In: Numerical Methods for Advection-Diffusion Problems, C.B. Vreugdenhil en B. Koren (eds.), Vieweg Verlag, Braunschweig, 1993, 321-334.

Chapter 13: Introduction to Multi-Grid

Rob Stevenson
Department of Mathematics, University of Utrecht, Budapestlaan 6, P.O. Box 80.010, 3508 TA Utrecht, The Netherlands.

13.1 Introduction

In this chapter we will give an introduction to the multi-grid method (MGM). The MGM, developed in the sixties and seventies, is an efficient iterative method for solving systems of equations that result from discretizing partial differential equations (PDEs).

We will restrict ourselves to discretizations of elliptic PDEs. Note, however, that an implicit time discretization of an advection-diffusion problem also leads to a sequence of elliptic problems.

Although, so far there is no rigorous theoretical foundation, multi-grid is also successfully applied to hyperbolic or mixed hyperbolic and elliptic equations such as those that arise from complicated flow problems. We refer to [2, 9] and the references cited there. The use of multiple "grids" can even be helpful to solve systems of equations without any geometric background ("algebraic multi-grid") (cf. [5]).

13.2 The model problem

The multi-grid principles are most easily explained by considering a model problem. We consider the symmetric finite difference discretization of

$$\begin{cases} -u'' = f \quad \text{on } (0,1) \\ u(0) = u(1) = 0 \end{cases} \tag{13.1}$$

on the grid $2^{-(l+1)}\mathbf{N} \cap (0,1)$ for some $l \in \mathbf{N}$. The resulting system of equations looks like

$$h_l^{-2} \begin{bmatrix} 2 & -1 & & \\ -1 & 2 & -1 & \\ & & \ddots & \\ & & -1 & 2 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{n_l} \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_{n_l} \end{bmatrix}, \tag{13.2}$$

where $h_l = 2^{-(l+1)}$, $n_l = h_l^{-1} - 1$, $f_i = f(ih_l)$ and u_i is an approximation of the solution u of (13.1) in the point ih_l.

We will identify $\mathbf{R}^{n_l}$ with the set of functions from $h_l\mathbf{N} \cap (0,1)$ to $\mathbf{R}$, that is, for $v \in \mathbf{R}^{n_l}$, we write $v(ih_l)$ instead of v_i. This notation allows expressing the dependence of (13.2) on the "discretization level" l by a subscript; we denote (13.2) as

$$A_l u_l = f_l. \tag{13.3}$$

Sometimes, this notation will also be used to indicate a general discretized (elliptic) boundary value problem. We equip $\mathbf{R}^{n_l}$ with (scaled) Euclidian scalar product

$$(v_l, w_l) = h_l \sum_{i=1}^{n_l} v_l(ih_l) w_l(ih_l) \tag{13.4}$$

and norm $\|v_l\| = (v_l, v_l)^{\frac{1}{2}}$.

Of course, the simple system (13.2) can be cheaply solved directly by making an LU-decomposition of A_l. Nevertheless, this system provides us with an example whereby we can explain the multi-grid principles, which can be applied far more generally, without becoming distracted by many technical details.

In the following section, we consider classical iterative methods for solving $A_l u_l = f_l$ such as the Jacobi method and the Gauss-Seidel method. It will appear that the number of iterations needed to solve the system within a given accuracy increases rapidly when the grid size h_l decreases. Because of this feature these methods are not feasible for solving large systems. However, most of these methods do share the property that, independently of h_l, the difference between the exact solution and the iterand becomes a "smooth" function after only a few iterations. This observation will lead us to the MGM in a natural way.

13.3 Defect correction

Let $\hat{u}_l$ be some approximation of the exact solution u_l of $A_l u_l = f_l$. Then the error $e_l := \hat{u}_l - u_l$ satisfies the defect equation

$$A_l e_l = d_l \tag{13.5}$$

with defect $d_l := (A_l(\hat{u}_l - u_l) =)\ A_l \hat{u}_l - f_l$. Instead of solving (13.5), which is as expensive as solving the original equation, we can improve our approximation $\hat{u}_l$ by correcting it with an "easily computable" approximation of e_l. So with B_l some cheap approximation of A_l^{-1}, we arrive at the (linear) iterative method

$$u_l^{(i+1)} = u_l^{(i)} - B_l(A_l u_l^{(i)} - f_l). \tag{13.6}$$

For the error $u_l^{(i)} - u_l$ we have the relation $u_l^{(i+1)} - u_l = M_l(u_l^{(i)} - u_l)$, with *error amplification operator* $M_l := I_l - B_l A_l$ (where I_l is the identity). Per iteration the norm of error is reduced (at least) by the *contraction number* $\|M_l\| = \sup_{e_l \neq 0} \|M_l e_l\| / \|e_l\|$. Because of $\lim_{k\to\infty} \|M_l^k\|^{1/k} = \rho(M_l)$, the spectral radius $\rho(M_l)$ is called the *(asymptotic) convergence rate*. An iterative method is convergent if and only if $\rho(M_l) < 1$.

Sometimes, we will denote the iteration (13.6) also as

$$u_l \leftarrow u_l - B_l(A_l u_l - f_l) \tag{13.7}$$

where u_l now should be understood as the current *approximation* of the exact solution.

Well-known examples of (linear) iterative methods are obtained by writing $A_l = D_l + L_l + U_l$, with D_l a diagonal matrix and L_l (U_l) a strictly lower (upper) triangular matrix, and taking $B_l = \omega D_l^{-1}$ (damped Jacobi method) or $B_l = (D_l + L_l)^{-1}$ (Gauss-Seidel method).

For ease of presentation, we will analyse the convergence behaviour of only the Jacobi method applied to our model problem (13.2). Using different techniques it has been verified that our observations hold for most of the classical iterative methods.

An (orthonormal) basis of eigenfunctions of A_l is given by $\{e_l^{(\mu)} : 1 \leq \mu \leq n_l\}$, defined by

$$e_l^{(\mu)}(ih_l) = \sqrt{2}\sin(\mu\pi i h_l)$$

($1 \leq i \leq n_l$) (see figure 13.1) with corresponding eigenvalues $\tau_l^{(\mu)} = 4h_l^{-2}\sin^2(\mu\pi h_l/2)$.

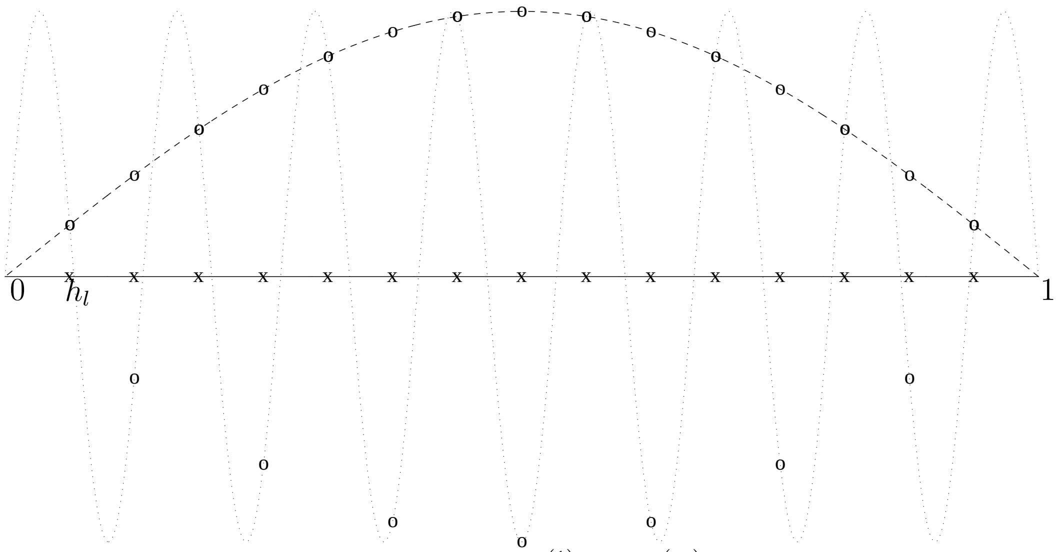


Figure 13.1: The eigenfunctions $e_l^{(1)}$ and $e_l^{(n_l)}$ of A_l in case $l = 3$.

Because of $\omega D_l^{-1} = \frac{1}{2}\omega h_l^2 I_l$, the eigenfunctions of A_l equal those of the error amplification operator $M_l^{(\mathrm{Jac})} = I_l - \omega D_l^{-1} A_l$. It holds that $M_l^{(\mathrm{Jac})} e_l^{(\mu)} = \lambda_l^{(\mu)}(\omega) e_l^{(\mu)}$, where $\lambda_l^{(\mu)}(\omega)$ $(= 1 - \frac{1}{2}\omega h_l^2 \tau_l^{(\mu)}) = 1 - 2\omega\sin^2(\mu\pi h_l/2)$. We conclude that

$$\|M_l^{(\mathrm{Jac})}(\omega)\| = \rho(M_l^{(\mathrm{Jac})}(\omega)) =$$
$$\max\left\{|\lambda_l^{(\mu)}(\omega)| : 1 \leq \mu \leq n_l\right\} = \begin{cases} \geq 1 & \text{if } \omega \notin (0,1] \text{ and } h_l \text{ is small enough} \\ \lambda_l^{(1)}(\omega) = 1 - \frac{\omega}{2}\pi^2 h_l^2 + \mathcal{O}(h_l^4) & \text{if } \omega \in (0,1] \end{cases}.$$

So, for $\omega \in (0,1]$, the damped Jacobi method is convergent. However, to ensure that an initial error $u_l^{(0)} - u_l$ is reduced by a factor ϵ, at least $\frac{\ln\epsilon}{\ln\lambda_l^{(1)}(\omega)} = -\frac{2\ln\epsilon}{\omega\pi^2}h_l^{-2} + \mathcal{O}(1)$ damped Jacobi iterations have to be performed; this number increases rapidly when h_l decreases.

A way to increase the speed of convergence of the method is to vary the parameter ω from step to step. Optimization of the choice of these ω's uniformly in h_l and $u_l^{(0)}$ gives the Chebyshev method. Optimization dependent on $u_l^{(0)}$ leads to the conjugate gradient method described in chapter 12 of this book. Also in the case of these two methods however, the number of iterations needed to reduce the error by some factor increases when the grid size h_l decreases.

13.4 Two-grid method

In this section, we discuss another improvement of our basic iterative method. In figure 13.2, for $\omega \in \{\frac{1}{2}, \frac{3}{4}, 1\}$ the eigenvalues $\lambda_l^{(\mu)}(\omega) = 1 - 2\omega\sin^2(\mu\pi h_l/2)$ of $M_l^{(\mathrm{Jac})}(\omega)$ are plotted as function of $\mu h_l \in [0,1]$ $(\supset \{h_l, 2h_l, \ldots, 1-h_l\})$. For $\omega < 1$, we see that the slow convergence is caused by eigenfunctions $e_l^{(\mu)}$ with μh_l close to 0, that is, by eigenfunctions that are "*smooth*". Oscillating errors of the form $\sum_{\mu h_l > \delta} c_l^{(\mu)} e_l^{(\mu)}$ are reduced

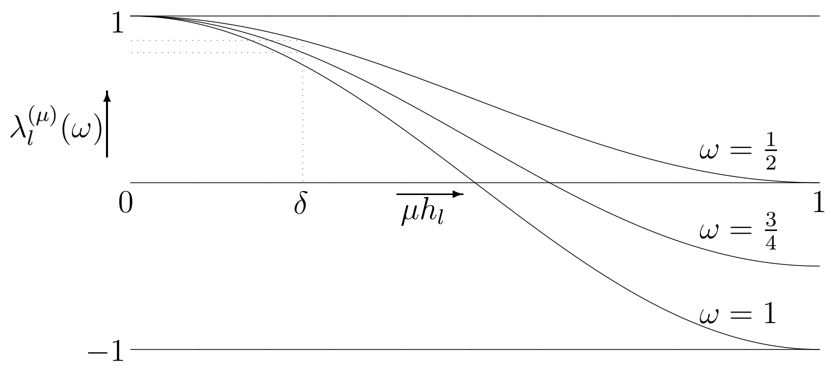


Figure 13.2: Eigenvalues of $M_l^{(\mathrm{Jac})}(\omega)$ as function of μh_l.

by a factor less than 1 *uniform in* h_l. Since the oscillatory components of a general error are well reduced, the error becomes a smooth function after only a few iterations. This is illustrated in figure 13.3. For this reason the damped Jacobi method with $\omega \in (0, 1)$

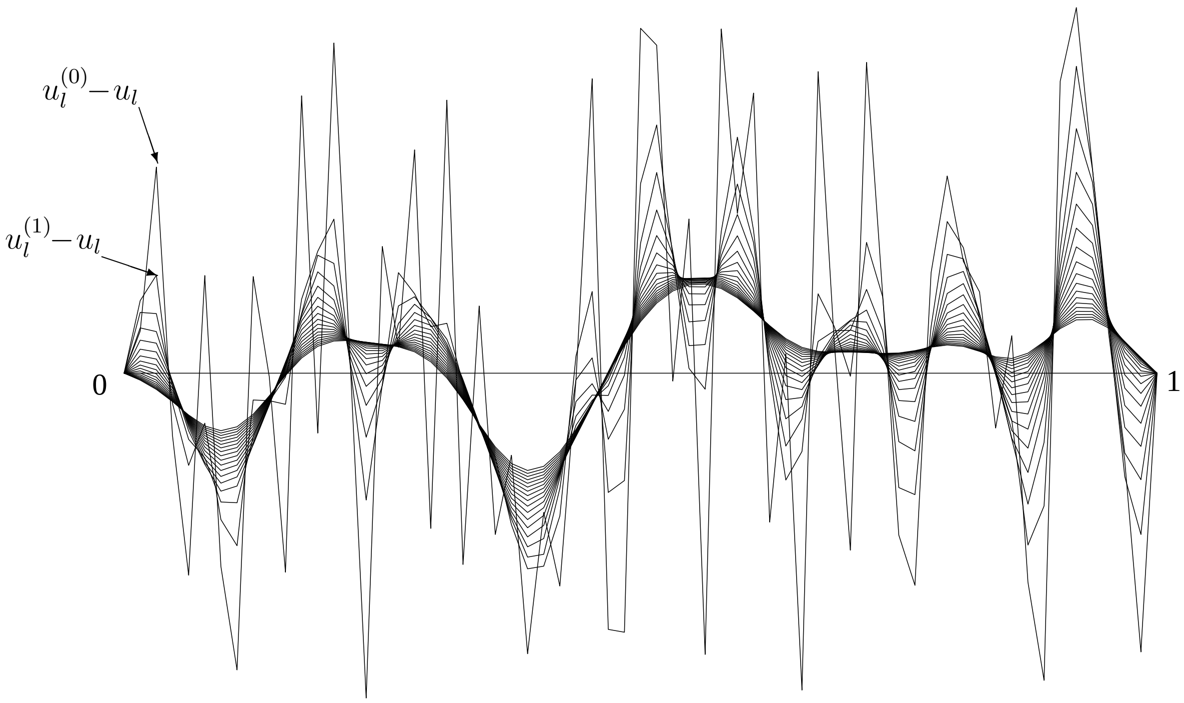
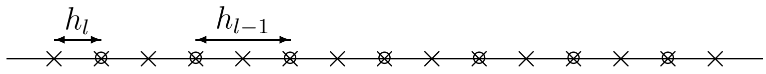


Figure 13.3: The evolution of the error $u_l^{(i)} - u_l$ ($i \in \{0, 1, \ldots, 17\}$) when applying the damped Jacobi method with $\omega = \frac{1}{2}$ and $l = 4$.

is called a *smoother* or a *relaxation method*.

To indicate that the underlying ideas are not restricted to the damped Jacobi method, in the sequel we denote an iteration of our smoother as $u_l \leftarrow \mathcal{S}_l(u_l, f_l)$ and its error amplification operator as S_l; these are the common notations for a general smoother in a multi-grid context.

A natural idea is to combine a smoother with a method designed to damp smooth errors: Suppose we have an approximation of the solution of $A_l u_l = f_l$ with a smooth error $e_l = \sum_{\mu h_l < \delta} c_l^{(\mu)} e_l^{(\mu)}$. Then also the defect d_l $(= A_l e_l) = \sum_{\mu h_l < \delta} \tau_l^{(\mu)} c_l^{(\mu)} e_l^{(\mu)}$ is a smooth function. Therefore we can expect that on the "coarse grid" $h_{l-1}\mathbf{N} \cap (0, 1)$ (see figure 13.4), the solution of $A_{l-1} e_{l-1} = d_{l-1}$ with $d_{l-1}(ih_{l-1}) := d_l(2ih_l)$ resembles e_l. Note that in (13.2), A_k is defined for all levels $k \geq 0$, in particular for $k = l - 1$. Again



×: fine grid point
○: coarse grid point

Figure 13.4: "Fine" grid $h_l\mathbf{N} \cap (0,1)$ and "coarse" grid $h_{l-1}\mathbf{N} \cap (0,1)$ in case $l = 3$.

by the smoothness of e_l, the linear interpolant $\frac{1}{2}(e_{l-1}(ih_{l-1}) + e_{l-1}((i+1)h_{l-1}))$ can be expected to give a good approximation of e_l in the "intermediate" point $(2i+1)h_l$.

So with *restriction* $r : \mathbf{R}^{n_l} \to \mathbf{R}^{n_{l-1}}$ and *prolongation* $p : \mathbf{R}^{n_{l-1}} \to \mathbf{R}^{n_l}$, defined by $(rv_l)(ih_{l-1}) = v_l(2ih_l)$ and

$$\begin{cases} (pv_{l-1})(2ih_l) &= v_{l-1}(ih_{l-1}) \\ (pv_{l-1})((2i+1)h_l) &= \frac{1}{2}\left(v_{l-1}(ih_{l-1}) + v_{l-1}((i+1)h_{l-1})\right) \end{cases}$$

(where we formally set $v_{l-1}(0) = v_{l-1}(1) = 0$), we arrive at the so-called coarse-grid correction (CGC)

$$u_l \leftarrow u_l - pA_{l-1}^{-1}r(A_l u_l - f_l). \tag{13.8}$$

Note that (13.8) is of the form (13.7) with $B_l = pA_{l-1}^{-1}r$. The error amplification operator is given by $M_l^{(\mathrm{CGC})} = I_l - pA_{l-1}^{-1}rA_l$.

Although the CGC is designed to reduce smooth errors, we do not want it to blow up any oscillating errors. However, as we will see, this is what happened with the above method: From $\sin((n_l+1-\mu)\pi ih_l) = -\sin(\mu\pi ih_l)$ when i is even, it follows that

$$-re_l^{(n_l+1-\mu)} = re_l^{(\mu)} = e_{l-1}^{(\mu)} \quad (1 \le \mu \le n_{l-1})$$

(this phenomenon is known under the name aliasing) and $re_l^{(n_{l-1}+1)} = 0$ (note that $n_l + 1 - (n_{l-1}+1) = n_{l-1}+1$). For the prolongation, one can check that

$$pe_{l-1}^{(\mu)} = \cos^2(\mu\pi h_l/2)e_l^{(\mu)} - \sin^2(\mu\pi h_l/2)e_l^{(n_l+1-\mu)} \quad (1 \le \mu \le n_{l-1}).$$

Therefore, with respect to the basis

$$\underbrace{e_l^{(1)}, e_l^{(n_l)}}, \underbrace{e_l^{(2)}, e_l^{(n_l-1)}}, \cdots, \underbrace{e_l^{(n_{l-1})}, e_l^{(n_{l-1}+2)}}, e_l^{(n_{l-1}+1)} \tag{13.9}$$

of $\mathbf{R}^{n_l}$, the matrix of $M_l^{(\mathrm{CGC})}$ is given by

$$\begin{bmatrix} \boxed{\hat{M}_l^{(\mathrm{CGC})}(1)} & & & \emptyset \\ & \ddots & & \\ & & \boxed{\hat{M}_l^{(\mathrm{CGC})}(n_{l-1})} & \\ \emptyset & & & \boxed{0} \end{bmatrix},$$

where

$$\hat{M}_l^{(\mathrm{CGC})}(\mu) = \left[\begin{array}{cc} 1 & 0 \\ 0 & 1 \end{array}\right] - \left[\begin{array}{c} \mathrm{c}^2 \\ -\mathrm{s}^2 \end{array}\right] (\tau_{l-1}^{(\mu)})^{-1} \left[\begin{array}{cc} 1 & -1 \end{array}\right] \left[\begin{array}{cc} \tau_l^{(\mu)} & 0 \\ 0 & \tau_l^{(n_l+1-\mu)} \end{array}\right]$$

$$= \left[\begin{array}{cc} 0 & \frac{\mathrm{c}^2}{\mathrm{s}^2} \\ \frac{\mathrm{s}^2}{\mathrm{c}^2} & 0 \end{array}\right] \qquad (1 \le \mu \le n_{l-1}) \tag{13.10}$$

with $\mathrm{c} = \cos(\mu\pi h_l/2)$ $(\in (\frac{1}{2}\sqrt{2}, 1))$ and $\mathrm{s} = \sin(\mu\pi h_l/2)$ $(\in (0, \frac{1}{2}\sqrt{2}))$. So a eigenfunction $e_l^{(\mu)}$ with $1 \le \mu \le n_{l-1}$ is mapped onto an eigenfunction $e_l^{(n_l+1-\mu)}$ and vice versa. By $\lim_{\mu h_l \downarrow 0} \frac{\mathrm{s}^2}{\mathrm{c}^2} = 0$, smooth eigenfunctions of A_l are well reduced in norm, but highly oscillating eigenfunctions are blown up.

The cause of this blowing up in norm of highly oscillating eigenfunctions can be found to be the mapping by r of $e_l^{(n_l+1-\mu)}$ onto $e_{l-1}^{(\mu)}$ without any reduction in norm. Clearly $e_{l-1}^{(\mu)}$ is not a good representation of $e_l^{(n_l+1-\mu)}$, but on the other hand such a representation does not exist on the coarse grid. It would therefore be better if oscillating functions were reduced in norm by the restriction so that at least the blowing up would be prevented. We therefore replace the trivial injection by the weighted restriction $r : \mathbf{R}^{n_l} \to \mathbf{R}^{n_{l-1}}$ defined by

$$(rv_l)(ih_{l-1}) = \tfrac{1}{4}v_l((2i-1)h_l) + \tfrac{1}{2}v_l(2ih_l) + \tfrac{1}{4}v_l((2i+1)h_l).$$

The weighted restriction can be seen as the injection preceded by an operator on the fine grid which hardly affects smooth functions but damps oscillating functions. In particular, it holds that

$$\begin{array}{rcl} re_l^{(\mu)} & = & \mathrm{c}^2 e_{l-1}^{(\mu)} \\ re_l^{(n_l+1-\mu)} & = & -\mathrm{s}^2 e_{l-1}^{(\mu)} \end{array} \quad (1 \le \mu \le n_{l-1}), \qquad re_l^{(n_{l-1}+1)} = 0$$

with which we now obtain

$$\hat{M}_l^{(\mathrm{CGC})}(\mu) = \left[\begin{array}{cc} \mathrm{s}^2 & \mathrm{c}^2 \\ \mathrm{s}^2 & \mathrm{c}^2 \end{array}\right] \quad \text{(cf. (13.10))}.$$

We conclude that with the weighted restriction the norm of an oscillating eigenfunction is increased by at most a factor $\sqrt{2}$ $(\sqrt{2\mathrm{c}^2} \in (1, \sqrt{2}))$, whereas smooth eigenfunctions are well reduced.

Direct computations show that for the weighted restriction we have the equalities

$$A_{l-1} = rA_lp \quad \text{(the so-called Galerkin approximation)} \tag{13.11}$$

and

$$r = p^* \tag{13.12}$$

with respect to our scalar products on $\mathbf{R}^{n_l}$ and $\mathbf{R}^{n_{l-1}}$ (cf. (13.4)). These relations can generally be used to *define* r and A_{l-1}. Indeed, for problems on 2- or 3-dimensional irregular domains it may be rather expensive to define an additional discretization on the coarse grid and a restriction. A consequence of (13.11) is that

$$\left(M_l^{(\mathrm{CGC})}\right)^2 = (I_l - pA_{l-1}^{-1}rA_l)(I_l - pA_{l-1}^{-1}rA_l) =$$
$$I_l - 2pA_{l-1}^{-1}rA_l + pA_{l-1}^{-1}rA_lpA_{l-1}^{-1}rA_l = I_l - pA_{l-1}^{-1}rA_l = M_l^{(\mathrm{CGC})},$$

that is, the CGC is a projection. Therefore it makes no sense to perform more than one consecutive CGCs.

We now have a method for damping oscillatory errors (a smoother) and the CGC for damping smooth errors. We combine these methods into one iterative method, the so-called two-grid method (TGM) which consists of successive ν_1 smoothing steps (the pre-smoothing steps) for some $\nu_1 \in \mathbf{N}$, a CGC and $\nu_2 \in \mathbf{N}$ post-smoothing steps:

procedure TGM(l, u_l, f_l)
    begin
        if $l = 0$ **then** $u_0 := A_0^{-1} f_0$ % no coarser grid available
        else
            for $i := 1$ **to** ν_1 **do** $u_l \leftarrow \mathcal{S}_l(u_l, f_l)$;

$$\left.\begin{array}{l} d_l := A_l u_l - f_l; \\ d_{l-1} := r d_l; \\ e_{l-1} := A_{l-1}^{-1} d_{l-1}; \\ u \leftarrow u - p e_{l-1}; \end{array}\right\} \text{ % or shortly } u_l \leftarrow u_l - p A_{l-1}^{-1} r (A_l u_l - f_l);$$

            for $i := 1$ **to** ν_2 **do** $u_l \leftarrow \mathcal{S}_l(u_l, f_l)$;
    end

The error amplification operator of the TGM is given by $M_l^{(\mathrm{TGM})} = S_l^{\nu_2} M_l^{(\mathrm{CGC})} S_l^{\nu_1}$.

To reduce the number of parameters, in the following analysis of this method in our model case we take $\nu_1 = \nu \in \mathbf{N}$, $\nu_2 = 0$ and damping parameter $\omega = \frac{1}{2}$. Recalling that $(M_l^{(\mathrm{Jac})} e_l^{(\mu)} =)\ S_l e_l^{(\mu)} = \mathrm{c}^2 e_l^{(\mu)}$ and thus $S_l e_l^{(n_l+1-\mu)} = \mathrm{s}^2 e_l^{(n_l+1-\mu)}$ ($\mathrm{c} = \cos(\mu\pi h_l/2)$, $\mathrm{s} = \sin(\mu\pi h_l/2)$), with respect to the basis (13.9) we have

$$M_l^{(\mathrm{TGM})} = \begin{bmatrix} \boxed{\hat{M}_l^{(\mathrm{TGM})}(1)} & & & \emptyset \\ & \ddots & & \\ & & \boxed{\hat{M}_l^{(\mathrm{TGM})}(n_{l-1})} & \\ \emptyset & & & \boxed{(\frac{1}{2})^\nu} \end{bmatrix}$$

with

$$\hat{M}_l^{(\mathrm{TGM})}(\mu) := \hat{M}_l^{(\mathrm{CGC})}(\mu) \begin{bmatrix} \mathrm{c}^2 & 0 \\ 0 & \mathrm{s}^2 \end{bmatrix}^\nu = \begin{bmatrix} \mathrm{s}^2 & \mathrm{c}^2 \\ \mathrm{s}^2 & \mathrm{c}^2 \end{bmatrix} \begin{bmatrix} \mathrm{c}^2 & 0 \\ 0 & \mathrm{s}^2 \end{bmatrix}^\nu .$$

From this we obtain for all $l \in \mathbf{N}$,

$$\begin{aligned} \rho\left(M_l^{(\mathrm{TGM})}\right) &= \max\left\{(\tfrac{1}{2})^\nu, \max_{1\le\mu\le n_{l-1}} \rho\left(\hat{M}_l^{(\mathrm{TGM})}(\mu)\right)\right\} \\ &= \max\left\{(\tfrac{1}{2})^\nu, \max_{1\le\mu\le n_{l-1}} \mathrm{s}^2(1-\mathrm{s}^2)^\nu + (1-\mathrm{s}^2)\mathrm{s}^{2\nu}\right\} \\ &\le \max\left\{\xi(1-\xi)^\nu + (1-\xi)\xi^\nu : \xi \in [0, \tfrac{1}{2}]\right\} =: \rho_\nu \end{aligned}$$

and

$$\begin{aligned} \|M_l^{(\mathrm{TGM})}\| &= \max\left\{(\tfrac{1}{2})^\nu, \max_{1\le\mu\le n_{l-1}} \rho\left(\hat{M}_l^{(\mathrm{TGM})}(\mu)\hat{M}_l^{(\mathrm{TGM})}(\mu)^{\mathrm{T}}\right)^{\frac{1}{2}}\right\} \\ &= \max\left\{(\tfrac{1}{2})^\nu, \max_{1\le\mu\le n_{l-1}} \left(2\left(\mathrm{s}^4(1-\mathrm{s}^2)^{2\nu} + (1-\mathrm{s}^2)^2\mathrm{s}^{4\nu}\right)\right)^{\frac{1}{2}}\right\} \\ &\le \max\left\{\left(2\left(\xi^2(1-\xi)^{2\nu} + (1-\xi)^2\xi^{2\nu}\right)\right)^{\frac{1}{2}} : \xi \in [0, \tfrac{1}{2}]\right\} =: \zeta_\nu . \end{aligned}$$

For each $\nu > 1$, ρ_ν and ζ_ν are less than 1. So in contrast to all classical iterative methods, the number of iterations needed to reduce the error by some factor is bounded *uniform in* h_l. It holds that $\lim_{\nu\to\infty} \nu\rho_\nu = \mathrm{e}^{-1}$ and $\lim_{\nu\to\infty} \nu\zeta_\nu = \mathrm{e}^{-1}\sqrt{2}$ (cf. [2, theorem 2.4.4]). Some values are given in the table below.

Table 13.1: Uniform bounds for the convergence rate and contraction number of the TGM.

ν	1	2	3	4	5
ρ_ν	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{8}$	.0832	.0671
ζ_ν	$\frac{1}{2}$	$\frac{1}{4}$	.150	.1159	.0947

Finally in this section, we make some comments on the convergence of a general TGM applied to some discretized elliptic boundary value problem. An analysis similar to our analysis of the model case can only be given for symmetric, constant coefficient boundary value problems on one- or multi-dimensional blocks which have been discretized on regular grids using symmetric formulas. However, using different techniques (cf. [2]), a contraction number less than 1 (and converging to zero when $\nu_1 + \nu_2 \to \infty$) *uniform in the grid size* can be shown for a much larger class of problems, including problems with variable coefficients, irregular domains, irregular grids and non-symmetric (advection-diffusion) problems.

Unfortunately, except in the case of some model problems, these techniques do not yield upperbounds for the contraction number which are sharp or even computable (for example, think of a bound of type $\frac{c}{1+c}$ with c some constant that not only depends on the TGM but also on the ellipticity of the boundary value problem, the smoothness of its coefficients and the shape of the boundary). Therefore, these techniques cannot be used for a (problem dependent) optimalization of the choice of the components of the TGM as there are smoother, prolongation, restriction, discretization on the coarse grid and the number of pre- and post-smoothing iterations. Then an alternative is to analyse the TGM "locally" in a way similar to our analysis of the model case. This technique, analogous to the Von Neumann stability analysis of time-dependent problems, is called *local mode analysis* in multi-grid context ([1]). Used with some care, local mode analysis is a valuable tool for making a good choice of the various components of the TGMs. Local mode analysis can be rigorously justified for TGMs that have been extended with "local smoothing steps along the boundary" (cf. [6]).

13.5 Multi-grid method

Although the TGM has turned out to be a, uniform in the grid size, fast converging iterative method, it is not an efficient method. In each iteration a system $A_{l-1}e_{l-1} = d_{l-1}$ has to be solved the costs of which are proportional to the costs of solving the original equation $A_l u_l = f_l$. For our model discretized 1D problem this is no drawback (nor is it a recommendation) since the original equation can be cheaply solved, but for discretizations of multi-dimensional boundary value problems it is! There, except in the case of very small systems the costs of a two-grid step are much larger than those of a Jacobi step.

To increase the efficiency of the method we note that it may not be necessary to solve the equation $A_{l-1}e_{l-1} = d_{l-1}$ $(= rd_l)$ exactly; since e_{l-1} is only used to supply the approximation pe_{l-1} of the solution of $A_l e_l = d_l$, a good approximation of e_{l-1} may suffice as well. A way to find such an approximation is to perform a couple (γ) of steps of an iterative method on the equation $A_{l-1}e_{l-1} = d_{l-1}$. A natural choice is the TGM but now on the grids $h_{l-1}\mathbf{N} \cap (0,1)$ and $h_{l-2}\mathbf{N} \cap (0,1)$ with, since e_l and thus e_{l-1} are hopefully small, initial approximation $e_{l-1}^{(0)} = 0$. Now when we encounter the problem of solving a system on the coarse grid $h_{l-2}\mathbf{N} \cap (0,1)$, we can apply the same idea recursively involving coarser and coarser grids until the coarsest grid $h_0\mathbf{N} \cap (0,1) = \{\frac{1}{2}\}$ is reached where the 1×1 system is solved directly. The iterative method described is the multi-grid method (MGM). A precise description is given by

procedure MGM(l, u_l, f_l)
    begin
        if $l = 0$ **then** $u_0 := A_0^{-1} f_0$ % no coarser grid available
        else
            for $i := 1$ **to** ν_1 **do** $u_l \leftarrow \mathcal{S}_l(u_l, f_l)$;
            $d_l := A_l u_l - f_l$;
            $d_{l-1} := r d_l$;
            $e_{l-1} := 0$; % these three lines replace the
            for $i := 1$ **to** γ **do** % statement $e_{l-1} := A_{l-1}^{-1} d_{l-1}$
                $e_{l-1} \leftarrow \mathrm{MGM}(l-1, e_{l-1}, d_{l-1})$; % in the two-grid algorithm
            $u_l \leftarrow u_l - p e_{l-1}$;
            for $i := 1$ **to** ν_2 **do** $u_l \leftarrow \mathcal{S}_l(u_l, f_l)$;
    end

For the most common choices $\gamma = 1$ and $\gamma = 2$, figures 13.5 and 13.6 give a schematic picture of the order in which the grids are visited. Due to the form of these diagrams the MGM with $\gamma = 1$ or $\gamma = 2$ is usually called V- or W-cycle respectively. The V-cycle without pre-smoothing steps depicted in figure 13.7 is called a sawtooth-cycle.

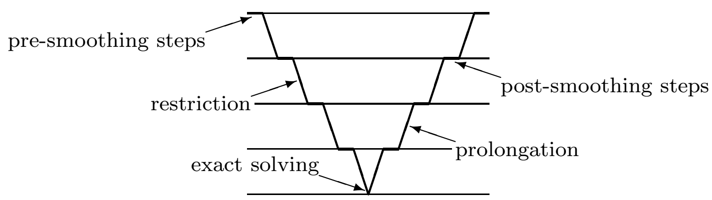


Figure 13.5: One iteration of the MGM with $\gamma = 1$ on level $l = 4$.

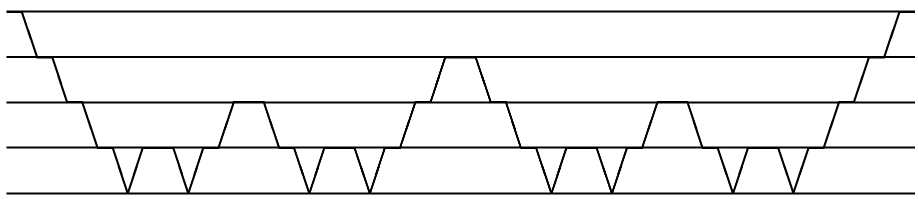


Figure 13.6: One iteration of the MGM with $\gamma = 2$ on level $l = 4$.

To find the error amplification operator of the MGM on level l, let us denote the exact solution of $A_{l-1}e_{l-1} = d_{l-1}$ $(= rd_l = r(A_l u_l - f_l))$ by $\overline{e}_{l-1}$. Instead of performing $u_l \leftarrow u_l - p\overline{e}_{l-1}$ as with the TGM, u_l is corrected in the "CGC-stage" of the MGM by means of

$$u_l \leftarrow u_l - pe_{l-1}, \tag{13.13}$$

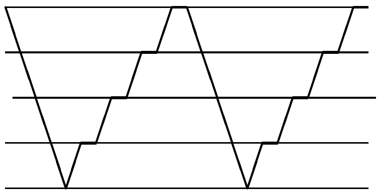


Figure 13.7: *Two* iterations of the sawtooth-cycle on level $l = 4$.

where e_{l-1} is obtained by γ iterations of the MGM on level $l-1$ on the equation $A_{l-1}e_{l-1} = d_{l-1}$ with initial error $0-\overline{e}_{l-1}$. We thus have $e_{l-1}-\overline{e}_{l-1} = (M_{l-1}^{(\mathrm{MGM})})^\gamma(0-\overline{e}_{l-1})$ and so $e_{l-1} = (I_{l-1} - (M_{l-1}^{(\mathrm{MGM})})^\gamma)\overline{e}_{l-1} = (I_{l-1} - (M_{l-1}^{(\mathrm{MGM})})^\gamma)A_{l-1}^{-1}r(A_l u_l - f_l)$. With this relation, (13.13) can be written as

$$u_l \leftarrow u_l - p(I_{l-1} - (M_{l-1}^{(\mathrm{MGM})})^\gamma)A_{l-1}^{-1}r(A_l u_l - f_l)$$

which is of the form (13.7) with $B_l = p(I_{l-1} - (M_{l-1}^{(\mathrm{MGM})})^\gamma)A_{l-1}^{-1}r$. Recalling that the error amplification operator belonging to (13.7) is given by $I_l - B_l A_l$ and also taking into account the pre- and post-smoothing iterations, we conclude that with $M_0^{(\mathrm{MGM})} := 0$, for $l > 0$ it holds that

$$\begin{aligned} M_l^{(\mathrm{MGM})} &= S_l^{\nu_2}\left(I_l - p(I_{l-1} - (M_{l-1}^{(\mathrm{MGM})})^\gamma)A_{l-1}^{-1}rA_l\right)S_l^{\nu_1} \\ &= M_l^{(\mathrm{TGM})} + S_l^{\nu_2}p(M_{l-1}^{(\mathrm{MGM})})^\gamma A_{l-1}^{-1}rA_lS_l^{\nu_1}. \end{aligned}$$

For our model problem one can check that for all $l \in \mathbf{N}$, $\|S_l\| \le 1$, $\|p\| \le 1$ and $\|A_{l-1}^{-1}rA_l\| \le \sqrt{2}$. If, as in the preceding section, we take $\nu_1 = \nu$, $\nu_2 = 0$ and $\omega = \frac{1}{2}$, then using $\|M_l^{(\mathrm{TGM})}\| \le \zeta_\nu$, we obtain

$$\|M_l^{(\mathrm{MGM})}\| \le \zeta_\nu + \sqrt{2}\,\|M_{l-1}^{(\mathrm{MGM})}\|^\gamma.$$

The equation $\zeta' = \zeta_\nu + \sqrt{2}\,\zeta'^\gamma$ has a solution $\zeta' = \zeta'_{\nu,\gamma} \in (\zeta_\nu, 1)$ if $\gamma \ge 2$ and $\zeta_\nu < \frac{1}{8}\sqrt{2}$ ($\Leftrightarrow \nu \ge 3$). This solution is clearly an upperbound for $\|M_l^{(\mathrm{MGM})}\|$ and so we have shown that at least for $\gamma \ge 2$ and $\nu \ge 3$ the contraction number of the MGM is less than 1 *uniform in* h_l.

The same argument can be used to prove convergence uniform in the grid size of a general MGM applied to some discretized elliptic boundary value problem if $\gamma \ge 2$, $\nu_1 + \nu_2$ is large enough and the corresponding TGM has a contraction number that converges to zero uniformly in the grid size when $\nu_1 + \nu_2 \to \infty$.

By means of a different technique, for a large class of problems including our model problem, convergence of the MGM uniform in the grid size can even be proved for the (cheaper) V-cycle ($\gamma = 1$) and for an arbitrary positive number of pre- and post-relaxations (cf. [2, 8], [3, chapter 5]).

Now we turn to the aspect of the cost of performing a general multi-grid step. The cost depends on the grid size ratio $\frac{h_{k-1}}{h_k}$ of subsequent levels and on the dimension d of the underlying domain. We fix the grid size ratio on its common choice 2. Then for the number of unknowns N_k on level k, we have $N_k \equiv 2^d N_{k-1}$. We make the natural assumption that the number of arithmetic operations needed to perform a restriction, prolongation or a fixed number of smoothing steps on level k are all proportional to N_k. Finally, let C be the number of arithmetic operations needed to solve a system $A_0u_0 = f_0$

on the coarsest grid. Then the number of arithmetic operations required to perform one multi-grid iteration on level l is proportional to

$$\begin{aligned}
&N_l + \gamma(N_{l-1} + \gamma(N_{l-2} + (\ldots + \gamma(N_1 + C)))) \\
&\equiv N_l \left(1 + (\gamma 2^{-d}) + (\gamma 2^{-d})^2 + \ldots + (\gamma 2^{-d})^{l-1} + (\gamma 2^{-d})^l \tfrac{C}{N_0 \gamma}\right) \\
&\equiv N_l
\end{aligned}$$

if $\gamma < 2^d$. We conclude that the MGM has the unique property that, if $\gamma < 2^d$, the costs per step are proportional to those of the Jacobi method, whilst for a large class of problems the contraction number is less than 1 *uniform in the grid size*. The storage requirements of the MGM are also proportional to those of the Jacobi method.

In practice, the most efficient MGMs are those with only a few (think of one or two) relaxations per step and $\gamma = 1$ or $\gamma = 2$ (if $d > 1$). Whereas the contraction number of the W-cycle closely resembles that of the corresponding TGM, the contraction number of the V-cycle is somewhat larger. For "neat" problems (smooth coefficients and boundaries) the V-cycle often yields the optimum algorithm. However, for example for boundary value problems on domains having a re-entrant corner, the W-cycle is a safer choice for obtaining a convergent method.

13.6 Multi-grid with nested iteration

The number of iterations needed to solve a system $A_l u_l = f_l$ within some given accuracy depends not only on the convergence speed of the iterative method used, but also on the initial error. In view of the multiple grids it is quite natural to find an accurate initial function by first solving $A_{l-1} u_{l-1} = f_{l-1}$ approximately and then to prolongate this approximation to the fine grid. An approximate solution of the system $A_{l-1} u_{l-1} = f_{l-1}$ can be found by applying this idea recursively, that is, solve $A_{l-2} u_{l-2} = f_{l-2}$ approximately, prolongate the approximation to the level $l-1$ and apply (on that level) a number of steps of the iterative method. By taking the MGM for the iterative method we arrive at the MGM with nested iteration, also called full multi-grid method (FMG):

procedure FGM(l, $(f_k)_{0 \le k \le l}$)
  begin
    $u_0 := A_0^{-1} f_0$;
    for $k := 1$ **to** l **do**
    begin $u_k := p u_{k-1}$;
      for $j := 1$ **to** i **do** MGM(k,u_k,f_k)
    end
  end

Note that the FMG method yields approximations for the solutions of $A_k u_k = f_k$ for $0 \le k \le l$.

Since the costs of a multi-grid iteration on level $k-1$ are about 2^{-d} times the costs of a multi-grid iteration on level k, the costs of a FMG iteration are about $i\left(1 + 2^{-d} + (2^{-d})^2 + (2^{-d})^3 + \ldots\right) \le \frac{i}{1-2^{-d}}$ times the costs of a multi-grid iteration and thus proportional to the number of unknowns.

The user of full multi-grid has to specify A_k and f_k not only for $k = l$ but also for $k \in \{0, 1, \ldots, l-1\}$. As discussed in section 4, the operators A_k for $k \in \{0, 1, \ldots, l-1\}$ can also be generated automatically using Galerkin approximations. However, in case of non-homogeneous boundary conditions there is no obvious way to do a similar thing for the right-hand sides. For this reason the black-box multi-grid solver discussed in the next chapter makes no use of the nested iteration technique.

To analyse the convergence behaviour of the FMG method, we first should ask ourselves how accurately we want to solve $A_l u_l = f_l$. The ultimate goal is to approximate the solution of the continuous (boundary value) problem. Therefore, there is no point in making an effort to solve the discretized problem with an (algebraic) error that is much less than the discretization error. On the other hand one should not be satisfied with an approximation with an (algebraic) error substantially larger than the discretization error because in that case it would be more efficient to use a less accurate discretization.

Now let $\sup_k \|M_k^{(\mathrm{MGM})}\| \le \zeta' < 1$. Suppose the call FMG($l$,$(f_k)_{0\le k\le l}$) yields an approximate solution u_k on some level $k \in [0, l-1]$ with (algebraic) error less than the discretization error on that level. Then if the prolongation p has a sufficiently large order, that is, if it is exact on polynomials of sufficiently high degree, we can expect the difference between pu_k and the exact solution of $A_{k+1}u_{k+1} = f_{k+1}$ to be less than some constant K, independent of k, times the discretization error on level $k+1$. So, if the number i in the FMG algorithm is chosen such that $\zeta^i K \le 1$, the application of i multi-grid iterations on level $k+1$ reduces the algebraic error again below the discretization error. Using an induction argument, we conclude that *one* FMG iteration yields an approximate solution of $A_l u_l = f_l$ with an (algebraic) error less than the discretization error.

If the costs are taken into account, which are proportional to the number of unknowns, the above analysis shows that the FMG method has an optimum complexity. Indeed, one cannot expect to solve a system up to an error that is less than the discretization error without performing some number of arithmetic operations per unknown! At least for large systems, this optimum complexity makes the FMG superior to all classical iterative methods including conjugate gradients. Although it has a contraction number less than 1 uniform in the grid size, even "plain" MGM requires a number of arithmetic operations per unknown proportional to $\ln h_l^{-1}$ for solving up to the discretization error. The cause of this is that the discretization error decreases when h_l decreases.

Finally in this section, we briefly discuss the question how to choose an initial function for our multi-grid iteration when a sequence of boundary value problems has to be solved generated by applying an implicit time discretization to an advection-diffusion problem. The general approach is to use the solution $u_k(t - \triangle t)$ found at the previous point of time as initial function for the iteration to determine $u_k(t)$. Yet, in case of multi-grid, this idea can be combined naturally with the nested iteration technique yielding a much smaller initial error. Indeed, the difference $u_k(t) - u_k(t - \triangle t)$ can be expected to be well approximated by $p(u_{k-1}(t) - u_{k-1}(t - \triangle t))$ which leads to choise $u_k(t - \triangle t) + p(u_{k-1}(t) - u_{k-1}(t - \triangle t))$ instead of $pu_{k-1}(t)$ as initial function in **procedure** FMG on the previous page.

13.7 Robustness

In the preceding sections, we emphasized on convergence uniform in the grid size. Another desirable property of an iterative method is that it is *robust*, which means that it converges uniformly with respect to relevant parameters in the problem.

For example, let us replace our model problem (13.2) by the upwind discretization of the one dimensional advection diffusion problem $\begin{cases} -\epsilon u'' + u' = f & \text{on } (0,1) \\ u(0) = u(1) = 0 \end{cases}$ $(\epsilon > 0)$, so that

$$A_l = h_l^{-2} \begin{bmatrix} 2\epsilon + h_l & -\epsilon & & \\ -(\epsilon + h_l) & 2\epsilon + h_l & -\epsilon & \\ & & \ddots & \\ & & -(\epsilon + h_l) & 2\epsilon + h_l \end{bmatrix}.$$

Then our MGM applied to this class of problems appears to be non-robust; the convergence rate tends to 1 if $h_l \leq \epsilon \downarrow 0$. In [4], it is proved that if we replace our linear interpolation by the problem dependent prolongation $p : \mathbf{R}^{n_{l-1}} \to \mathbf{R}^{n_l}$ defined by

$$\begin{cases} (pv_{l-1})(2ih_l) &= v_{l-1}(ih_l) \\ (pv_{l-1})((2i+1)h_l) &= \dfrac{(\epsilon + h_l)v_{l-1}(ih_{l-1}) + \epsilon v_{l-1}((i+1)h_{l-1})}{2\epsilon + h_l} \end{cases}$$

and we also adapt our r and $(A_k)_{0 \leq k < l}$ according to $r = p^*$ and $A_k = rA_{k+1}p$, then the resulting MGM has a contraction number less than 1 uniform in h_l *and* ϵ, that is, the method is robust. Note that for $\epsilon \downarrow 0$, the above prolongation turns into a one-sided piecewise constant interpolation in the direction of the flow (here left-to-right). If, in addition, the damped Jacobi smoother is replaced by the Gauss-Seidel smoother with respect to the ordering of the grid points according to the flow direction, then the contraction number even tends to zero if $\epsilon \downarrow 0$ uniform in the grid size. Note that the Gauss-Seidel smoother is an exact solver for the system in case $\epsilon = 0$.

Another example is obtained by the (discretized) boundary value problem $\begin{cases} -(au')' = f & \text{on } (0,1) \\ u(0) = u(1) = 0 \end{cases}$ $(a : [0,1] \to (0,\infty))$. For "smooth" functions a, our standard MGM works well. However, for non-smooth a, particularly for a having discontinuities, adaptations of p now dependent on a are again necessary (cf. [2, §10.3.1], [9, §5.4]).

For 2D advection-diffusion problems having a dominating advection term or discontinuous coefficients, it is much harder to find a suitable prolongation p. In chapter 14, a multi-grid code is described that generates such p automatically .

It is not always sufficient (or necessary) to adapt the prolongation to obtain a robust MGM; the smoother has to be chosen with some care as well. In practical MGMs, the damped Jacobi smoother is seldom used. Because of their robustness for large classes of problems including advection-diffusion problems, popular smoothers are the (modified) incomplete LU smoothers or the incomplete block LU smoothers (cf. [2, 9, 7]).

13.8 Conclusions

As we have seen, at least for "sufficiently large" systems the MGM is superior to other iterative methods. The question of what "sufficiently large" means in this context can only be answered by performing numerical experiments. We refer the reader to the next chapter and to the literature. Generally, it is observed that the performance of multi-grid for "medium-sized" systems (a 65×65 grid is mentioned) is comparable to that of preconditioned conjugate gradients. For larger systems the MGM is more efficient. On the other hand, conjugate gradients are easier to programme.

In this introduction, many important multi-grid issues have not been discussed, for example the non-linear MGM. For further reading, we refer to the excellent monograph by Hackbusch ([2]) which contains an abundance of information about theory and practice as well as an historical overview of the development of multi-grid. Especially for readers who do not have a mathematical background, the book of Wesseling ([9]) provides a good alternative. The book Multigrid Methods ([3]) is a compilation of several papers by different authors. One of the contributions is an interesting convergence analysis along different lines than the analysis of Hackbusch. In all three books one can find an extensive bibliography.

References

[1] Brandt, A.: Multi-level adaptive solutions to boundary value problems, *Math. Comput.*, **31** (1977), pp. 333–390.

[2] Hackbusch, W.: *Multi-Grid Methods and Applications* (Springer-Verlag, Berlin, 1985).

[3] McCormick, S.F., editor: *Multigrid Methods*, Frontiers in Applied Mathematics (SIAM, Philadelphia, Pennsylvania, 1987).

[4] Reusken, A.: Multigrid with matrix-dependent transfer operators for a singular perturbation problem, *Computing*, **50** (1993), pp. 199–211.

[5] Ruge, J.W. and Stüben, K.: Algebraic multigrid, Chapter 4 of [3].

[6] Stevenson, R.P.: *On the Validity of Local Mode Analysis of Multi-Grid Methods*, PhD thesis (University of Utrecht, 1990).

[7] Stevenson, R.P.: Modified ILU as a smoother, *Numer. Math.*, **68** (1994), pp. 295–309.

[8] Stevenson, R.P.: New estimates of the contraction number of V-cycle multi-grid with applications to anisotropic equations, *Incomplete Decompositions, Proceedings of the Eighth GAMM Seminar*, Notes on Numerical Fluid Mechanics, (Vieweg Verlag, Braunschweig, 1992).

[9] Wesseling, P.: *An Introduction to Multigrid Methods* (John Wiley & Sons, Chichester, 1992).