i=1}^{N-1} a(P_i v, v) \right)^{\frac{1}{2}} \left(\sum_{i=1}^{N-1} a(P_i P_0 v, P_o v) \right)^{\frac{1}{2}}. \tag{4.11}$$

For the second term we get

$$\begin{aligned} T_2 &\leq \sum_{i=1}^{N-1} \sum_{j=1}^{i} \theta_{ij} a(P_i v, v)^{\frac{1}{2}} a(P_j E_{j-1} v, E_{j-1} v)^{\frac{1}{2}} \\ &\leq ||\theta||_2 \left(\sum_{i=1}^{N-1} a(P_i v, v) \right)^{\frac{1}{2}} \left(\sum_{j=1}^{N-1} a(P_j E_{j-1} v, E_{j-1} v) \right)^{\frac{1}{2}}. \end{aligned} \tag{4.12}$$

Then (4.10), (4.11) and (4.12) gives

$$\begin{aligned} \sum_{i=0}^{N-1} a(P_i v, v) &\leq 2 \sum_{i=0}^{N-1} a(P_i P_0 v, P_0 v) + 2||\theta||_2^2 \sum_{j=1}^{N-1} a(P_j E_{j-1} v, E_{j-1} v) \\ &\leq 2C_2 a(P_0 v, P_0 v) + 2||\theta||_2^2 \sum_{j=1}^{N-1} a(P_j E_{j-1} v, E_{j-1} v) \\ &\leq C_1 \sum_{j=0}^{N-1} a(P_j E_{j-1} v, E_{j-1} v) \end{aligned}$$

□

Theorem 4.3. *If there exist constants C_0, $C_2 > 0$ such that*

$$C_0 a(v, v) \leq a(P_{AS} v, v) \leq C_2 a(v, v) \quad \forall v \in V, \tag{$\star$}$$

and a constant $C_1 := 2 \max\{C_2, ||\theta||_2^2\}$, then we have

$$||E_{N-1} v||_a^2 \leq \left(1 - \frac{C_0}{C_1} \right) ||v||_a^2,$$

where $||v||_a^2 := a(v, v)$.

Proof. From Lemma 4.2 we get for all $v \in V$

$$a(v,v) \le C_0^{-1} a(P_{AS}v,v) \le C_1 C_0^{-1} \underbrace{\sum_{i=0}^{N-1} a(P_i E_{i-1}v, E_{i-1}v)}_{\overset{4.1}{=} a(v,v) - a(E_{N-1}v, E_{N-1}v)}$$

and hence

$$a(E_{N-1}v, E_{N-1}v) \le a(v,v) - \frac{C_0}{C_1} a(v,v) = \left(1 - \frac{C_0}{C_1}\right) a(v,v).$$

□

Now, we have to introduce the projection $Q_j : V \to V_j$ by defining

$$(Q_j u, v) = (u,v) \quad \forall u \in V,\ v \in V_j. \tag{4.13}$$

Moreover, define $A_j : V_j \to V_j$ by

$$(A_j u, v) = a(u,v) \quad \forall u,\ v \in V_j, \tag{4.14}$$

and $A : V \to V$ by

$$(Au, v) = a(u,v) \quad \forall u,\ v \in V. \tag{4.15}$$

Note that P_j and Q_j are different, but we may describe P_j by Q_j and $a(\cdot,\cdot)$ as the following lemma shows.

Lemma 4.4. *[2] For all $u \in V$ and $j = 0, \dots, N-1$ we have*

$$A_j P_j u = Q_j A u.$$

Proof. For $v \in V_j$ and $u \in V$ we have

$$(A_j P_j u, v) \overset{(4.14)}{=} a(P_j u, v) \overset{(4.7)}{=} a(u,v) \overset{(4.15)}{=} (Au, v) \overset{(4.13)}{=} (Q_j A u, v),$$

and hence

$$A_j P_j u = Q_j A u.$$

□

Moreover we know that for all $u \in V$ and $v \in V_j$

$$a(A_j^{-1} Q_j A u, v) \overset{(4.14)}{=} (A_j [A_j^{-1} Q_j A u], v) = (Q_j A u, v) \overset{4.4}{=} a(P_j u, v).$$

Therefore

$$P_j = A_j^{-1} Q_j A,$$

and hence

$$P_{AS} = \sum_{j=0}^{N-1} A_j^{-1} Q_j A =: BA,$$

where

$$B = \sum_{j=0}^{N-1} A_j^{-1} Q_j.$$

Lemma 4.5. *(a) If there exists a constant $c_1 > 0$ such that for all $u \in V$ there exists a decomposition $u = \sum_{j=0}^{N-1} u_j$, $u_j \in V_j$ such that*

$$\sum_{j=0}^{N-1} a(u_j, u_j) \le \frac{1}{c_1} a(u,u),$$

then there holds:

$$\lambda_{\min}(P_{AS}) \ge c_1.$$

(b) If there exists a constant $c_2 > 0$ such that for all $u = \sum_{j=0}^{N-1} u_j$, $u_j \in V_j$ holds

$$a(u,u) \leq c_2 \sum_{j=0}^{N-1} a(u_j, u_j),$$

then it follows that

$$\lambda_{\max}(P_{AS}) \leq c_2.$$

Proof. (a) Using Cauchy-Schwarz, we get

$$\begin{aligned} a(u,u) &= a(u, \sum_{j=0}^{N-1} u_j) = \sum_{j=0}^{N-1} a(u, u_j) = \sum_{j=0}^{N-1} a(P_j u, u_j) \\ &\leq \left(\sum_{j=0}^{N-1} a(P_j u, P_j u) \right)^{\frac{1}{2}} \left(\sum_{j=0}^{N-1} a(u_j, u_j) \right)^{\frac{1}{2}}. \end{aligned}$$

Consequently, we have

$$\frac{\sum_{j=0}^{N-1} a(P_j u, P_j u)}{a(u,u)} \geq \frac{a(u,u)}{\sum_{j=0}^{N-1} a(u_j, u_j)} \geq c_1, \tag{4.16}$$

due to our assumption. On the other hand, since $P_{AS} = \sum_{j=0}^{N-1} P_j$, since $a(\cdot,\cdot)$ is symmetric and due to (4.7), we have

$$a(P_{AS}u, u) = \sum_{j=0}^{N-1} a(P_j u, u) = \sum_{j=0}^{N-1} a(u, P_j u) = \sum_{j=0}^{N-1} a(P_j u, P_j u). \tag{4.17}$$

Now, (4.16) and (4.17) give

$$\frac{a(P_{AS}u, u)}{a(u,u)} \geq c_1.$$

Then

$$\lambda_{\min}(P_{AS}) = \min_{\substack{u \in V \\ u \neq 0}} \frac{a(P_{AS}u, u)}{a(u,u)},$$

(cf. Rayleigh quotient), gives

$$\lambda_{\min}(P_{AS}) \geq c_1.$$

(b) Applying the assumption to $P_{AS}u = \sum_{j=0}^{N-1} P_j u$, where $P_j u \in V_j$, gives

$$\underbrace{\sum_{j=0}^{N-1} a(P_j u, P_j u)}_{= a(P_{AS}u, u)} \geq \frac{1}{c_2} a(P_{AS}u, P_{AS}u).$$

Hence

$$\lambda_{\max}(P_{AS}) = \sup_{\substack{u \in V \\ u \neq 0}} \frac{a(P_{AS}u, u)}{a(u,u)} \leq c_2.$$

□

Now, $\lambda_{\min}(P_{AS}) \geq c_1$ and $\lambda_{\max}(P_{AS}) \leq c_2$ implies that

$$c_1 a(u,u) \leq a(P_{AS}u, u) \leq c_2 a(u,u),$$

i.e. the assumption $(\star)$ of Theorem 4.3 is satisfied.

Implementation

For $a(\cdot,\cdot)$ given in (4.2), u given as in (4.4) and the basis functions as in (4.5), we get, since ϕ_i and ϕ_j have no common support for $|i-j| \geq 2$,

$$a(\phi_i,\phi_j) = a(i-j),$$

where

$$a(k) = \begin{cases} -\frac{1}{h} & , k=-1 \\ \frac{2}{h} & , k=0 \\ -\frac{1}{h} & , k=1 \\ 0 & , \text{otherwise} \end{cases} .$$

First of all we need a subroutine `smul11` which performs the multiplication of the matrix $A := (a_{ij}) = a(i-j)$ with a vector b.

```
      subroutine smul11(a,b,h,n1) ! n1=n-1
      integer n1,i,l
      double precision a(-1:1), b(1:n1), h(1:n1), s

      do 10 i=1,n1
        s = 0.0
        do 40 l=-1,1
          if ( (i+l) .ge. 1 ) .and. ( (i+l) .le. n1 ) then
            s = s + a(l)*b(i+l)
          end if
40      continue
        h(i) = s
10    continue
      end
```

Note that this subroutine considers the fact that $b(0) = b(n) = 0$. As result we get

$$h(i) = \sum_{j=1}^{N-1} a(\phi_i,\phi_j)b_j = \sum_{j=1}^{N-1} a(i-j)b_j = \sum_{l=-1}^{1} a(l)b_{i+l}.$$

Now the program `unifem` takes the following form:

```
      program unifem

      declarations
      reading of the parameters, calculation of the right-hand side
     +                           of the Galerkin matix
      cg algorithm
      end

      subroutine smul
      end
```

Remember that the subroutine `cg` was introduced in section 5.2.

Now we consider the implementation of the (ASM) method with overlapping intervalls. In the following `nov` denotes the number of overlapping intervals and `nsi` denotes the number of smaller subintervalls. Hence $\frac{n}{nsi}$ is the number of elements of each subinterval. In the variable `sxm` we will store the left end of the subinterval and in `sxm1` the left end of the subinterval for each subintervl H_i, $i = 0,\ldots,$ `nsi`-1.Now the main part of the program is:

```
5000 continue
     call smul11(a,x,r,n1)
     do 5020 i=0,n1-1
       r(i) = b(i) - r(i)
5020 continue
     do 5030 i=0,n1-1
       dx(i) = 0.0
5030 continue
     do 5040 m=0,nsi-1
       sxm = min( max( (m*n)/nsi - nov, 0 ), n )
       sxm1 = min( max( (m+1)*n/nsi + nov, 0 ), n)
       sxn = sxm1 - sxm - 1
       do 5050 i=0, sxn-1
         q(i) = x(i+sxm)
5050   continue
       call subsolve(a,z,q,sxn) ! z=A^{-1}q
       do 5060 i=0,sxn-1
         dx(i+sxm) = dx(i+sxm) + omega*z(i)
5060   continue
5040 continue
```

Now, we have to add the correction to the approximate solution, define a stopping criterion and if it is not satisfied, we have to go back to 5000.

4.3 The BPX Preconditioner

In section 4.1 we considered the problem (4.1) for nested spaces $V_0 \subset \ldots \subset V_{N-1}$ which is a multilevel method.

In this section we follow the seminal work by Bramble, Pasciak, Xu [1]. We decompose the space $\mathcal{M}$ as

$$\mathcal{M}_1 \subset \ldots \mathcal{M}_J = \mathcal{M}.$$

Suppose that A_k, P_k and Q_k are defined as in section 4.2 and consider the following subspaces of $\mathcal{M}$

$$O_k := \{(Q_k - Q_{k-1})\psi : \psi \in \mathcal{M}\}, \quad k = 1, \ldots, J, \quad Q_0 v := 0, \quad \forall v \in \mathcal{M}.$$

Lemma 4.6. *$\mathcal{M} = O_1 \oplus \ldots \oplus O_J$ and $O_i \perp O_k$ if $i \neq k$.*

Proof. For $\psi \in \mathcal{M}$ we have, since $\psi = Q_J \psi$,

$$\psi = Q_J\psi - \sum_{k=1}^{J-1} Q_k\psi + \sum_{k=1}^{J-1} Q_k\psi - \underbrace{Q_0\psi}_{=0} = \sum_{k=1}^{J} \underbrace{(Q_k - Q_{k-1})\psi}_{\in O_k}$$

and this yields $\mathcal{M} = O_1 + \ldots + O_J$.

Let $x \in O_k$ and $y \in O_l$ $(k < l)$. Then there exists an $\eta \in \mathcal{M}$ such that

$$y = (Q_l - Q_{l-1})\eta.$$

Thus we have

$$(x, y) = (x, (Q_l - Q_{l-1})\eta) = (x, (\underbrace{Q_k Q_l}_{=Q_k} - \underbrace{Q_k Q_{l-1}}_{=Q_k})\eta) = 0,$$

which gives

$$O_k \perp O_l, \quad k < l$$

which finishes the proof. □

Now, we consider

$$B := \sum_{k=1}^{J} \lambda_k^{-1}(Q_k - Q_{k-1})$$

where λ_k is the largest eigenvalue of A_k. By the definition of O_k and Lemma 4.6 it is easy to show, that B is symmetric and positive definite because every $u \in \mathcal{M}$ can be written as

$$u = \sum_{i=1}^{J}(Q_k - Q_{k-1})u.$$

Lemma 4.7.

$$a(BAv, v) = \sum_{k=1}^{J} \lambda_k^{-1} \|(Q_k - Q_{k-1})Av\|^2, \quad \forall v \in \mathcal{M}$$

where $|| \cdot ||^2 = (\cdot, \cdot)$.

Proof. For $v \in \mathcal{M}$ we can write $Av = \sum_{k=1}^{J}(Q_k - Q_{k-1})Av$. Thus

$$\begin{aligned} a(BAv, v) &= (\sum_{k=1}^{J} \lambda_k^{-1}(Q_k - Q_{k-1})Av, Av) \\ &= (\sum_{k=1}^{J} \lambda_k^{-1}(Q_k - Q_{k-1})Av, \sum_{l=1}^{J}(Q_l - Q_{l-1})Av) \\ &\overset{4.6}{=} \sum_{k=1}^{J} \lambda_k^{-1}((Q_k - Q_{k-1})Av, (Q_k - Q_{k-1})Av) \\ &= \sum_{k=1}^{J} \lambda_k^{-1}\|(Q_k - Q_{k-1})Av\|^2. \end{aligned}$$

□

Lemma 4.8.

$$a(BAv, v) \leq Ja(v, v), \quad \forall v \in \mathcal{M}.$$

Proof. First note that for all $v \in \mathcal{M}$

$$a(P_k v, v)^2 \overset{C.S.}{\leq} a(P_k v, P_k v)\, a(v, v) = a(P_k v, v)\, a(v, v)$$

and thus

$$a(P_k v, v) \leq a(v, v). \tag{4.18}$$

Hence

$$\begin{aligned} a(BAv, v) &\overset{4.7}{=} \sum_{k=1}^{J} \lambda_k^{-1}\|(Q_k - Q_{k-1})Av\|^2 \\ &= \sum_{k=1}^{J} \lambda_k^{-1}\|(I - Q_{k-1})Q_k Av\|^2 \\ &\leq \sum_{k=1}^{J} \lambda_k^{-1}|||I - Q_{k-1}|||^2 \|Q_k Av\|^2 \\ &\leq \sum_{k=1}^{J} \lambda_k^{-1}\|Q_k Av\|^2 \overset{4.4}{=} \sum_{k=1}^{J} \lambda_k^{-1}\|A_k P_k v\|^2 \\ &\overset{(4.14)}{=} \sum_{k=1}^{J} \lambda_k^{-1} a(P_k v, A_k P_k v) \leq \sum_{k=1}^{J} a(P_k v, P_k v) \\ &= \sum_{k=1}^{J} a(P_k v, v) \overset{(4.18)}{\leq} Ja(v, v). \end{aligned}$$

For this estimate we have used the fact that the operator norm of a projection is bounded by one and that for a projection Q_{k-1}, $I - Q_{k-1}$ is also a projection. Moreover we have used that

$$\lambda_k = \max_{u \in \mathcal{M}_k} \frac{a(u, A_k u)}{a(u,u)}.$$

Theorem 4.9. *Assume that there exists a $c_1 > 0$ such that*

$$\textbf{(A1)} \quad \forall k = 1, \ldots, J \quad \|(I - Q_{k-1})v\|^2 \le c_1 \lambda_k^{-1} A \langle v, v \rangle, \quad \forall v \in \mathcal{M}.$$

Then

$$c_1^{-1} J^{-1} a(v,v) \le a(BAv, v) \le J a(v,v).$$

Proof. We only have to prove the left inequality; the right inequality is Lemma 4.8.

$$\begin{aligned}
a(v,v) &= \sum_{k=1}^{J} a((Q_k - Q_{k-1})v, v) = \sum_{k=1}^{J} \left\langle (I - Q_{k-1})v, (Q_k - Q_{k-1})Av \right\rangle \\
&\overset{C.S.}{\le} \sum_{k=1}^{J} \|(I - Q_{k-1})v\| \cdot \|(Q_k - Q_{k-1})Av\| \\
&\overset{(A1)}{\le} c_1^{\frac{1}{2}} \sum_{k=1}^{J} \lambda_k^{-\frac{1}{2}} a(v,v)^{\frac{1}{2}} \cdot \|(Q_k - Q_{k-1})Av\| \\
&\overset{C.S.}{\le} c_1^{\frac{1}{2}} \left(\sum_{k=1}^{J} a(v,v) \right)^{\frac{1}{2}} \left(\sum_{k=1}^{J} \lambda_k^{-1} \cdot \|(Q_k - Q_{k-1})Av\|^2 \right)^{\frac{1}{2}} \\
&\overset{4.7}{=} (c_1 J)^{\frac{1}{2}} a(v,v)^{\frac{1}{2}} a(BAv, v)^{\frac{1}{2}},
\end{aligned}$$

and hence

$$a(v,v) \le c_1 J a(BAv, v).$$

Now, we construct a more effective preconditioner by modifying B.

$$\begin{aligned}
B &= \sum_{k=1}^{J} \lambda_k^{-1}(Q_k - Q_{k-1}) = \sum_{k=1}^{J} \lambda_k^{-1} Q_k - \sum_{k=1}^{J-1} \lambda_{k+1}^{-1} Q_k \\
&= \sum_{k=1}^{J-1} (\lambda_k^{-1} - \lambda_{k+1}^{-1}) Q_k - \lambda_J^{-1} I
\end{aligned}$$

Assume that there exists a $\sigma > 1$ such that $\lambda_{k+1} \ge \sigma \lambda_k$ for all $k = 1, \ldots, J-1$. Then it is easy to verify that

$$\hat{B} := \sum_{k=1}^{J} \lambda_k^{-1} Q_k$$

satisfies

$$(1 - \frac{1}{\sigma}) \langle \hat{B}u, u \rangle \le \langle Bu, u \rangle \le \langle \hat{B}u, u \rangle.$$

If we replace $\lambda_k^{-1} I$ in $\hat{B}$ by the smoother R_k we get the so-called BPX preconditioner

$$\mathcal{B} := \sum_{k=1}^{J} R_k Q_k$$

which was introduced in 1990 by Bramble, Pasciak and Xu [1]. In the following we will need the assumpion

(A2) There exist $c_1, c_2 > 0$ independent of J such that

$$\frac{c_2 \|u\|^2}{\lambda_k} \le \langle R_k u, u \rangle \le c_3 \langle A_k^{-1} u, u \rangle, \quad \forall u \in \mathcal{M}_k.$$

For example $R_k = \lambda_k^{-1} I$ satisfies (A2).

Corollary 4.10. *Let (A1),(A2) be satisfied. Then*

$$c_1^{-1}c_2 J^{-1} A\langle v,v\rangle \le a(\mathcal{B}Av,v) \le c_3 J a(v,v).$$

Proof.

$$\begin{aligned} a(\mathcal{B}Av,v) &= \sum_{k=1}^{J} \underbrace{a(R_kQ_kAv,v)}_{=(R_kQ_kAv,A_kP_kv)} = \sum_{k=1}^{J}(R_kA_kP_kv,A_kP_kv) \\ &\overset{(A2)}{\le} c_2\sum_{k=1}^{J}\langle P_kv,A_kP_kv\rangle = c_2\sum_{k=1}^{J} a(P_kv,v) \le c_2 J a(v,v) \end{aligned}$$

and

$$\begin{aligned} a(\mathcal{B}Av,v) =& \sum_{k=1}^{J} \underbrace{a(R_kQ_kAv,v)}_{=(R_kQ_kAv,Q_kAv)} \overset{(A2)}{\ge} c_2\sum_{k=1}^{J}\lambda_k^{-1}\|Q_kAv\|^2 \\ &= c_2\sum_{k=1}^{J}\lambda_k^{-1}\langle Q_kAv,Av\rangle \ge c_2\langle BAv,Av\rangle \\ &= c_2 a(BAv,v) \overset{4.9}{\ge} c_2c_1^{-1}J^{-1}A\langle v,v\rangle \end{aligned}.$$

4.3.1 Application

Consider the problem

Find u satisfying $-\triangle u = f$ in $\Omega \subseteq \mathbb{R}^2$ $u = 0$ auf $\partial\Omega$ (polygonal)

We consider a quasiuniform regular triangulation T_N (N is number of nodes) with meshsize h on Ω. Let $\mathcal{M}_h := \overset{\circ}{S}{}_h^1(\Omega)$ be the set of piecewise linear functions on triangles $\triangle_N \in T_N$. Then the Galerkin method is:

Find $u_h \in \mathcal{M}_h$ such that

$$a(u_h,v) := \int_\Omega \nabla u_h \nabla v = \int_\Omega fv, \qquad \forall v \in \mathcal{M}_h.$$

Let $\{\phi_l^h\}_{l=1}^N$ be a basis for $\mathcal{M}_h$. Then we define a smoother R_h by

$$R_hv := \sum_{l=1}^{N}\langle v,\phi_l^h\rangle \phi_l^h, \qquad \forall v \in \mathcal{M}_h.$$

Writing $u_h = \sum_{k=1}^{N} u_k\phi_k$, we get a linear system of the form

$$A_h\mathbf{u} = \mathbf{b}.$$

(A_h denotes the discretization of A in $\mathcal{M}_h$.) Let λ_h be the maximum eigenvalue of A_h. Then there exist $c_0, c_1 > 0$ independent of h such that

$$\frac{c_0}{h^2} \le \lambda_h \le \frac{c_1}{h^2}. \tag{4.19}$$

Without proof we state the following lemma.

Lemma 4.11. R_h *satisfies (A2).*

Proof. Let $v \in \mathcal{M}_h$ and $\mathbf{a} = \{\alpha_l\}_{l=1}^N$ where α_l is the value of v at the l^{th} node of the triangulation T_N. We define the matrix

$$G_h := \left\{ \left\langle \phi_h^l, \phi_h^m \right\rangle \right\}_{l,m=1}^N .$$

Note

$$\langle R_h v, v \rangle = \sum_{l=1}^N \left\langle v, \phi_h^l \right\rangle^2 = (G_h \mathbf{a}, G_h \mathbf{a})$$

where $(\cdot, \cdot)$ denotes the Euclidian inner product. Since the triangulation T_N is quasiuniform we have

$$h^2 (\mathbf{a}, \mathbf{a}) \approx \|u\|^2 = (G_h \mathbf{a}, \mathbf{a})$$

and this implies

$$h^2 \|v\|^2 \approx h^4 (\mathbf{a}, \mathbf{a}) \approx (G_h \mathbf{a}, G_h \mathbf{a}) = \langle R_h v, v \rangle \ .$$

Using (4.19) we get

$$c_0 \frac{\|v\|^2}{\lambda_h} \leq c_2 \langle R_h v, v \rangle \leq c_3 \frac{\|v\|^2}{\lambda_h}, \quad \forall v \in \mathcal{M}_h$$

and by definition of λ_h we get the statement of this lemma. □

4.4 Connection between ASM and BPX

Now we want to show the connection between the (ASM) and the BPX preconditioner. Let

$$V_h = V_H + V_{h,1} + \ldots + V_{h,N-1},$$

where $V_{h,j} = \text{span}\{\phi_j^h\}$, $V_H = \overset{\circ}{S}_{2h}$. We want to solve

$$-u'' = f \ \text{ on } (0,1)$$
$$u(0) = u(1) = 0.$$

Then the bilinear form $a(\cdot, \cdot)$ is given by

$$a(u, v) = \int_0^1 u'(x)\, v'(x)\, dx.$$

Now, let

$$P = \sum_{j=0}^{N-1} P_j,$$

where the projection $P_j \, j = 1, \ldots, N-1$ is given by

$$a(P_j v_h, w_{h,j}) = a(v_h, w_{h,j}) \quad \forall w_{h,j} \in V_{h,j},\, v_h \in V_h,$$

and P_0 is the projection onto V_H. Our aim is to prove for $P_{AS} = P = \sum_{j=0}^{N-1} P_j$ that $\kappa(P) = \mathit{const.}$, independently of h, i.e. that there exist constants c_1, $c_2 > 0$ independent of h such that

$$c_1 a(v, v) \leq a(Pv, v) \leq c_2 a(v, v) \quad \forall v \in V_h.$$

Lemma 4.12. *There exists a constant $C_1 > 0$ independent of h such that*

$$\lambda_{\max}(P) \leq C_1.$$

Proof. We have to show that

$$a(Pv_h, v_h) \leq C_1 a(v_h, v_h) \quad \forall v_h \in V_h. \tag{4.20}$$

Let $T := \sum_{j=1}^{N-1} P_j$, then $P = P_0 + T$. Hence

$$a(Pv_h, v_h) \leq a(P_0 v_h, v_h) + a(Tv_h, v_h).$$

For the first term we have

$$\begin{aligned} a(P_0 v_h, v_h) &\overset{C.S.}{\leq} a(P_0 v_h, P_0 v_h)^{\frac{1}{2}} a(v_h, v_h)^{\frac{1}{2}} \\ &\leq ca(v_h, v_h), \end{aligned}$$

cf. the proof of Lemma 4.8. Since $V_{h,i} = \operatorname{span}\{\phi_i^h\}$, we have for all $v_h \in V_h$

$$P_i v_h = \frac{a(v_h, \phi_i^h)}{a(\phi_i^h, \phi_i^h)} \phi_i^h,$$

since $a(P_i v_h, \phi_i^h) = a(v_h, \phi_i^h)$. Therefore

$$Tv_h = \sum_{i=1}^{N-1} \frac{a(v_h, \phi_i^h)}{a(\phi_i^h, \phi_i^h)} \phi_i^h.$$

We do the poof now for a uniform grid. We have to show

$$a(Tv_h, v_h) = \sum_{i=1}^{N-1} \frac{a(v_h, \phi_i^h)^2}{a(\phi_i^h, \phi_i^h)} \leq ca(v_h, v_h) \quad \forall v_h \in V_h.$$

On a uniform grid there holds

$$M_{ij} := a(\phi_i^h, \phi_j^h) = \begin{cases} \frac{2}{h} & , i = j \\ -\frac{1}{h} & , |i-j| = 1 \\ 0 & , \text{otherwise} \end{cases}.$$

Let $M := (M_{ij})$, i.e. M is the Galerkin matrix, for which $M = M^t$. Then, writing $v_h = \sum_{i=1}^{N-1} v_i \phi_i^h$, we get

$$a(Tv_h, v_h) = \sum_{i=1}^{N-1} \frac{a(v_h, \phi_i^h)^2}{a(\phi_i^h, \phi_i^h)} = \frac{h}{2} \sum_{i=1}^{N-1} \underbrace{a(v_h, \phi_i^h)^2}_{=((M\mathbf{v})_i)^2} = \frac{h}{2} \mathbf{v}^t M^t M \mathbf{v},$$

since $a(v_h, \phi_i^h) = \sum_{j=1}^{N-1} v_j a(\phi_j^h, \phi_i^h) = (\mathbf{v}^t M)_i$, where $\mathbf{v} = (v_1, \ldots, v_{N-1})$. Remember that for each symmetric matrix M there exists a unitary matrix U such that $M = U^t \Lambda U$, where $\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_{N-1})$ and λ_i are the eigenvalues of M. Moreover all the eigenvalues of M are positive, since M is positive definite. Hence

$$a(Tv_h, v_h) = \frac{h}{2} \mathbf{v}^t U^t \Lambda \underbrace{UU^t}_{=I} \Lambda U \mathbf{v}$$

$$
\begin{aligned}
&= \frac{h}{2}\mathbf{v}^t U^t \Lambda^2 U \mathbf{v} = \frac{h}{2}\sum_{i=1}^{N-1}(\lambda_i (U\mathbf{v})_i)^2 \\
&\leq \frac{h}{2}\lambda_{\max}(M)\sum_{i=1}^{N-1}\lambda_i (U\mathbf{v})_i^2 \\
&= \frac{h}{2}\lambda_{\max}(M)\underbrace{\mathbf{v}^t U^t \Lambda U \mathbf{v}}_{=\mathbf{v}^t M\mathbf{v}=a(v_h,v_h)} .
\end{aligned}
$$

Furthermore we have

$$\lambda_{\max}(M) = ||M||_2 \leq ||M||_\infty = \frac{4}{h},$$

and hence

$$a(Tv_h, v_h) \leq 2a(v_h, v_h)$$

which finishes the proof.

Now, let for $u, v \in V$

$$\cos(u,v) := \frac{a(u,v)}{||u||_a ||v||_a},$$

and for subspaces V_1, V_2 of V

$$\cos(V_1, V_2) := \sup_{u_1 \in V_1, u_2 \in V_2} \cos(u_1, u_2).$$

Let $\theta = \{\theta_{ij}\}$, where $\theta_{ij} = \cos(V_i, V_j)$. Then for $u = \sum u_i$, $u_i \in V_i$ we get

$$
\begin{aligned}
a(u,u) &= \sum_{i,j} a(u_i, u_j) \leq \sum_{i,j}\theta_{ij}||u_i||_a ||u_j||_a \\
&\leq ||\theta||_2 \sum_i a(u_i, u_i)
\end{aligned}
$$

and hence by Lemma 4.5

$$\lambda_{\max}(P) \leq ||\theta||_2.$$

Proposition 4.13. *Let $l < k$ then we have for $i = 1, \ldots, N_l$, $j = 1, \ldots, N_k$*

$$\cos(V_i^l, V_j^k) = c\gamma^{|k-l-1|d/2}, \quad \gamma \in (0,1).$$

Proof. Let $\Omega_i^l := \operatorname{supp} u_i^l$, $u_i^l \in V_i^l$, $u_j^k \in V_j^k$. Then

$$a(u_i^l, u_j^k) \leq a_{\Omega_j^k}(u_i^l, u_i^l)^{\frac{1}{2}} a(u_j^k, u_j^k)^{\frac{1}{2}},$$

where $a_{\Omega_j^k}(u,u) := \int\limits_{\Omega_j^k} \nabla u \, \nabla u$. Now choose an element $\tau^l \subset \Omega_i^l$ such that $|\nabla u_i^l| =$ const., then

$$a_{\Omega_j^k \cap \tau^l}(u_i^l, u_i^l) = \frac{\operatorname{meas}(\Omega_j^k \cap \tau^l)}{\operatorname{meas}(\tau^l)} a_{\tau^l}(u_i^l, u_i^l) \leq c\frac{h_k^d}{h_l^d} a_{\tau^l}(u_i^l, u_i^l),$$

where $\operatorname{meas}(\tau^l)$ denotes the measure of τ^l. Now summation over all $\tau^l \subset \Omega_i^l$ yields

$$a_{\Omega_j^k}(u_i^l, u_i^l) \leq c\gamma^{d(k-l-1)} a_{\Omega_i^l}(u_i^l, u_i^l)$$

and hence

$$\cos(V_i^l, V_j^k) = c\gamma^{|k-l-1|d/2}, \quad \gamma \in (0,1).$$

Note that for example for $h_l = 2^{-l}$, we have $\gamma = \frac{1}{2}$. □

Now, using the fact that only some θ_{ij} are nonzero, the Cauchy-Schwarz inequality yields that

$$||\theta||_2 \le ||\theta||_\infty \le c \quad \text{independet of } L.$$

In order to estimate the smallest eigenvalue we have to consider some more facts:

For Ω convex, Nitsche's trick yields for the H^1-projection $P_{V^l} : H^1(\Omega) \to V^l$ that

$$||P_{V^l}u - u||_{L^2(\Omega)} \le ch_l|u|_{H^1(\Omega)} \quad \forall u \in H^1(\Omega).$$

Proposition 4.14. *[8] Now let* $P_{V^0} := 0$, $u^l := (P_{V^l} - P_{V^{l-1}})u$. *Then we have*

$$a(u^l, u^k) = 0, k \neq l, \;\; \textit{and} \;\; u^l = (I - P_{V^{l-1}})u^l. \tag{4.21}$$

Proof. Remember that for $l < k$, we have $v^l \in V^l \subseteq V^{k-1} \subset V^k$. Hence

$$(I - P_{V^{l-1}})v^l = (I - P_{V^{l-1}})(P_{V^l} - P_{V^{l-1}})v = v^l - (P_{V^{l-1}} - P_{V^{l-1}})v = v^l.$$

Moreover

$$a(v^k, v^l) = a(P_{V^k}v, v^l) - a(P_{V^{k-1}}v, v^l) = a(v, v^l) - a(v, v^l) = 0,$$

according to the definition of the projection P_{V^k} since $v^l \in V^k$ which finishes the proof. □

Then by Nitsche's trick we get

$$||u^l||_{L^2(\Omega)} \le ch_{l-1}|u^l|_{H^1(\Omega)}. \tag{4.22}$$

Now let $u = P_{V^L}u = \sum_{l=1}^{L} u^l$, $u^l \in V^l$ (H^1-orthogonal partition), and $u^l = \sum_{i=1}^{N_l} u_i^l$, where $u_i^l = \Pi^l(\theta_i^l u^l) \in V_i^l$, with Π^l the interpolation operator from $C^0 \to V^l$ and $\theta_i^l = \phi_i^l$ the hat functions. Note that $\{\theta_i^l\}$ is a partition of unity. Hence we have for a partition $\{x_i\}$: $\theta_i^l u^l(x_i) = u^l(x_i)$ and $\theta_i^l u^l(x_j) = 0$, for $j \neq i$. Moreover $\Pi^l(\theta_i^l \phi_i^l(x)) = \phi_i^l(x)$. Then it is possible to show that

$$|u_i^l|^2_{H^1(\Omega_i^l)} \le c\left(|u^l|^2_{H^1(\Omega_i^l)} + \frac{1}{h_{l-1}^2}||u_i^l||^2_{L^2(\Omega_i^l)}\right),$$

and hence

$$\sum_i |u_i^l|^2_{H^1(\Omega_i^l)} \le c\big(|u^l|^2_{H^1(\Omega)} + \frac{1}{h_{l-1}^2}||u^l||^2_{L^2(\Omega)}\big), \tag{4.23}$$

since $\sum_i ||u_i^l||^2_{L^2(\Omega_i^l)} \le c||u^l||^2_{L^2(\Omega)}$.

Proof. (of $\sum_i ||u_i^l||^2_{L^2(\Omega_i^l)} \le c||u^l||^2_{L^2(\Omega)}$)

Let $u^l = \sum_i \tilde{u}_i^l \phi_i^l$. Then

$$\sum_i ||u_i^l||^2_{L^2(\Omega_i^l)} = \sum_i \left(\tilde{u}_i^l\right)^2 ||\phi_i^l||^2_{L^2(\Omega_i^l)}.$$

Note that

$$(\phi_i^l, \phi_j^l) = \begin{cases} \frac{2}{3}h \;, \; i = j \\ \frac{1}{6}h \;, \; |i - j| = 1 \\ \;\; 0 \;, \text{otherwise} \end{cases}.$$

Hence

$$\sum_i \left(\tilde{u}_i^l\right)^2 ||\phi_i^l||^2_{L^2(\Omega_i^l)} = \frac{2}{3}h \underbrace{\sum_i \left(\tilde{u}_i^l\right)^2}_{=\mathbf{u}^l\mathbf{u}^l},$$

where $\mathbf{u}^l = \left(\tilde{u}_i^l\right)_{i=1,\ldots,N_l}$. On the other hand

$$c||u^l||^2_{L^2(\Omega)} = c\mathbf{u}^{l^T} G\mathbf{u}^l,$$

where G is the mass matrix, i.e. it has $\frac{2}{3}h$ on the diagonal, $\frac{1}{6}h$ on the subdiagonals and zeros elsewhere. Thus it remains to show that

$$\frac{2}{3}h\mathbf{u}^{l^T}\mathbf{u}^l \leq c\mathbf{u}^{l^T} G\mathbf{u}^l.$$

We know that

$$\lambda_{\min}(G) \leq \frac{\mathbf{u}^{l^T} G\mathbf{u}^l}{\mathbf{u}^{l^T}\mathbf{u}^l}.$$

Moreover from the Gerschgorin theorem we get that

$$\lambda_i^l \in \{x : |x - \frac{2}{3}h| \leq \frac{1}{3}h\} \quad \forall i,$$

hence

$$\lambda_{\min}(G) \geq \frac{1}{3}h,$$

and thus alltogether

$$\underbrace{\frac{1}{3}h\mathbf{u}^{l^T}\mathbf{u}^l}_{=\frac{1}{2}\sum_i ||u_i^l||^2_{L^2(\Omega_i^l)}} \leq \mathbf{u}^{l^T} G\mathbf{u}^l = ||u^l||^2_{L^2(\Omega)},$$

which finishes the proof. □

Now from (4.22) and (4.23) we get

$$\sum_i |u_i^l|^2_{H^1(\Omega_i^l)} \leq c|u^l|^2_{H^1(\Omega)} = ca(u^l, u^l),$$

and further

$$\begin{aligned}
\sum_l \sum_i a(u_i^l, u_i^l) &= \sum_l \sum_i |u_i^l|^2_{H^1(\Omega_i^l)} \\
&\leq c\sum_l |u^l|^2_{H^1(\Omega)} = c\left|\sum_l u^l\right|^2_{H^1(\Omega)} \\
&= c|u|^2_{H^1(\Omega)} = ca(u,u),
\end{aligned}$$

due to (4.21). Then by the Lemma 4.5 we get

$$\lambda_{\min}(P) \geq c^{-1} \quad \text{independent of } L.$$

Note that we may embed a nonconvex Ω in a convex region and then we get the same result.

4.4.1 Implementation

Let

$$\mathcal{B}_J v = \sum_{k=1}^{J} \frac{1}{\lambda_k} \sum_{l} (v, \phi_l^k)\, \phi_l^k.$$

Remember that we are working with spaces $\mathcal{M}_k$, $n_k := \dim \mathcal{M}_k$, $\mathcal{M}_k \subset \mathcal{M}_{k+1}$. A basis for $\mathcal{M}_k$ is given by $\left\{\phi_l^k\right\}_{l=1}^{n_k}$. Let the so called coarsening matrix $C_k = \{c_{il}^k\}$ be defined by

$$\phi_i^k = \sum_{l=1}^{n_{k+1}} c_{il}^k \phi_l^{k+1}.$$

Then the restriction is given by

$$(v, \phi_i^k) = (v, \sum_{l=1}^{n_{k+1}} c_{il}^k \phi_l^{k+1}) = \sum_{l=1}^{n_{k+1}} c_{il}^k (v, \phi_l^{k+1}). \tag{4.24}$$

For the prolongation $v = \sum_{i=1}^{n_k} v_i^k \phi_i^k$ is given and we have to look for $v = \sum_{i=1}^{n_{k+1}} v_i^{k+1} \phi_i^{k+1}$. Hence

$$\begin{aligned} v = \sum_{i=1}^{n_k} v_i^k \phi_i^k &= \sum_{i=1}^{n_k} v_i^k \sum_{l=1}^{n_{k+1}} c_{il}^k \phi_l^{k+1} \\ &= \sum_{l=1}^{n_{k+1}} \left(\sum_{i=1}^{n_k} v_i^k c_{il}^k \right) \phi_l^{k+1}, \end{aligned}$$

and thus

$$v_l^{k+1} := \sum_{i=1}^{n_k} v_i^k c_{il}^k. \tag{4.25}$$

Then, if (v, ϕ_l^k) is given on level $k = J$ we may calculate (v, ϕ_l^k) on level $1 \le k \le J-1$ with aid of (4.24). Let $w_l^k = (v, \phi_l^k)$, then $w_i^k = \sum_{l=1}^{n_{k+1}} c_{il}^k w_l^{k+1}$, i.e. in matrix vector notation

$$w^k = C_k w^{k+1}.$$

Now, let $B_k v = \sum_{l=1}^{n_k} b_l^k \phi_l^k$. Then

$$B_1 v = \sum_{l=1}^{n_1} b_l^1 \phi_l^1 \stackrel{!}{=} \frac{1}{\lambda_1} \sum_{l=1}^{n_1} w_l^1 \phi_l^1,$$

which gives

$$b_l^1 = \frac{1}{\lambda_1} w_l^1.$$

Analogously, we get from (4.25)

$$\begin{aligned} B_{k+1} v := \frac{1}{\lambda_{k+1}} \sum_{l=1}^{n_{k+1}} (v, \phi_l^{k+1}) \phi_l^{k+1} + B_k v \\ = \frac{1}{\lambda_{k+1}} \sum_{l=1}^{n_{k+1}} w_l^{k+1} \phi_l^{k+1} + \sum_{i=1}^{n_k} b_i^k \phi_i^k \\ = \sum_{l=1}^{n_{k+1}} \left(\frac{1}{\lambda_{k+1}} w_l^{k+1} + \sum_{i=1}^{n_k} b_i^k c_{il}^k \right) \phi_l^{k+1}, \end{aligned}$$

which gives

$$b_l^{k+1} = \frac{1}{\lambda_{k+1}} w_l^{k+1} + \sum_{i=1}^{n_k} b_i^k c_{il}^k,$$

or, in matrix vector notation

$$b^{k+1} = \frac{1}{\lambda_{k+1}} w^{k+1} + C_k^t b^k.$$

Note that using hat functions ϕ_l^k is given by $\phi_l^k = \frac{1}{2}\phi_{2l-1}^{k+1} + \phi_{2l}^{k+1} + \frac{1}{2}\phi_{2l+1}^{k+1}$. Now the `subroutine BPX` is given by

```
subroutine BPX(z,r,mlev,xm,xn,lambdan)
integer i,k,l,i1
integer mlev, xm(0:mlev), xn(0:mlev)
double precision r(0:*), z(0:*), lambdan(0:*)

do i=1,mlev
  i1 = xm(i)
  do l=0,xn(i)-1
    r(i1) = 0.5*r(xm(i-1) + 2*l) + r(xm(i-1) + 2*l + 1)
           + 0.5*r(xm(i-1) + 2*l + 2)
    i1 = i1 + 1
  end do
end do
do i=mlev,0,-1
  i1 = xm(i)
  do k=0,xn(i)-1
    z(i1) = r(i1)/lambdan(i)
    i1 = i1+1
  end do
  if (i .ne. mlev) then
    i1 = xm(i+1)
    do l=0,xn(i+1)-1
      z(xm(i) + 2*l) = z(xm(i) + 2*l) + 0.5*z(i1)
      z(xm(i) + 2*l + 1) = z(xm(i) + 2*l + 1) + z(i1)
      z(xm(i) + 2*l + 2) = z(xm(i) + 2*l + 2) + 0.5*z(i1)
      i1 = i1+1
    end do
   endif
  end do
 end
```

Note that the coarsening matrix for this case is of the form

$$C = \frac{1}{2}\begin{pmatrix} 1\ 2\ 1 & & & & \\ & 1\ 2\ 1 & & & \\ & & \ddots & & \\ & & & 1\ 2\ 1 & \\ & & & & 1\ 2\ 1 \end{pmatrix}.$$

Remember that for the multilevel additive Schwarz method we had

$$P_{MAS} = \sum_{l=1}^{J} T_l v = \sum_{l=1}^{J} \sum_{i=1}^{N_l} \frac{a(v,\phi_i^l)}{a(\phi_i^l,\phi_i^l)} \phi_i^l,$$

and for the BPX preconditioner

$$\mathcal{B}_{BPX}v = \sum_{l=1}^{J} \frac{1}{\lambda_l} \sum_{i=1}^{N_l} (v, \phi_i^l)\phi_i^l.$$

If we write $P_{MAS}v = B_{MAS}Av$, then we have

$$B_{MAS}v = \sum_{l=1}^{J} \sum_{i=1}^{N_l} \frac{(v, \phi_i^l)}{a(\phi_i^l, \phi_i^l)} \phi_i^l.$$

Hence the analogy between the two preconditioners is

$$\frac{1}{\lambda_l} \sim \frac{1}{a(\phi_i^l, \phi_i^l)}.$$

Note that $a(\phi_i^l, \phi_i^l)$ is independent of i, since we are dealing with hat functions. On a quasi-uniform mesh ($c_1 \leq \frac{h_{\max}}{h_{\min}} \leq c_2$) B_{MAS} behaves asymptotically like $\mathcal{B}_{BPX}$, but the first one yields better constants, since it is not only multiplied by a global constant $\frac{1}{\lambda_l}$.

In order to use the `subroutine BPX` in our main program we have to add the following lines. (Note that we asuume that $x_l = x_i^l \phi_i^l$, where $x_i^l = x(xm(l) + i)$.)

```
integer xm(0:16), xn(0:16)
double precision x(0:2*65536)
integer n                          ! number of intervals
integer n1

n1 = n - 1                         ! number of unknowns
xm(0) = 0.0
xn(0) = n1
do i = 1, mlev                    ! mlev = no. of levels
  xn(i) = (xn(i-1) + 1)/2 - 1      ! degree of freedom
  xm(i) = xm(i-1) + xn(i-1)
end do
```

Note that the last equality sets `xm(i)` equal to the beginning of the vector on the finer level plus the length of the vector on the finer level.

For the coarsening matrix we had the equation

$$\phi_i^l = \sum_{k=1}^{n_{l+1}} c_{ik}^l \phi_k^{l+1},$$

and hence

$$a(\phi_i^l, \phi_j^l) = \sum_{k=1}^{n_{l+1}} \sum_{m=1}^{n_{l+1}} c_{ik}^l c_{jm}^l a(\phi_k^{l+1}, \phi_m^{l+1}).$$

Therefore the effort to calculate all the ϕ_i^l is n^2 on the finest level, then $9\left(\frac{n}{2}\right)^2$ on the next coarser one, and so on. Hence alltogether we have an effort of $9\frac{1}{3}n^2 = 3n^2$. But actually we only need the diagonal elements, i.e. the effort would be $n + \frac{n}{2} + \frac{n}{3} + \ldots = 2n$.

The difference in the implementation for $\mathcal{B}_{BPX}$ and B_{MAS} is that after the restriction we have to divide by λ_l in the first case and by the diagonal element in the second case. To do so we need an additional vector for the B_{MAS} preconditioner which is of the same form as `xm, xn`:

```
double precision am(0:2*65536)
```

After that we have to do the prolongation in both cases.

Remark 3 *For further reading on Schwarz methods, we refer to the book by Toselli and Widlund [5] for the finite element method. For the boundary element method see [4].*

5

The Conjugate Gradient Method

Let $A \in \mathbb{R}^{n\times n}$ be symmetric and positive definite and $b \in \mathbb{R}^n$. We define the convex function

$$f : \mathbb{R}^n \longrightarrow \mathbb{R}, \quad f(x) := \frac{1}{2}\langle Ax, x\rangle - \langle b, x\rangle .$$

Since f is convex every relative extremum must be an absolute minimum. Hence

$$\nabla f(x) = (Ax - b)^T \overset{!}{=} 0 \quad \Longleftrightarrow x = A^{-1}b,$$

yields the absolute minimum of f. Thus the search for the absolute minimum of f is equivalent to find the solution of $Ax = b$.

Definition 5.1. *Let $p_0, \ldots, p_{n-1} \in \mathbb{R}^n \setminus \{0\}$ form a basis for $\mathbb{R}^n$. $\{p_0, \ldots, p_{n-1}\}$ is called* **conjugated with respect to** A *if*

$$\langle Ap_i, p_j\rangle = 0, \qquad \forall i \neq j = 0, \ldots, n-1.$$

For a basis $\{p_0, \ldots, p_{n-1}\}$ of $\mathbb{R}^n$, conjugated with respect to A, we consider the following method:

$$\boxed{\begin{array}{l} x_0 \in \mathbb{R}^n \text{ starting vector (arbitrary)} \\ x_{k+1} := x_k - \alpha_k p_k, \quad k = 0, \ldots, n-1 \\ \text{with } \alpha_k \text{ such that} \\ f(x_k - \alpha_k p_k) \leq f(x_k - \alpha p_k), \quad \forall \alpha \in \mathbb{R}. \end{array}} \tag{5.1}$$

The condition for α_k means that we have to minimize $f(x_k + \alpha p_k)$ at α. This implies

$$\begin{aligned} \left.\frac{\partial f(x_k - \alpha p_k)}{\partial \alpha}\right|_{\alpha=\alpha_k} &= \left\langle \nabla f(x_k - \alpha_k p_k), -p_k \right\rangle \\ &= -\langle Ax_k - b, p_k\rangle + \alpha_k \langle Ap_k, p_k\rangle \overset{!}{=} 0 \end{aligned}$$

$$\Longleftrightarrow \alpha_k = \frac{\langle Ax_k - b, p_k\rangle}{\langle Ap_k, p_k\rangle}. \tag{5.2}$$

Theorem 5.2. *The method (5.1) yields* $x_n = A^{-1}b$.

Proof. For $j < k = 1, \ldots, n-1$ we calculate

$$\langle Ax_{k+1} - b, p_j\rangle = \langle Ax_k - b, p_j\rangle - \alpha_k \langle Ap_k, p_j\rangle = \langle Ax_k - b, p_j\rangle \tag{5.3}$$

and for $k = 0, \ldots, n-1$

$$\begin{aligned}\langle Ax_{k+1} - b, p_k \rangle &= \langle Ax_k - b, p_k \rangle - \alpha_k \langle Ap_k, p_k \rangle \\ &\overset{(5.2)}{=} \langle Ax_k - b, p_k \rangle - \tfrac{\langle Ax_k - b, p_k \rangle}{\langle Ap_k, p_k \rangle} \langle Ap_k, p_k \rangle = 0.\end{aligned}$$

Therefore we get for all $j = 0, \dots, n-1$,

$$\langle Ax_n - b, p_j \rangle \overset{(5.3)}{=} \langle Ax_{j+1} - b, p_j \rangle = 0$$

and this implies $Ax_n - b = 0$. □

Note, that application of (5.1) implies that we have to calculate a conjugated basis (maybe by the Gram-Schmidt orthogonalisation method). But using this method we get $x_m = A^{-1}b$ for $m < n$ in many cases or it is sufficient to calculate $x_m = A^{-1}b$ for $m < n$, since the desired accuracy is already obtained. Thus it is wasteful to compute $p_0, \dots, p_{n-1}$. Moreover, computing the basis by Gram-Schmidt or other methods may yield complex vectors. Therefore we use a cheaper method to calculate the p_i during computation.

Conjugate Gradient Algorithm
$x_0 \in \mathbb{R}^n$ starting vector (arbitrary)
$p_0 := Ax_0 - b$
$x_{k+1} := x_k - \alpha_k p_k \quad , \quad k = 0, \dots, n-1$
(α_k as in (5.2))
$p_{k+1} := Ax_{k+1} - b - \beta_k p_k$
with β_k such that
$\langle Ap_{k+1}, p_k \rangle = 0$ (5.4)

We stop, if $p_m = 0$ for a $m < n$ or if we have determined x_n.

The condition on β_k leads to

$$\begin{aligned}\langle Ap_{k+1}, p_k \rangle = \langle Ax_{k+1} - b, Ap_k \rangle - \beta_k \langle Ap_k, p_k \rangle &\overset{!}{=} 0 \\ \Longrightarrow \qquad \beta_k = \frac{\langle Ax_{k+1} - b, Ap_k \rangle}{\langle Ap_k, p_k \rangle}&. \end{aligned} \tag{5.5}$$

Theorem 5.3. *If the conjugate gradient algorithm stops after m steps, we have $x^m = A^{-1}b$.*

Proof. If the algorithm stops after m steps we have $p_i \neq 0 \quad \forall i = 0, \dots, m-1$. Defining $r_j := Ax_j - b \quad (j = 0, \dots, m)$ we get by the construction of p_{j+1}

$$p_{j+1} = r_{j+1} - \beta_j p_j, \tag{5.6}$$

and

$$r_{j+1} = Ax_{j+1} - b = Ax_j - b - \alpha_j Ap_j = r_j - \alpha_j Ap_j. \tag{5.7}$$

Multiplication of this equation by p_j yields

$$\langle r_{j+1}, p_j \rangle = \langle r_j, p_j \rangle - \alpha_j \langle Ap_j, p_j \rangle = 0. \tag{5.8}$$

Firstly, we show by induction that for all $k = 1, \dots, m-1$ we have

$$\langle r_k, r_j \rangle = 0 \text{ and } \langle Ap_k, p_j \rangle = 0, \qquad \forall j = 0, \dots, k-1. \tag{5.9}$$

$k = 1$: β_0 is constructed such that automatically $\langle Ap_1, p_0 \rangle = 0$. Moreover $\langle r_1, r_0 \rangle = 0$ by (5.8) (note $p_0 = r_0$).

$k \to k+1$:

$$\begin{aligned}\langle r_{k+1}, r_k \rangle &\overset{(5.6)}{=} \underbrace{\langle r_{k+1}, p_k \rangle}_{=0\ (5.8)} + \beta_{k-1} \langle r_{k+1}, p_{k-1} \rangle \\ &\overset{(5.7)}{=} \beta_{k-1} \underbrace{\langle r_k, p_{k-1} \rangle}_{=0\ (5.8)} - \beta_{k-1}\alpha_k \langle A p_k, p_{k-1} \rangle = 0,\end{aligned}$$

since due to the algorithm $\langle Ap_k, p_{k-1} \rangle = 0$. Furthermore

$$\langle r_{k+1}, r_0 \rangle \overset{(5.7)}{=} \langle r_k, r_0 \rangle - \alpha_k \langle Ap_k, p_0 \rangle = 0$$

due to the induction hypothesis. For all $0 < j < k$ we have

$$\langle r_{k+1}, r_j \rangle \overset{(5.7)}{=} \underbrace{\langle r_k, r_j \rangle}_{=0} - \alpha_k \langle Ap_k, r_j \rangle = 0,$$

due to the induction hypothesis. Moreover, we have $\langle Ap_{k+1}, p_k \rangle = 0$ by the construction of β_k and for all $j < k$ we get

$$\begin{aligned}\langle Ap_{k+1}, p_j \rangle &\overset{(5.6)}{=} \langle r_{k+1}, Ap_j \rangle - \beta_k \underbrace{\langle p_k, Ap_j \rangle}_{=0} \\ &\overset{(5.7)}{=} \left\langle r_{k+1}, \tfrac{r_j - r_{j+1}}{\alpha_j} \right\rangle = 0,\end{aligned}$$

which finishes the proof of (5.9).

Now, if $m = n$ we have $x_n = A^{-1}b$ by Theorem 5.2.

If $m = 0$ we have $Ax_0 - b = r_0 = p_0 = 0$ and hence $r_0 = 0$ implies that x_0 is the solution of $Ax = b$.

If $0 < m < n$ we get (with $p_m = 0$)

$$0 = \langle p_m, r_m \rangle \overset{(5.6)}{=} \langle r_m, r_m \rangle - \beta_{m-1} \underbrace{\langle p_{m-1}, r_m \rangle}_{=0\ (5.8)} = \langle r_m, r_m \rangle$$

and this implies $r_m = 0$ and thus $x_m = A^{-1}b$. □

5.1 The Preconditioned Conjugate Gradient Method

The preconditioner B (symmetric and positive definite) transforms the problem $Ax = b$ to

$$\hat{A}x = Bb =: \hat{b}, \quad \hat{A} := BA. \tag{5.10}$$

$\hat{A}$ is symmetric and positive definite with respect to the inner product $[\,\cdot\,,\,\cdot\,] := \langle B^{-1}\,\cdot\,,\,\cdot\,\rangle$. Applying the conjugate gradient method using this inner product to solve the problem (5.10), we get the following algorithm:

Preconditioned Conjugate Gradient Algorithm
$x_0 \in \mathbb{R}^n$ starting vector (arbitrary)
$p_0 := BAx_0 - Bb$
$x_{k+1} := x_k - \hat{\alpha}_k p_k \quad , \quad k = 0, \ldots, n-1$
$p_{k+1} := BAx_{k+1} - Bb - \hat{\beta}_k p_k$

For $\hat{\alpha}_k$ we obtain

$$\hat{\alpha}_k = \frac{[\hat{A}x_{k+1} - \hat{b}, p_k]}{[\hat{A}p_k, p_k]} = \frac{\langle B^{-1}(BAx_{k+1} - Bb), p_k \rangle}{\langle B^{-1}BAp_k, p_k \rangle} = \alpha_k \tag{5.11}$$

and for $\hat{\beta}_k$

$$\hat{\beta}_k = \frac{\left[\hat{A}x_{k+1} - \hat{b}, \hat{A}p_k\right]}{\left[\hat{A}p_k, p_k\right]} = \frac{\langle Ax_{k+1} - b, BAp_k\rangle}{\langle Ap_k, p_k\rangle}. \tag{5.12}$$

Now it may be shown that compared to a linear iteration method the error is decreasing faster for the preconditioned conjugate gradient method. More precisely, we have for the error $e_k := x_k - x$

$$||e_k|| \leq c \left(\frac{\kappa(BA)^{\frac{1}{2}} - 1}{\kappa(BA)^{\frac{1}{2}} + 1} \right)^k ||e_0||,$$

and

$$\frac{\kappa(BA)^{\frac{1}{2}} - 1}{\kappa(BA)^{\frac{1}{2}} + 1} = \frac{\kappa(BA) - 1}{\kappa(BA) + 2\kappa(BA)^{\frac{1}{2}} + 1} < \frac{\kappa(BA) - 1}{\kappa(BA) + 1}.$$

Now, we consider the problem

$$\boxed{\begin{array}{c} -\Delta u = f \text{ in } \Omega \subset \mathbb{R}^2 \\ u = 0 \text{ on } \partial\Omega. \end{array}} \tag{5.13}$$

The domain Ω is supposed to be a square and we take a uniform mesh of square elements R_i with sidelength h. Multiplying (5.13) by a test function $v \in C^0(\bar{\Omega}) \cap C^1(\Omega)$ which is piecewise bilinear and satisfies $v = 0$ on $\partial\Omega$ and then integrating by parts gives the weak formulation

$$\int_\Omega \nabla u \, \nabla v = \int_\Omega f \, v. \tag{5.14}$$

We are looking for a piecewise bilinear approximation u_N to the solution u of (5.13). Hence on each square R_i we have $u_N|_{R_i} = (a_i x_1 + b_i)(c_i x_2 + d_i)$. If u_N satisfies $u_N(x_k) = 0$ in all nodes x_k on $\partial\Omega$, then we get $u_N|_{\partial\Omega} = 0$. If we denote the basis functions by ϕ_i, $i = 1, \ldots, N$, where N is the number of inner nodes, we may write u_N as

$$u_N = \sum_{j=1}^N a_j \phi_j.$$

Then for $v \in \text{span}\{\phi_j\}_{j=1}^N$ (5.14) becomes

$$\sum_{j=1}^N a_j \int_\Omega \nabla\phi_j \, \nabla\phi_k = \int_\Omega f \, \phi_k, \quad k = 1, \ldots, N.$$

Hence we get a system of the form

$$A\mathbf{a} = \mathbf{b},$$

where $A = \left(\int_\Omega \nabla\phi_j \, \nabla\phi_k\right)_{j,k=1}^N$ is symmetric and sparse. Now, the question of convergence of u_N to u arises. Moreover, if the system is very large we would like to split it up into overlapping sections. Again we have to consider the convergence.

5.2 Implementation

We want to develop a FORTRAN program for the conjugate gradient method. The pseudo code for the algorithm is as follows

input:$x = 0$, $r =$RHS

output:x = approximated solution, r = residual, k = number of iteratations performed

$k = 0, x_o = 0, r_0 =$RHS

while ($r_k \neq 0$) (test of convergence)

$z_k = B_{MAS} r_k$ (call preconditioning routine)

$k = k + 1$

if $k = 1$

$\beta_1 = 0$ and $p_1 = z_0$

else

$\beta_k = (r_{k-1}, z_{k-1})/(r_{k-2}, z_{k-2})$

$p_k = z_{k-1} + \beta_k p_{k-1}$

endif

$\alpha_k = (r_{k-1}, z_{k-1}/(p_k, Ap_k)$

$x_k = x_{k-1} + \alpha_k p_k$

$r_k = r_{k-1} - \alpha_k A p_k$

end

$x = x_k$.

Now this pseudo code has to be translated into FORTRAN. First of all we need a subroutine `sprod` to calculate an inner product and a subroutine `zaxpy` to calculate $z = ax + y$ where z, a, y are n-dimensional vectors.

```
      subroutine sprod(a,b,ab,n)
      integer n,i
      double precision a(1:n), b(1:n), ab
      ab = 0.0
      do 10 i=1,n
        ab = ab+a(i)*b(i)
   10 continue
      end


      subroutine zaxpy(a,x,y,z,n)
      integer n,i
      double precision a, x(1:n), y(1:n), z(1:n)
      do 10 i=1,n
        z(i) = a*x(i) + y(i)
   10 continue
      end
```

Moreover we need a subroutine `mul` which performs a matrix vector multiplication for a tridiagonal matrix A:

```
      mul(A,p,q,n)
      integer nmax
      parameter (nmax = 65536)
      integer i
      double precision A(-1:1), p(1:nmax), q(1:nmax)
      do 10 i=1,n
        q(i) = p(i-1)*A(-1) + p(i)*A(0) + p(i+1)*A(1)
   10 continue
      end
```

Now, the main program may be written as

```
      program cg
```

```
      integer nmax
      parameter (nmax = 65536)
      double precision x(1:nmax), u(1:nmax), z(1:nmax), r(1:nmax)
      double precision q(1:nmax), b(1:nmax), p(1:nmax)
      double precision alpha, beta, rz, rz1, pq, rr
      integer i, n, k
      double precision A(-1:1) ! this is for the 1-D case
C     double precision A(-1:1,-1:1) ! this is for the 2-D case
      k = 0
      do 10 i=1,n
        x(i) = 0.0
        r(i) = b(i)
   10 continue
   20 continue
      call sprod(r,r,rr,n)
      if (sqrt(rr) .le. 10e-6) goto 1000 ! stopping criterion
      do 30 i=1,n
        z(i) = r(i)
   30 continue
      k = k+1
      if (k .eq. 1) then
        beta = 0.0
        do 40 i=1,n
          p(i) = z(i)
   40   continue
        call sprod(r,z,rz,n)
      else
        call sprod(r,z,rz,n)
        beta = rz/rz1
        call zaxpy(beta,p,z,p,n)
      endif
      cal mul(A,p,q,n) ! q=Ap
      call sprod(p,q,pq,n)
      alpha = rz/pq
      call zaxpy(alpha,p,x,x,n)
      call zaxpy(-alpha,q,r,r,n)
      rz1 = rz
      goto 20
 1000 continue
      end
```

Now it is left as an exercise for the reader to solve the following problem with the conjugate gradient method:

$$
\begin{aligned}
-u''(x) &= \pi^2 \sin(\pi x), \qquad x \in (-1,1) \\
u(-1) &= u(1) = 0.
\end{aligned}
$$

Note that the exact solution to this problem is $u(x) = \sin(\pi x)$.

References

1. J. H. Bramble, J. E. Pasciak, and J. Xu. Parallel multilevel preconditioners. *Mathematics of Computation*, 55(191):1–22, 1990.
2. J.H. Bramble. Multigrid methods, pitman research notes in math. series 294, 1993.
3. W. Hackbusch. *Multi-Grid Methods and Applications*. Springer Series in Computational Mathematics, Vol 4. Springer Berlin Heidelberg, 2013.
4. E.P. Stephan and T. Tran. *Optimal Preconditioners for Boundary Element Methods – Domain Decomposition and Multilevel Methods*. Springer Series in Computational Mathematics. Springer Berlin Heidelberg, to appear.
5. A. Toselli and O. Widlund. *Domain Decomposition Methods - Algorithms and Theory*. Springer Series in Computational Mathematics. Springer Berlin Heidelberg, 2006.
6. T. von Petersdorff and E. P. Stephan. On the convergence of the multigrid method for a hypersingular integral equation of the first kind. *Numerische Mathematik*, 57:379–391, 1990.
7. T. von Petersdorff and E. P. Stephan. Multigrid solvers and preconditioners for first kind integral equations. *Numerical Methods for Partial Differential Equations*, 8(5):443–450, 1992.
8. X. Zhang. Multilevel schwarz methods. *Numerische Mathematik*, 63(1):521–539, 1992.