

DELFT UNIVERSITY OF TECHNOLOGY

REPORT 82-05

A robust and efficient
multigrid method

by

P. Wesseling



Reports of the Department of Mathematics and Informatics no. 82-05

Delft 1982

Abstract

A multigrid method is described for an arbitrary elliptic partial differential equation with continuous coefficients on a rectangular region. The method is perceived by the user just as any other solver of linear algebraic systems. The user has to specify the matrix and right-hand-side only and does not need to be familiar with multigrid methods. The main ingredients of the method are the use of incomplete LU-decomposition for smoothing, Galerkin coarse-grid approximation, 7-point prolongation and restriction, and a fixed multigrid strategy. Application to various test-problems shows the method to be efficient and robust. Its efficiency is compared with the efficiency of other multigrid methods.

Key words

Numerical methods for elliptic equations, multigrid methods, incomplete LU-decomposition.

To appear in the Proceedings of the Conference on Multigrid Methods,
Cologne, November 23-27, 1981.

1. Introduction.

Both in theory and practice, multigrid methods solve elliptic boundary value problems in $O(N)$ operations, with N the number of points in the computational grid. This compares favorably with other methods. Only for separable equations on rectangular regions there are methods of similar efficiency; see Schumann (1980) for a recent survey of work in this area. Or, they have a computational complexity of $O(N^\alpha)$ with $\alpha > 1$. Of these methods, those of the preconditioned Lanczos type deserve special mention, because they have $\alpha=1.25$ (as has been proved by Gustafsson (1978) for the Poisson equation, but $\alpha=1.25$ seems to hold more generally), which is low, whereas they have broad applicability. The original version (ICCG, Meijerink and van der Vorst (1977)) is restricted to self-adjoint equations, but extension to the general case is possible and has been given by, among others, Wesseling and Sonneveld (1980), who also describe numerical experiments comparing a multigrid and a preconditioned Lanczos method, and by van der Vorst (1981). Such comparisons are also described by Kettler and Meijerink (1981) and Kettler (1982), who furthermore obtain spectacular results by combining a multigrid and a conjugate gradient method.

For large-scale calculations in engineering and physics, multigrid methods potentially far surpass other known methods as far as efficiency is concerned, with the possible exception of preconditioned Lanczos methods. However, routine application of multigrid methods to large-scale problems is at present still hampered by the fact, that the promise of computational efficiency is not always fulfilled in the hands of a non-expert, and that multigrid methods are more complicated than classical iterative methods. There are many ways to implement the basic ideas underlying multigrid methods, and the way in which this is done may make the efficiency problem-dependent.

The aim of this paper is to present the details of a multigrid method, called MGD1 for brevity. The method is constructed such that it is perceived by the user just as any other solver of linear systems. That is, it operates only on the matrix that is given and does not refer to the underlying differential equation and boundary conditions. This means that powerful ideas concerning adaptive multigrid methods put forward by Brandt (1977, 1979) are deliberately not used. Of course, the basic multigrid methodology is employed, as typified in various ways by the work of Fedorenko (1962), Bakhvalov (1966), Astrachancev (1971), Brandt (1973), Frederickson (1975), Wachspress (1975), Hackbusch (1978a), Wesseling (1977), and become widely known and appreciated by

Brandt (1977). It depends on the user (is he familiar with multigrid methods or not) and on the problem, whether an adaptive or a non-adaptive approach, such as presented here, is to be preferred.

The method MGD1 is fast for a large class of elliptic boundary value problems, as will be made plausible and demonstrated experimentally. Operation counts and rates of convergence are given, and a comparison of computational efficiency with other methods is made.

The main characteristics of the method are the use of incomplete LU-decomposition for smoothing, 7-point prolongation and restriction operators, coarse-grid Galerkin approximation, and the use of a fixed multigrid strategy, that will be called the sawtooth cycle. It does not need to be adapted to the problem, and is expected to be useful to the non-specialist. A FORTRAN program will be described in a forthcoming report.

2. A multigrid method.

A multigrid method will be presented for the solution of an elliptic boundary value problem on a rectangle, discretized by finite differences.

A computational grid Ω^ℓ and a corresponding set of grid-functions U^ℓ are defined as follows:

$$\begin{aligned}\Omega^\ell &\equiv \{(x_0, y_0) \cup (x_1, y_1), x_1 = x_0 + ih_x, y_j = y_0 + jh_y, i, j = 1(1)2^\ell\}, \\ U^\ell &\equiv \{u^\ell: \Omega^\ell \rightarrow \mathbb{R}\}.\end{aligned}\tag{2.1}$$

The formulation (2.1) allows elimination of physical boundaries where Dirichlet boundary conditions are given. For example, if all boundary conditions are of Dirichlet type and the region is the unit square one may choose: $h_x = h_y = h = (2^\ell + 2)^{-1}$, $x_0 = y_0 = h$. It is not really necessary to have the number of x- and y-gridlines equal. By coordinate stretching one can generate a non-equidistant mesh in the physical plane. The linear algebraic system generated by the difference scheme is denoted by:

$$A^\ell u^\ell = f^\ell.\tag{2.2}$$

The multigrid method makes use of a hierarchy of computational grids Ω^k and corresponding sets of grid-functions U^k , $k = \ell - 1(-1)1$, defined by (2.1) with ℓ replaced by k . On the coarser grids (i.e. grids with larger step-size, hence smaller k) equation (2.2) is approximated by:

$$A^k u^k = f^k, \quad k = \ell-1(-1)1. \quad (2.3)$$

Furthermore, let there be given a restriction operator r^k and a prolongation operator p^k :

$$r^k : U^k \rightarrow U^{k-1}, \quad p^k : U^{k-1} \rightarrow U^k. \quad (2.4)$$

A^k, f^k, r^k, p^k will be specified later.

For the so-called smoothing process use is made of incomplete LU-(ILU-) decomposition. Temporarily suppressing the superscript k , we assume that we have a lower and an upper triangular matrix L and U respectively, such that

$$LU = A + C. \quad (2.5)$$

L, U and C will be specified later. Consider the following iterative process for solving (2.2) or (2.3):

$$\begin{aligned} x &:= x + (LU)^{-1} (f - Ax), \\ \text{or} \\ x &:= (LU)^{-1} (f + Cx). \end{aligned} \quad (2.6)$$

The multigrid method presented in the following quasi-Algol program can be regarded as a method to accelerate the iterative process (2.6):

multigrid method MGD1:

```

begin  $f^{\ell-1} := r^{\ell} C^{\ell} (u^{\ell} - u^{\ell-1});$ 
  for  $k := \ell-1(-1)2$  do  $f^{k-1} := r^k f^k;$ 
   $u^1 := (L^1 U^1)^{-1} f^1;$ 
  for  $k := 2(1)\ell-1$  do  $u^k := (L^k U^k)^{-1} (C^k p^k u^{k-1} + f^k);$ 
   $\bar{u}^{\ell} := u^{\ell} + p^{\ell} u^{\ell-1};$ 
   $u^{\ell} := (L^{\ell} U^{\ell})^{-1} (C^{\ell} \bar{u}^{\ell} + f^{\ell})$ 
end of one iteration with MGD1;
  
```

Using (2.5), one may verify that $C^{\ell} (u^{\ell} - u^{\ell-1}) = f^{\ell} - A^{\ell} u^{\ell}$ is the residue associated with the current iterand u^{ℓ} . As we shall see, C^{ℓ} has only two non-zero diagonals, hence $C^{\ell} (u^{\ell} - u^{\ell-1})$ is a fast way to compute the residue $f^{\ell} - A^{\ell} u^{\ell}$. When starting, \bar{u}^{ℓ} and u^{ℓ} are not available. A cheap way to get started is to use the initial estimate $u^{\ell} = 0$, and replace $C^{\ell} (u^{\ell} - u^{\ell-1})$ by $f^{\ell} - A^{\ell} u^{\ell} = f^{\ell}$. In this case

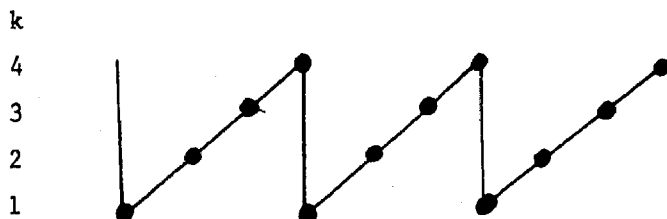
the computation effectively starts on the coarsest grid, which is advantageous, because no effort is wasted in correcting a perhaps unfortunate first guess. If one has a good initial estimate available, one may generate u^ℓ and $u^{-\ell}$ as follows:

$$u^{-\ell} := \text{initial estimate};$$

$$u^\ell := (L^\ell U^\ell)^{-1} (C^\ell u^{-\ell} + f^\ell);$$

In this case the computation starts with a smoothing step on the finest grid, which costs little more than a straightforward residue calculation.

The method is not recursive, and is easily implemented in FORTRAN. Each coarse grid is visited only once, and one smoothing step is performed after each coarse grid correction. This multigrid strategy may be depicted graphically as follows (for 4 grids):



Each dot represents a smoothing operation. This diagram suggests "sawtooth cycle" as an appropriate name for this strategy. This is probably the simplest multigrid strategy that one can think of; cf. Brandt (1977), Hackbusch (1981), Hemker (1981) for a description of many possible multigrid strategies.

A measure of the computational cost of one execution of multigrid method MGD1 may be obtained by counting the arithmetic operations that are visible in the mathematical formulae. The following table gives the operation count per grid-point of Ω^k for various parts of the algorithm.

C^k	r^k	$(L^k U^k)^{-1}$	p^k
3	2	13	1.5

The results for $(L^k U^k)^{-1}$, r^k and p^k follow from subsequent sections. With $n_k = (2^k + 1)^2$ the number of grid-points of Ω^k , we obtain the following total count:

$$6n_\ell + 2 \sum_{k=2}^{\ell-1} n_k + 13n_1 + 18.5 \sum_{k=2}^{\ell-1} n_k + 2.5n_\ell + 17n_\ell \cong 30.4^\ell,$$

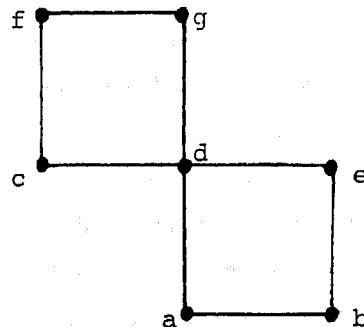
or about 30 operations per grid-point of Ω^ℓ .

The decisions made in the design of MGD1 are based on comparative experiments described by Wesseling (1980) and Mol (1981).

3. Incomplete LU-decomposition.

ILU- decomposition was used by Meijerink and van der Vorst (1977) as an effective preconditioning for conjugate gradient methods, and was introduced by Wesseling and Sonneveld (1980) as a smoothing process for multigrid methods. For experiments with and analysis of various ILU smoothing processes, see Wesseling and Sonneveld (1980), Hemker (1980a), Mol (1981), Kettler and Meijerink (1981), and the extensive treatment by Kettler (1982).

The general second order elliptic differential operator can be approximated by central or one-sided differences using the 7-point finite difference molecule depicted here. If no mixed derivative is present atoms b and f are superfluous, and we have the familiar 5-point molecule. Let the points of the computational grid Ω^k be ordered as follows: $(0,0), (1,0), (2,0), \dots, (2,0), (0,1), (1,1), (2,1), \dots, (2^k, 2^k)$. Then the finite difference matrix A^k has 7 non-zero diagonals, labeled left-to-right as a, b, ..., g, each of which corresponds with the atom with the same label. By non-zero we mean: possibly non-zero.



The ILU-decomposition to be employed here can be described as follows. In the same locations as a, b, c, L is prescribed to have non-zero diagonals α, β, γ respectively. The main diagonal of L is specified to be unity. At locations d, e, f, g, U has non-zero diagonals $\delta, \epsilon, \zeta, \eta$, respectively. The rest of L and U is zero. L and U can be conveniently computed by Crout-like formulae, as follows, on an $m*n$ grid. Subscript k is the row-number.

$$\begin{aligned}
 \alpha_k &= a_k / \delta_{k-m}, & \beta_k &= (b_k - \alpha_k \epsilon_{k-m}) / \delta_{k-m+1}, \\
 \gamma_k &= (c_k - \alpha_k \zeta_{k-m}) / \delta_{k-1}, & \delta_k &= d_k - \gamma_k \epsilon_{k-1} - \beta_k \zeta_{k-m+1} - \alpha_k \eta_{k-m}, \\
 \epsilon_k &= e_k - \beta_k \eta_{k-m+1}, & \zeta_k &= f_k - \gamma_k \eta_{k-1}, \quad \eta_k = g_k.
 \end{aligned}
 \tag{3.1}$$

Quantities that are not defined are to be replaced by 0.

The error matrix $C = LU - A$ has only two non-zero diagonals ψ and λ , located next to and inside b and f , respectively, and given by:

$$\psi_k = \beta_k \varepsilon_{k-m+1}, \quad \lambda_k = \gamma_k \zeta_{k-1}. \quad (3.2)$$

The solution of $LU = q$ is obtained by back-substitution:

$$u_i := q_i - \gamma_i u_{i-1} - \beta_i u_{i-m+1} - \alpha_i u_{i-m}, \quad i=1(1)mn; \quad (3.3)$$

$$u_i := (u_i - \varepsilon_i u_{i+1} - \zeta_i u_{i+m-1} - \eta_i u_{i+m}) / \delta_i, \quad i=mn(-1)1.$$

It is easily verified, that the construction of L and U , the construction of C , and the solution of $LUu=q$ takes 17, 2 and 13 arithmetic operations per grid-point, respectively.

The computation of $Cu+f$ takes 4 operations, because C has only 2 non-zero diagonals, hence a complete smoothing step $u := (LU)^{-1}(Cu+f)$ takes 17 operations. L and U can be stored in the space for A , because A is not needed. If A has only 5 non-zero diagonals, this requires an extra storage of 2 real per grid point. Storage of C also requires 2 reals per grid-point, but C can also be computed instead of being stored, in which case the cost of a smoothing step increases from 17 to 19. The memory requirement for the quantities defined on the coarse grid is about $(1/4 + 1/16 + \dots) = 1/3$ of the memory requirement on the finest grid.

We will not dwell upon the existence and stability of the ILU-decomposition just described. Meijerink and van der Vorst (1977) have shown existence if A is a symmetric M -matrix, i.e. $a_{ij} = a_{ji}$, $a_{ij} < 0$ for $i \neq j$ and $A^{-1} > 0$. They also note that one has existence under much more general circumstances. We have found experimentally, that the ILU-decomposition exists and provides an efficient smoothing process, if in the non-self-adjoint case a sufficient amount of artificial viscosity is introduced on the finest grid. This is also necessary for all other smoothing processes that we know of.

In order to make Fourier methods applicable, in smoothing (and two-level) analysis it must be assumed that the values of the elements of L and U do not vary along diagonals; if (3.1) is used this is usually not the case near boundaries, and in certain strongly anisotropic diffusion problems the influence of the boundaries on the ILU-decomposition extends inwards over many meshes. In such cases smoothing and two-level analysis are not realistic for the present method MGD1.

4. Prolongation and restriction.

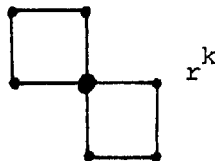
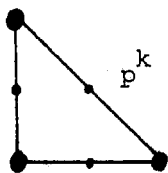
Let the value of the grid-function u^k in the point $(s \cdot 2^{-k}, t \cdot 2^{-k})$ be denoted by u_{st}^k . Prolongation and restriction are defined by:

$$(p^k u^{k-1})_{2s, 2t} = u_{st}^{k-1}, \quad (p^k u^{k-1})_{2s+1, 2t} = \frac{1}{2} (u_{st}^{k-1} + u_{s+1, t}^{k-1}), \quad (4.1)$$

$$(p^k u^{k-1})_{2s, 2t+1} = \frac{1}{2} (u_{st}^{k-1} + u_{s, t+1}^{k-1}), \quad (p^k u^{k-1})_{2s+1, 2t+1} = \frac{1}{2} (u_{st}^{k-1} + u_{s+1, t+1}^{k-1}).$$

$$(r^k u^k)_{st} = u_{2s, 2t}^k + \frac{1}{2} (u_{2s+1, 2t}^k + u_{2s, 2t+1}^k + u_{2s-1, 2t}^k + u_{2s, 2t-1}^k + u_{2s+1, 2t-1}^k + u_{2s-1, 2t+1}^k). \quad (4.2)$$

The following diagrams may clarify the structure of p^k and r^k .



This p^k provides linear interpolation, while having a sparser matrix representation than all other linear interpolation operators, and r^k is its adjoint, in the sense that

$$(p^k u^{k-1}, v^k)_k = (u^{k-1}, r^k v^k)_{k-1}, \quad \forall v^k \in U^k, \quad (4.3)$$

with $(u^k, v^k)_k \equiv \sum_{\Omega^k} u_{ij}^k v_{ij}^k$. Because r^k is a weighted average of 7 points, we

call this 7-point prolongation and restriction.

5. Galerkin coarse grid approximation.

The coarse grid operators A^k are defined as follows:

$$A^{k-1} = r^k A^k p^k, \quad k = \ell(-1)2. \quad (5.1)$$

We call this a Galerkin approximation because (5.1) implies:

$$(A^k p^k u^{k-1}, p^k v^{k-1})_k = (A^{k-1} u^{k-1}, v^{k-1})_{k-1}, \quad \forall v^{k-1} \in U^{k-1}, \quad (5.2)$$

if (4.3) holds; hence, we have a case of projection in a lower-dimensional subspace.

A more obvious way to generate A^{k-1} is to use a finite difference method. Under (5.1), if r^k and p^k are as in section 4, standard central equidistant finite difference approximations of the operators $\partial^2/\partial x^2$, $\partial^2/\partial y^2$, $\partial^2/\partial x\partial y$ using the 7-point difference molecule of section 3 are invariant, hence Galerkin and finite difference approximation are identical. When lower derivatives occur, or the coefficients are variable, or the mesh non-equidistant (which is the case on coarser grids if on the finest grid Dirichlet boundaries are eliminated), the two approximations differ. Their mutual relationship closely resembles the relationship between finite difference and finite element approximations. The Galerkin method automatically generates accurate approximations and takes care of special circumstances, such as changing mesh-size or varying coefficients, but the numerical analyst can always achieve the same accuracy with an ably designed finite difference approximation. The transformation of upwind differences by (5.1) is interesting; the difference molecules on the finest and five coarser grids are given below:

$$\begin{array}{cccc}
 0 & 0 & -1 & 1 & -5 & 5 & -21 & 21 \\
 -1 & 1 & 0 & -5 & 4 & 1 & -15 & 8 & 7 & -51 & 16 & 35 \\
 & 0 & 0 & -1 & 1 & -5 & 5 & -21 & 21 & & & \\
 & & & (*2^{-2}) & & (*2^{-3}) & & (*2^{-4}) & & & & \\
 \\
 -85 & 85 & & & -341 & 341 & & & & & & \\
 -187 & 32 & 155 & & -715 & 64 & 651 & & & & & \\
 & -85 & 85 & & -341 & 341 & & & & & & \\
 & (*2^{-5}) & & & (*2^{-6}) & & & & & & &
 \end{array}$$

For the derivation of these molecules, the formulae given by Mol (1981) have been used. Apparently, upwind differencing is gradually replaced by central differencing, plus a higher order truncation error containing a mixed third derivative. Diagonal dominance is lost, but in practice we have never encountered numerical "wiggles" or instability of the ILU-decomposition; note that as the grid gets coarser, the ILU-decomposition becomes more exact. Later, successful experiments with the upwind-discretized convection-diffusion equation will be reported.

One way of programming (5.1) is as follows. It is based on a datastructure used earlier by Frederickson (1975). In this section, Greek subscripts are 2-tuples identifying points of the computational grid, e.g. $\alpha=(\alpha_1, \alpha_2)$ indicates

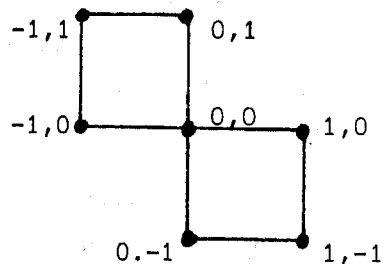
the grid-point with indices (α_1, α_2) (cf. the ordering introduced in section 3)

To the atoms of the difference molecule 2-tuples are assigned according to the accompanying diagram;

these are also identified by Greek subscripts.

By $A_{\alpha\beta}^k$ we denote the element of the matrix A^k in row number $1+\alpha_1+m\alpha_2$ and column $1+\alpha_1+m\alpha_2+\beta_1+m\beta_2$, with m the number of grid-points of Ω^k in the

x-direction. For example, $\beta=(1,-1)$ corresponds with the b-diagonal of section 3. If α is outside Ω^k or β is outside the molecule then $A_{\alpha\beta}^k$ is defined to be zero.



With these conventions, matrix-vector multiplication can be formulated as follows:

$$(A^k u^k)_\alpha = \sum_{\beta} A_{\alpha\beta}^k u_{\alpha+\beta}^k, \quad (5.3)$$

with range $\beta = Z \times Z$. Restriction and prolongation can be represented as follows:

$$(r^k u^k)_\alpha = \frac{1}{2} \sum_{\beta} \mu_{\beta} u_{2\alpha+\beta}^k, \quad (5.4)$$

$$(p^k u^k)_\alpha = \frac{1}{2} \sum_{\beta} \mu_{\alpha-2\beta} u_{\beta}^{k-1}, \quad (5.5)$$

with the weight factors μ_{β} defined by (cf. (4.1) and (4.2)):

if β is inside the molecule, then $\mu_{\beta}=1$, except $\mu_{0,0}=2$; outside the molecule, $\mu_{\beta}=0$. Eq. (5.5) is not a convenient way to compute $p^k u^k$, but it can be used to derive a useful formula for $r^k A^k p^k$. From (5.3)-(5.5) it follows that for any u^{k-1} :

$$(r^k A^k p^k u^{k-1})_\alpha = \frac{1}{2} \sum_{\beta} \mu_{\beta} \sum_{\gamma} A_{2\alpha+\beta, \gamma}^k \frac{1}{2} \sum_{\delta} \mu_{2\alpha+\beta+\gamma-2\delta} u_{\delta}^{k-1} \quad (5.6)$$

Since the range of β , γ and δ may be taken to be all of $Z \times Z$, change of variables is easy. Let $\delta'=\alpha+\delta$, $\gamma'=\beta+\gamma-2\delta'$, then (5.6) takes on a form from which we may conclude (omitting primes):

$$(r^k A^k p^k)_{\alpha\delta} = \frac{1}{4} \sum_{\beta, \gamma} \mu_{\beta} \mu_{\gamma} A_{2\alpha+\beta, 2\delta+\gamma-\beta}^k \quad (5.7)$$

It is found that if A^k has a general 7 (or fewer)-point structure, then A^{k-1} has a 7-point structure. An important point is that $2\delta+\gamma-\beta$ is only 95(61)

times inside the molecule of A^k if A^k is a 7-(5-)point molecule, for all $7^3=343$ possible combinations of β, γ and δ . This is exploited by putting the α -loop inside the δ -, β -and γ -loops, and arrange the computation as follows:

```

 $A^{k-1} := 0;$ 
for  $\delta \in$  molecule do
for  $\beta \in$  molecule do
for  $\gamma \in$  molecule while  $2\delta+\gamma-\beta$  molecule do
begin  $\mu = \mu_\beta \mu_\gamma;$ 
for  $\alpha \in \Omega^{k-1}$  while  $2\alpha+\beta \in \Omega^k$  do
 $A_{\alpha\delta}^{k-1} := A_{\alpha\delta}^{k-1} + \mu A_{2\alpha+\beta, 2\delta+\gamma-\beta}^k$ 
end;
 $A^{k-1} := A^{k-1}/4;$ 

```

The cost of the inner loop is 2 operations, hence the total cost is $2 \cdot 95(2 \cdot 61)$ for a 7-(5-) point operator A^k . The division by 16 adds one operation, so that our final conclusion is, that the construction of A^{k-1} takes 191 (123) operations per grid-point of Ω^{k-1} for a 7-(5-) point operator A^k . The total work for the construction of A^k , $k=\ell-1(-1)1$ in operations per grid-point of Ω^ℓ is about $191/3 \approx 64$ (7-point A^ℓ), or $123/4 + 191/12 \approx 47$ (5-point A^ℓ). This has to be done only once, before the multigrid iterations start. It depends on the complexity of A^ℓ whether computation of A^k , $k < \ell$ is cheaper with the finite difference method or with the above Galerkin method. But more important is the fact, that the Galerkin method enables us to work only with the given matrix A^ℓ and not to refer to the underlying problem (equation and boundary conditions), and that always good coarse grid approximations are obtained automatically.

6. Numerical experiments

Standardized test-problems are useful for demonstrating and comparing the applicability and performance of multigrid methods. If the coefficients are constant, smoothing analysis (cf. Brandt (1977)) can help to understand and construct efficient smoothing processes. Two-level analysis (cf. Brandt and

Dinar (1979), Foerster et al. (1981), Ries et al. (1981)) realistically predicts the rate of convergence of certain (not all, for example not for the sawtooth cycle described previously) multigrid methods; for a certain method applied to the Poisson equation Braess (1981) has given a rigorous prediction. Such analyses use Fourier methods; see Hemker (1980b) for an introduction to the Fourier analysis of multigrid methods. Where Fourier analysis is not applicable, one can, given sufficient computer time, compute the spectral norm and spectral radius (i.e. the asymptotic rate of convergence) numerically, or just observe the rate of convergence and use heuristic arguments in order to verify the soundness and efficiency of a multigrid method. In practice, quite often the observed rate of convergence of a good multigrid method is considerably better than the asymptotic rate, which is not reached because already after a few iterations discretization or even machine accuracy is obtained.

The test problems should be standardized, so that results reported by different authors can be easily compared and reproduced. The fact that constant coefficient test problems have a very special type of spectrum (cf. Curtis (1981)) makes them somewhat exceptional; therefore test problems with variable coefficients should also be included. Unless one wishes to design a method especially for a specific problem the special properties of a given test problem, such as for example a coefficient being constant, should not be exploited, and not be taken into account in operation counts.

We have looked for suitable test problems that have already been treated by other authors, and will report results for the following problems:

$$(i) \quad \phi_{xx} + \phi_{yy} = 4, \quad (6.1)$$

$$(ii) \quad \phi_{xx} + 0.01 \phi_{yy} = 2.02, \quad (6.2)$$

$$(iii) \quad 0.01 \phi_{xx} + \phi_{yy} = 2.02, \quad (6.3)$$

$$(iv) \quad \phi_{xx} + 1.7 \phi_{xy} + \phi_{yy} = 4, \quad (6.4)$$

$$(v) \quad u\phi_x + v\phi_y = 0.001(\phi_{xx} + \phi_{yy}) - 1, \quad (6.5)$$

$$(u,v) = (1,0), (0,1), (1,1), (1,-1),$$

$$(a) \quad (b) \quad (c) \quad (d)$$

$$(vi) \quad (a\phi_x)_x + (a\phi_y)_y = 0, \quad a = |\sin kx \sin ky|. \quad (6.6)$$

The region is $\Omega = (0,1) \times (0,1)$. The computational grid is equidistant. For problems (i) - (iv) the boundary condition is $\phi|_{\partial\Omega} = x^2 + y^2$, exact solution: $\phi = x^2 + y^2$. For (v) and (vi), $\phi|_{\partial\Omega} = 0$, exact solution not known for (v), zero for (vi). Lack of time prevented us from including cases with Neumann boundary conditions, but this should certainly be included, because the rate of convergence, of some multigrid methods (but not MGD1, except for (iii)) is affected by the type of boundary condition. Standard 5-point central differencing is used on the finest grid except for (v), where upwind differencing according to Il'in (1969) is employed. There is no particular reason for using Il'in discretization in the present context; any other form of upwind discretization would lead to roughly the same results (this is an example where Galerkin coarse grid discretization is cheaper than finite differences, because of the cost of the Il'in coefficients). The initial guess of the solution is $\phi \equiv 0$, except for (vi), where the initial ϕ is uniformly randomly distributed. The boundary conditions are not eliminated from the equations.

Besides being important for many applications in its own right (but fast solvers of Fourier analysis/cyclic reduction type (see e.g. Schumann (1980)) exist already for some time, and are competitive with multigrid methods) problem (i) typifies self-adjoint elliptic equations with smoothly varying coefficients. Problems (ii) and (iii) represent anisotropic diffusion problems or problems with strong coordinate stretching in one direction. Although mathematically almost identical, the rate of convergence of some methods (including MGD1) may differ significantly between (ii) and (iii). In (iv) we have a mixed derivative, which does not occur often in mathematical physics, but which we include for completeness. Non-orthogonal coordinate transformations give rise to mixed derivatives. The convection-diffusion problem (v) represents a singular perturbation problem of a type that is ubiquitous in fluid dynamics. Finally, (vi) represents problems with slowly or rapidly but continuously varying coefficients.

Because some or even all of the specific difficulties represented by these test problems may occur simultaneously in a given application, a method should perform well for more than one problem. The method MGD1 described here performs efficiently for all six test problems, although for (iii) it is somewhat less efficient than for the other five. For fluid mechanical applications (i) and (v) should be mastered, for reservoir engineering the method should be able to handle all problems, and in addition the more difficult test problems of Stone and Kershaw (see Kettler and Meijerink (1981))

cf. Kettler (1982)), in which the coefficients are discontinuous, and the region non-rectangular. The inclusion of a Navier-Stokes test problem seems desirable; see Wesseling and Sonneveld (1980) for some results.

In the following table we list publications that give results for the test-problems above. In some cases, the right-hand-side and boundary conditions may be different.

Test-problem	(i)	(ii)	(iii)	(iv)	(v)	(vi)
Brandt (1977)	*					
Hackbusch (1978b)	*, Δ	*		*		
Nicolaides (1979)	*, Δ					*
W,S,M	*	*	*	*	*	*
F,R,B,S	*, Δ	*	*		∇	

Table 6.1 Publications on test-problems (i) - (vi)

Publications that consider related methods are grouped together. W,S,M stands for Wesseling and Sonneveld (1980), Wesseling (1980), Mol (1981); F,R,B,S for Foerster et al. (1981), Ries et al. (1981), Börgers (1981), Stüben et al. (1982). The bulk of the results for MGD1 quoted here are taken from W,S,M. In these publications small inconsequential differences occur due to the fact, that M eliminates the boundary conditions, whereas W does not. Where the symbols vary along a row in table 6.1, different algorithms were used for different test-problems, otherwise the same algorithm was used. Where no entry occurs, this means that we have no results at our disposal; the method may or may not be applicable. Some publications treat other test-problems, not discussed here, as well.

The following table gives results for method MGD1.

	(i)	(ii)	(iii)	(iv)
M, ℓ	8,6	10,6	4,4	7,6
ρ ,t	0.033, 20	0.15, 36	0.0016, 11	0.025, 19
	(va)	(vb)	(vc)	(vd)
M, ℓ	3,4	2,4	1,4	4,4
ρ ,t	0.0030, 12	$7 \cdot 10^{-5}$, 7	$3 \cdot 10^{-9}$, 4	0.040, 21

Table 6.2 Method MGD1 applied to test-problems (i) - (v).

In table 6.2, M is the number of iterations that were carried out; ℓ determines the number of grid-points $(2^\ell+1)*(2^\ell+1)$ of the finest grid; ρ is the average reduction factor, defined by: ρ^M =quotient of Euclidean norms of residues $Ax=f$ before and after M iterations; $t=-30/^{10}\log\rho$ is the number of operations per grid-point of the finest grid for 0.1 reduction of the residual. For (v) the boundary conditions were eliminated; this is of no consequence.

Of course, ρ depends on M and on the initial guess, hence, ρ is afflicted with a certain arbitrariness, which the spectral norm and radius lack. For (i) it has been determined numerically, that the spectral radius $\rho_\infty=0.090$ ($t=29$).

Clearly, for (ii) the results are worse than for the other cases. For this case the smoothing factor of ILU goes to 1 as $\epsilon \rightarrow 0$, and with Neumann boundary conditions along $x=0, x=1$ MGD1 does not work (in that case the differential problem (6.2) is badly posed). Kettler (1982) gives an ILU-decomposition which does not suffer from this defect. With Dirichlet boundary conditions, however, MGD1 seems to be dependable for (ii), as is suggested by the following results of more extensive experiments with (ii); ρ_∞ is the spectral radius.

$\ell \backslash \epsilon$	0.5	10^{-1}	10^{-2}	10^{-4}
3	.038	.11	.042	.001
4	.091	.22	.19	.003
5	.10	.26	.41	.017
6	.10	.27	.55	.068

Table 6.3 Estimated ρ_∞ for $\phi_{xx} + \epsilon\phi_{yy} = 2+2\epsilon, \phi|_{\partial\Omega} = x^2+y^2$.

Comparison with the work of Hackbusch (1978) is relatively straightforward, because his method is similar to MGD1, the main differences being the use of two checkerboard (CH) or zebra-Gauss-Seidel relaxations (Z) for smoothing, and the use of 9-point prolongation and restriction. The main difference in computational work will be due to the difference in smoothing strategy. Assuming variable coefficients, two applications of CH take 18 operations per grid-point. Solution of a tri-diagonal system takes 8 operations, or 5 if the LU-decompositions are stored at a cost of 2 reals per grid-point. Including

the cost of the right-hand-sides we arrive at a cost of two applications of Z of 24, or 18 with extra storage, except for test-problem (iv), where the cost is 32 or 26. CH is not applicable to (ii), (iii), (iv). Recalling that the cost of ILU is 17 and the total cost of MGD1 is 30, we estimate the total cost of the methods used by Hackbusch to be 32 with CH, 42 with Z or 32 with extra storage, and 56 or 46 for test-problem (iv). From Hackbusch (1978b) we then deduce the following table.

	(i), CH	(i), Z	(ii), Z	(iv), Z
M, l	6,6	8,6	8,6	8,6
ρ, t	.048,24	.038,30(23)	.063,35(27)	.199,60(46)

Table 6.4 Computational cost of methods of Hackbusch (1978b).

CH: checkerboard-Gauss-Seidel; Z: zebra-Gauss-Seidel.

Between brackets: with extra storage.

Z is not applicable to (iii), because the lines are chosen in the x-direction. Also including y-lines would perhaps change the efficiency for (i) and (iv) little, but almost double the cost for (ii) and (iii). Because MGD1 and the methods of Hackbusch have much in common this is mainly a practical comparison of Gauss-Seidel and ILU smoothing methods.

The results of Brandt (1977) will not be discussed, because this early work has been extended by F,R,B,S, who give results of two-level analysis for a variety of combinations of restrictions, prolongations and smoothing processes. In the so-called W-cycle (with double the two-level cost) two-level analysis usually gives a good estimate for ρ_∞ . For (i) the best method (using CH smoothing and total-reduction concepts) in Ries et al. (1981) results in: $\rho_\infty=0.074$, cost=23.5, t=21. In the cost estimate the special values of the Poisson coefficients are exploited. The best method in Stüben et al. (1982) (using alternating direction Z smoothing) has for (i), (ii) and (iii):

$$\rho_\infty=0.023, \text{ cost}=51.6, \text{ t}=31.5, \text{ (i),}$$

$$\rho_\infty=0.119, \text{ cost}=51.6, \text{ t}=55.8, \text{ (ii) and (iii),}$$

with the cost estimate valid for variable coefficients. For (v) a detailed analysis is included by Börgers (1981); for the best method (using CH smoothing) for (v) with $\epsilon=10^{-5}$ on a 65x65 grid the following results emerge:

$\rho=0.41$, cost=28, $t=72$, (va),
 $\rho=0.37$, cost=28, $t=65$, (vc),

with ρ the average reduction factor over the last 14 of 20 iterations. When comparing with table 6.2 one has to keep in mind, that table 6.2 gives only the initial rate of convergence, observed during the first few iterations. For (va,b,c) this is irrelevant, because here MGD1 is almost exact. For comparison we give the following numerical estimate for (vd) with $\epsilon=10^{-5}$ on a 65×65 grid: $\rho_{\infty}=0.29$, $t=56$. This concludes our comparison with F,R,S,B.

Nicolaides (1979) reports experiments with two multigrid-finite-element methods. One of these, using "linear elements", results for (i) in the same system of equations on the finest grid that we are considering here. Prolongation, restriction and coarse grid approximation are much the same as in MGD1, but smoothing consists of a few Gauss-Seidel relaxations before and after coarse grid correction. CPU-time measurements are given in units of a Gauss-Seidel relaxation, for which we take a cost of 9 (assuming variable coefficients). On a 64×64 grid $t=35.1$ (cf. table 6.2) is reported, with a smooth initial guess, for (i) with right-hand-side zero. For (vi) Nicolaides (1979) reports experiments with bilinear elements only corresponding to a 9-point discretization. Here the cost of one relaxation is 17 (for variable coefficients). The following results are obtained.

k	2	4	8	16	32
N	76	82	99	107	104
MGD1	25	25	25	26	26

Table 6.5 Values of t for test-problem (vi)

Three multigrid iterations were carried out on a 65×65 grid; k is the coefficient in (6.6), t as in table 6.2, N stands for Nicolaides. The initial guess is uniformly randomly distributed. Because in this case there are significant differences between coarse-grid Galerkin and difference approximation, we have also tried MGD1 with finite differences; this is denoted as MGD1*.

On a 33×33 grid we have estimated the spectral radius ρ_{∞} , and obtain the following result.

k	8	16	32
MGD1	.31	.18	.13
MGD1*	.30	.30	>1

Table 6.4 Values of ρ_∞ for test-problem (iv).

As is to be expected, with coarse-grid finite difference approximation the method deteriorates as the rate of variation of the coefficients in the differential equation increases. This can probably be remedied by taking a suitable average of the coefficients when constructing coarse-grid finite difference approximations. The Galerkin approximation used in MGD1 does this automatically.

7. Final remarks.

A multigrid method (MGD1) has been presented, that is perceived by the user as any other linear systems solver, and requires no insight in the properties of multigrid methods. The user has to specify the matrix and the right-hand-side only. This is made possible by using Galerkin coarse-grid approximations and a fixed multigrid strategy, the so-called sawtooth cycle. The matrix should represent a 5- or a 7-point discretization of an elliptic equation on a rectangle in the usual way.

In cases with rapidly varying coefficients, the Galerkin method has the additional advantage of providing better coarse-grid approximations than straightforward finite differences.

MGD1 works well for a large variety of problems, including non-self-adjoint singularly perturbed equations, equations with a mixed derivative, and strongly anisotropic diffusion problems, except for a certain combination of anisotropic direction and Neumann boundary conditions. This restriction is removed, and the method is generalized to more general domains and discontinuous coefficients by Kettler and Meijerink (1981), cf. Kettler (1982).

Comparison with available results for other methods shows, that the robustness of MGD1 and the abandoning of any form of adaptivity is not paid for in terms of efficiency. In fact, at the moment MGD1 seems generally to be somewhat faster than other, more specialized methods, except when these exploit special values of the coefficients.

In the near future even more efficient methods may evolve, because it seems unlikely that the full potential of the various methods is already fully exhausted. The combination of conjugate gradient and multigrid methods (Kettler and Meijerink (1981)) is very promising, cf. Kettler (1982). The structure of multigrid methods is very rich. Various elements of the method can be chosen in many ways; smoothing process, prolongation, restriction, coarse-grid approximation, multigrid cycle. One can speak of a multigrid "philosophy", or perhaps better of a multigrid methodology. No single "best" or "standard" method is likely to emerge, just as in other areas of numerical mathematics.

References

- G.P. Astrachancev, An iterative method of solving elliptic net problems. USSR Comp. Math. Math. Phys. 11, 2, 171-182, 1971.
- N.S. Bakhvalov, On the convergence of a relaxation method with natural constraints on the elliptic operator, USSR Comp. Math. Math. Phys. 6, no.5, 101-135, 1966.
- C. Börgers, Private communication: Mehrgitterverfahren für eine Mehrstellendiskretisierung der Poisson-Gleichung und für eine zweidimensionale singular gestörte Aufgabe. Diplomarbeit (prof. Trottenberg), 1981.
- D. Braess, The contraction number of a multigrid method for solving the Poisson equation. Num. Math. 37, 387-404, 1981.
- A. Brandt, Multi-level adaptive technique (MLAT) for fast numerical solution to boundary value problems. In: Proc. Third Int. Conf. Num. Meth. Fluid Dyn., Paris, 1972, Lect. Notes in Phys. 18, 82-89, Springer-Verlag, Berlin etc. 1973.
- A. Brandt, Multi-level adaptive solutions to boundary-value problems, Math. Comp. 31 333-390, 1977.
- A. Brandt, Multi-level adaptive solutions to singular perturbation problems. In: Numerical analysis of singular perturbation problems (P.W. Hemker, J.J. Miller, eds.), 53-142, Academic Press, New York, 1979.
- A. Brandt and N. Dinar: Multi-grid solution to elliptic flow problems. In: Numerical methods for partial differential equations (S.V. Parter, ed.), 53-149. New York etc., Academic Press, 1979.
- A.R. Curtis, On a property of some test equations for finite difference or finite element methods. IMA J. Num. Anal. 1, 369-375, 1981.
- R.P. Fedorenko, A relaxation method for solving elliptic difference equations, USSR Comp. Math. Math. Phys. 1, 1092-1096, 1962.
- H. Foerster, K. Stüben, U. Trottenberg, Non-standard multigrid techniques using checkered relaxation and intermediate grids. In: Elliptic problem solvers (M. Schulz, ed.), 285-300, Academic Press, New York etc., 1981.
- P.O. Frederickson, Fast approximate inversion of large sparse linear systems, Mathematics report 7-75, Lakehead University, Thunder Bay, Canada, 1975.
- I. Gustafsson, A class of first order factorization methods. BIT 18, 142-156, 1978.
- W. Hackbusch, A fast iterative method for solving Poisson's equation in a general region. In: Numerical treatment of differential equations,

- Oberwolfach 1976 (R. Bulirsch, R.D. Grigorieff, J. Schröder, eds.).
Lecture Notes in Math. 631, Springer-Verlag, Berlin etc., 1978a.
- W. Hackbusch, On the multigrid method applied to difference equations,
Computing 20 pp. 291-306, 1978b.
- W. Hackbusch, On the convergence of multi-grid iterations. Beiträge zur
Numerischen Mathematik 2, 213-239, 1981.
- P.W. Hemker, The incomplete LU-decomposition as a relaxation method in multi-
grid algorithms. In: Boundary and interior layers - computational and
asymptotic methods, Proceedings, Dublin 1980, (J.J.H. Miller, ed.), Boole-
Press, Dublin 1980a.
- P.W. Hemker, Fourier analysis of gridfunctions, prolongations and
restrictions. Mathematical Centre, 413 Kruislaan, Amsterdam, Report
NW 93/80, 1980b.
- P.W. Hemker, Introduction to multigrid methods, Nieuw Archief v. Wiskunde 29,
71-101, 1981.
- A.M. Il'in, Differencing scheme for a differential equation with a small
parameter affecting the highest derivative. Math. Notes Acad. Sc. USSR 6,
596-602, 1969.
- R. Kettler, in the present volume (1982).
- R. Kettler and J.A. Meijerink, A multigrid method and a combined multigrid-
conjugate gradient method for elliptic problems with strongly discontinuous
coefficients in general domains. KSEPL, Volmerlaan 6, Rijswijk, The
Netherlands, Publication 604, 1981.
- J.A. Meijerink and H.A. van der Vorst, An iterative solution method for linear
systems of which the coefficient matrix is a symmetric M-matrix, Math.
Comp. 31 pp. 148-162, 1977.
- W.J.A. Mol, On the choice of suitable operators and parameters in multigrid
methods, Report NW 107/81, Mathematical Centre, Amsterdam, 1981.
- R.A. Nicolaides, On some theoretical and practical aspects of multigrid
methods. Math. Comp. 33, pp. 933-952, 1979.
- M. Ries, U. Trottenberg, G. Winter, A note on MGR methods. Universität Bonn,
preprint no. 461, 1981.
- U. Schumann, Fast elliptic solvers and their application in fluid dynamics.
In: W. Kollmann (ed.), Computational fluid dynamics, Hemisphere, Washington
etc., 1980.
- K. Stüben, C.A. Thole, U. Trottenberg, private communication, 1982.
- H.A. van der Vorst, Iterative solution methods for certain sparse linear
systems with a non-symmetric matrix arising from PDE-problems. J. Comp.
Phys. 44, 1-20, 1981.
- E.L. Wachspress, A rational finite element basis, chapter 10. Academic Press,
New York, 1975.
- P. Wesseling, Numerical solution of the stationary Navier-Stokes equations by
means of a multiple grid method and Newton iteration. Report NA-18, Delft
University of Technology, 1977.
- P. Wesseling, Theoretical and practical aspects of a multigrid method, Report
NA-37, Delft University of Technology, 1980.
- P. Wesseling and P. Sonneveld, Numerical experiments with a multiple grid and
a preconditioned Lanczos type method. In: Approximation methods for Navier-
Stokes problems, Proceedings, Paderborn 1979 (R. Rautmann, ed.), Lecture
Notes in Math. 771, 543-562, Springer-Verlag, Berlin etc. 1980.