# 5 Multigrid Methods

Multigrid methods belong to the class of preconditioned defect correction methods, in which the preconditioning uses a hierarchy of problems of similar structure but decreasing dimension. They are particularly designed for the solution of the linear systems resulting from the discretization of partial differential equations by grid methods such as finite difference or finite element schemes. But special versions of this method can also be applied to other types of problems not necessarily originating from differential equations. Its main concept is based on the existence of a superposed "continuous" problem of infinite dimension, from which all the smaller problems are obtained in a nested way by some projection process (e. g., a "finite difference discretization" or a "finite element Galerkin method"). On the largest subspace (on the finest grid) the errors and the corresponding defects are decomposed into "high-frequency" and "low-frequency" components, which are treated separately by simple fixed-point iterations for "smoothing" out the former and by correcting the latter on "coarser" subspaces (the "preconditioning" or "coarse-space correction"). This "smoothing" and "coarse-space correcting" is applied recursively on the sequence of nested spaces leading to the full "multigrid algorithm". By an appropriate combination of all these algorithmic components one obtains an "optimal" solution algorithm, which solves a linear system of dimension $n$, such as the model problem considered above, in $\mathcal{O}(n)$ arithmetic operations. In the following, for notational simplicity, we will decribe and analyze the multigrid method within the framework of a low-order finite element Galerkin discretization of the model problem of Section 3.4. In fact, on uniform Cartesian meshes this discretization is closely related (almost equivalent) to the special finite difference scheme considered in Section 3.4. For the details of such a finite element scheme and its error analysis, we refer to the literature, e. g., Rannacher [3].

## 5.1 Multigrid methods for linear systems

For illustration, we consider the linear system

$$A_h x_h = b_h, \tag{5.1.1}$$

representing the discretization of the two-dimensional model problem of Section 3.4 on a finite difference mesh $\mathbb{T}_h$ with mesh size $h \approx m^{-1}$ and dimension $n = m^2 \approx h^{-4}$. Here and below, the quantities related to a fixed subspace (corresponding to a mesh $\mathbb{T}_h$) are labeled by the subscript $h$.

The solution of problem (5.1.1) is approximated by the damped Richardson iteration

$$x_h^{t+1} = x_h^t + \theta_h(b_h - A_h x_h^t) = (I_h - \theta_h A_h)x_h^t + \theta_h b_h, \tag{5.1.2}$$

with a damping parameter $0 < \theta_h \leq 1$. The symmetric, positive definite matrix $A_h$ possesses an ONS of eigenvectors $\{w_h^i,\, i = 1, ..., n_h\}$ corresponding to the ordered eigenvalue

$\lambda_{\min}(A_h) = \lambda_1 \leq ... \leq \lambda_n = \lambda_{\max}(A_h) =: \Lambda_h$. The expansion of the initial error

$$e_h^0 := x_h^0 - x_h = \sum_{i=1}^{n_h} \varepsilon_i w_h^i,$$

induces the corresponding expansion of the iterated errors

$$e_h^t = (I_h - \theta_h A_h)^t e_h^0 = \sum_{i=1}^{n_h} \varepsilon_i (I_h - \theta_h A_h)^t w_h^i = \sum_{i=1}^{n_h} \varepsilon_i (1 - \theta_h \lambda_i)^t w_h^i.$$

Consequently, denoting by $|\cdot|$ the Euclidean norm on $\mathbb{R}^{n_h}$,

$$|e_h^t|^2 = \sum_{i=1}^{n_h} \varepsilon_i^2 (1 - \theta_h \lambda_i)^{2t}. \tag{5.1.3}$$

The assumption $0 < \theta_h \leq \Lambda_h^{-1}$ is sufficient for the convergence of the Richardson iteration. Because of $|1 - \theta_h \lambda_i| \ll 1$ for larger $\lambda_i$ and $|1 - \theta_h \lambda_1| \approx 1$ "high-frequency" components of the error decay fast, but "low-frequency" components only very slowly. The same holds for the residuum $r_h^t = b_h - A_h x_h^t = A_h e_h^t$. Hence already after a few iterations there holds

$$|r_h^t|^2 \approx \sum_{i=1}^{[N/2]} \varepsilon_i^2 \lambda_i^2 (1 - \theta_h \lambda_i)^{2t}, \qquad [n/2] := \max\{m \in \mathbf{N} | \, m \leq n/2\}. \tag{5.1.4}$$

This may be interpreted as follows: The iterated defect $r_h^t$ on the mesh $\mathbb{T}_h$ is "smooth". Hence, it can be approximated well on the coarser mesh $\mathbb{T}_{2h}$ with mesh size $2h$. The resulting defect equation for the computation of the correction to the approximation $x_h^t$ on $\mathbb{T}_h$ would be less costly because of its smaller dimension $n_{2h} \approx n_h/4$.

This defect correction process in combination with recursive coarsening can be carried on to a coarsest mesh, on which the defect equation can finally be solved exactly. The most important components of this multigrid process are the "smoothing iteration", $x_h^\nu = S_h^\nu(x_h^0)$ and certain transfer operations between functions defined on different meshes. The smoothing operation $S_h(\cdot)$ is usually given in form of a simple fixed-point iteration (e. g., the Richardson iteration)

$$x_h^{\nu+1} = S_h(x_h^\nu) := (I_h - C_h^{-1} A_h) x_h^\nu + C_h^{-1} b_h,$$

with the iteration matrix $S_h := I_h - C_h^{-1} A_h$.

### 5.1.1 Multigrid methods in the "finite element" context

For the formulation of the multigrid process, we consider a sequence of nested grids $\mathbb{T}_l = \mathbb{T}_{h_l}$, $l = 0, ..., L$, of increasing fineness $h_0 > ... > h_l > ... > h_L$ (for instance obtained by successively refining a coarse starting grid) and corresponding finite element spaces $V_l := V_{h_l}$ of increasing dimension $n_l$, which are subspaces of the "continuous" solu-

tion space $V = H_0^1(\Omega)$ (first-order Sobolev[1] space on $\Omega$ including zero Dirichlet boundary conditions). Here, we think of spaces of continuous, with respect to the mesh $\mathbb{T}_h$ piecewise *linear* (on triangular meshes) or piecewise (isoparametric) *bilinear* (on quadrilateral meshes) functions. For simplicity, we assume that the finite element spaces are hierachically ordered,

$$V_0 \subset V_1 \subset ... \subset V_l \subset ... \subset V_L\,. \tag{5.1.5}$$

This structural assumption eases the analysis of the multigrid process but is not essential for its practical success.

### The finite element Galerkin scheme

As usual, we write the continuous problem and its corresponding finite element Galerkin approximation in compact variational form

$$a(u,\varphi) = (f,\varphi)_{L^2} \quad \forall \varphi \in V, \tag{5.1.6}$$

and, analogously on the mesh $\mathbb{T}_h$

$$a(u_h,\varphi_h) = (f,\varphi_h)_{L^2} \quad \forall \varphi_h \in V_h. \tag{5.1.7}$$

Here, $a(u,\varphi) := (Lu,\varphi)_{L^2}$ is the "energy bilinear form" corresponding to the (elliptic) differential operator $L$ and $(f,\varphi)_{L^2}$ the $L^2$-scalar product on the solution domain $\Omega$. In the model problem discussed above this notation has the explicit form $Lu = -\Delta u$ and

$$a(u,\varphi) = \int_\Omega \nabla u(x)\nabla\varphi(x)\,dx, \quad (f,\varphi)_{L^2} = \int_\Omega f(x)\varphi(x)\,dx.$$

The finite element subspace $V_h \subset V$ has a natural so-called "nodal basis" (Lagrange basis) $\{b^1,\ldots,b^{n_h}\}$ characterized by the interpolation property $b^i(a_j) = \delta_{ij}$, $i,j = 1,\ldots,n_h$, where $a_j$ are the nodal points of the mesh $\mathbb{T}_h$. Between the finite element function $u_h \in V_h$ and the corresponding nodal-value vector $x_h \in \mathbb{R}^{n_h}$, we have the connection $u_h(a_j) = x_{h,j}$, $j = 1,...,n_h$,

$$u_h = \sum_{j=1}^{n_h} x_{h,j} b^j = \sum_{j=1}^{n_h} u_h(a_j) b^j.$$

Using this notation the discrete problems (5.1.7) can be written in the following form:

$$\sum_{j=1}^{n_h} x_{h,j}\, a(b^j,b^i) = (f,b^i)_{L^2}, \quad i = 1,\ldots,n_h,$$

[1]Sergei Lvovich Sobolev (1908–1989): Russian mathematician; worked in Leningrad (St. Petersburg) and later at the famous Steklov-Institute for Mathematics of the Academy of Sciences in Moscow; fundamental contributions to the theory of partial differential equations concept of generalized (distributional) solutions, "Sobolev spaces"; worked also on numerical methods, numerical quadrature.

which is equivalent to the linear system

$$A_h x_h = b_h, \tag{5.1.8}$$

with the "system matrix" ("stiffness matrix") $A_h = (a_{ij})_{i,j=1}^{n_h} \in \mathbb{R}^{n_h \times n_h}$ and "load vector" $b_h = (b_j)_{j=1}^{n_h} \in \mathbb{R}^{n_h}$ defined by

$$a_{ij} := a(b^j, b^i), \quad b_j := (f, b^j)_{L^2}, \quad i, j = 1, \ldots, n_h.$$

For finite element functions $u_h = \sum_{i=1}^{n_h} x_{h,i} b^i$ and $v_h = \sum_{i=1}^{n_h} y_{h,i} b^i$ there holds

$$a(u_h, v_h) = (A_h x_h, y_h)_2.$$

The system matrix $A_h$ is symmetric and positive definite by construction and has a condition number of size $\mathrm{cond}_2(A_h) = \mathcal{O}(h^{-2})$. Additionally, we will use the so-called "mass matrix" $M_h = (m_{ij})_{i,j=1}^{n_h}$ defined by

$$m_{ij} := (b^j, b^i)_{L^2}, \quad i, j = 1, \ldots, n_h.$$

For finite element functions $u_h = \sum_{i=1}^{n_h} x_{h,i} b^i$ and $v_h = \sum_{i=1}^{n_h} y_{h,i} b^i$ there holds

$$(u_h, v_h)_{L^2} = (M_h x_h, y_h)_2.$$

The mass matrix $M_h$ is also symmetric and positive definite by construction and has a condition number of size $\mathrm{cond}_2(A_h) = \mathcal{O}(1)$.

For the exact "discrete" solution $u_h \in V_h$ there holds the error estimate

$$\|u - u_h\|_{L^2} \leq c\, h^2 \, \|f\|_{L^2}. \tag{5.1.9}$$

Now, we seek a solution process which produces an approximation $\tilde{u}_h \in V_h$ to $u_h$ satisfying

$$\|u_h - \tilde{u}_h\|_{L^2} \leq c\, h^2 \, \|f\|_{L^2}. \tag{5.1.10}$$

This process is called "complexity-optimal" if the arithmetic work for achieving this accuracy is of size $\mathcal{O}(n_h)$ uniformly with respect to the mesh size $h$. We will see below that the multigrid method is actually optimal is this sense if all its components are properly chosen.

### The multigrid process

Let $u_L^0 \in V_L$ be an initial guess for the exact discrete solution $u_L \in V_L$ on mesh level $L$ (For example, $u_L^0 = 0$ or $u_L^0 = u_{L-1}$ if such a coarse-grid solution is available.). Then, $u_L^0$ is "smoothed" ("pre-smoothed") by applying $\nu$ steps, e. g., of the damped Richardson iteration starting from $\bar{u}_L^0 := u_L^0$. This reads in variational from as follows:

$$(\bar{u}_L^k, \varphi_L)_{L^2} = (\bar{u}_L^{k-1}, \varphi_L)_{L^2} + \theta_L \big\{ (f, \varphi_L)_{L^2} - a(\bar{u}_L^{k-1}, \varphi_L) \big\} \quad \forall \varphi_L \in V_L, \tag{5.1.11}$$

where $\theta_L = \lambda_{max}(A_h)^{-1}$. For the resulting smoothed approximation $\bar{u}_L^\nu$, we define the "defect" $d_L \in V_L$ (without actually computing it) as follows:

$$(d_L, \varphi_L)_{L^2} := (f, \varphi_L)_{L^2} - a(\bar{u}_L^\nu, \varphi_L), \quad \varphi_L \in V_L. \tag{5.1.12}$$

Since $V_{L-1} \subset V_L$, we obtain the "defect equation" on the next coarser mesh $\mathbb{T}_{L-1}$ as

$$a(q_{L-1}, \varphi_{L-1}) = (d_L, \varphi_{L-1})_{L^2} = (f, \varphi_{L-1})_{L^2} - a(\bar{u}_L^\nu, \varphi_{L-1}) \quad \forall \varphi_{L-1} \in V_{L-1}. \tag{5.1.13}$$

The correction $q_{L-1} \in V_{L-1}$ is now computed either exactly (for instance by a direct solver) or only approximately by a defect correction iteration $q_{L-1}^0 \to q_{L-1}^R$ using the sequence of coarser meshes $\mathbb{T}_{L-2}, ..., \mathbb{T}_0$. The result $q_{L-1}^R \in V_{L-1}$ is then interpreted as an element of $V_L$ and used for correcting the preliminary approximation $\bar{u}_L^\nu$:

$$\bar{\bar{u}}_L^0 := \bar{u}_L^\nu + \omega_L q_{L-1}^R. \tag{5.1.14}$$

The correction step may involve a damping parameter $\omega_L \in (0, 1]$ in order to minimize the residual of $\bar{\bar{u}}_L$. This practically very useful trick will not be further discussed here, i. e., in the following, we will mostly set $\omega_L = 1$. The obtained corrected approximation $\bar{\bar{u}}_L$ is again smoothed ("post-smoothing") by applying another $\mu$ steps of the damped Richardson iteration starting from $\bar{\bar{u}}_L^0 := \bar{\bar{u}}_L$:

$$(\bar{\bar{u}}_L^k, \varphi_L)_{L^2} = (\bar{\bar{u}}_L^{k-1}, \varphi_L)_{L^2} + \theta_L \big\{ (f, \varphi_L)_{L^2} - a(\bar{\bar{u}}_L^{k-1}, \varphi_L) \big\} \quad \forall \varphi_L \in V_L. \tag{5.1.15}$$

The result is then accepted as the next multigrid iterate, $u_L^1 := \bar{\bar{u}}_L^\mu$, completing one step of the multigrid iteration ("multigrid cycle") on mesh level $L$. Each such cycle consists of $\nu + \mu$ Richardson smoothing steps (on level $L$), which each requires the inversion of the mass matrix $M_h$, and the solution of the correction equation on the next coarser mesh.

Now, we will formulate the multigrid algorithm using a more abstract, functional analytic notation, in order to better understand its structure and to ease its convergence analysis. To the system matrices $A_l = A_{h_l}$ on the sequence of meshes $\mathbb{T}_l$, $l = 0, 1, \ldots, L$, we associate operators $\mathcal{A}_l : V_l \to V_l$ by setting

$$(\mathcal{A}_l v_l, w_l)_{L^2} = a(v_l, w_l) = (A_l y_l, z_l)_2 \quad \forall v_l, w_l \in V_l, \tag{5.1.16}$$

where $v_l = (y_{l,i})_{i=1}^{n_l}$, $w_l = (z_{l,i})_{i=1}^{n_l}$. Further, let $\mathcal{S}_l(\cdot)$ denote the corresponding smoothing operations with (linear) iteration operators (Richardson operator) $\mathcal{S}_l = \mathcal{I}_l - \theta_l \mathcal{A}_l : V_l \to V_l$ where $\mathcal{A}_l$ is the "system operator" defined above and $\mathcal{I}_l$ denotes the identity operator on $V_l$. Finally, we introduce the operators by which the transfers of functions between consecutive subspaces are accomplished:

$$r_l^{l-1} : V_l \to V_{l-1} \ \text{("restriction")}, \quad p_{l-1}^l : V_{l-1} \to V_l \ \text{("prolongation")}.$$

In the finite element context these operators are naturally chosen as $r_l^{l-1} = \mathcal{P}_{l-1}$, the $L^2$ projection onto $V_{l-1}$, and $p_{l-1}^l = id.$, the natural embedding of $V_{l-1} \subset V_l$ into $V_l$.

Now, using this notation, we reformulate the multigrid process introduced above for solving the linear system on the finest mesh $\mathbb{T}_L$:

$$\mathcal{A}_L u_L = f_L := \mathcal{P}_L f. \tag{5.1.17}$$

*Multigrid process:* Starting from an initial vector $u_L^0 \in V_L$ iterates $u_L^t$ are constructed by the recursive formula

$$u_L^{(t+1)} = \mathrm{MG}(L, u_L^{(t)}, f_L). \tag{5.1.18}$$

Let the $t$-th multigrid iterate $u_L^{(t)}$ be determined.

*Coarse grid solution:* For $l = 0$, the operation $\mathrm{MG}(0, 0, g_0)$ yields the exact solution of the system $\mathcal{A}_0 v_0 = g_0$ (obtained for instance by a direct method),

$$v_0 = \mathrm{MG}(0, \cdot, g_0) = \mathcal{A}_0^{-1} g_0. \tag{5.1.19}$$

*Recursion:* Let for some $1 \le l \le L$ the system $\mathcal{A}_l v_l = g_l$ to be solved. With parameter values $\nu$, $\mu \ge 1$ the value

$$\mathrm{MG}(l, v_l^0, g_l) := v_l^1 \approx v_l \tag{5.1.20}$$

is recursively definined by the following steps:

i) *Pre-smoothimg:*

$$\bar{v}_l := \mathcal{S}_l^{\nu}(v_l^0); \tag{5.1.21}$$

ii) *Defect formation:*

$$d_l := g_l - \mathcal{A}_l \bar{v}_l; \tag{5.1.22}$$

iii) *Restriction:*

$$\tilde{d}_{l-1} := r_l^{l-1} d_l; \tag{5.1.23}$$

iv) *Defect equation:* Starting from $q_{l-1}^0 := 0$ for $1 \le r \le R$ the iteration proceeds as follows:

$$q_{l-1}^r := \mathrm{MG}(l-1, q_{l-1}^{r-1}, \tilde{d}_{l-1}); \tag{5.1.24}$$

$$\tag{5.1.25}$$

v) *Prolongation:*

$$q_l := p_{l-1}^l q_{l-1}^R\,; \tag{5.1.26}$$

vi) *Correction:* With a damping parameter $\omega_l \in (0, 1]$,

$$\bar{\bar{v}}_l := \bar{v}_l + \omega_l q_l; \tag{5.1.27}$$

vii) *Post-smoothing:*

$$v_l^1 := \mathcal{S}_l^\mu(\bar{\bar{v}}_l); \tag{5.1.28}$$

In case $l = L$, one sets:

$$u_L^{t+1} := v_l^1. \tag{5.1.29}$$

We collect the afore mentioned algorithmic steps into a compact systematics of the multigrid cycle $u_L^t \to u_L^{t+1}$:

$$\begin{aligned} u_L^t \;\to\; \bar{u}_L^t = S_L^\nu(u_L^t) \;\to\; d_L &= f_L - \mathcal{A}_L \bar{u}_L^t \\ \downarrow\quad \tilde{d}_{L-1} &= r_L^{L-1} d_{L-1} \quad \text{(restriction)} \\ q_{L-1} &= \tilde{\mathcal{A}}_{L-1}^{-1} \tilde{d}_{L-1} \quad (R\text{-times defect correction}) \\ \downarrow\quad \tilde{q}_L &= p_{L-1}^L q_{L-1} \quad \text{(prolongation)} \\ \bar{\bar{u}}_L^t = \bar{u}_L^t + \omega_L \tilde{q}_L \;\to\; u_L^{t+1} &= S_L^\mu(\bar{\bar{u}}_L^t). \end{aligned}$$

If the defect equation $\mathcal{A}_{L-1} q_{L-1} = \tilde{d}_{L-1}$ on the coarser mesh $\mathbb{T}_{L-1}$ is solved "exactly", one speaks of a "two-grid method". In practice, the two-grid process is continued recursively to the "multigrid method" up to the coarsest mesh $\mathbb{T}_0$. This process can be organized in various ways depending essentially on the choice of the iteration parameter $R$, which determines how often the defect correction step is repeated on each mesh level. In practice, for economical reasons, only the cases $R = 1$ and $R = 2$ play a role. The schemes of the corresponding multigrid cycles, the "V-cycle" and the "W-cycle", are shown in Fig. 5.1. Here, "•" represent "smoothing" and "defect correction" on the meshes $\mathbb{T}_l$, and lines "–" stand for the transfer between consecutive mesh levels.

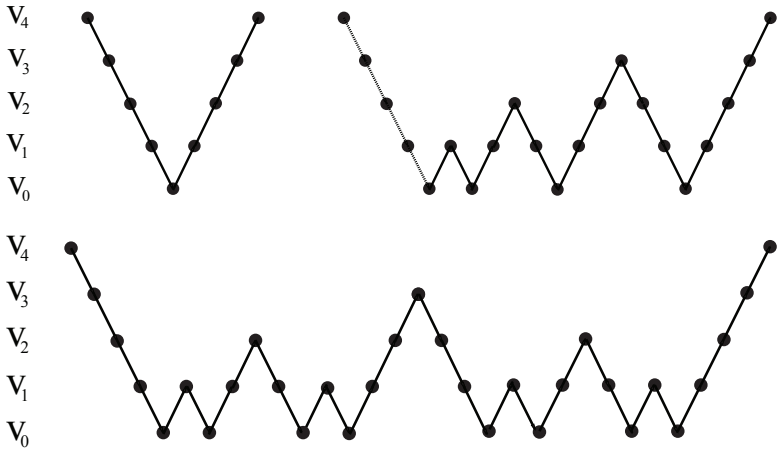


Figure 5.1: *Scheme of a multigrid algorithm organized as V- (top left), F- (top right), and W-cycle (bottom line).*

The V-cycle is very efficient (if it works at all), but often suffers from instabilities caused by irregularities in the problem to be solved, such as strong anisotropies in the differential operator, boundary layers, corner singularities, nonuniformities and deteriorations in the meshes (local mesh refinement and cell stretching), etc.. In contrast to that, the W-cycle is more robust but usually significantly more expensive. Multigrid methods with $R \geq 3$ are too inefficient. A good compromise between robustness and efficiency is provided by the so-called "F-cycle" shown in Fig. 5.1. This process is usually started on the finest mesh $\mathbb{T}_L$ with an arbitrary initial guess $u_L^0$ (most often $u_L^0 = 0$). However, for economical reasons, one may start the whole multigrid process on the coarsest mesh $\mathbb{T}_0$ and then use the approximate solutions obtained on intermediate meshes as starting values for the iteration on the next finer meshes. This "nested" version of the multigrid method will be studied more systematically below.

*Nested multigrid:* Starting from some initial vector $u_0 := \mathcal{A}_0^{-1} f_0$ on the coarsest mesh $\mathbb{T}_0$, for $l = 1, ..., L$, successively approximations $\tilde{u}_l \approx u_l$ are computed by the following recursion:

$$
\begin{aligned}
& u_l^0 = p_{l-1}^l \tilde{u}_{l-1} \\
& u_l^t = \mathrm{MG}(l, u_l^{t-1}, f_l), \quad t = 1, ..., t_l, \quad \|u_l^{t_l} - u_l\|_{L^2} \leq \hat{c}\, h_l^2 \, \|f\|_{L^2}, \\
& \tilde{u}_l = u_l^{t_l}.
\end{aligned}
$$

**Remark 5.1:** Though the multigrid iteration in V-cycle modus may be unstable, it can be used as preconditioners for an "outer" CG (in the symmetric case) or GMRES iteration (in the nonsymmetric case). This approach combines the robustness of the Krylov space method with the efficiency of the multigrid methods and has been used very successfully for the solution of various nonstandard problems, involving singularities, indefiniteness, saddle-point structure, and multi-physics coupling.

**Remark 5.2:** There is not something like *the* multigrid algorithm. The successful realization of the multigrid concept requires a careful choice of the various components such as the "smoother" $\mathcal{S}_l$, the coarse-mesh operators $\mathcal{A}_l$, and the mesh-transfer operators $r_l^{l-1}$, $p_{l-1}^l$, specially adapted to the particular structure of the problem considered. In the following, we discuss these algorithmic componenents in the context of the finite element discretization, e. g., of the model problem from above.

i) *Smoothers:* "Smoothers" are usually simple fixed-point iterations, which could principally also be used as "solvers", but with a very bad convergence rate. They are applied on each mesh level only a few times ($\nu, \mu \sim 1 - 4$), for damping out the high-frequency components in the errors or the residuals. In the following, we consider the damped Richardson iteration with iteration matrix

$$\mathcal{S}_l := \mathcal{I}_l - \theta_l \mathcal{A}_l \,, \quad \theta_l = \lambda_{max}(\mathcal{A}_l)^{-1}, \tag{5.1.30}$$

as smoother, which, however, only works for very simple (scalar) and non-degenerate problems.

**Remark 5.3:** More powerful smoothers are based on the Gauß-Seidel and the ILU iteration. These methods also work well for problems with certain pathologies. For example, in case of strong advection in the differential equation, if the mesh points are numbered in the direction of the transport, the system matrix has a dominant lower triangular part $L$, for which the Gauß-Seidel method is nearly "exact". For problems with degenerate coefficients in one space direction or on strongly anisotropic meshes the system matrix has a dominant tridiagonal part, for which the ILU method is nearly "exact". For indefinite saddle-point problems certain "block" iterations are used, which are specially adapted to the structure of the problem. Examples are the so-called "Vanka-type" smoothers, which are used in solving the "incompressible" Navier-Stokes equations in Fluid Mechanics.

ii) *Grid transfers:* In the context of a finite element discretization with nested subspaces $V_0 \subset V_1 \subset ... \subset V_l \subset ... \subset V_L$ the generic choice of the prolongation $p_{l-1}^l : V_{l-1} \to V_l$ is the cellwise embedding, and of the restriction $r_l^{l-1} : V_l \to V_{l-1}$ the $L^2$ projection. For other discretizations (e. g., finite difference schemes), one uses appropriate interpolation operators (e. g., bi/trilinear interpolation).

iii) *Corse-grid operators:* The operators $\mathcal{A}_l$ on the several spaces $V_l$ do not need to correspond to the same discretization of the original "continuous" problem. This aspect becomes important in the use of mesh-dependent numerical diffusion ("upwinding", "streamline diffusion", etc.) for the treatment of stronger transport. Here, we only consider the ideal case that all $\mathcal{A}_l$ are defined by the same finite element discretization on the mesh family $\{\mathbb{T}_l\}_{l=0,...,L}$. In this case, we have the following useful identity:

$$\begin{aligned}(\mathcal{A}_{l-1}v_{l-1}, w_{l-1})_{L^2} &= a(v_{l-1}, w_{l-1}) \\ &= a(p_{l-1}^l v_{l-1}, p_{l-1}^l w_{l-1}) \\ &= (\mathcal{A}_l p_{l-1}^l v_{l-1}, p_{l-1}^l w_{l-1})_{L^2} = (r_l^{l-1}\mathcal{A}_l p_{l-1}^l v_{l-1}, w_{l-1})_{L^2},\end{aligned} \tag{5.1.31}$$

for all $w_{l-1} \in V_{l-1}$, which means that

$$\mathcal{A}_{l-1} = r_l^{l-1}\mathcal{A}_l p_{l-1}^l. \tag{5.1.32}$$

iv) *Coarse-grid correction:* The correction step contains a damping parameter $\omega_l \in (0, 1]$. It has proved very useful in practice to choose this parameter such that the defect $\mathcal{A}_l\bar{v}_l - \tilde{d}_{l-1}$ becomes minimal. This leads to the formula

$$\omega_l = \frac{(\mathcal{A}_l\bar{v}_l, \tilde{d}_{l-1} - \mathcal{A}_l\bar{v}_l)_{L^2}}{\|\mathcal{A}_l\bar{v}_l\|_{L^2}^2}. \tag{5.1.33}$$

In the following analysis of the multigrid process, for simplicity, we will make the choice $\omega_l = 1$.

### 5.1.2 Convergence analysis

The classical analysis of the multigrid process is based on its interpretation as a defect-correction iteration and the concept of recursive application of the two-grid method. For

simplicity, we assume that only pre-smoothing is used, i. e., $\nu > 0$, $\mu = 0$, and that in the correction step no damping is applied, i. e., $\omega_l = 1$. Then, the two-grid algorithm can be written in the following form:

$$\begin{aligned} u_L^{t+1} &= S_L^\nu(u_L^t) + p_{L-1}^L \mathcal{A}_{L-1}^{-1} r_L^{L-1}\big(f_L - \mathcal{A}_L S_L^\nu(u_L^t)\big) \\ &= S_L^\nu(u_L^t) + p_{L-1}^L \mathcal{A}_{L-1}^{-1} r_L^{L-1}\mathcal{A}_L\big(u_L - S_L^\nu(u_L^t)\big). \end{aligned}$$

Hence, for the iteration error $e_L^t := u_L^t - u_L$ there holds

$$e_L^{t+1} = \big(\mathcal{I}_L - p_{L-1}^L \mathcal{A}_{L-1}^{-1} r_L^{L-1}\mathcal{A}_L\big)\big(S_L^\nu(u_L^t) - u_L\big). \tag{5.1.34}$$

The smoothing operation is given in (affin)-lineare form as

$$S_L(v_L) := S_L v_L + g_L,$$

and as fixed-point iteration satisfies $S_L(u_L) = u_L$. From this, we conclude that

$$S_L^\nu(u_L^t) - u_L = S_L\big(S_L^{\nu-1}(u_L^t) - u_L\big) = ... = S_L^\nu e_L^t.$$

With the so-called "two-grid operator"

$$\mathrm{ZG}_L(\nu) := (\mathcal{I}_L - p_{L-1}^L \mathcal{A}_{L-1}^{-1} r_L^{L-1}\mathcal{A}_L)\mathcal{S}_L^\nu$$

there consequently holds

$$e_L^{t+1} = \mathrm{ZG}_L(\nu) e_L^t\,. \tag{5.1.35}$$

**Theorem 5.1 (Two-grid convergence):** *For sufficiently frequent smoothimg, $\nu > 0$, the two-grid method converges with a rate independent of the mesh level $L$:*

$$\|\mathrm{ZG}_L(\nu)\|_{L^2} \le \rho_{\mathrm{ZG}}(\nu) = c\,\nu^{-1} < 1\,. \tag{5.1.36}$$

**Proof.** We write

$$\mathrm{ZG}_L(\nu) = (\mathcal{A}_L^{-1} - p_{L-1}^L \mathcal{A}_{L-1}^{-1} r_L^{L-1})\mathcal{A}_L\mathcal{S}_L^\nu \tag{5.1.37}$$

and estimate as follows:

$$\|\mathrm{ZG}_L(\nu)\|_{L^2} \le \|\mathcal{A}_L^{-1} - p_{L-1}^L \mathcal{A}_{L-1}^{-1} r_L^{L-1}\|_{L^2}\|\mathcal{A}_L\mathcal{S}_L^\nu\|_{L^2}. \tag{5.1.38}$$

The first term on the right-hand side describes the quality of the approximation of the fine-grid solution on the next coarser mesh, while the second term represents the smoothing effect. The goal of the further analysis is now to show that the smoother $S_L(\cdot)$ possesses the so-called "smoothing property",

$$\|\mathcal{A}_L\mathcal{S}_L^\nu v_L\|_{L^2} \le c_s \nu^{-1} h_L^{-2}\|v_L\|_{L^2}, \quad v_L \in V_L\,, \tag{5.1.39}$$

and the coarse-grid correction possesses the so-called "approximation property" ,

$$\|(\mathcal{A}_L^{-1} - p_{L-1}^L \mathcal{A}_{L-1}^{-1} r_L^{L-1}) v_L\|_{L^2} \le c_a h_L^2 \|v_L\|_{L^2}, \quad v_L \in V_L\,, \tag{5.1.40}$$

with positive constants $c_s$, $c_a$ independent of the mesh level $L$. Combination of these two estimates then yields the asserted estimate (5.1.36). For sufficiently frequent smoothing, we have $\rho_{ZG} := c\nu^{-1} < 1$ and the two-grid algorithm converges with a rate uniformly w.r.t. the mesh level $L$. All constants appearing in the following are independent of $L$.

i) *Smoothing property:* The selfadjoint operator $\mathcal{A}_L$ possesses real, positive eigenvalues $0 < \lambda_1 \le ... \le \lambda_i \le ... \le \lambda_{n_L} =: \Lambda_L$ and a corresponding $L^2$-ONS of eigenfunctions $\{w^1, ..., w^{n_L}\}$, such that each $v_L \in V_L$ can be written as $v_L = \sum_{i=1}^{n_L} \gamma_i w^i$, $\gamma_i = (v_L, w^i)_{L^2}$. For the Richardson iteration operator,

$$\mathcal{S}_L := \mathcal{I}_L - \theta_L \mathcal{A}_L : V_L \to V_L, \quad \theta_L = \Lambda_L^{-1}, \tag{5.1.41}$$

there holds

$$\mathcal{A}_L \mathcal{S}_L^\nu v_L = \sum_{i=1}^{n_L} \gamma_i \lambda_i \Big(1 - \frac{\lambda_i}{\Lambda_L}\Big)^\nu w^i, \tag{5.1.42}$$

and, consequently,

$$\begin{aligned}
\|\mathcal{A}_L \mathcal{S}_L^\nu v_L\|_{L^2}^2 &= \sum_{i=1}^{n_L} \gamma_i^2 \lambda_i^2 \Big(1 - \frac{\lambda_i}{\Lambda_L}\Big)^{2\nu} \\
&\le \Lambda_L^2 \max_{1\le i\le n_L} \Big\{ \Big(\frac{\lambda_i}{\Lambda_L}\Big)^2 \Big(1 - \frac{\lambda_i}{\Lambda_L}\Big)^{2\nu} \Big\} \sum_{i=1}^{n_L} \gamma_i^2 \\
&= \Lambda_L^2 \max_{1\le i\le n_L} \Big\{ \Big(\frac{\lambda_i}{\Lambda_L}\Big)^2 \Big(1 - \frac{\lambda_i}{\Lambda_L}\Big)^{2\nu} \Big\} \|v_L\|_{L^2}^2.
\end{aligned}$$

By the relation (exercise)

$$\max_{0\le x\le 1} \{x^2 (1-x)^{2\nu}\} \le (1+\nu)^{-2} \tag{5.1.43}$$

it follows that

$$\|\mathcal{A}_L \mathcal{S}_L^\nu v_L\|_{L^2}^2 \le \Lambda_L^2 (1+\nu)^{-2} \|v_L\|_{L^2}^2. \tag{5.1.44}$$

The relation $\Lambda_L \le c h_L^{-2}$ eventually implies the asserted inequality for the Richardson iteration operator:

$$\|\mathcal{A}_L \mathcal{S}_L^\nu\|_{L^2} \le c_s \nu^{-1} h_L^{-2}, \quad \nu \ge 1. \tag{5.1.45}$$

ii) *Approximation property:* We recall that in the present context of nested subspaces $V_l$ prolongationen and restriction operators are given by

$$p_{L-1}^L = id. \text{ (identity)}, \quad r_L^{L-1} = \mathcal{P}_{L-1} \ (L^2 \text{ projection}).$$

Further, the operator $\mathcal{A}_L : V_L \to V_L$ satisfies

$$(\mathcal{A}_L v_L, \varphi_L)_{L^2} = a(v_L, \varphi_L), \quad v_L, \varphi_L \in V_L.$$

For an arbitrary but fixed $f_L \in V_L$ and functions $v_L := \mathcal{A}_L^{-1} f_L$, $v_{L-1} := \mathcal{A}_{L-1}^{-1} r_L^{L-1} f_L$ there holds:

$$\begin{aligned} a(v_L, \varphi_L) &= (f_L, \varphi_L)_{L^2} \quad \forall \varphi_L \in V_L, \\ a(v_{L-1}, \varphi_{L-1}) &= (f_L, \varphi_{L-1})_{L^2} \quad \forall \varphi_{L-1} \in V_{L-1}. \end{aligned}$$

To the function $v_L \in V_L$, we associate a function $v \in V \cap H^2(\Omega)$ as solution of the "continuous" boundary value problem

$$Lv = f_L \ \text{ in } \Omega, \quad v = 0 \ \text{ on } \partial\Omega, \tag{5.1.46}$$

or in "weak" formulation

$$a(v, \varphi) = (f_L, \varphi)_{L^2} \quad \forall \varphi \in V. \tag{5.1.47}$$

For this auxiliary problem, we have the following a priori estimate

$$\|\nabla^2 v\|_{L^2} \le c \|f_L\|_{L^2}. \tag{5.1.48}$$

There holds

$$\begin{aligned} a(v_L, \varphi_L) &= (f_L, \varphi_L)_{L^2} = a(v, \varphi_L), \quad \varphi_L \in V_L, \\ a(v_{L-1}, \varphi_{L-1}) &= (f_L, \varphi_{L-1})_{L^2} = a(v, \varphi_{L-1}), \quad \varphi_{L-1} \in V_{L-1}, \end{aligned}$$

i. e., $v_L$ and $v_{L-1}$ are the Ritz projections of $v$ into $V_L$ and $V_{L-1}$, respectively. For these the following $L^2$-error estimates hold true:

$$\|v_L - v\|_{L^2} \le c h_L^2 \|\nabla^2 v\|_{L^2}, \quad \|v_{L-1} - v\|_{L^2} \le c h_{L-1}^2 \|\nabla^2 v\|_{L^2}. \tag{5.1.49}$$

This together with the a priori estimate (5.1.48) and observing $h_{L-1} \le 4 h_L$ implies that

$$\|v_L - v_{L-1}\|_{L^2} \le c h_L^2 \|\nabla^2 v\|_{L^2} \le c h_L^2 \|f_L\|_{L^2}, \tag{5.1.50}$$

and, consequently,

$$\|\mathcal{A}_L^{-1} f_L - p_{L-1}^L \mathcal{A}_{L-1}^{-1} r_L^{L-1} f_L\|_{L^2} \le c h_L^2 \|f_L\|_{L^2}. \tag{5.1.51}$$

From this, we obtain the desired estimate

$$\|\mathcal{A}_L^{-1} - p_{L-1}^L \mathcal{A}_{L-1}^{-1} r_L^{L-1}\|_{L^2} \le c h_L^2, \tag{5.1.52}$$

which completes the proof. Q.E.D.

The foregoing result for the two-grid algorithm will now be used for inferring the convergence of the full multigrid method.

**Theorem 5.2 (Multigrid conver- gence):** *Suppose that the two-grid algorithm converges with rate* $\rho_{ZG}(\nu) \to 0$ *for* $\nu \to \infty$, *uniformly with respect to the mesh level* $L$. *Then, for sufficiently frequent smoothing the multigrid method with* $R \geq 2$ *(W-cycle) converges with rate* $\rho_{MG} < 1$ *independent of the mesh level* $L$,

$$\|u_L - \mathrm{MG}(L, u_L^t, f_L)\|_{L^2} \leq \rho_{\mathrm{MG}} \|u_L - u_L^t\|_{L^2}. \tag{5.1.53}$$

**Proof.** The proof is given by induction with respect to the mesh level $L$. We consider only the relevant case $R = 2$ (W-cycle) and, for simplicity, will not try to optimize the constants occurring in the course of the argument. Let $\nu$ be chosen sufficiently large such that the convergence rate of the two-grid algorithm is $\rho_{ZG} \leq 1/8$. We want to show that then the convergence rate of the full multigrid algorithm is $\rho_{MG} \leq 1/4$, uniformly with respect to the mesh level $L$. For $L = 1$ this is obviously fulfilled. Suppose now that also $\rho_{MG} \leq 1/4$ for mesh level $L-1$. Then, on mesh level $L$, starting from the iterate $u_L^t$, with the approximative solution $q_{L-1}^2$ (after 2-fold application of the coarse-mesh correction) and the exact solution $\hat{q}_{L-1}$ of the defect equation on mesh level $L-1$, there holds

$$u_L^{t+1} = \mathrm{MG}(L, u_L^t, f_L) = ZG(L, u_L^t, f_L) + p_{L-1}^L(q_{L-1}^2 - \hat{q}_{L-1}). \tag{5.1.54}$$

According to the induction assumption (observing that the starting value of the multigrid iteration on mesh level $L-1$ is zero and that $\hat{\rho}_{L-1} = \mathcal{A}_{L-1}^{-1} r_L^{L-1} d_L$) it follows that

$$\|\hat{q}_{L-1} - q_{L-1}^2\|_{L^2} \leq \rho_{\mathrm{MG}}^2 \|\hat{q}_{L-1}\|_{L^2} = \rho_{\mathrm{MG}}^2 \|\mathcal{A}_{L-1}^{-1} r_L^{L-1} \mathcal{A}_L \mathcal{S}_L^\nu (u_L - u_L^t)\|_{L^2}. \tag{5.1.55}$$

Combination of the last two relations implies for the iteration error $e_L^t := u_L^t - u_L$ that

$$\|e_L^{t+1}\|_{L^2} \leq \left(\rho_{\mathrm{ZG}} + \rho_{\mathrm{MG}}^2 \|\mathcal{A}_{L-1}^{-1} r_L^{L-1} \mathcal{A}_L \mathcal{S}_L^\nu\|_{L^2}\right) \|e_L^t\|_{L^2}. \tag{5.1.56}$$

The norm on the right-hand side has been estimated already in connection with the convergence analysis of the two-grid algorithm. Recalling the two-grid operator

$$\mathrm{ZG}_L = (\mathcal{A}_L^{-1} - p_{L-1}^L \mathcal{A}_{L-1}^{-1} r_L^{L-1}) \mathcal{A}_L \mathcal{S}_L^\nu = S_L^\nu - p_L^{L-1} \mathcal{A}_{L-1}^{-1} r_L^{L-1} \mathcal{A}_L \mathcal{S}_L^\nu,$$

there holds

$$\mathcal{A}_{L-1}^{-1} r_L^{L-1} \mathcal{A}_L \mathcal{S}_L^\nu = S_L^\nu - \mathrm{ZG}_L,$$

und, consequently,

$$\|\mathcal{A}_{L-1}^{-1} r_L^{L-1} \mathcal{A}_L \mathcal{S}_L^\nu\|_{L^2} \leq \|S_L^\nu\|_{L^2} + \|\mathrm{ZG}_L\|_{L^2} \leq 1 + \rho_{\mathrm{ZG}} \leq 2. \tag{5.1.57}$$

This eventually implies

$$\|e_L^{t+1}\|_{L^2} \leq \left(\rho_{\mathrm{ZG}} + 2\rho_{\mathrm{MG}}^2\right) \|e_L^t\|_{L^2}. \tag{5.1.58}$$

By the assumption on $\rho_{ZG}$ and the induction assumption, we conclude

$$\|e_L^{t+1}\|_{L^2} \leq \left(\tfrac{1}{8} + 2\tfrac{1}{16}\right) \|e_L^t\|_{L^2} \leq \tfrac{1}{4}\|e_L^t\|_{L^2}, \tag{5.1.59}$$

which completes the proof. Q.E.D.

**Remark 5.4:** For well-conditioned problems (symmetric, positive definite operator, regular coefficients, quasi-uniform meshes, etc.) one achieves multigrid convergence rates in the range $\rho_{MG} = 0,05 - 0,5$. The above analysis only applies to the W-cycle since in part (ii), we need that $R \geq 2$. The V-cycle cannot be treated on the basis of the two-grid analysis. In the literature there are more general approaches, which allow to prove convergence of multigrid methods under much weaker conditions.

Next, we analyze the computational complexity of the full multigrid algorithm. For this, we introduce the following notation:

$$\begin{aligned}
\mathrm{OP}(T) &= \text{number of a. op. for performing the operation } T,\\
R &= \text{number of defect-correction steps on the different mesh levels},\\
n_l &= \dim V_l \approx h_l^{-d} \ (d = \text{space dimension}),\\
\kappa &= \max_{1\leq l\leq L} n_{l-1}/n_l < 1,\\
C_0 &= \mathrm{OP}(\mathcal{A}_0^{-1})/n_0,\\
C_s &= \max_{1\leq l\leq L}\{\mathrm{OP}(\mathcal{S}_l)/n_l\}, \quad C_d = \max_{1\leq l\leq L}\{\mathrm{OP}(d_l)/n_l\},\\
C_r &= \max_{1\leq l\leq L}\{\mathrm{OP}(r_l)/n_l\}, \quad C_p = \max_{1\leq l\leq L}\{\mathrm{OP}(p_l)/n_l\}.
\end{aligned}$$

In practice mostly $\kappa \approx 2^{-d}$, $C_s \approx C_d \approx C_r \approx C_p \approx \#\{a_{nm} \neq 0\}$ and $C_0 n_0 \ll n_L$.

**Theorem 5.3 (Multigrid complexity):** *Under the condition $q := R\kappa < 1$, for the full multigrid cycle $\mathrm{MG}_L$ there holds*

$$\mathrm{OP}(\mathrm{MG}_L) \leq C_L n_L, \tag{5.1.60}$$

*where*

$$C_L = \frac{(\nu+\mu)C_s + C_d + C_r + C_p}{1-q} + C_0 q^L.$$

*The multigrid algorithm for approximating the $n_L$-dimensional discrete solution $u_L \in V_L$ on the finest mesh $\mathbb{T}_L$ within discretization accuracy $\mathcal{O}(h_L^2)$ requires $\mathcal{O}(n_L \ln(n_L))$ a. op., and therefore has (almost) optimal complexity.*

**Proof.** We set $C_l := \mathrm{OP}(\mathrm{MG}_l)/n_l$. One multigrid cycle contains the $R$-fold application of the same algorithm on the next coarser mesh. Observing $n_{l-1} \leq \kappa n_l$ and setting $\hat{C} := (\nu+\mu)C_s + C_d + C_r + C_p$ it follows that

$$C_L n_L = \mathrm{OP}(\mathrm{MG}_L) \leq \hat{C} n_L + R \cdot \mathrm{OP}(\mathrm{MG}_{L-1}) = \hat{C} n_L + R \cdot C_{L-1} n_{L-1} \leq \hat{C} n_L + q C_{L-1} n_L,$$

and consequently $C_L \le \hat{C} + qC_{L-1}$. Recursive use of this relation yields

$$\begin{aligned} C_L &\le \hat{C} + q\big(\hat{C} + qC_{L-2}\big) = \hat{C}(1+q) + q^2 C_{L-1} \\ &\vdots \\ &\le \hat{C}(1 + q + q^2 + ... + q^{L-1}) + q^L C_0 \le \frac{\hat{C}}{1-q} + q^L C_0. \end{aligned}$$

This implies the asserted estimate (5.1.60). The total complexity of the algotrithm then results from the relations

$$\rho_{\mathrm{MG}}^t \approx h_L^2 \approx n_L^{-2/d}, \qquad t \approx -\frac{\ln(n_L)}{\ln(\rho_{\mathrm{MG}})}.$$

The proof is complete. Q.E.D.

It should be emphasized that in the proof of (5.1.60) the assumption

$$q := R\kappa = R \max_{1 \le l \le L} n_{l-1}/n_l < 1 \tag{5.1.61}$$

is essential. This means for the W-cycle ($R = 2$) that by the transition from mesh $\mathbb{T}_{l-1}$ to the next finer mesh $\mathbb{T}_l$ the number of mesh points (dimension of spaces) sufficiently increases, comparibly to the situation of uniform mesh refinement,

$$n_l \approx 4n_{l-1}. \tag{5.1.62}$$

**Remark 5.5:** In an adaptively driven mesh refinement process with only local mesh refinement the condition (5.1.61) is usually not satisfied. Mostly only $n_l \approx 2n_{l-1}$ can be expected. In such a case the multigrid process needs to be modified in order to preserve good efficiency. This may be accomplished by applying the cost-intensive smoothing only to those mesh points, which have been newly created by the transition from mesh $\mathbb{T}_{l-1}$ to mesh $\mathbb{T}_l$. The implementation of a multigrid algorithm on locally refined meshes requires much care in order to achieve optimal complexity of the overall algorithm.

The nested multigrid algorithm turns out to be really complexity optimal, as it requires only $O(n_L)$ a. op. for producing a sufficiently accurate approximation to the discrete solution $u_L \in V_L$.

**Theorem 5.4 (Nested multigrid):** *The nested multigrid algorithm is of optimal complexity, i. e., it delivers an approximation to the discrete solution $u_L \in V_L$ on the finest mesh $\mathbb{T}_L$ with discretization accuracy $\mathcal{O}(h_L^2)$ with complexity $\mathcal{O}(n_L)$ a. op. independent of the mesh level $L$.*

**Proof.** The accuracy requirement for the multigrid algorithm on mesh level $\mathbb{T}_L$ is

$$\|e_L^t\|_{L^2} \le \hat{c} h_L^2 \|f\|_{L^2}. \tag{5.1.63}$$

i) First, we want to show that, under the assumptions of Theorem 5.2, the result (5.1.63) is achievable by the nested multigrid algorithm on each mesh level $L$ by using a fixed (sufficiently large) number $t_*$ of multigrid cycles. Let $e_L^t := u_L^t - u_L$ be again the iteration error on mesh level $L$. By assumption $e_0^t = 0$, $t \geq 1$. In case $u_L^0 := u_{L-1}^t$ there holds

$$\begin{aligned}\|e_L^t\|_{L^2} &\leq \rho_{\mathrm{MG}}^t \|e_L^0\|_{L^2} = \rho_{\mathrm{MG}}^t \|u_{L-1}^t - u_L\|_{L^2} \\ &\leq \rho_{\mathrm{MG}}^t \big(\|u_{L-1}^t - u_{L-1}\|_{L^2} + \|u_{L-1} - u\|_{L^2} + \|u - u_L\|_{L^2}\big) \\ &\leq \rho_{\mathrm{MG}}^t \big(\|e_{L-1}^t\|_{L^2} + ch_L^2 \|f\|_{L^2}\big).\end{aligned}$$

Recursive use of this relation for $L \geq l \geq 1$ then yields (observing $h_l \leq \kappa^{l-L} h_L$)

$$\begin{aligned}\|e_L^t\|_{L^2} &\leq \rho_{\mathrm{MG}}^t \big(\rho_{\mathrm{MG}}^t big(\|e_{L-2}^t\|_{L^2} + ch_{L-1}^2 \|f\|_{L^2}\big) + ch_L^2 \|f\|_{L^2}\big) \\ &\quad \vdots \\ &\leq \rho_{\mathrm{MG}}^{Lt} \|e_0^t\|_{L^2} + \big(c\rho_{\mathrm{MG}}^t h_L^2 + c\rho_{\mathrm{MG}}^{2t} h_{L-1}^2 + \ldots + c\rho_{\mathrm{MG}}^{Lt} h_1^2\big) \|f\|_{L^2} \\ &= ch_L^2 \kappa^2 \left(\rho_{\mathrm{MG}}^t \kappa^{-2\cdot 1} + \rho_{\mathrm{MG}}^{2t} \kappa^{-2\cdot 2} + \ldots + \rho_{\mathrm{MG}}^{Lt} \kappa^{-2L}\right) \|f\|_{L^2} \\ &\leq ch_L^2 \kappa^2 \|f\|_{L^2} \, \frac{\kappa^{-2} \rho_{\mathrm{MG}}^t}{1 - \kappa^{-2} \rho_{\mathrm{MG}}^t},\end{aligned}$$

provided that $\kappa^{-2} \rho_{\mathrm{MG}}^t < 1$. Obviously there exists a $t_*$, such that (5.1.63) is satisfied for $t \geq t_*$ uniformly with respect to $L$.

ii) We can now carry out the complexity analysis. Theorem 5.3 states that one cycle of the simple multigrid algorithm $\mathrm{MG}(l, \cdot, \cdot)$ on the $l$-th mesh level requires $W_l \leq c_* n_l$ a. op. (uniformly with respect to $l$). Let now $\hat{W}_l$ be the number of a. op. of the nested multigrid algorithm on mesh level $l$. Then, by construction there holds

$$\hat{W}_L \leq \hat{W}_{L-1} + t_* W_L \leq \hat{W}_{L-1} + t_* c_* n_L.$$

Iterating this relation, we obtain with $\kappa := \max_{1 \leq l \leq L} n_{l-1}/n_l < 1$ that

$$\begin{aligned}\hat{W}_L &\leq \hat{W}_{L-1} + t_* c_* n_L \leq \hat{W}_{L-2} + t_* c_* n_{L-1} + t_* c_* n_L \\ &\quad \vdots \\ &\leq t_* c_* \{n_L + \ldots + n_0\} \leq c t_* c_* n_L \{1 + \ldots + \kappa^L\} \leq \frac{c t_* c_*}{1 - \kappa} n_L,\end{aligned}$$

what was to be shown. Q.E.D.

## 5.2 Multigrid methods for eigenvalue problems (a short review)

The application of the "multigrid concept" to the solution of high-dimensional eigenvalue problems can follow different pathes. First, there is the possibility of using it directly for the eigenvalue problem based on its reformulation as a nonlinear system of equations, which allows for the formation of "residuals". Second, the multigrid concept may be

used as components of other iterative methods, such as the Krylov space methods, for accelerating certain computation-intensive substeps. In the following, we will only briefly describe these different approaches.

### 5.2.1 Direct multigrid approach

The algebraic eigenvalue problem

$$Az = \lambda z, \quad \lambda \in \mathbb{C},\ z \in \mathbb{C}^n,\ \ \|z\|_2 = 1, \tag{5.2.64}$$

is equivalent to the following nonlinear system of equations

$$\left\{ \begin{array}{c} Az - \lambda z \\ \|z\|_2^2 - 1 \end{array} \right\} = 0. \tag{5.2.65}$$

To this system, we may apply a nonlinear version of the multigrid method described in Section 5.1 again yielding an algorithm of optimal complexity, at least in principly (for details see, e. g., Brand et al. [27] and Hackbusch [37]). However, this approach suffers from stability problems in case of irregularities of the underlying continuous problem, such as anisotropies in the operator, the domain or the computational mesh, which may spoil the convergence of the method or render it inefficient. One cause may be the lack of approximation property in case that the continuous eigenvalue problem is not well approximated on coarser meshes, which is essential for the convergence of the multigrid method. The possibility of such a pathological situation is illustrated by the following example, which suggests to use the multigrid concept not directly but rather for accelerating the cost-intensive components of other more robust methods such as the Krylov space methods (or the Jacobi-Davidson method) described above.

**Example 5.1:** We consider the following non-symmetric convection-diffusion eigenvalue problem on the unit square $\Omega = (0,1)^2 \in \mathbb{R}^2$:

$$-\nu\Delta u + b \cdot \nabla u = \lambda u, \ \text{ in } \Omega, \quad u = 0, \ \text{ on } \partial\Omega, \tag{5.2.66}$$

with coefficients $\nu > 0$ and $c = (c_1, c_2) \in \mathbb{R}^2$. The (real) eigenvalues are explicitly given by

$$\lambda = \frac{b_1^2 + b_2^2}{4\nu} + \nu\pi^2(n_1^2 + n_2^2), \quad n_1, n_2 \in \mathbb{N},$$

with corresponding (non-normalized) eigenfunctions

$$w(x_1, x_2) = \exp\Big(\frac{b_1 x_1 + b_2 x_2}{2\nu}\Big) \sin(n_1\pi x_1) \sin(n_2\pi x_2).$$

The corresponding adjoint eigenvalue problem has the eigenfunctions

$$w^*(x_1, x_2) = \exp\Big(-\frac{b_1 x_1 + b_2 x_2}{2\nu}\Big) \sin(n_1\pi x_1) \sin(n_2\pi x_2).$$

This shows, first, that the underlying differential operator in (5.2.66) is non-normal and secondly that the eigenfunctions develop strong boundary layers for small parameter values $\nu$ (transport-dominant case). In particular, the eigenvalues depend very strongly on $\nu$. For the "direct" application of the multigrid method to this problem, this means that the "coarse-grid problems", due to insufficient mesh resolution, have completely different eigenvalues than the "fine-grid" problem, leading to insufficient approximation for computing meaningful corrections. This renders the multigrid iteration, being based on successive smoothing and coarse-grid correction, inefficient and may even completely spoil convergence.

### 5.2.2 Accelerated Arnoldi and Lanczos method

The most computation-intensive part of the Arnoldi and Lanczos methods in the case of the approximation of the smallest (by modulus) eigenvalues of a high-dimensional matrix $A \in \mathbb{K}^{n\times n}$ is the generation of the Krylov space

$$K_m = \operatorname{span}\{q, A^{-1}q, \ldots, (A^{-1})^{m-1}q\},$$

which requires the solution of a small number $m \ll n$ but high-dimensional linear systems with $A$ as coefficient matrix. Even though the Krylov space does not need to be explicitly constructed in the course of the modified Gram-Schmidt algorithm for the generation of an orthonormal basis $\{q^1, \ldots, q^m\}$ of $K_m$, this process requires the same amount of computation. This computational "acceleration" by use of multigrid techniques is exploited in Section 4.3.2 on the computation of pseudospectra. We want to illustrate this for the simpler situation of the "inverse iteration" for computing the smallest eigenvalue of a symmetric, positive definite matrix $A \in \mathbb{R}^{n\times n}$.

Recall Example 4.1 in Section 4.1.1, the eigenvalue problem of the Laplace operator on the unit square:

$$\begin{aligned} -\frac{\partial^2 w}{\partial x^2}(x,y) - \frac{\partial^2 w}{\partial y^2}(x,y) &= \mu w(x,y) \quad \text{for } (x,y) \in \Omega, \\ w(x,y) &= 0 \quad \text{for } (x,y) \in \partial\Omega. \end{aligned} \tag{5.2.67}$$

The discretization of this eigenvalue problem by the 5-point difference scheme on a uniform Cartesian mesh or the related finite element method with piecewise linear trial functions leads to the matrix eigenvalue problem

$$Az = \lambda z, \quad \lambda = h^2\mu, \tag{5.2.68}$$

with the same block-tridiagonal matrix $A$ as occurring in the corresponding discretization of the boundary value problem discussed in Section 3.4. We are interested in the smallest eigenvalue $\lambda_1 = \lambda_{\min}$ of $A$, which by $h^{-2}\lambda_{\min} \approx \mu_{\min}$ yields an approximation to the smallest eigenvalue of problem (5.2.67). For $\lambda_1$ and the next eigenvalue $\lambda_2 > \lambda_1$ there holds

$$\lambda_1 = 2\pi^2 h^2 + O(h^4), \quad \lambda_2 = 5\pi^2 h^2 + O(h^4).$$

For computing $\lambda_1$, we may use the inverse iteration with shift $\lambda = 0$. This requires in each step the solution of a problem like

$$A\tilde{z}^t = z^{t-1}, \quad z^t := \|\tilde{z}^t\|_2^{-1}\tilde{z}^t. \tag{5.2.69}$$

For the corresponding eigenvalue approximation

$$\frac{1}{\lambda_1^t} := \frac{(A^{-1}z^t, z^t)_2}{\|z^t\|_2^2} = (\tilde{z}^{t+1}, z^t)_2, \tag{5.2.70}$$

there holds the error estimate (see exercise in Section 4.1.1)

$$\Big|\frac{1}{\lambda_1^t} - \frac{1}{\lambda_1}\Big| \le \Big|\frac{1}{\lambda_1}\Big| \frac{\|z^0\|_2^2}{|\alpha_1|^2}\Big(\frac{\lambda_2}{\lambda_1}\Big)^{2t}, \tag{5.2.71}$$

where $\alpha_1$ is the coefficient in the expansion of $z^0$ with respect to the eigenvector $w^1$. From this relation, we infer that

$$|\lambda_1 - \lambda_1^t| \le \lambda_1^t \frac{\|z^0\|_2^2}{|\alpha_1|^2}\Big(\frac{\lambda_2}{\lambda_1}\Big)^{2t}. \tag{5.2.72}$$

Observing that $\lambda_1^t \approx \lambda_1 \approx h^2$ and $h^2\|z^0\|_2^2 = h^2\sum_{i=1}^n |z_i^0|^2 \approx \|v^0\|_{L^2}^2$, where $v^0 \in H_0^1(\Omega)$ is the continuous eigenfunction corresponding to the eigenvector $z^0$, we obtain

$$|\lambda_1 - \lambda_1^t| \le c\Big(\frac{\lambda_2}{\lambda_1}\Big)^{2t} \le c\,0.4^{2t}. \tag{5.2.73}$$

i. e., the convergence is independent of the mesh size $h$ or the dimension $n = m^2 \approx h^{-2}$ of $A$. However, in view of the relation $\mu_1 = h^{-2}\lambda_1$ achieving a prescribed accuracy in the approximation of $\mu_1$ requires the scaling of the tolerance in computing $\lambda_1$ by a factor $h^2$, which introduces a logarithmic $h$-dependence in the work count of the algorithm,

$$t(\varepsilon) \approx \frac{\log(\varepsilon h^2)}{\log(2/5)} \approx \log(n). \tag{5.2.74}$$

Now, using a multigrid solver of optimal complexity $\mathcal{O}(n)$ in each iteration step (4.1.20) the total complexity of computing the smallest eigenvalue $\lambda_1$ becomes $\mathcal{O}(n\log(n))$.

**Remark 5.6:** For the systematic use of multigrid acceleration within the Jacobi-Davidson method for nonsymmetric eigenvalue problems, we refer to Heuveline & Bertsch [41]. This combination of a robust iteration and multigrid acceleration seems presently to be the most efficient approach to solving large-scale symmetric or unsymmetric eigenvalue problems.

## 5.3 Exercises

**Exercise 5.1:** Consider the discretization of the Poisson problem

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

on the unit square $\Omega = (0,1)^2 \subset \mathbb{R}^2$ by the finite element Galerkin method using linear shape and test functions on a uniform Cartesian triangulation $\mathbb{T}_h = \{K\}$ with cells $K$ (rectangular triangles) of width $h > 0$. The lowest-order finite element space on the mesh $\mathbb{T}_h$ is given by

$$V_h = \{v_h \in C(\bar{\Omega}) \mid v_h|_K \in P_1(K), \ K \in \mathbb{T}_h, \ v_h|_{\partial\Omega} = 0\}.$$

Its dimension is $\dim V_h = n_h$, which coincides with the number of interior nodal points $a_i$, $i = 1, \ldots, n_h$, of the mesh $\mathbb{T}_h$. Let $\{\varphi_h^1, \ldots, \varphi_h^{n_h}\}$ denote the usual "nodal basis" (so-called "Lagrange basis") of the finite element subspace $V_h$ defined by the interpolation condition $\varphi_h^i(a_j) = \delta_{ij}$. Make a sketch of this situation, especially of the mesh $\mathbb{T}_h$ and a nodal basis function $\varphi_h^i$.

Then, the finite element Galerkin approximation in the space $V_h$ as described in the text results in the following linear system for the nodal value vector $x_h = (x_h^1, \ldots, x_h^{n_h})$:

$$A_h x_h = b_h,$$

with the matrix $A_h = (a_{ij})_{i,j=1}^{n_h}$ and right-hand side vector $b_h = (b_i)_{i=1}^{n_h}$ given by $a_{ij} = (\nabla\varphi_h^j, \nabla\varphi_h^i)_{L^2}$ and $b_i = (f, \varphi_h^i)_{L^2}$. Evaluate these elements $a_{ij}$ and $b_i$ using the trapezoidal rule for triangles

$$\int_K w(x)\, dx \approx \frac{|K|}{3} \sum_{i=1}^{3} w(a_i),$$

where $a_i$, $i = 1, 2, 3$, are the three vertices of the triangle $K$ and $|K|$ its area. This quadrature rule is exact for linear polynomials. The result is a matrix and right-hand side vector which are exactly the same as resulting from the finite difference discretization of the Poisson problem on the mesh $\mathbb{T}_h$ described in the text.

**Exercise 5.2:** Analyze the proof for the convergence of the two-grid algorithm given in the text for its possible extension to the case the restriction $r_l^{l-1} : V_l \to V_{l-1}$ is defined by local bilinear interpolation rather than by global $L^2$-projection onto the coarser mesh $\mathbb{T}_{l-1}$. What is the resulting difficulty? Do you have an idea how to get around it?

**Exercise 5.3:** The FE-discretization of the convection-diffusion problem

$$-\Delta u + \partial_1 u = f \quad \text{in } \Omega, \quad u = 0 \ \text{ on } \partial\Omega,$$

leads to asymmetric system matrices $A_h$. In this case the analysis of the multigrid algorithm requires some modifications. Try to extend the proof given in the text for the

convergence of the two-grid algorithm for this case if again the (damped) Richardson iteration is chosen as smoother,

$$x_h^{t+1} = x_h^t - \theta_t(A_h x_h^t - b_h), \quad t = 0, 1, 2, \ldots .$$

What is the resulting difficulty and how can one get around it?

**Exercise 5.4:** Consider the discretization of the Poisson problem

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

on the unit square $\Omega = (0,1)^2 \subset \mathbb{R}^2$ by the finite element Galerkin method using linear shape and test functions. Let $(\mathbb{T}_l)_{l\geq 0}$ be a sequence of equidistant Cartesian meshes of width $h_l = 2^{-l}$. The discrete equations on mesh level $l$ are solved by a multigrid method with (damped) Richardson smooting and the natural embedding as prolongation and the $L^2$ projection as restriction. The number of pre- and postsmoothing steps is $\nu = 2$ and $\mu = 0$, respectively. How many arithmetic operations are approximately required for a V-cycle and a W-cycle depending on the dimension $n_l = \dim V_l$?

### 5.3.1 General exercises

**Exercise 5.5:** Give short answers to the following questions:

a) When is a matrix $A \in \mathbb{R}^{n\times n}$ called "diagonalizable"?

b) When is a matrix $A \in \mathbb{R}^{n\times n}$ called "diagonally dominant"?

c) What is a "normal" matrix and is a Hermitian matrix always "normal"?

d) What is the relation between the "power method" and the "inverse iteration"?

e) What is the Rayleigh quotient of a Hermitian matrix $A \in \mathbb{C}^{n\times n}$ and a given vector $v \in \mathbb{C}^n \setminus \{0\}$. What is it used for?

f) What is the spectral condition number of a matrix $A \in \mathbb{C}^{n\times n}$?

g) What is a "Gerschgorin circle"?

h) What is the use of "restriction" and "prolongation" within the multigrid method?

i) What is a "Krylov space" $K_m$ corresponding to a matrix $A \in \mathbb{C}^{n\times n}$?

j) What does the adjective "damped" refer to within the Richardson method?

k) What is the difference between the "classical" and the "modified" Gram-Schmidt algorithm for orthonormalization?

**Exercise 5.6:** Consider the following matrices:

$$A_1 = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}, \qquad A_2 = \begin{bmatrix} 2 & -1 & 1 \\ -1 & 2 & -1 \\ 1 & -1 & 2 \end{bmatrix}, \qquad A_3 = \begin{bmatrix} 2 & -1 & 1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix}.$$

For which of these matrices do the Jacobi method, the Gauß-Seidel method and the CG method converge unconditionally for any given initial point $u^0$?

**Exercise 5.7:** Derive best possible estimates for the eigenvalues of the matrix

$$A = \begin{bmatrix} 1 & 10^{-3} & 10^{-4} \\ 10^{-3} & 2 & 10^{-3} \\ 10^{-4} & 10^{-3} & 3 \end{bmatrix}$$

by the enclusion lemma of Gerschgorin. (Hint: Precondition the matrix by scaling, i. e., by using an appropriate similarity transformation with a diagonal matrix $A \to D^{-1}AD$.)

**Exercise 5.8:** Formulate the power method for computing the largest (by modulus) eigenvalue of a matrix $A \in \mathbb{C}^{n\times n}$. Distinguish between the case of a general matrix and the special case of a Hermitian matrix.

i) Under which conditions is convergence guaranteed?

ii) Which of these conditions is the most critical one?

iii) State an estimate for the convergence speed.