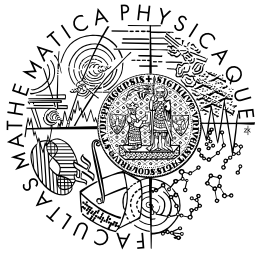


FACULTY
OF MATHEMATICS
AND PHYSICS
Charles University

BACHELOR THESIS

Petr Vacek

Multilevel methods and adaptivity

Department of Numerical Mathematics

Supervisor of the bachelor thesis:	prof. Ing. Zdeněk Strakoš, DrSc.
Study programme:	Mathematics
Study branch:	General Mathematics

Prague 2016

I declare that I carried out this bachelor thesis independently, and only with the cited sources, literature and other professional sources.

I understand that my work relates to the rights and obligations under the Act No. 121/2000 Sb., the Copyright Act, as amended, in particular the fact that the Charles University has the right to conclude a license agreement on the use of this work as a school work pursuant to Section 60 subsection 1 of the Copyright Act.

In ..... date ..... signature of the author

Title: Multilevel methods and adaptivity

Author: Petr Vacek

Department: Department of Numerical Mathematics

Supervisor: prof. Ing. Zdeněk Strakoš, DrSc., Department of Numerical Mathematics

Abstract: After introduction of the model problem we derive its weak formulation, show the existence and the uniqueness of the solution, and present the Galerkin finite element method. Then we briefly describe some of the stationary iterative methods and their smoothing property. We present the most common multigrid schemes, i.e. two-grid correction scheme, V-cycle scheme, and the full multigrid algorithm. Then we perform numerical experiment showing the differences between the use of the direct and iterative coarsest grid solver in V-cycle scheme and experiment considering a perturbation of the correction vector simulating a fault of a computational device.

Keywords: numerical PDE, finite element method, multilevel methods, multigrid

I would like to thank my supervisor prof. Ing. Zdeněk Strakoš, DrSc. for his many helpful comments and suggestions. I am grateful to Mgr. Jan Papež and RNDr. Jaroslav Hron, PhD. for helping me overcome many difficulties that occurred during the implementation of the experiments and writhing this thesis. Most of all, I would like to thank my family for their lasting support. Without their love, care, and encouragement this thesis would not have been written.

Contents

Introduction **2**

1 Model problem and its discretization **3**
1.1 Function spaces 3
1.2 Model problem 5
1.2.1 Physical experiments modeled by Poisson equation 5
1.2.2 Weak formulation 5
1.2.3 Existence and uniqueness of the solution 7
1.3 Galerkin method 9
1.3.1 Finite element method 11

2 Multigrid methods **14**
2.1 Stationary iterative methods 14
2.1.1 Derivation of stationary iterative methods 14
2.1.2 Convergence of stationary iterative methods 15
2.1.3 Smoothing property of stationary iterative methods 16
2.2 Multigrid shemes 21
2.2.1 Two-grid correction scheme 21
2.2.2 The V-cycle scheme 22
2.2.3 The μ-cycle scheme 23
2.2.4 The full multigrid algorithm 24

3 Numerical experiments **26**
3.1 Definition of problems 26
3.2 Experiment 1 – Discretization error 28
3.3 Experiment 2 – V-cycle scheme with iterative coarsest grid solver 29
3.4 Experiment 3 – V-cycle with fault coarsest grid correction 30

Conclusion **34**

Bibliography **35**

Introduction

Many real-world problems in various scientific areas are modeled by the partial differential equations (PDEs). Numerical solution of linear PDEs typically consists of two stages. After the discretization of the continuous problem, for example using the finite element method, the resulting system of linear algebraic equations is solved. A moderately sized linear algebraic system can be solved by direct methods; for large systems the (preconditioned) iterative methods become competitive and with increasing size they represent the only viable alternative. Moreover, iterative methods enable saving the computational work by stopping whenever the algebraic error drops to the level at which it does not significantly affect the whole error; see e.g., [16].

Multilevel methods and, more specifically, multigrid methods are well–known for being one of the fastest iterative methods for solving the systems of linear algebraic equations resulting from the discretization of PDEs. In the case of elliptic PDEs, they can reach an optimal or nearly-optimal efficiency, i.e. the computational work is asymptotically proportional to the number of unknowns; see e.g. [1]. Multigrid methods are based on reducing "high- frequency" error components and "low frequency" error components using different strategies. The "high- frequency" error components are eliminated by applying a stationary iterative method, whereas the "low frequency" error component vanish due to the coarse grid correction. The coarse grid correction consist of a direct solve of the system with the matrix associated to the coarsest grid. In the current literature on multigrid convergence, it is assumed that this solve is performed exactly.

The extremely large problems, which contain 10^{11} or more unknowns, are solved on high performance computing systems. With increasing size of the systems, the probability of failure of one or more system devices may not be non-negligible; see, e.g., [10]. Therefore fail-safe performance is becoming a major concern in numerical computing.

The goals of the thesis are to present an brief overview of the theoretical basis of the finite element discretization and multigrid methods, and to perform experiments considering an inexact solve at coarse grid correction and a perturbation of the correction vector simulating a fault of a computational device.

Thesis is organized as follows. In the first chapter we introduce the model problem, derive its weak formulation, show the existence and the uniqueness of the solution, and present the Galerkin finite element method. After briefly describing the stationary iterative methods and their smoothing property, we present the most common multigrid schemes in Chapter 2. Chapter 3 provides the results of numerical experiments. In Experiment 1 we focus on the behavior of the discretization error for higher-order finite element approximations. The differences between the use of the direct and iterative coarsest grid solver, are studied in Experiment 2. In Experiment 3 we simulate the fault leading to the situation where one or a small number of components of the coarse grid correction are corrupted. Thesis ends with concluding remarks.

1. Model problem and its discretization

In this chapter we first introduce the model problem, derive its *weak formulation*, and discuss existence and uniqueness of the solution. Then we present the *finite elements method*, which we use to transform the model problem to the system of linear algebraic equations.

The text of the chapter is based on [6, Chapters 1–4], [12, Chapters 2–3], and [15, Chapter 1].

1.1 Function spaces

In order to formulate the model problem and especially to derive the weak solution we will need the following function spaces.

For an open set $\Omega \subset \mathbb{R}^d$ $(d = 1, 2, 3)$ and $k \in \mathbb{N}$, $C^k(\Omega)$ denotes the space of all functions continuous in Ω for which any derivative up to the order k is continuous in Ω as well; i.e., $D^\alpha u \in C(\Omega)$ for any multiindex $\alpha = (\alpha_1, \ldots, \alpha_d),\ \alpha_i \in \mathbb{N} \cup \{0\}$, $|\alpha| := \sum_{i=1}^d \alpha_i = k$. The space $C^k(\overline{\Omega})$ consists of functions u belonging to $C^k(\Omega)$ such that for any multiindex α with $|\alpha| \leq k$ the function $D^\alpha u$ admits a continuous extension to $\overline{\Omega}$; $C^\infty(\Omega) := \cap_{k=1}^\infty C^k(\Omega)$ and similarly for $C^\infty(\overline{\Omega})$. Finally, $\mathscr{D}(\Omega)$ contains functions from $C^\infty(\Omega)$ with the compact support in Ω.

Lebesgue spaces

For $1 \leq p \leq \infty$, the *Lebesgue spaces* $L^p(\Omega)$ with the norms $\|\cdot\|_{L^p(\Omega)}$ are defined in the following manner. For $1 \leq p < \infty$,

$$L^p(\Omega) := \{u : \Omega \to \mathbb{R}; u \text{ is measurable and } \|u\|_{L^p(\Omega)} := \left(\int_\Omega |u|^p\right)^{1/p} < \infty\};$$

for $p = \infty$,

$$L^\infty(\Omega) := \{u : \Omega \to \mathbb{R}; u \text{ is measurable and } \|u\|_{L^\infty(\Omega)} < \infty\},$$

where $\|\cdot\|_{L^\infty(\Omega)}$ is the proper generalization of the maximum norm to measurable functions. The technical difference is that the values of a function on a set of measure zero don't affect the value of the $\|\cdot\|_{L^\infty(\Omega)}$ norm, i.e., (with $|\Upsilon|_d$ denoting the d-dimensional Lebesgue measure of $\Upsilon \subset \Omega$)

$$\|u\|_{L^\infty(\Omega)} := \inf_{\{\Upsilon \subset \Omega; |\Upsilon|_d = 0\}} \sup_{\{x \in \Omega \setminus \Upsilon\}} \{|u(x)| < \infty\},$$

see, e.g. [12, Chapter 2].

Sobolev spaces

The *Sobolev spaces* $W^{1,p}(\Omega)$ with norms $\|\cdot\|_{W^{1,p}}$, where $1 \leq p \leq \infty$, consist of those functions $u \in L^p(\Omega)$ for which all first *weak derivatives* belong to $L^p(\Omega)$;

i.e., for all $i = 1, \dots, d$ there are $f_i \in L^p(\Omega)$ such that

$$\int_\Omega u \frac{\partial \varphi}{\partial x_i} = -\int_\Omega f_i \varphi \quad \text{for all } \varphi \in \mathscr{D}(\Omega).$$

We usually write $\partial u/\partial x_i$ instead of f_i and ∇u instead of $(f_1, \dots, f_d)$.[a] In this formalism

$$W^{1,p}(\Omega) := \{u \in L^p(\Omega); \nabla u \in L^p(\Omega)^d\}$$

with the norm for $1 \le p < \infty$

$$\|u\|_{W^{1,p}} := \left(\|u\|^p_{L^p(\Omega)} + \|\nabla u\|^p_{L^p(\Omega)}\right)^{\frac{1}{p}} = \left(\|u\|^p_{L^p(\Omega)} + \left\| \left(\sum_{i=1}^d \left(\frac{\partial u}{\partial x_i} \right)^2 \right)^{\frac{1}{2}} \right\|^p_{L^p(\Omega)} \right)^{\frac{1}{p}}$$

and for $p = \infty$

$$\|u\|_{W^{1,\infty}} := \|u\|_{L^\infty(\Omega)} + \|\nabla u\|_{L^\infty(\Omega)}.$$

We denote $H^k(\Omega) := W^{k,2}(\Omega)$. The spaces $H^1(\Omega)$ and $L^2(\Omega)$ are the Hilbert spaces (see, e.g., [12, Section 2.1]) with the respective inner products

$$(u, v)_{L^2(\Omega)} := \int_\Omega uv, \qquad (u, v)_{H^1(\Omega)} := \int_\Omega (uv + \nabla u \cdot \nabla v)\,.$$

Therefore we can write

$$\|u\|_{L^2(\Omega)} = (u, u)^{1/2}_{L^2(\Omega)}, \qquad \|u\|_{H^1(\Omega)} = (u, u)^{1/2}_{H^1(\Omega)}.$$

Let $\Omega \subset \mathbb{R}^d$, $(d = 1, 2, 3)$ be open, bounded, connected set. We say that Ω is a set with *Lipschitz boundary*, and then write $\partial\Omega$ is Lipschitz, if there is $\ell \in \mathbb{N}$ and the numbers $\alpha_1 > 0$ and $\alpha_2 > 0$, such that, the boundary is described by ℓ mutually overlapping Lipschitz maps $\varrho_1, \dots, \varrho_\ell$, such that, for each map $\varrho \in \{\varrho_1, \dots, \varrho_\ell\}$, upon appropriately reorienting the coordinate axis, the sets

$$\{x \in \mathbb{R}^d; \quad \max_{i=1,\dots,d-1} |x_i| \le \alpha_1, \quad \varrho(x_1, \dots, x_{d-1}) < x_d \le \varrho(x_1, \dots, x_{d-1}) + \alpha_2\}$$

are subsets of Ω and the sets

$$\{x \in \mathbb{R}^d; \quad \max_{i=1,\dots,d-1} |x_i| \le \alpha_1, \quad \varrho(x_1, \dots, x_{d-1}) - \alpha_2 < x_d \le \varrho(x_1, \dots, x_{d-1})\}$$

are contained in $\mathbb{R}^d \setminus \overline{\Omega}$; see [12, Chapter 2].

If $\partial\Omega$ is Lipschitz, then there exists a linear bounded operator $\gamma : H^1(\Omega) \to L^2(\partial\Omega)$, (see, e.g., [12, Sections 2.1–2.2]) called the *trace operator*, that generalizes the concept of restriction of a $C(\overline{\Omega})$-function to the boundary to functions from $H^1(\Omega)$. The boundedness of γ implies that

$$\text{there exists } C_{tr} > 0 : \quad \|v\|_{L^2(\partial\Omega)} := \big\|\gamma(v)\big\|_{L^2(\partial\Omega)} \le C_{tr} \|v\|_{H^1(\Omega)} \ \ \forall v \in H^1(\Omega). \tag{1.1}$$

[a] We use the following notation in agreement with [12]. Vectors with components corresponding to the individual dimensions in $\mathbb{R}^d$ are *row vectors*. On the contrary, algebraic vectors associated with the discrete algebraic formulations of various problems using matrix representations are *column vectors*.

1.2 Model problem

Let Ω be a bounded *domain* in $\subset \mathbb{R}^d$, $(d = 1, 2, 3)$. *Domain* is a connected open set. Assuming that $\partial\Omega$ is Lipschitz and consist of two mutually disjoint parts $\partial\Omega_D$ and $\partial\Omega_N$ we consider the following problem: Given $f : \Omega \to \mathbb{R}$, $g : \partial\Omega_D \to \mathbb{R}$, $h : \partial\Omega_N \to \mathbb{R}$ and $\mathbb{K} = (K_{ij})_{i,j=1}^d : \Omega \to \mathbb{R}^{d\times d}$, find $u : \overline{\Omega} \to \mathbb{R}$ satisfying

$$-\nabla \cdot \big((\nabla u)\mathbb{K}\big) = f \quad \text{in} \quad \Omega, \tag{1.2}$$

$$u = g \quad \text{on} \quad \partial\Omega_D, \tag{1.3}$$

$$\frac{\partial u}{\partial \eta} = h \quad \text{on} \quad \partial\Omega_N. \tag{1.4}$$

Let n be an outer normal vector defined at (almost) all points of the boundary $\partial\Omega$, i.e., $n = (n_1, \ldots, n_d) : \partial\Omega \to \mathbb{R}^d$. The symbol $\partial u/\partial\eta$ stands for the derivative with respect to the co-normal vector $\eta : \Omega_N \to \mathbb{R}^d$, defined as $\eta := n\mathbb{K}^T$, which meas that

$$\frac{\partial u}{\partial \eta} = \nabla u \cdot \eta = (\nabla u)\eta^T = (\nabla u)\mathbb{K}n^T = (\nabla u)\mathbb{K} \cdot n, \tag{1.5}$$

where z^T denotes the transposition of the matrix (vector) z.

1.2.1 Physical experiments modeled by Poisson equation

For the tenzor $\mathbb{K}$ equal to the identity, i.e. $\mathbb{K} = I$, the equation (1.2) is called *the Poisson equation* and has several physical interpretations. In this text we will describe the modeling of the steady state temperature distribution. For another examples see [6, Section 1.1.1].

Steady-state temperature distribution

Let us consider a physical body in shape Ω. Function f stands for the heat sources and/or sinks in the body Ω. If $f(x) > 0$ for some $x \in \Omega$, then heat energy is being added at that point at a rate $f(x)$ (in appropriate units). If $f(x) < 0$, the energy is being removed at x. Solution u describes the steady-state temperature distribution in Ω.

In this context, Dirichlet boundary condition (1.3) indicates that the temperature of the physical body is held fixed at the boundary, specifically, that the temperature at $x \in \partial\Omega$ is held fixed at $g(x)$.

Neumann boundary condition (1.4) indicates that the *heat flux* across the boundary is of prescribed value h. The heat flux is the flow of the heat energy, in units of energy per time per length.

1.2.2 Weak formulation

To derive the *weak formulation* of our model problem (1.2)–(1.4) and to prove the existence and the uniqueness of its solution we will need following assumptions:

I) $f \in L^2(\Omega)$

II) $\mathbb{K} = (K_{ij})_{i,j=1}^d$ is a symmetric tensor, i.e., $K_{ij} = K_{ji}$, $i, j = 1, \ldots, d$

III) $\mathbb{K}$ is a uniformly positive tensor, i.e., there exist constant $c_{\mathbb{K}} > 0$ such that

$$c_{\mathbb{K}}\|z\|^2 \leq z^T\mathbb{K}(x)z, \quad \forall x \in \Omega, \forall z \in \mathbb{R}^d$$

IV) $K_{ij} \in L^\infty(\Omega)$ for all $i, j = 1, \ldots, d$

V) $g \in L^2(\partial\Omega_D)$ and $h \in L^2(\partial\Omega_N)$

VI) There exists function $u_D \in H^1(\Omega)$ such that $\gamma(u_D) = g$.

Now we multiply the equation (1.2) by an arbitrary

$$v \in \mathcal{H}^1_0 := \{v \in H^1(\Omega); \gamma(v) = 0 \quad \text{on } \partial\Omega_D\}$$

and integrate over Ω

$$-\int_\Omega \nabla \cdot \big((\nabla u)\mathbb{K}\big)\, v = \int_\Omega fv.$$

Using the Green's identity (see, e.g., [6, Section 2.1.2]), one arrives at

$$\int_\Omega (\nabla u)\mathbb{K} \cdot \nabla v - \int_{\partial\Omega} \big((\nabla u)\mathbb{K} \cdot n\big)\, v = \int_\Omega fv.$$

Considering the condition $\gamma(v) = 0$ on $\partial\Omega_D$ and using the formula (1.5), we conclude that

$$\int_\Omega (\nabla u)\mathbb{K} \cdot \nabla v = \int_{\partial\Omega_N} hv + \int_\Omega fv \quad \text{for all } v \in \mathcal{H}^1_0. \tag{1.6}$$

After defining the bilinear form

$$a(u, v) := \big((\nabla u)\mathbb{K}, \nabla v\big)_\Omega := \int_\Omega (\nabla u)\mathbb{K} \cdot \nabla v,$$

and the linear functional

$$\ell(v) := (h, v)_{\partial\Omega_N} + (f, v)_\Omega := \int_{\partial\Omega_N} hv + \int_\Omega fv$$

the equation (1.6) can be restated in compact form

$$a(u, v) = \ell(v) \quad \text{for all } v \in \mathcal{H}^1_0.$$

If we denote

$$\mathcal{H}^1_D := \{v \in H^1(\Omega); \gamma(v) = g \quad \text{on } \partial\Omega_D\} = \mathcal{H}^1_0 + u_D,$$

we can write down the *weak formulation* of the model problem (1.2)–(1.4):

$$\text{Find } u \in \mathcal{H}^1_D : \qquad a(u, v) = \ell(v) \quad \text{for all } v \in \mathcal{H}^1_0. \tag{1.7}$$

Function u satisfying (1.7) is called the *weak solution* of the problem (1.2)–(1.4).

1.2.3 Existence and uniqueness of the solution

Before trying to solve any mathematical equation, it is essential to ask whether its solution exists and, if so, whether the solution is unique. Our aim in this section is to summarize the well-known results for the weak formulation (1.7).

First we consider the case where the Dirichlet boundary condition (1.3) is prescribed on non-trivial part of the boundary $\partial\Omega$, i.e. $\int_{\partial\Omega_D} \neq 0$. The weak formulation (1.7) is equivalent to

$$\text{Find } w \in \mathcal{H}^1_0 : \qquad a(w,v) = \hat{\ell}(v) := \ell(v) - a(u_D, v) \quad \text{for all } v \in \mathcal{H}^1_0, \tag{1.8}$$

with $u = w + u_D$. To prove the existence and uniqueness of the solution w of (1.8) (respectively u of (1.7)) we will use Lax–Milgram theorem. The proof of this theorem is based on the Riesz representation theorem and can be found, e.g., in [5, Section 6.2.1.].

Theorem 1.1 (Lax–Milgram theorem)**.** *Suppose that V is a Hilbert space and $a(\cdot,\cdot)$ is a bilinear form on V that is bounded and V-elliptic, i.e., there exist $\alpha > 0$ and $\beta > 0$ such that*

$$\begin{aligned} \big|a(u,v)\big| &\le \beta \|u\|_V \|v\|_V \quad \textit{for all } u, v \in V, \\ a(v,v) &\ge \alpha \|v\|_V^2 \quad \textit{for all } v \in V. \end{aligned}$$

Then, given an $\ell \in V^*$, where V^* denotes the dual space of V, there exists a unique $w \in V$ such that

$$a(w,v) = \ell(v) \quad \textit{for all } v \in V.$$

Moreover, w depends continuously on ℓ;

$$\|w\|_V \le \frac{1}{\alpha} \|\ell\|_{V^*}. \tag{1.9}$$

In our case we have $V := \mathcal{H}^1_0$. The boundedness of $a(\cdot,\cdot)$ results from the following estimate:

$$\begin{aligned} |a(u,v)| &= \left| \int_\Omega (\nabla u)\mathbb{K} \cdot \nabla v \right| \\ &\le \int_\Omega |(\nabla u)\mathbb{K} \cdot \nabla v| \\ &\le \max_{i,j=1,\dots,d} \big\|K_{i,j}\big\|_{L^\infty(\Omega)} \|\nabla u\|_{L^2(\Omega)} \|\nabla v\|_{L^2(\Omega)} \quad \text{(by the Hölder inequality)} \\ &\le \max_{i,j=1,\dots,d} \big\|K_{i,j}\big\|_{L^\infty(\Omega)} \|u\|_{H^1(\Omega)} \|v\|_{H^1(\Omega)} \qquad \text{for all } u, v \in \mathcal{H}^1_0. \end{aligned} \tag{1.10}$$

To prove that $a(\cdot,\cdot)$ is $\mathcal{H}^1_0$-elliptic we need the *Poincaré inequality* (see, e.g., [6, p. 46]): Let Ω be a domain with Lipschitz boundary, then there exists a positive constant C_Ω, depending only on Ω, such that

$$\int_\Omega \nabla v \cdot \nabla v \ge C_\Omega \|v\|_{H^1(\Omega)} \quad \text{for all } v \in \mathcal{H}^1_0(\Omega). \tag{1.11}$$

Now let $v \in \mathcal{H}_0^1$, then

$$\begin{aligned} a(v,v) &= \int_\Omega (\nabla v)\mathbb{K}\cdot\nabla v \\ &\geq c_{\mathbb{K}} \int_\Omega \nabla v\cdot\nabla v \quad \text{(we used that } \mathbb{K} \text{ is uniformly positive)} \\ &\geq c_{\mathbb{K}} C_\Omega \|v\|^2_{H^1(\Omega)} \quad \text{(by the Poicaré inequality (1.11)).} \end{aligned} \tag{1.12}$$

Since $a(\cdot,\cdot)$ is a $\mathcal{H}_0^1$-elliptic symmetric bilinear form it is also an inner product on $\mathcal{H}_0^1$. This inner product is called the *energy inner product* and it can be shown that $\mathcal{H}_0^1$ equipted with $a(\cdot,\cdot)$is a Hilbert space; see, e.g., [6, p. 43]. Thus the energy inner product introduces the *energy norm*:

$$\|v\|_a := \sqrt{a(v,v)} \quad \text{for all } v \in \mathcal{H}_0^1,$$

which is topologically equivalent with the norm $\|\cdot\|_{H^1(\Omega)}$.

The fact that $\hat{\ell}$ is linear bounded functional, i.e., it belongs to the dual space of $\mathcal{H}_0^1(\Omega)$, follows from the inequality, for $v \in \mathcal{H}_0^1$,

$$\begin{aligned} |\hat{\ell}(v)| &= \left| -\int_\Omega (\nabla u_D)\mathbb{K}\cdot\nabla v + \int_{\partial\Omega_N} hv + \int_\Omega fv \right| \\ &\leq \int_\Omega \left|(\nabla u_D)\mathbb{K}\cdot\nabla v\right| + \int_{\partial\Omega_N} |hv| + \int_\Omega |fv| \\ &\leq \max_{i,j=1,\dots,d} \left\|K_{i,j}\right\|_{L^\infty(\Omega)} \|\nabla u_D\|_{L^2(\Omega)} \|\nabla v\|_{L^2(\Omega)} + \|h\|_{L^2(\partial\Omega_N)} \|v\|_{L^2(\partial\Omega_N)} \\ &\quad + \|f\|_{L^2(\Omega)} \|v\|_{L^2(\Omega)} \\ &\leq \left(\max_{i,j=1,\dots,d} \left\|K_{i,j}\right\|_{L^\infty(\Omega)} \|\nabla u_D\|_{L^2(\Omega)} + \|h\|_{L^2(\partial\Omega_N)} C_{tr} + \|f\|_{L^2(\Omega)} \right) \|v\|_{H^1(\Omega)} . \end{aligned}$$

We used the estimate (1.10), the Hölder inequality, and the trace inequality (1.1).

As the assumptions of the Lax-Milgram theorem are verified, it yields the existence and uniqueness of the solution w of (1.8) (respectively u of (1.7)). From (1.9),

$$\|w\|_{H^1(\Omega)} \leq \frac{1}{\alpha} \|\hat{\ell}\|_{(H^1(\Omega))^*}.$$

Note that the assumption of the Lax-Milgram theorem are fulfilled for any finite-dimensional subspace $\mathcal{V}_h \subset \mathcal{H}_0^1$, therefore the problem:

$$\text{Find } w_h \in \mathcal{V}_h : \qquad a(w_h, v_h) = \hat{\ell}(v_h) \quad \text{for all } v_h \in \mathcal{V}_h \tag{1.13}$$

has also the unique solution w_h. We will use this fact in the derivation of Galerkin method.

Previous results were derived for the weak formulation of our model problem (1.2)-(1.4) with Dirichlet or mixed boundary conditions. Let us now consider the model problem with pure Neumann boundary condition, i.e.,

$$\begin{aligned} -\nabla\cdot\big((\nabla u)\mathbb{K}\big) &= f \quad \text{in} \quad \Omega, \\ \frac{\partial u}{\partial \eta} &= h \quad \text{on} \quad \partial\Omega. \end{aligned} \tag{1.14}$$

Let u be the solution of (1.14), then using the divergence theorem (see, e.g., [6, Section 2.1.1]) we obtain the *compatibility condition*:

$$\int_\Omega f = -\int_\Omega \nabla\cdot\big((\nabla u)\mathbb{K}\big) = -\int_{\partial\Omega}(\nabla u)\mathbb{K}\cdot n = -\int_{\partial\Omega}\frac{\partial u}{\partial\eta} = -\int_{\partial\Omega} h.$$

We see that the problem (1.14) has a solution only if the compatibility condition holds.

It can be easily verified that if u satisfies (1.14), then for any constant $C\in\mathbb{R}$ $u+C$ also satisfies (1.14).

Removing this ambiguity e.g. by requiring $\int_\Omega u = 0$ we can again use the Lax-Milgram theorem to prove the existence and the uniqueness of the solution u; see, e.g., [6, p.47-48].

1.3 Galerkin method

In this section we present discretization of the model problem by the Galerkin method. The resulting problem can be equivalently formulated as a system of linear algebraic equations. In Section 1.3.1 the finite element method is introduced.

Let $\mathcal{V}_h$ be a finite dimensional subspace of $\mathcal{H}^1_0$. The *Galerkin approximate solution* of (1.8) is defined as the solution w_h of the problem

$$\text{Find } w_h\in\mathcal{V}_h: \qquad a(w_h,v_h)=\hat\ell(v_h) \quad \text{for all } v_h\in\mathcal{V}_h. \tag{1.15}$$

Let $N\in\mathbb{N}$ be the dimension of $\mathcal{V}_h$ and $\{\varphi_1,\dots,\varphi_N\}$ its basis, i.e., any $v_h\in\mathcal{V}_h$ can be written as $v_h=\sum_{i=1}^N\alpha_i\varphi_i$, where $\alpha_i\in\mathbb{R}$, $i=1,\dots,N$. From the linearity of $a(\cdot,\cdot)$ and $\hat{\ell(\cdot)}$ the problem (1.15) is equivalent to

$$\text{Find } w_h\in\mathcal{V}_h: \qquad a(w_h,\varphi_i)=\hat\ell(\varphi_i) \quad \text{for all } i=1,\dots,N. \tag{1.16}$$

Moreover considering $w_h=\sum_{j=1}^N\zeta_j\varphi_j$ for some coefficients $\zeta_j\in\mathbb{R}$, $j=1,\dots,N$ we have

$$\sum_{j=1}^N\zeta_j a(\varphi_j,\varphi_i)=\hat\ell(\varphi_i) \quad \text{for all } i=1,\dots,N.$$

Writing these equations in matrix-vector notation, we can reformulate (1.15) as

$$\text{Find } x_h\in\mathbb{R}^N: \qquad A_hx_h=b_h, \tag{1.17}$$

where

$$A_h=(a_{i,j})_{i,j=1}^N, \quad a_{i,j}=\int_\Omega(\nabla\varphi_j)\mathbb{K}\cdot\nabla\varphi_i$$

$$b_h=\begin{bmatrix}\xi_1\\ \vdots\\ \xi_N\end{bmatrix}, \quad \xi_i=\int_{\partial\Omega_N}h\varphi_i+\int_\Omega f\varphi_i-\int_\Omega(\nabla u_D)\mathbb{K}\cdot\nabla\varphi_i$$

$$x_h=\begin{bmatrix}\zeta_1\\ \vdots\\ \zeta_N\end{bmatrix}, \quad w_h=\sum_{i=1}^n\zeta_j\varphi_j.$$

The matrix A_h is called the *stiffness matrix* and it is symmetric and positive-definite. With no risk of confusion we will drop the index h where appropriate. The symmetry of A results from the symmetry of the bilinear form $a(\cdot,\cdot)$. To show positive-definiteness of A, let $y = (y_1 \ldots, y_N)^T \in \mathbb{R}^N$, $y \neq 0$, and consider a function $v = \sum_{i=1}^N y_i \varphi_i$. The following estimate

$$\begin{aligned} y^T A y &= \sum_{i=1}^N \sum_{j=1}^N y_i y_j a(\varphi_i, \varphi_j) = a\left(\sum_{i=1}^N y_i \varphi_i, \sum_{j=1}^N y_j \varphi_j\right) \\ &= a(v,v) = \|v\|_a^2 > 0, \end{aligned} \tag{1.18}$$

holds for every $y \neq 0$ and therefore A is positive-definite. The relation (1.18) also justifies the definition of the *algebraic energy norm* on $\mathbb{R}^N$

$$\|y\|_A^2 := y^T A y = a(v,v) = \|v\|_a^2,$$

for every $y \in \mathbb{R}^N$ and $v = \sum_{i=1}^N y_i \varphi_i$.

Now we will look on the relationship between the solution w of the weak formulation (1.8) and the solution w_h of (1.15). Subtracting (1.15) from (1.8) we get the *Galerkin orthogonality*

$$a(w - w_h, v_h) = 0 \quad \text{for all } v_h \in \mathcal{V}_h, \tag{1.19}$$

i.e., the *discretization error* $w - w_h$ is orthogonal to the subspace $\mathcal{V}_h$ with respect to the energy inner product. This also means that w_h is the best approximation (in the energy norm) to w in the space $\mathcal{V}_h$, i.e.,

$$\|w - w_h\|_a = \min_{v_h \in \mathcal{V}_h} \|w - v_h\|_a.$$

We will use the Galerkin orthogonality to derive the estimate on the discretization error $w - w_h$ in $H^1(\Omega)$-norm. Since $v_h - w_h \in \mathcal{V}_h$ the Galerkin orthogonality gives

$$a(w - w_h, v_h - w_h) = 0 \quad \text{for all } v_h \in \mathcal{V}_h.$$

Therefore,

$$\begin{aligned} a(w - w_h, w - w_h) &= a(w - w_h, w - v_h + v_h - w_h) \\ &= a(w - w_h, w - v_h) + a(w - w_h, v_h - w_h) \\ &= a(w - w_h, w - v_h) \end{aligned}$$

holds for every $v_h \in \mathcal{V}_h$. Using the fact that $a(\cdot,\cdot)$ is $\mathcal{H}_0^1$-elliptic and bounded with positive constants α, β (see Section 1.2.3.) we get:

$$\begin{aligned} \|w - w_h\|_{H^1(\Omega)}^2 &\leq \frac{1}{\alpha}\, a(w - w_h, w - w_h) \\ &= \frac{1}{\alpha}\, a(w - w_h, w - v_h) \\ &\leq \frac{\beta}{\alpha}\, \|w - w_h\|_{H^1(\Omega)} \|w - v_h\|_{H^1(\Omega)}, \end{aligned}$$

for every $v_h \in \mathcal{V}_h$. Dividing both sides by $\|w - w_h\|_{H^1(\Omega)}$ gives the result known as *Cea lemma* (see, e.g., [12, pp. 65–66]):

$$\|w - w_h\|_{H^1(\Omega)} \leq \frac{\beta}{\alpha}\, \|w - v_h\|_{H^1(\Omega)} \quad \text{for all } v_h \in \mathcal{V}_h. \tag{1.20}$$

1.3.1 Finite element method

In this subsection we introduce the *finite element method*, which is a Galerkin method with specific construction of the finite-dimensional subspace $\mathcal{V}_h \subset \mathcal{H}_0^1$. In the finite element method the Galerkin solution is searched in the subspace consisting of *piecewise polynomial* functions.

Piecewise polynomial functions defined on a triangular mesh

To the ease of presentation, suppose now that Ω is a polygonal domain in $\mathbb{R}^2$. The finite element method can be generalized to the case of non-polygonal domain in $\mathbb{R}^2$ or to three dimensional problems; see, e.g., [6, 2].

To define a piecewise polynomial over a domain Ω, the domain must be partitioned into subdomains. The collection of subdomains is referred to as a *mesh*. In this thesis we consider so called *conforming triangulation*, i.e. the partition $\mathcal{T}_h$ of domain Ω into triangles such that the intersection of any two triangles is a common vertex or a common edge. Figure 1.1a shows a conforming triangulation of a square. Another approaches using the partition of Ω into rectangles or quadrilaterals can be found, e.g., in [6, Section 4.5].

By h_K we denote the diameter of a triangle $K \in \mathcal{T}_h$. The *mesh size* h is defined as the maximum of these diameters, i.e.

$$h := \max_{K \in \mathcal{T}_h} h_K = \max_{K \in \mathcal{T}_h} \operatorname{diam}(K).$$

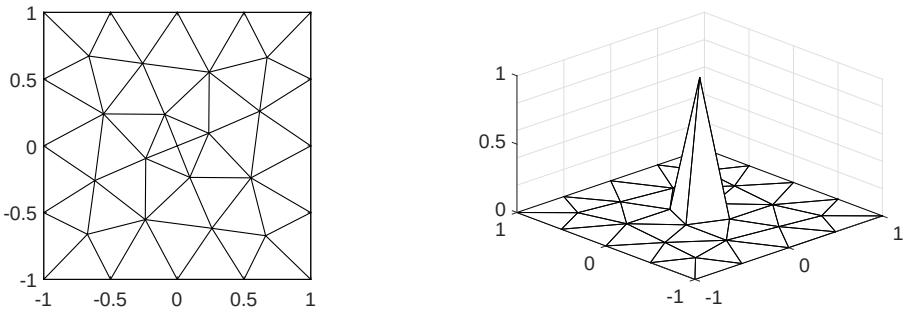
The space of piecewise polynomial functions of maximum degree $p \in \mathbb{N}$ is defined as

$$\mathcal{V}_{h,p} := \{v \in C(\overline{\Omega}); v|_K \in P^p(K) \quad \forall K \in \mathcal{T}_h \quad \text{and} \quad v = 0 \quad \text{on} \quad \partial\Omega\},$$

where $P^p(K)$ denotes the space of polynomial functions of maximum degree p on the triangle K. There holds $\mathcal{V}_{h,p} \subset \mathcal{H}_0^1$, the proof can be found, e.g., in [6, p. 71].

Basis functions of $\mathcal{V}_{h,1}$

As an example, we present a basis of the space $\mathcal{V}_{h,1}$ of piecewise linear functions. Discussion concerning the cases with higher polynomial degree can be found, e.g., in [6, Chapter 4].



(a) Conforming triangulation of a square (b) Basis function of $\mathcal{V}^{h,1}$

Figure 1.1: Example of a triangulation and a basis function.

Considering a triangulation $\mathcal{T}_h$ we denote by $z_1, \ldots, z_N$ the *free vertices* of the triangulation. Vertex z is said to be *free*, if $z \notin \partial\Omega_D$. Let $\varphi_i \in \mathcal{V}_{h,1}$ be a function corresponding to a free vertex z_i of the triangulation $\mathcal{T}_h$ such that

$$\varphi_i(z_j) = \begin{cases} 1, & i = j, \\ 0, & i \neq j. \end{cases} \tag{1.21}$$

Functions φ_i, $i = 1, \ldots, N$ are uniquely defined by the condition (1.21) and together form a basis of the space $\mathcal{V}_{h,1}$; see, e.g., [6, Section 4.1]. Figure 1.1b illustrates the basis function φ_i.

Recalling the definition of the stiffness matrix A,

$$A = (a_{i,j})_{i,j=1}^{N}, \quad a_{i,j} = \int_\Omega (\nabla\varphi_j)\mathbb{K} \cdot \nabla\varphi_i,$$

we conclude that thanks to the small support of basis functions φ_i, the entries $a_{i,j}$ are zero for the most of choices of i and j. To be precise $a_{i,j} \neq 0$ only if $i = j$ or if z_i and z_j are vertices of the same element K of the triangulation $\mathcal{T}_h$.

Convergence of the finite element method

Our aim in this paragraph is to state *a priori* estimates on the discretization error $u - u^h$ in H^1 and L^2 norms. The estimates depend on the mesh size h and the regularity of the exact solution u of (1.7). It moreover contains unspecified multiplication factors.

In the following theorem we assume that we are given a *nondegenerate family of triangulations* with mesh size h decreasing to zero. We say that a family $\{\mathcal{T}_h\}$ is *nondegenerate* if there exists a constant $\rho > 0$ such that

$$\frac{\operatorname{diam} B_K}{h_K} \geq \rho \quad \text{for all } K \in \mathcal{T}_h \text{ and all } \mathcal{T}_h \in \{\mathcal{T}_h\},$$

where B_K is the largest ball contained in K.

Theorem 1.2 ([6, Theorem 5.3.])**.** *Suppose $\{\mathcal{T}_h\}$ is a nondegenerate family of triangulation of a polygonal domain $\Omega \subset \mathbb{R}^2$ with constant $\rho > 0$, and suppose that $u \in H^{p+1}(\Omega)$. Then there exists a constant C depending on Ω and the value of ρ such that for any $\mathcal{T}_h \in \{\mathcal{T}_h\}$ the corresponding piecewise polynomial interpolant $u_I \in \mathcal{V}_{h,p}$ of u satisfies*

$$\|u - u_I\|_{H^1(\Omega)} \leq C\ h^p |u|_{H^{p+1}(\Omega)}$$

and

$$\|u - u_I\|_{L^2(\Omega)} \leq C\ h^{p+1} |u|_{H^{p+1}(\Omega)}$$

Here $|u|_{H^{p+1}(\Omega))}$ is the seminorm

$$|u|^2_{H^{p+1}(\Omega)} = \sum_{i+j=p+1} \int_\Omega \left| \frac{\partial^{p+1} u}{\partial x^i \partial y^j} \right|^2.$$

For the simplicity of exposition we further consider the problem (1.2)-(1.4) with the homogeneous Dirichlet boundary condition, i.e., with $g = 0$. Note that since $g = 0$ and consequently $u_D = 0$, the weak formulations (1.7) and (1.8) are identical. Let u be the solution of the weak formulation (1.7), the inequality (1.20) has in this case the form

$$\|u - u_h\|_{H^1(\Omega)} \leq \frac{\beta}{\alpha}\|u - v_h\|_{H^1(\Omega)} \quad \text{for all } v_h \in \mathcal{V}_{h,p}.$$

Supposing that all the assumptions of Theorem 1.2 are fulfilled, it yields

$$\|u - u_h\|_{H^1(\Omega)} \leq \frac{\beta}{\alpha} C\ h^p |u|_{H^{p+1}(\Omega)}. \tag{1.22}$$

Same results can be derived for the case of inhomogeneous Dirichlet boundary condition; see, e.g., [6, Section 5.3]. For the sake of completeness we present the estimate in the L^2-norm, which holds under the same assumptions as the previous estimate (1.22) see, e.g., [6, Section 5.4]

$$\|u - u_h\|_{L^2(\Omega)} \leq \frac{\beta}{\alpha} \widetilde{C}\ h^{p+1} |u|_{H^{p+1}(\Omega)}, \tag{1.23}$$

for some constant $\widetilde{C}$.

2. Multigrid methods

In this chapter we present multigrid methods for solving the systems of linear algebraic equations resulting from the discretization of model problem (1.2)–(1.4). For the simplicity of exposition we further consider $\mathbb{K} = \kappa I$, where $\kappa : \Omega \to \mathbb{R}$ is a real function such that the tensor $\mathbb{K}$ fulfills the assumptions II)-IV) formulated in Section 1.2.2. After we briefly describe some of the *stationary iterative methods* that are used in multigrid algorithms we present the multigrid concept and the most common schemes.

2.1 Stationary iterative methods

Let us start with a brief look at the classical stationary iterative methods, following [7, 11]; see also [3, 4, 6] and references therein.

2.1.1 Derivation of stationary iterative methods

Consider a system of linear algebraic equations $Ax = b$, where A is a regular matrix and b is a given right side vector. Let $x^{(0)}$ be an initial guess for the solution of $Ax = b$. If we could compute the *error* $e^{(0)} := A^{-1}b - x^{(0)}$, then we could correct our approximation and find the solution $x = x^{(0)} + e^{(0)}$. Unfortunately, the error $e^{(0)}$ is not available (without solving a linear system that is as difficult as the original problem), but we can compute the *residual* $r^{(0)} := b - Ax^{(0)}$.

As $Ae^{(0)} = A(A^{-1}b - x^{(0)}) = f - Ax^{(0)} = r^{(0)}$, the error $e^{(0)}$ satisfies the *residual equation* $Ae^{(0)} = r^{(0)}$. Suppose that M is a matrix such that $M^{-1}A$ approximates in some way the identity and that linear systems with matrix M are "easy" to solve. Multiplying the residual equation $Ae^{(0)} = r^{(0)}$ by M^{-1} from right give us $e^{(0)} \approx M^{-1}r^{(0)}$. Therefore, after solving "easy" equation for $z^{(0)}$: $Mz^{(0)} = r^{(0)}$, we can compute new hopefully improved approximation

$$x^{(1)} := x^{(0)} + z^{(0)} = x^{(0)} + M^{-1}r^{(0)}.$$

This process could be repeated, giving the *stationary iterative method*

$$x^{(k)} := x^{(k-1)} + z^{(k-1)} = x^{(k-1)} + M^{-1}r^{(k-1)}, \quad k = 1, 2, \ldots, \tag{2.1}$$

where $r^{(k-1)} := b - Ax^{(k-1)}$. An actual implementation of (2.1) might use the following algorithm formulated, e.g., in [7, Section 12.2.2].

Simple iteration

1. Given an initial guess $x^{(0)}$, compute $r^{(0)} = b - Ax^{(0)}$.
2. Solve $Mz^{(0)} = r^{(0)}$ for $z^{(0)}$.
3. For $k = 1, 2, \ldots$
   - Set $x^{(k)} = x^{(k-1)} + z^{(k-1)}$.

- Compute $r^{(k)} = b - Ax^{(k)}$.
- Solve $Mz^{(k)} = r^{(k)}$ for $z^{(k)}$.

The matrix M is called the *preconditioner*. Consider the decomposition of A in the form $A = D + L + U$, where D is the diagonal of A, L is the strict lower and U is the strict upper triangle of A , and let $\omega, \tilde{\omega} \in \mathbb{R}$. The previous algorithm goes by different names, according to the choice of M:

$$M = D \qquad \textit{Jacobi} \text{ method}, \tag{2.2}$$
$$M = \omega^{-1}D \qquad \textit{damped Jacobi} \text{ method}, \tag{2.3}$$
$$M = D + L \qquad \textit{Gauss–Seidel} \text{ method}, \tag{2.4}$$
$$M = \tilde{\omega}^{-1}D + L \qquad \textit{successive overrelaxation (SOR)} \text{ method}. \tag{2.5}$$

Another way how to derive these methods is to use the *matrix splitting*. If we write matrix A in the form $A = M - N$ then, supposing that M is invertible, we can rewrite $Ax = b$ as

$$Mx = Nx + b,$$

or equivalently

$$x = M^{-1}Nx + M^{-1}b.$$

Using the fixed point iteration, see, e.g., [7, Section 4.5], we obtain:

$$x^{(k)} = M^{-1}Nx^{(k-1)} + M^{-1}b. \tag{2.6}$$

To see that (2.6) is equivalent to (2.1), note that $M^{-1}N = I - M^{-1}A$ and substituting into (2.6),

$$\begin{aligned} x^{(k)} &= (I - M^{-1}A)x^{(k-1)} + M^{-1}b \\ &= x^{(k-1)} + M^{-1}(b - Ax^{(k-1)}) \\ &= x^{(k-1)} + z^{(k-1)}. \end{aligned} \tag{2.7}$$

We have discussed only few of the stationary iterative methods. More details and interesting numerical experiments can be found, e.g., in [3, Chapter 2].

2.1.2 Convergence of stationary iterative methods

Let $e^{(k)} := A^{-1}b - x^{(k)}$ denote the error in the approximation $x^{(k)}$. It follows from (2.7) that

$$\begin{aligned} e^{(k)} &= e^{(k-1)} - M^{-1}Ae^{(k-1)} \\ &= (I - M^{-1}A)e^{(k-1)} \\ &= (I - M^{-1}A)^k e^{(0)}. \end{aligned} \tag{2.8}$$

Taking norms on both side of (2.8) and using the Cauchy–Schwarz inequality, we find that

$$\|e^{(k)}\| \leq \ \|(I - M^{-1}A)^k\| \cdot \|e^{(0)}\|, \tag{2.9}$$

where $\|\cdot\|$ can be any vector norm and we take the matrix norm to be the one *induced* by the vector norm: $\|B\| := \max_{\|y\|=1}\|By\|$.

Apart from trivial cases, even in exact arithmetic these methods do not terminate with the exact solution of given problem. Therefore it is fully justified to analyze their convergence behavior using asymptotics and to study limits for $k \to \infty$. We see that if $\lim_{k\to\infty}(I - M^{-1}A)^k = 0$, then the error $e^{(k)}$ converges to zero. The opposite implication, i.e. if the error $e^{(k)}$ converges to zero then $\lim_{k\to\infty}(I - M^{-1}A)^k = 0$, also holds; see, e.g., [7, Theorem 12.2.1.]. We summarize these results in Theorem 2.1.

Theorem 2.1 ([7, Theorem 12.2.1.]). *The error in interation* (2.1) *converges to zero and* $x^{(k)}$ *converges to* $A^{-1}b$ *as* $k \to \infty$*, for every initial guess* $x^{(0)}$*, if and only if*

$$\lim_{k\to\infty}(I - M^{-1}A)^k = 0. \tag{2.10}$$

To specify when matrix $I - M^{-1}A$ satisfies the condition (2.10) we present the following theorem.

Theorem 2.2 ([7, Theorem 12.2.3.]). *Let* G *be a square matrix. Then*

$$\lim_{k\to\infty} G^k = 0,$$

if and only if $\rho(G)$*, the spectral radius of matrix* G *, i.e.,*

$$\rho(G) := \max\{|\lambda| : \lambda \textit{ is an eigenvalue of } G\},$$

satisfies $\rho(G) < 1$*.*

Using Theorem 2.2 we can restate Theorem 2.1 as

Theorem 2.3 ([7, Theorem 12.2.4.]). *The error in interation* (2.1) *converges to zero and* $x^{(k)}$ *converges to* $A^{-1}b$ *as* $k \to \infty$*, for every initial guess* $x^{(0)}$*, if and only if*

$$\rho(I - M^{-1}A) < 1.$$

While Theorem 2.3 gives a necessary and sufficient condition for convergence of the iterative method (2.1), it may not be easy to check. In general, the spectral radius of the iteration matrix $I - M^{-1}A$ is unknown and checking whether it is less than 1 can be complicated. There are some circumstances, however, in which this condition can be verified. For example, when applying the Gauss–Seidel method to system with a symmetric positive-define matrix A, then $\rho(I - M^{-1}A) < 1$ and consequently the method converges; see, e.g., [7, pp. 335-336].

Figure 2.1 illustrates the convergence of the Jacobi, Gauss–Seidel, and SOR iterative methods for a linear system arising from the finite element discretization of model problem (1.2)-(1.4). This behavior is typical for a class of matrices called 2-*cyclic* see, e.g., the discussion in [9, Chapter 5].

2.1.3 Smoothing property of stationary iterative methods

If we apply stationary iterative methods to the system of linear algebraic equations arising from the finite element discretization of the model problem (1.2)–(1.4), the "high frequency" error components are reduced rapidly and the "low

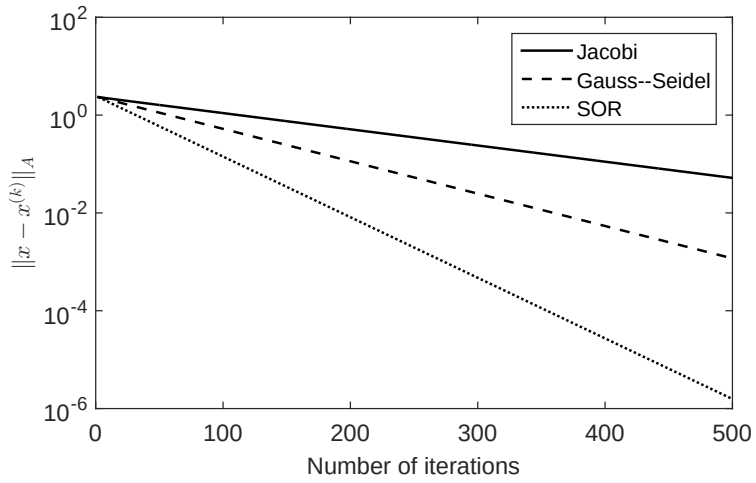


Figure 2.1: Convergence of the Jacobi, Gauss–Seidel, and SOR iterative methods for a system of linear algebraic equations resulting from the finite element discretization of the model problem (1.2)–(1.4).

frequency" error components start to dominate; see, e.g., [3, Chapter 2], [6, Section 13.1] and [9, Chapter 10]. This effect is known as the *smoothing property* of stationary iterative methods.

Following the exposition in [3, Chapter 2] we will explain the smoothing property of the damped Jacobi method (see Section 2.1.1) on a one-dimensional example. Consider the symmetric positive–define matrix $A \in \mathbb{R}^{N\times N}$, $N \in \mathbb{N}$ arisen from the discretization of one-dimensional Poisson equation, see [3], in the form

$$A = \begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{pmatrix}. \tag{2.11}$$

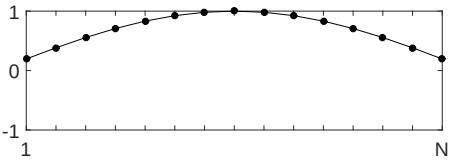
The matrix A has N eigenvalues, for $j = 1, \ldots, N$, they are given by (see [3, p. 18])

$$\lambda_j(A) = 4\sin^2\left(\frac{j\pi}{2(N+1)}\right). \tag{2.12}$$

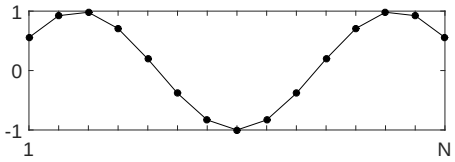
The N corresponding eigevectors are given by

$$v_j = \begin{pmatrix} \sin\left(\frac{j\pi}{N+1}\right) \\ \sin\left(\frac{2j\pi}{N+1}\right) \\ \vdots \\ \sin\left(\frac{Nj\pi}{N+1}\right) \end{pmatrix}. \tag{2.13}$$

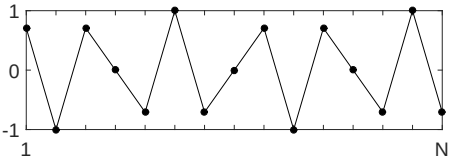
Eigenvectors v_j, $j = 1, \ldots, N$ are called (discrete) *Fourier modes*. We say that all v_j with $1 \le j < (N-1)/2$ are *low–frequency* or *smooth* modes and all v_j with $(N-1)/2 \le j \le N$ are *high–frequency* or *oscillatory* modes; see Figure 2.2,



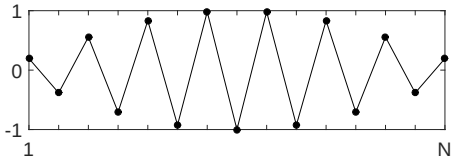
(a) eigenvector v_1



(b) eigenvector v_3



(c) eigenvector v_{12}



(d) eigenvector v_{15}

Figure 2.2: Graphs of the eigenvectors of A on a grid with 15 points.

where eigenvectors v_1, v_3, v_{12}, v_{15} are plotted. The error after k iterations of the damped Jacobi method, is given by (see Section 2.1.2)

$$e^{(k)} = (I - \omega D^{-1}A)^k e^{(0)}. \tag{2.14}$$

Note that for the particular matrix A given by (2.11), $D^{-1} = \frac{1}{2}I$, and

$$e^{(k)} = (I - \frac{\omega}{2}A)^k e^{(0)}. \tag{2.15}$$

Having known the eigenvalues of A we conclude that the eigenvalues of the matrix $G := I - \frac{\omega}{2}A$ are

$$\lambda_j(G) = 1 - 2\omega \sin^2\left(\frac{j\pi}{2(N+1)}\right), \quad 1 \leq j \leq N \tag{2.16}$$

and the eigenvectors of A and G are the same. Note that according to the Theorem 2.3 the damped Jacobi method applied to matrix A converges if and only if $\omega \in (0, 1]$. Since the eigenvectors of matrix A respectively matrix G form a base of $\mathbb{R}^N$, it is possible to represent $e^{(0)}$ in the form

$$e^{(0)} = \sum_{j=1}^{N} c_j v_j, \tag{2.17}$$

where the coefficients $c_j \in \mathbb{R}$ give the "amount" of each mode in the error. Substituting (2.17) into (2.15) yields

$$e^{(k)} = \sum_{j=1}^{N} c_j (I - \frac{\omega}{2}A)^k v_j = \sum_{j=1}^{N} c_j G^k v_j = \sum_{j=1}^{N} c_j \lambda_j^k(G) v_j. \tag{2.18}$$

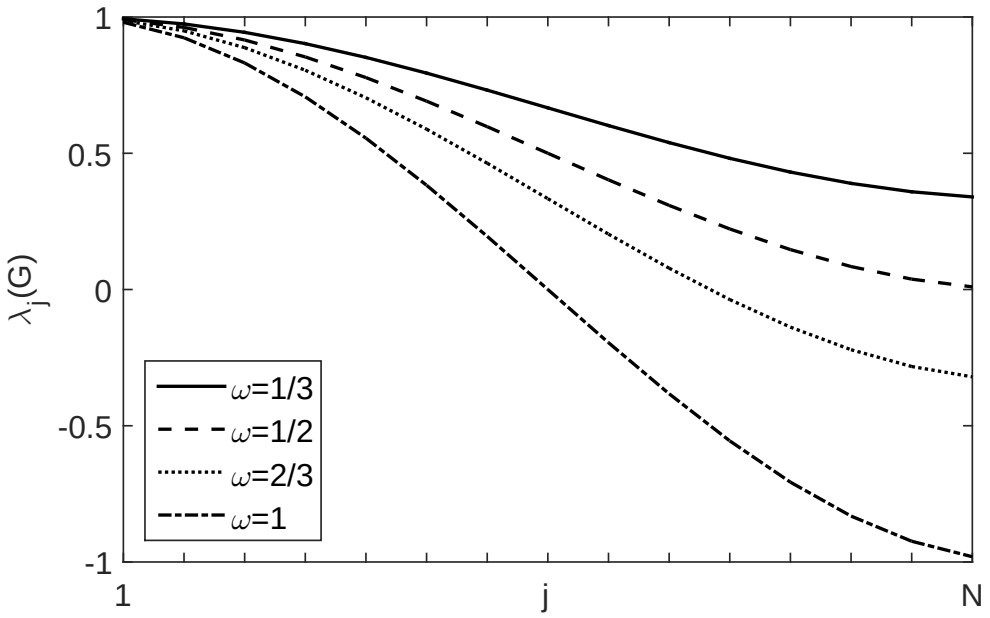


Figure 2.3: Eigenvalues of the matrix G for $\omega = \frac{1}{3}, \frac{1}{2}, \frac{2}{3}, 1$. The eigenvalues $\lambda_j(G)$ are plotted as if j were a continuous variable.

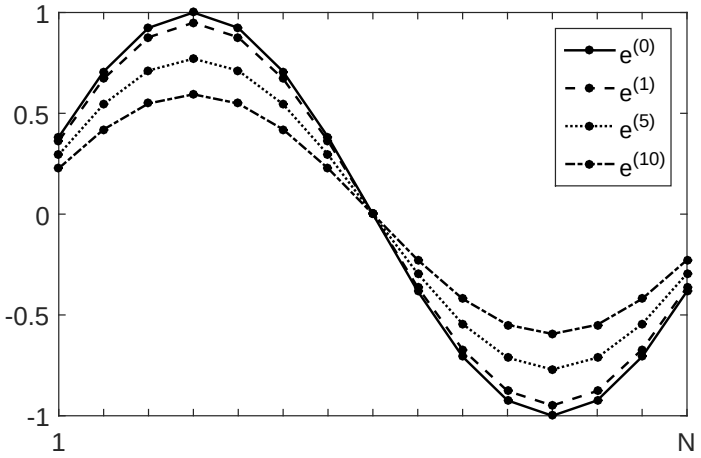
This expansion for $e^{(k)}$ shows that after k iterations, the jth mode of the initial error has been reduced by $\lambda_j^k(G)$. The eigenvalues $\lambda_j^k(G)$ depends on the choice of parameter ω; see Figure 2.3, where eigenvalues $\lambda_j^k(G)$ are plotted for four different values of ω. However for all values of ω satisfying $\omega \in (0, 1]$ the eigenvalue $\lambda_1(G)$, i.e. the value associated with the smoothest mode, is always close to one; see, e.g., [3, p. 21]. Therefore, no value of ω will reduce the smooth components of the error effectively. Optional value providing the the best damping of the oscillatory components is $2/3$; see, e.g., [3, p. 21]. Note that with $\omega = 2/3$, we have $|\lambda_j(G)| < 1/3$ for all $(N-1)/2 \leq j \leq N$, i.e. the oscillatory components are reduced at least by a factor of three with each iteration.

In Figure 2.4 the damped Jacobi method with $\omega = 2/3$ is applied to the problem with matrix A, $N = 15$. Figure 2.4a shows the smooth initial error and errors after one, five and ten iterations. We see that the smooth initial error is damped very slowly. Figure 2.4a shows a more oscillatory error and errors after one and two iterations. The damping is much more dramatic. Figure 2.4c illustrates the selectivity of the smoothing property. This experiment starts with initial error consisting of the smooth and oscillatory mode. After two iterations, the high-frequency components of the error are eliminated and the the low frequency error components start to dominate, i.e. the error is *smoothed*.

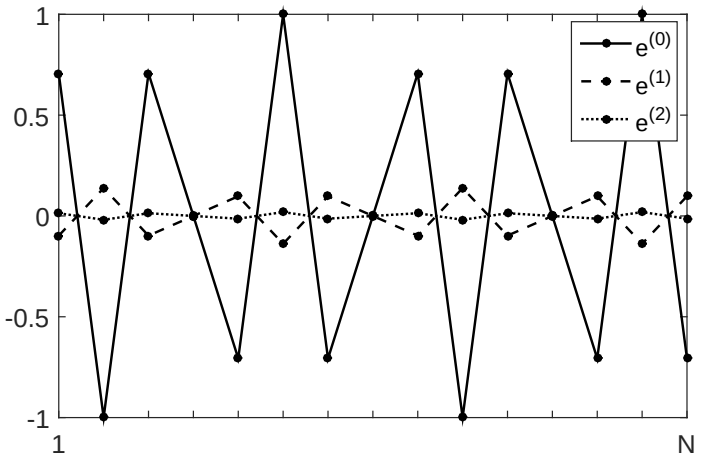
Using (2.18) the kth residual can be written as

$$r^{(k)} = Ae^{(k)} = A\sum_{j=1}^{N} c_j\lambda_j^k(G)v_j = \sum_{j=1}^{N} c_j\lambda_j^k(G)Av_j = \sum_{j=1}^{N} c_j\lambda_j^k(G)\lambda_j(A)v_j. \quad (2.19)$$

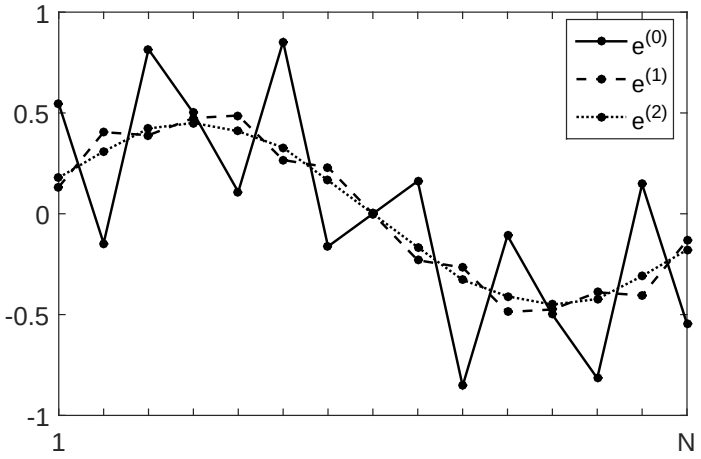
Since $|\lambda_j(A)| < 1$ for all $1 \leq j \leq N$, we conclude that after k iterations the residual is smoothed even more then the error.



(a)



(b)



(c)

Figure 2.4: Damped Jacobi method with $\omega = 2/3$ applied to a problem with matrix A, $N = 15$ with initial error consisting of (a) v_2, (b) v_{15}, and (c) $(v_2 + v_{12})/2$.

2.2 Multigrid shemes

In this section we follow [3, Chapter 3], [17],[7, Section 14.6] and present the most common multigrid schemes. First we will describe the concept of multigrid on the simplest scheme, i.e. the *two-grid correction* scheme. Then we will state its recursive modifications the *V-cycle* scheme, the μ*-cycle* scheme and the *full multigrid* algorithm.

2.2.1 Two-grid correction scheme

Consider two triangulations $\mathcal{T}_h$ and $\mathcal{T}_H$ of Ω, such that the corresponding subspaces $\mathcal{V}^{h,p}, \mathcal{V}^{H,p}$ satisfy $\mathcal{V}^{H,p} \subset \mathcal{V}^{h,p}$, i.e. $\mathcal{T}_h$ is a *refinement* of $\mathcal{T}_H$. We will call, in agreement with standard multigrid literature, $\mathcal{T}_h$ a *fine grid* and $\mathcal{T}_H$ a *coarse grid*, and denote $N_h := \dim(\mathcal{V}^{h,p})$, $N_H := \dim(\mathcal{V}^{H,p})$. We consider linear interpolation and restriction operators $\mathcal{I} : \mathcal{V}^{H,p} \to \mathcal{V}^{h,p}$, $\mathcal{R} : \mathcal{V}^{h,p} \to \mathcal{V}^{H,p}$ that have algebraic representation in the form of rectangular matrices $I \in \mathbb{R}^{N_h \times N_H}$, $R \in \mathbb{R}^{N_H \times N_h}$. Let $A_h x_h = b_h$ and $A_H x_H = b_H$ be systems of linear algebraic equations arising from the finite element discretization of the model problem (1.2)–(1.4) using the triangulation $\mathcal{T}_h$ and $\mathcal{T}_H$, respectively; see Section 1.3. The two-grid correction scheme solves the system $A_h x_h = b_h$ using a stationary iterative method and the *coarse-grid correction* as follows (see, e.g., [3, p. 37], [7, Section 14.6])

Two-grid correction scheme

1. (*pre-smoothing*)
   - Perform m_1 iterations of a stationary method applied to $A_h x_h = b_h$ with an initial guess y_h to obtain an approximation y_h.
2. (*coarse-grid correction*)
   - Compute the residual $r_h = b_h - A_h y_h$.
   - Restrict the residual to the coarse grid as $r_H = R r_h$.
   - Solve $A_H e_H = r_H$ for e_H.
   - Interpolate the error e_H to the fine grid as $e_h = I e_H$.
   - Correct the approximation y_h by $y_h := y_h + e_h$.
3. (*post-smoothing*)
   - Perform m_2 iterations of a stationary method applied to $A_h x_h = b_h$ with an initial guess y_h to obtain an approximation y_h.

The pre-smoothing has the effect of damping out the oscillatory components of the error; see Section 2.1.3. After a few steps, the error and the residual are smoothed and the additional iterations are not effective. Hence, we use the coarse-grid correction, which eliminates the low frequency error components. We compute the residual and restrict it to the coarse grid. Note that since the residual is a smooth vector, it can be well-represented on the coarse grid. This is illustrated in Figure 2.5. We see that the restriction keeps the essential behavior

of the smooth vector while an oscillating vector is misrepresented as a smooth. We may expect that the restriction to the coarse grid has the same property also in the d-dimensional case.

After solving the system $A_H e_H = r_H$ we interpolate the error as $e_h = I e_H$ and correct our approximation $y_h := y_h + e_h$. The post-smoothing smooths out the oscillations that may occur in e_h due to the interpolation.

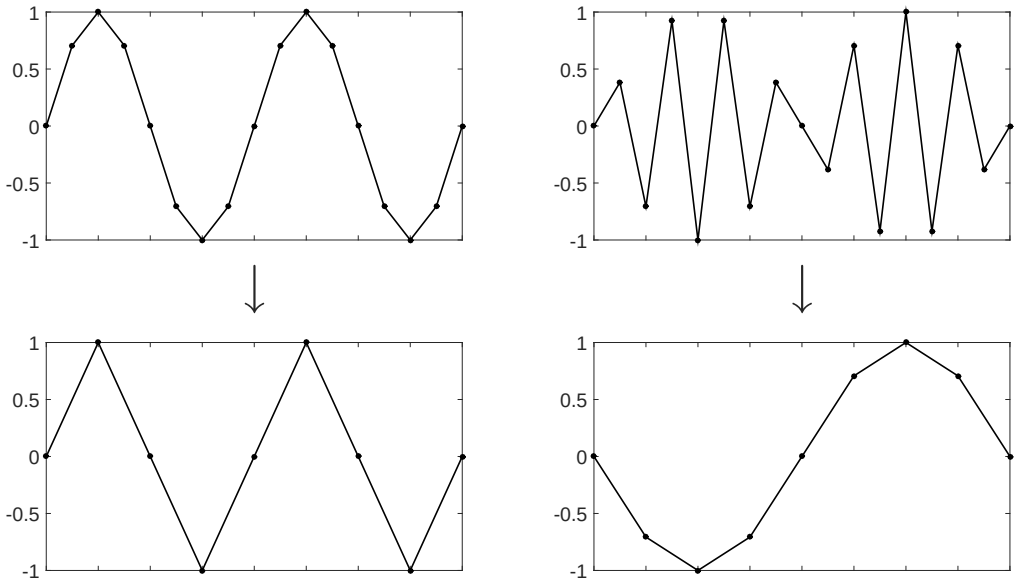


Figure 2.5: Restriction of a smooth vector (on the left) and the oscillatory vector (on the right) to a coarse grid.

2.2.2 The V-cycle scheme

The system $A_H e_H = r_H$ is smaller than the original one but it can be still too large to be solved efficiently by a direct method. However, the two-grid correction scheme can be applied recursively until the restricted system is small enough to be solved by a direct method. The resulting method is called the *V-cycle* scheme. The name arises from the pattern pictured in Figure 2.6a, which shows the method beginning on the finest grid, descending to the coarsest grid, and then returning back to the finest grid. To present the *V-cycle* scheme we will need the following notation.

Let us consider a sequence of triangulations of Ω (grids): $\mathcal{T}_1, \ldots, \mathcal{T}_J$, where $\mathcal{T}_j$ is a refinement of $\mathcal{T}_{j+1}$ for every $j = 1, \ldots, J-1$. Let A_j be the matrix arising from the finite element discretization of the model problem (1.2)–(1.4) using the triangulation $\mathcal{T}_j$ (grid j). Let matrix I_j^{j+1} represents the interpolation operator from the grid $j+1$ to the grid j and let matrix R_{j+1}^j represents the restriction operator from the grid j to the grid $j+1$.

The V-cycle scheme is defined as (see, e.g., [3, p. 40])

The V-cycle scheme

1. Perform m_1 iterations of a stationary method applied to $A_1 x_1 = b_1$ with an initial guess y_1 to obtain an approximation y_1.
2. Compute $b_2 = R_2^1(b_1 - A_1 y_1)$.
3. For $j = 2, \ldots, J-1$,
   - Perform m_1 iterations of a stationary method applied to $A_j x_j = b_j$ with an initial guess $y_j = 0$ to obtain an approximation y_j.
   - Compute $b_{j+1} = R_{j+1}^j(b_j - A_j y_j)$.
4. Solve $A_J y_J = b_J$ for y_J.
5. For $j = J-1, \ldots, 1$,
   - Correct the approximation y_j by $y_j := y_j + I_j^{j+1} y_{j+1}$.
   - Perform m_2 iterations of a stationary method applied to $A_j x_j = b_j$ with an initial guess y_j to obtain an approximation y_j.

The V-cycle scheme has also a compact recursive definition, which is given as follows; see, e.g., [3, p. 40], [17, p. 19].

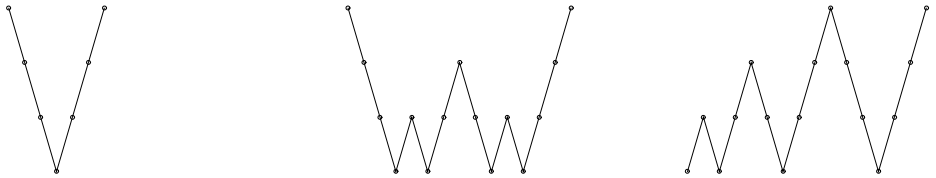
The V-cycle scheme $y_j = V(y_j, b_j, j)$

1. If $j = J$, solve $A_J y_J = b_J$ and go to step 8.
2. Perform m_1 iterations of a stationary method applied to $A_j x_j = b_j$ with an initial guess y_j to obtain an approximation y_j.
3. Compute $b_{j+1} = R_{j+1}^j(b_j - A_j y_j)$.
4. Set $y_{j+1} := 0$.
5. Call V-cycle scheme recursively $y_{j+1} := V(y_{j+1}, b_{j+1}, j+1)$.
6. Correct the approximation y_j by $y_j := y_j + I_j^{j+1} y_{j+1}$.
7. Perform m_2 iterations of a stationary method applied to $A_j x_j = b_j$ with an initial guess y_j to obtain an approximation y_j.
8. Return y_j.

The V-cycle scheme can be iterated as many time as needed to reduce the error to an acceptable level. For the first iteration we use the initial guess $y_0 = 0$.

2.2.3 The μ-cycle scheme

The V-cycle is just one of a family of multigrid cycling schemes. The entire family is called the *μ-cycle* schemes and is defined recursively by following; see, e.g., [3, p. 42].



(a) V-cycle (b) W-cycle (c) Full multigrid V-cycle

Figure 2.6: Multigrid cycling patterns.

The μ-cycle scheme $y_j = M\mu(y_j, b_j, j, \mu)$

1. If $j = J$, solve $A_J y_J = b_J$ and go to step 8.
2. Perform m_1 iterations of a stationary method applied to $A_j x_j = b_j$ with an initial guess y_j to obtain an approximation y_j.
3. Compute $b_{j+1} = R_{j+1}^{j}(b_j - A_j y_j)$.
4. Set $y_{j+1} := 0$.
5. Repeat μ times:
   - Call recursively $y_{j+1} := M\mu(y_{j+1}, b_{j+1}, j+1, \mu)$.
6. Correct the approximation y_j by $y_j := y_j + I_j^{j+1} y_{j+1}$.
7. Perform m_2 iterations of a stationary method applied to $A_j x_j = b_j$ with an initial guess y_j to obtain an approximation y_j.
8. Return y_j.

In practice, only $\mu = 1$ (which gives the *V-cycle*) and $\mu = 2$ are used. Figure 2.6b shows the pattern for $\mu = 2$ and the resulting *W-cycle*.

2.2.4 The full multigrid algorithm

The μ-cycle schemes start with an initial guess. It is obvious that we want to provide the best possible initial guess. A natural approach is to first solve the problem on a coarser grid and interpolate this solution to the fine grid to be used as a first approximation. This is applied recursively, yielding the full multigrid algorithm; see, e.g., [3, p. 42].

The full multigrid algorithm

1. For $j = 1, \ldots, J-1$
   - Restrict vector b_j to a coarser grid $j+1$ as $b_{j+1} = R_{j+1}^{j} b_j$.
2. Solve $A_J y_J = b_J$ for y_J.
3. For $j = J-1, \ldots, 1$

- Interpolate y_{j+1} to the grid j as $y_j = I_j^{j+1} y_{j+1}$.
- Repeat μ times:
  - Call V-cycle scheme $y_j := V(y_j, b_j, j)$.

Expressed recursively, the algorithm has the following compact form; see, e.g., [3, p. 42], [17, p. 22].

The full multigrid algorithm $y_j = FMG(b_j, j)$

1. If $j = J$, solve $A_J y_J = b_J$ and go to step 8.
2. Restrict the right side vector b_j to a coarser grid $j+1$ as $b_{j+1} = R_{j+1}^j b_j$.
3. Call the full multigrid algorithm recursively $y_{j+1} = FMG(b_{j+1}, j+1)$.
4. Interpolate y_{j+1} to the grid j as $y_j = I_j^{j+1} y_{j+1}$.
5. Repeat μ times:
   - Call V-cycle scheme $y_j := V(y_j, b_j, j)$.
6. Return y_j.

Figure 2.6c shows the pattern for $\mu = 1$.

The convergence of described multigrid methods is based on showing that one step of the method is a contraction. It has been studied in many papers; we refer especially to Hackbusch [8] and McCormick [13].

3. Numerical experiments

After the definition of four test problems we focus on the behavior of the discretization error for higher-order finite element approximations in Experiment 1. The differences between the use of the direct and iterative coarsest grid solver in the V-cycle scheme, are studied in Experiment 2. In Experiment 3 we simulate the fault of a computational device leading to the situation where one or a small number of components of the coarse grid correction are corrupted.

The multigrid solvers used in Experiments 2-3 are always run with zero initial guess.

3.1 Definition of problems

The first three problems are from the class

$$\begin{aligned} -\Delta u &= f \quad \text{in } \Omega = (-1,1)^2, \\ u &= 0 \quad \text{on } \partial\Omega. \end{aligned} \tag{3.1}$$

Problem 1

We consider the problem (3.1) with the manufactured solution u

$$u(x,y) = (x-1)(x+1)(y-1)(y+1),$$

and the right-hand side

$$f(x,y) = -2(x^2+y^2-2).$$

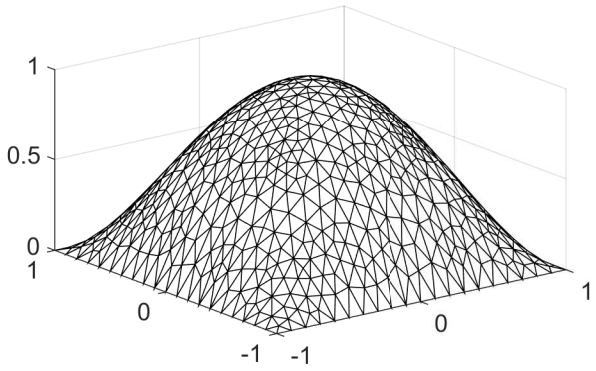


Figure 3.1: Problem 1, solution u.

Problem 2

As the second problem we consider (3.1) with the manufactured solution

$$u(x,y) = \sin(2\pi x)\sin(2\pi y)$$

and the right-hand side

$$f(x,y) = 8\pi^2 \sin(2\pi x)\sin(2\pi y).$$

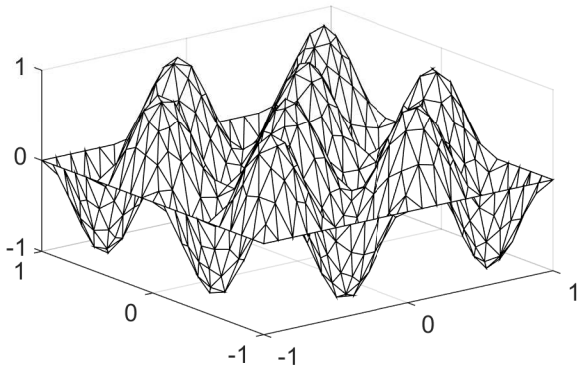


Figure 3.2: Problem 2, solution u.

Problem 3

We consider the problem (3.1) with the manufactured solution

$$u(x,y) = ((x+1)(x-1)(y+1)(y-1))e^{-100(x^2+y^2)}$$

and the right-hand side

$$\begin{aligned} f(x,y) = \; & e^{-100\left(x^2+y^2\right)}\left(40000x^4\left(y^2-1\right) + x^2\left(40000y^4 - 82000y^2 + 41202\right)\right) \\ & + e^{-100\left(x^2+y^2\right)}\left(-40000y^4 + 41202y^2 - 404\right). \end{aligned}$$

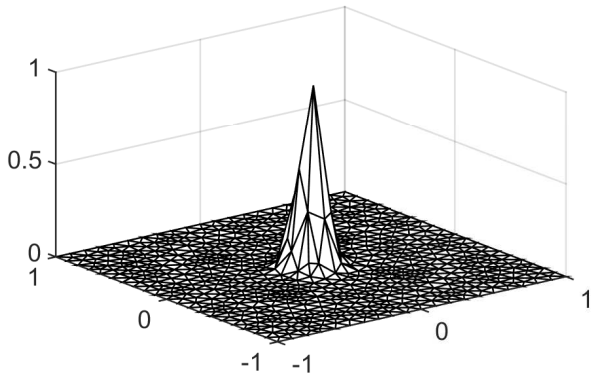
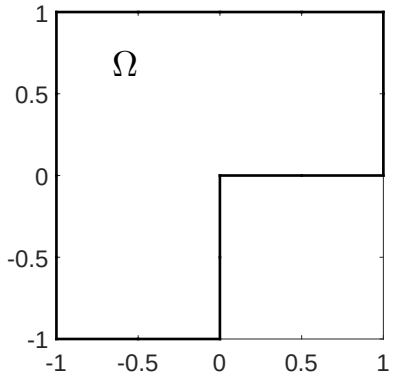


Figure 3.3: Problem 3, solution u.

Problem 4

Consider the L-shape domain Ω



and the problem

$$\begin{aligned} -\Delta u &= f \quad \text{in } \Omega, \\ u &= g \quad \text{on } \partial\Omega, \end{aligned}$$

with right-hand side f and Dirichlet boundary condition g imposed such that the solution has in polar coordinates (r, θ) the form

$$u(r, \theta) = r^{2/3} \sin\left(\frac{2}{3}\theta\right).$$

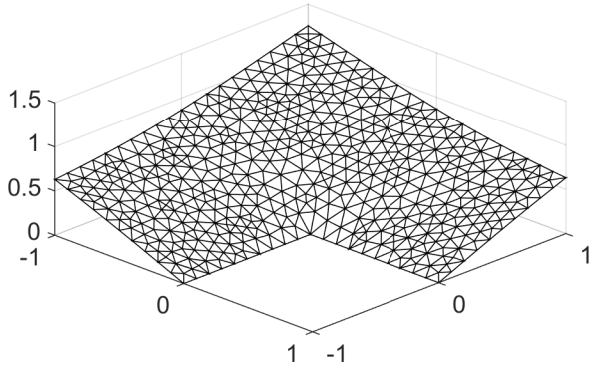


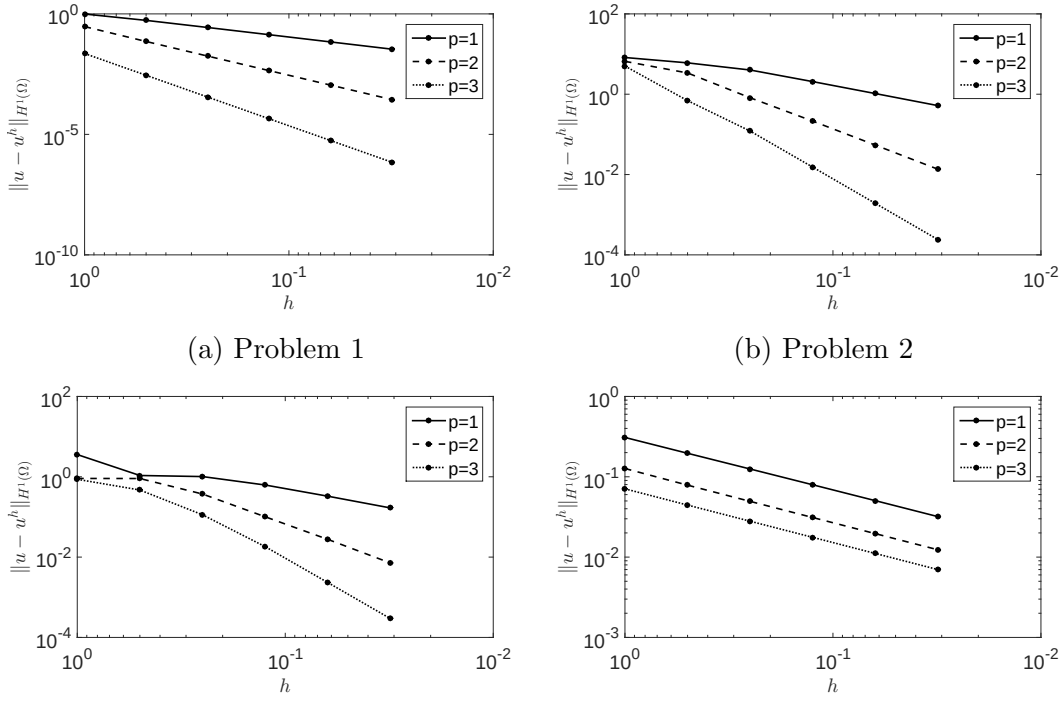
Figure 3.4: Problem 4, solution u.

3.2 Experiment 1 – Discretization error

In the first experiment we focus on the discretization error in the finite element method. We discretize each of the Problems 1–4 using the finite element method (see Section 1.3.1) with varying polynomial degree. The arisen systems of algebraic equations are solved using the MATLAB backslash operator that gives, for our experiments, sufficiently accurate approximations (i.e. approximations with a normwise relative backward error on the machine precision level). We substitute

the corresponding approximation for the Galerkin solution u^h; see Section 1.3. With the knowledge of the exact solutions u we evaluate H^1 norm $\|u - u^h\|_{H^1(\Omega)}$ of the discretization error (up to the quadrature error that is significantly smaller and is neglected here).

Figure 3.5 shows the norm of the discretization error for three different choices of the subspaces $\mathcal{V}_{h,p}$, $p = 1, 2, 3$, on the sequence of uniformly refined meshes, where each triangle is refined into four congruent triangles, i.e. the mesh size h is reduced by factor 1/2.



(a) Problem 1

(b) Problem 2

(c) Problem 3

(d) Problem 4

Figure 3.5: Discretization error in the finite element method with varying polynomial degree $p = 1, 2, 3$, and mesh size h for Problems 1–4.

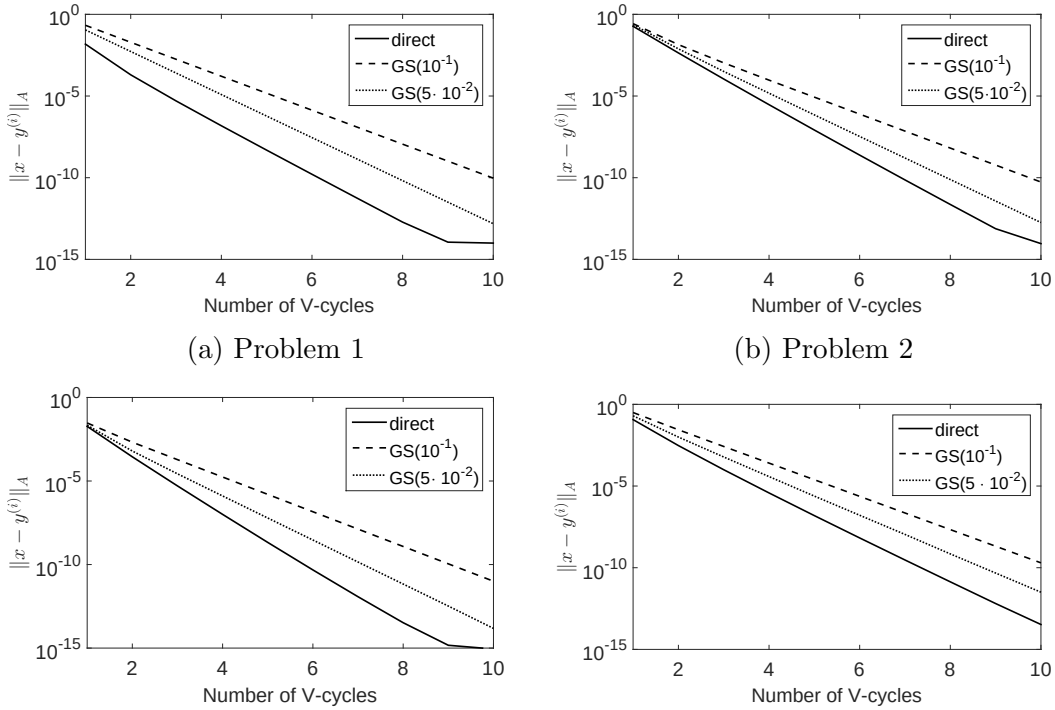
Since the solutions of Problems 1–3 are sufficiently regular (in fact they belong to the space $C^\infty(\Omega)$) we can observe the (asymptotic) convergence of the discretization error norm in agreement with the a priory estimate (1.22) presented in Section 1.3.1. In Problem 4 the solution exhibits a singularity at the reentrant corner (see, e.g., [14]) and we observe the same rate of convergence for polynomial degrees $p = 1, 2, 3$. Recall that here we consider the uniform mesh refinement.

3.3 Experiment 2 – V-cycle scheme with iterative coarsest grid solver

One of the key ingredient of multigrid methods presented in Section 2.2 is the solution of the system of linear algebraic equations corresponding to the coarsest grid. In this experiment we focus on the behavior of the algebraic error in the multigrid V-cycle scheme, where the linear system on the coarsest grid is solved (nearly) exactly or with non-negligible algebraic error.

We consider the V-cycle scheme with three grids and with three pre-smoothing and post-smoothing steps of the Gauss–Seidel method. The system corresponding to the coarsest grid is solved directly using the MATLAB backslash operator (in the figures denoted as `direct`) or iteratively using the Gauss–Seidel method. The Gauss–Seidel method is terminated when the energy norm of the algebraic error drops below the energy norm of the initial error multiplied by the factor 10^{-1} (in the figures $\texttt{GS}(10^{-1})$) or $5 \cdot 10^{-2}$ ($\texttt{GS}(5 \cdot 10^{-2})$).

We discretize Problems 1–4 using the piecewise linear finite element approximations. Resulting matrices are of dimensions approximately 2400 (the finest grid) and 140 (the coarsest grid). Let x denote the solution of the system corresponding to the finest grid and let $y^{(i)}$, $i = 1, \ldots, 10$, denote the approximations given by the V-cycle scheme after i-th iteration. Figure 3.6 shows the energy norm $\|x - y^{(i)}\|_A$ of algebraic error for Problems 1–4 and for three different variants of the coarsest grid solver.



(a) Problem 1

(b) Problem 2

(c) Problem 3

(d) Problem 4

Figure 3.6: Algebraic error in V-cycle iterations with different variants of the coarsest grid solver for Problems 1–4.

We observe that the V-cycle scheme converges for all three variants. Slower convergence of the iterative variants is not surprising such as the fact that variant with higher accuracy ($\texttt{GS}(5 \cdot 10^{-2})$) converges faster than the less accurate one.

3.4 Experiment 3 – V-cycle with fault coarsest grid correction

When solving extreme scale problems the probability of failure of one or more devices of the high performance computing system may not be non-negligible;

see, e.g., [10]. This could lead to the situation where one or a small number of components of currently computed vector are corrupted.

In this experiment we consider the two-grid correction scheme (see Section 2.2), with three pre-smoothing and post-smoothing steps of the Gauss–Seidel method. We consider the *fault two-grid correction scheme* where every coarse grid solve fails in the way that one (fixed) component of the solution e_H is corrupted; here we are using the notation established in Section 2.2. For simplicity, we assume that we are able to detect the failure and replace the defective component by zero. Replacing the corrupted component by zero seems to be justifiable from two reasons. First, the coarse grid correction e_H converges asymptotically to zero vector (as the algebraic error converges to zero). Second, replacing the corrupted component by zero can be understood as not using the coarse grid correction for this component. The correction scheme presented in Section 2.2, i.e. without the failures, will be hereafter called *fault-free two-grid correction scheme*.

We use both the fault and the fault-free two-grid correction schemes for solving the system of linear algebraic equations resulting form the piecewise linear finite element discretization of Problem 1, and plot the energy norms [a] $\|x - y^{(i)}\|_A$ of algebraic errors in Figure 3.7. The sizes of the matrices are 2141 (at the fine grid) and 517 (at the coarse grid).

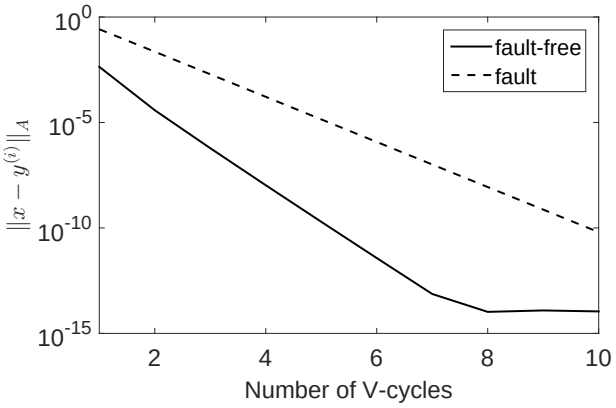
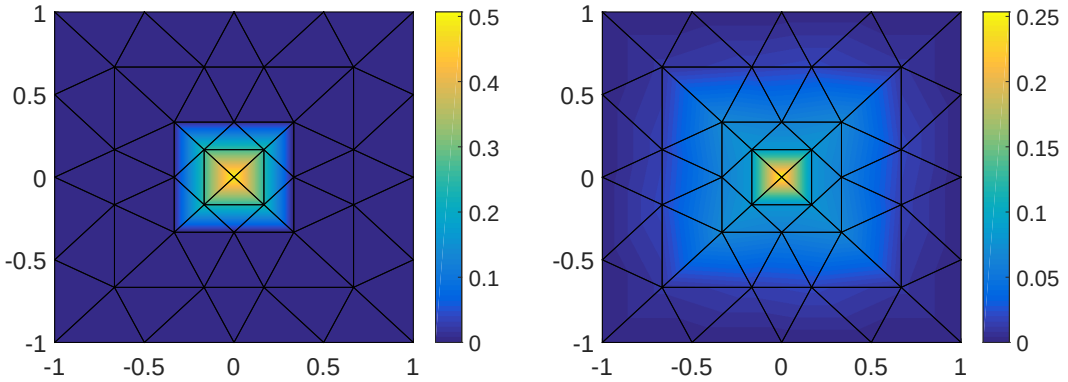


Figure 3.7: Algebraic error in fault and fault-free two-grid correction scheme for Problem 1.

We see that the fault algorithm converges, however its convergence is slower than in the fault-free variant. To further examine the process, we run the fault and fault-free two-grid correction scheme simultaneously and plot the difference of the currently computed approximations in the stages of the two-grid correction scheme in the first iteration. We use upper index F to denote the vectors in the fault variant, e.g. y_h^F, whereas vectors corresponding to the fault-free variant stays denoted as in Section 2.2.

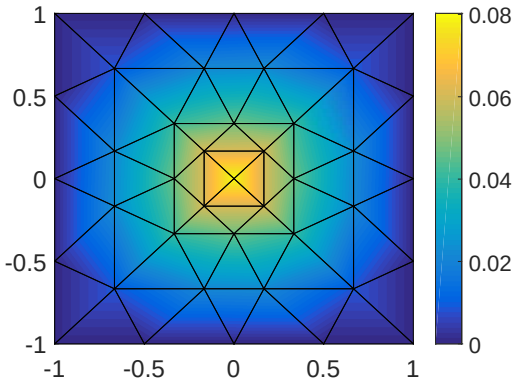
Figure 3.8 shows the piecewise linear functions determined by the difference $y_h^{[0]} - y_h^{[0],F}$ of fault and fault-free approximations after the coarse-grid correction (in (a)), and by the difference $y_h^{[j]} - y_h^{[j],F}$ of the approximations after

[a] As in Experiment 2, x denotes the exact solution of the system corresponding to the fine grid and $y^{(i)}$, $i = 1, \ldots, 10$ the approximations given by the two-grid correction scheme after i-th iteration.

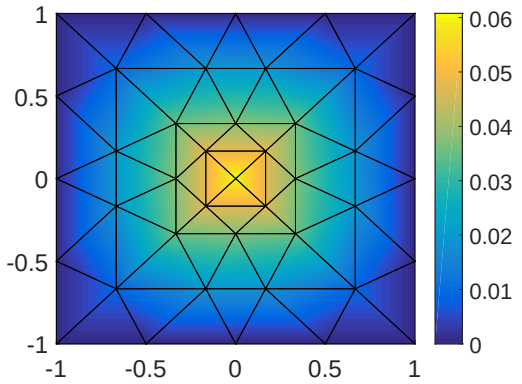


(a) $y_h^{[0]} - y_h^{[0],F}$

(b) $y_h^{[1]} - y_h^{[1],F}$



(c) $y_h^{[2]} - y_h^{[2],F}$



(d) $y_h^{[3]} - y_h^{[3],F}$

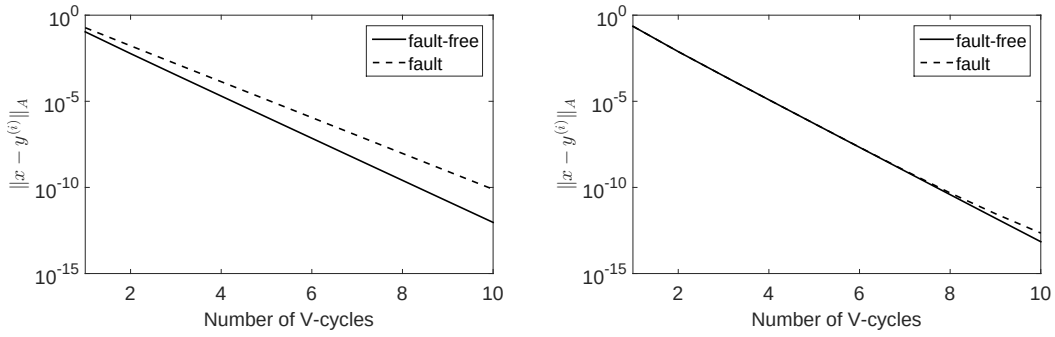
Figure 3.8: Differences between the fault and fault-free approximations in stages of the two-grid correction scheme.

$j = 1, 2, 3$ post-smoothing Gauss–Seidel steps (in (b),(c),(d))[b].

We observe that the difference $y_h^{[0]} - y_h^{[0],F}$ has been significantly reduced within three post smoothing steps of the Gauss–Seidel iteration. Our explanation is as follows. The error $x_h - y_h^{[0],F}$ of the fault two-grid correction scheme can be decomposed as $x_h - y_h^{[0],F} = (x_h - y_h^{[0]}) + (y_h^{[0]} - y_h^{[0],F})$, where $y_h^{[0]} - y_h^{[0],F}$ is a high frequency vector. The error $x_h - y_h^{[j],F}$ of the approximation $y_h^{[j],F}$ after j Gauss–Seidel iterations is equal to $(x_h - y_h^{[j]}) + (y_h^{[j]} - y_h^{[j],F})$. Since applying the Gauss–Seidel method is a a linear procedure, the difference $y_h^{[j]} - y_h^{[j],F}$ is equal to the vector obtained after j iterations of the Gauss–Seidel method for solving the system with the same matrix, zero right-hand side, and the starting vector $y_h^{[0]} - y_h^{[0],F}$. The smoothing property of Gauss–Seidel method then ensures that $y_h^{[0]} - y_h^{[0],F}$ is efficiently reduced; see the discussion in Section 2.1.3.

Figure 3.9 gives resuts analogous to those presented above in Figure 3.8.

[b]Remark the difference between $y^{(j)}$, which denotes the approximation given by the two-grid correction after j-th iteration, and $y_h^{[j]}$, which denotes the approximation after j post-smoothing Gauss–Seidel steps in the two-grid correction scheme.



(a) Problem 1 (b) Problem 2

(c) Problem 3 (d) Problem 4

Figure 3.9: Algebraic error in fault and fault-free V-cycle scheme with five grid, and three pre-smoothing and post-smoothing Gauss-Seidel iterations for Problems 1–4. The sizes of the matrices are 2000 (the finest grid) and 5 (the coarsest grid).

Conclusion

Multigrid methods show that the combination of two simple methods, i.e. a stationary iterative method and the coarse grid correction, can lead to extremely effective algorithms.

Our experiments illustrate, on simple model problems, that multigrid methods are able to deal with the error of the coarse grid approximation caused by solving the coarsest grid problem inexactly or with the error in one or a small number of components of the coarse grid approximation caused by a failure of computational devices. However, for difficult problems, e.g. problems with the inhomogeneous diffusion tensor $\mathbb{K}$, the results may differ.

The current analysis of multigrid methods assumes that the coarsest problem is solved exactly. From this point of view, the analysis of numerical stability of multigrid algorithms is at its very beginning and may be object of our future work.

Bibliography

[1] A. Brandt and O. E. Livne. *Multigrid techniques—1984 guide with applications to fluid dynamics*, volume 67 of *Classics in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2011. Revised edition of the 1984 original.

[2] S. C. Brenner and L. R. Scott. *The mathematical theory of finite element methods*, volume 15 of *Texts in Applied Mathematics*. Springer-Verlag, New York, 1994.

[3] W. L. Briggs, V. E. Henson, and S. F. McCormick. *A multigrid tutorial*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, second edition, 2000.

[4] J. Duintjer Tebbens, I. Hnětynková, M. Plešinger, Z. Strakoš, and P. Tichý. *Analýza metod pro maticové výpočty: základní metody*. Matfyzpress, Prague, 2012.

[5] L. C. Evans. *Partial differential equations*, volume 19 of *Graduate Studies in Mathematics*. American Mathematical Society, Providence, RI, second edition, 2010.

[6] M. S. Gockenbach. *Understanding and implementing the finite element method*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2006.

[7] A. Greenbaum and T. P. Chartier. *Numerical Methods: Design, Analysis, and Computer Implementation of Algorithms*. Princeton University Press, Princeton, NJ, 2012.

[8] W. Hackbusch. *Multigrid methods and applications*, volume 4 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 1985.

[9] W. Hackbusch. *Iterative solution of large sparse systems of equations*, volume 95 of *Applied Mathematical Sciences*. Springer-Verlag, New York, 1994. Translated and revised from the 1991 German original.

[10] M. Huber, B. Gmeiner, U. Rüde, and B. Wohlmuth. Resilience for multigrid software at the extreme scale. *ArXiv e-prints*, 2015. http://arxiv.org/abs/1506.06185.

[11] J. Liesen and Z. Strakoš. *Krylov subspace methods*. Numerical Mathematics and Scientific Computation. Oxford University Press, Oxford, 2013. Principles and analysis.

[12] J. Málek and Z. Strakoš. *Preconditioning and the conjugate gradient method in the context of solving PDEs*, volume 1 of *SIAM Spotlights*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2015.

[13] S. McCormick. *Multigrid Methods*. Frontiers in Applied Mathematics. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1987.

[14] W. F. Mitchell. A collection of 2D elliptic problems for testing adaptive grid refinement algorithms. *Appl. Math. Comput.*, 220:350–364, 2013.

[15] J. Papež. Estimation of the algebraic error and stopping criteria in numerical solution of partial diferential equations. Master's thesis, Charles University in Prague, Faculty of Mathematics and Physics, 2011.

[16] J. Papež, J. Liesen, and Z. Strakoš. Distribution of the discretization and algebraic error in numerical solution of partial differential equations. *Linear Algebra Appl.*, 449:89–114, 2014.

[17] I. Yavneh. Why multigrid methods are so efficient. *Computing in science & engineering*, 8(6):12–22, 2006.