

MULTIGRID METHODS: FUNDAMENTAL ALGORITHMS, MODEL PROBLEM ANALYSIS AND APPLICATIONS

Klaus Stüben*
Ulrich Trottenberg**

*Gesellschaft für Mathematik und Datenverarbeitung
Postfach 1240, D-5205 St. Augustin 1, Germany

**Institut für Angewandte Mathematik, Universität Bonn,
Wegeler Straße 6, D-5300 Bonn, Germany

Contents

PART I. MULTIGRID IDEA

1. Introduction
 - 1.1 Historical remarks and current perspectives
 - 1.2 Contents of this paper, acknowledgements
 - 1.3 Some notations
 - 1.3.1 Continuous boundary value problems
 - 1.3.2 Discrete boundary value problems
 - 1.3.3 Model problem (P)
 - 1.3.4 General difference stars on rectangular grids
 - 1.3.5 Restriction and interpolation operators
 - 1.3.6 Some remarks on the parameter h and admissible meshsizes
2. The multigrid idea, multigrid components
 - 2.1 Iteration by approximate solution of the defect equation
 - 2.2 Relaxation and coarse-grid correction
 - 2.3 Structure of an (h,H) two-grid operator
 - 2.4 Some specifications and extensions
 - 2.4.1 Choice of the coarser grid
 - 2.4.2 Choice of the coarse-grid difference operator
 - 2.4.3 More general smoothing procedures
 - 2.4.4 Coarse-to-fine transfer using the grid equation
 - 2.4.5 More general treatment of boundary conditions
3. Analysis of a sample $(h,2h)$ two-grid method for Poisson's equation
 - 3.1 An $(h,2h)$ -algorithm
 - 3.2 The relaxation operator
 - 3.3 The coarse-grid-correction operator
 - 3.4 Spectral radius of the two-grid operator
 - 3.5 Norms of the two-grid operator
 - 3.6 Algorithmic variants
 - 3.6.1 Use of straight injection for the fine-to-coarse transfer
 - 3.6.2 Jacobi ω -relaxation with several parameters

PART II. FUNDAMENTAL ALGORITHMS

4. Complete multigrid cycle
 - 4.1 Notation, sequence of grids and operators
 - 4.2 Recursive definition of a complete multigrid cycle
 - 4.3 The iteration operator for a complete multigrid operator, h -independent convergence
 - 4.4 Computational work and efficiency
 - 4.5 Other coarse-grid operators, extensions
5. Nonlinear multigrid methods, the full approximation scheme (FAS)
 - 5.1 Indirect application of multigrid methods to nonlinear problem
 - 5.2 The full approximation scheme
 - 5.3 A simple example
 - 5.4 A remark on nonlinear relaxation methods
 - 5.5 Some additional remarks
 - 5.5.1 An exemplary bifurcation problem
 - 5.5.2 The (h,H) -relative truncation error and the dual view of multigrid methods
6. The full multigrid method (nested iteration)
 - 6.1 Idea and purpose
 - 6.2 Structure of the full multigrid method
 - 6.3 A simple theoretical result
 - 6.4 Computational work, some practical remarks

PART III: FOURIER ANALYSIS OF MULTIGRID METHODS

7. The concept of model problems analysis, smoothing and two-grid convergence factors
 - 7.1 Assumptions on the difference operator
 - 7.2 The (h,H) coarse-grid correction operator
 - 7.3 Smoothing operators
 - 7.4 Two-grid operator
 - 7.5 General definition of smoothing factors
 - 7.6 Modifications and extensions
8. Applications of model problem analysis
 - 8.1 Analytic results for an efficient two-grid method
 - 8.2 Further results for Poisson's equation
 - 8.3 Results for the anisotropic model equation
9. Local Fourier analysis and some general theoretical approaches
 - 9.1 Purpose and formal tools of local Fourier analysis
 - 9.2 Applications of local Fourier analysis
 - 9.3 A short discussion of other theoretical approaches
 - 9.3.1 Splitting of the two-grid operator norm into a product
 - 9.3.2 Splitting of the two-grid operator norm into a sum
 - 9.3.3 Further remarks on the definition of "low" and "high"

PART IV. STANDARD APPLICATIONS

10. Multigrid programs for standard applications
 - 10.1 Description of domains and discretization
 - 10.2 Helmholtz equation (with variable c)
 - 10.3 Anisotropic operators
 - 10.4 More general situations

11. Multigrid methods on composite meshes
 - 11.1 Composite mesh discretization and a "naive" multigrid approach
 - 11.2 A "direct" multigrid method for composite meshes
 - 11.3 Some results for a model problem

Appendix: A sample multigrid program (FORTRAN listing)

References

1. Introduction

This paper gives a systematic introduction to multigrid methods for the solution of elliptic differential equations. The paper is based on the two introductory lectures held by the authors on the occasion of the "Conference on Multigrid Methods". It includes basic ideas (Part I) and fundamental methodical approaches (Part II), theoretical approaches (Part III) and simple applications (Part IV). The paper is to be seen in the context of the two other introductory papers in which Wolfgang Hackbusch outlines his general theory of multigrid methods and Achim Brandt gives a guide to the practical realization of multigrid methods. Brandt's paper deals, in particular, with problems of a more general type (systems of differential equations in higher dimensions) than that of the problems we consider in our paper. Brandt also discusses more sophisticated multigrid techniques.

Although our description of the multigrid principle and of the fundamental methodical approaches is quite general, the concrete considerations in this paper refer - in accordance with its introductory character - to a limited class of problems: We explicitly treat only scalar equations in two dimensions; the underlying discretizations are based on finite difference methods. Mostly we are concerned with second order Dirichlet boundary value problems. Most of these restrictions, in particular the restriction to two dimensions, are mainly for the sake of technical simplification.

In this introduction we give a short survey of the development of multigrid methods and on the state of the art (Section 1.1). We will then describe contents and objectives of this paper in some more detail (Section 1.2). In Section 1.3, we will introduce some fundamental notation which is needed.

1.1 Historical remarks and current perspectives

Multigrid history. The multigrid principle (for discrete elliptic boundary value problems) is extremely simple: Approximations with *smooth errors* are obtained very efficiently by applying suitable *relaxation methods*. Because of the error smoothness, corrections of these approximations can be calculated on *coarser grids*. This basic idea can be used *recursively* employing coarser and coarser grids. This leads then to "*(asymptotically) optimal*" iterative methods, i.e. methods for which the computational work required for achieving a fixed accuracy is proportional to the number of discrete unknowns. If the multigrid methods are then combined with the idea of *nested iteration* (use of coarser grids to obtain good initial approximations on finer grids), a suitable algorithmization even yields methods for which the computational work required for achieving the discretisation accuracy is still proportional to the number of discrete unknowns.

Consequently, we may distinguish three elements (stages):

- (1) error smoothing by relaxation,
- (2) calculation of corrections on coarser grids and recursive application,
- (3) combination with nested iteration.

Looking back on the development of multigrid methods we see that the above elements, if considered separately, have already been known or used for a long time before they were combined to efficient multigrid methods. Especially the error smoothing effect of relaxation methods belongs to the classical inventory of numerical knowledge. The idea to use this effect for convergence acceleration can already be found in the early literature (e.g. Southwell [92], [93]; Stiefel [94]). However, the recursive use of coarser grids is not yet elaborate there. But it is only this recursion which gives the above mentioned "optimality".

On the other hand, the recursive application of coarser grids for an efficient solution of specific discrete elliptic boundary value problems was used in the context of "*reduction methods*" introduced by Schröder [86] (see also [85], [87], [88]). Here, however, no explicit error smoothing is performed. Elimination techniques are used instead which transform the original problem "*equivalently*" to coarser grids. (These elimination techniques restrict the range of direct application of reduction methods to a small class of problems.)

Finally, the self-suggesting idea of nested iterations has in principle been known for a long time.

The first studies introducing and investigating multigrid methods in a narrow

sense (elements (1) and (2)) are those by Fedorenko [34], [35] and then that of Bakhvalov [6]. While in [35] Fedorenko restricts the convergence investigation to the Poisson equation in the unit square, Bakhvalov [6] discusses general elliptic boundary value problems of second order with variable coefficients (in the unit square). Bakhvalov also indicates the possibility of combining multigrid methods with nested iteration (element (3)).

Though the studies published by Fedorenko and Bakhvalov have, in principle, shown the asymptotic optimality of the multigrid approach (and to a certain extent its generality as well), their actual efficiency is first recognised only by Achi Brandt (by 1970). Studying adaptive grid refinements and their relation to fast solvers, Brandt has been led to the papers of Fedorenko and Bakhvalov through information given by Olof Widlund. In the first two papers [15], [16] and later on summarised in the systematic work [17], Brandt shows the actual efficiency of multigrid methods. His essential contributions (in the early studies) concern the introduction of non-linear multigrid methods ("FAS-scheme") and adaptive techniques ("MLAT"), the discussion of general domains and local grid refinements, the systematic application of the nested iteration idea ("full multigrid" FMG) and - last but not least - the provision of the tool of the "local Fourier analysis" for theoretical investigation and method optimisation.

Representative for the further multigrid development are the following papers which we would like to mention as being "historically" relevant contributions.

In [4] Astrakhantsev generalises Bakhvalov's convergence result to general boundary conditions; like Bakhvalov he uses a variational formulation in his theoretical approach. - In [39], Frederickson introduces an approximate multigrid-like solver which can be regarded as a forerunner of the "MGR methods", which were developed later on. - After a first study of multigrid methods for Poisson's equation in a square [75], Nicolaides discusses multigrid ideas in connection with finite element discretisations systematically in [76]. -

In the years 1975/76, Hackbusch develops the fundamental elements of multigrid methods anew without having knowledge of the existing literature. It is again Olof Widlund who informs Hackbusch about the studies which are already available. Hackbusch's first systematic report [42] contains many theoretical and practical investigations which have been taken up and developed further by several authors. So one finds considerations of the "model problem analysis" type, the use of "red black" and "four colour" relaxation methods for smoothing, the treatment of non-rectangular domains and of nonlinear problems etc. In the papers [43], [45], [49], Hackbusch then presents a general convergence theory of multigrid methods.

The recent development. Since about 1977 multigrid methods have increasingly gained broad acceptance. This more recent development shall not be described here in detail. (A survey of the literature presently available is given by the multi-grid bibliography in this Proceedings.) However, we want to mention some important fields of applications and mathematical areas to which multigrid methods have been applied and extended. The field of finite elements which has first been of a more theoretical interest to multigrid methods (see, for example, [76], [43], [8]) is now undergoing an intensive practical investigation (see, for example, [9], [32]). Apart from linear and non-linear boundary value problems (scalar equations and systems) eigenvalue problems and bifurcation problems (see, for example, [44], [27], [73]) are treated as well. Parabolic (see, for example, [33], [90], [63]) and other time-dependent and non-elliptic problems (see e.g. [23], [22], [84]) are attracting more and more interest. All these situations occur in numerical fluid dynamics, probably the most challenging field for multigrid methods. Here the studies are now concentrating on singular perturbation phenomena, transonic flow, shocks, the treatment of Euler equations and of the full Navier Stokes equations.

Apart from differential equations, integral equations can also be efficiently solved by multigrid methods (see e.g. [25] and the whole complex of multigrid methods "of the second kind" [48], [57]). Furthermore, multigrid-like methods are also being suggested for the solution of special systems of equations without continuous background [25]. A certain amount of multi-level structure (at least the nested iteration idea) can also be found in algorithms used in pattern recognition.

Perhaps as important as the extension of the field of applications of multigrid methods is the combination of the multigrid idea with other numerical and more general mathematical principles. In this context we would like to mention the combination with *extrapolation* and *defect correction methods* (see e.g. [25], [5], [51], [56]). Finally, there are considerations which refer to the optimal use of multigrid methods on *vector* and *parallel computers* (and the construction of corresponding multigrid components) (see, for example, [24]) as well as to approaches within computer architecture concerning a direct mapping of the multigrid principle onto a suitable - perhaps *pyramidal* - multiprocessor structure (see corresponding remarks in [103]).

Delayed acceptance, resentments. The historical survey has shown that the acceptance of multigrid methods was first a rather troublesome process. Only the rapid development of recent years has convinced most people working in the field of numerical methods for partial differential equations of the sensational possibi-

lities provided by the multigrid principle.

Nevertheless, even today's situation is still unsatisfactory in several respects. If this is true for the development of standard methods, it applies all the more to the area of really difficult, complex applications. With respect to standard applications, we would like to discuss this in some detail (since this area is in the center of this introductory paper) and with respect to the complex applications, for example in fluid dynamics, we would like to confine ourselves to some remarks.

As far as standard problems (simple elliptic 2D problems of second order) are concerned, the opinion prevailed for a long time - even and just among experts - that, despite of their "asymptotic optimality", multigrid methods were in reality far from being as efficient as the "direct fast solvers" (such as the Buneman algorithm [29] or the method of total reduction [88]) and their combination with capacitance matrix techniques [81]. Only by providing generally available programs (such as MG00, MG01, see chapter 10), has it been proved in practice that suitable multigrid methods are at least competitive in these areas as well. The decisive advantage of multigrid methods is however that they can be applied easily to problems which do not meet - or do not fully meet - the requirements demanded by direct fast solvers and capacitance matrix techniques.

Doubts in the high efficiency of multigrid methods were also fed by the multi-grid convergence theories. The abstract theories are often far too pessimistic and do usually not provide constructive criteria for the construction of optimal methods for concrete situations (see also Section 9.3). Only the *model problem analysis* (see Chapters 3, 7 and 8) and *local Fourier analysis* (see Sections 9.1, 9.2) yield quantitative results to be used for the construction of algorithms. On the other hand, these theoretical approaches, being relatively simple from the mathematical viewpoint, also have disadvantages: The model problem analysis can be applied directly to a small class of problems only, and local Fourier analysis is based on idealising assumptions.

As a consequence, even in the field of standard applications the disagreement about which approach would really supply the "best" or the "most robust" algorithms, is not completely settled as yet. For example, as far as the smoothing methods are concerned, each expert recommends "his" method and emphasises its benefits (A.Brandt recommends standard relaxation techniques - pointwise, linewise and "distributed"; Wesseling the ILU smoothing, Jameson smoothing methods of the ADI type, we recommend MGR methods....). Since so far systematic and fair comparisons were hardly available, it was also impossible, until recently, to obtain reliable statements on which method should be preferred in which situation. Among users this

confusion has led to misunderstandings and false conclusions.

While in the field of standard problems the differences in efficiency shown by the various algorithms are, after all, not very large and the disagreement previously mentioned is therefore of a more or less academic nature, the disagreement in the field of non-elementary applications is of direct practical importance and it has especially unpleasant consequences there.

Such a controversy exists, for example, in the field of fluid dynamics between many numerical practitioners who like to take up multigrid methods and multigrid experts (even among the practically oriented experts) who like to develop "optimal" methods from a more fundamental viewpoint. With respect to more complex problems the experts usually supply efficient algorithms for simplified situations only and do not go to the work of solving full fledged industrial problems. The practitioners are therefore sceptical about the full applicability of multigrid methods. They mostly prefer to include single multigrid components in certain parts of available software. Thus, they obtain improvements which are possibly rather impressive, but, on the other hand, they regard their scepticism as being justified since the improvements obtained are far from being as large as predicted for "optimal" methods. However the multigrid experts also feel justified since they regard the stepwise inclusion of multigrid elements in the available "non-multigrid software" as being unsatisfactory in any case. This discrepancy can be found in many publications and comments and it was also reflected on the conference which is the subject of these proceedings. There is some hope, that these proceedings contribute towards bridging the gap between multigrid experts and practitioners.

1.2. Contents of this paper, acknowledgements

In part I, we describe the multigrid idea (Chapter 2) and give a first analysis of a sample method for Poisson's equation. For both chapters we have intentionally chosen a very detailed and elementary representation. The sample method considered in Chapter 3 is a rather inefficient method (since Jacobi relaxation is used for smoothing), but it has the advantage of being particularly theoretically transparent. The theoretical considerations and the tools introduced in Chapter 3 are characteristic for the model problem analysis which is discussed more systematically in part III.

Part II (Chapters 4,5,6) describes the well-known fundamental multigrid techniques: the *recursively defined complete multigrid cycle* (Chapter 4), the *non-linear full approximation scheme* (Chapter 5), and the *full multigrid method* (Chapter 6).

Parts III and IV, in particular Chapters 7,8 (together with Chapter 3) and 10, 11, inform about results which are largely new and have not been published as yet.

Part III discusses the concepts of the so-called *model problem analysis* and *local Fourier analysis*. For a certain class of model problems and a certain class of multigrid algorithms, it is possible to give exact statements (not estimates) on the convergence behaviour of the method in question using basic tools of discrete Fourier analysis. In Chapter 7, we introduce the required formalism. In this context, various cases of the coarse grid definition are discussed.

Readers who are interested in concrete results rather than in the technically quite complicated formalism should proceed to Chapter 8. All results in this chapter refer to *standard coarsening* (doubling the meshwidths); the emphasis lies on the discussion of efficient smoothing methods, namely on *RB* (= *red black*), *ZEBRA*, and *alternating ZEBRA relaxation*. Within the class of methods discussed, the model problem analysis allows the construction of optimal multigrid components.

Problems and methods which can no longer be treated rigorously by model problem analysis may possibly be studied by means of Fourier analysis (Chapter 9). In this context, however, no exact statements on the problem given are obtained but only statements on an idealised problem (and thus on an idealised method) where, in particular, the influence of the boundary and the boundary conditions are neglected. The exact statements on the idealised problem (and method) are then regarded as approximate statements on the original problem (and method). Subjects of this idealizing local Fourier analysis are, for example, the usual *Gauß-Seidel-relaxation* method (with *lexicographic ordering* of the grid points) and *ILU-smoothing*. Among other things, we make a short comparison of ILU-smoothing with ZEBRA relaxation in Section 9.2. - In Section 9.3., we make some remarks on more abstract convergence theories.

On the basis of the model problem and local Fourier analysis, the programs MG00 and MG01 for elliptic "standard problems" have been developed. MG01 is described in Chapter 10. - Chapter 11 describes the possibility of applying multigrid methods in combination with simultaneous use of various coordinate systems to a given problem (composite mesh system).

This is not the first introductory paper to multigrid methods (see [17], [55], [52]). In our presentation, the emphasis lies on the theoretical and practical discussion of the following central problem: How are the different multigrid components to be chosen in concrete situations? Clearly, there are different possible objectives which can be pursued in answering this question, e.g. efficiency, simplicity or/and robustness of the respective algorithms. In our paper we tend toward demonstrating the efficiency of multigrid methods (for standard applications) rather than their generality. This shall, however, by no means modify or question the generality of the principle.

Acknowledgements

For various pleasant discussions concerning this paper (or certain parts of its contents) we would like to thank Achi Brandt, Wolfgang Hackbusch, Theodor Meis, • Olof Widlund, and Kristian Witsch.

Christoph Börgers and Clemens August Thole read the manuscript and checked proofs and examples. Kristian Witsch supported us in providing the sample program. Kurt Brand and Horst Schwichtenberg accomplished several technical tasks.

Rudolph Lorentz corrected our use of the English language and also Ursula Bernhard supported us in this respect.

Gertrud Jacobs typed the manuscript, never tiring in making subsequent changes. She was supported partly by Elisabeth Harf. Maria Heckenbach drew the illustrating figures.

We owe sincere thanks to all of them.

1.3 Some notation

In this section, we want to list the basic notation needed for our description of discrete elliptic problems and their multigrid treatment. (Most of the notation below will - for clarity - be shortly explained once more when it occurs in the paper for the first time.)

1.3.1 Continuous boundary value problems

Linear boundary value problems are denoted by

$$\begin{aligned} L^\Omega u &= f^\Omega(x) \quad (x \in \Omega) \\ L^\Gamma u &= f^\Gamma(x) \quad (x \in \Gamma := \partial\Omega). \end{aligned} \tag{1.1}$$

Here $x = (x_1, \dots, x_d)$ and Ω is a given domain with boundary Γ . L^Ω is a linear (elliptic) differential operator on Ω and L^Γ stands for one or several linear boundary operators. f^Ω denotes a given function on Ω and f^Γ one or several functions on Γ . Solutions of (1.1) are always denoted by $u = u(x)$. For brevity, we also write simply $Lu = f$ instead of (1.1). All concrete considerations refer to the case $d=2$.

Nonlinear differential operators are denoted by L rather than L .

1.3.2 Discrete boundary value problems

For discrete problems, we use the terminology of *grid functions*, *grid operators* and *grid equations* (rather than matrix terminology). The discrete analog of (1.1) is denoted by

$$\begin{aligned} L_h^\Omega u_h &= f_h^\Omega(x) \quad (x \in \Omega_h) \\ L_h^\Gamma u_h &= f_h^\Gamma(x) \quad (x \in \Gamma_h). \end{aligned} \tag{1.2}$$

h is a (formal) discretization parameter here. The discrete solution u_h is a grid-function defined on $\Omega_h \cup \Gamma_h$. f_h^Ω and f_h^Γ are discrete analogs of f^Ω and f^Γ . L_h^Ω and L_h^Γ are grid operators, i.e. mappings between spaces of grid functions. (L_h^Ω is also called a *discrete* or *difference operator*, L_h^Γ a *discrete boundary operator*.)

For simplicity, we will usually assume that the discrete boundary equations are eliminated from (1.2). This is, for example, quite natural in case of second order equations with Dirichlet boundary conditions (for an example, see Section 1.3.3). We then simply write

$$L_h u_h = f_h \quad (\Omega_h). \quad (1.3)$$

Here u_h and f_h are grid functions on Ω_h and L_h is a linear operator

$$L_h : \mathcal{G}(\Omega_h) \rightarrow \mathcal{G}(\Omega_h) \quad (1.4)$$

where $\mathcal{G}(\Omega_h)$ denotes the linear space of grid functions on Ω_h . Clearly, (1.3) represents a system of linear algebraic equations. We consider it, however, as one *grid equation*.

At many places in this paper, Ω is a rectangular domain $\Omega = (0, A_1) \times (0, A_2)$ and Ω_h a rectangular grid "matching well" with Ω . In this case, h stands for a vector of meshsizes: $h = (h_{x_1}, h_{x_2})$ and Ω_h is described by

$$\Omega_h := \Omega \cap G_h \quad (1.5)$$

where G_h denotes the infinite grid

$$G_h := \{x = \kappa \cdot h : \kappa \in \mathbb{Z}^2\}, \quad h_{x_j} = A_j / N_j, \quad N_j \in \mathbb{N} \quad (j = 1, 2). \quad (1.6)$$

Here $\kappa \cdot h := (\kappa_1 h_{x_1}, \kappa_2 h_{x_2})$. In the special case of square grids, we simply write $h = h_{x_1} = h_{x_2}$.

For (1.5) and (1.6), the space of grid functions $\mathcal{G}(\Omega_h)$ is canonically endowed with the *Euclidian* (more precisely, the *discrete L_2 -*) *inner product*

$$(u_h, w_h)_2 := \frac{1}{N_1 N_2} \sum_{x \in \Omega_h} u_h(x) \bar{w}_h(x), \quad \text{if } \|u_h\|_2 := \sqrt{(u_h, u_h)_2}. \quad (1.7)$$

The corresponding operator norm is the *spectral norm*, denoted by $\|\cdot\|_S$. For L_h symmetric and positive definite, we also consider the *energy inner product*

$$(u_h, w_h)_E := (L_h u_h, w_h)_2 \quad (1.8)$$

and the corresponding operator norm $\|\cdot\|_E$.

1.3.3 Model problem (P)

For demonstration purposes, we sometimes refer to the simple case of Poisson's equation with Dirichlet boundary conditions on the unit square, namely

$$\begin{aligned} L^\Omega u &:= -\Delta u = f^\Omega(x) \quad (x \in \Omega := (0, 1)^2) \\ L^\Gamma u &:= u = f^\Gamma(x) \quad (x \in \Gamma). \end{aligned} \quad (1.9)$$

We speak of *model problem (P)*, if this problem is discretized on a square h -grid

using the ordinary 5-point approximation (with order of consistency 2). In particular, we then have

$$L_h^\Omega = -\Delta_h \triangleq \frac{1}{h^2} \begin{bmatrix} & -1 \\ -1 & 4 & -1 \\ & -1 \end{bmatrix}_h ,$$

$$\Omega_h = \Omega \cap G_h, \quad G_h = \{x = \kappa \cdot h : \kappa \in \mathbb{Z}^2\}, \quad h = 1/N \quad (N \in \mathbb{N}). \quad (1.10)$$

(The notation of difference stars used here is described in more detail below.)

Eliminating the discrete boundary conditions leads to a grid equation (1.3). More precisely, L_h in (1.3) is then given by the difference star in (1.10) degenerating to a 4-point star near the edges and to a 3-point star near the corners. Clearly, the right hand side f_h^T then also includes the boundary values: certain terms of the form f_h^T/h^2 have to be added to f_h^Ω at grid points near the boundary. Whenever we refer to model problem (P), we mean the corresponding grid equation (1.3).

Another important discrete problem considered in this paper, is the *anisotropic model problem*. The difference to model problem (P) is that L^Ω in (1.9) is replaced by

$$L^\Omega u := -\varepsilon u_{x_1 x_1} - u_{x_2 x_2}. \quad (1.11)$$

The standard 5-point approximation of L^Ω is given by

$$L_h^\Omega \triangleq \frac{1}{h^2} \begin{bmatrix} & -1 \\ -\varepsilon & 2(1+\varepsilon) & -\varepsilon \\ & -1 \end{bmatrix}_h . \quad (1.12)$$

1.3.4 General difference stars on rectangular grids

For the concrete definition of discrete operators L_h^Ω (on rectangular grids) the terminology of *difference stars* is convenient. We will make use of this terminology throughout this paper. For a simplified introduction of this terminology, we make use of the infinite grid in (1.6). (Note that the following definitions can also be understood locally for fixed $x \in \Omega_h$.)

A general difference approximation at some $x \in G_h$ is of the form

$$L_h^\Omega w_h(x) = \sum_{\kappa \in V} s_\kappa w_\kappa(x + \kappa \cdot h) \quad (1.13)$$

where \mathbb{V} denotes a certain finite subset of \mathbb{Z}^2 (containing $(0,0)$). In the terminology of difference stars this is written as

$$L_h^\Omega w_h(x) = [s_\kappa]_h w_h(x) := \begin{bmatrix} & \cdot & \cdot & \cdot & \cdot \\ & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & s_{-1,1} & s_{0,1} & s_{1,1} & \cdot & \cdot \\ \cdot & \cdot & s_{-1,0} & s_{0,0} & s_{1,0} & \cdot & \cdot \\ \cdot & \cdot & s_{-1,-1} & s_{0,-1} & s_{1,-1} & \cdot & \cdot \\ & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \end{bmatrix}_h w_h(x) \quad (1.14)$$

The coefficients s_κ depend, of course, on h and possibly also on x . In our theoretical considerations, however, the s_κ do not depend on x . We will mainly consider discrete operators L_h^Ω on rectangular domains which can be described by 5-point and compact 9-point stars

$$\begin{bmatrix} & s_{0,1} & \\ s_{-1,0} & s_{0,0} & s_{1,0} \\ & s_{0,-1} & \end{bmatrix}_h, \quad \begin{bmatrix} s_{-1,1} & s_{0,1} & s_{1,1} \\ s_{-1,0} & s_{0,0} & s_{1,0} \\ s_{-1,-1} & s_{0,-1} & s_{1,-1} \end{bmatrix}_h. \quad (1.15)$$

L_h^Ω is then identified with its difference star:

$$L_h^\Omega \triangleq [s_\kappa]_h. \quad (1.16)$$

We sometimes also identify L_h in (1.3) with the difference star corresponding to L_h^Ω . This always means that near boundaries the star is assumed to be properly modified (as described above for model problem (P)).

1.3.5 Restriction and interpolation operators

Apart from discrete operators L_h , we need restriction and interpolation operators for the intergrid transfer of grid functions in the multigrid context. For their description, we use a star terminology also. We illustrate this for the grid transfer between the grid G_h and the grid corresponding to the meshsize $2h$, namely

$$G_{2h} = \{x = 2\kappa \cdot h : \kappa \in \mathbb{Z}^2\}. \quad (1.17)$$

A *restriction operator* I_h^{2h} maps h -grid functions into $2h$ -grid functions:

$$(I_h^{2h} w_h)(x) = \sum_{\kappa \in \mathbb{V}} \hat{t}_\kappa w_h(x + \kappa \cdot h) \quad (x \in G_{2h}). \quad (1.18)$$

Here \mathbb{V} again denotes some finite subset of \mathbb{Z}^2 . The coefficients \hat{t}_κ may depend on h and x . Throughout this paper, however, the \hat{t}_κ are constants (depending neither on h nor on x). For (1.18) we write

$$I_h^{2h} \triangleq [\hat{t}_\kappa]_h^{2h} := \begin{bmatrix} \cdot & \cdot & \cdot & & \\ \cdot & \hat{t}_{-1,1} & \hat{t}_{0,1} & \hat{t}_{1,1} & \cdot \cdot \\ \cdot & \hat{t}_{-1,0} & \hat{t}_{0,0} & \hat{t}_{1,0} & \cdot \cdot \\ \cdot & \hat{t}_{-1,-1} & \hat{t}_{0,-1} & \hat{t}_{1,-1} & \cdot \cdot \\ \cdot & \cdot & \cdot & & \\ \cdot & \cdot & \cdot & & \end{bmatrix} \begin{matrix} 2h \\ h \end{matrix} \quad (1.19)$$

The most frequently used restriction operator is the operator of *full weighting* (FW):

$$\frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} \begin{matrix} 2h \\ h \end{matrix} . \quad (1.20)$$

Similarly, an *interpolation (prolongation) operator* maps $2h$ -grid functions into h -grid functions. For the description of such operators, we introduce the following notation:

$$I_{2h}^h \triangleq [\hat{t}_\kappa]_{2h}^h := \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & & \\ \cdot & \cdot & \cdot & \cdot & & \\ \cdot & \hat{t}_{-1,1} & \hat{t}_{0,1} & \hat{t}_{1,1} & \cdot \cdot & \\ \cdot & \hat{t}_{-1,0} & \hat{t}_{0,0} & \hat{t}_{1,0} & \cdot \cdot & \\ \cdot & \hat{t}_{-1,-1} & \hat{t}_{0,-1} & \hat{t}_{1,-1} & \cdot \cdot & \\ \cdot & \cdot & \cdot & \cdot & & \\ \cdot & \cdot & \cdot & \cdot & & \end{bmatrix} \begin{matrix} h \\ 2h \end{matrix} \quad (1.21)$$

The meaning of this star terminology is that coarse-grid values are "distributed" to the fine grid weighted by \hat{t}_κ . More precisely, (1.21) means that a $2h$ -grid function w_{2h} is mapped into the h -grid function w_h defined by

$$w_h := \sum_{y \in G_{2h}} w_{2h,y} \quad (1.22)$$

where $w_{h,y}$ is the h -grid function (with finite support)

$$w_{h,y}(x) := \begin{cases} w_{2h}(y) \hat{t}_\kappa^\vee & \text{for } x = y + \kappa \cdot h \text{ with } \kappa \in \mathbb{V} \\ 0 & \text{for } x = y + \kappa \cdot h \text{ with } \kappa \notin \mathbb{V}. \end{cases} \quad (1.23)$$

Note that, for fixed $x \in G_h$, only a finite number of summands $w_{h,y}(x)$ in (1.22) is nonzero.

The most frequently used interpolation method is the one of *bilinear interpolation* from G_{2h} to G_h . The corresponding star is given by

$$\frac{1}{4} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} \begin{bmatrix} h \\ 2h \end{bmatrix} \quad (1.24)$$

Obviously, this interpolation operator corresponds to the FW restriction operator (1.20): these two operators are *adjoint* to each other (see [50], [109]). Without going into details, we mention that - in a certain sense - such a correspondence holds for general restriction and interpolation operators if $\hat{t}_k = t_{-k}/4$.

1.3.6 Some remarks on the parameter h and admissible meshesizes

So far, we have sometimes considered h as a formal parameter (also used for non-rectangular domains and meshes) and at other places as a vector of meshesizes $h=(h_{x_1}, h_{x_2})$ or as a scalar parameter of square meshes. Similarly, an index H is used to denote quantities on (or involving) any coarser grid Ω_H , e.g., L_H^H , I_H^H , I_H^h . In concrete cases, H denotes the meshsize of this coarser grid, e.g., $H=2h$ or $H=(2h_{x_1}, h_{x_2})$ etc. Even then, H is often considered as a formal parameter: For example, if L_h is given on Ω_h , then L_{2h} does not necessarily mean the $2h$ -discretization of L which corresponds to L_h ; in general, L_{2h} may be any discrete operator on Ω_{2h} . This use of the index $2h$ is somewhat inconsistent. It is, however, common and very convenient for reasons of technical simplicity. In concrete cases, it will always be clear which coarse grid quantities are actually meant.

In many places within this paper we consider certain important quantities as a function of h , e.g. asymptotic convergence factors $\rho(h)$. Of particular interest are their suprema with respect to h . These suprema have to be taken over those h which are meaningful and of interest within the multigrid context and the respective considerations. We denote the corresponding range of *admissible* h by \mathcal{H}^* . Thus we write, e.g.,

$$\rho^* := \sup \{\rho(h) : h \in \mathcal{H}^*\}. \quad (1.25)$$

Assuming rectangular meshes, the general form of \mathcal{H}^* is

$$\mathcal{H}^* := \{h : h \leq h^*, \quad h_{x_2}/h_{x_1} = q^*\}. \quad (1.26)$$

Here q^* denotes a given fixed meshsize ratio and h^* denotes the (vector of) maximal meshsizes of interest. The inequality in (1.26) has to be understood component-wise. Furthermore, " \leq " implicitly restricts the meshsizes h to those for which the process of constructing coarser grid at hand is meaningful. For example, if Ω is a rectangle and $H=2h$, we admit only those h for which both Ω_h and Ω_{2h} match well with the given domain. For square meshsizes (i.e. $q^*=1$), both h and h^* are scalar quantities. In the case of model problem (P), \mathcal{H}^* is defined by

$$\mathcal{H}^* := \{h : h \leq h^*, \quad h = 1/N \text{ (N even)}\}, \quad h^* := 1/4. \quad (1.27)$$

In all concrete cases considered in this paper, the precise meaning of \mathcal{H}^* will be clear.

2. The multigrid idea, multigrid components

In this section we describe the fundamental multigrid idea. For this purpose, we consider a discrete linear elliptic boundary value problem, formally given by

$$L_h u_h = f_h \quad (\Omega_h). \quad (2.1)$$

Here L_h is a linear operator

$$L_h : \mathbb{G}(\Omega_h) \rightarrow \mathbb{G}(\Omega_h), \quad (2.2)$$

and $\mathbb{G}(\Omega_h)$ denotes the linear space of grid functions on Ω_h . We assume Ω_h to consist of $N = N_h$ grid points corresponding to the unknown grid values of u_h . Thus $\mathbb{G}(\Omega_h)$ is of dimension N . Furthermore, we assume L_h^{-1} to exist.

As we are not going to give concrete quantitative results in this section, we do not make precise assumptions on the discrete operator L_h , the right hand side f_h and the grid Ω_h . A simple but characteristic example will be treated explicitly in Chapter 3.

In (2.1), we have assumed that the boundary conditions of the corresponding continuous problem (1.1) have been "eliminated". Thus f_h consists of some discrete analogue of the right hand side f^Ω as well as of the boundary values f^Γ . The assumption about the elimination of boundary conditions allows a simple description of the multigrid idea. We point out, however, that this treatment of the boundary conditions may not give the best multigrid methods in general cases. In certain simple cases (e.g. Dirichlet boundary conditions for second order equations) this approach is indeed suitable. In general, a separate treatment of the differential equation and the boundary conditions by multigrid techniques should be taken into consideration (see Section 2.4.5).

2.1 Iteration by approximate solution of the defect equation

Let u_h^j be any approximation of the solution u_h of (2.1). Then by

$$v_h^j := u_h - u_h^j \quad (2.3)$$

we denote the *error* of u_h^j (also regarded as *correction* of u_h^j), and by

$$d_h^j := f_h - L_h u_h^j \quad (2.4)$$

the *defect* (or *residual*) of u_h^j . Trivially, the *defect equation*

$$L_h v_h^j = d_h^j \quad (2.5)$$

is equivalent to the original equation, yielding

$$u_h = u_h^j + v_h^j.$$

The defect equation and approximations to it play an essential role in our description of the multigrid idea.

We begin the description by pointing out that most of the classical iterative methods for the solution of (2.1) can also be interpreted as approximations to (2.5): If in (2.5) L_h is replaced by any "simpler" operator \hat{L}_h such that \hat{L}_h^{-1} exists, the solution \hat{v}_h^j of

$$\hat{L}_h \hat{v}_h^j = d_h^j \quad (2.6)$$

gives a new approximation

$$u_h^{j+1} = u_h^j + \hat{v}_h^j.$$

Starting with some u_h^0 , the successive application of this process defines an iterative procedure. Obviously, the *iteration operator* of this method is given by

$$I_h - B_h L_h : \mathbb{G}(\Omega_h) \rightarrow \mathbb{G}(\Omega_h), \quad (2.7)$$

where $B_h := \hat{L}_h^{-1}$ and I_h denotes the identity on $\mathbb{G}(\Omega_h)$. We have

$$v_h^{j+1} = (I_h - B_h L_h) v_h^j \quad (j = 0, 1, 2, \dots) \quad (2.8)$$

for the errors and

$$d_h^{j+1} = L_h (I_h - B_h L_h) L_h^{-1} d_h^j = (I_h - L_h B_h) d_h^j \quad (j = 0, 1, 2, \dots) \quad (2.9)$$

for the defects.

It is a well-known fact that the asymptotic convergence properties of the above iterative process are characterized by the *spectral radius (asymptotic convergence factor)* of the iteration operator, i.e.

$$\rho(I_h - B_h L_h) = \max \{ |\lambda| : \lambda \text{ eigenvalue of } I_h - B_h L_h \}. \quad (2.10)$$

If some norm $\|\cdot\|$ is introduced in $\mathbb{G}(\Omega_h)$, the corresponding operator norms

$$\|I_h - B_h L_h\|, \|I_h - L_h B_h\| \quad (2.11)$$

give the *error reducing factor* and the *defect reducing factor*, respectively, per iteration step.

The following examples of iterative methods are of direct relevance to the multigrid method explained below:

Example 2.1a: The classical Jacobi method (or simultaneous displacement procedure or total step procedure) is characterized by replacing L_h by its "diagonal" part (in matrix terminology).

Example 2.1b: Similarly, the classical Gauss-Seidel method (or successive displacement procedure or single step procedure) is obtained by replacing L_h by its "upper triangular" part (in matrix terminology). As the structure of the matrix associated with L_h depends on the enumeration (ordering) of the grid points of Ω_h , \hat{L}_h also depends essentially on this enumeration,

Example 2.2: A quite different choice of \hat{L}_h (or, more precisely, of B_h), which will lead us to the multigrid idea, consists in using an appropriate approximation L_H of L_h on a coarser grid Ω_H . This means that the defect equation (2.5) is replaced by an equation

$$L_H \hat{v}_H^j = d_H^j. \quad (2.12)$$

Here we assume

$$L_H : \mathcal{G}(\Omega_H) \rightarrow \mathcal{G}(\Omega_H), \quad \dim \mathcal{G}(\Omega_H) \ll \dim \mathcal{G}(\Omega_h) \quad (2.13)$$

and L_H^{-1} to exist. As d_H^j and \hat{v}_H^j are grid functions on the coarser grid Ω_H , we assume two (linear) transfer operators

$$I_h^H : \mathcal{G}(\Omega_h) \rightarrow \mathcal{G}(\Omega_H), \quad I_H^h : \mathcal{G}(\Omega_H) \rightarrow \mathcal{G}(\Omega_h). \quad (2.14)$$

to be given. I_h^H is used to restrict d_h^j to Ω_H :

$$d_H^j := I_h^H d_h^j, \quad (2.15)$$

and I_H^h is used to interpolate (or prolongate) the correction \hat{v}_H^j to Ω_h :

$$\hat{v}_H^j := I_H^h \hat{v}_H^j. \quad (2.16)$$

Altogether, one iteration step (calculating u_h^{j+1} from u_h^j) proceeds as follows:

- Compute the defect:	$d_h^j := f_h - L_h u_h^j.$
- Restrict the defect (fine-to-coarse transfer):	$d_H^j := I_h^H d_h^j.$
- Solve exactly on Ω_H :	$L_H \hat{v}_H^j = d_H^j.$
- Interpolate the correction (coarse-to-fine transfer):	$\hat{v}_h^j := I_H^h \hat{v}_H^j.$
- Compute new approximation:	$u_h^{j+1} := u_h^j + \hat{v}_h^j.$

This process is illustrated in Figure 2.1. The associated iteration operator is given by

$$I_h - B_h L_h \quad \text{with} \quad B_h = I_H^h L_H^{-1} I_h^H. \quad (2.17)$$

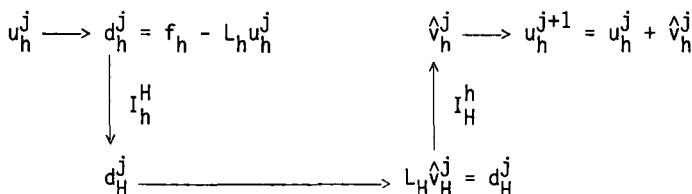


Figure 2.1: Structure of a coarse-grid correction process.

2.2 Relaxation and coarse-grid correction

If the methods in Example 2.1 and 2.2 are regarded as iterative methods for the solution of (2.1), they turn out to have very unsatisfactory convergence properties (or they are even not convergent at all). We are going to discuss this in more detail. In particular, we shall recognize that both types of methods behave very differently. This difference in behavior can be exploited: suitable combinations of both types of methods have very good convergence properties. Such combinations yield so-called *two-grid methods* which are the basis of *multigrid methods*.

Relaxation methods. Methods as considered in Example 2.1 are called *relaxation methods* in this paper. We also use this term for Jacobi- and Gauss-Seidel-type methods with under- or overrelaxation parameters, for Jacobi and Gauss-Seidel block-relaxation and related techniques.

The convergence properties of relaxation methods when applied to h-discrete elliptic equations (2.1) are known to become very bad if $h \rightarrow 0$. For Poisson's equa-

tion (model problem (P)), for instance, the spectral radii of the usual Jacobi and Gauss-Seidel methods behave like $1-0(h^2)$; SOR with optimal overrelaxation parameter has a spectral radius $1-0(h)$ [105].

The error reducing properties of these methods may be analyzed expanding the errors v_h^j into discrete Fourier series. In terms of this Fourier expansion one may roughly distinguish between *smooth (low-frequency)* and *non-smooth (high-frequency)* error components. It is already well-known from classical investigations on relaxation methods that especially the smooth error components are responsible for the slow asymptotic convergence (also see Section 3.2).

On the other hand, suitable relaxation methods are very efficient in smoothing the errors, i.e. in reducing the high-frequency error components (see Figure 2.2). The smoothing properties of relaxation methods are measured by a *smoothing factor* (see Sections 3.2 and 7.5), i.e. the worst (largest) factor, by which high-frequency error components are reduced per relaxation step. This factor refers to the high-frequency error components in a similar way as the spectral radius refers to all error components. We shall see that appropriate relaxation methods are characterized by smoothing factors which are smaller than 0.5, independent of h .

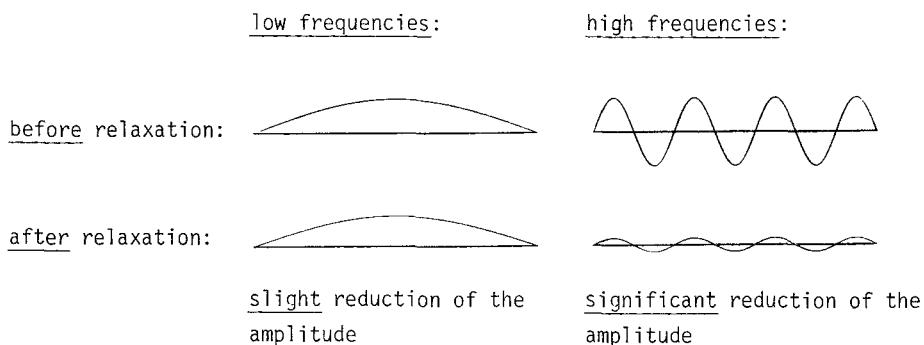


Figure 2.2: Typical error smoothing behavior of appropriate relaxation methods.

For convenience, we introduce the notation

$$\bar{w}_h = \text{RELAX}^\nu(w_h, L_h, f_h) \quad (2.18)$$

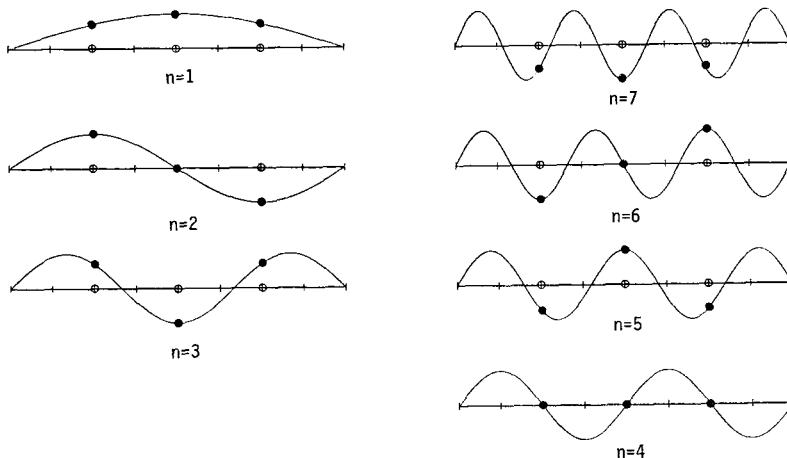
to denote the result \bar{w}_h of ν relaxation steps applied to (2.1) starting with w_h as first approximation. If $\nu=1$ we omit the upper index.

Coarse-grid correction. Processes as considered in Example 2.2 are called *coarse-grid correction* (CGC) processes. Our first observation is that such a process (with linear operators I_h^H, I_H^h), used as an iterative method by itself, is not convergent:

Lemma 2.1: $\rho(I_h - I_H^h L_H^{-1} I_h^H L_h) \geq 1$.

This follows directly from the fact that any $w_h \in \mathcal{G}(\Omega_h)$ ($w_h \neq 0$) such that $L_h w_h$ lies in the kernel of I_h^H , is reproduced by the corresponding iteration operator.

This shows that the coarse-grid defect equation (2.12) in general is not a reasonable approximation for the original defect equation (2.5). In particular, those components of v_h^j which cannot be represented on the coarse grid Ω_H (which are - so to say - "not visible" on the H-grid) can, of course, not be reduced by use of this grid. We illustrate this in Figure 2.3 for $H=2h$.



Components which are visible
on Ω_H (wavelength $> 4h$)

Components which are not visible
on Ω_H (wavelength $\leq 4h$)

Figure 2.3: $\sin(n\pi x)$, low ($n=1,2,3$) and high ($n=4,5,6,7$) frequency components
for $h=1/8$ and $H=1/4$

We conclude from the above considerations:

The error v_h^j which is determined by the original defect equation (2.5), can be approximated well by use of the H-grid only if this error is "smooth", i.e. if its

high-frequency components are small compared to its low-frequency components. This is, however, exactly what can be achieved by suitable relaxation methods very efficiently. (Of course, the terms "high" and "low" have to be related to the given h- and H-grid, see Chapter 7.)

2.3 Structure of an (h,H) two-grid iteration operator

Summarizing the above considerations, it is reasonable to combine the two processes of relaxation and of coarse-grid correction. In this way we obtain an iterative (h,H) two-grid method. Each iteration step of such a method consists of a *smoothing* and a *coarse-grid correction* part.

One step of such an (h,H)-method (calculating u_h^{j+1} from u_h^j) proceeds as follows:

(1) Smoothing part I:

- Compute \bar{u}_h^j by applying $v_1 (\geq 0)$ steps of a given relaxation method to u_h^j :

$$\bar{u}_h^j := \text{RELAX}^{v_1}(u_h^j, L_h, f_h).$$

(2) Coarse-grid correction part:

- Compute the defect: $\hat{d}_h^j := f_h - L_h \bar{u}_h^j$.
- Restrict the defect (fine-to-coarse transfer): $\hat{d}_H^j := I_h^H \hat{d}_h^j$.
- Solve exactly on Ω_H : $L_H \hat{v}_H^j = \hat{d}_H^j$. (2.19)
- Interpolate the correction (coarse-to-fine transfer): $\hat{v}_h^j := I_H^h \hat{v}_H^j$.
- Compute the corrected approximation: $\bar{u}_h^j + \hat{v}_h^j$.

(3) Smoothing part II:

- Compute u_h^{j+1} by applying $v_2 (\geq 0)$ steps of the given relaxation method to $\bar{u}_h^j + \hat{v}_h^j$:

$$u_h^{j+1} := \text{RELAX}^{v_2}(\bar{u}_h^j + \hat{v}_h^j, L_h, f_h).$$

This process is illustrated in Figure 2.4. The difference to Figure 2.1 lies in the additional relaxation steps before and after the coarse-grid correction.

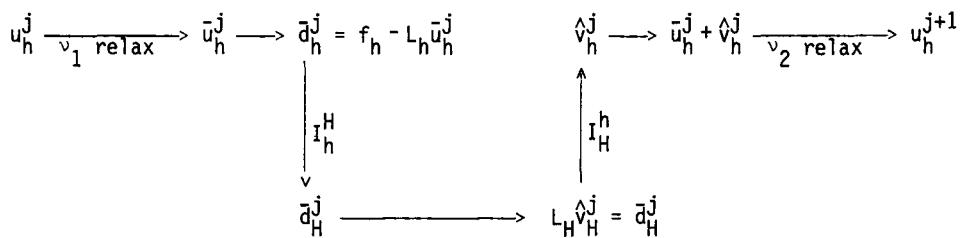


Figure 2.4: Structure of an (h,H) two-grid method

From the above description, one immediately obtains the iteration operator M_h^H of the (h,H) two-grid method:

$$\text{Lemma 2.2: } M_h^H = S_h^{v_2} K_h^H S_h^{v_1} \text{ with } K_h^H := I_h - I_H^h L_H^{-1} I_H^H L_h.$$

Here S_h denotes the iteration operator corresponding to the relaxation process used. Obviously, the following individual components of the (h,H) -method have to be specified:

- the relaxation procedure, characterized by S_h ;
- the numbers v_1, v_2 of relaxation steps;
- the coarse grid Ω_H ;
- the fine-to-coarse restriction operator I_h^H ;
- the coarse-grid operator L_H ;
- the coarse-to-fine interpolation operator I_H^h .

Experience with multigrid methods shows that the choice of these components has - on the one hand - a strong influence on the efficiency of the resulting algorithms. On the other hand, there seem to be no general rules on how to choose the individual components in order to construct optimal algorithms. One can, however, recommend certain choices for certain situations. Whenever possible, such recommendations should, of course, be theoretically founded. The main objective of the *model problem* and the *local mode analysis* is to determine the asymptotic convergence factor $\rho(M_h^H)$ or suitable norms $\|M_h^H\|$ and to investigate the influence of the above mentioned choices on $\rho(M_h^H)$, $\|M_h^H\|$. In Chapter 3, we will have a preliminary discussion of this question for the special case of Poisson's equation.

The (h,H) -method is not yet a real *multigrid* method as only one coarser grid is used so far. In practice, the exact solution of the defect equation (2.19) on Ω_H is replaced by an approximate solution, which is obtained by using still coarser

grids. A straightforward recursive definition of a corresponding multigrid iteration will be given in Chapter 4. In any case, two-grid methods are the basis for multi-grid processes.

2.4 Some specifications and extensions

Some of the assumptions in the previous sections were made in order to keep the description of the basic multigrid idea simple. Not all multigrid methods used in practice satisfy these assumptions. We want to mention some important modifications in this section. First, however, we want to specify some of the quantities used above and to introduce the corresponding notation.

2.4.1 Choice of the coarser grid

The most important and most frequently used choice of Ω_H is characterized by doubling the given meshsize h , i.e. $H=2h$. Most of the results and considerations in this paper refer to this choice which will be called *standard coarsening*.

If the meshsize h is doubled in one direction only, i.e. $H=(2h_{x_1}, h_{x_2})$ or $H=(h_{x_1}, 2h_{x_2})$, we speak of *semi-coarsening*. This is of interest for anisotropic and certain singularly perturbed differential operators [25],[23]. Furthermore, semi-coarsening is natural for the so-called MG-AR methods [82],[111].

We speak of *red-black coarsening*, if the coarse grid points are distributed in the fine grid in a checkerboard manner. We will consider this coarsening only for square grids ($h_{x_1}=h_{x_2}$). In this case, Ω_H can obviously be identified with a rotated grid of meshsize $\sqrt{2}h$. In particular, red-black coarsening is characteristic for the so-called MG-TR methods [82],[111].

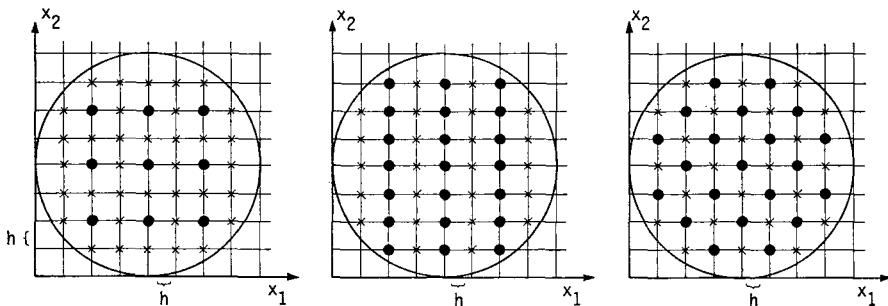


Figure 2.5: Example of standard, semi and red-black coarsening. The grid points of Ω_H are marked by dots. The grid points of Ω_h are just the union of points marked by crosses or dots.

2.4.2 Choice of the coarse-grid difference operator

In this paper, we mainly consider multigrid methods in which L_H is constructed with respect to Ω_H in a way analogous to the construction of L_h with respect to Ω_h . This is, however, not necessary: In principle, L_H may be any reasonable difference operator approximating L_h in some sense. In particular, we want to mention the important case of the *Galerkin approach* [109],[70],[42], which has a natural background in the finite-element area. Here the transfer operators I_h^H and I_H^h are used to define the coarse-grid operator L_H :

$$L_H := I_h^H L_h I_H^h. \quad (2.21)$$

2.4.3 More general smoothing procedures

Instead of applying the same relaxation operators S_h v_1 times before and v_2 times after the coarse-grid correction step, one may, of course, use different operators in the different smoothing steps. This straightforward extension takes, for example, the possibility into account that different relaxation parameters are chosen in different relaxation steps (see Section 3.6). M_h^H is then given by

$$M_h^H = S_{h,v_1} \cdot \dots \cdot S_{h,v_1+1} K_h^H S_{h,v_1} \cdot \dots \cdot S_{h,1}$$

where K_h^H is defined in Lemma 2.2.

Furthermore, not only relaxation methods may be used for error smoothing: any (iterative) procedure which has good smoothing properties and requires little computational work per iteration step, can, in principle, be used as a smoother in the multigrid context. In particular, certain pre-conditioning methods (various *incomplete LU-decomposition* methods) have been shown to yield good and robust smoothers [54],[60],[109],[110], also see Section 9.2.

2.4.4 Coarse-to-fine transfer using the grid equation

In certain efficient multigrid algorithms, the right hand side of the (original) grid equation (2.1) is used within the coarse-to-fine transfer (instead of performing a pure interpolation) [82],[111]. Note that such a coarse-to-fine tranfer can, in general, not be described by a linear operator but rather by an affine operator. Clearly, in such cases, also Lemma 2.1 is no longer true: The coarse-grid correction may now have a spectral radius < 1 (see [82]).

Coarse-to-fine transfers as mentioned above can often be interpreted as an interpolation (in the usual sense) followed by one (or one half) step of a suitable relax-

ation process [82]. They are of particular interest for the full multigrid method (FMG, see Chapter 6; [36]) also.

2.4.5 More general treatment of boundary conditions

In Section 2.1, we have assumed for simplicity that the (discrete) boundary conditions have been "eliminated". Because of this assumption we had to consider only one grid equation

$$L_h u_h = f_h \quad (\Omega_h) \quad (2.22)$$

with $u_h, f_h \in \mathbb{G}(\Omega_h)$, $L_h : \mathbb{G}(\Omega_h) \rightarrow \mathbb{G}(\Omega_h)$. In this introductory paper, we will always make this assumption. For example, for the multigrid treatment of second order equations with Dirichlet boundary conditions in general bounded domains (see Chapter 10) or with Neumann boundary conditions in rectangular domains [37],[36], the elimination of boundary conditions is a well established technique. In more general cases, however, a different treatment of boundary conditions may be necessary.

For its formal description, (2.22) has to be replaced by a system of grid equations (1.2) where Ω_h denotes an *interior grid* and Γ_h a *boundary grid*. In principle, all considerations and explanations of the previous sections can be interpreted with respect to this more general situation. All multigrid components $(S_h, I_h^H, I_H^h, \dots)$ have of course, to be defined separately for Ω_h and for Γ_h . For a more detailed description of these techniques, we refer to Brandt [25], Section 5. In particular, the question of suitable *boundary relaxation methods* is discussed there.

3. Analysis of a sample ($h,2h$) two-grid method for Poisson's equation

In this chapter, we consider a very simple two-grid method for the model problem (P). This is a concrete example for the general description of two-grid methods as given in Section 2.3. By determining the *spectral radius* $\rho(M_h^{2h})$ of the associated iteration operator M_h^{2h} , we prove the h -independency of its convergence factor. This result is valid even if only one relaxation step ($v=v_1=1$) is performed per two-grid iteration. The considerations of this section are representative for what we call *model problem analysis* (see Chapters 7 and 8).

In Section 3.1 we define the algorithm. Section 3.2 refers to the relaxation operator S_h which is used for smoothing; on the basis of Fourier analysis, we give a first definition of a smoothing factor there. The Fourier representation of the coarse-grid correction operator K_h^{2h} is given in Section 3.3. Using the representations of S_h and K_h^{2h} , we determine the spectral radius $\rho(M_h^{2h})$ (Section 3.4). In Section 3.5 we are concerned with norms for M_h^{2h} , namely with its *spectral* and its *energy norm*. In Section 3.6 we outline algorithmic variants and their influence on the theoretical results.

Within the two-grid method which is considered here, we use a *Jacobi (under-) relaxation method* for smoothing. This smoothing method is by far not the most efficient one; it is, however, theoretically transparent and allows a simple rigorous and quantitative analysis. A disadvantage of Jacobi's method is the need of a relaxation parameter for good smoothing; this is not typical for smoothing techniques in general. (Other relaxation methods without this disadvantage will be discussed, e.g., in the Chapters 7 and 8.)

3.1 An ($h,2h$)-algorithm

For the solution of the discrete Poisson equation on the unit square, model problem (P) (see (1.10)), we consider an algorithm as described in Section 2.3, the j -th iteration of which (computing u_h^{j+1} from u_h^j) consists of the following components:

Smoothing part I: Apply v_1 iteration steps of Jacobi ω -relaxation to u_h^j with fixed relaxation parameter ω (the choice of ω will be discussed in the following section):

$$\bar{u}_h^j := \text{RELAX}^{v_1}(u_h^j, L_h, f_h; \omega). \quad (3.1)$$

Coarse-grid correction on

$$\Omega_{2h} = (0,1)^2 \cap G_{2h} = \{x = (2ih, 2jh) : i, j = 1, 2, \dots, N/2-1\} \quad (3.2)$$

proceeding as follows:

- Compute the defect $\tilde{d}_h^j := f_h - L_h \bar{u}_h^j$.
- Restrict the defect \tilde{d}_h^j (fine-to-coarse transfer) using the *full weighting* (FW) operator, i.e.

$$I_h^{2h} \triangleq \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} h^2 . \quad (3.3)$$

- Compute the exact solution \hat{v}_{2h}^j of the Ω_{2h} -defect equation $L_{2h} \hat{v}_{2h}^j = \tilde{d}_{2h}^j$. Here L_{2h} is defined analogously to L_h .
- Interpolate the correction \hat{v}_h^j (coarse-to-fine transfer) using *bilinear interpolation*, i.e.

$$I_{2h}^h \triangleq \frac{1}{4} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} h_{2h}^h . \quad (3.4)$$

- Compute the corrected approximation on Ω_h : $\bar{u}_h^j + \hat{v}_h^j$.

Smoothing part II: Compute u_h^{j+1} by applying v_2 smoothing steps of Jacobi ω -relaxation to $\bar{u}_h^j + \hat{v}_h^j$, i.e.

$$u_h^{j+1} = \text{RELAX}^{v_2}(\bar{u}_h^j + \hat{v}_h^j, L_h, f_h; \omega). \quad (3.5)$$

By Lemma 2.2, the iteration operator of this $(h, 2h)$ two-grid method is given by

$$M_h^{2h} = M_h^{2h}(v_1, v_2; \omega) = S_h^{v_2}(\omega) K_h^{2h} S_h^{v_1}(\omega) \quad \text{with} \quad K_h^{2h} = I_h^h - I_{2h}^h L_{2h}^{-1} I_h^{2h} L_h \quad (3.6)$$

where $S_h = S_h(\omega)$ denotes the iteration operator which corresponds to Jacobi ω -relaxation.

3.2 The relaxation operator

One step of Jacobi ω -relaxation applied to problem (P) with first approximation w_h , i.e.

$$\bar{w}_h = \text{RELAX}(w_h, L_h, f_h; \omega) \quad (3.7)$$

is defined by

$$\bar{w}_h = w_h + \omega(z_h - w_h), \quad \frac{4}{h^2} z_h(x) + L_h^- w_h(x) = f_h(x) \quad (x \in \Omega_h). \quad (3.8)$$

Here L_h^- denotes the "off-diagonal" part of L_h , namely

$$L_h^- w_h(x) := L_h w_h(x) - \frac{4}{h^2} w_h(x) \quad (x \in \Omega_h). \quad (3.9)$$

Clearly, the corresponding iteration operator is given by

$$S_h = S_h(\omega) = I_h - \frac{\omega h^2}{4} L_h. \quad (3.10)$$

Let us first recall some well-known facts about the convergence properties of Jacobi ω -relaxation. These facts can be derived easily by considering the eigenfunctions of S_h , which are the same as those of L_h , namely

$$\varphi_n(x) = 2 \sin(n_1 \pi x_1) \sin(n_2 \pi x_2) \quad (x \in \Omega_h; |n| \leq N-1) \quad (3.11)$$

where $n = (n_1, n_2) \in \mathbb{N}^2$ and $|n| = \max(n_1, n_2)$. The corresponding eigenvalues of S_h are

$$\chi_n = \chi_n(\omega) = 1 - \frac{\omega}{2} (2 - \cos(n_1 \pi h) - \cos(n_2 \pi h)). \quad (3.12)$$

For the spectral radius $\rho(S_h) = \max \{|\chi_n| : |n| \leq N-1\}$ we obtain

$$\text{for } 0 < \omega \leq 1: \quad \rho(S_h) = |\chi_{1,1}| = 1 - \omega(1 - \cos \pi h) = 1 - O(h^2); \quad (3.13)$$

$$\text{for } \omega \leq 0 \text{ or } \omega > 1: \quad \rho(S_h) \geq 1 \quad (\text{if } h \text{ small enough}).$$

In particular, with respect to the (very unsatisfactory) asymptotic convergence, there is no use in introducing the relaxation parameter: $\omega=1$ is the best choice.

The situation is quite different with respect to the smoothing properties of Jacobi ω -relaxation. For $0 < \omega \leq 1$, we first observe by (3.13) that it is the smoothest eigenfunction $\varphi_{1,1}$ which is responsible for the slow convergence of Jacobi's method. Highly oscillating eigenfunctions are reduced much faster if ω is chosen properly. To see this, we expand the errors before and after one relaxation step, namely

$$v_h := u_h - w_h \quad \text{and} \quad \bar{v}_h := u_h - \bar{w}_h,$$

into discrete eigenfunction series:

$$v_h = \sum_{|n| \leq N-1} \alpha_n \varphi_n, \quad \bar{v}_h = \sum_{|n| \leq N-1} \chi_n \varphi_n. \quad (3.14)$$

The smoothing properties of $S_h(\omega)$ are measured by distinguishing low and high frequencies (with respect to the coarser grid Ω_{2h} used). As motivated in Section 2.2,

it is reasonable to define as

$$\begin{aligned} \text{low frequencies: } \varphi_n & \text{ with } |n| < N/2, \\ \text{high frequencies: } \varphi_n & \text{ with } N/2 \leq |n| \leq N-1. \end{aligned} \quad (3.15)$$

In other words: The low frequencies are those eigenfunctions of L_h , which are representable also on the coarser grid Ω_{2h} . The high frequencies are "not visible" on Ω_{2h} at all. (Cf. Figure 2.3, where this distinction was illustrated for the corresponding 1D-case.)

We now define the *smoothing factor* $\mu(h;\omega)$ of S_h (and its supremum $\mu^*(\omega)$ over h) as the worst factor by which high frequency error components are reduced per relaxation step, i.e.

$$\begin{aligned} \mu(h;\omega) &:= \max \{|x_n| : N/2 \leq |n| \leq N-1\}, \\ \mu^*(\omega) &:= \sup \{\mu(h;\omega) : h \leq 1/4\}. \end{aligned} \quad (3.16)$$

Remark: This or similar definitions of the smoothing factor can also be used for some other simple smoothing methods. It has, however, to be substantially refined for smoothing methods like RB and ZEBRA relaxation which are much more efficient in smoothing than Jacobi's method. We give a refined definition in Section 7.5.

Inserting (3.12), we get from (3.16)

$$\begin{aligned} \mu(h;\omega) &= \max \{|1-\omega(2-\cos\pi h)/2|, |1-\omega(1+\cos\pi h)|\}, \\ \mu^*(\omega) &= \max \{|1-\omega/2|, |1-2\omega|\}. \end{aligned} \quad (3.17)$$

This shows that Jacobi's relaxation has no smoothing properties for $\omega \leq 0$ or $\omega > 1$:

$$\mu(h;\omega) \geq 1 \quad \text{if } \omega \leq 0 \text{ or } \omega > 1 \quad (\text{and } h \text{ is sufficiently small}).$$

For $0 < \omega < 1$, however, the smoothing factor is smaller than 1 and bounded away from 1, independently of h . For $\omega=1$, we have a smoothing factor of $1-O(h^2)$ only. In particular, we find by (3.17):

$$\mu(h;\omega) = \begin{cases} \cos\pi h & \text{if } \omega = 1 \\ (2+\cos\pi h)/4 & \text{if } \omega = 1/2 \\ (1+2\cos\pi h)/5 & \text{if } \omega = 4/5 \end{cases} \quad \mu^*(\omega) = \begin{cases} 1 & \text{if } \omega = 1 \\ 3/4 & \text{if } \omega = 1/2 \\ 3/5 & \text{if } \omega = 4/5 \end{cases}.$$

The choice $\omega=4/5$ is optimal in the following sense:

$$\inf \{\mu^*(\omega) : 0 \leq \omega \leq 1\} = \mu^*(4/5) = 3/5. \quad (3.18)$$

With respect to $\mu(h, \omega)$, one obtains

$$\inf \{\mu(h; \omega) : 0 \leq \omega \leq 1\} = \mu(h; \frac{4}{4+\cos\pi h}) = \frac{3\cos\pi h}{4+\cos\pi h} = \frac{3}{5} - O(h^2).$$

3.3 The coarse-grid correction operator

For the coarse-grid correction operator

$$K_h^{2h} = I_h - I_{2h}^h L_{2h}^{-1} I_h^{2h} L_h,$$

it turns out that the (at most) 4-dimensional subspaces of $\mathbb{G}(\Omega_h)$

$$E_{h,n} := \text{span } \{\varphi_{n_1, n_2}; \varphi_{N-n_1, N-n_2}; -\varphi_{N-n_1, n_2}; -\varphi_{n_1, N-n_2}\} \quad (|n| \leq N/2) \quad (3.19)$$

are invariant under K_h^{2h} , i.e.

$$K_h^{2h} : E_{h,n} \rightarrow E_{h,n} \quad (|n| \leq N/2). \quad (3.20)$$

Consequently, as the φ_n ($|n| \leq N-1$) form an orthonormal basis of $\mathbb{G}(\Omega_h)$ (with respect to (1.7)), K_h^{2h} is orthogonally equivalent to a block-diagonal matrix consisting of (at most) (4,4)-blocks $\hat{K}_{h,n}^{2h}$. This is a characteristic feature of what we call *model problem analysis*.

For a detailed description of the matrices $\hat{K}_{h,n}^{2h}$, we also need a basis of eigenfunctions of L_{2h} . A suitable basis is given by

$$\Phi_n(x) := 2 \sin(n_1 \pi x_1) \sin(n_2 \pi x_2) \quad (x \in \Omega_{2h}; |n| \leq N/2-1). \quad (3.21)$$

On Ω_{2h} , the φ_n and the basis functions of $E_{h,n}$ coincide:

$$\begin{aligned} \varphi_{n_1, n_2}(x) &= \varphi_{N-n_1, N-n_2}(x) = -\varphi_{N-n_1, n_2}(x) = -\varphi_{n_1, N-n_2}(x) = \Phi_{n_1, n_2}(x) \\ &\quad (x \in \Omega_{2h}; |n| \leq N/2-1). \end{aligned} \quad (3.22)$$

For $n_1 = N/2$ and/or $n_2 = N/2$, the spaces $E_{h,n}$ are 1-/2-dimensional, respectively, and their basis functions coincide on Ω_{2h} with the zero grid function.

The transfer operators I_{2h}^h and I_h^{2h} have the characteristic properties

$$\left. \begin{aligned} I_{2h}^h : \text{span } \{\Phi_n\} &\rightarrow E_{h,n} \quad (|n| \leq N/2-1), \\ I_h^{2h} : E_{h,n} &\rightarrow \text{span } \{\Phi_n\} \quad (|n| \leq N/2-1), \\ I_h^{2h} \varphi_n &= 0 \quad (n_1 = N/2 \text{ and/or } n_2 = N/2). \end{aligned} \right\} \quad (3.23)$$

In more detail, we have for fixed n ($|n| \leq N/2-1$)

$$I_{2h}^h \Phi_{n_1, n_2} = (1-\xi)(1-\eta)\varphi_{n_1, n_2} + \xi\eta\varphi_{N-n_1, N-n_2} - \xi(1-\eta)\varphi_{N-n_1, n_2} - (1-\xi)\eta\varphi_{n_1, N-n_2} \quad (3.24)$$

and

$$I_h^{2h} \begin{Bmatrix} \varphi_{n_1, n_2} \\ \varphi_{N-n_2, N-n_2} \\ -\varphi_{N-n_1, n_2} \\ -\varphi_{n_1, N-n_2} \end{Bmatrix} = \begin{Bmatrix} (1-\xi)(1-\eta) \\ \xi\eta \\ \xi(1-\eta) \\ (1-\xi)\eta \end{Bmatrix} \Phi_{n_1, n_2}. \quad (3.25)$$

Here we use the abbreviations

$$\xi = \sin^2(n_1\pi/2N), \quad \eta = \sin^2(n_2\pi/2N). \quad (3.26)$$

Together with the fact that the φ_n and the Φ_n are eigenfunctions of L_h and L_{2h} , respectively, (3.20) follows immediately. In particular, we obtain

$$\hat{k}_{h,n}^{2h} = \begin{cases} I - \frac{1}{\Lambda} \begin{bmatrix} b_i c_j \\ \end{bmatrix}_{4,4} & (\text{if } |n| < N/2) \\ (2,2)\text{-identity matrix} & (\text{if } n_1 \text{ or } n_2 = N/2) \\ (1,1)\text{-identity matrix} & (\text{if } n_1 = n_2 = N/2) \end{cases} \quad (3.27)$$

with $\Lambda = \xi(1-\xi) + \eta(1-\eta)$ and

$$\begin{aligned} b_1 &= (1-\xi)(1-\eta), & b_2 &= \xi\eta, & b_3 &= \xi(1-\eta), & b_4 &= (1-\xi)\eta, \\ c_1 &= b_1(\xi+\eta), & c_2 &= b_2(2-\xi-\eta), & c_3 &= b_3(1-\xi+\eta), & c_4 &= b_4(1+\xi-\eta). \end{aligned}$$

3.4 Spectral radius of the two-grid operator

The invariance of the spaces $E_{h,n}$ ($|n| \leq N/2$) under κ_h^{2h} and under S_h also imply their invariance under M_h^{2h} . Using (3.12), (3.27) and the abbreviations (3.26) one immediately obtains the $E_{h,n}$ -representation $\hat{M}_{h,n}^{2h}$ of M_h^{2h} :

$$\hat{M}_{h,n}^{2h} = \hat{M}_{h,n}^{2h}(v_1, v_2; \omega) = \hat{S}_{h,n}^{v_2}(\omega) \hat{k}_{h,n}^{2h} \hat{S}_{h,n}^{v_1}(\omega) \quad (3.28)$$

where

$$\hat{S}_{h,n}(\omega) = \begin{cases} \begin{bmatrix} 1-\omega(\xi+\eta) & & \\ & 1-\omega(2-\xi-\eta) & \\ & & 1-\omega(1-\xi+\eta) \end{bmatrix} & (\text{if } |n| < N/2) \\ \begin{bmatrix} & & 1-\omega(1+\xi-\eta) \\ \begin{bmatrix} 1-\omega(\xi+\eta) & & \\ & 1-\omega(2-\xi-\eta) \end{bmatrix} & & \\ & & 1-\omega(\xi+\eta) \end{bmatrix}_{2,2} & (\text{if } n_1 \text{ or } n_2 = N/2) \\ \begin{bmatrix} & & \\ & & 1-\omega(\xi+\eta) \end{bmatrix}_{1,1} & (\text{if } n_1 = n_2 = N/2) \end{cases} \quad (3.29)$$

Thus the determination of the spectral radius $\rho(M_h^{2h})$ has been reduced to the calculation of the spectral radii of (at most) (4,4)-matrices:

$$\rho(M_h^{2h}) = \max \{\rho(M_{h,n}^{2h}) : |n| \leq N/2\}. \quad (3.30)$$

This quantity depends in particular on the parameter ω and on $v := v_1 + v_2$. (Since $\rho(AB) = \rho(BA)$ for any linear operators A and B, $\rho(M_h^{2h})$ does not depend on v_1 and v_2 individually.) In the following, we shall use the notation

$$\rho(h,v;\omega) := \rho(M_h^{2h}(v_1, v_2; \omega)). \quad (3.31)$$

Usually, one is more interested in

$$\rho^*(v;\omega) := \sup \{\rho(h,v;\omega) : h \leq 1/4\} \quad (3.32)$$

than in $\rho(h,v;\omega)$ for fixed h . From the representation (3.28) (with (3.27) and (3.29)), one recognizes that $\rho(M_{h,n}^{2h})$ can be written as a certain function $f(\xi, \eta)$. Using this function, ρ^* is conveniently computed as

$$\rho^*(v,\omega) = \sup \{f(\xi, \eta) : 0 < \xi, \eta \leq 1/2\}. \quad (3.33)$$

(For simplicity, we use the term "convergence factor" for both ρ and ρ^* . It will be clear from the context, which quantity is actually considered.)

In Table 3.1, we have listed ρ^* as a function of v for the two relaxation parameters $\omega=0.5$ and $\omega=0.8$. First, we recognize that - as already suggested by the respective smoothing factors, see Section 3.2 - the parameter $\omega=0.8$ indeed yields better convergence factors ρ^* than $\omega=0.5$. One sees that the convergence factors ρ^* decrease for increasing v . This does not mean, however, that large values of v are suitable with respect to efficiency, as also the computational work increases with v . We postpone the question of efficiency: In Chapter 4 we will discuss the computational work in connection with complete multigrid iterations; results concerning the efficiency of several methods will be given, e.g., in Chapter 8.

That too large values of v are useless, can already be seen from a comparison of ρ^* and $(\mu^*)^v$. For small values of v we observe a remarkable accordance between these quantities. If v increases, however, $(\mu^*)^v$ is no longer a good prediction for ρ^* : the high smoothing effect is not fully exploited as the reduction of low error frequencies by one coarse-grid correction step is not good enough, or the smoothing effect is even partly destroyed by the coarse-grid correction (which introduces new high frequencies by itself). Typically, one has $\rho^*(v;\omega) \sim \text{const}/v$ ($v \rightarrow \infty$) (see, for instance, Theorem 8.1). The difference between $(\mu^*)^v$ and ρ^* occurs all the sooner, the better the smoothing properties of S_h are ($v \geq 4$ for $\omega = 0.8$, $v \geq 8$ for $\omega = 0.5$).

v	$\omega = 0.5$		$\omega = 0.8$	
	$(\mu^*(\omega))^v$	$\rho^*(v;\omega)$	$(\mu^*(\omega))^v$	$\rho^*(v;\omega)$
1	0.750	0.750	0.600	0.600
2	0.563	0.563	0.360	0.360
3	0.422	0.422	0.216	0.216
4	0.316	0.316	0.130	0.137
5	0.237	0.237	0.078	0.113
6	0.178	0.178	0.047	0.097
7	0.133	0.133	0.028	0.085
8	0.100	0.118	0.017	0.076
9	0.075	0.106	0.010	0.068
10	0.056	0.097	0.006	0.062

Table 3.1: Comparison of smoothing factors μ^* and two-grid convergence factors ρ^* for different v and ω .

Finally, we give some values for $\rho(h,v;\omega)$ as a function of h . The corresponding results in Table 3.2 show that ρ tends to ρ^* rather quickly. Thus, in the cases considered, the main information about the two-grid convergence is contained in ρ^* .

h	$\omega = 0.5$				$\omega = 0.8$			
	$v=1$	$v=2$	$v=3$	$v=4$	$v=1$	$v=2$	$v=3$	$v=4$
1/4	0.677	0.458	0.310	0.217	0.483	0.233	0.171	0.130
1/8	0.731	0.534	0.391	0.285	0.570	0.324	0.185	0.130
1/16	0.745	0.555	0.414	0.308	0.592	0.351	0.208	0.135
1/32	0.749	0.561	0.420	0.314	0.598	0.358	0.214	0.137
1/64	0.750	0.562	0.421	0.316	0.600	0.359	0.215	0.137
1/128	0.750	0.562	0.422	0.316	0.600	0.360	0.216	0.137
$\rho^*(v;\omega)$	0.750	0.563	0.422	0.316	0.600	0.360	0.216	0.137

Table 3.2: The two-grid convergence factor $\rho(h,v;\omega)$ as a function of h

3.5 Norms of the two-grid operator

Whereas the spectral radius $\rho(M_h^{2h})$ gives insight into the asymptotic convergence behavior of a two-grid method, norms are needed to measure the actual error (or defect) reducing per iteration step. In particular, essential use of norms of M_h^{2h} is made in the theoretical investigations of complete multigrid iterations (Chapter 4) and of the full multigrid method (Chapter 6).

There are many reasonable possibilities to choose norms. Of course, different choices of norms will in general lead to very different results. A general observation is that the spectral radius ρ is less sensitive with respect to algorithmical details than norms usually are. For example, norms considerably depend on v_1 and on v_2 , whereas ρ depends only on the sum $v=v_1+v_2$.

In this paper, we mainly consider the operatornorm $\| \cdot \|_S$ corresponding to the Euclidian inner product (1.7) on $\mathbb{G}(\Omega_h)$, i.e. the *spectralnorm*

$$\| M \|_S = \sqrt{\rho(MM^*)}, \quad (3.34)$$

where M denotes any linear operator $M : \mathbb{G}(\Omega_h) \rightarrow \mathbb{G}(\Omega_h)$. Apart from the error reduction ($M=M_h^{2h}$), we sometimes also consider the defect reduction ($M=L_h M_h^{2h} L_h^{-1}$).

For positive-definite symmetric operators L_h , the *energy norm* (which is induced by the inner product (1.8)) is also of - mainly theoretical - interest. The corresponding operatornorm is given by

$$\| M \|_E = \| L_h^{1/2} M L_h^{-1/2} \|_S = \sqrt{\rho(L_h M L_h^{-1} M^*)}. \quad (3.35)$$

(Here M^* denotes the operator adjoint to M with respect to the Euclidian inner product.) We introduce the following notations:

$$\sigma_S := \| M_h^{2h} \|_S, \quad \sigma_E := \| M_h^{2h} \|_E, \quad \sigma_d := \| L_h M_h^{2h} L_h^{-1} \|_S. \quad (3.36)$$

In particular, these quantities depend on h, v_1, v_2 and ω . By σ_S^*, σ_E^* and σ_d^* we denote the suprema of σ_S, σ_E and σ_d with respect to h , e.g.

$$\sigma_S^*(v_1, v_2; \omega) := \sup \{ \sigma_S(h, v_1, v_2; \omega) : h \leq 1/4 \}. \quad (3.37)$$

All the above norms can be determined from the representation (3.28) in much the same way as ρ and ρ^* . In particular, one obtains

$$\begin{aligned}\sigma_S &= \max \{ \| \hat{M}_{h,n}^{2h} \|_S : |n| \leq N/2 \}, \\ \sigma_E &= \max \{ \| \hat{L}_{h,n}^{1/2} \hat{M}_{h,n}^{2h} \hat{L}_{h,n}^{-1/2} \|_S : |n| \leq N/2 \}, \\ \sigma_d &= \max \{ \| \hat{L}_{h,n} \hat{M}_{h,n}^{2h} \hat{L}_{h,n}^{-1} \|_S : |n| \leq N/2 \},\end{aligned}\quad (3.38)$$

where $\hat{L}_{h,n}$ denotes the (diagonal) matrix representation of L_h with respect to $E_{h,n}$. The computation of σ_S^* , σ_E^* and σ_d^* can be performed analogously as for ρ^* (cf. (3.33)).

For our sample $(h,2h)$ -method, we have listed several values σ_S^* , σ_E^* and σ_d^* in Tables 3.3a and 3.3b. For comparison, we also recall the corresponding ρ^* -values already given in Table 3.1. It is, of course, a general aim to have not only a small spectral radius of M_h^{2h} but also small norms. In both tables, 3.3a and 3.3b, we have underlined those norm-values which are optimal (=spectral radius). According to these results, it seems to be reasonable to choose v_1 and v_2 not very different from each other (and rather $v_1 \geq v_2$ than $v_1 \leq v_2$).

(v_1, v_2)	$\rho^*(v; \omega)$	$\sigma_S^*(v_1, v_2; \omega)$	$\sigma_E^*(v_1, v_2; \omega)$	$\sigma_d^*(v_1, v_2; \omega)$
(1,0)	0.750	<u>0.750</u>	<u>0.750</u>	1.118
(0,1)		1.118	<u>0.750</u>	<u>0.750</u>
(2,0)	0.563	<u>0.563</u>	<u>0.563</u>	1.031
(1,1)		<u>0.563</u>	<u>0.563</u>	<u>0.563</u>
(0,2)		1.031	<u>0.563</u>	<u>0.563</u>
(3,0)	0.422	<u>0.422</u>	<u>0.422</u>	1.008
(2,1)		<u>0.422</u>	<u>0.422</u>	0.515
(1,2)		0.515	<u>0.422</u>	<u>0.422</u>
(0,3)		1.008	<u>0.422</u>	<u>0.422</u>
(4,0)	0.316	<u>0.316</u>	0.323	1.002
(3,1)		<u>0.316</u>	<u>0.316</u>	0.504
(2,2)		<u>0.316</u>	<u>0.316</u>	<u>0.316</u>
(1,3)		0.504	<u>0.316</u>	<u>0.316</u>
(0,4)		1.002	0.323	<u>0.316</u>

Table 3.3a: Spectral radii and norms for $\omega = 0.5$

(v_1, v_2)	$\rho^*(v; \omega)$	$\sigma_S^*(v_1, v_2; \omega)$	$\sigma_E^*(v_1, v_2; \omega)$	$\sigma_d^*(v_1, v_2; \omega)$
(1,0)		<u>0.600</u>	<u>0.600</u>	1.020
(0,1)	0.600	1.020	<u>0.600</u>	<u>0.600</u>
(2,0)		<u>0.360</u>	<u>0.360</u>	1.000
(1,1)	0.360	<u>0.360</u>	<u>0.360</u>	<u>0.360</u>
(0,2)		1.000	<u>0.360</u>	<u>0.360</u>
(3,0)		<u>0.216</u>	0.269	1.000
(2,1)		<u>0.216</u>	<u>0.216</u>	0.239
(1,2)	0.216	0.239	<u>0.216</u>	<u>0.216</u>
(0,3)		1.000	0.269	<u>0.216</u>
(4,0)		0.148	0.233	1.000
(3,1)		<u>0.137</u>	0.140	0.209
(2,2)	0.137	<u>0.137</u>	<u>0.137</u>	<u>0.137</u>
(1,3)		0.209	0.140	<u>0.137</u>
(0,4)		1.000	0.233	0.148

Table 3.3b: Same as Table 3.3a for $\omega = 0.8$

The following equalities hold between the quantities considered:

$$\begin{aligned} \sigma_S^*(v_1, v_2; \omega) &= \sigma_d^*(v_2, v_1; \omega), \quad \sigma_E^*(v_1, v_2; \omega) = \sigma_E^*(v_2, v_1; \omega), \\ \sigma_E^*(v_1, v_2; \omega) &= \rho^*(v; \omega) \quad (\text{if } v_1 = v_2). \end{aligned} \quad (3.39)$$

They are an immediate consequence of the relations

$$\begin{aligned} (M_h^{2h}(v_1, v_2; \omega))^* &= L_h M_h^{2h}(v_2, v_1; \omega) L_h^{-1}, \\ (L_h^{1/2} M_h^{2h}(v_1, v_2; \omega) L_h^{-1/2})^* &= L_h^{1/2} M_h^{2h}(v_2, v_1; \omega) L_h^{-1/2} \end{aligned} \quad (3.40)$$

which hold in our particular example and can easily be verified.

3.6 Algorithmic variants

We want to mention two modifications of the $(h, 2h)$ -method considered and to show their influence on the quantities introduced above.

3.6.1 Use of straight injection for the fine-to-coarse transfer

In practice the FW operator (3.3) may often be replaced by simpler restriction operators. The simplest (and cheapest) fine-to-coarse transfer is given by the oper-

ator of straight injection (INJ)

$$I_h^{2h} = [1]_h^{2h}, \text{ i.e. } (I_h^{2h} w_h)(x) = w_h(x) \quad (x \in \Omega_{2h}). \quad (3.41)$$

Heuristically, it is clear that this operator should give similar results as the FW operator (3.3) as long as the defects (to which I_h^{2h} is applied) are really smooth.

If the INJ operator is used in our sample method, the theoretical considerations have to be modified only slightly: Instead of (3.25), we now have

$$I_h^{2h} \begin{Bmatrix} \varphi_{n_1, n_2} \\ \varphi_{N-n_1, N-n_2} \\ -\varphi_{N-n_1, n_2} \\ -\varphi_{n_1, N-n_2} \end{Bmatrix} = \Phi_{n_1, n_2}. \quad (3.42)$$

With this modification, one can calculate ρ^* , σ_S^* , σ_E^* , σ_d^* as the previous sections. Calculating ρ^* , it turns out that the asymptotic convergence properties are not influenced significantly by this exchange of the fine-to-coarse transfer operator: One obtains the same ρ^* -values as shown in Table 3.1 if v is not too large ($v \leq 7$ for $\omega = 0.5$ and $v \leq 3$ for $\omega = 0.8$). For larger values of v , the asymptotic convergence factor is even slightly better if INJ is used instead of FW.

The behavior of the norms σ_S^* and σ_E^* , however, is quite different now: We find for all v and ω :

$$\sigma_S^*(v; \omega) = \sigma_E^*(v; \omega) = \infty. \quad (3.43)$$

The reason for this can easily be seen by applying M_h^{2h} to one of the highest frequencies, e.g. to $\varphi_{N-1,1}$: This frequency is mapped into a grid function which contains the low frequency component $O(1/h^2) \varphi_{1,1}$.

The above behavior of σ_S^* and σ_E^* is characteristic for the use of straight injection within multigrid processes. For many theoretical approaches, where the above norms are needed, the INJ operator is therefore useless. On the other hand, in practice, straight injection gives often similar (or even better) results as full weighting. One should be aware, however, that errors which contain significant highest frequency components (see above), may be enlarged considerably if only one multigrid iteration step is performed. (This is the usual application in the FMG method, see Chapter 6!)

3.6.2 Jacobi ω -relaxation with several parameters

As we have seen above, the smoothing properties of Jacobi ω -relaxation significantly depend on the choice of ω . If $v \geq 2$ relaxation steps are carried out (per two-grid iteration), one can try to use different parameters $\omega_1, \dots, \omega_v$ in each step in order to improve the total smoothing effect. A straightforward extension of the definition of the smoothing factor (3.16) to this more general case is given by

$$\mu(h, v; \omega_1, \dots, \omega_v) := \sqrt{v} \max \{ |x_n(\omega_1) \cdots x_n(\omega_v)| : N/2 \leq |n| \leq N-1 \}. \quad (3.44)$$

Instead of (3.17), one now gets:

$$\mu^*(v; \omega_1, \dots, \omega_v) = \sqrt{v} \max \{ |(1-\omega_1 t) \cdots (1-\omega_v t)| : 1/2 \leq t \leq 2 \}.$$

Minimizing μ^* with respect to $\omega_1, \dots, \omega_v$ (for fixed v) gives the optimal parameters

$$\omega_j = \left(\frac{5}{4} + \frac{3}{4} \cos \left(\frac{2j-1}{2v} \pi \right) \right)^{-1} \quad (j = 1, \dots, v) \quad (3.45)$$

(zeros of Chebyshev polynomials).

These parameters are used in Table 3.4 where some values for μ^* and ρ^* are given. As one can see, the use of different relaxation parameters gives some improvement in the case considered (cf. Table 3.1). One should, however, take the following into account: Firstly, the explicit determination of optimal parameters is restricted to rather special situations. Secondly, as we have already pointed out previously, there are more efficient smoothing methods (for Poisson-like equations) than Jacobi ω -relaxation, which do not even need a parameter (for example, RB relaxation, see Section 8.2).

v	$(\mu^*(v; \omega_1, \dots, \omega_v))^v$	$\rho^*(v; \omega_1, \dots, \omega_v)$
1	0.600	0.600
2	0.220	0.220
3	0.074	0.126
4	0.025	0.110

Table 3.4: Jacobi ω -relaxation with optimal parameters (3.45)

4. Complete multigrid cycle

Up to now, we have described the multigrid principle only in its two-grid version. We have, however, already pointed out that two-grid methods - usually - are not used in practice: they serve only as the (theoretical) basis for the real multigrid method.

The multigrid idea starts from the observation that in a convergent two-grid method it is not necessary to solve the coarse-grid defect equation (2.19)

$$L_H \hat{v}_H^j = \bar{d}_H^j \quad (4.1)$$

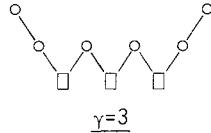
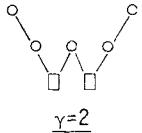
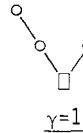
exactly. Instead, without essential loss of convergence speed, one may replace \hat{v}_H^j by a suitable approximation. A natural way to obtain such an approximation is to apply an analogous two-grid method to (4.1) also, where an even coarser grid than Ω_H is used. Clearly, if the convergence factor of this two-grid method is small enough, it is sufficient to perform only a few, say γ (see Figure 4.1), iteration steps to obtain a good enough approximation to the solution of (4.1). This idea can, in a straightforward manner, be applied recursively, using coarser and coarser grids, down to some coarsest grid. On this coarsest grid any solution method may be used (e.g. a direct method or the smoothing process itself if it has sufficiently good convergence properties on the coarsest grid).

Most parts of the considerations in this chapter are independent of the way in which coarser grids are constructed. Usually, however, we have standard coarsening in mind. In particular, in this case, the asymptotic optimality of multigrid methods follows easily from a very simple result on their h -independent convergence (see Section 4.3) and on the computational work needed (see Section 4.4).

4.1 Notation, sequence of grids and operators

Before we provide the notation for a formal description of the multigrid recursion, let us illustrate the structure of one iteration step (cycle) of a multigrid method with a few pictures which are given in Figure 4.1. Here \circ , \square , \backslash and $/$ mean smoothing, solving exactly, fine-to-coarse and coarse-to-fine transfer, respectively. With respect to the computational work (see Section 4.4), mainly the case $\gamma \leq 2$ is of practical interest. For obvious reasons, we refer to the cases $\gamma=1$ and $\gamma=2$ as to *V-cycles* and *W-cycles*, respectively.

two-grid method: three-grid method:



four-grid method:

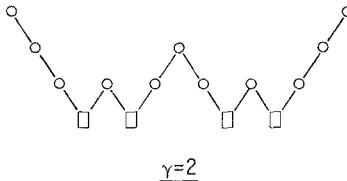
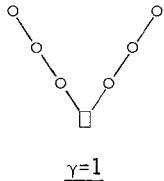


Figure 4.1: Structure of one multigrid cycle for different numbers of grids and different values of γ .

For a formal description of multigrid methods we now use a sequence of increasingly finer grids Ω_{h_ℓ} , characterized by a sequence of meshsizes h_ℓ ($\ell=0,1,2,\dots$). For simplicity, we replace the index h_ℓ by ℓ (for grids, grid functions and grid operators) in the following. For each Ω_ℓ , we assume linear operators

$$\begin{aligned} L_\ell : \mathbb{G}(\Omega_\ell) &\rightarrow \mathbb{G}(\Omega_\ell), & S_\ell : \mathbb{G}(\Omega_\ell) &\rightarrow \mathbb{G}(\Omega_\ell), \\ I_\ell^{\ell-1} : \mathbb{G}(\Omega_\ell) &\rightarrow \mathbb{G}(\Omega_{\ell-1}), & I_{\ell-1}^\ell : \mathbb{G}(\Omega_{\ell-1}) &\rightarrow \mathbb{G}(\Omega_\ell) \end{aligned} \quad (4.2)$$

and discrete equations

$$L_\ell u_\ell = f_\ell \quad (\Omega_\ell) \quad (4.3)$$

(with L_ℓ invertible) to be given. Here $\mathbb{G}(\Omega_\ell)$ denotes the space of gridfunctions on Ω_ℓ . The operators S_ℓ denote the linear iteration operators corresponding to given relaxation methods. The result \bar{w}_ℓ of v relaxation steps (applied to $L_\ell u_\ell = f_\ell$ with first approximation w_ℓ) will be denoted by

$$\bar{w}_\ell = \text{RELAX}^v(w_\ell, L_\ell, f_\ell). \quad (4.4)$$

4.2 Recursive definition of a complete multigrid cycle

A convenient way to define a complete multigrid iteration step (*cycle*) is to use an Algol-like description. The recursive definition of a multigrid cycle then can easily be established using a self-calling procedure. A description of this type is

given in [50], Section 1.3. As an alternative, we here give a description using a flow chart. This may be useful if a multigrid procedure is to be implemented by a FORTRAN program.

We describe one step of a *multigrid iteration* - more precisely of an $(\ell+1)$ -grid iteration - to solve the difference equations

$$L_\ell u_\ell = f_\ell \quad (\Omega_\ell) \quad (4.5)$$

for a fixed $\ell \geq 1$. For this, the grids Ω_k and the operators L_k ($k=\ell, \ell-1, \dots, 0$) as well as $S_k, I_k^{k-1}, I_{k-1}^k$ ($k=\ell, \ell-1, \dots, 1$) are used. The parameters v_1, v_2 and γ are assumed to be fixed (i.e. independent of k and ℓ). If some approximation u_ℓ^j of u_ℓ is given, the calculation of a new approximation u_ℓ^{j+1} proceeds as follows:

If $\ell = 1$: Like in Section 2.3 with Ω_1, Ω_0 instead of Ω_h, Ω_H , respectively.

If $\ell > 1$:

(1) Smoothing part I:

- Compute \bar{u}_ℓ^j by applying v_1 (≥ 0) smoothing steps to u_ℓ^j :

$$\bar{u}_\ell^j := \text{RELAX}^{v_1}(u_\ell^j, L_\ell, f_\ell).$$

(2) Coarse-grid correction:

- Compute the defect:

$$\bar{d}_\ell^j := f_\ell - L_\ell \bar{u}_\ell^j.$$

- Restrict the defect:

$$\bar{d}_{\ell-1}^j := I_\ell^{\ell-1} \bar{d}_\ell^j.$$

- Compute an approximate solution $\tilde{v}_{\ell-1}^j$ of the defect equation on $\Omega_{\ell-1}$

$$L_{\ell-1} \tilde{v}_{\ell-1}^j = \bar{d}_{\ell-1}^j \quad (4.6)$$

by performing $\gamma \geq 1$ iterations of the ℓ -grid method (using the grids $\Omega_{\ell-1}, \Omega_{\ell-2}, \dots, \Omega_0$ and the corresponding grid operators) applied to (4.6) with the zero grid function as first approximation.

- Interpolate the correction:

$$\tilde{v}_\ell^j := I_{\ell-1}^\ell \tilde{v}_{\ell-1}^j.$$

- Compute the corrected approximation on Ω_ℓ :

$$\bar{u}_\ell^j + \tilde{v}_\ell^j.$$

(3) Smoothing part II:

- Compute u_ℓ^{j+1} by applying v_2 (≥ 0) smoothing steps to $\bar{u}_\ell^j + \tilde{v}_\ell^j$:

$$u_\ell^{j+1} := \text{RELAX}^{v_2}(\bar{u}_\ell^j + \tilde{v}_\ell^j, L_\ell, f_\ell).$$

The same process is described in the flow-chart below. There a switching parameter $0 \leq C(k) \leq \gamma$ is introduced to control when to go to a coarser grid and when to go back to a finer grid.

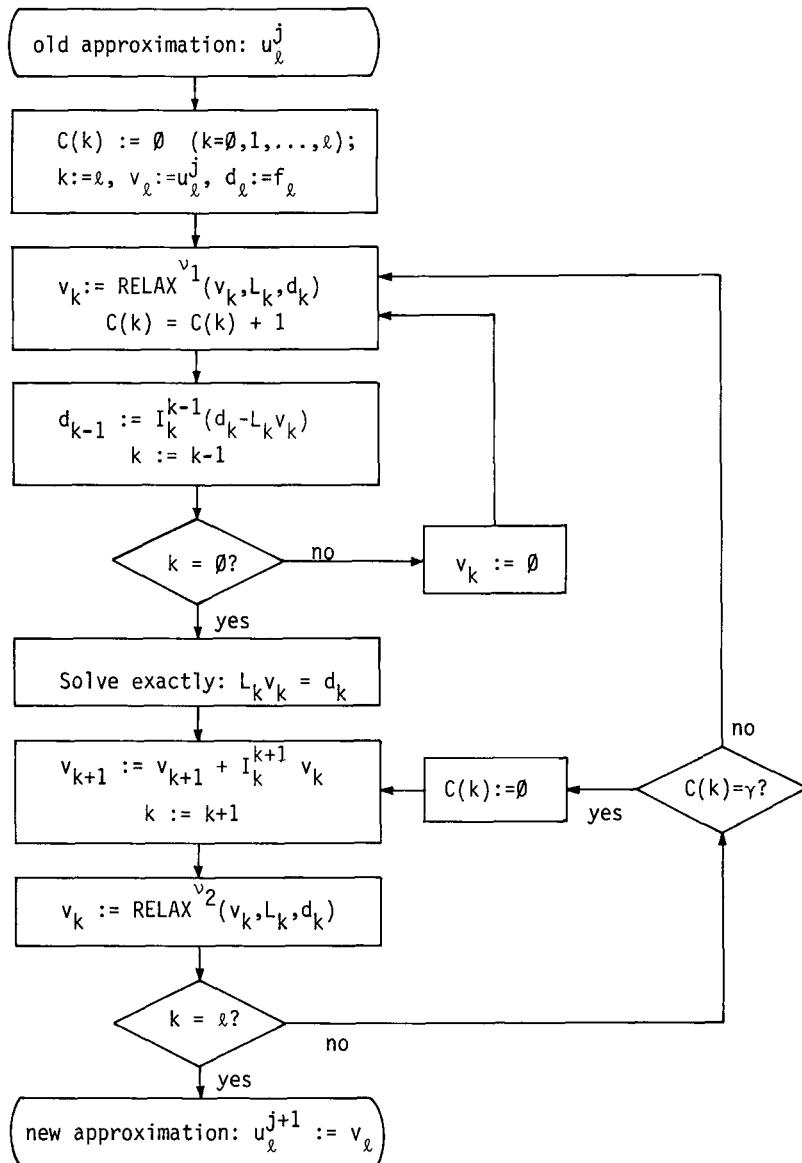


Figure 4.2: Flow-chart for one multigrid iteration step to solve $L_\ell u_\ell = f_\ell$ ($\ell \geq 1$).

So far, we have assumed that the parameters v_1, v_2 and the recursion parameter γ are fixed numbers. This is, of course, not necessary. In particular, γ may depend on k (and possibly on ℓ). Certain combinations of $\gamma=1$ and $\gamma=2$ are indeed used in practice. We will discuss one such choice in the following section.

Remark: For theoretical investigations of multigrid methods (h -independency of convergence factors ρ , etc.) one usually assumes that $h=h_\ell$, the meshsize of the finest grid, tends to 0, or one looks for $\rho^* = \sup \rho$ with respect to h . For such asymptotic investigations, we always regard the coarsest grid to be fixed (meshsize h_0) and let the number ℓ of grids tend to infinity. Only in connection with this conception do we use the term *complete* multigrid methods. (Some authors, however, discuss the convergence properties of multigrid methods not for fixed h_0 , but for a fixed ratio of the finest and coarsest meshsize used. Since the coarsest grid then becomes finer and finer for $h \rightarrow 0$, this assumption is unsatisfactory in several respects, in particular with respect to the total computational work, see Section 4.4.)

4.3 The iteration operator for a complete multigrid cycle; h -independent convergence

The simple theoretical considerations of this section have been presented in a similar form by Hackbusch [50]. Here, we always allow γ to depend on k : $\gamma = \gamma_k$. We will, however, only add the index k if this dependence is to be stressed. By M_ℓ , we denote the iteration operator of the multigrid method described in the previous section.

Lemma 4.1: M_ℓ is given by the following recursion:

$$\begin{aligned} M_1 &= S_1^{v_2} (I_1 - I_0^1 L_0^{-1} I_1^0 L_1) S_1^{v_1} \\ M_{k+1} &= S_{k+1}^{v_2} (I_{k+1} - I_k^{k+1} (I_k - M_k^\gamma) L_k^{-1} I_{k+1}^k L_{k+1}) S_{k+1}^{v_1} \quad (k=1, \dots, \ell-1) \end{aligned} \quad (4.7)$$

An explicit proof of (4.7) can easily be given by means of induction on ℓ . Implicitly, a proof is also contained in the following remark.

Remark: The difference between the $(h_\ell, h_{\ell-1})$ two-grid operator which solves (4.5), namely

$$M_\ell^{\ell-1} = S_\ell^{v_2} (I_\ell - I_{\ell-1}^\ell L_{\ell-1}^{-1} I_\ell^{\ell-1} L_\ell) S_\ell^{v_1}, \quad (4.8)$$

and the above multigrid operator M_ℓ is obviously that

$$L_{\ell-1}^{-1} \text{ is replaced by } (I_{\ell-1} - M_{\ell-1}^\gamma) L_{\ell-1}^{-1}. \quad (4.9)$$

This reflects the fact that the coarse-grid equation (4.6) is solved approximately by γ ($=\gamma_{\ell-1}$) multigrid steps on the grid $\Omega_{\ell-1}$ starting with an initial approximation = 0. (Here we use the simple consideration: If any non-singular system of linear equations $Aw = r$ is solved approximately by γ steps of an iterative method $w^{j+1} = Mw^j + s$ with $w^0 = 0$, then the γ -th iterate can be represented as $w^\gamma = (I - M^\gamma)^{-1}r$.)

For the following norm estimations, it is convenient to write M_ℓ as a perturbation of $M_{\ell-1}^{\ell-1}$. Lemma 4.1 yields

Corollary 4.2: For $k=1, \dots, \ell-1$ the equations

$$M_{k+1} = M_{k+1}^k + A_k^{k+1} M_k^\gamma A_{k+1}^k \quad (4.10)$$

hold, where

$$\begin{aligned} A_k^{k+1} &:= S_{k+1}^{v_2} I_k^{k+1} : G(\Omega_k) \rightarrow G(\Omega_{k+1}), \\ A_{k+1}^k &:= L_k^{-1} I_{k+1}^k L_{k+1} S_{k+1}^{v_1} : G(\Omega_{k+1}) \rightarrow G(\Omega_k) \end{aligned} \quad (4.11)$$

and M_{k+1}^k is as in (4.8) with $k+1$ instead of ℓ .

From this representation, one can immediately derive an estimate for $\|M_\ell\|$, provided that estimates for $\|M_{k+1}^k\|$, $\|A_k^{k+1}\|$ and $\|A_{k+1}^k\|$ ($k \leq \ell-1$) are known. Here $\|\cdot\|$ denotes any reasonable operator norm.

Lemma 4.3: Let the following estimates hold uniformly with respect to k ($\leq \ell-1$):

$$\|M_{k+1}^k\| \leq \sigma^*, \quad \|A_k^{k+1}\| \cdot \|A_{k+1}^k\| \leq C. \quad (4.12)$$

Then we have $\|M_\ell\| \leq \eta_\ell$ where η_ℓ is recursively defined by

$$\eta_1 := \sigma^*, \quad \eta_{k+1} := \sigma^* + C\eta_k^\gamma \quad (k=1, \dots, \ell-1). \quad (4.13)$$

Remark: Clearly, we could also have admitted bounds σ^* and C in (4.12) which depend on k . In particular, this may be advantageous if these bounds achieve their maximal values for small k (which is typical for certain indefinite problems).

From (4.13), one can already conclude the h -(ℓ -)independent convergence of multigrid methods. Mainly, one has to assume that the corresponding (h_{k+1}, h_k) two-grid methods converge for all k with σ^* sufficiently small. Furthermore, one has to make a decision on the choice of γ . We consider two cases:

$$(1) \quad \gamma_k \equiv 2 \quad (k=1, 2, \dots; W\text{-cycle}) \quad (4.14)$$

$$(2) \quad \gamma_k = \begin{cases} 1 & (k \text{ odd}) \\ 2 & (k \text{ even}) \end{cases} \quad (4.15)$$

The second choice (see Figure 4.3) is of particular interest in connection with semi or red-black coarsening (see Section 4.4).

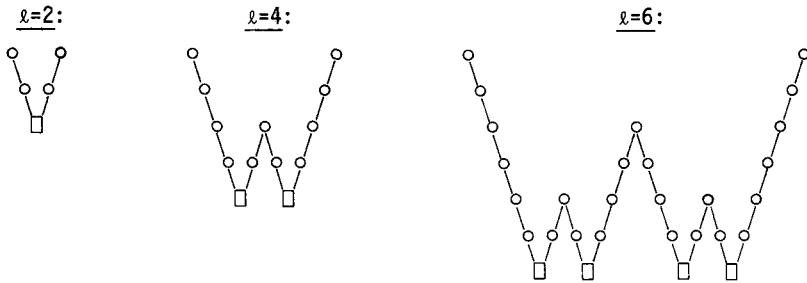


Figure 4.3: Structure of one multigrid cycle for ℓ even and γ as defined in (4.15)

Corollary 4.4: In the case (1) we have the following uniform estimate provided that $4C\sigma^* \leq 1$:

$$\|\mathbf{M}_\ell\| \leq n := (1 - \sqrt{1 - 4C\sigma^*})/2C \leq 2\sigma^* \quad (\ell \geq 1). \quad (4.16)$$

Similarly, in the case (2) we obtain

$$\|\mathbf{M}_\ell\| \leq \begin{cases} (1 - \sqrt{1 - 4C^2(1+C)\sigma^*})/2C^2 \leq 2\sigma^*(1+C) & (\ell \text{ even}) \\ (1 - 2C^2\sigma^* - \sqrt{1 - 4C^2(1+C)\sigma^*})/2C^3 \leq \sigma^*(1+2C)/C & (\ell \text{ odd}) \end{cases} \quad (4.17a)$$

if $4C^2(1+C)\sigma^* \leq 1$.

Remarks:

- (1) If $\gamma=2$, i.e. if W-cycles are used, we obviously obtain, under the sole assumption that σ^* is small enough, $n \approx \sigma^*$ for the bound n given in (4.16). For example, if $C=1$, then we get from (4.16)

$$n \leq 0.113 \quad \text{if } \sigma^* \leq 0.1.$$

(Typically, the constant C is ≥ 1 , but not very large. For instance, if we use $\|\cdot\| = \|\cdot\|_E$ in the sample method treated in Sections 3.1 - 3.5, we obtain $C = 1$, independent of v_1 and v_2 . For $\|\cdot\| = \|\cdot\|_S$ we have $C \leq \sqrt{2}$ for all choices of v_1 and v_2 and we have $C \rightarrow 1$ if $v_1 \rightarrow \infty$.) In this sense we may say: If a given two-grid method converges sufficiently well (small enough σ^*), then the corresponding multigrid method with $\gamma=2$ will have similar convergence pro-

roperties. In this respect, for the construction of multigrid methods, it is usually sufficient to analyze only the corresponding two-grid method. Furthermore, there usually is no need to work with $\gamma > 2$.

- (2) If γ is defined as in (4.15), the bounds for $\|\mathbf{M}_\ell\|$ become somewhat worse than those for $\gamma=2$. For example, from (4.17a) we obtain that the upper bound of $\|\mathbf{M}_\ell\|$ (ℓ even) approaches $\sigma^*(1+C)$ if σ^* is small enough (instead of $n \approx \sigma^*$ in the case $\gamma=2$). On the other hand, a smaller amount of numerical work is needed for one cycle (cf. Section 4.4) than for the W-cycle.
- (3) If $\gamma=1$, i.e. if V-cycles are used, Lemma 4.3 gives no ℓ -independent upper bound for $\|\mathbf{M}_\ell\|$ if $C \geq 1$. However, instead of $\gamma=1$ one could use, e.g., $\gamma=\gamma_k$ with $\gamma_k=1$ if $\ell \geq k \leq \ell-\ell_0$ and $\gamma_k=2$ otherwise. For larger values of ℓ_0 , this would result in only a very slight increase of the computational work compared to the V-cycle. For a cycle of this type, Lemma 4.3 could, in principle, be used to derive ℓ -independent bounds for $\|\mathbf{M}_\ell\|$. If ℓ_0 is large, we would then, however, have to assume σ^* to be very small and the estimate would become completely unrealistic from a practical point of view.

There is another approach in proving ℓ -independent convergence of multigrid methods which is also applicable to V-cycles. This approach was first presented by Braess [13]. Hackbusch has also incorporated the corresponding idea into his theory (see [50], Section 4.3). The approach is based on the energy norm and makes essential use of the following assumptions:

- L_ℓ is symmetric and positive definite.
- The restriction operators I_k^{k-1} and the interpolation operators I_{k-1}^k are adjoint to each other: $I_k^{k-1} = (I_{k-1}^k)^*$ ($k = 1, \dots, \ell$).
- The coarse-grid difference operators L_k ($k = 0, 1, \dots, \ell-1$) are defined to be the "Galerkin operators" (see Section 4.5): $L_{k-1} := I_k^{k-1} L_k I_{k-1}^k$ ($k=1, \dots, \ell$).
- The difference operators and the smoothing operators are supposed to commute. (This assumption can easily be weakened up to a certain extend [100]).

We remark that under these assumptions the h -independent convergence of V-cycle methods can also be shown in the framework of local Fourier analysis [100].

In practice, if σ^* is small enough such that W-cycles have good convergence properties, usually also V-cycles may be used (even if the above listed assumptions are not satisfied). Often, the convergence properties of V-cycles are somewhat worse than those of W-cycles, but with respect to efficiency nevertheless competitive (also see Section 10.2).

4.4 Computational work and efficiency

The fact that a certain method has an h -independent convergence factor says nothing about its efficiency as long as the computational work is not taken into account. In the following, we will estimate the computational work of a multigrid method. It will turn out that the number of arithmetic operations needed for one multigrid cycle is proportional to the number of grid points of the finest grid (under quite natural assumptions which are satisfied for reasonable multigrid methods). Together with the h -independent convergence, this means that multigrid methods are asymptotically optimal. The constant of proportionality depends on the type of the cycle, i.e. on γ , the type of coarsening and the other multigrid components. For reasonable choices of these components, the constants of proportionality are small.

From the recursive definition of a multigrid cycle as given in Section 4.2, it immediately follows that the *computational work* W_ℓ per multigrid cycle Ω_ℓ is recursively given by

$$W_1 = W_1^0 + W_0, \quad W_{k+1} = W_{k+1}^k + \gamma_k W_k \quad (k = 1, \dots, \ell-1). \quad (4.18)$$

Here W_{k+1}^k denotes the computational work of one (h_{k+1}, h_k) two-grid cycle excluding the work needed to solve the defect equations on Ω_k , and W_0 denotes the work needed to compute the exact solution on the coarsest grid Ω_0 . By "computational work", we always denote some reasonable measure, for example, the number of arithmetic operations needed. If γ is independent of k , we obtain from (4.18)

$$W_\ell = \sum_{k=1}^{\ell} \gamma^{\ell-k} W_k^{k-1} + \gamma^{\ell-1} W_0 \quad (\ell \geq 1). \quad (4.19)$$

Let us first discuss the case of standard coarsening with γ independent of k . Obviously, we have in this case

$$\mathcal{N}_k \doteq 4 \mathcal{N}_{k-1} \quad (k = 1, 2, \dots) \quad (4.20)$$

where $\mathcal{N}_k = \# \Omega_k$ (number of gridpoints on Ω_k) and " \doteq " means equality up to lower order terms (boundary effects). Furthermore, we assume that the multigrid components (relaxation, computation of defects, fine-to-coarse and coarse-to-fine transfers) require a number of arithmetic operations per point of the respective grids which is bounded by a constant C , independent of k :

$$W_k^{k-1} \doteq C \mathcal{N}_k \quad (k = 1, 2, \dots). \quad (4.21)$$

(As above, " \doteq " means " \leq " up to lower order terms.) This is a typical feature of multigrid methods. In particular, (4.21) is satisfied with \doteq instead of \leq if all multigrid components are constructed in the same way on all grids.

Under these assumptions, one immediately obtains from (4.19) the following estimate for the total computational work W_ℓ of one complete multigrid cycle:

$$W_\ell \leq \begin{cases} \frac{4}{3} C N_\ell & (\text{for } \gamma=1) \\ 2 C N_\ell & (\text{for } \gamma=2) \\ 4 C N_\ell & (\text{for } \gamma=3) \\ O(N_\ell \log N_\ell) & (\text{for } \gamma=4) \end{cases} \quad (4.22)$$

This estimate of W_ℓ together with the h -independent convergence as discussed in the previous section shows the asymptotic optimality of iterative multigrid methods if $\gamma \leq 3$ and standard coarsening is used. (As mentioned in the previous section, for V-cycles h -independent convergence has been proved - so far - only under certain additional assumptions. In practice, however, this convergence behavior can be observed in much more general situations. In this respect, we have asymptotic optimality also for $\gamma=1$.)

Remark: w_k^{k-1} in (4.18) is determined by the computational work needed for the individual multigrid components of the (h_k, h_{k-1}) two-grid method, namely

$$w_k^{k-1} = (v w_0 + w_1 + w_2) N_k. \quad (4.23)$$

Here $v=v_1+v_2$ is the number of relaxation steps used; w_0 , w_1 and w_2 are measures for the computational work per grid point of N_k needed for the single components, namely

w_0 : one relaxation step on N_k ;

w_1 : computation of the defect and its transfer to N_{k-1} ;

w_2 : interpolation of the correction to N_k and its addition to the previous approximation.

Usually (in particular, when the multigrid components are constructed in the same way on all grids), w_0 , w_1 and w_2 are independent of k . In general, however, they may depend on k .

Example 4.1: If the multigrid algorithm which corresponds to our sample method (Section 3.1) is arranged suitably, we obtain the following operation count

	w_0	w_1	w_2	w_k^{k-1}
$+$ /-	5	$25/4$	$7/4$	$(5v + 8) N_k$
*	1	$5/4$	$3/4$	$(v + 2) N_k$

If we count additions and multiplications in the same way, we obtain (4.22) with $C=6v+10$. Further examples, namely for more efficient methods and more general problems, will be given in Chapters 8, 9 and 10.

For other grid coarsenings than the standard coarsening,

$$\mathcal{N}_k = \tau \mathcal{N}_{k-1} \quad (k = 1, 2, \dots) \quad \text{with } \tau > 1.$$

and for γ independent of k we obtain

$$W_\ell \leq \begin{cases} \frac{\tau}{\tau-\gamma} C \mathcal{N}_\ell & (\text{for } \gamma < \tau) \\ O(\mathcal{N}_\ell \log \mathcal{N}_\ell) & (\text{for } \gamma = \tau) \end{cases} \quad (4.24)$$

instead of (4.22).

If we consider, for example, red-black coarsening or semi-coarsening, we have $\tau=2$. In this case, we already see that W -cycles do not yield an asymptotically optimal multigrid method: For fixed γ , only $\gamma=1$ yields a cycle for which W_ℓ is proportional to \mathcal{N}_ℓ . Because of the theoretical restrictions in proving the h -independent convergence of pure V-cycle iterations, the choice $\gamma=\gamma_k$ as given in (4.15) is of particular interest for $\tau=2$. We obtain for $\tau=2$:

$$W_\ell \leq \begin{cases} 2C \mathcal{N}_\ell & (\text{for } \gamma = 1) \\ 3C \mathcal{N}_\ell & (\text{for } \ell \text{ even and } \gamma \text{ as defined in (4.15)}) \\ 4C \mathcal{N}_\ell & (\text{for } \ell \text{ odd and } \gamma \text{ as defined in (4.15)}) \\ O(\mathcal{N}_\ell \log \mathcal{N}_\ell) & (\text{for } \gamma = 2). \end{cases}$$

Clearly, for $\gamma=\gamma_k$ we have to use the more general formula

$$W_\ell = \sum_{k=1}^{\ell} \left(\prod_{j=k}^{\ell-1} \gamma_j \right) W_k^{k-1} + \left(\prod_{j=1}^{\ell-1} \gamma_j \right) W_0 \quad (\ell \geq 1)$$

instead of (4.19).

Remark: There are, of course, many other possible choices of $\gamma=\gamma_k$ and still more general ways to construct a cycle. We mention here just the so-called *F-cycle* [25] which is illustrated in Figure 4.4. The corresponding iteration operator M_ℓ^F is recursively defined by

$$\mathbf{M}_1^F = \mathbf{M}_1 \quad (\text{as in (4.7)})$$

$$\mathbf{M}_{k+1}^F = S_{k+1}^{V_2} (I_{k+1} - I_k^{k+1} (I_k - \mathbf{M}_k^V \mathbf{M}_k^F) L_k^{-1} I_{k+1}^k L_{k+1}) S_{k+1}^{V_1} \quad (k=1, \dots, \ell-1).$$

Here \mathbf{M}_k^V is the corresponding V-cycle iteration operator (i.e. (4.7) with $\gamma=1$ and k instead of $k+1$).

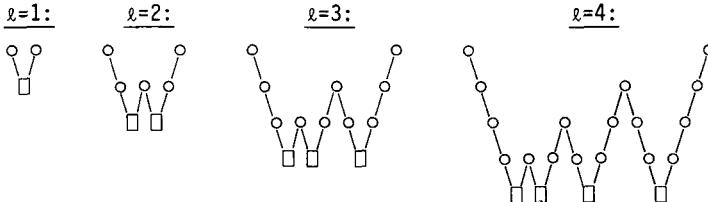


Figure 4.4: Structure of an F-cycle

In *self-adaptive* algorithms as proposed by Brandt [25],[18], no fixed cycles are used: Switching from one grid to another (to a finer or a coarser one) is controlled by suitable accommodative criteria.

Efficiency of multigrid iterations. Let us make some comments on the questions of efficiency and of efficiency measures for multigrid methods. Trivially, the efficiency of an iterative method is determined by both its convergence factor and the computational work needed per iteration step. Reasonable measures of efficiency are

$$\text{op}(\epsilon) := w \frac{\log \epsilon}{\text{T}_{\text{op}}} , \quad (4.25)$$

the number of numerical operations (per point of the finest grid) required to reduce the error by a factor of ϵ , or

$$\rho_{\text{eff}} := \rho^{1/w} . \quad (4.26)$$

Here ρ characterizes the convergence factor of the method at hand and w the work needed per iteration step and per point of the finest grid.

Although (4.25) and (4.26), at first sight, look simple and well-defined, there is little uniformity in the multigrid literature with respect to the definition of ρ and w : As for ρ , one might use (empirical) asymptotic convergence factors or (empirical) error reducing factors (norms) for a suitable multigrid cycle (e.g., V- or W-cycle). Similarly, there are different possible definitions of w .

A choice which is usually reasonable and which avoids the arbitrariness in the decision about the type of cycle and the norms, is for ρ to use the asymptotic convergence factor of the corresponding two-grid operator $M_\ell^{\ell-1}$ (or its supremum

ρ^* with respect to ℓ). This choice corresponds to the work quantity

$$w := w_\ell^{\ell-1} / \mathcal{N}_\ell \quad (4.27)$$

in a natural way.

One still has to decide about the question of how to count the different arithmetic operations. Furthermore, there are several multigrid components (like linewise relaxation and ILU-smoothing, see Chapters 8 and 9) which permit certain precomputations. As these computations have to be performed only once, they may be excluded from the operation count as long as the MG method at hand is considered as a purely iterative solver. (The above view of how to measure efficiency is no longer correct if multigrid methods are used in the full multigrid mode (see Chapter 6).)

In Chapters 8 and 9 we give results on ρ^* , σ_s^* , σ_E^* etc. for several methods along with an operation count in terms of (4.27).

4.5 Other coarse-grid operators, extensions

In Section 4.1 we have assumed a fixed sequence of difference operators L_ℓ ($\ell=0,1,2,\dots$) to be given. We had in mind there that the L_ℓ were chosen in a uniform manner on all grid Ω_ℓ , e.g. always using the same discretization. For fixed ℓ , the grid equation (4.5) was solved by using the L_k ($k=\ell-1,\ell-2,\dots,0$) as coarse-grid difference operators. Another way of defining these coarse-grid operators (maintaining the sequence of transfer operators) has already been mentioned in Section 2.4.2:

- *The Galerkin approach:* Here, for a given fine-grid operator L_ℓ , the grid equation (4.5) is solved using coarse-grid operators L_k ($k=\ell-1,\dots,0$) which are recursively defined by

$$L_k := I_{k+1}^k L_{k+1} L_k^{k+1} \quad (k = \ell-1, \ell-2, \dots, 0).$$

From this recursion one sees that no fixed sequence of coarse-grid operators is defined. Instead, the L_k depend on the operator L_ℓ which is given on the finest grid. This means that we have to work with a "triangular" scheme of operators

$$L_k^{(\ell)} \quad (k = \ell, \ell-1, \dots; \ell = 0, 1, 2, \dots)$$

where

$$L_\ell^{(\ell)} = L_\ell \quad \text{and} \quad L_k^{(\ell)} := I_{k+1}^k L_{k+1}^{(\ell)} I_k^{k+1} \quad (k = \ell-1, \ell-2, \dots, 0). \quad (4.28)$$

The description of the multigrid method given in Section 4.2 carries over to this more general case: One simply has to replace L_ℓ by $L_\ell^{(\ell)}$ and L_k by $L_k^{(\ell)}$ ($k = \ell-1, \dots, 0$) for all ℓ . (Of course, relaxation processes now have to be given with smoothing operators $S_k^{(\ell)}$ corresponding to the $L_k^{(\ell)}$.) All further results of this chapter still hold after a few obvious changes. In particular, all operators occurring in (4.10) now depend not only on k but also on ℓ . Lemma 4.3 carries over to this situation. One now, however, needs norm estimates (4.12) which hold for all operators within the above "triangular" scheme of operators.

The main practical difference to the approach in the previous sections is that the $L_k^{(\ell)}$ are not known in advance but have to be calculated from the recursion formula (4.28):

Example 4.2: (a) If $L_\ell^{(\ell)}$ ($\ell = 0, 1, 2, \dots$) and $I_\ell^{\ell-1}$, $I_{\ell-1}^\ell$ ($\ell = 1, 2, \dots$) are defined as in the sample method (see Section 3.1, in particular (3.3), (3.4)), we obtain

$$L_{\ell-1}^{(\ell)} = I_\ell^{\ell-1} L_\ell^{(\ell)} I_{\ell-1}^\ell \triangleq \frac{1}{h_{\ell-1}^2} \begin{bmatrix} -1/4 & -1/2 & -1/4 \\ -1/2 & 3 & -1/2 \\ -1/4 & -1/2 & -1/4 \end{bmatrix} h_{\ell-1}. \quad (4.29)$$

For $\ell \rightarrow \infty$ and k fixed, $L_k^{(\ell)}$ tends to a difference operator which is characterized by the difference star

$$\frac{1}{h_k^2} \begin{bmatrix} -1/3 & -1/3 & -1/3 \\ -1/3 & 8/3 & -1/3 \\ -1/3 & -1/3 & -1/3 \end{bmatrix} h_k. \quad (4.30)$$

This is a well-known approximation for the Laplace operator which occurs in connection with bilinear finite elements.

(b) If, for $k=\ell$, (4.30) is used as difference operator $L_\ell^{(\ell)}$ on the finest grid Ω_ℓ (and if $I_\ell^{\ell-1}$, $I_{\ell-1}^\ell$ are chosen as above), $L_\ell^{(\ell)}$ is "reproduced" by the Galerkin-recursion (4.28): $L_{\ell-1}^{(\ell)}$ is just (4.30) with $k=\ell-1$.

(c) The 5-point Laplace difference operator $L_\ell^{(\ell)}$ from (a) is reproduced by the Galerkin recursion, if we use different transfer operators, namely the 7-point operators [109]

$$I_l^{l-1} \triangleq \frac{1}{8} \begin{bmatrix} 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 \end{bmatrix}^h_{l-1} \quad , \quad I_{l-1}^l \triangleq \frac{1}{2} \begin{bmatrix} 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 \end{bmatrix}^h_l . \quad (4.31)$$

- *Reduction-type approach:* For 1D-problems (discrete ordinary boundary value problems), it usually is possible to define operators I_h^{2h} and L_{2h} such that the coarse-grid equation is equivalent to the original grid equation for $x \in \Omega_{2h}$. If, in the coarse-to-fine transfer, the original grid equation is used (for $x \in \Omega_h \setminus \Omega_{2h}$, cf. Section 2.4.4), then one obtains the exact discrete solution on Ω_h after one cycle only: These methods degenerate to direct solvers (without performing smoothing steps), and they coincide with so-called 1D-reduction methods [87],[104] (also called cyclic reduction [99].)

Such a transformation of a fine-grid equation to an equivalent equation on some coarser grid can also be carried out for certain 2D- (and 3D-) problems. Corresponding methods are known as *total*, *cyclic* and *alternating reduction methods* [87],[29], [88]. These methods differ mainly with respect to the coarsening (successive red-black coarsening, successive semi-coarsening and alternating semi-coarsening, respectively). All of these methods have the disadvantage of being directly applicable only to a rather small class of problems. Moreover, the coarse-grid operators (and the fine-to-coarse operators) become more and more complicated. For the total and alternating reduction method, however, the corresponding difference stars can be "truncated". This possibility (in combination with certain smoothing processes) is the basis of the *MGR principle* [82],[37] yielding particularly efficient multigrid solvers (MG-TR and MG-AR methods).

The reduction-type approach shows a connection between multigrid methods and certain direct solvers for a given discrete problem. The following approach is, on the contrary, closer related to the original continuous problem and not to a fixed discrete problem.

- *Double discretization:* In principle, multigrid methods may be applied with different operators in the relaxation process (L_ℓ) and in the process of the calculation of defects (L_ℓ^*). Brandt [25],[23] recommends such "double discretization" multigrid methods for certain applications. For example, the operators L_ℓ^* may be of a higher order of consistency than the operators L_ℓ . As a consequence, one can expect to obtain higher order accuracy although only low order operators are employed for smoothing. In particular, it is possible to use unstable higher order operators L_ℓ^* for the defect computation.

The latter choice is of particular interest in connection with singular perturbation problems. For instance, the L_ℓ^* may be (unstable) operators of second order consistency based on central differencing, whereas the L_ℓ are defined by introducing a certain amount of artificial ellipticity (leading to first order consistency). This concept is discussed in [25] and, in detail, in [21],[12].

The double discretization idea is clearly related to the defect correction principle [96],[5]. Possibilities to combine multigrid and defect correction methods are discussed in several papers (see, e.g., [25],[5],[51],[56]).

5. Nonlinear multigrid methods, the full approximation scheme (FAS)

So far, we have discussed multigrid methods only in connection with linear problems. Clearly, if a linear multigrid method is combined with some iterative (global) linearization process like Newton's method, it can also be used for the solution of nonlinear problems. This "indirect" application of (linear) multigrid methods to nonlinear problems is more or less straightforward (see Section 5.1).

The multigrid idea can, however, also be applied directly to nonlinear problems. Again, we only need a procedure for smoothing errors and a procedure for approximating corrections on coarser grids. For error smoothing, suitable relaxation methods for nonlinear equations now have to be used. This "direct" approach leads to nonlinear multigrid methods in form of the so-called *full approximation technique* (*full approximation scheme*, "FAS", introduced by Brandt [16],[17]). In this approach, no global linearization has to be carried out explicitly (except perhaps on the coarsest grid). We will describe the nonlinear multigrid methods in Section 5.2.

In Section 5.3 we point out the close relationship between the nonlinear multigrid method and the indirect approach, giving some numerical results. This relationship can be exploited for a convergence theory of the nonlinear methods. We do not give such proofs here (see Hackbusch [50]). However, some simple theoretical considerations concerning the appropriate choice of relaxation methods for nonlinear problems are given in Section 5.4.

In Section 5.5.1, we make some remarks on the multigrid treatment of an exemplary bifurcation problem, in which a global constraint has to be taken into account. Furthermore, there are several specific features of the full approximation scheme which are the starting point for more sophisticated multigrid techniques. In this respect, FAS is of interest for linear problems also (although it is then theoretically equivalent to the usual linear scheme). Some of the more sophisticated techniques will be sketched in Section 5.5.2.

5.1 Indirect application of multigrid methods to nonlinear problems

In the following, we consider a discrete elliptic equation

$$L_h u_h = f_h \quad (\Omega_h). \quad (5.1)$$

Here $L_h : \mathbb{G}(\Omega_h) \rightarrow \mathbb{G}(\Omega_h)$ is assumed to be a nonlinear operator; $f_h \in \mathbb{G}(\Omega_h)$ is a given grid function (which is introduced for technical reasons only). In order not to have too many formal requirements and restrictions, we assume explicitly only that this equation has at least one isolated solution u_h . All other assumptions are

implicitly contained in the following considerations.

For the solution of (5.1), an iterative (global) linearization method

$$L_h u_h^j + L_h^j v_h^j = f_h, \quad u_h^{j+1} = u_h^j + v_h^j \quad (j = 0, 1, 2, \dots) \quad (5.2)$$

may be used. Here L_h^j is some linear approximation of $L_h'(u_h^j)$ which characterizes the iteration process. In particular, we consider Newton's method ($L_h^j = L_h'(u_h^j)$). In each step of the iteration (5.2), a (linear) multigrid method can be applied to solve the linear equations

$$L_h^j v_h^j = d_h^j := f_h - L_h u_h^j. \quad (5.3)$$

One way to combine Newton's method with an iterative linear multigrid method for (5.3) (of the type shown in Figure 4.2), is to adapt the number of multigrid iterations in each Newton step. Here the aim is to exploit the convergence speed of Newton's method as far as possible. For example, if Newton's method converges quadratically, the number of MG iterations should roughly be doubled from one Newton step to the next. The main problem in this approach is that one has to use an appropriate control technique in order to obtain the information needed about the convergence of Newton's method. We shall refer to this approach as to *method I*.

Another possibility is to fix the number of multigrid iterations per Newton step. For example, one may perform only one multigrid iteration per Newton step. As a consequence, Newton's method is, of course, truncated to a linearly convergent method. A disadvantage of this approach, which we will refer to as *method II*, is the larger amount of linearization work. On the other hand, no control technique is needed as in I.

A few numerical results and a short comparison with a nonlinear multigrid method of FAS type will be given in Section 5.3. This comparison refers to the special case that (5.1) is the 5-point discretization of

$$L^\Omega u = -\Delta u + g(x, u) = f^\Omega(x) \quad (x \in \Omega) \quad (5.4)$$

with Dirichlet boundary conditions on a bounded region Ω . In terms of (5.2), (5.3), Newton's method then reads as

$$L_h v_h^j + c_h^j(x) v_h^j = d_h^j(x) \quad (x \in \Omega_h) \quad \text{with} \quad c_h^j(x) = \frac{\partial g}{\partial u}(x, u_h^j(x)). \quad (5.5)$$

Here L_h is given by the 5-point discretization of $-\Delta$. Thus, in each Newton step, a discrete Helmholtz-like equation has to be solved. With respect to algorithmic simplifications, let us add the following

Remark: There are several reasons, why, in practice, Newton's method is often replaced by - only linearly convergent - "approximate" Newton's methods. Very simple approximate methods are obtained when, for instance, $c_h^j(x)$ in (5.5) is replaced by

$$c_h^0(x) := \frac{\partial g}{\partial u}(x, u_h^0(x)) \quad (\text{modified Newton's method})$$

or by a constant, e.g.

$$\hat{c}_h^j := 1/2 (\min_x c_h^j(x) + \max_x c_h^j(x)). \quad (5.6)$$

The latter simplification is of particular interest in connection with the application of so-called direct Fast Elliptic Solvers (like Buneman's algorithm) for which a constant Helmholtz- c is required. We point out that a simplification of the type (5.6) is not needed if multigrid methods are used. (The application of Fast Elliptic Solvers has been studied systematically in [90] in connection with a nonlinear parabolic problem.)

5.2 The full approximation scheme

Similar to the linear case, the nonlinear FAS multigrid method can be recursively defined on the basis of an FAS two-grid method. Thus we again start with the description of one iteration step of the (h,H) two-grid method for (5.1), computing u_h^{j+1} from u_h^j . An illustration of this step, which is analogous to the one given in Figure 2.4 for the linear two-grid method, is given in Figure 5.1.

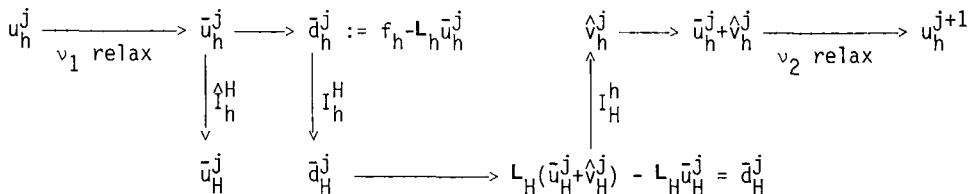


Figure 5.1: FAS (h,H) two-grid method

In this description, "relax" stands for a nonlinear relaxation procedure which has suitable error smoothing properties. As in the linear case, v_1 smoothing steps are performed before and v_2 smoothing steps are performed after the coarse-grid correction. In contrast to the linear case, not only is the defect \tilde{d}_h^j transferred to the coarse grid (by some linear operator I_h^H), but also the relaxed approximation \tilde{u}_h^j itself (by some linear operator \hat{I}_h^H , which may be different from I_h^H). This is necessary, as in the nonlinear case the Ω_h -defect equation is given by

$$L_h(\bar{u}_h^j + v_h^j) - L_h \bar{u}_h^j = \bar{d}_h^j. \quad (5.7)$$

This equation is approximated on Ω_H by

$$L_H(\bar{u}_H^j + \hat{v}_H^j) - L_H \bar{u}_H^j = \bar{d}_H^j, \quad (5.8)$$

or equivalently by

$$L_H w_H^j = \bar{d}_H^j + L_H \bar{u}_H^j, \quad \hat{v}_H^j := w_H^j - \bar{u}_H^j. \quad (5.9)$$

This means that in the FAS mode on the coarse grid, one does not solve for the correction \hat{v}_H^j , but rather for the "full approximation" w_H^j . Of course, transferred back to the fine grid Ω_h , is not w_H^j but the correction \hat{v}_H^j . This is important since only correction (and defect) quantities are smoothed by relaxation processes and can therefore be approximated well on coarser grids (see the explanations in Section 2, which in principle apply also to the nonlinear case). Clearly, if L_h is a linear operator, the FAS two-grid method is equivalent to the linear method, see Figure 2.4, which is called the *correction scheme* (CS) by Brandt.

In the corresponding nonlinear multigrid process, the nonlinear coarse-grid equation in (5.9) is not solved exactly, but approximately by several multigrid steps using still coarser grids. This leads to the following algorithmic description of one step of the FAS multigrid method. Here we use notations analogous to those in Sections 4.1, 4.2. In particular, we assume a sequence of grids Ω_ℓ and grid operators $L_\ell, I_\ell^{\ell-1}, I_\ell^{\ell-1}, I_{\ell-1}^\ell$ etc. to be given. One *FAS multigrid* (more precisely: $(\ell+1)$ -grid) step for the solution of

$$L_\ell u_\ell = f_\ell \quad (\ell \geq 1, \text{ fixed}) \quad (5.10)$$

proceeds as follows:

If $\ell = 1$, we just have the two-grid method described above with Ω_0 and Ω_1 instead of Ω_H and Ω_h , respectively.

If $\ell > 1$:

(1) Smoothing part I:

- Compute \bar{u}_ℓ^j by applying $v_1 (\geq 0)$ smoothing steps to u_ℓ^j :

$$\bar{u}_\ell^j := \text{RELAX}^{v_1}(u_\ell^j, L_\ell, f_\ell).$$

(2) Coarse-grid correction:

- Compute the defect:

$$\bar{d}_\ell^j := f_\ell - L_\ell \bar{u}_\ell^j.$$

- Restrict the defect:

$$\bar{d}_{\ell-1}^j := I_{\ell}^{\ell-1} \bar{d}_{\ell}^j.$$

- Restrict \bar{u}_{ℓ}^j :

$$\bar{u}_{\ell-1}^j := \hat{I}_{\ell}^{\ell-1} \bar{u}_{\ell}^j.$$

- Compute an approximate solution $\tilde{w}_{\ell-1}^j$ of

$$L_{\ell-1} w_{\ell-1}^j = \bar{d}_{\ell-1}^j + L_{\ell-1} \bar{u}_{\ell-1}^j \quad (5.11)$$

by applying $\gamma \geq 1$ steps of the FAS ℓ -grid method (using the grids $\Omega_0, \dots, \Omega_{\ell-1}$) to (5.11) with $\bar{u}_{\ell-1}^j$ as first approximation. Then compute the correction

$$\tilde{v}_{\ell-1}^j := \tilde{w}_{\ell-1}^j - \bar{u}_{\ell-1}^j.$$

- Interpolate the correction:

$$\tilde{v}_{\ell}^j := I_{\ell-1}^{\ell} \tilde{v}_{\ell-1}^j.$$

- Compute the corrected approximation on Ω_{ℓ} :

$$\bar{u}_{\ell}^j + \tilde{v}_{\ell}^j.$$

(3) Smoothing part II:

- Compute u_{ℓ}^{j+1} by applying $v_2 (\geq 0)$ smoothing steps to $\bar{u}_{\ell}^j + \tilde{v}_{\ell}^j$:

$$u_{\ell}^{j+1} := \text{RELAX}^{v_2}(\bar{u}_{\ell}^j + \tilde{v}_{\ell}^j, L_{\ell}, f_{\ell}).$$

One sees from this description that no global linearization is needed in the FAS multigrid process, except perhaps on the coarsest grid. Apart from that, only (non-linear) relaxation methods are required as well as (linear) fine-to-coarse and coarse-to-fine transfer operators.

Concerning the concrete choice of the occurring multigrid components, one can orient oneself to the corresponding linearized problem. For the latter, techniques such as given in Chapters 3,7,8 and 9, can be applied. As to the relaxation methods, there usually exist (several) nonlinear analogs to a given linear relaxation method (see, e.g., [80]). We will make some remarks about the smoothing properties of a simple nonlinear relaxation method in Section 5.4.

5.3 A simple example

The nonlinear multigrid method as described in the previous section and the indirect multigrid approaches as outlined in Section 5.1 are quite different algorithmically, but closely related from a theoretical point of view. In particular, consider one iteration step of method II (cf. Section 5.1; one linear multigrid cy-

cle per linearization step) and one FAS cycle. Without going into details, we only mention that - apart from the solution process on the coarsest grid - the main difference between these two cycles lies in the relaxation process (which in the one case refers to L_h and in the other case refers to its current linearization L_h^j). To make this clear, it is useful to write the linear multigrid cycle in the FAS form also.

The similarity of these approaches is reflected by the numerical results for the following

Example:

$$\mathbf{L}^\Omega u = -\Delta u + e^u = f^\Omega(x) \quad (x \in \Omega), \quad u = f^\Gamma(x) \quad (x \in \Gamma) \quad (5.12)$$

with solution $u(x) = \sin 3(x_1 + x_2)$. The domain Ω is composed of semicircles and straight lines as shown in Figure 5.2. This problem is discretized with the usual 5-point formula (and $h=h_{x_1}=h_{x_2}$) except for grid points near the boundary, where the Shortley-Weller approximation is used (cf. Section 10.1).

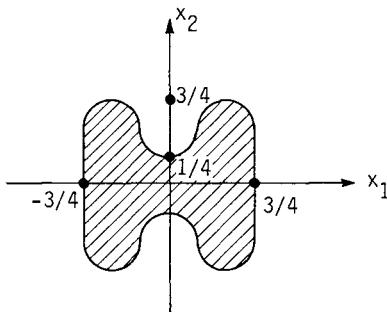


Figure 5.2: Domain Ω treated in (5.12)

Table 5.1 shows some numerical results if multigrid methods are applied to this problem. Here the columns I and II refer to the indirect methods I and II as described in Section 5.1. In II, one MG cycle is performed per (global) Newton-step, in I the number of MG cycles is doubled from one Newton-step to the next. The concrete (linear) MG program used is a version of the MGØ1 program collection described in Sections 10.1 and 10.2 (with $v_1=2$, $v_2=1$ and $\gamma=2$). The last column in Table 5.1 shows the corresponding FAS results. The concrete algorithm used here is a nonlinear analog of the linear one described in Section 10.2: The RB relaxation is replaced by a corresponding nonlinear relaxation method (performing one Newton step for each single equation in relaxing at the corresponding grid point). The operator \hat{I}_h^H which occurs in the description of the FAS method, is chosen to be straight injection (cf. Section 3.6). For all methods, the zero grid function is used as first approximation.

multigrid steps	method		
	I	II	FAS
1	0.18(+2)	0.18(+2)	0.14(+2)
2	0.20(0)	0.20(0)	0.20(0)
3	0.86(-2)	0.55(-2)	0.54(-2)
4	0.14(-3)	0.14(-3)	0.14(-3)
5	0.43(-5)	0.42(-5)	0.42(-5)
6	0.13(-6)	0.13(-6)	0.13(-6)
7	0.47(-8)	0.39(-8)	0.39(-8)
8	0.13(-9)	0.12(-9)	0.12(-9)
9	0.42(-11)	0.40(-11)	0.39(-11)

Table 5.1: Behavior of the $\| \cdot \|_2$ -error (with respect to the discrete solution, $h=1/32$) in case of direct and indirect applications of multigrid methods to problem (5.12). For the indirect methods I and II, horizontal lines indicate that a new (global) Newton step is performed. The first approximation used is the zero grid function.

The numbers shown in Table 5.1 are the $\| \cdot \|_2$ -errors with respect to the discrete solution after each MG cycle. The FAS approach and method II give indeed very similar results in this example. The FAS algorithm is the (technically) simplest of the three algorithms used. As pointed out by Brandt [25], Section 8.3, there are several other advantages of FAS over the indirect methods.

The numerical similarity between method II and the FAS algorithm can, of course, only be expected if the first approximation used is sufficiently close to the solution (so that the convergence of Newton's method is sufficiently good). If we replace L^Ω in (5.12) by

$$L^\Omega u = -\Delta u + \lambda e^u \quad (5.13)$$

it turns out that - for $\lambda \geq 0$ - the dependence of the FAS method on the first approximation is much less sensitive than that of method II. For an example see Table 5.2 where results analogous to those in Table 5.1 are given for $\lambda = 100$. (The corresponding solution and the first approximation are chosen as above.)

MG steps:	1	2	3	4	5	6
method II	0.26(+3)	0.33(+2)	0.38(0)	0.67(-2)	0.11(-3)	0.19(-5)
FAS	0.14(+2)	0.21(0)	0.39(-2)	0.75(-4)	0.17(-5)	0.41(-7)

Table 5.2: Results corresponding to those in Table 5.1 for $\lambda = 100$ (see (5.13))

5.4 A remark on nonlinear relaxation methods

Relaxation methods for linear problems usually have several analogs for nonlinear problems (see [80],[101]). In the MG context, we are mainly interested in the smoothing properties of such nonlinear relaxation methods. We want to discuss this question briefly for a simple nonlinear problem of the type (5.4). For simplicity, we consider only Jacobi's method. A corresponding analysis can, however, also be made for other relaxation methods.

Let, in particular, a nonlinear counterpart of model problem (P) be given, namely

$$L_h u_h := L_h u_h + g(x, u_h) = f_h(x), \quad (x \in \Omega_h) \quad (5.14)$$

where L_h and Ω_h are given as in model problem (P). Using the same notation as in Section 3.2, in particular,

$$\bar{L}_h w_h := L_h w_h - \frac{4}{h^2} w_h,$$

one complete step of the *nonlinear Jacobi w -relaxation* is defined by $\bar{w}_h = w_h + \omega(z_h - w_h)$ and

$$\frac{4}{h^2} z_h(x) + \bar{L}_h w_h(x) + g(x, z_h(x)) = f_h(x) \quad (x \in \Omega_h). \quad (5.15)$$

In practice, one may replace (5.15) by one Newton-step for each single equation (*Jacobi-Newton w -relaxation*):

$$\frac{4}{h^2} z_h(x) + \bar{L}_h w_h(x) + g(x, w_h(x)) + g_u(x, w_h(x))(z_h(x) - w_h(x)) = f_h(x) \quad (x \in \Omega_h). \quad (5.16)$$

An even simpler linearized version of (5.15) which does not use any derivatives at all, is mentioned by Hackbusch [50], Section 7.1. Here (5.15) is simply replaced by

$$\frac{4}{h^2} z_h(x) + \bar{L}_h w_h(x) + g(x, w_h(x)) = f_h(x) \quad (x \in \Omega_h). \quad (5.17)$$

We call this method *Jacobi-Picard w -relaxation*.

The latter relaxation method should, however, be used with care. One difficulty which arises in connection with this method can already be demonstrated by looking at the special case of g being a linear function of u , namely

$$g(x, u) = cu, \text{ with constant } c > 0. \quad (5.18)$$

In this linear case, the relaxation operators of the Jacobi-Newton and the Jacobi-Picard methods are given by, respectively,

$$\begin{aligned} S_h^N &= (1 - \frac{\omega ch^2}{4+ch^2}) I_h - \frac{\omega h^2}{4+ch^2} L_h \quad (\omega = \omega^N), \\ S_h^P &= (1 - \frac{\omega ch^2}{4}) I_h - \frac{\omega h^2}{4} L_h \quad (\omega = \omega^P). \end{aligned} \quad (5.19)$$

Obviously, both operators coincide if

$$\omega^N = \frac{4+ch^2}{4} \omega^P. \quad (5.20)$$

It is therefore sufficient to analyze $S_h^N = S_h^N(\omega)$.

By considerations similar to those in Section 3.2, we obtain the eigenvalues χ_n^N and the smoothing factor μ^N of $S_h^N(\omega)$:

$$\chi_n^N(\omega) = \frac{4}{4+ch^2} \chi_n(\omega) + \frac{(1-\omega)ch^2}{4+ch^2} \quad (\chi_n \text{ in (3.12)}), \quad (5.21)$$

$$\mu^N(h; \omega, c) = \max \{ |1-\omega(1 - \frac{2\cos\pi h}{4+ch^2})|, |1-\omega(1 + \frac{4\cos\pi h}{4+ch^2})| \}. \quad (5.22)$$

From this, we see that, for any fixed $0 < \omega < 1$ (and any fixed h), the smoothing properties of S_h^N become better for increasing c :

$$\mu^N(h; \omega, c) \searrow 1 - \omega \quad (0 \leq c \rightarrow \infty). \quad (5.23)$$

The situation is quite different for the Jacobi-Picard method: From the relation (5.20), we see that for any fixed $\omega = \omega^P$, $0 < \omega < 1$, the Jacobi-Picard method has no smoothing properties (in fact, strong divergence occurs), if ch^2 is sufficiently large.

In our special case (5.18), of course, one could overcome this difficulty by choosing ω^P small enough, as a function of ch^2 . This is, however, no longer possible for the more general case (5.14). Although the above analysis cannot be applied directly to this general case, it is clear that the unfavorable behavior of the Jacobi-Picard method carries over:

The Jacobi-Picard method, with fixed $\omega = \omega P$, cannot be used for smoothing purposes whenever $h^2 g_u(x, u_h(x))$ is large compared to 1 for certain $x \in \Omega_h$. If, however, $0 \leq h^2 g_u(x, u_h(x)) < 1$ ($x \in \Omega_h$) and if w_h is sufficiently close to u_h , the Jacobi-Picard method should give results similar to those of the Jacobi-Newton method.

We finally remark that the application of one Newton step in relaxing each single equation, seems to be a reasonable possibility for rather general nonlinear smoothing methods. This has been confirmed by several numerical experiments.

Remark: As a by-product, in formula (5.22) we have determined the smoothing factor μ of Jacobi ω -relaxation for the Helmholtz equation (with Helmholtz constant $c \geq 0$). For any fixed h and c one can easily determine the optimal relaxation parameter. One obtains:

$$\omega_{\text{opt}} = \frac{4+ch^2}{4+ch^2+\cos\pi h}, \quad \mu_{\text{opt}} = \frac{3\cos\pi h}{4+ch^2+\cos\pi h}.$$

In particular, the optimal relaxation parameter and the corresponding smoothing factor approach 1 and 0, respectively, if c tends to infinity.

5.5 Some additional remarks

In this section, we want to mention only two further complexes which are connected with FAS multigrid methods:

- the application of multigrid methods to bifurcation problems,
- the dual view of multigrid methods.

5.5.1 An exemplary bifurcation problem

In the general description of the FAS multigrid method in Section 5.2, we have considered a discrete nonlinear elliptic operator L_h without making particular assumptions on it. Even in linear cases, but clearly even more in nonlinear cases, there are many situations where the fundamental algorithms have to be modified to take special features of a given problem into account.

For demonstration purposes, we want to mention only one such case here. Consider the following discrete problem for $\lambda > 0$:

$$L_h^{\Omega} u_h = -\Delta_h u_h - \lambda e^u_h = 0 \quad (\Omega_h), \quad u_h = 0 \quad (\partial\Omega_h) \quad (5.24)$$

on the unit square $\Omega = (0,1)^2$ with $h = h_{x_1} = h_{x_2}$. It is well known that the corresponding continuous problem has solutions which behave - as a function of λ - as illustrated in Figure 5.3. Methods of multigrid type have been used by several authors [30], [73], [74] to treat this problem.

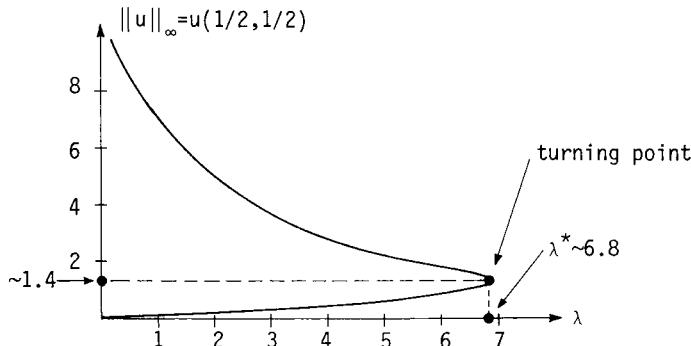


Figure 5.3: "Bifurcation diagram" for $-\Delta u - \lambda e^u = 0$ (Ω), $u=0$ ($\partial\Omega$) showing $u(P)$ (for $P=(1/2,1/2)$) as a function of λ

For fixed $\lambda < \lambda^*$ (with λ not too close to λ^*), any reasonable algorithm of the type described in Section 5.2, e.g. the one mentioned briefly in Section 5.3 (with zero first approximation), will succeed in computing the "lower" solution (cf. Figure 5.3). Clearly, if λ approaches the critical value λ^* , the multigrid method will fail to converge (cf. [30]). Without going into details, we point out that this effect is caused by certain low frequency error components which are not reduced by coarse-grid corrections. This effect is very similar to the one which occurs if Helmholtz' equation is treated by multigrid methods for a fixed Helmholtz constant which is close to an eigenvalue of the discrete Laplacian.

This difficulty can easily be overcome by combining the multigrid approach with a suitable continuation method. In this particular example, one can, for instance, regard λ as an additional variable and prescribe another parameter ($\|u_h\|_2$ or $u_h(P)$ at the center point $P=(1/2,1/2)$ of Ω_h). Thus we consider (5.24) no longer as a boundary value problem with λ fixed, but as a nonlinear eigenvalue problem of the form

$$B_h(u_h, \lambda) = 0 \quad (\Omega_h), \quad u_h = 0 \quad (\partial\Omega_h),$$

together with an additional constraint, e.g., $u_h(P) = \delta$.

The FAS multigrid method (i.e. mainly its smoothing part) has to be modified with respect to this problem. We briefly describe one possibility how this modification can be done. (For other possibilities see, for example, [73].) We assume that the

constraint is given by $u_h(P) = \delta$. One smoothing step is defined such that it takes both the variable λ and the constraint into account, e.g., by the following partial steps:

- (1) Apply one (nonlinear) relaxation step to the actual approximation of u_h (including the value at P).
- (2) Multiply the relaxed approximation by a factor such that the constraint is satisfied afterwards.
- (3) Compute a new value for λ by using the difference equations (e.g., using a suitable average over all equations).

If a reasonable FAS method is constructed which uses this kind of a smoothing process on all grids, one can march along the curve in Figure 5.3, starting somewhere on the lower "branch". Taking the previous solution (and the corresponding value of λ) as a first approximation and enlarging δ , one will come around the turning point without any difficulty. (Corresponding experiments have been carried out in cooperation with A. Brandt.)

However, somewhere beyond the turning point one has to be careful with the use of standard pointwise relaxation methods. Difficulties will arise if the coarsest grid used is too coarse. This is due to the fact that the "diagonal elements" of the corresponding relaxation operator (by which one has to divide) may be close to 0 or change sign on the coarsest grids. This difficulty can be overcome by simple modifications of the relaxation method used. In practice a combination of the (nonlinear) Gauß-Seidel method with Kaczmarz' method (see [21]), for instance, turned out to be suitable.

Already from the 1D-case it is known [11] that a discrete problem of the type considered here may have a very large variety of solutions. In our numerical experiments, we indeed found several solutions. In particular, for large values of δ (roughly $\delta \sim 9$ and $h=1/32$), the constraint used seems to be no longer suitable in order to distinguish between different discrete solutions. What happens in practice (convergence, divergence, jump to another branch of discrete solutions) depends somewhat on the algorithmical details (e.g., on the coarsest grid used). One should try to use more sophisticated continuation methods in this critical field (e.g. the arclength continuation method [30]).

5.5.2 The (h,H) -relative truncation error and the dual view of multigrid methods

The coarse-grid defect equation (5.11) of the FAS scheme can be written in the form

$$L_H w_h^j = I_h^H f_h + \tau_h^H [u_h^j] \quad (5.25)$$

where

$$\tau_h^H [z_h] := L_H I_h^H z_h - I_h^H L_h z_h.$$

Trivially, the following identity holds for the discrete solution u_h :

$$L_H (\hat{I}_h^H u_h) = I_h^H f_h + \tau_h^H [u_h]. \quad (5.26)$$

$\tau_h^H [u_h]$ is called the (h, H) -relative truncation error (with respect to I_h^H , \hat{I}_h^H).

With respect to the grids Ω_h and Ω_H , τ_h^H plays a similar role as the truncation error (local discretization error)

$$\tau_h[u] := L_h I_h^h u - I^h L u \quad (5.27)$$

of the continuous solution u with respect to Ω and Ω_h . (Here I^h denotes the straight injection operator from Ω to Ω_h .) If, in particular, $\Omega_H = \Omega_h$ and \hat{I}_h^H is the straight injection operator, we see from (5.26) that $\tau_h^H [u_h]$ can be interpreted in the following way: τ_h^H is that quantity which has to be added to the right hand side $I_h^H f_h$ to obtain the values of the fine-grid solution u_h (on Ω_H) by solving the coarse-grid equation (5.26).

The quantity τ_h^H (and approximations to it) are the starting point for several more sophisticated multigrid techniques which we only want to list here (for details and further references, see [25]):

- τ -estimation;
- τ -extrapolation;
- frozen- τ techniques, especially for the treatment of parabolic problems [63];
- adaptive multigrid techniques, where the switching from fine to coarse and from coarse to fine is controlled by defects and τ -quantities;
- small storage algorithms.

All these techniques can be regarded from the *dual point of view* - as Brandt calls it. Here not the fine grid Ω_h is regarded as the primary grid, but the coarse grid Ω_H . The objective is to obtain - by treating coarse-grid equations - a solution which has the fine-grid accuracy. For this purpose, of course, a suitable approximation for $\tau_h^H [u_h]$ has to be provided.

6. The full multigrid method (nested iteration)

So far, we have considered only iterative multigrid methods for the solution of discrete elliptic boundary value problems. Although suitable versions of these methods are very efficient, their efficiency can still be improved essentially if they are used in form of the *full multigrid* (FMG) technique [25],[19], also called *nested iteration* [50],[49]. Although this approach can be applied to linear as well as to nonlinear problems (in connection with FAS), we restrict ourselves to considering the linear case.

6.1 Idea and purpose

The full multigrid approach can be regarded from different points of view. We here prefer a quite narrow interpretation of this method, namely as an "approximate direct solver" (for a given h -discrete elliptic problem (2.1)) which is characterized by the following features:

- (1) An approximation \tilde{u}_h of the discrete solution u_h is computed up to an error $\|\tilde{u}_h - u_h\|$ which is smaller than the discretization error $\|u - u_h\|$.
- (2) The number of arithmetic operations needed is proportional to the number of grid-points of Ω_h (with only a small constant of proportionality).

We would like to point out that the idea of nested iteration is a well-known numerical principle which has also been used successfully independently of the multigrid concept. Within an arbitrary iterative process (e.g. SOR) for the solution of a given discrete problem, this principle simply means that lower (coarser) discretization levels are used in order to provide good initial approximations for the iteration on the next higher (finer) discretization level [64],[65]. With classical methods like SOR, however, it is not possible to achieve both, (1) and (2).

6.2 Structure of the full multigrid method

The FMG method, as we understand it here, is most easily explained by a flow chart. We refer to a general elliptic boundary value problem

$$L^\Omega u = f^\Omega \quad (\Omega), \quad L^\Gamma u = f^\Gamma \quad (\Gamma := \partial\Omega) \quad (6.1)$$

on a bounded domain Ω . As in Section 4.2., let

$$L_\ell u_\ell = f_\ell \quad (\Omega_\ell) \quad (\ell = 0, 1, 2, \dots) \quad (6.2)$$

be a sequence of discrete approximations to (6.1). For all $\ell \geq 1$, we denote by

$$\text{MGI}^r(\cdot, \ell, L_\ell, f_\ell) : \mathcal{G}(\Omega_\ell) \rightarrow \mathcal{G}(\Omega_\ell) \quad (6.3)$$

a procedure consisting of r iteration steps of a suitable iterative multigrid method for (6.2) (using grids $\Omega_0, \dots, \Omega_\ell$; cf. Figure 4.2). In the following, standard values for r are $r=1$ and $r=2$ (fixed). In general, however, r may depend on ℓ .

Now let an arbitrary but fixed ℓ be chosen with $h=h_\ell$. Then the full multigrid method proceeds as shown in Figure 6.1. The final FMG approximation is denoted by \tilde{u}_ℓ .

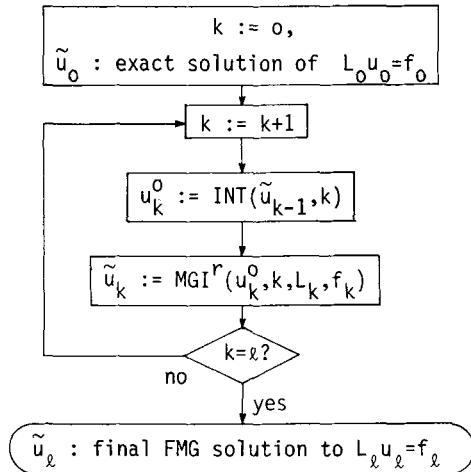


Figure 6.1: FMG method for the approximate solution of $L_\ell u_\ell = f_\ell$ (Ω_ℓ)

In Figure 6.1, the notation

$$\text{INT}(\cdot, k) : \mathcal{G}(\Omega_{k-1}) \rightarrow \mathcal{G}(\Omega_k) \quad (6.4)$$

is used for a suitable interpolation procedure. In general, we assume INT to be an affine operator of the form

$$\text{INT}(u_{k-1}, k) = \Pi_{k-1}^k u_{k-1} + w_k \quad (6.5)$$

with a linear operator

$$\Pi_{k-1}^k : \mathcal{G}(\Omega_{k-1}) \rightarrow \mathcal{G}(\Omega_k)$$

and a certain fixed gridfunction $w_k \in \mathcal{G}(\Omega_k)$. This representation of INT allows one to take the given boundary conditions into account within the interpolation process or to make use of L^Ω and of f^Ω as well ("interpolation using the grid equation" as mentioned already in Section 2.4; also see [36]). In order to distin-

guish the interpolation (6.5) from the interpolation used within the MG iterations, we call it *FMG interpolation*.

6.3 A simple theoretical result

Under reasonable assumptions, the FMG method asymptotically (i.e. for sufficiently large ℓ) has indeed the properties (1) and (2) which were pointed out in Section 6.1. This can be shown by very simple estimations. Several such estimates have been given, which differ only slightly (see, e.g., [50] and reference "[11]" in [36]). For completeness, we give an estimate, too.

By $M_\ell : \mathcal{E}(\Omega_\ell) \rightarrow \mathcal{E}(\Omega_\ell)$ we denote the iteration operator corresponding to (6.3). Let any reasonable vector norm $\| \cdot \|$ with corresponding operator norm be given on $\mathcal{E}(\Omega_\ell)$ ($\ell = 0, 1, 2, \dots$). For simplicity, we restrict ourselves to square grids in this section. (Otherwise, in the following inequalities, h_ℓ would have to be replaced by its maximal component.) Now we make the following assumptions.

(1) Let the norms of M_ℓ and $\Pi_{\ell-1}^\ell$ be uniformly bounded:

$$\| M_\ell \| \leq \eta < 1, \quad \| \Pi_{\ell-1}^\ell \| \leq C \quad (\ell = 1, 2, \dots). \quad (6.6)$$

(2) The discretization error and the FMG interpolation error are assumed to be of order κ_1 and κ_2 , respectively:

$$\| u - u_\ell \| \leq K_1 h_\ell^{\kappa_1}, \quad \| u - \Pi_{\ell-1}^\ell u - w_\ell \| \leq K_2 h_\ell^{\kappa_2} \quad (6.7)$$

with $\kappa_1, \kappa_2, K_1, K_2$ independent of ℓ .

(3) We assume a fixed meshsize ratio $\zeta > 1$: $h_{\ell-1} = \zeta h_\ell$ ($\ell = 1, 2, \dots$).

Under these assumptions we obtain the following

Lemma 6.1: Let $\kappa_2 \geq \kappa_1$ and $\eta^r A < 1$ with $A := C\zeta^{-1}$. Then the following estimate holds for all $\ell \geq 1$:

$$\| \tilde{u}_\ell - u_\ell \| \leq \delta^* h_\ell^{\kappa_1} \quad \text{with} \quad \delta^* := \frac{\eta^r K}{1-\eta^r A} + o(1) \quad (\ell \rightarrow \infty) \quad (6.8)$$

and

$$K := \begin{cases} K_1(1+A) + K_2 & (\text{if } \kappa_2 = \kappa_1) \\ K_1(1+A) & (\text{if } \kappa_2 > \kappa_1). \end{cases} \quad (6.9)$$

If we additionally have a lower bound for the discretization error

$$\|u - u_\ell\| \geq \hat{K}_1 h_\ell^{\kappa_1} \quad (\hat{K}_1 > 0, \text{ independent of } \ell),$$

then we obtain by (6.8)

$$\|\tilde{u}_\ell - u_\ell\| \leq \beta^* \|u - u_\ell\| \quad \text{with} \quad \beta^* := \frac{K}{\hat{K}_1} \frac{n^r}{1-n^r A} + o(1) \quad (\ell \rightarrow \infty). \quad (6.10)$$

Proof: By definition of the FMG method, we have for all $\ell \geq 1$

$$\tilde{u}_\ell - u_\ell = M_\ell^r (u_\ell^0 - u_\ell), \quad u_\ell^0 = \Pi_{\ell-1}^\ell \tilde{u}_{\ell-1} + w_\ell.$$

Using the identity

$$u_\ell^0 - u_\ell = \Pi_{\ell-1}^\ell (\tilde{u}_{\ell-1} - u_{\ell-1}) + \Pi_{\ell-1}^\ell (u_{\ell-1} - u) + (\Pi_{\ell-1}^\ell u + w_\ell - u) + (u - u_\ell),$$

we get the recursive estimation

$$\delta_\ell \leq n^r (A \delta_{\ell-1} + K_1(1+A) + K_2 h_\ell^{\kappa_2 - \kappa_1}), \quad \delta_\ell := \|\tilde{u}_\ell - u_\ell\| / h_\ell^{\kappa_1}.$$

From this, (6.8) follows by a simple calculation.

Remarks:

- (1) In the case $\kappa_2 = \kappa_1$, one sees from the definition of K in (6.9) that the contribution of the FMG interpolation constant K_2 to the bounds δ^* and β^* may become arbitrarily large (depending on u). In particular, in (6.10), K_1/\hat{K}_1 often can be assumed to be bounded independently of u (see below), not, however, K_2/\hat{K}_1 . If, on the other hand, $\kappa_2 > \kappa_1$, then the contribution of K_2 asymptotically vanishes. Thus, it might be advantageous to choose an FMG interpolation of an order which is higher than that of the discretization.

- (2) If $\kappa_2 > \kappa_1$ and we additionally assume an expansion

$$u_\ell = u + h_\ell^{\kappa_1} e + o(h_\ell^{\kappa_1}), \quad (6.11)$$

then we can replace K/\hat{K}_1 in (6.10) by $A+1$. Moreover, under certain stronger assumptions (see Hackbusch [50], Chapter 5, where the FMG interpolation itself is assumed to be described by a linear operator), one can replace $A+1$ by ζ^{κ_1-1} . (Notice that the assumption (5.2) in [50] does not distinguish between κ_1 and κ_2 .)

6.4 Computational work, some practical remarks

The computational work W_ℓ^{FMG} needed for the FMG method (as described in Figure 6.1) can easily be estimated. By arguments similar to those in Section 4.4, one immediately obtains in terms of W_ℓ (cf. (4.22)), for example in case of standard coarsening ($\zeta=2$):

$$W_\ell^{\text{FMG}} \leq \frac{4}{3} r W_\ell + \frac{4}{3} W_{\ell-1}^{\text{INT}} \quad (6.12)$$

(neglecting lower order terms). Here W_{k-1}^{INT} denotes the work needed for the FMG interpolation process from grid Ω_{k-1} to grid Ω_k ($k=1, \dots, \ell$).

Whether (1) in Section 6.1 is satisfied (for sufficiently large ℓ and fixed r , usually $r=1$ or $r=2$), clearly depends essentially on the size of n . Let us assume in the following that $\zeta=2$, $\kappa_2 > \kappa_1 = 2$ and that (6.11) is satisfied. According to Remark (2) above, β^* in (6.10) can then be replaced by

$$\beta^* = \eta^r \frac{1+4C}{1-4Cn^r} \quad (6.13)$$

which now depends on C , η and r only. Here $C \geq 1$, but for reasonable FMG interpolation processes, C usually can be assumed not to be very large.

Example: Let us consider model problem (P). An appropriate and especially cheap FMG interpolation is described in the following. This interpolation is an example of an interpolation "using the grid equation". (It is used in the sample program listed in the appendix.) The process $u_k^0 = \text{INT}(\tilde{u}_{k-1}, k)$ (cf. Figure 6.1) proceeds in three partial steps (1), (2), (3):

(1) At points $x \in \Omega_k \cap \Omega_{k-1}$ define $u_k^0(x) := \tilde{u}_{k-1}(x)$.

(2) At points $x \in \Omega_k \setminus \Omega_{k-1}$, $x = \kappa \cdot h$ with $\kappa_1 + \kappa_2$ even, we define $u_k^0(x)$ by

$$\frac{1}{2h_k^2} \begin{bmatrix} -1 & -1 \\ & 4 \\ -1 & -1 \end{bmatrix} h_k u_k^0(x) = f^\Omega(x). \quad (6.14)$$

Note that this is an explicit equation (corresponding to a diagonal matrix). At boundary points, $u_k^0(x)$ is assumed to be replaced by the corresponding boundary values given.

(3) At points $x \in \Omega_k$, $x = \kappa \cdot h$ with $\kappa_1 + \kappa_2$ odd, we define $u_k^0(x)$ by

$$\frac{1}{h_k^2} \begin{bmatrix} -1 & & \\ -1 & 4 & -1 \\ & -1 & \end{bmatrix} h_k u_k^0(x) = f^\Omega(x). \quad (6.15)$$

Again, this is an explicit equation. (We remark that this partial step can be interpreted as one half-step of RB relaxation (see Section 7.3)).

For this FMG interpolation one has $C=1$ (with respect to $\|\cdot\|_2$) and $\kappa_2=4$. Table 6.1 shows the dependence of β^* on n in this case. We see that, for $r=1$, $n \leq 0.1$ is sufficient to guarantee a value of $\beta^* < 1$ (and by this also (1) in Section 6.1). n -values of this size are typical for efficient multigrid methods for standard applications. This is confirmed, e.g., by the numerical results of Chapter 10. For the underlying example of model problem (P), one can construct multigrid methods with $n \leq 0.1$ by using the theoretical results of Chapter 8 together with the considerations of Section 4.3 (see, for example, Corollary 8.4).

$r \backslash n$	0.5	0.25	0.20	0.10	0.05	0.01
1	-	∞	>1	0.833	0.313	0.052
2	∞	0.417	0.238	0.052	0.013	0.001

Table 6.1: Values of β^* (6.13) as a function of n if $C=1$

Remark: Instead of the FMG interpolation described above, one could also use, e.g., cubic interpolation. This is, however, somewhat more expensive. (Cubic interpolation is used within the program MG01 described in Chapter 10.)

Clearly, the efficiency of an FMG method depends on the numerical work needed to ensure (1) in Section 6.1. Some concrete examples on computational work and accuracy will be given in Chapter 10. We here only want to make the following point clear: The efficiency of an FMG process is not solely connected to the efficiency of the (iterative) multigrid process which it is based upon. It is, for example, not worthwhile to construct an FMG method using a multigrid cycle which is characterized by a convergence factor n , much smaller than needed to achieve the discretization accuracy. Such a multigrid method will usually be too expensive per cycle.

Finally, we want to point out that the FMG concept is only the starting point for several more sophisticated techniques (see Part II in Brandt [25] for some details and references). The common idea of these more advanced approaches is the orientation to the continuous solution u rather than to the discrete solution u_h .

7. The concept of model problem analysis, smoothing and two-grid convergence factors

So far, we have described the fundamental multigrid methods in a rather general form. The practically very important question of how the individual multigrid components should be chosen for concrete situations has, however, not yet been discussed systematically. As pointed out in the introduction, there are two related approaches in treating this question: the *model problem* and the *local Fourier analysis*.

A first example of model problem analysis has been already considered in Chapter 3. The main characteristics of this example which carry over to the general concept of both, the model problem and local Fourier analysis are the following:

- For the given linear difference operator L_h a basis of discrete eigenfunctions is known.
- The $(h,2h)$ two-grid method is constructed in such a way, that the corresponding iteration operator M_h^{2h} can be represented with respect to the L_h -eigenbasis by a block-matrix consisting of small blocks only. (In the particular example at most $(4,4)$ -blocks occurred.)

Because of these properties, the determination of the corresponding two-grid convergence factors turned out to be relatively easy. Furthermore, we were able to give a straightforward definition of a smoothing factor by making use of the fact that the eigenfunctions of L_h and the smoothing operator S_h were the same.

In this chapter, the analysis of Chapter 3 is generalized in several respects. In Section 7.1, we describe the class of problems under consideration. Section 7.2 refers to the Fourier representation of coarse-grid correction operators. Here other coarse grids than Ω_{2h} are admitted. In Section 7.3, we give Fourier representations of efficient smoothing methods like RB and ZEBRA relaxation. A general definition of smoothing factors is given in Section 7.5. Some other classes of problems which can be treated similarly are sketched in Section 7.6.

Concrete results of the model problem analysis are presented in Chapter 8. In Chapter 9, the local Fourier analysis and its relation to the model problem analysis are described. Furthermore some other theoretical approaches are outlined.

7.1 Assumptions on the difference operator

Let

$$L_h u_h = f_h \quad (\Omega_h) \tag{7.1}$$

be a given grid equation on a rectangular grid $\Omega_h = \Omega \cap G_h$

$$\Omega = (0, A_1) \times (0, A_2), \quad G_h = \{x = \kappa \cdot h : \kappa \in \mathbb{Z}^2\}; \quad h_{x_j} = A_j / N_j, \quad N_j \text{ even } (j=1,2), \quad (7.2)$$

$h = (h_{x_1}, h_{x_2})$, $\kappa \cdot h := (\kappa_1 h_{x_1}, \kappa_2 h_{x_2})$. We consider invertible operators L_h for which the following grid functions form a basis of eigenfunctions:

$$\varphi_n(x) := 2 \sin(n_1 \pi x_1 / A_1) \sin(n_2 \pi x_2 / A_2) \quad (x \in \Omega_h; 0 < n < N) \quad (7.3)$$

with $n = (n_1, n_2)$, $N = (N_1, N_2)$. (The inequalities are to be understood componentwise.) These φ_n are orthonormal with respect to (1.7). Examples of operators L_h with this property are given below.

For the formalism used in the following, it is convenient to write the grid functions (7.3) in the form

$$\varphi(\theta, x) := 2 \sin(\theta_1 x_1 / h_{x_1}) \sin(\theta_2 x_2 / h_{x_2}) \quad (x \in \Omega_h; \theta \in T_h) \quad (7.4)$$

with

$$T_h = \{\theta = (\frac{n_1}{N_1} \pi, \frac{n_2}{N_2} \pi) : 0 < n < N\} \subset (0, \pi)^2. \quad (7.5)$$

(For technical reasons, we sometimes consider $\varphi(\theta, x)$ in (7.4) also for certain $\theta \in (-\pi, \pi]^2$. Whenever such a θ occurs in the following - see (7.22), for example - then there exists a $\tilde{\theta} \in T_h$ such that $\varphi(\theta, x) = \pm \varphi(\tilde{\theta}, x)$ ($x \in \Omega_h$). So this only means that some of the above basis functions are used with a negative sign.)

The following continuous problems typically lead to operators L_h of the above type:

Example 7.1: Let a linear boundary value problem (1.1) be given on Ω with

$$L^\Omega u = -a_1 u_{x_1 x_1} - a_2 u_{x_2 x_2} + cu \quad (a_1, a_2 > 0, c \in \mathbb{R}) \quad (7.6)$$

and Dirichlet boundary conditions

$$u = f^\Gamma(x) \quad (x \in \Gamma). \quad (7.7)$$

We consider discretizations of L^Ω on Ω_h which can be described by a symmetric compact 9-point star, i.e.

$$L_h^\Omega \triangleq \begin{bmatrix} s_{-1,1} & s_{0,1} & s_{1,1} \\ s_{-1,0} & s_{0,0} & s_{1,0} \\ s_{-1,-1} & s_{0,-1} & s_{1,-1} \end{bmatrix}_h \quad \text{with} \quad \begin{cases} s_{1,1} = s_{-1,1} = s_{1,-1} = s_{-1,-1}, \\ s_{1,0} = s_{-1,0}, s_{0,1} = s_{0,-1}. \end{cases} \quad (7.8)$$

(Clearly, these coefficients depend on h .) Every discretization of this type leads - after elimination of the boundary conditions - to a discrete system (7.1). The grid functions (7.4) are eigenfunctions with corresponding eigenvalues

$$\lambda(\theta, h) = \sum_{|\kappa| \leq 1} s_\kappa \cos(\theta \kappa) \quad (\theta \in T_h), \quad (\kappa = (\kappa_1, \kappa_2), \quad |\kappa| = \max(|\kappa_1|, |\kappa_2|)). \quad (7.9)$$

In particular, higher-order discretizations of the "Mehrstellen"-type ("Hermitian methods") [31] are admitted here.

Example 7.2: Let a fourth-order boundary value problem (1.1) be given on Ω with

$$L^\Omega u = \Delta \Delta u - b \Delta u + c u \quad (b, c \in \mathbb{R}) \quad (7.10)$$

and boundary conditions

$$u = f_1^\Gamma(x), \quad -\Delta u = f_2^\Gamma(x) \quad (x \in \Gamma). \quad (7.11)$$

Approximating all occurring Δ -operators by symmetric 5- or compact 9-point difference stars and eliminating the boundary conditions, again leads to a discrete operator L_h which satisfies the above assumptions. (L_h^Ω is then described by a 13- or 25-point star, respectively.)

Although for the following considerations we mainly need the assumption (7.3), for the derivation of concrete results we shall additionally assume that L_h is given by a compact 9-point star (7.8). This is done, in order to avoid some technical complications which are introduced by larger stars (e.g. within the relaxation process), see also Section 7.6.

7.2 The (h, H) coarse-grid correction operator

For the definition of the coarse-grid correction (CGC) operator

$$K_h^H = I_h - I_H^h L_H^{-1} I_h^H L_h \quad (7.12)$$

we first need a coarser grid Ω_H with $\mathcal{N}_H \ll \mathcal{N}_h$. In the following, we consider four choices of Ω_H (cf. Section 2.4.1). In all these cases we have $\Omega_H = \Omega \cap G_H$. For the cases (1),(2) and (4), G_H is defined analogously to G_h in (7.2) with different meshsizes only. For the case (3), G_H is explicitly defined below.

- (1) Standard coarsening: $H=2h$.
- (2) Semi-coarsening: x_1 -coarsening, $H=(2h_{x_1}, h_{x_2})$ or x_2 -coarsening, $H=(h_{x_1}, 2h_{x_2})$.
- (3) Red-black coarsening: For simplicity, we consider this type of coarsening only

for square grids Ω_h ($h = h_{x_1} = h_{x_2}$). Then $\Omega_H = \Omega \cap G_H$, where G_H can be identified with a rotated grid of meshsize $H = \sqrt{2}h$:

$$G_H = \{x = \kappa \cdot h : \kappa \in \mathbb{Z}^2, \kappa_1 + \kappa_2 \text{ even}\}.$$

(4) Quadrupling h: $H = 4h$. In this case we assume that N_j is a multiple of 4.

In all these cases there is a natural distinction between *low* and *high* (h -)frequencies. As before, the low frequencies are those which can be represented (are visible) on the coarser grid Ω_H also. We will define the low and high frequencies for each of the above cases separately. Corresponding to this distinction, the space $\mathbf{G}(\Omega_h)$ is written as an orthogonal sum

$$\mathbf{G}(\Omega_h) = \mathbf{G}^{\text{low}}(\Omega_h) \perp \mathbf{G}^{\text{high}}(\Omega_h). \quad (7.13)$$

This splitting of $\mathbf{G}(\Omega_h)$ can simply be described by a corresponding splitting of the index set T_h in (7.5)

$$T_h = T_h^{\text{low}} \cup T_h^{\text{high}}, \quad \#T_h^{\text{low}} = N_H \quad (7.14)$$

(Here $\#T_h^{\text{low}}$ denotes the number of elements in T_h^{low} .) In all cases considered, there is a natural identification of $\mathbf{G}^{\text{low}}(\Omega_h)$ and $\mathbf{G}(\Omega_H)$: If we denote by ϕ the restrictions of low frequencies φ to Ω_H

$$\phi(\theta, x) := \varphi(\theta, x) \quad (x \in \Omega_H; \theta \in T_h^{\text{low}}) \quad (7.15)$$

then these ϕ form a basis of $\mathbf{G}(\Omega_H)$:

$$\mathbf{G}(\Omega_H) = \text{span } \{\phi(\theta, \cdot) : \theta \in T_h^{\text{low}}\}. \quad (7.16)$$

The high frequencies have the property that for all $\theta' \in T_h^{\text{high}}$

$$\text{either } \varphi(\theta', x) = \pm \phi(\theta, x) \quad (x \in \Omega_H) \quad \text{for a suitable } \theta \in T_h^{\text{low}}, \quad (7.17)$$

$$\text{or } \varphi(\theta', x) \equiv 0 \quad (x \in \Omega_H). \quad (7.18)$$

As in Section 3.3, we will introduce low-dimensional subspaces $E_{h,\theta}^H$ of $\mathbf{G}(\Omega_h)$. Each of these spaces is defined as the span of functions (7.4) which coincide on Ω_H (up to sign). The $E_{h,\theta}^H$ are called spaces of H -harmonics. Because of (7.17), (7.18), their basis functions coincide on Ω_H either with $\pm \phi(\theta, x)$ (for a certain $\theta \in T_h^{\text{low}}$) or with 0. For each of the coarser grids Ω_H considered, the corresponding spaces $E_{h,\theta}^H$ of H -harmonics yield a splitting of $\mathbf{G}(\Omega_h)$ into an orthogonal sum

$$\mathbf{G}(\Omega_h) = \bigoplus_{\theta \in T_h^H} E_{h,\theta}^H \quad (7.19)$$

Here the index set $T_h^H \subset T_h$ is a superset of T_h^{low} taking the degenerate cases (7.18) into account (see the examples below).

In the following, we will specify the above quantities for the different coarsenings (1)-(4). We use the notation $|\theta| := \max(|\theta_1|, |\theta_2|)$. Figure 7.1 illustrates the respective θ -ranges of the low frequencies (hatched part). Furthermore, exemplary θ -values of frequencies (7.4) are shown there which coincide on Ω_H with a low frequency (up to sign). This low frequency is marked by "•" and the corresponding high frequencies are marked by "o".

- case (1), standard coarsening, $H=2h$:

$$T_h^{\text{low}} = \{\theta \in T_h : |\theta| < \pi/2\}; \quad T_h^H = \{\theta \in T_h : |\theta| \leq \pi/2\}; \quad (7.20)$$

$$E_{h,\theta}^H = \text{span}\{\varphi(\theta^\alpha, x) : \alpha = (0,0); (1,1); (1,0); (0,1)\}, \quad \theta \in T_h^H \quad (7.21)$$

with

$$\theta^\alpha = \theta - \pi\alpha, \quad \alpha = (\alpha_1, \alpha_2). \quad (7.22)$$

The spaces $E_{h,\theta}^H$ are of dimension 4 if $\theta \in T_h^{\text{low}}$. Otherwise they are of dimension 2 or 1.

- case (2), x_1 -coarsening, $H=(2h_{x_1}, h_{x_2})$ (analogous for x_2 -coarsening):

$$T_h^{\text{low}} = \{\theta \in T_h : \theta_1 < \pi/2\}; \quad T_h^H = \{\theta \in T_h : \theta_1 \leq \pi/2\}; \quad (7.23)$$

$$E_{h,\theta}^H = \text{span}\{\varphi(\theta^\alpha, x) : \alpha = (0,0); (1,0)\}, \quad \theta \in T_h^H \quad (7.24)$$

with θ^α as defined in (7.22). The spaces $E_{h,\theta}^H$ are of dimension 2 and 1 for $\theta \in T_h^{\text{low}}$ and $\theta \notin T_h^{\text{low}}$, respectively.

- case (3), red-black coarsening, " $H=\sqrt{2}h$ " ($h=h_{x_1}=h_{x_2}$):

$$T_h^{\text{low}} = \{\theta \in T_h : \theta_2 \leq \pi - \theta_1 \text{ (if } \theta_1 \leq \pi/2\text{)}, \theta_2 < \pi - \theta_1 \text{ (if } \theta_1 > \pi/2\text{)}\} = T_h^H; \quad (7.25)$$

$$E_{h,\theta}^H = \text{span}\{\varphi(\theta^\alpha, x) : \alpha = (0,0); (1,1)\}, \quad \theta \in T_h^H \quad (7.26)$$

with θ^α as defined in (7.22). The spaces $E_{h,\theta}^H$ are of dimension 2 if $\theta \neq (\pi/2, \pi/2)$. For $\theta = (\pi/2, \pi/2)$, $E_{h,\theta}^H$ is of dimension 1.

Remark: The above definition of low frequencies is in accordance with the natural interpretation of them being representable on the coarse grid Ω_H . This can easily be seen by use of a transformation to the rotated coordinates

$$\xi_1 = (x_1 + x_2)/\sqrt{2}, \quad \xi_2 = (x_1 - x_2)/\sqrt{2}.$$

- case (4), quadrupling h, H=4h:

$$T_h^{\text{low}} = \{\theta \in T_h : |\theta| < \pi/4\}; \quad T_h^H = \{\theta \in T_h : |\theta| \leq \pi/4\}; \quad (7.27)$$

$$E_{h,\theta}^H = \text{span } \{\varphi(\theta^\alpha, x) : \alpha_1, \alpha_2 \in \{0, -1/2, 1/2, 1\}\}, \quad \theta \in T_h^H \quad (7.28)$$

with θ^α as defined in (7.22). The spaces $E_{h,\theta}^H$ are of dimension 16 if $\theta \in T_h^{\text{low}}$. Otherwise they are of dimension 8 or 4.

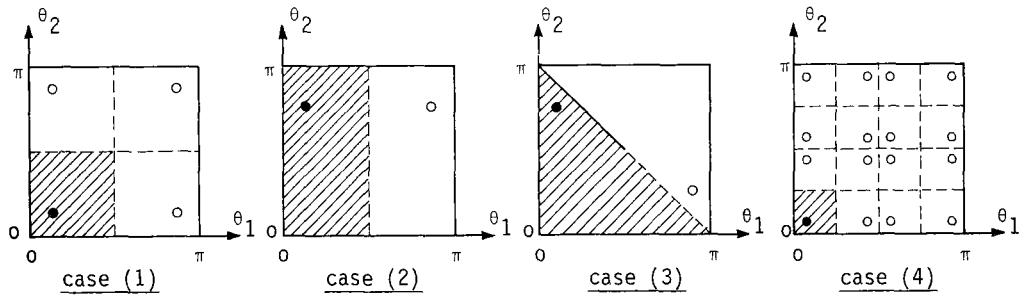


Figure 7.1: θ -range of low frequencies for standard coarsening, x_1 -coarsening, red-black coarsening and quadrupling, respectively

In the model problem analysis we consider (h,H) two-grid methods such that the corresponding coarse-grid correction operator (7.12) leaves the spaces of H -harmonics invariant:

$$K_h^H : E_{h,\theta}^H \rightarrow E_{h,\theta}^H \quad (\theta \in T_h^H). \quad (7.29)$$

For this, we first assume that L_H^{-1} exists. Furthermore, the individual components I_h^H , L_h and I_H^h of K_h^H are supposed to have the following properties

$$\left. \begin{array}{ll} I_h^H : & E_{h,\theta}^H \rightarrow \text{span } \{\Phi(\theta, x)\} \quad (\theta \in T_h^H), \\ L_H : & \text{span } \{\Phi(\theta, x)\} \rightarrow \text{span } \{\Phi(\theta, x)\} \quad (\theta \in T_h^{\text{low}}), \\ I_H^h : & \text{span } \{\Phi(\theta, x)\} \rightarrow E_{h,\theta}^H \quad (\theta \in T_h^{\text{low}}). \end{array} \right\} \quad (7.30)$$

Here we use the formal notation

$$\Phi(\theta, x) := 0 \quad (x \in \Omega_H) \quad \text{for } \theta \in T_h^H \setminus T_h^{\text{low}}.$$

Obviously, the assumptions (7.30) imply (7.29). By this, K_h^H is orthogonally equivalent to a block matrix. The corresponding blocks - represented with respect to the

basis of $E_{h,\theta}^H$ given above - will be denoted by $\hat{K}_{h,\theta}^H$:

$$\hat{K}_{h,\theta}^H \triangleq K_h^H \Big|_{E_{h,\theta}^H} \quad (\theta \in T_h^H). \quad (7.31)$$

The dimensions of these blocks clearly are equal to the dimensions of the respective spaces $E_{h,\theta}^H$. We remark that $\hat{K}_{h,\theta}^H$ is the identity matrix if $\theta \in T_h^H \setminus T_h^{\text{low}}$.

As an example, we give the representation of K_h^H for the case of standard coarsening and 9-point operators:

Lemma 7.1: Let L_h and L_{2h} be given by symmetric compact 9-point stars with coefficients s_κ (cf. (7.8)) and \hat{s}_κ , respectively. We assume L_{2h}^{-1} to exist on $\mathbb{G}(\Omega_{2h})$. Furthermore, I_h^{2h} and I_{2h}^h are supposed to be defined by symmetric 9-point stars

$$I_h^{2h} \triangleq \begin{bmatrix} \hat{t}_{-1,1} & \hat{t}_{0,1} & \hat{t}_{1,1} \\ \hat{t}_{-1,0} & \hat{t}_{0,0} & \hat{t}_{1,0} \\ \hat{t}_{-1,-1} & \hat{t}_{0,-1} & \hat{t}_{1,-1} \end{bmatrix}_h^{2h}, \quad I_{2h}^h \triangleq \begin{bmatrix} \hat{t}_{-1,1} & \hat{t}_{0,1} & \hat{t}_{1,1} \\ \hat{t}_{-1,0} & \hat{t}_{0,0} & \hat{t}_{1,0} \\ \hat{t}_{-1,-1} & \hat{t}_{0,-1} & \hat{t}_{1,-1} \end{bmatrix}_{2h}^h \quad (7.32)$$

with $\hat{t}_\kappa = \hat{t}_{\kappa'}$, $\hat{t}_\kappa = \hat{t}_{\kappa'}$, if $|\kappa_1| = |\kappa'_1|$, $|\kappa_2| = |\kappa'_2|$.

Then the properties (7.29), (7.30) are valid. In particular, we have for all $\theta \in T_h^{\text{low}}$ and all α as given in (7.21) (and θ^α as in (7.22)):

$$\left. \begin{aligned} L_h \varphi(\theta^\alpha, x) &= \lambda(\theta^\alpha, h) \varphi(\theta^\alpha, x), & L_{2h} \Phi(\theta, x) &= \Lambda(\theta, h) \Phi(\theta, x), \\ I_h^{2h} \varphi(\theta^\alpha, x) &= q(\theta^\alpha) \varphi(\theta, x), & I_{2h}^h \Phi(\theta, x) &= \sum_\alpha p_\alpha(\theta) \varphi(\theta^\alpha, x) \end{aligned} \right\} \quad (7.33)$$

with

$$\left. \begin{aligned} \lambda(\theta^\alpha, h) &= \sum_{|\kappa| \leq 1} s_\kappa \cos(\theta^\alpha \kappa), & \Lambda(\theta, h) &= \sum_{|\kappa| \leq 1} \hat{s}_\kappa \cos(2\theta\kappa) \\ q(\theta^\alpha) &= \sum_{|\kappa| \leq 1} \hat{t}_\kappa \cos(\theta^\alpha \kappa), & p_\alpha(\theta) &= \frac{1}{4} \sum_{|\kappa| \leq 1} \hat{t}_\kappa \cos(\theta^\alpha \kappa). \end{aligned} \right\} \quad (7.34)$$

From this, we obtain the following matrix representation for K_h^{2h} with respect to $E_{h,\theta}^{2h}$ ($\theta \in T_h^{\text{low}}$):

$$\hat{K}_{h,\theta}^{2h} = I - \frac{1}{\Lambda} \begin{pmatrix} B_i C_j \end{pmatrix}_{4,4} \quad (7.35)$$

with $\Lambda = \Lambda(\theta, h)$ and

$$\begin{aligned} B_1 &= p_{00}(\theta), & C_1 &= q(\theta^{00})\lambda(\theta^{00}, h), \\ B_2 &= p_{11}(\theta), & C_2 &= q(\theta^{11})\lambda(\theta^{11}, h), \\ B_3 &= p_{10}(\theta), & C_3 &= q(\theta^{10})\lambda(\theta^{10}, h), \\ B_4 &= p_{01}(\theta), & C_4 &= q(\theta^{01})\lambda(\theta^{01}, h). \end{aligned}$$

The statements of the Lemma can be verified easily. Corresponding results can be derived for the other types of coarsening considered: In all these cases, the essential assumption needed is that the respective operators can be described by symmetric difference stars.

Example 7.3: If L_h , L_{2h} , I_h^{2h} and I_{2h}^h are defined as in the sample method in Section 3.1, we obtain for $\theta \in T_h^{\text{low}}$ and all α :

$$\lambda(\theta^\alpha, h) = 2(2 - \cos\theta_1^\alpha - \cos\theta_2^\alpha)/h^2, \quad \Lambda(\theta, h) = \lambda(2\theta, 2h),$$

$$q(\theta^\alpha) = p_\alpha(\theta) = (1 + \cos\theta_1^\alpha)(1 + \cos\theta_2^\alpha)/4.$$

Remark: For the above representation of coarse-grid operators K_h^H we have made essential use of the space of H-harmonics. These are the minimal invariant spaces of K_h^H occurring because of the natural assumptions (7.30). Often, however, it is convenient to consider matrix representations of K_h^H with respect to different (larger) invariant spaces. For instance, operators K_h^H corresponding to red-black or semi-coarsening may - under the assumption (7.29) - be represented with respect to the spaces of 2h-harmonics as well. (This has technical advantages if K_h^H is combined with one of the smoothing operators considered in the next section.) Adding a '^' to any of the occurring grid operators always denotes their representation with respect to suitable spaces of harmonics (more precisely: with respect to the respective basis, see (7.21), (7.24), (7.26) and (7.28)). From the context, it will be clear which representations are actually used.

7.3 Smoothing operators

If a suitable representation of K_h^H by block matrices is given (cf. the remark above), one can obtain corresponding representations also for two-grid operators M_h^H if the smoothing operator used can be decomposed in a similar way. In this section, we give Fourier representations for some important smoothing methods, namely for *RB* and *ZEBRA relaxation*. From the latter a corresponding representation for *alternating ZEBRA relaxation* can be derived easily. Other important smoothing methods

like Gauß-Seidel relaxation with lexicographic ordering of the grid points or ILU-smoothing do not have comparable invariance properties. We will treat them in connection with local Fourier analysis in Chapter 9.

Before we give the Fourier representations mentioned above, we make a few comments on RB and ZEBRA relaxation. In order to avoid some technical complications, we restrict our considerations to 9-point difference operators (7.8).

One complete step of RB relaxation (on Ω_h) consists of two half-steps. For this, the grid points of Ω_h are divided - in a checkerboard manner - into two sets of points, the "red" and the "black" points. In the first half-step all red grid points are relaxed in a Jacobi-like manner (i.e. the actual change of grid values is not done until all red points are passed through). Using the new values at the red points, the second half-step relaxes all black grid points in an analogous way. For both half-steps, a relaxation parameter ω may be used. The above partial relaxation steps are described by "partial step operators" which will be denoted by S_h^{red} and S_h^{black} , respectively.

Notice that only for 5-point difference operators does RB relaxation coincide with Gauß-Seidel ω -relaxation with red-black ordering of the grid points. For "Poisson-like" equations (discretized by 5-point stars) this method of relaxation (with relaxation parameter $\omega=1$) probably yields the most efficient smoother of all.

ZEBRA relaxation (for compact 9-point difference operators) simply means Gauß-Seidel line-relaxation for which the lines (rows or columns) are not passed through one after the other but in a zebra-wise manner (first the even lines and then the odd lines or vice versa). We use the notation $x_1\text{-ZEBRA}$ for ZEBRA row-relaxation ($x_2\text{-ZEBRA}$ for ZEBRA column-relaxation). The respective partial step operators are denoted by $S_h^{x_1\text{-even}}$, $S_h^{x_1\text{-odd}}$, $(S_h^{x_2\text{-even}}, S_h^{x_2\text{-odd}})$. ZEBRA relaxation is of particular interest for anisotropic operators $-\epsilon u_{x_1 x_1} - u_{x_2 x_2} : x_1 - (x_2 -)$. ZEBRA has very efficient smoothing properties if $1 \ll \epsilon$ ($0 < \epsilon \ll 1$).

Alternating ZEBRA relaxation is defined by a combination of x_1 - and x_2 -ZEBRA. This yields a robust smoother, which is, for example, suitable for anisotropic operators with $\epsilon = \epsilon(x)$ such that $0 < \epsilon(x) \ll 1$ and $\epsilon(x) \gg 1$ for different x .

The partial relaxation steps mentioned above, both of RB and ZEBRA relaxation, can be written in the following way: For any approximation w_h to the solution u_h of (7.1), the new approximation \bar{w}_h is given by

$$\bar{w}_h = w_h + \omega(z_h - w_h) \quad (7.36)$$

(with some relaxation parameter ω) where z_h is determined by

$$\begin{aligned} L_h^0 z_h(x) + L_h^- w_h(x) &= f_h(x) \quad (x \in \tilde{\Omega}_h) \\ z_h(x) &= w_h(x) \quad (x \in \Omega_h \setminus \tilde{\Omega}_h). \end{aligned} \quad (7.37)$$

Here $\tilde{\Omega}_h$ is the subset of Ω_h consisting of those grid points, which are to be relaxed, and L_h^0 and L_h^- are connected by

$$L_h = L_h^0 + L_h^-.$$

L_h^0 is assumed to be invertible. Both $\tilde{\Omega}_h$ and L_h^0 are characteristic for the respective partial step. They are specified in Table 7.1.

partial relaxation step		$\tilde{\Omega}_h$	L_h^0
RB	red points	$\{x \in \Omega_h : x = \kappa \cdot h, \kappa_1 + \kappa_2 \text{ even}\}$	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & s_{0,0} & 0 \\ 0 & 0 & 0 \end{bmatrix}_h$
	black points	$\{x \in \Omega_h : x = \kappa \cdot h, \kappa_1 + \kappa_2 \text{ odd}\}$	
x_1 -ZEBRA	even rows	$\{x \in \Omega_h : x = \kappa \cdot h, \kappa_2 \text{ even}\}$	$\begin{bmatrix} 0 & 0 & 0 \\ s_{-1,0} & s_{0,0} & s_{1,0} \\ 0 & 0 & 0 \end{bmatrix}_h$
	odd rows	$\{x \in \Omega_h : x = \kappa \cdot h, \kappa_2 \text{ odd}\}$	
x_2 -ZEBRA	even columns	$\{x \in \Omega_h : x = \kappa \cdot h, \kappa_1 \text{ even}\}$	$\begin{bmatrix} 0 & s_{0,1} & 0 \\ 0 & s_{0,0} & 0 \\ 0 & s_{0,-1} & 0 \end{bmatrix}_h$
	odd columns	$\{x \in \Omega_h : x = \kappa \cdot h, \kappa_1 \text{ odd}\}$	

Table 7.1: Definition of $\tilde{\Omega}_h$ and L_h^0 in (7.37)

For the derivation of Fourier representations for the partial step operators, we consider the errors $v_h = u_h - w_h$, $\bar{v}_h = u_h - \bar{w}_h$. From (7.36), (7.37) one obtains

$$\bar{v}_h(x) = \begin{cases} (I_h - \omega(L_h^0)^{-1} L_h) v_h(x) & (x \in \tilde{\Omega}_h) \\ v_h(x) & (x \in \Omega_h \setminus \tilde{\Omega}_h) \end{cases} \quad (7.38)$$

In particular, any frequency $\varphi(\theta, x)$ ($\theta \in T_h$) is mapped into

$$\bar{\varphi}(\theta, x) := \begin{cases} A(\theta, h; \omega) \varphi(\theta, x) & (x \in \tilde{\Omega}_h) \\ \varphi(\theta, x) & (x \in \Omega_h \setminus \tilde{\Omega}_h) \end{cases} \quad (7.39)$$

Here

$$A(\theta, h; \omega) = 1 - \omega \frac{\lambda(\theta, h)}{\lambda^0(\theta, h)} \quad (7.40)$$

and $\lambda(\theta, h)$, $\lambda^0(\theta, h)$ denote the eigenvalues of L_h and L_h^0 , respectively, with respect to $\varphi(\theta, x)$ ($\theta \in T_h$). In particular, we have:

$$\lambda^0(\theta, h) = \begin{cases} s_{00} & (\text{RB}) \\ \sum_{k=0}^{\infty} s_k \cos(\theta k) & (x_1\text{-ZEBRA}) \\ \sum_{k=0}^{\infty} s_k \cos(\theta k) & (x_2\text{-ZEBRA}) \end{cases} \quad (7.41)$$

(as for $\lambda(\theta, h)$, see (7.9)).

Clearly, (7.39) is not yet a Fourier representation of the partial step operator defined by (7.38). However, $\bar{\varphi}$ in (7.39) can always be written as a linear combination of only a few of the eigenfrequencies (7.4), depending on the concrete "relaxation pattern" used. It turns out that all of the above partial step operators - and by this also the complete relaxation operators - have invariant subspaces which already occurred in connection with the coarse-grid correction operators in the previous section. The *minimal invariant subspaces* actually depend on the relaxation pattern used. In particular, the iteration operators of

RB, x_1 -ZEBRA, x_2 -ZEBRA and alternating ZEBRA

leave the spaces of harmonics invariant which correspond to

red-black, x_2 -, x_1 - and standard coarsening,

respectively. Instead of giving representations with respect to these minimal spaces, we prefer to use the spaces of $2h$ -harmonics throughout. This is mainly for two reasons:

- (1) Most concrete results given refer to the case of standard coarsening.
- (2) If, e.g., semi-coarsening is combined with RB relaxation, the minimal invariant spaces of the corresponding operators K_h^H and S_h are different. Therefore, the respective representations are not compatible. Both operators - and therefore also M_h^H - can, however, be represented with respect to the $2h$ -harmonics.

The Fourier representations of all partial step operators mentioned above are given in Table 7.2. The distribution of zero entries in the respective matrices show that they can be decomposed still further. As one easily recognizes, this reflects just the invariance properties with respect to the minimal spaces mentioned above.

In the case of alternating ZEBRA, four partial steps are connected (e.g. x_2 -odd, x_2 -even, x_1 -even, x_1 -odd). The corresponding product matrices do not allow a further decomposition

Fourier representation of partial step operators with respect to $E_{h,\theta}^{2h}$			
relaxation	first partial step	second partial step	$\theta \in T_h^{2h}$
RB 	$\hat{S}_{h,\theta}^{\text{red}} = \frac{1}{2} \begin{bmatrix} A_{00}+1 & A_{11}-1 & & \\ A_{00}-1 & A_{11}+1 & & \\ & & 0 & \\ & & & A_{10}+1 & A_{01}-1 \\ & & 0 & & \\ & & & A_{10}-1 & A_{01}+1 \end{bmatrix}$	$\hat{S}_{h,\theta}^{\text{black}} = \frac{1}{2} \begin{bmatrix} A_{00}+1 & -A_{11}+1 & & \\ -A_{00}+1 & A_{11}+1 & & \\ & & 0 & \\ & & & A_{10}+1 & -A_{01}+1 \\ & & 0 & & \\ & & & -A_{10}+1 & A_{01}+1 \end{bmatrix}$	$ \theta < \pi/2$
	$\hat{S}_{h,\theta}^{\text{red}} = \frac{1}{2} \begin{bmatrix} A_{00}+1 & A_{11}-i & & \\ A_{00}-1 & A_{11}+1 & & \\ & & & \\ & & & \end{bmatrix}$	$\hat{S}_{h,\theta}^{\text{black}} = \frac{1}{2} \begin{bmatrix} A_{00}+1 & -A_{11}+1 & & \\ -A_{00}+1 & A_{11}+1 & & \\ & & & \\ & & & \end{bmatrix}$	$ \theta = \pi/2,$ $\theta \approx (\pi/2, \pi/2)$
	$\hat{S}_{h,\theta}^{\text{red}} = (A_{00}) = (1-\omega)$	$\hat{S}_{h,\theta}^{\text{black}} = (1)$	$\theta = (\pi/2, \pi/2)$
x_1 -ZEBRA 	$\hat{S}_{h,\theta}^{x_1\text{-even}} = \frac{1}{2} \begin{bmatrix} A_{00}+1 & 0 & 0 & A_{01}-1 \\ 0 & A_{11}+1 & A_{10}-1 & 0 \\ 0 & A_{11}-1 & A_{10}+1 & 0 \\ A_{00}-1 & 0 & 0 & A_{01}+1 \end{bmatrix}$	$\hat{S}_{h,\theta}^{x_1\text{-odd}} = \frac{1}{2} \begin{bmatrix} A_{00}+1 & 0 & 0 & -A_{01}+1 \\ 0 & A_{11}+1 & -A_{10}+1 & 0 \\ 0 & -A_{11}+1 & A_{10}+1 & 0 \\ -A_{00}+1 & 0 & 0 & A_{01}+1 \end{bmatrix}$	$ \theta < \pi/2$
	$\hat{S}_{h,\theta}^{x_1\text{-even}} = \frac{1}{2} \begin{bmatrix} A_{00}+1 & A_{01}-1 \\ A_{00}-1 & A_{01}+1 \end{bmatrix}$	$\hat{S}_{h,\theta}^{x_1\text{-odd}} = \frac{1}{2} \begin{bmatrix} A_{00}+1 & -A_{01}+1 \\ -A_{00}+1 & A_{01}+1 \end{bmatrix}$	$ \theta = \pi/2,$ $\theta \neq (\pi/2, \pi/2)$
	$\hat{S}_{h,\theta}^{x_1\text{-even}} = (1)$	$\hat{S}_{h,\theta}^{x_1\text{-odd}} = (A_{00}) = (1-\omega)$	$\theta = (\pi/2, \pi/2)$
x_2 -ZEBRA 	$\hat{S}_{h,\theta}^{x_2\text{-even}} = \frac{1}{2} \begin{bmatrix} A_{00}+1 & 0 & A_{10}-1 & 0 \\ 0 & A_{11}+1 & 0 & A_{01}-1 \\ A_{00}-1 & 0 & A_{10}+1 & 0 \\ 0 & A_{11}-1 & 0 & A_{01}+1 \end{bmatrix}$	$\hat{S}_{h,\theta}^{x_2\text{-odd}} = \frac{1}{2} \begin{bmatrix} A_{00}+1 & 0 & -A_{10}+1 & 0 \\ 0 & A_{11}+1 & 0 & -A_{01}+1 \\ -A_{00}+1 & 0 & A_{10}+1 & 0 \\ 0 & -A_{11}+1 & 0 & A_{01}+1 \end{bmatrix}$	$ \theta < \pi/2$
	$\hat{S}_{h,\theta}^{x_2\text{-even}} = \frac{1}{2} \begin{bmatrix} A_{00}+1 & A_{10}-1 \\ A_{00}-1 & A_{10}+1 \end{bmatrix}$	$\hat{S}_{h,\theta}^{x_2\text{-odd}} = \frac{1}{2} \begin{bmatrix} A_{00}+1 & -A_{10}+1 \\ -A_{00}+1 & A_{10}+1 \end{bmatrix}$	$ \theta = \pi/2,$ $\theta \neq (\pi/2, \pi/2)$
	$\hat{S}_{h,\theta}^{x_2\text{-even}} = (1)$	$\hat{S}_{h,\theta}^{x_2\text{-odd}} = (A_{00}) = (1-\omega)$	$\theta = (\pi/2, \pi/2)$

Table 7.2: Fourier representations with respect to $E_{h,\theta}^{2h}$, $\theta \in T_h^{2h}$ (see 7.21)). Here we use the abbreviation (for any fixed θ , h , ω and all α as in (7.21)):

$$A_\alpha := A(\theta^\alpha, h; \omega) \quad (7.42)$$

with A as defined in (7.40), (7.41). The leftmost column illustrates which basis functions of $E_{h,\theta}^{2h}$ are actually coupled (cf. Figure 7.1, "standard coarsening").

We conclude this section with some remarks concerning simplifications and extensions:

Remarks: (1) In the case of RB relaxation with $\omega=1$ for 5-point operators L_h , the relations

$$A_{11} = -A_{00}, \quad A_{01} = -A_{10} \quad (7.43)$$

hold. Therefore, the representation $\hat{S}_{h,\theta}$ of the complete RB relaxation operator $S_h = S_h^{\text{black}} S_h^{\text{red}}$ becomes

$$\hat{S}_{h,\theta} = \begin{cases} \frac{1}{2} \begin{array}{c|c} \begin{matrix} A_{00}(1+A_{00}) & -A_{00}(1+A_{00}) \\ A_{00}(1-A_{00}) & -A_{00}(1-A_{00}) \end{matrix} & 0 \\ \hline 0 & \begin{matrix} A_{10}(1+A_{10}) & -A_{10}(1+A_{10}) \\ A_{10}(1-A_{10}) & -A_{10}(1-A_{10}) \end{matrix} \end{array} & (|\theta| < \pi/2) \\ \frac{1}{2} \begin{array}{c|c} \begin{matrix} A_{00}(1+A_{00}) & -A_{00}(1+A_{00}) \\ A_{00}(1-A_{00}) & -A_{00}(1-A_{00}) \end{matrix} & \begin{matrix} A_{10}(1+A_{10}) & -A_{10}(1+A_{10}) \\ A_{10}(1-A_{10}) & -A_{10}(1-A_{10}) \end{matrix} \end{array} & (|\theta|=\pi/2, \theta \neq (\pi/2, \pi/2)) \\ (0) & (\theta=(\pi/2, \pi/2)) \end{cases} \quad (7.44)$$

Obviously, all (2,2)-blocks of $\hat{S}_{h,\theta}$ are of rank 1. The range of S_h therefore has only about half the dimension of $\mathcal{G}(\Omega_h)$. This fact can be exploited for the computation of two-grid convergence factors (see, for example, Section 8.1).

(2) Similar relations hold in the case of ZEBRA relaxation with $\omega=1$ (for 9-point operators L_h):

$$\begin{aligned} A_{01} &= -A_{00}, \quad A_{10} = -A_{11} && (\text{for } x_1\text{-ZEBRA}), \\ A_{10} &= -A_{00}, \quad A_{01} = -A_{11} && (\text{for } x_2\text{-ZEBRA}). \end{aligned} \quad (7.45)$$

They lead to corresponding simplifications as in (7.44).

(3) One complete step of Jacobi ω -relaxation (both point and line relaxation) is a special case of (7.38), characterized by $\tilde{\Omega}_h = \Omega_h$. For the three cases point, x_1 -line and x_2 -line relaxation, L_h^0 is here defined in the same way as for RB, x_1 -ZEBRA and x_2 -ZEBRA, respectively (see Table 7.1). The corresponding relaxation operators S_h are represented by diagonal matrices: We have

$$S_h \varphi(\theta, x) = A(\theta, h; \omega) \varphi(\theta, x) \quad (\theta \in T_h) \quad (7.46)$$

with A defined in (7.40), (7.41).

(4) Analogous to RB relaxation ("two-colour relaxation"), one can define a general k -colour relaxation [24]. Here Ω_h is divided into k natural subsets each of which is passed through in a Jacobi-like manner. Thus one complete step of k -colour relaxation consists of k partial steps of the form (7.37). For compact 9-point operators as considered here, 4-colour relaxation coincides with Gauß-Seidel relaxation with a proper enumeration of grid points. The above Fourier analysis can immediately be applied to the corresponding partial step operators (giving full $(4,4)$ matrices now). See [98] for details.

7.4 Two-grid operator

In Chapter 8, we will give results for several fundamental model problems and two-grid methods in terms of the following quantities

$$\begin{aligned} \rho(h, v) &:= \rho(M_h^H(v_1, v_2)), & \rho^*(v) &:= \sup \{\rho(h, v) : h \in \mathcal{H}^*\}; \\ \sigma_S(h, v_1, v_2) &:= \|M_h^H(v_1, v_2)\|_S, & \sigma_S^*(v_1, v_2) &:= \sup \{\sigma_S(h, v_1, v_2) : h \in \mathcal{H}^*\}; \\ \sigma_d(h, v_1, v_2) &:= \|L_h M_h^H(v_1, v_2) L_h^{-1}\|_S, & \sigma_d^*(v_1, v_2) &:= \sup \{\sigma_d(h, v_1, v_2) : h \in \mathcal{H}^*\}. \end{aligned}$$

In addition, we will consider

$$\sigma_E(h, v_1, v_2) := \|M_h^H(v_1, v_2)\|_E, \quad \sigma_E^*(v_1, v_2) := \sup \{\sigma_E(h, v_1, v_2) : h \in \mathcal{H}\}$$

which is a meaningful norm if L_h is positive definite. Here \mathcal{H}^* is defined as in (1.26) with some vector of coarsest meshsizes h^* and fixed meshsize ratio q^* . Because of space limitations, we will confine ourselves to the detailed discussion of methods using standard coarsening in Chapter 8. (For methods which use red-black or semi-coarsening, see, for example, [82], [111]).

The actual computation of the above quantities is performed by use of Fourier representations for M_h^H (as considered in the previous sections). In the case of standard coarsening, for instance, we have

$$\left. \begin{aligned} \rho(h, v) &= \max \{\rho(\hat{M}_{h,\theta}^{2h}) : \theta \in T_h^{2h}\}, \\ \sigma_S(h, v_1, v_2) &= \max \{\|\hat{M}_{h,\theta}^{2h}\|_S : \theta \in T_h^{2h}\}, \\ \sigma_d(h, v_1, v_2) &= \max \{\|\hat{L}_{h,\theta} \hat{M}_{h,\theta}^{2h} \hat{L}_{h,\theta}^{-1}\|_S : \theta \in T_h^{2h}\}, \\ \sigma_E(h, v_1, v_2) &= \max \{\|\hat{L}_{h,\theta}^{1/2} \hat{M}_{h,\theta}^{2h} \hat{L}_{h,\theta}^{-1/2}\|_S : \theta \in T_h^{2h}\}. \end{aligned} \right\} \quad (7.47)$$

Here ' \wedge ' denotes the representation of all operators occurring with respect to the spaces of $2h$ -harmonics.

7.5 General definition of smoothing factors

Clearly, the asymptotic convergence behavior of a two-grid method is characterized by the spectral radius $\rho(M_h^H)$ of the corresponding iteration operator. A priori, however, it is not clear how to measure the smoothing properties of a general relaxation process.

The purpose of introducing such a smoothing measure is to separate the influences of the two main parts of a two-grid method, the smoothing and the coarse-grid correction part, on the error reduction. A reasonable "smoothing factor" should - at least - give some information on the reduction of the high-frequency error part which is achieved by one smoothing step. The following definition does not take into account any transfer or coarse-grid difference operators. Nevertheless, the smoothing factor yields some information on the behavior of a two-grid method when some "idealized" assumptions on the coarse-grid correction part are made (see below).

In Section 3.2, we defined such a smoothing factor μ (or μ^*) to be the worst factor by which high-frequency error components are reduced per relaxation step. This was possible as all high frequencies were eigenfunctions of S_h . The situation is the same here, as long as we use Jacobi ω -relaxation (point- or line-wise, see Remark (3) in Section 7.3). For relaxation processes with more complicated smoothing operators like RB or ZEBRA relaxation, we have to extend the definition of a smoothing factor.

For the general definition of a smoothing factor, we now assume that only a coarser grid Ω_h and a corresponding splitting

$$\mathbb{G}(\Omega_h) = \mathbb{G}^{\text{low}}(\Omega_h) \oplus \mathbb{G}^{\text{high}}(\Omega_h) \quad (7.48)$$

are given (several examples have been discussed in Section 7.2). As motivated above, the "real" coarse-grid operator K_h^H is replaced by an "ideal" operator Q_h^H which has the following properties: Q_h^H annihilates the low-frequency error components and leaves the high-frequency components unchanged. Thus Q_h^H is a projection operator onto the space of high frequencies:

$$Q_h^H w_h := \begin{cases} w_h & (w_h \in \mathbb{G}^{\text{high}}(\Omega_h)) \\ 0 & (w_h \in \mathbb{G}^{\text{low}}(\Omega_h)) \end{cases} \quad (7.49)$$

Then the quantity

$$\rho(S_h^v Q_h^H S_h^v) = \rho(Q_h^H S_h^v)$$

should be a good measure for the total smoothing effect of applying v smoothing steps of S_h . In addition, one can hope that this quantity gives a realistic prediction of the spectral radius of $M_h^H (= \rho(Q_h^H S_h^v))$ as long as the transfer and coarse-grid difference operators are "sufficiently good".

This leads to the following

Definition: We define the *smoothing factor* μ of S_h by

$$\mu(h, v) := \sqrt{\rho(Q_h^H S_h^v)} \quad (7.50)$$

which measures the average smoothing effect per smoothing step. As usual, we denote by μ^* the corresponding supremum with respect to the admissible meshsizes h :

$$\mu^*(v) := \sup \{ \mu(h, v) : h \in \mathcal{H} \}. \quad (7.51)$$

Remark: As one can easily see, the above definition is indeed an extension of the one given in Section 3.2: In the case considered there, S_h had, in particular, the property

$$S_h : \mathbb{E}^{\text{high}}(\Omega_h) \rightarrow \mathbb{E}^{\text{high}}(\Omega_h). \quad (7.52)$$

In this case, (7.50) can be simplified to

$$\mu(h, v) := \rho(S_h^{\text{high}}) \quad \text{with} \quad S_h^{\text{high}} := S_h|_{\mathbb{E}^{\text{high}}(\Omega_h)}. \quad (7.53)$$

This coincides with the definition (3.16) given in Section 3.2. In particular, μ is independent of v . In general, however, low and high frequencies are intermixed by S_h , and μ does depend on v .

As an elucidation, we now consider operators S_h corresponding to any of the complete relaxation methods considered in Section 7.3. For the actual computation of smoothing factors of these methods, we can take advantage of their respective invariance properties. In Table 7.2 we have given the matrix representations $\hat{S}_{h,\theta}$ of S_h with respect to the spaces of $2h$ -harmonics. Trivially, Q_h^H is represented by diagonal matrices $\hat{Q}_{h,\theta}^H$ ($\theta \in T_h^{2h}$) with respect to the corresponding basis (see (7.21)). These matrices are given in Table 7.3 (for all different coarsenings). Using these representations, we obtain (for any fixed type of coarsening):

$$\mu(h, v) = \max \{ \sqrt{\rho(\hat{Q}_{h,\theta}^H \hat{S}_{h,\theta}^v)} : \theta \in T_h^{2h} \}. \quad (7.54)$$

coarsening	matrix representation of Q_h^H with respect to $E_{h,\theta}^{2h}$ ($\theta \in T_h^{2h}$)			
standard coarsening	$ \theta < \pi/2$	$ \theta = \pi/2, \theta \neq (\pi/2, \pi/2)$	$\theta = (\pi/2, \pi/2)$	
	$\begin{bmatrix} 0 & & & \\ 1 & 1 & & \\ & 1 & 1 & \\ & & 1 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & \\ 1 & 1 \end{bmatrix}$		(1)
x_1 -coarsening	$ \theta < \pi/2$	$ \theta = \pi/2, \theta_1 < \pi/2$	$ \theta = \pi/2, \theta_2 < \pi/2$	$\theta = (\pi/2, \pi/2)$
	$\begin{bmatrix} 0 & & & \\ & 1 & & \\ & 1 & 1 & \\ & & 0 & \end{bmatrix}$	$\begin{bmatrix} 0 & \\ & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & \\ 1 & 1 \end{bmatrix}$	(1)
x_2 -coarsening	$ \theta < \pi/2$	$ \theta = \pi/2, \theta_1 < \pi/2$	$ \theta = \pi/2, \theta_2 < \pi/2$	$\theta = (\pi/2, \pi/2)$
	$\begin{bmatrix} 0 & & & \\ & 1 & & \\ & & 0 & \\ & & & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & \\ 1 & 1 \end{bmatrix}$	$\begin{bmatrix} 0 & \\ & 1 \end{bmatrix}$	(1)
red-black coarsening	$ \theta < \pi/2, \theta_2 \geq \theta_1$	$ \theta < \pi/2, \theta_2 < \theta_1$	$ \theta = \pi/2, \theta \neq (\pi/2, \pi/2)$	$\theta = (\pi/2, \pi/2)$
	$\begin{bmatrix} 0 & & & \\ & 1 & & \\ & 1 & 1 & \\ & & 0 & \end{bmatrix}$	$\begin{bmatrix} 0 & & \\ & 1 & & \\ & & 0 & \\ & & & 1 \end{bmatrix}$	$\begin{bmatrix} 0 & \\ & 1 \end{bmatrix}$	(0)
quadrupling	$ \theta < \pi/4$	$\pi/4 \leq \theta < \pi/2$	$ \theta = \pi/2, \theta \neq (\pi/2, \pi/2)$	$\theta = (\pi/2, \pi/2)$
	$\begin{bmatrix} 0 & & & \\ & 1 & & \\ & 1 & 1 & \\ & & 1 & \end{bmatrix}$	$\begin{bmatrix} 1 & & \\ 1 & 1 & \\ 1 & & 1 \end{bmatrix}$	$\begin{bmatrix} 1 & \\ 1 & 1 \end{bmatrix}$	(1)

Table 7.3: Matrix representations $\hat{Q}_{h,\theta}^H$ of the projection operator Q_h^H with respect to $E_{h,\theta}^{2h}$ ($\theta \in T_h^{2h}$) for different coarsenings.

Table 7.4 gives a comparison of smoothing factors $\mu^*(v)$ for all combinations of smoothers and coarsenings treated in this chapter. Table 7.4a refers to the model problem (P) and Table 7.4b to the anisotropic model operator with $\epsilon=0.1$ (cf. Section 1.3.3). The range \mathcal{X}^* of admissible h is

$$\mathcal{X}^* := \{h : h \leq h^*, h=1/N, N \text{ even}\}, \quad h^* := 1/4.$$

We are not going to discuss all of the results in Table 7.4 and their significance for the construction of multigrid methods in detail here. We mainly pick out some exemplary cases which are of interest for the discussions of two-grid methods in Chapters 8 and 9. In all of the examples below, the following quantity $x(v)$ occurs:

$$x(v) := \left(\frac{2v-1}{2v} \right)^{\frac{2}{v}} \sqrt[2]{\frac{v}{2(2v-1)}}. \quad (7.55)$$

Some particular values are

$$x(1) = 0.125, \quad x(2) = 0.229\dots, \quad x(3) = 0.322\dots, \quad x(v) \sim \sqrt[2]{1/4ev} \quad (v \text{ "large"}). \quad (7.56)$$

		type of relaxation			
coarsening	ν	RB	x_1 -ZEBRA	x_2 -ZEBRA	alt.ZEBRA
standard coarsening	1	0.250	0.250	0.250	0.048
	2	0.250	0.250	0.250	0.118
	3	0.322	0.322	0.322	-
quadrupling	1	0.729	0.598	0.598	0.380
	2	0.729	0.598	0.598	0.380
	3	0.729	0.598	0.598	-
x_1 -coarsening	1	0.375	0.250	0.125	0.028
	2	0.306	0.250	0.230	0.092
	3	0.322	0.250	0.322	-
x_2 -coarsening	1	0.375	0.125	0.250	0.028
	2	0.306	0.230	0.250	0.092
	3	0.322	0.322	0.250	-
red-black coarsening	1	0.125	0.125	0.125	0.012
	2	0.230	0.230	0.230	0.078
	3	0.322	0.322	0.322	-

Table 7.4a: Smoothing factors $\mu^*(\nu)$ for model problem (P)

		type of relaxation			
coarsening	ν	RB	x_1 -ZEBRA	x_2 -ZEBRA	alt.ZEBRA
standard coarsening	1	0.826	0.826	0.125	0.102
	2	0.826	0.826	0.230	0.203
	3	0.826	0.826	0.322	-
quadrupling	1	0.947	0.944	0.500	0.496
	2	0.947	0.944	0.500	0.496
	3	0.947	0.944	0.500	-
x_1 -coarsening	1	0.868	0.826	0.125	0.101
	2	0.847	0.826	0.230	0.203
	3	0.840	0.826	0.322	-
x_2 -coarsening	1	0.125	0.125	0.008	0.002
	2	0.230	0.230	0.032	0.032
	3	0.322	0.322	0.100	-
red-black coarsening	1	0.744	0.694	0.125	0.084
	2	0.706	0.694	0.230	0.195
	3	0.693	0.694	0.322	-

Table 7.4b: Analogous to Table 7.4a for the anisotropic model operator ($\epsilon=0.1$)

Examples: (1) We consider RB relaxation (with relaxation parameter $\omega=1$) in connection with model problem (P). The matrix representation of one complete RB step with respect to $E_{h,\theta}^{2h}$ ($\theta \in T_h^{2h}$) is given in (7.44). We have for fixed h and $\theta \in T_h^{2h}$

$$A_{00} = A(\theta^{00}, h) = (\cos\theta_2 + \cos\theta_1)/2, \quad A_{10} = A(\theta^{10}, h) = (\cos\theta_2 - \cos\theta_1)/2. \quad (7.57)$$

With respect to standard coarsening, the smoothing factor $\mu(h, v)$ can be computed by (7.54) using the corresponding matrix representation of Q_h^{2h} in Table 7.3. For $\mu^*(v)$ one obtains

$$\mu^*(v) = \sqrt{\sup \{ \max \{ |A_{00}^{2v-1}(1-A_{00})|/2, |A_{10}^{2v}| \} : |\theta| \leq \pi/2 \}} \quad (7.58)$$

$$= \max \{0.25, \chi(v)\} = \begin{cases} 0.25 & (v \leq 2) \\ \chi(v) & (v \geq 3) \end{cases}. \quad (7.59)$$

A comparison with Jacobi ω -relaxation (cf. (3.18)) shows that the smoothing properties of RB are - for relevant (small) values of v - considerably better. For large values of v , however, the smoothing factor of RB approaches 1 (cf. (7.56)). This is due to the coupling of low and high frequencies within one RB step. Thus, here already the smoothing factor indicates that the performance of too many smoothing steps ($v>3$, say) is not recommendable.

(2) For RB relaxation and the 5-point discretization (1.12) of the anisotropic model equation, we have

$$A_{00} = (\cos\theta_2 + \epsilon \cos\theta_1)/(1+\epsilon), \quad A_{10} = (\cos\theta_2 - \epsilon \cos\theta_1)/(1+\epsilon) \quad (7.60)$$

instead of (7.57). Evaluating (7.58), we obtain the smoothing factor with respect to standard coarsening

$$\mu^*(v) = \max \{(\epsilon^+/(1+\epsilon))^2, \chi(v)\}, \quad \epsilon^+ := \max \{1, \epsilon\}. \quad (7.61)$$

In particular, we obtain (for fixed v) $\mu^*(v) \rightarrow 1$ if either $\epsilon \rightarrow 0$ or $\epsilon \rightarrow \infty$, showing that the smoothing properties of RB relaxation become very bad if ϵ is considerably different from 1. For some concrete values, see the Tables 7.4 and 8.3. Heuristically, the reason for this is clear: If ϵ is, for example, very small, there is nearly no coupling of grid values in x_1 -direction (cf. (1.12)). In particular, there is hardly a smoothing effect on the error in this direction. The error is, however, smoothed in the x_2 -direction. This can be seen from the next example:

(3) Let the problem and the smoother be the same as in Example (2). Now, however, we consider the smoothing effect of RB relaxation with respect to x_2 -coarsening. With A_{00} and A_{10} as in (7.60) we now obtain

$$\mu^*(v) = \sqrt{\sup \{ \max \{ |A_{00}^{2v-1}(1-A_{00})|/2, |A_{10}^{2v-1}(1-A_{10})|/2 \} : |\theta| \leq \pi/2 \}} \quad (7.62)$$

and

$$\mu^*(v) \rightarrow \begin{cases} x(v) & (\epsilon \rightarrow 0) \\ 1 & (\epsilon \rightarrow \infty). \end{cases} \quad (7.63)$$

Thus RB relaxation has, with respect to x_2 -coarsening and for small ϵ , the same smoothing properties as in Example (1) if $v \geq 3$. (For $v=1$ or 2 they are even better.) For large ϵ , however, RB cannot be used in connection with x_2 -coarsening: Then x_1 -coarsening has to be used instead. This result is in full accordance with the heuristic explanation given in (2).

(4) We have seen above that - for the anisotropic model equation with ϵ considerably different from 1 - RB has good smoothing properties only if it is combined with semi-coarsening. (In fact, this is true for any pointwise relaxation method by the same heuristic argument which was given in Example (2).) In order to use standard coarsening, the smoothing process has to be changed: One can use, for instance, ZEBRA relaxation. The matrix representation of ZEBRA relaxation can be computed from Table 7.2 (also see (7.45)). Let us consider x_2 -ZEBRA. With

$$A_{00} = \epsilon \cos \theta_1 / (1 + \epsilon - \cos \theta_2), \quad A_{11} = -\epsilon \cos \theta_1 / (1 + \epsilon + \cos \theta_2), \quad (7.64)$$

one obtains the same formula as (7.58) with A_{10} replaced by A_{11} and by that

$$\mu^*(v) = \max \{ (\epsilon/(1+\epsilon))^2, x(v) \} \rightarrow \begin{cases} x(v) & (\epsilon \rightarrow 0) \\ 1 & (\epsilon \rightarrow \infty). \end{cases} \quad (7.65)$$

In particular, x_2 -ZEBRA has, in connection with standard coarsening and for small ϵ , the same good smoothing properties as RB had in connection with x_2 -coarsening (7.63). For large ϵ , however, x_2 -ZEBRA is not suitable: x_1 -ZEBRA has to be used instead. If one smoothing step is defined by one step of alternating ZEBRA, one can show that this kind of a smoother has very good smoothing properties, independent of the size of ϵ . Some explicit values are given in Table 7.4 (also see Table 8.4b). In the judgement of alternating ZEBRA one has, of course, to take into account that this smoother needs twice the work per step as one single ZEBRA step.

We want to make one final remark on coarsening by quadrupling h. This kind of coarsening leads to multigrid algorithms which are (slightly) cheaper per cycle than corresponding ones obtained by, for example, standard coarsening (cf. (4.24)). On the other hand, the smoothing factors shown in Table 7.4 indicate that this saving of computational work does not pay: the smoothing factors which correspond to coarsening by quadrupling h are much worse than those which correspond to standard coarsening.

7.6 Modifications and extensions

The concept of model problem analysis was so far described on the basis of the discrete sine functions (7.4). For second order differential operators (7.6) (on rectangular domains), these sine functions naturally occur as eigenfunctions of L_h in connection with *Dirichlet boundary conditions*.

The whole concept carries over, if, instead of (7.4), cosine or (complex) exponential functions (or certain combinations of them) are used. More precisely, we may consider functions

$$\varphi(\theta, x) = \varphi_1(\theta_1, x_1) \varphi_2(\theta_2, x_2) \quad (7.66)$$

where φ_1 and φ_2 are functions of the form

$$\varphi_j(\theta_j, x_j) = \begin{cases} \sin(\theta_j x_j / h_{x_j}) \\ \cos(\theta_j x_j / h_{x_j}) , \\ e^{i\theta_j x_j / h_{x_j}} \end{cases}, \quad (j = 1, 2). \quad (7.67)$$

Examples: (1) Let the same differential equation (7.6) be given as in Example 7.1, but with *Neumann boundary conditions*

$$\partial u / \partial n = f^T(x) \quad (x \in \Gamma) \quad (7.68)$$

instead of (7.7). For L^Ω , we consider any discretization of the form (7.8) applied at all points of $\bar{\Omega} \cap G_h$ using auxiliary grid points outside Ω ; these points are also used in discretizing the Neumann boundary conditions by two-point central differences. The grid values corresponding to auxiliary points are assumed to be eliminated then. Thus, we obtain a grid equation (7.1) on $\Omega_h := \bar{\Omega} \cap G_h$ the eigenfunctions of which are (up to normalization)

$$\varphi(\theta, x) = \cos(\theta_1 x_1 / h_{x_1}) \cos(\theta_2 x_2 / h_{x_2}) \quad (\theta \in T_h) \quad (7.69)$$

with T_h as in (7.5) but with $0 \leq n \leq N$ instead of $0 < n < N$.

(2) Similarly, (7.6) with *periodic boundary conditions* can be discretized in a straightforward manner such that one obtains a grid equation (7.1) on

$$\Omega_h := \{x \in G_h : 0 \leq x_j < A_j \quad (j=1,2)\} \quad (7.70)$$

with eigenfunctions

$$\varphi(\theta, x) = e^{i\theta_1 x_1 / h_{x_1}} e^{i\theta_2 x_2 / h_{x_2}} \quad (\theta \in T_h), \quad T_h := \{\theta = \left(\frac{2n_1}{N_2} \pi, \frac{2n_2}{N_2} \pi \right) : -N/2 < n \leq N/2\}. \quad (7.71)$$

Clearly, mixtures of the above functions correspond to certain mixed boundary conditions. All these problems can be analyzed in much the same way as shown in Sections 7.1 through 7.5. There are only a few technical changes necessary. In particular, changes are caused by the grid Ω_h (which is slightly different now) and by the different θ -ranges. For demonstration purposes, we make some comments on the Neumann case as described above in Example (1).

In this case, the definition of high and low frequencies given in Section 7.2 is maintained except that now additional frequencies occur. If we include these additional frequencies, the definition of the spaces of harmonics remains also unchanged. As the functions (7.69) are not orthogonal with respect to (1.7), the definition of the inner product has to be modified: We replace (1.7) by

$$(u_h, w_h)_2 := \frac{1}{N_1 N_2} \sum_{x \in \Omega_h} u_h(x) \bar{w}_h(x) c_h(x) \quad (7.72)$$

where

$$c_h(x) := \begin{cases} 1 & (x \in \Omega) \\ 1/2 & (x \in \partial\Omega, \text{ edges}) \\ 1/4 & (x \in \partial\Omega, \text{ corners}). \end{cases}$$

With respect to this inner product, the functions in (7.69) are orthogonal (and can be normalized).

Taking these formal changes into account, the results of the previous sections carry over in a straightforward manner. In particular, (7.33) and (7.34) of Lemma 7.1 remain valid if we extend the definition of I_h^{2h} and I_{2h}^h to grid points at the boundary of Ω . This extension has to be done in a way which takes the symmetric properties of the above cosine functions into account. For instance, at the left boundary (away from the corners), I_h^{2h} and I_{2h}^h are defined by

$$I_h^{2h} : \begin{bmatrix} 0 & \hat{t}_{0,1} & 2\hat{t}_{1,1} \\ 0 & \hat{t}_{0,0} & 2\hat{t}_{1,0} \\ 0 & \hat{t}_{0,-1} & 2\hat{t}_{1,-1} \end{bmatrix}_h^{2h}, \quad I_{2h}^h : \begin{bmatrix} 0 & \hat{t}_{0,1} & \hat{t}_{1,1} \\ 0 & \hat{t}_{0,0} & \hat{t}_{1,0} \\ 0 & \hat{t}_{0,-1} & \hat{t}_{1,-1} \end{bmatrix}_h^{2h} \quad (7.73)$$

and at the upper left corner by

$$I_h^{2h} : \begin{bmatrix} 0 & 0 & 0 \\ 0 & \hat{t}_{0,0} & 2\hat{t}_{1,0} \\ 0 & 2\hat{t}_{0,-1} & 4\hat{t}_{1,-1} \end{bmatrix}_h^{2h}, \quad I_{2h}^h : \begin{bmatrix} 0 & 0 & 0 \\ 0 & \frac{\sqrt{3}}{2}t_{0,0} & \frac{\sqrt{3}}{2}t_{1,0} \\ 0 & \frac{\sqrt{3}}{2}t_{0,-1} & \frac{\sqrt{3}}{2}t_{1,-1} \end{bmatrix}_{2h}^h. \quad (7.74)$$

Also, the analysis of relaxation operators given in Section 7.3 and their matrix representations do not change essentially. The definition of the smoothing factor applies directly. Altogether, one can compute spectral radii and norm quantities of two-grid methods just as before and one obtains nearly the same results. (The minor differences are caused only by the fact that the number of spaces of harmonics is slightly larger now. These differences vanish if one maximizes with respect to h .)

Remark: The model problems treated in Chapter 8 (5-point discretization of the Poisson equation and the anisotropic model equation) become singular in connection with pure Neumann conditions. Such cases were excluded so far as we always assumed L_h^{-1} and L_{2h}^{-1} to exist. It is quite easy to see, however, how the analysis has to be changed in order to include the above singular cases: One can use slightly shrunken spaces $\tilde{\mathcal{G}}(\Omega_h)$ and $\tilde{\mathcal{G}}(\Omega_{2h})$ defined by

$$\tilde{\mathcal{G}}(\Omega_h) := \mathcal{G}(\Omega_h) \setminus E_{h,0}^{2h}, \quad \tilde{\mathcal{G}}(\Omega_{2h}) := \mathcal{G}(\Omega_{2h}) \setminus \text{span}\{1\}.$$

where $E_{h,0}^{2h}$ denotes that space of harmonics which contains the lowest (constant) frequency. With respect to these spaces, L_h^{-1} and L_{2h}^{-1} exist and the analysis can be performed as usual. (One can easily arrange things so that the remaining three non-constant harmonics in $E_{h,0}^{2h}$ give no significant contribution.) These changes in the analysis also have direct analogs in the algorithmic construction of a corresponding two- (and multi-) grid method. Without going into details, we only mention that the results for Neumann boundary conditions are exactly the same as for Dirichlet conditions as far as ρ^* , σ_S^* (now defined with respect to (7.72)) are concerned. For more details and, in particular, for numerical investigations, we refer to [10].

Similar analogs as in the Dirichlet and Neumann case do hold also for the periodic case. We are not going to discuss the model problem analysis with respect to (7.71) here. We want to point out, however, that the technical details of an analysis with respect to (7.71) are implicitly contained in the local Fourier analysis as described in Chapter 9.

In most of our concrete investigations in the previous sections we have restricted our considerations to grid operators (L_h, I_h^H, I_H^h) which are described by 5- or compact 9-point stars. Technically, however, the model problem analysis also can be applied in connection with "larger" stars. For this, all larger stars have to be

changed near the boundary in such a way that the functions (7.4) - or more generally (7.66) - remain eigenfunctions of L_h^H and M_h^H has still the required invariance properties (see also the changes of I_h^H and I_H^h in (7.73) and (7.74)). Instead of performing such changes explicitly, there is an equivalent way of treating such situations which is theoretically more transparent: One considers the given problem not on Ω_h , but extends it (equivalently) to the infinite grid G_h . For this, for all grid functions occurring one has to perform certain continuation processes. The kind of continuation is, of course, given by the type of boundary conditions at hand. Without going into details, we only mention that there is a direct correspondence between an antisymmetric, symmetric and periodic continuation of grid functions and Dirichlet, Neumann and periodic boundary conditions, respectively. (Such continuation processes have been studied extensively in the framework of reduction methods [87], [71].)

Remark: Although larger stars can be included in the concept of model problem analysis, one has to be cautious with the interpretation of corresponding results. If changes near the boundary are performed in the way as sketched above, these operators may loose some of their properties. For example, a higher order discretization or interpolation becomes of lower order near the boundary. On the other hand, if changes near the boundary, e.g. for L_h , are done such that the order of approximation is maintained, the model problem analysis can, in general, not be applied directly as the functions (7.4) or (7.66) may possibly no longer be eigenfunctions.

8. Applications of model problem analysis

In this Chapter, we give some exemplary results derived by the analysis introduced in the previous chapter. We consider some of the most fundamental problems and algorithms and discuss them briefly.

In Section 8.1, we treat Poisson's equation, namely model problem (P), in detail. We consider MG components which lead to very efficient MG algorithms. The main intention here is to show how model problem analysis can be used to obtain explicit analytic expressions for convergence factors. Sections 8.2 and 8.3 give several additional results in form of tables. Section 8.2 again refers to Poisson's equation. In particular, we make some efficiency considerations. Besides the usual 5-point discretization, we consider the 9-point "Mehrstellen" discretization, also. Section 8.3 refers to the anisotropic model operator.

Other operators which can be treated by means of the model problem analysis as well, are not discussed here in detail. Such problems are: Helmholtz' equation (with positive or negative Helmholtz' constant), 4th order equations (cf. Example 7.2), problems with Neumann or periodic boundary conditions (cf. Section 7.6), certain systems of elliptic equations, higher order discretizations, etc.

All results of this chapter refer, for simplicity, to standard coarsening. As for results concerning red-black and semi-coarsening (MGR methods), we refer to [82],[111]. Furthermore, we restrict ourselves to square meshsizes $h=h_{x_1}=h_{x_2}$. This can be done without loss of generality, as the discretization of

$$-\varepsilon u_{x_1 x_1} - u_{x_2 x_2} \quad \text{with} \quad h_{x_1} = h/q^*, \quad h_{x_2} = h \quad (8.1)$$

is clearly equivalent to the discretization of

$$-\tilde{\varepsilon} u_{x_1 x_1} - u_{x_2 x_2} \quad \text{with} \quad h_{x_1} = h_{x_2} = h \quad \text{and} \quad \tilde{\varepsilon} = \varepsilon(q^*)^2. \quad (8.2)$$

As already pointed out in the introduction, we regard the model problem analysis (and the local Fourier analysis) as the theoretical basis for the construction of efficient multigrid solvers. In particular, the results given in this chapter have been used in developing the programs MG00 and MG01 for the solution of certain standard elliptic equations of second order (see Chapter 10 and [36]). The connection between model problem analysis and standard applications is - roughly - given by the following:

Each of the model problems represents a considerably larger class of more general "standard" problems (with variable coefficients, on more general domains etc.) to which the quantitative theoretical results obtained for the corresponding model problem carry over in practice. This "stable behavior" of multigrid methods is a general experience familiar to multigrid experts and has been demonstrated by a great number of systematical experiments. It is because of this behavior that the model problem analysis becomes really worthwhile for practical purposes.

The above behavior is heuristically explained by the fact that the spectral properties of M_h^{2h} , by which the two-grid convergence behavior is determined, are "robust" with respect to small changes of a given model problem (for example, with respect to changes of L_h or the domain). In particular, the influence of a given relaxation technique is nearly the same for neighboring problems; this is mainly due to the local nature of relaxation processes.

8.1 Analytic results for an efficient two-grid method

We consider an $(h, 2h)$ two-grid method for model problem (P) which is very similar to the sample method treated in Chapter 3. The only difference is that now RB relaxation (with relaxation parameter $\omega=1$) is used instead of Jacobi ω -relaxation. This small change of the algorithm will prove to be essential for the resulting efficiency. We obtain for the asymptotic convergence factor $\rho^*(v) = \sup \{ \rho(h, v) : h \in \mathcal{H} \}$, $h^* = 1/4$:

Theorem 8.1: Let

$$M_h^{2h} = S_h^{v_2} (I_h - I_{2h}^h L_{2h}^{-1} I_h^{2h} L_h) S_h^{v_1} \quad (8.3)$$

where L_h , I_h^h , I_{2h}^h , L_{2h} are defined as in Section 3.1 and S_h characterizes one complete step of RB relaxation (cf. Section 7.3). Then we have for $v = v_1 + v_2$

$$\rho^*(v) = \begin{cases} 1/4 & (v = 1) \\ \frac{1}{2v} \left(\frac{v}{v+1}\right)^{v+1} & (v \geq 2). \end{cases} \quad (8.4)$$

In particular,

$$\rho^*(2) = 2/27 = 0.074\ldots, \quad \rho^*(3) = 0.052\ldots; \quad v \rho^*(v) \rightarrow 1/2e \quad (v \rightarrow \infty). \quad (8.5)$$

Remark: We obtain the same result on ρ^* if Poisson's equation is considered on a rectangular domain (with $h_{x_1} = h_{x_2}$) rather than on a square. The necessary changes in the proof below are obvious.

The asymptotic behavior of ρ^* shown in (8.5) already indicates that - for reasons of efficiency - it is not reasonable to choose too large a value of v . Taking computational work into account, we will see in Section 8.2 that $v=2$ is the optimal value. The corresponding convergence factor shows that the mere replacement of Jacobi ω -relaxation by RB relaxation leads to a considerably faster method (cf. Table 3.1). In addition, it requires less numerical operations per iteration step, see Section 8.2.

Before we give a proof for the above theorem, we would like to formulate related statements with respect to norms (Supplements 8.2 and 8.3). The norms σ_S^* and σ_E^* can be determined in a similar way as ρ^* . (We give some explicit values in Section 8.2.) For the method at hand, however, certain modified norm values turn out to be more interesting from a theoretical and practical point of view. In the following supplement concerning the spectral norm, we take advantage of the fact that S_h has a nullspace of a high dimension (cf. Remark (1) in Section 7.3). For $v_2 \geq 1$, we have

$$M_h^{2h} : \mathbb{G}(\Omega_h) \rightarrow \tilde{\mathbb{G}}(\Omega_h) \quad \text{where} \quad \tilde{\mathbb{G}}(\Omega_h) := S_h(\mathbb{G}(\Omega_h)). \quad (8.6)$$

We therefore consider

$$\tilde{\sigma}_S(h, v_1, v_2) := \| \tilde{M}_h^{2h} \|_S \quad \text{where} \quad \tilde{M}_h^{2h} := M_h^{2h} \Big|_{\tilde{\mathbb{G}}(\Omega_h)} \quad (8.7)$$

For the corresponding supremum $\tilde{\sigma}_S^*(v_1, v_2) := \sup \{ \tilde{\sigma}_S(h, v_1, v_2) : h \in \mathcal{H}^* \}$, $h^* = 1/4$, we obtain the

Supplement 8.2: Let the same assumptions as in Theorem 8.1 be satisfied. Then

$$\tilde{\sigma}_S^*(v_1, v_2) = \begin{cases} 1/\sqrt{2} & \text{for } v_1 = 0, v_2 \geq 1 \\ \rho^*(v) & \text{for } v_1 \geq 1, v_2 \geq 1. \end{cases} \quad (8.8)$$

The practical meaning of this result is that (for $v_1 \geq 1$, $v_2 \geq 1$) the actual $\| \cdot \|_2$ -error reduction per two-grid step is described by ρ^* if the error of the very first approximation is an element of $\tilde{\mathbb{G}}(\Omega_h)$. Such a first approximation, however, can easily be obtained by applying one RB step to any first approximation. (In fact, applying only one "black" partial step is sufficient.)

We have already mentioned before that $v=2$ is the optimal choice for v . By (8.8) we see that one should use $v_1=v_2=1$.

A somewhat more essential modification of the original algorithm yields the following statement on the energy norm.

Supplement 8.3: If, in contrast to (8.3), the two-grid method is arranged in a "symmetric way", namely if

$$M_h^{2h} = S_h^{(2)} (I_h - I_{2h}^h L_{2h}^{-1} I_h^{2h} L_h) S_h^{(1)} \quad (8.9)$$

where $S_h^{(1)}$ denotes any product of partial step operators S_h^{red} , S_h^{black} and $S_h^{(2)}$ denotes the corresponding product taken in the reversed order, then we obtain

$$\| M_h^{2h} \|_E = \rho(M_h^{2h}). \quad (8.10)$$

In the following, we give the main steps of the proof of Theorem 8.1. Several technical details of the rather involved considerations have been omitted.

Proof of Theorem 8.1: Let $h=1/N$ (N even, $N \geq 4$). Without loss of generality we assume $v_1=0$, $v_2=v$. With \tilde{M}_h^{2h} as defined in (8.7), we then have

$$\rho(M_h^{2h}) = \rho(\tilde{M}_h^{2h}).$$

Because of

$$\tilde{M}_h^{2h} : \tilde{E}_{h,\theta}^{2h} \rightarrow \tilde{E}_{h,\theta}^{2h} \quad (\theta \in T_h^{2h}) \quad \text{with} \quad \tilde{E}_{h,\theta}^{2h} := S_h(E_{h,\theta}^{2h}), \quad S_h = S_h^{\text{black}} S_h^{\text{red}},$$

we need to consider only the restrictions $\tilde{M}_{h,\theta}^{2h}$ of \tilde{M}_h^{2h} to $\tilde{E}_{h,\theta}^{2h}$:

$$\rho(\tilde{M}_h^{2h}) = \max \{ \rho(\tilde{M}_{h,\theta}^{2h}) : \theta \in T_h^{2h} \}.$$

Let us now assume a fixed $\theta \in T_h^{2h}$, $|\theta| < \pi/2$ to be given. (Omitting the case $|\theta| = \pi/2$, does not influence the result.) From (7.44) we see that $\tilde{E}_{h,\theta}^{2h}$ is of dimension 2 and is spanned by $\psi_1, \psi_2 \in \mathcal{G}(\Omega_h)$:

$$\psi_1(x) := (1+a)\varphi(\theta^{00}, x) + (1-a)\varphi(\theta^{11}, x), \quad \psi_2(x) := (1+b)\varphi(\theta^{10}, x) + (1-b)\varphi(\theta^{01}, x) \quad (x \in \Omega_h). \quad (8.11)$$

Here we use the abbreviations

$$a := 1-\xi-\eta, \quad b := \xi-\eta; \quad \xi := \sin^2(\theta_1/2), \quad \eta := \sin^2(\theta_2/2). \quad (8.12)$$

The matrix representation of $\tilde{M}_{h,\theta}^{2h}$ with respect to (8.11) can be shown to be

$$\tilde{M}_{h,\theta}^{2h} \triangleq M(a,b;v) := \begin{pmatrix} m_{ij}(a,b;v) \end{pmatrix}_{2,2}$$

where

$$m_{11}(a,b;v) = \frac{a^{2v}}{2} (1-a^2 - \frac{2a^2 b^2}{1-a^2-b^2}), \quad m_{22}(a,b;v) := m_{11}(b,a;v),$$

$$m_{12}(a,b;v) = -\frac{a^{2v}}{2} (1+b^2 + \frac{2a^2 b^2}{1-a^2-b^2}), \quad m_{21}(a,b;v) := m_{12}(b,a;v).$$

Allowing ϑ to range continuously in $(0, \pi/2)$, we obtain (cf. (8.12))

$$\rho^*(v) = \sup \{F(a,b;v) : 0 < a < 1, 0 \leq b < \min \{a, 1-a\}\} \quad (8.13)$$

with $F(a,b;v) := \rho(M(a,b;v)) (= \rho(\tilde{M}_{h,\vartheta}^{2h}))$. Thus we have to compute the supremum of the function F with respect to (a,b) . (This function is plotted in Figure 8.1 with respect to the coordinates $0 \leq \xi, \eta \leq 1/2$.)

We first note that

$$m_{1,2} m_{2,1} \geq 0, \quad m_{11} \geq 0, \quad m_{22} \geq 0.$$

In particular, the eigenvalues of $M(a,b;v)$ are real for all a,b,v and we obtain

$$F(a,b;v) = (T + \sqrt{T^2 - 4D}) / 2$$

with

$$T = T(a,b;v) = m_{11} + m_{22}, \quad D = D(a,b;v) = m_{11}m_{22} - m_{12}m_{21}$$

As the behavior of $F(a,b;v)$ is different for $v=1$ and $v>1$ (cf. Figure 8.1), we treat these cases separately.

If $v=1$: Using polar coordinates $a=r\cos(\vartheta)$, $b=r\sin(\vartheta)$, one computes

$$F(a,b;1) = (r^2(1-r^2) + r^4 \sin^2(2\vartheta))/2.$$

The right hand side is monotonically increasing with respect to ϑ ($0 \leq \vartheta \leq \pi/4$). Thus F achieves its maximum at the upper boundary of the (a,b) -range, i.e. for either $b=a$, $a \leq 1/2$ or $b=1-a$, $a \geq 1/2$. Because $F(a,a)=a^2$ and $F(a,1-a)=a(1-a)$, F achieves its maximum exactly for $a=b=1/2$ (corresponding to $\xi=1/2$, $\eta=0$ or $\theta_1=\pi/2$, $\theta_2=0$). We obtain $\rho^*(1)=1/4$.

If $v=2$: For this case the use of the coordinates r and $t:=\sin^2(2\vartheta)$ turns out to be suitable. Note that t has the ranges

$$\begin{aligned} 0 \leq t \leq 1 &\quad \text{if} \quad 0 \leq r \leq 1/\sqrt{2}, \\ 0 \leq t \leq (1/r^2 - 1)^2 &\quad \text{if} \quad 1/\sqrt{2} \leq r \leq 1. \end{aligned}$$

Some technical calculations yield

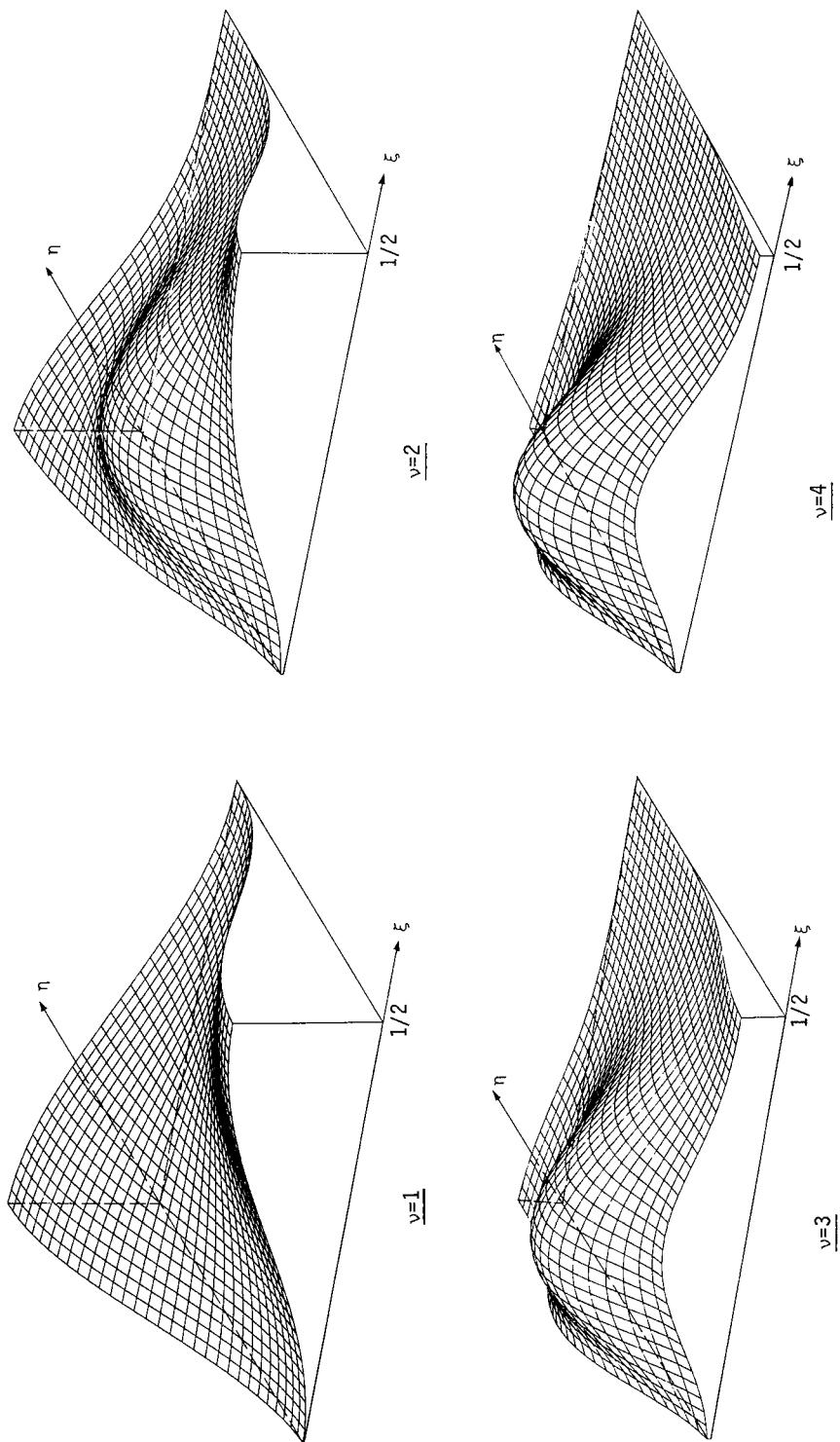


Figure 8.1: F in (8.13) as a function of ξ, η ($0 \leq \xi, \eta \leq 1/2$). The maximum of F gives ρ^* for the two-grid method considered in Theorem 8.1 (in particular, $I_h^{2h} : FW$).

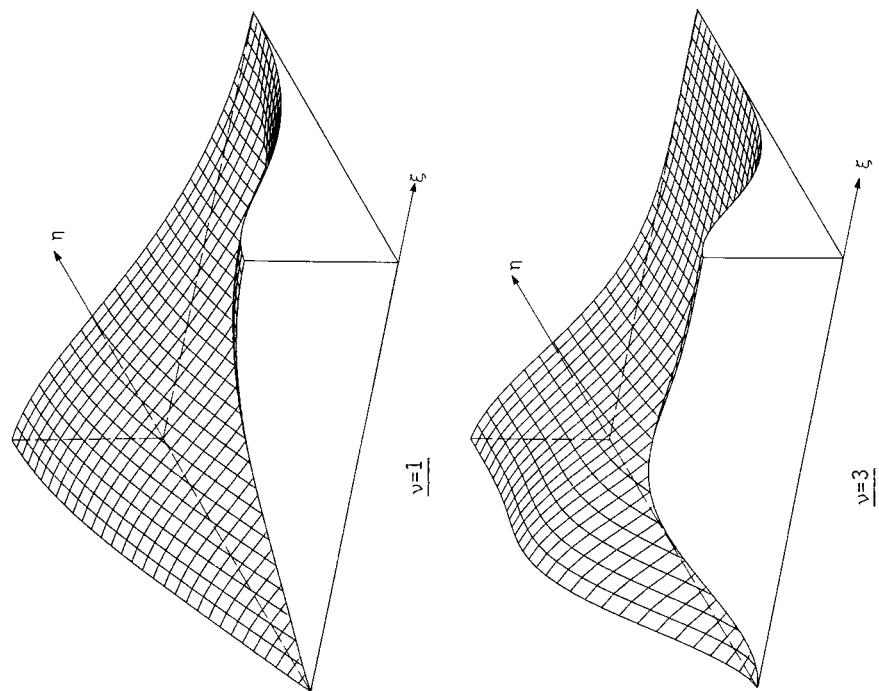
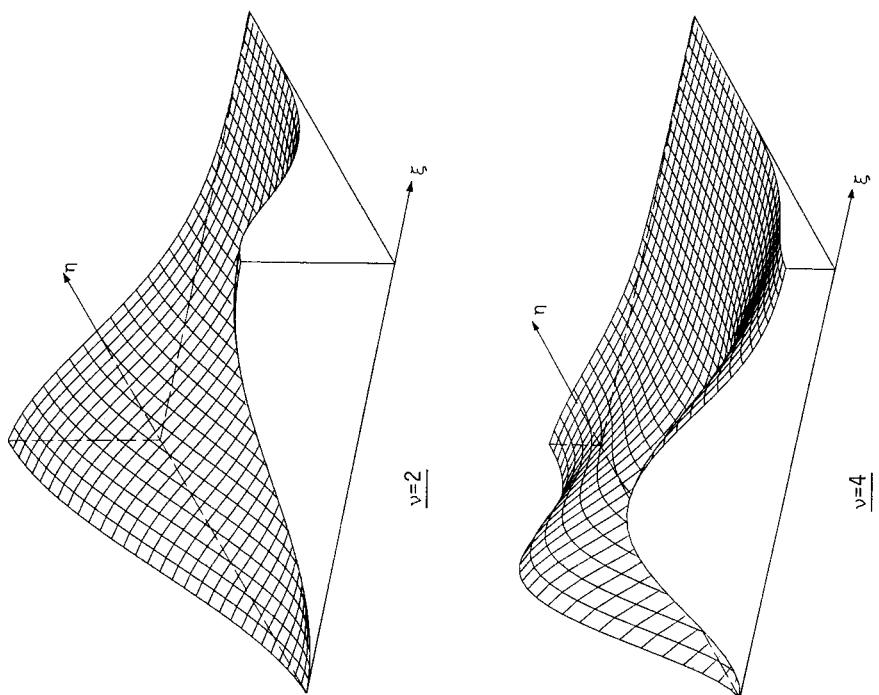


Figure 8.2: Analogous to Figure 8.1. The difference is now that the FW operator is replaced by the HW operator.

$$T(a,b;2) = \frac{r^4(1-r^2)}{2} + \frac{tr^4(tr^4 + 5r^2(1-r^2)-2)}{8(1-r^2)},$$

$$D(a,b;2) = -t^2 r^{10} (1+tr^2/(1-r^2))/32.$$

Now it is not difficult to verify that $\tilde{F}(r,t) := F(a,b;2)$ achieves - for fixed r - its maximum for $t=0$:

$$\max_t \{\tilde{F}(r,t)\} = \tilde{F}(r,0) = r^4(1-r^2)/2. \quad (8.14)$$

Maximizing with respect to r , we obtain $\rho^*(2)=2/27$ and this value is achieved for $b=0$, $a=\sqrt{2/3}$ (corresponding to $\xi=\eta=(1-\sqrt{2/3})/2$).

If $v>2$: This case can be deduced from the case $v=2$. Because $b \leq a$, we obtain

$$0 \leq T(a,b;v) \leq a^{2(v-2)} T(a,b;2), \quad 0 \leq -D(a,b;v) \leq -a^{4(v-2)} D(a,b;2)$$

and thus

$$F(a,b;v) \leq a^{2(v-2)} F(a,b;2) \leq r^{2(v-2)} F(a,b;2).$$

Using (8.14), we obtain

$$F(a,b;v) \leq r^{2(v-2)} \tilde{F}(r,t) \leq r^{2v} (1-r^2)/2.$$

Maximizing with respect to r , we obtain

$$\rho^*(v) \leq \frac{1}{2v} \left(\frac{v}{v+1}\right)^{v+1}.$$

On the other hand, the value on the right hand side is really achieved by $F(a,b;v)$ for $b=0$, $a=\sqrt{v/(v+1)}$ (corresponding to $\xi=\eta=(1-\sqrt{v/(v+1)})/2$). This proves the Theorem. \square

To our knowledge, this is the first time that an explicit analytic expression for ρ^* (not a bound) has been derived for an efficient $(h,2h)$ method. So far, comparable results have been obtained for certain MGR methods only [82]. Because of the red-black coarsening, the analytic situation is simpler there as one has to do only with $(2,2)$ -instead of $(4,4)$ -matrices. This was also the main "trick" in the above proof: By introducing \tilde{M}_h^{2h} instead M_h^{2h} , all calculations were reduced to $(2,2)$ -matrices.

On the proof of Supplement 8.2: This proof is similar to that of Theorem 8.1: We have

$$\tilde{\sigma}_S^*(v_1, v_2) = \sup \{ \| M(a,b;v_1, v_2) \|_S : 0 < a < 1, 0 \leq b < \min\{a, 1-a\} \}$$

where $M(a,b;v_1, v_2)$ denotes the matrix representation of $\tilde{M}_{h,\theta}^{2h}$ with respect to the

normalized basis $\bar{\psi}_1(x) := \psi_1(x)/\|\psi_1\|_2$, $\bar{\psi}_2(x) := \psi_2(x)/\|\psi_2\|_2$ (cf. 8.11)). We do not give the details of the complicated considerations here. We only want to mention that for $v_1 \geq 1$, $v_2 \geq 1$ the above supremum is achieved for $b=0$. For $b=0$, however, M turns out to be a symmetric matrix. This is the reason why $\tilde{\sigma}_S^* = \rho^*$ then.

□

Proof of Supplement 8.3: We first note that both S_h^{red} and S_h^{black} are orthogonal projection operators with respect to the energy inner product (1.8) (cf. [13]). In particular, they are self-adjoint with respect to this inner product meaning that

$$L_h S_h^{\text{red}} = (S_h^{\text{red}})^* L_h, \quad L_h S_h^{\text{black}} = (S_h^{\text{black}})^* L_h.$$

Here "*" denotes the adjoint with respect to the Euclidian inner product (1.7). From this, we obtain (8.10) because

$$(L_h^{1/2} M_h^{2h} L_h^{-1/2})^* = L_h^{1/2} M_h^{2h} L_h^{-1/2}.$$

□

Both results (8.8) and (8.10) can be used to obtain norm estimates for complete cycles based on the respective two-grid methods. Compared to the general estimation in Lemma 4.3, we have the advantage here that the spectral radius ρ^* can be used instead of a norm σ^* . For both cases, however, the estimate used in Lemma 4.3 has to be modified slightly. For (8.10) it is immediately clear from the definition of M_h^{2h} in (8.9) how this modification has to be done. If one wants to apply (8.8), one has to guarantee that the first approximation used on the coarser grids (after every fine-to-coarse transfer) is such that the $\|\cdot\|_2$ -error reduction can be estimated by ρ^* . (This is not true if the zero grid function is used as first approximation.) It is sufficient, however, to modify the recursion of MG cycle (cf. (4.10)) in the following way: Apply one extra "black" partial relaxation step after each fine-to-coarse transfer. This means that the MG iteration operator is now - different from (4.10) - recursively defined by

$$M_{k+1} := M_{k+1}^k + A_k^{k+1} M_k^\gamma S_k^{\text{black}} A_{k+1}^k \quad (k = 1, 2, \dots, \ell-1) \quad (8.15)$$

with two-grid operators corresponding to (8.3). We obtain the

Corollary 8.4: Let $\gamma=2$ and $v_1 \geq 1$, $v_2 \geq 1$. Then one obtains the following bound on $\rho(M_\ell)$ for M_ℓ as defined in (8.15)

$$\rho(M_\ell(v_1, v_2)) \leq \|\tilde{M}_\ell(v_1, v_2)\|_S \leq \sqrt{(1-4\rho^*(v))}/2 \quad (\ell = 1, 2, \dots). \quad (8.16)$$

Here \tilde{M}_ℓ denotes the restriction of M_ℓ to $\tilde{G}(\Omega_\ell)$. In particular, for $v_1=v_2=1$

$$\rho(M_\ell) < 0.081 \quad (\ell = 1, 2, \dots).$$

Proof: The estimate in (8.16) follows by a recursive estimation analogous to the one used in Corollary 4.4 but now applied to (8.15). One only has to observe that $\|B_k^{k+1}\|_S \cdot \|B_{k+1}^k\|_S \leq 1$ ($k=1, 2, \dots$) where

$$B_k^{k+1} := A_k^{k+1} \Big|_{\tilde{\mathcal{G}}(\Omega_k)}, \quad B_{k+1}^k := S_k^{\text{black}} A_{k+1}^k \Big|_{\tilde{\mathcal{G}}(\Omega_{k+1})}.$$

This estimate can be verified by use of the Fourier representations of B_k^{k+1} and B_{k+1}^k .

□

Remarks: (1) If we replace the FW operator used above by the operator of *half weighting* (HW), i.e.

$$I_h^{2h} \triangleq \frac{1}{8} \begin{bmatrix} 1 & 1 \\ 1 & 4 & 1 \\ 1 & & 1 \end{bmatrix}_h^{2h}, \quad (8.17)$$

we obtain a method which is still more efficient than the one using FW. For the corresponding two-grid operator M_h^{2h} , we can make a similar analysis with respect to ρ^* and σ_S^* as above. We do not give the details of the corresponding analysis here, but show the most important values in Table 8.1 of the next section. (See also Figure 8.2: This Figure corresponds to Figure 8.1 and it shows how the function F in (8.13) changes, if the FW operator used there is replaced by the HW operator.) The FORTRAN program listed in the appendix is the MG version corresponding to this two-grid method.

(2) In contrast to the HW operator, the operator of *straight injection* (INJ, see (3.41)) does not lead to a reasonable two-grid method here. This follows immediately from the observation that the defect after one RB step is zero at all black points and nonzero otherwise.

8.2 Further results for Poisson's equation

In this section, we give some more $(h, 2h)$ -results on Poisson's equation with Dirichlet boundary conditions on rectangular domains. As before, we restrict ourselves to the case of square grids $h=h_{x_1}=h_{x_2}$. For the case of non-square grids, see Section 8.3.

The first Table 8.1a recalls results on μ^* and ρ^* already given in Sections 7.5 and 8.1. In addition, corresponding convergence factors are given for the case

that the FW operator (3.3) is replaced by the HW operator (8.17). One can see that the use of the HW operator (for $v \geq 3$) leads to a method which is both more rapidly convergent and cheaper. In the table, the operation count is given in terms of W_h^{2h}/W_h (cf. Section 4.4). (Details on the operation count of the single MG components are given in Table 8.1c.) Using the efficiency measure (4.26), (4.27) (and counting additions and multiplications weighted equally), we compute from Table 8.1a that the optimal values v for FW and HW are given by $v=2$ and $v=3$, respectively.

Remark: Note that after one RB relaxation step the defect is zero at all black points. This means that the fine-to-coarse transfers become fairly simple. In particular, the application of the HW operator simply means that residuals are transferred from the finer to the coarser grid by multiplying them by a factor of 0.5. (This kind of transfer operator is also called *half injection* (HI) in the following.) Similarly, the performance of the coarse-to-fine transfer is simplified if it is followed by one step of RB relaxation: interpolation then has to be carried out explicitly only at black points.

Table 8.1b shows the quantities σ_S^* , σ_E^* and σ_d^* for the respective methods. In order to obtain values less than one for all these quantities, we see that one should choose neither $v_1=0$ nor $v_2=0$. As for the optimal v -values mentioned above, one should choose $v_1=1$, $v_2=1$ and $v_1=2$, $v_2=1$ in case of FW and HW, respectively, in order to obtain the smallest norm values.

Remark: For $v_1=0$ and HW, we see from the table that $\sigma_S^*=\sigma_E^*=\infty$. The reason for this can easily be seen by a similar consideration as for Jacobi's method in connection with the INJ operator (cf. (3.43)). In contrast to Jacobi's method, this effect vanishes here if $v_1 \geq 1$. A similar behavior does not occur in connection with the FW restriction operator; this is due to the fact that the FW operator acts like a "filter": High frequencies are essentially damped by the FW operator, more precisely, we have for $\alpha \neq (0,0)$ (cf. (7.33), (7.34))

$$q(\theta^\alpha) = 0(|\theta|)^2 \quad (|\theta| \rightarrow 0). \quad (8.18)$$

		I_h^{2h} : FW			I_h^{2h} : HW		
v	$(\mu^*)^v$	ρ^*	# Add	# Mult	ρ^*	# Add	# Mult
1	0.250	0.250	6.75	2.25	0.500	5.5	1.75
2	0.063	0.074	9.75	3.25	0.125	8.5	2.75
3	0.034	0.053	12.75	4.25	0.034	11.5	3.75
4	0.025	0.041	15.75	5.25	0.025	14.5	4.75

Table 8.1a: μ^* , ρ^* and computational work W_h^{2h}/W_h in case of smoothing by RB relaxation (for 5-point Laplace discretization)

(v_1, v_2)	$I_h^{2h} : FW$			$I_h^{2h} : HW$		
	σ_S^*	σ_E^*	σ_d^*	σ_S^*	σ_E^*	σ_d^*
(1,0)	0.559	0.500	1.414	0.707	0.514	1.414
(0,1)	1.414	0.500	0.559	∞	∞	1.000
(2,0)	0.200	0.285	1.414	0.191	0.300	1.414
(1,1)	0.141	0.125	0.141	0.707	0.257	0.280
(0,2)	1.414	0.285	0.212	∞	∞	1.000
(3,0)	0.137	0.226	1.414	0.115	0.233	1.414
(2,1)	0.081	0.062	0.081	0.070	0.064	0.096
(1,2)	0.081	0.062	0.081	0.707	0.165	0.108
(0,3)	1.414	0.226	0.144	∞	∞	1.000
(4,0)	0.105	0.193	1.414	0.082	0.197	1.414
(3,1)	0.062	0.046	0.062	0.039	0.036	0.063
(2,2)	0.062	0.046	0.062	0.038	0.029	0.037
(1,3)	0.062	0.046	0.062	0.707	0.131	0.073
(0,4)	1.414	0.193	0.109	∞	∞	1.000

Table 8.1b: Norm values corresponding to the method in Table 8.1a

MG component	# Add	# Mult
one RB step	3	1
I_h^{2h}/FW	2.75	0.75
I_h^{2h}/HW	1.5	0.25
I_{2h}^h (if followed by RB step)	1	0.5

Table 8.1c: Operation count for the individual MG components used in Tables 8.1a and 8.1b (number of operations per point of Ω_h). The numbers given for I_h^{2h} and I_{2h}^h include the work needed for the computation of the defect and adding the correction, respectively.

As in the case of Jacobi ω -relaxation, one could try to introduce one (or more) parameters into the process of RB relaxation. In contrast to Jacobi's method, however, this results only in a non-significant improvement of the quantities in Table 8.1. Taking also work into account, the corresponding methods are even less efficient: $\omega=1$ is indeed the best value here. Also see Figure 9.1, where the dependence on ω is shown for the case of Gauß-Seidel relaxation with lexicographic ordering of the grid points. The dependence shown there is typical for RB relaxation, also.

We point out once more that for Jacobi's method the use of a parameter is unavoidable to ensure good smoothing. This is a typical feature of Jacobi relaxation which makes it inefficient in variable coefficient cases. But even if the optimal parameter is known, Jacobi's method usually is considerably less efficient in smoothing than, e.g., RB relaxation (cf. [25], Section 3.5).

Remark: For the Helmholtz operator

$$L_h^\Omega u = -\Delta u + cu \quad (8.19)$$

one obtains - for non-negative c - similar results as shown in Table 8.1 (for the methods considered there). For any fixed h , all two-grid quantities shown become even better for increasing c . For $c < 0$, the above results still hold if the coarsest mesh size used (h^*) is assumed to be sufficiently fine.

Table 8.2 refers to the 9-point "Mehrstellen"-discretization

$$L_h^\Omega \triangleq \frac{1}{h^2} \begin{bmatrix} -1 & -4 & -1 \\ -4 & 20 & -4 \\ -1 & -4 & -1 \end{bmatrix}_h. \quad (8.20)$$

Again, we use I_{2h}^h defined by linear interpolation and we define L_{2h} in the same way as L_h . We compare RB with Jacobi (JAC) and "four-colour" relaxation (FC, see Remark (4) in Section 7.3). As before we have to use a parameter ω for Jacobi relaxation. In Table 8.2 we used $\omega=0.5$ and $\omega=10/11$ (which is optimal with respect to the smoothing factor). As for the fine-to-coarse transfer, we compare several operators which are meaningful for the respective relaxation methods.

Table 8.2 shows only two choices of v_1 and v_2 , namely $v_1=v_2=1$ and $v_1=2, v_2=1$ (other choices lead to less efficient methods). The best method considered is the one using FC relaxation followed by the one using RB relaxation (and FW restriction). As in the 5-point discretization, Jacobi's method leads to less efficient algorithms, even with the parameter ω chosen for optimal smoothing. We point out that all norm values σ_S^* and σ_E^* are infinite except for FW. The "Mehrstellen" discretization and corresponding two-grid methods have been studied systematically in [12].

Remark: Instead of using the 9-point operator for L_h and L_{2h} , one could have equally well used the usual 5-point operator on the coarser grid. This would not change the results given in Table 8.2 considerably.

Relax	$(\mu^*)^\nu$	I_h^{2h}	ρ^*	σ_S^*	σ_E^*	σ_d^*
RB	0.040	FW	0.066	0.205	0.078	0.205
		HW	0.152	∞	∞	0.205
		HI	0.241	∞	∞	0.317
FC	0.048	FW	0.039	0.185	0.069	0.185
JAC ($\omega=0.5$)	0.490	FW	0.490	0.490	0.490	0.490
		HW	0.490	∞	∞	0.490
		INJ	0.490	∞	∞	0.648
JAC ($\omega=10/11$)	0.207	FW	0.207	0.207	0.207	0.207
		HW	0.207	∞	∞	0.207
		INJ	0.207	∞	∞	0.515

Table 8.2a: Results for the 9-point "Mehrstellen" operator and $(v_1, v_2) = (1,1)$

Relax	$(\mu^*)^\nu$	I_h^{2h}	ρ^*	σ_S^*	σ_E^*	σ_d^*
RB	0.024	FW	0.036	0.052	0.048	0.202
		HW	0.063	∞	∞	0.202
		HI	0.144	∞	∞	0.273
FC	0.030	FW	0.023	0.036	0.037	0.181
JAC ($\omega=0.5$)	0.343	FW	0.343	0.343	0.343	0.405
		HW	0.343	∞	∞	0.403
		INJ	0.343	∞	∞	0.414
JAC ($\omega=10/11$)	0.094	FW	0.101	0.101	0.106	0.121
		HW	0.094	∞	∞	0.104
		INJ	0.094	∞	∞	0.228

Table 8.2b: Analogous to Table 8.2a for $(v_1, v_2) = (2,1)$

8.3 Results for the anisotropic model equation

We now consider the anisotropic model equation

$$L^\Omega u = -\varepsilon u_{x_1 x_1} - u_{x_2 x_2} = f^\Omega(x) \quad (x \in \Omega) \quad (8.21)$$

with Dirichlet boundary conditions on a rectangular domain Ω . We assume this problem to be discretized by the 5-point operator

$$L_h^{\Omega} \triangleq \frac{1}{h^2} \begin{bmatrix} -1 & & \\ -\epsilon & 2(1+\epsilon) & -\epsilon \\ & -1 & \end{bmatrix}_h \quad (8.22)$$

on a square grid $h=h_{x_1}=h_{x_2}$.

As pointed out in Section 7.5, pointwise relaxation methods - if applied in connection with standard coarsening - have very unsatisfactory smoothing properties for $\epsilon \gg 1$ or $0 < \epsilon \ll 1$. This deficiency is fully reflected by the two-grid convergence properties. We have listed some of the corresponding quantities in Table 8.3 for the case of RB relaxation with $\omega=1$, $v_1=2$, $v_2=1$. Note that the half weighting operator yields unbounded norms σ_S^* and σ_E^* as soon as $\epsilon \neq 1$. This shows the special role of the half weighting operator in connection with model problem (P).

ϵ	$(\mu^*)^\nu$	I_h^{2h} : FW			I_h^{2h} : HW		
		ρ^*	σ_S^*	σ_E^*	ρ^*	σ_S^*	σ_E^*
1.0	0.034	0.053	0.081	0.062	0.034	0.070	0.064
0.5	0.088	0.088	0.158	0.134	0.132	∞	∞
0.1	0.564	0.564	0.849	0.621	0.621	∞	∞
0.01	0.942	0.942	1.829	0.951	0.951	∞	∞

Table 8.3: Results for the anisotropic model operator using RB relaxation
 $(I_h^{2h} : \text{linear interpolation}; (v_1, v_2) = (2, 1))$

According to our considerations on smoothing factors in Section 7.5, ZEBRA relaxation (with $\omega=1$) has very good smoothing properties for problem (8.22) if applied in the "correct" direction. Table 8.4a gives results for the case of x_2 -ZEBRA relaxation (using the FW restriction operator). This direction of lines is correct for $\epsilon \leq 1$. In full accordance with the prediction obtained from its smoothing factor, x_2 -ZEBRA does not yield a good method if $\epsilon \gg 1$. In this case x_1 -ZEBRA would be appropriate.

$(v_1, v_2) = (1, 0)$				$(v_1, v_2) = (1, 1)$				
ϵ	$(\mu^*)^\vee$	ρ^*	σ_S^*	σ_E^*	$(\mu^*)^\vee$	ρ^*	σ_S^*	σ_E^*
1000	0.998	0.998	1.99	0.999	0.996	0.996	1.99	0.997
100	0.980	0.980	1.94	0.990	0.961	0.961	1.903	0.971
10	0.826	0.826	1.53	0.909	0.683	0.683	1.316	0.751
2	0.444	0.444	0.801	0.666	0.198	0.198	0.548	0.296
1	0.250	0.250	0.559	0.500	0.063	0.063	0.317	0.125
0.5	0.125	0.112	0.351	0.368	0.053	0.028	0.170	0.049
0.1	0.125	0.111	0.181	0.289	0.053	0.047	0.082	0.065
0.01	0.125	0.123	0.177	0.274	0.053	0.052	0.089	0.073
0.001	0.125	0.124	0.175	0.272	0.053	0.053	0.090	0.074
w_h^{2h}/w_h	#Add: 8.25, #Mult: 4.5			#Add: 12.25, #Mult: 7.5				

Table 8.4a: Results for the anisotropic model operator using x_2 -ZEBRA relaxation
 $(I_h^{2h}: \text{FW restriction}, I_{2h}^h: \text{linear interpolation})$

In order to get rid of the above dependence on the size of ϵ , we can use alternating ZEBRA relaxation. Corresponding results are given in Table 8.4b. Here, one smoothing step is arranged in the following "symmetric" way

$$S_h = S_h^{x_1-\text{odd}} \cdot S_h^{x_1-\text{even}} S_h^{x_2-\text{even}} S_h^{x_2-\text{odd}}. \quad (8.23)$$

This arrangement of one smoothing step is more favourable than others: in particular, it leads to values of ρ^* which are symmetric with respect to ϵ and $1/\epsilon$.

Table 8.4b clearly shows that the convergence properties of the $(h, 2h)$ -method using alternating ZEBRA relaxation are very good independent of the size of ϵ . If $\epsilon \rightarrow 0$ or $\epsilon \rightarrow \infty$, the asymptotic convergence behavior becomes the same as that of ZEBRA relaxation with lines in the correct direction. This is clear heuristically, as only the "correct" partial steps contained in one alternating ZEBRA step are really "active" then.

ϵ	$(v_1, v_2) = (1, 0)$				$(v_1, v_2) = (1, 1)$			
	$(\mu^*)^\nu$	ρ^*	σ_S^*	σ_E^*	$(\mu^*)^\nu$	ρ^*	σ_S^*	σ_E^*
1000	0.125	0.124	0.249	0.273	0.053	0.053	0.126	0.074
100	0.122	0.119	0.242	0.271	0.051	0.051	0.119	0.071
10	0.102	0.082	0.198	0.266	0.041	0.038	0.075	0.049
2	0.061	0.019	0.171	0.279	0.020	0.013	0.020	0.016
1	0.048	0.023	0.198	0.305	0.014	0.009	0.031	0.009
0.5	0.061	0.019	0.242	0.347	0.020	0.013	0.041	0.016
0.1	0.102	0.082	0.337	0.448	0.041	0.038	0.062	0.049
0.01	0.122	0.119	0.379	0.493	0.051	0.051	0.085	0.074
0.001	0.125	0.124	0.384	0.5	0.053	0.053	0.089	0.074
W_h^{2h}/M_h	#Add: 12, #Mult: 7.25				#Add: 20, #Mult: 13.25			

Table 8.4b: Analogous to Table 8.4a with alternating ZEBRA instead of x_2 -ZEBRA

MG component	# Add	# Mult
one ZEBRA step	4	3
I_h^{2h} (if preceded by <u>odd</u> partial step):	3.25	1
I_{2h}^h (if followed by <u>odd</u> partial step):	0.75	0.5
I_{2h}^h (if followed by <u>even</u> partial step):	1	0.5

Table 8.4c: Operation count for the individual MG components used in Tables 8.4a and 8.4b (number of operations per point of Ω_h). The numbers given for I_h^{2h} and I_{2h}^h include the work needed for the computation of the defect and adding the correction, respectively.

Remarks: (1) With respect to a relaxation parameter ω , we have a similar behavior as already mentioned for RB relaxation: In general, the use of a parameter $\omega=1$ does not pay.

(2) As for the operation count given in Table 8.4, the work needed for the matrix decompositions (in solving the occurring tridiagonal systems), is not counted: It has to be performed only once and the corresponding amount of operations is only proportional to $\sqrt{M_h}$.

(3) Clearly, with respect to efficiency, one should take into account that one step of alternating ZEBRA relaxation requires about twice the work of one single ZEBRA step. Thus, if we have to do with a fixed constant ϵ , alternating relaxation does not pay. In practice, the same holds if $\epsilon = \epsilon(x)$ with either $\epsilon(x) \geq 1$ or $0 < \epsilon(x) \leq 1$. On the other hand, if $\epsilon(x) \gg 1$ and $0 < \epsilon(x) \ll 1$ in different parts of a given domain, alternating relaxation is needed (for examples, see Section 10.3).

9. Local Fourier analysis and some general theoretical approaches

The model problem analysis gives realistic quantitative results on the convergence behavior of certain multigrid methods. Its main disadvantages is, however, that it can directly be applied only to a rather small class of problems. In this chapter we review some more general theoretical approaches, namely

- local Fourier analysis (Brandt) which can be regarded as an extension of the model problem analysis;
- more abstract convergence theories.

In Section 9.1 we describe the purpose of local Fourier analysis, provide the formal tools needed and point out the connections to the model problem analysis. Section 9.2 gives some concrete results and comparisons. In particular, multigrid methods based on Gauß-Seidel relaxation (with lexicographic ordering of the grid points) and ILU smoothing are considered. As far as general theories are concerned, we will shortly discuss only two of them (Section 9.3).

All considerations in this section directly refer only to two-grid methods of $(h, 2h)$ -type.

9.1 Purpose and formal tools of local Fourier analysis

Local Fourier analysis (or *local mode analysis*) was introduced by Brandt in [16] and developed further in several papers (see [25] and the references given there). This theoretical approach can be regarded from different points of view and has several objectives. We here consider it as a general tool for providing quantitative and realistic insight into the smoothing properties of relaxation methods and the convergence properties of two-grid methods. As for more sophisticated aspects of the local Fourier analysis (FMG mode analysis, connections to h -ellipticity and interior stability etc.) and for extensions (to systems of differential equations etc.), see Brandt [25] and the references cited there.

As mentioned already in Section 7.6, there are close connections between the local Fourier analysis and the model problem analysis as presented in Chapter 7 and 8. First of all, local mode analysis can, in a sense, be regarded as an analysis for special model problems, namely those with periodic boundary conditions (on rectangular domains). On the other hand, all results obtained by the model problem analysis in Chapters 7 and 8 can be derived by means of local Fourier analysis also. This application of local results to "global" situations is possible since the model problems considered were required to satisfy certain essential assumptions. Roughly

speaking, these assumptions were: the "symmetry" of all grid operators occurring in terms of their corresponding difference stars (cf. Lemma 7.1) as well as certain "continuation properties" for the Ω_h -grid functions occurring with respect to the infinite grid G_h (9.1). The latter assumption has been used in Chapter 7 only implicitly; the three types of boundary conditions - Dirichlet, Neumann and periodic - are associated with an antisymmetric, symmetric and periodic continuation of grid functions, respectively. An analysis of model problems derived from the concept of local Fourier analysis has been sketched in [98].

The more important objective of local Fourier analysis is, however, to give approximative predictions or estimations of μ^* , ρ^* etc. even in those cases for which a rigorous determination of these quantities is very difficult (or practically impossible). The "local" character of this theoretical approach consists in the following: A given discrete operator L_h with variable coefficients (on a regular grid) is considered in the neighborhood of a fixed $x_0 \in \Omega_h$. Here L_h is replaced by a formal discrete operator \mathcal{L}_h with constant coefficients (obtained by "freezing" those of L_h at x_0). The resulting operator \mathcal{L}_h is then defined on the infinite grid

$$G_h := \{x = \kappa \cdot h : \kappa \in \mathbb{Z}^2\} \quad \text{with} \quad \kappa \cdot h = (\kappa_1 h_{x_1}, \kappa_2 h_{x_2}). \quad (9.1)$$

As domain of \mathcal{L}_h we consider the (complex) linear space \mathcal{E}_h spanned by all frequencies

$$\varphi(\theta, x) := e^{i\theta x/h} \quad (x \in G_h) \quad \text{for} \quad \theta \in \mathbb{R}^2, \quad -\pi < \theta \leq \pi. \quad (9.2)$$

Here the inequality has to be understood with respect to θ_1 and θ_2 , and x/h is defined by $x/h := (x_1/h_{x_1}, x_2/h_{x_2})$. Obviously, for any $\theta \in \mathbb{R}^2$, $e^{i\theta x/h}$ can be identified with one of the basis elements in (9.2) because

$$e^{i\theta x/h} = e^{i\theta' x/h} \quad (x \in G_h) \quad \text{if} \quad \theta = \theta' \pmod{2\pi}. \quad (9.3)$$

Note that \mathcal{E}_h is a space with a non-denumerable basis. This is the main formal difference compared to the model problem analysis where the discrete operators were defined on finite dimensional spaces ($\theta \in T_h$). The use of infinite-dimensional spaces and operators (with θ varying continuously in $-\pi < \theta \leq \pi$) gives some technical simplifications in the analysis. Clearly, we are, after all, only interested in two-grid methods for finite-dimensional discrete operators, in practice.

As with \mathcal{L}_h , we replace all other operators by the corresponding infinite-grid operators. We denote them by

$$\mathcal{S}_h, \mathcal{T}_h, \mathcal{T}_h^{2h}, \mathcal{L}_{2h}, \mathcal{T}_{2h}^h, \mathcal{K}_h^{2h}, \mathcal{M}_h^{2h}. \quad (9.4)$$

(For simplicity, we restrict ourselves to the case of standard coarsening. All other types of coarsenings considered in Chapter 7 can be treated analogously however.)

On the coarser grid G_{2h} , the space \mathcal{E}_{2h} is defined analogously to \mathcal{E}_h with h replaced by $2h$. By \mathcal{L}_h and \mathcal{M}_h^{2h} , both an "ideal" difference operator and an "ideal" two-grid operator are defined. In particular, boundary effects are not taken into account by considering these operators. This ideal situation can be analyzed in much the same way as was described in Chapter 7, if complex Fourier frequencies (9.2) are used instead of sine functions (7.4). In the following, we are going to list the necessary tools only briefly without going into details.

Let \mathcal{L}_h be any difference operator on G_h described by a general difference star (1.14):

$$\mathcal{L}_h w_h(x) := \sum_{\kappa \in V} s_\kappa w_h(x + \kappa \cdot h) \quad (x \in G_h). \quad (9.5)$$

Obviously, all basis functions (9.2) are eigenfunctions of \mathcal{L}_h :

$$\mathcal{L}_h \varphi(\theta, x) = \lambda(\theta, h) \varphi(\theta, x) \quad (-\pi < \theta \leq \pi), \quad \lambda(\theta, h) = \sum_{\kappa \in V} s_\kappa e^{i\theta\kappa}. \quad (9.6)$$

In direct analogy to Section 7.2, we define low and high frequencies (with respect to the coarser grid G_{2h}):

$$\varphi(\theta, \cdot) \text{ is called } \begin{cases} \text{low frequency,} & \text{if } -\pi/2 < \theta \leq \pi/2 \\ \text{high frequency,} & \text{otherwise.} \end{cases} \quad (9.7)$$

Corresponding to this definition, the space \mathcal{E}_h is split into an orthogonal sum

$$\mathcal{E}_h = \mathcal{E}_h^{\text{low}} \oplus \mathcal{E}_h^{\text{high}}. \quad (9.8)$$

Here "orthogonal" is meant with respect to the inner product

$$(v_h, w_h) := \lim_{n \rightarrow \infty} \frac{1}{4n^2} \sum_{|\kappa| \leq n} v_h(\kappa \cdot h) \overline{w_h(\kappa \cdot h)}. \quad (9.9)$$

The basis functions (9.2) of \mathcal{E}_h are orthonormal with respect to (9.9). Again, there is a natural identification of $\mathcal{E}_h^{\text{low}}$ and the space \mathcal{E}_{2h} . This becomes obvious if \mathcal{E}_{2h} is written in the form

$$\mathcal{E}_{2h} = \text{span} \{ \Phi(\theta, \cdot) : -\pi/2 < \theta \leq \pi/2 \}, \quad \Phi(\theta, x) := \varphi(\theta, x) \quad (x \in G_{2h}). \quad (9.10)$$

As in Section 7.2, we define spaces $\mathcal{E}_{h,\theta}$ of $(2h\text{-})$ harmonics to be spanned by just those basis functions of \mathcal{E}_h which coincide on the coarser grid used. Because of

$$\varphi(\theta, x) = \varphi(\theta', x) \quad (x \in G_{2h}) \quad \text{iff} \quad \theta = \theta' \pmod{\pi}, \quad (9.11)$$

we have for all $-\pi/2 < \theta \leq \pi/2$

$$\begin{aligned}\mathcal{E}_{h,\theta} &= \text{span } \{\varphi(\theta', \cdot) : -\pi < \theta' \leq \pi, \theta' \equiv \theta \pmod{\pi}\} \\ &= \text{span } \{\varphi(\theta^\alpha, \cdot) : \alpha = (0,0); (1,1); (1,0); (0,1)\}\end{aligned}\tag{9.12}$$

with

$$\theta^\alpha := \theta - \begin{pmatrix} \alpha_1 \text{sign}(\theta_1) \\ \alpha_2 \text{sign}(\theta_2) \end{pmatrix} \pi, \quad \text{sign}(t) := \begin{cases} 1 & (t > 0) \\ -1 & (t \leq 0). \end{cases}\tag{9.13}$$

The transfer operators \mathcal{T}_h^{2h} , \mathcal{T}_{2h}^h and the coarse-grid difference operator \mathcal{L}_{2h} are assumed to be represented by difference stars with coefficients \hat{t}_k^+ , \hat{t}_k^- and \hat{s}_k , respectively. We obtain for all $-\pi/2 < \theta \leq \pi/2$ and all α , θ^α as defined in (9.12), (9.13) (cf. Lemma 7.1):

$$\begin{aligned}\mathcal{L}_{2h} \Phi(\theta, x) &= \Lambda(\theta, h) \Phi(\theta, x), & \Lambda(\theta, h) &= \sum_{k \in V} \hat{s}_k e^{2i\theta k}; \\ \mathcal{T}_h^{2h} \varphi(\theta^\alpha, x) &= q(\theta^\alpha) \varphi(\theta, x), & q(\theta^\alpha) &= \sum_{k \in V} \hat{t}_k e^{i\theta^\alpha k}; \\ \mathcal{T}_{2h}^h \Phi(\theta, h) &= \sum_\alpha p_\alpha(\theta) \varphi(\theta^\alpha, x), & p_\alpha(\theta) &= \frac{1}{4} \sum_{k \in V} \hat{t}_{-k} e^{i\theta^\alpha k}.\end{aligned}\tag{9.14}$$

For the definition of relaxation operators \mathcal{J}_h , we use the following notation (which is a generalization of the one used in Section 7.3). The difference operator \mathcal{L}_h is split into a sum

$$\mathcal{L}_h = \mathcal{L}_h^+ + \mathcal{L}_h^0 + \mathcal{L}_h^-\tag{9.15}$$

defined by a corresponding splitting of the index set V in (9.5) into three disjoint subsets V^+ , V^0 , V^- :

- V^0 : characterizes those grid points where grid values are solved for simultaneously within the relaxation process;
- V^- : characterizes those grid points where previous grid values are used within the relaxation process;
- V^+ : characterizes those grid points where already new grid values are available and are used instead of the previous values.

Using these notations, all common relaxation operators can be described in a formally simple manner. Considering errors v_h and \bar{v}_h before and after one complete or one partial relaxation step, we have (with some relaxation parameter ω)

$$(\mathcal{L}_h^0 + \omega \mathcal{L}_h^+) \bar{v}_h(x) - ((1-\omega) \mathcal{L}_h^0 - \omega \mathcal{L}_h^-) v_h(x) = 0 \quad (x \in \tilde{G}_h) \\ \bar{v}_h(x) = v_h(x) \quad (x \in G_h \setminus \tilde{G}_h). \quad (9.16)$$

Here \tilde{G}_h is either G_h (for complete relaxation steps) or a certain subset of G_h (for partial relaxation steps). Corresponding to the splitting of \mathcal{L}_h , we have a splitting of its eigenvalues $\lambda(\theta, h) = \lambda^+(\theta, h) + \lambda^0(\theta, h) + \lambda^-(\theta, h)$:

$$\lambda^+(\theta, h) = \sum_{\kappa \in V^+} s_\kappa e^{i\theta\kappa}, \quad \lambda^0(\theta, h) = \sum_{\kappa \in V^0} s_\kappa e^{i\theta\kappa}, \quad \lambda^-(\theta, h) = \sum_{\kappa \in V^-} s_\kappa e^{i\theta\kappa}. \quad (9.17)$$

We assume throughout that

$$\lambda^0(\theta, h) + \omega \lambda^+(\theta, h) \neq 0 \quad (-\pi < \theta \leq \pi).$$

The following examples show how common relaxation methods are included in the above concept. As for more examples, see [98], [16].

Examples: (1) Gauß-Seidel point relaxation with lexicographic ordering of the grid points (GS; here: from left to right and from bottom to top) is characterized by $\tilde{G}_h = G_h$ and

$$V^0 = \{(0,0)\}, \quad V^+ = \{\kappa \in V : \kappa_1 < 0 \text{ or } (\kappa_1 = 0 \text{ and } \kappa_2 < 0)\}.$$

(2) Gauß-Seidel line relaxation with line-by-line ordering (line GS; here: lines in x_2 -direction marching from left to right) is characterized by $\tilde{G}_h = G_h$ and

$$V^0 = \{\kappa \in V : \kappa_1 = 0\}, \quad V^+ = \{\kappa \in V : \kappa_1 < 0\}.$$

(3) Jacobi relaxation is characterized by $\tilde{G}_h = G_h$ and

$$V^0 = \{(0,0)\}, \quad V^+ = \emptyset.$$

(4) RB relaxation proceeds in two partial steps. For both steps the splitting of V is the same as in (3). In the first partial step, \tilde{G}_h consists of the "red" and in the second step \tilde{G}_h consists of the "black" points (cf. Table 7.1).

(5) Gauß-Seidel point relaxation with red-black ordering of the grid points (red-black GS) is obtained by choosing \tilde{G}_h as in (4) and

$$V^0 = \{(0,0)\}, \quad V^+ = \{\kappa \in V : \kappa_1 < 0 \text{ or } (\kappa_1 = 0 \text{ and } \kappa_2 < 0)\} \cap \{\kappa \in V : \kappa_1 + \kappa_2 \text{ even}\}$$

(As pointed out before, this method is equivalent to RB relaxation iff \mathcal{L}_h is described by a 5-point star.)

It is quite easy to construct Fourier representations for all common relaxation operators (with respect to the basis (9.2) of \mathcal{E}_h). In particular, if $\tilde{G}_h = G_h$, then the relaxation operator \mathcal{J}_h defined by (9.16) is represented by

$$\mathcal{J}_h \varphi(\theta, x) = A(\theta, h; \omega) \varphi(\theta, x) \quad (-\pi < \theta \leq \pi) \quad (9.18)$$

with

$$A(\theta, h; \omega) := \frac{(1-\omega)\lambda^0(\theta, h) - \omega \lambda^+(\theta, h)}{\lambda^0(\theta, h) + \omega \lambda^+(\theta, h)}. \quad (9.19)$$

For methods like RB or ZEBRA, the $\varphi(\theta, \cdot)$ are no longer eigenfunctions of the respective operators \mathcal{J}_h . However, the spaces $\mathcal{E}_{h,\theta}$ of harmonics are still invariant:

$$\mathcal{J}_h : \mathcal{E}_{h,\theta} \rightarrow \mathcal{E}_{h,\theta} \quad (-\pi/2 < \theta \leq \pi/2) \quad (9.20)$$

This is analogous to the finite-grid situation studied in Section 7.3. In fact, for RB and ZEBRA (on the infinite grid) one obtains formally the same (4,4)-matrix representations (with respect to $\mathcal{E}_{h,\theta}$) as we have shown in Table 7.2. Clearly, θ ranges in $-\pi/2 < \theta \leq \pi/2$ instead of T_h^{2h} . (The "degenerate" cases shown in Table 7.2 do not occur here.) Also the definition of A_α in Table 7.2 has to be understood with respect to all $-\pi/2 < \theta \leq \pi/2$ and with A, θ^α as defined in (9.19), (9.13), respectively.

Remark: In contrast to RB and ZEBRA, there is no comparable equivalence of finite-grid analysis (concerning S_h) and infinite-grid analysis (concerning \mathcal{J}_h) in the cases of, for instance, GS relaxation (1) and line GS relaxation (2). In fact, with respect to the basis (7.4), all frequencies are intermixed by the relaxation operator S_h (leading to a representation of S_h by a full matrix). Therefore these methods cannot be directly analyzed in the framework of the model problem analysis. The reason for the different structure of S_h and \mathcal{J}_h lies in the fact that $V^+ \neq \emptyset$ for (1) and (2) (meaning that there is - so to say - no natural starting point for the relaxation on the infinite grid). For RB and ZEBRA we have $V^+ = \emptyset$; here any point (or line) can be used to start the relaxation. In general, the analyses of S_h and \mathcal{J}_h differ due to algorithmic changes near the boundary. Concerning the high frequencies, however, the boundary influence is not essential. Therefore, the smoothing properties of \mathcal{J}_h should be a good approximation to those of S_h in all cases.

The general definition of a *smoothing factor* given in Section 7.5 carries over to the infinite-grid operators in a straightforward manner. In particular, whenever a matrix representation $\hat{\mathcal{J}}_{h,\theta}$ of \mathcal{J}_h (with respect to $\mathcal{E}_{h,\theta}$) is available, we can compute μ by

$$\mu(h, v) = \sup \{ \sqrt[p]{\rho(Q_{h,\theta}^{2h}, \hat{\mathcal{J}}_{h,\theta}^v)} : -\pi/2 < \theta \leq \pi/2 \} \quad (9.21)$$

where

$$\hat{Q}_{h,\theta}^{2h} = \begin{bmatrix} 0 & & & \\ & 1 & & \\ & & 1 & \\ & & & 1 \end{bmatrix} .$$

(This quantity is denoted by $\bar{\mu}_v$ in Brandt [25].) As usual, we define

$$\mu^*(v) := \sup \{ \mu(h, v) : h \in \mathcal{X}^* \} \quad (9.22)$$

We give exemplary values of $\mu^*(v)$ in the following section. For more examples, see [98], [16].

Let us now consider a two-grid operator for \mathcal{L}_h . For the relaxation operator \mathcal{T}_h used, we assume (9.20) to hold. In order for the two-grid operator to be well-defined, we replace the domain \mathcal{E}_h of \mathcal{L}_h by a slightly shrunk subspace $\tilde{\mathcal{E}}_h$ such that \mathcal{L}_h^{-1} exists and also \mathcal{L}_{2h}^{-1} can be reasonably defined on the coarser grid. We choose

$$\tilde{\mathcal{E}}_h := \mathcal{E}_h \setminus \bigoplus_{\theta \in \Psi} \mathcal{E}_{h,\theta} \quad (9.23)$$

Here the set Ψ contains just those $-\pi/2 < \theta \leq \pi/2$ with

- either $\Lambda(\theta, h) = 0$,
- or $\lambda(\theta', h) = 0$ for some $-\pi < \theta' \leq \pi$ with $\theta' \equiv \theta \pmod{\pi}$.

Clearly, \mathcal{L}_h^{-1} exists on $\tilde{\mathcal{E}}_h$. The transfer operators have the properties (cf. (9.14))

$$\mathcal{T}_h^{2h} : \tilde{\mathcal{E}}_h \rightarrow \tilde{\mathcal{E}}_{2h}, \quad \mathcal{T}_{2h}^h : \tilde{\mathcal{E}}_{2h} \rightarrow \tilde{\mathcal{E}}_h$$

with

$$\tilde{\mathcal{E}}_{2h} := \mathcal{E}_{2h} \setminus \text{span} \{ \Phi(\theta, \cdot) : \theta \in \Psi \}.$$

\mathcal{L}_{2h} , regarded as an operator

$$\mathcal{L}_{2h} : \tilde{\mathcal{E}}_{2h} \rightarrow \tilde{\mathcal{E}}_{2h},$$

is invertable. Thus \mathcal{M}_h^{2h} is well-defined as an operator on $\tilde{\mathcal{E}}_h$. In particular, we have

$$\mathcal{M}_h^{2h} : \mathcal{E}_{h,\theta} \rightarrow \mathcal{E}_{h,\theta} \quad (-\pi/2 < \theta \leq \pi/2, \theta \notin \Psi). \quad (9.24)$$

Using these invariance properties, we can compute

$$\begin{aligned} \rho(h, v) &:= \sup_{\theta} \{ \rho(\hat{Q}_{h,\theta}^{2h}) \}, & \sigma_d(h, v_1, v_2) &:= \sup_{\theta} \{ \| \hat{\mathcal{L}}_{h,\theta} \hat{Q}_{h,\theta}^{2h} \hat{\mathcal{L}}_{h,\theta}^{-1} \|_S \}, \\ \sigma_S(h, v_1, v_2) &:= \sup_{\theta} \{ \| \hat{Q}_{h,\theta}^{2h} \|_S \}, & \sigma_E(h, v_1, v_2) &:= \sup_{\theta} \{ \| \hat{\mathcal{L}}_{h,\theta}^{1/2} \hat{Q}_{h,\theta}^{2h} \hat{\mathcal{L}}_{h,\theta}^{-1/2} \|_S \}, \end{aligned} \quad (9.25)$$

where θ ranges as in (9.24) and $\hat{\mathcal{M}}_{h,\theta}^{2h}$ and $\hat{\mathcal{L}}_{h,\theta}$ denote the matrix representations of \mathcal{M}_h^{2h} and \mathcal{L}_h with respect to $\ell_{h,\theta}$. Clearly, σ_E only makes sense if \mathcal{L}_h is positive definite. In the next section we give exemplary results on ρ^* , σ_S^* , σ_E^* and σ_d^* which are, as usual, defined by taking the supremum of the respective values ρ , σ_S , σ_E , σ_d over the admissible values $h \in \mathcal{K}$.

Remarks: (1) Note that, in general, ρ in (9.25) is not equal to the spectral radius of \mathcal{M}_h^{2h} (with respect to the norm $\|\cdot\|_2$ induced by (9.9)). If \mathcal{M}_h^{2h} is not bounded with respect to $\|\cdot\|_2$, the spectrum of \mathcal{M}_h^{2h} may, for example, be the whole complex plane, whereas ρ may have a finite value. Clearly, this phenomenon cannot occur for finite-dimensional operators, which we are actually interested in (cf. the corresponding remark above). We have allowed θ to range continuously over $-\pi/2 < \theta \leq \pi/2$ only for reasons of technical simplifications. Therefore, ρ as defined in (9.25) is the practically relevant quantity and not the spectral radius of \mathcal{M}_h^{2h} .

(2) If \mathcal{L}_h and \mathcal{L}_{2h} consist of "principal terms" only, more precisely, if for fixed meshsize ratio q^* and some $m \in \mathbb{N}$

$$s_\kappa h_{x_1}^m, \quad \hat{s}_\kappa h_{x_1}^m \quad (\kappa \in \mathbb{W}) \quad (9.26)$$

are independent of h (and if the star coefficients of \mathcal{L}_h^{2h} and \mathcal{L}_{2h} are independent of h) then \mathcal{M}_h^{2h} also does not depend explicitly on h . In particular, we then have

$$\rho = \rho^*, \quad \sigma_S = \sigma_S^*, \quad \sigma_E = \sigma_E^*, \quad \sigma_d = \sigma_d^*. \quad (9.27)$$

Clearly, if \mathcal{L}_h and \mathcal{L}_{2h} do contain lower order terms, for $h \rightarrow 0$ only their principal parts are relevant for the two-grid convergence properties.

9.2 Applications of local Fourier analysis

In this section, we give some results obtained by the application of local Fourier analysis. The main objective here is to treat some methods which are similar to those of Chapter 8 but which cannot be analyzed by the model problem analysis and to compare the corresponding results with those of Chapter 8. As in Chapter 8, we again restrict ourselves to square grids $h=h_{x_1}=h_{x_2}$.

All examples considered satisfy (9.27). The list of applications could easily be extended as the concept of local Fourier analysis is very general. Many more applications can be found in the papers of Brandt (see [25] and the references cited there). For a detailed discussion of operators also containing lower order terms, in particular of singularly perturbed problems, we refer to [12].

Tables 9.1, 9.2 and 9.4 directly correspond to Tables 8.1, 8.2 and 8.4, respectively. We consider the same difference operators and similar MG components as in Chapter 8 (now, of course, on the infinite grid G_h). As in Chapter 8, we define the coarse-grid difference operator \mathcal{L}_{2h}^{2h} in the same way as the fine-grid operator \mathcal{L}_h^{2h} ; the coarse-to-fine grid transfer \mathcal{T}_{2h}^h again is assumed to be given by bilinear interpolation. Differences in the MG components are in the smoothing and the fine-to-coarse transfer components. In particular, we treat GS and line GS relaxation (cf. Examples (1) and (2) in the previous Section). Table 9.5 refers to the equation $-\Delta u = \epsilon u_{x_1 x_2}$. As the (standard) 9-point discretization of this problem leads to a non-symmetric difference star, the analysis of Chapter 7 is not applicable to this problem. We conclude this section with some considerations on ILU smoothing. As the ILU smoothers are not included in the concept of relaxation methods given so far, we give a short description of the (local) ILU Fourier representation.

In Table 9.1, we give results for the 5-point discretization of the Laplace operator. The relaxation method used is GS with relaxation parameter $\omega=1$. As for \mathcal{T}_h^{2h} , we compare the FW restriction operator (3.3) with the simplest transfer operator possible, namely the operator of straight injection (INJ). Of course, we could also have used the HW restriction operator. This operator leads to results somewhere between FW and INJ.

Comparing Table 9.1a with Table 8.1a, one sees that the use of RB relaxation leads to considerably more efficient methods than GS. If one takes the numerical work into account, Table 9.1a shows that the restriction operator INJ is more favourable than FW. The application of the INJ operator has, however, the disadvantage that the spectral and the energy norms of the corresponding local two-grid operators are not bounded (see Table 9.1b). Although we obtain this result only in the framework of infinite-grid analysis, practically the same effect occurs for finite-grid algorithms (if $h>0$). Thus one has to be careful with this method (both if it is used in a complete cycle or in the FMG version). As already observed before, FW can be regarded as more robust, at least from a theoretical point of view.

		$I_h^{2h} : \text{FW}$			$I_h^{2h} : \text{INJ}$		
v	$(\mu^*)^v$	ρ^*	# Add	# Mult	ρ^*	# Add	# Mult
1	0.500	0.400	12	3	0.447	7	2.25
2	0.250	0.193	16	4	0.200	11	3.25
3	0.125	0.119	20	5	0.089	15	4.25
4	0.063	0.084	24	6	0.042	19	5.25

Table 9.1a: μ^* , ρ^* and computational work in case of smoothing by GS relaxation (for 5-point Laplace discretization)

(v_1, v_2)	I_h^{2h} : FW			I_h^{2h} : INJ		
	σ_S^*	σ_E^*	σ_d^*	σ_S^*	σ_E^*	σ_d^*
(1,0)	0.447	0.447	1.000	∞	∞	1.067
(0,1)	1.000	0.447	0.447	∞	∞	1.743
(2,0)	0.208	0.232	1.000	∞	∞	1.006
(1,1)	0.203	0.202	0.203	∞	∞	0.356
(0,2)	1.000	0.232	0.208	∞	∞	1.733
(3,0)	0.128	0.183	1.000	∞	∞	1.001
(2,1)	0.119	0.119	0.131	∞	∞	0.132
(1,2)	0.131	0.119	0.119	∞	∞	0.335
(0,3)	1.000	0.183	0.128	∞	∞	1.732
(4,0)	0.091	0.156	1.000	∞	∞	1.000
(3,1)	0.084	0.084	0.106	∞	∞	0.103
(2,2)	0.084	0.084	0.084	∞	∞	0.112
(1,3)	0.106	0.084	0.084	∞	∞	0.334
(0,4)	1.000	0.156	0.091	∞	∞	1.732

Table 9.1b: Norm values corresponding to the methods in Table 9.1a.

Using a relaxation parameter $\omega=1$ in the methods considered above does not improve the two-grid results. The reason for introducing a parameter in the classical approach to SOR is to speed up the relaxation method as an iterative solver by itself: mainly the reduction of low frequencies is improved by a parameter. The reduction of high frequencies (and by that the smoothing property) usually becomes even worse. Figure 9.1 shows graphically the dependence of μ^* and ρ^* (for FW) on the choice of ω . The optimal value of ω is not exactly 1, but very close to 1. (The gain in convergence speed by using the optimal ω would be only very small and does not pay if the additional work (2 operations per point and smoothing step) is taken into account.)

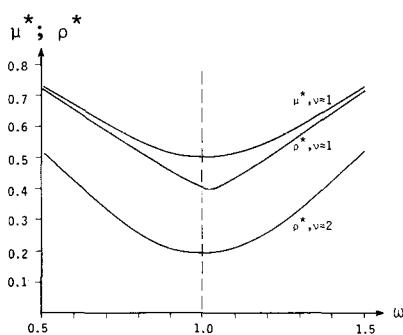


Figure 9.1: μ^* , ρ^* as a function of ω for the method as considered in Table 9.1a (using FW)

Table 9.2 shows two-grid quantities for the 9-point "Mehrstellen" discretization (8.20) using GS and red-black GS (see Example (5) in the previous section) as smoothing methods. Both methods are combined with reasonable fine-to-coarse transfer operators. From the numbers given in Table 9.2, we see that it is advisable to order the grid points in a red-black manner rather than simply lexicographically and use either the HW operator (with $v_1=2, v_2=1$) or the FW operator. Except for FW, we again obtain $\sigma_S^*=\sigma_E^*=\infty$. Compared to the corresponding methods in Table 8.2, we see that red-black GS is slightly preferable to RB. RB has, however, the advantage of being analyzable in the framework of model problem analysis. FC relaxation still yields the most efficient method of all and is also analyzable by the model problem analysis. Many more considerations on the Mehrstellenverfahren, both theoretical and practical, can be found in [12]. In particular, the influence of relaxation parameters is investigated there. Again, the main result is that the use of relaxation parameters in combination with RB, FC, GS and red-black GS does not pay if numerical work is taken into account.

		$(v_1, v_2)=(1,1)$			$(v_1, v_2)=(2,1)$		
Relax	I_h^{2h}	$(\mu^*)^\nu$	ρ^*	σ_S^*	$(\mu^*)^\nu$	ρ^*	σ_S^*
GS	FW		0.134	0.179		0.075	0.076
	HW	0.223	0.141	∞	0.105	0.061	∞
	INJ		0.178	∞		0.075	∞
red-black GS	FW		0.044	0.12		0.029	0.042
	HW	0.060	0.10	∞	0.027	0.029	∞
	HI		0.13	∞		0.097	∞

Table 9.2: Results for the 9-point "Mehrstellen"-operator

Tables 9.4a and 9.4b refer to the anisotropic model operator in (8.22) and contain results obtained by using line GS and alternating line GS as smoothers. They are to be compared with Tables 8.4a and 8.4b, respectively. They demonstrate that the use of ZEBRA and alternating ZEBRA is considerably preferable to line GS and alternating line GS, respectively, both with respect to convergence speed and to numerical work. In fact, the relation of ZEBRA to line GS is here quite similar to that of RB to GS (for the Laplace operator). In particular, the use of relaxation parameters does not improve the methods essentially any further.

ϵ	$(v_1, v_2) = (1, 0)$				$(v_1, v_2) = (1, 1)$			
	$(\mu^*)^\vee$	ρ^*	σ_S^*	σ_E^*	$(\mu^*)^\vee$	ρ^*	σ_S^*	σ_E^*
1000	0.998	0.998	1.41	0.998	0.996	0.996	1.41	0.996
100	0.980	0.980	1.36	0.980	0.961	0.961	1.33	0.961
10	0.833	0.833	1.00	0.833	0.694	0.694	0.878	0.694
2	0.500	0.500	0.500	0.500	0.250	0.250	0.351	0.250
1	0.447	0.333	0.471	0.447	0.200	0.134	0.351	0.200
0.5	0.447	0.333	0.471	0.447	0.200	0.134	0.351	0.200
0.1	0.447	0.333	0.471	0.447	0.200	0.134	0.351	0.200
0.01	0.447	0.333	0.471	0.447	0.200	0.134	0.351	0.200
0.001	0.447	0.333	0.471	0.447	0.200	0.134	0.351	0.200
W_h^{2h}/N_h	# Add: 11.75, # Mult: 5.5				# Add: 15.75, # Mult: 8.5			

Table 9.4a: Results for the anisotropic model operator using x_2 -line GS relaxation
 $(J_h^{2h}$: FW restriction, J_{2h}^h : linear interpolation)

ϵ	$(v_1, v_2) = (1, 0)$				$(v_1, v_2) = (1, 1)$			
	$(\mu^*)^\vee$	ρ^*	σ_S^*	σ_E^*	$(\mu^*)^\vee$	ρ^*	σ_S^*	σ_E^*
1000	0.446	0.332	0.469	0.446	0.199	0.133	0.35	0.199
100	0.438	0.321	0.453	0.438	0.192	0.126	0.337	0.192
10	0.373	0.238	0.373	0.373	0.139	0.088	0.245	0.139
2	0.223	0.111	0.223	0.236	0.050	0.047	0.112	0.055
1	0.149	0.086	0.149	0.183	0.022	0.042	0.067	0.042
0.5	0.223	0.111	0.223	0.236	0.050	0.047	0.112	0.055
0.1	0.373	0.238	0.373	0.373	0.139	0.088	0.245	0.139
0.01	0.438	0.321	0.453	0.438	0.192	0.126	0.337	0.192
0.001	0.446	0.332	0.469	0.446	0.199	0.133	0.35	0.199
W_h^{2h}/N_h	# Add: 15.75, # Mult: 8.5				# Add: 23.75, # Mult: 14.5			

Table 9.4b: As in Table 9.4a with alternating line GS instead of x_2 -line GS

The discrete problems considered so far are model problems in the sense of Section 7.1. This means that appropriate multigrid methods can be constructed for them which allow the application of the analysis of Chapter 7. This is different for the differential operator

$$L^\Omega = -\Delta u - \varepsilon u_{x_1 x_2}, \quad (9.28)$$

since this operator cannot be consistently discretized by a symmetric difference star. In Table 8.5, we show results for the discretization

$$\mathcal{L}_h \stackrel{\Delta}{=} \frac{1}{h^2} \begin{bmatrix} \frac{\varepsilon}{4} & -1 & -\frac{\varepsilon}{4} \\ -1 & 4 & -1 \\ -\frac{\varepsilon}{4} & -1 & \frac{\varepsilon}{4} \end{bmatrix} h \quad (9.29)$$

using local Fourier analysis. For $\varepsilon=0$, we have just the Laplace operator; if $|\varepsilon| \geq 2$ the operator (9.28) is no longer elliptic. We see from the table that - for the values $\varepsilon \neq 0$ shown - RB and GS behave much the same way. In particular, $v=1$ already yields the best values. The efficiency of both methods deteriorates considerably for $|\varepsilon| \rightarrow 2$. To improve the efficiency for this problem, one should take the characteristic unsymmetry of the difference star into account. Also, the use of ZEBRA or alternating ZEBRA relaxation does not lead to considerably more efficient methods as these smoothers are primarily suitable for unsymmetries with respect to the axes. We are not going to discuss the problem of optimal smoothing in more detail here. See, however, the use of ILU-smoothing below.

ε	$(v_1, v_2) = (1, 0)$				$(v_1, v_2) = (1, 1)$			
	$(\mu^*)^v$	ρ^*	σ_S^*	σ_E^*	$(\mu^*)^v$	ρ^*	σ_S^*	σ_E^*
-1.7	0.678	0.623	0.656	0.676	0.460	0.543	0.563	0.546
-1.5	0.651	0.569	0.600	0.620	0.424	0.461	0.474	0.464
-1.0	0.591	0.480	0.514	0.523	0.350	0.325	0.332	0.329
0.0	0.500	0.400	0.447	0.447	0.250	0.193	0.203	0.202
1.0	0.538	0.467	0.557	0.538	0.289	0.260	0.361	0.302
1.5	0.568	0.509	0.675	0.619	0.342	0.420	0.550	0.450
1.7	0.610	0.588	0.760	0.678	0.372	0.516	0.676	0.540

Table 9.5a: Results for the operator (9.29) using GS relaxation (γ_h^{2h} : FW restriction; γ_h^h : linear interpolation)

ϵ	$(v_1, v_2) = (1, 0)$				$(v_1, v_2) = (1, 1)$			
	$(\mu^*)^\vee$	ρ^*	σ_S^*	σ_E^*	$(\mu^*)^\vee$	ρ^*	σ_S^*	σ_E^*
-1.7	0.568	0.596	0.772	0.698	0.324	0.539	0.696	0.561
-1.5	0.520	0.521	0.686	0.637	0.271	0.443	0.576	0.471
-1.0	0.405	0.363	0.559	0.513	0.163	0.226	0.364	0.301
0.0	0.250	0.250	0.559	0.500	0.063	0.074	0.141	0.125
1.0	0.405	0.363	0.559	0.513	0.163	0.266	0.364	0.301
1.5	0.520	0.521	0.686	0.637	0.271	0.443	0.576	0.471
1.7	0.568	0.596	0.772	0.698	0.324	0.539	0.696	0.561

Table 9.5b: Analogous to Table 9.5a with RB instead of GS.

There are smoothing methods which are not contained in the general description of relaxation methods given in Section 9.1. For example, *ILU-smoothing* has attracted attention recently and is supposed to be distinguished by its "robustness" [110],[60]. We shortly describe how ILU-smoothing can be treated in the framework of local Fourier analysis. As for concrete algorithms, see [110].

For a given discrete difference operator \mathcal{L}_h (9.5), an incomplete ILU decomposition is generally described by a splitting

$$\mathcal{L}_h = \mathcal{L}_h^L \mathcal{L}_h^U + \mathcal{L}_h^R \quad (9.30)$$

with certain "lower" and "upper" difference operators \mathcal{L}_h^L , \mathcal{L}_h^U and a "remainder" operator \mathcal{L}_h^R .

Examples: For a given 5-point operator

$$\mathcal{L}_h \triangleq \begin{bmatrix} 0 & * & 0 \\ * & * & * \\ 0 & * & 0 \end{bmatrix} h$$

it is reasonable to use a "7-point" ILU-decomposition [110]:

$$\mathcal{L}_h^L \triangleq \begin{bmatrix} 0 & 0 & 0 \\ * & * & 0 \\ 0 & * & * \end{bmatrix} h, \quad \mathcal{L}_h^U \triangleq \begin{bmatrix} * & * & 0 \\ 0 & * & * \\ 0 & 0 & 0 \end{bmatrix} h, \quad \mathcal{L}_h^R \triangleq \begin{bmatrix} * & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & * \end{bmatrix} h. \quad (9.31)$$

For a 9-point operator

$$\mathcal{L}_h \triangleq \begin{bmatrix} * & * & * \\ * & * & * \\ * & * & * \end{bmatrix}_h$$

it is convenient to use a "9-point" ILU-decomposition:

$$\mathcal{L}_h^L \triangleq \begin{bmatrix} 0 & 0 & 0 \\ * & * & 0 \\ * & * & * \end{bmatrix}_h, \quad \mathcal{L}_h^U \triangleq \begin{bmatrix} * & * & * \\ 0 & * & * \\ 0 & 0 & 0 \end{bmatrix}_h, \quad \mathcal{L}_h^R \triangleq \begin{bmatrix} * & 0 & 0 & 0 & 0 \\ * & 0 & 0 & 0 & * \\ 0 & 0 & 0 & 0 & * \end{bmatrix}_h. \quad (9.32)$$

Here "*" stands for certain non-zero elements. These decompositions are unique if, for example, the center element of \mathcal{L}_h^L is scaled to one.

We denote the eigenvalues of \mathcal{L}_h^L , \mathcal{L}_h^U and \mathcal{L}_h^R (cf. (9.6)) by
 $\lambda(\theta, h)$, $\lambda^L(\theta, h)$, $\lambda^U(\theta, h)$ ($-\pi < \theta \leq \pi$),

respectively. In terms of errors v_h , \bar{v}_h (before and after one ILU smoothing step), we obtain

$$\mathcal{L}_h^L \mathcal{L}_h^U \bar{v}_h = -\mathcal{L}_h^R v_h. \quad (9.33)$$

Thus, the corresponding smoothing operator is given by

$$\mathcal{S}_h = -(\mathcal{L}_h^L \mathcal{L}_h^U)^{-1} \mathcal{L}_h^R. \quad (9.34)$$

With respect to the basis (9.2), \mathcal{S}_h is represented by

$$\mathcal{S}_h \varphi(\theta, x) = A(\theta, h) \varphi(\theta, x) \quad (-\pi < \theta \leq \pi) \quad \text{with} \quad A(\theta, h) := \frac{\lambda^L(\theta, h) \lambda^U(\theta, h) - \lambda(\theta, h)}{\lambda^L(\theta, h) \lambda^U(\theta, h)}. \quad (9.35)$$

Using this representation, local mode analysis can be applied in connection with ILU-smoothing in a straightforward manner.

Table 9.6a shows results for the anisotropic model operator in (8.22) using 7-point ILU for smoothing. As transfer operators, the 7-point operators (4.31) are used. The coarse-grid difference operator \mathcal{L}_{2h} is assumed to be the same as \mathcal{L}_h with $2h$ instead of h . (We note that this operator coincides with the Galerkin operator $\mathcal{L}_{2h} := \mathcal{T}_h^{2h} \mathcal{L}_h \mathcal{T}_{2h}^h$ here.)

The results of Tables 9.6a show, in particular, that the convergence properties of the above method are not "symmetric" with respect to the size of ϵ : We have good convergence for $\epsilon \leq 1$ but ρ^* tends to 1 for $\epsilon \rightarrow \infty$. This behavior is simi-

lar to the one observed in connection with x_2 -ZEBRA relaxation in Table 8.4a. There, however, the dependence of ρ^* on the size of ϵ was more critical: even for values of ϵ which are only slightly larger than 1, one has to use x_1 -ZEBRA relaxation instead of x_2 -ZEBRA relaxation. If ZEBRA relaxation is used with lines in the appropriate direction, we see that ZEBRA clearly is superior to ILU.

On the other hand, in practice the less sensitive ϵ -dependence of ILU-smoothing has advantages if ϵ depends on x and if $\epsilon(x) > 1$ and $\epsilon(x) < 1$ in different parts of Ω : If ϵ does not become too large, ILU-smoothing will still be satisfactory. If we wanted to use ZEBRA relaxation, we would have to apply it in the alternating version. Table 9.6b gives a comparison of the computational work for ZEBRA and ILU in case of arbitrary 5-point difference operators (with variable coefficients). From this operation count and the convergence behavior shown in Tables 8.4b and 9.6a, one sees that for values of ϵ which are not too large both methods, ILU and alternating ZEBRA, are comparable in efficiency. The asymptotic convergence behavior of alternating ZEBRA, however, is completely symmetric with respect to ϵ . (Of course, ILU could be applied in an alternating manner also.)

For the FMG method (cf. Chapter 6) applied to anisotropic problems, we note that the use of alternating ZEBRA is slightly preferable over ILU: For FMG the pre-computations contained in Table 9.6b can no longer be neglected. This results in more efficient algorithms if alternating ZEBRA is used (in particular, if it is compared with "alternating ILU" which needs twice the amount of pre-computations as shown in Table 9.6b), [100].

ϵ	$(v_1, v_2) = (1, 0)$			$(v_1, v_2) = (1, 1)$		
	$(\mu^*)^\nu$	ρ^*	σ_S^*	$(\mu^*)^\nu$	ρ^*	σ_S^*
1000	0.843	0.843	1.20	0.710	0.710	1.10
100	0.607	0.607	0.858	0.368	0.368	0.710
10	0.273	0.273	0.386	0.075	0.095	0.287
2	0.146	0.128	0.146	0.021	0.034	0.119
1	0.126	0.126	0.131	0.016	0.027	0.085
0.5	0.144	0.144	0.144	0.021	0.022	0.071
0.1	0.165	0.165	0.165	0.027	0.027	0.067
0.01	0.171	0.171	0.171	0.029	0.029	0.068
0.001	0.171	0.172	0.172	0.029	0.029	0.068
W_h^{2h}/λ_h	# Add: 12.5, # Mult: 11			# Add: 20.5, # Mult: 19		

Table 9.6a: Results for the anisotropic model operator using 7-point ILU (9.31) for smoothing (and 7-point restriction and interpolation (4.31))

	pre-computations per point of fine grid			W_h^{2h}/W_h for $(v_1, v_2) = (1, 0)$	W_h^{2h}/W_h for $(v_1, v_2) = (1, 1)$
method	# Add	# Mult	# Div	# Add	# Mult
ZEBRA	1	2	1	8.25	8.5
alt.ZEBRA	2	4	2	12.25	13.5
ILU	5	12	1	12.5	20.5
					21

Table 9.6b: Comparison of computational work for arbitrary 5-point difference operators (variable coefficients) and the two-grid methods used in Table 8.4a, 8.4b and 9.6a, respectively.

In Table 9.7, we show results for (9.29) corresponding to Table 9.5: the smoothing methods used there have simply been replaced by 9-point ILU smoothing (9.32). The results of Table 9.7 show that this method behaves considerably better for $\epsilon \geq 0$. For $\epsilon < 0$, it behaves similarly to those of Table 9.5. This "unsymmetric behavior" is due to a corresponding unsymmetry in the ILU-decomposition: The convergence behavior shown in Table 9.7 is just reversed if we use a 9-point decomposition of the form

$$\mathcal{L}_h^L \triangleq \begin{bmatrix} * & * & 0 \\ * & * & 0 \\ * & 0 & 0 \end{bmatrix} h, \quad \mathcal{L}_h^U \triangleq \begin{bmatrix} 0 & 0 & * \\ 0 & * & * \\ 0 & * & * \end{bmatrix} h, \quad \mathcal{L}_h^R \triangleq \begin{bmatrix} 0 & * & * \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ * & * & 0 \end{bmatrix} h$$

instead of (9.32). Clearly, both decompositions can be used for smoothing in an alternating manner, giving a robust method for boundary value problems involving (9.29).

ϵ	$(v_1, v_2) = (1, 0)$		$(v_1, v_2) = (1, 1)$	
	ρ^*	σ_S^*	ρ^*	σ_S^*
-1.9	0.573	2.41	0.513	2.40
-1.7	0.402	0.775	0.307	0.740
-1.5	0.300	0.451	0.200	0.417
-1.0	0.165	0.214	0.083	0.172
0.0	0.126	0.126	0.021	0.021
1.0	0.075	0.076	0.030	0.038
1.5	0.083	0.088	0.047	0.049
1.7	0.107	0.108	0.061	0.062
1.9	0.142	0.142	0.084	0.084

Table 9.7: Results for the operator (9.29) using 9-point ILU-smoothing (9.32)
 $(J_h^{2h}$: FW restriction, J_{2h}^h : linear interpolation)

9.3 A short discussion of other theoretical approaches

As mentioned already in the introduction, all theoretical approaches to multigrid methods known so far have some specific disadvantages:

- Whereas the model problem analysis can be directly applied only to a rather small class of problems, the local Fourier analysis refers to certain "ideal" situations.
- For general problems, rigorous results are obtained by several more abstract approaches. These show the h -independent convergence behavior of multigrid methods in principle, but only under algorithmic assumptions or with constants, which are (very) unrealistic in concrete cases. Usually, these general approaches do not give the full impression about the high efficiency of multigrid methods, nor can they be used for the optimal (or even a reasonable) arrangement of algorithms.

As in the model problem and local Fourier analysis, in any proof for h -independent convergence of two-grid methods one has to distinguish the effects of the methods on low- and on high-frequency error components. The proofs differ, among other things, with respect to the definition of "low" and "high" and in the way how the above effects are measured.

We want to sketch two of the general approaches here and make some additional remarks. For simplicity, we restrict ourselves to $(h, 2h)$ two-grid methods.

9.3.1 Splitting of the two-grid operator norm into a product

In Hackbusch's approach (see [50] and the references cited there) the above effects are measured implicitly by use of a certain splitting of $\|M_h^{2h}\|$ (for a suitably chosen norm) into a product. For $v_1=v$, $v_2=0$ this splitting is of the form:

$$\|M_h^{2h}\| \leq \|K_h^{2h} L_h^{-1}\| \cdot \|L_h S_h^v\|. \quad (9.36)$$

The following properties are assumed to hold:

(1) approximation property:

$$\|K_h^{2h} L_h^{-1}\| = \|L_h^{-1} - I_{2h}^h L_{2h}^{-1} I_h^{2h}\| \leq Ch^\delta \quad (\delta > 0), \quad (9.37)$$

(2) smoothing property:

$$\|L_h S_h^v\| \leq n(v)h^{-\delta} \quad \text{with} \quad n(v) \rightarrow 0 \quad (v \rightarrow \infty). \quad (9.38)$$

Under these assumptions, the h -independent boundedness of $\|M_h^{2h}\|$ (by a constant less than one) immediately follows for sufficiently large v . Hackbusch was able to verify the assumptions (1) and (2) (or suitable generalizations) for quite general classes of problems. In particular, for second order problems with sufficient regularity, (1) and (2) can be fulfilled - by a suitable choice of MG components - with $\|\cdot\| = \|\cdot\|_S$, $\delta=2$ and $\eta(v) \sim 1/v$. For details see [50], Chapter 3.

Thus Hackbusch's approach is very satisfactory from a theoretical point of view. With respect to its quantitative part, however, there are several limitations and disadvantages in this theory.

- Firstly, in its simplest form the theory is an asymptotic theory, in particular, with respect to v . If the constant C in (9.37) is large, this has to be compensated by choosing v sufficiently large. Thus, in general, this theory does not allow one to prove the h -independent convergence of a given two-grid method (with fixed, small v).
- In its general form, the theory does not distinguish between very different situations (e.g. Poisson-like operators, anisotropic operators, singular perturbed operators, highly indefinite cases, etc.). We know, however, that multigrid algorithms have to be arranged quite differently for such different situations in order to become efficient. In Hackbusch's theory, these differences may be reflected by the size of the constants occurring. If one wants to obtain results which take specific situations into account, one has to choose - often very complicated - special norms, discrete spaces etc. (For a rather simple example, see [50], Chapter 6.)
- The theory directly applies only to norms, not to the spectral radius of a two-grid iteration operator. Thus, in any case, the suitable choice of norms is a substantial requirement of this approach. There are, however, practically important multigrid methods for which $\|M_h^{2h}\|_S$ is unbounded as $h \rightarrow 0$ (see Chapter 8 and the previous section). These methods cannot be treated with Hackbusch's approach (at least not in the spectral or energy norm). In particular, the use of "straight injection" in the fine-to-coarse transfer causes this difficulty.
- In (1) and (2) above, only $v_2=0$ (or $v_1=0$) are suitable choices of v_1, v_2 . In practice, however, multigrid methods with $v_1 > 0$ and $v_2 > 0$ usually have advantages: see, e.g., Tables 3.3, 8.1 and [25], Section 6.1 (also cf. [50], Section 3.5).

Example: We consider model problem (P) and the two-grid method which was treated in Theorem 8.1. If we were to specialize the general estimates of Hackbusch's theory to this situation, the constants occurring would become by far too large. But even if we compute the optimal norm bounds in (1) and (2) (with $\delta=2$ and $\|\cdot\| = \|\cdot\|_S$) by use of Fourier representations, the resulting bounds for $\|M_h^{2h}\|_S$ differ considerably from ρ^* (cf. Section 8.1). Table 9.8 compares corresponding norm bounds for $\|M_h^{2h}\|_S$

with ρ^* for different values of v . The optimal value of v for the method at hand was found to be $v=2$ in Section 8.2 giving $\rho^*=0.074$. Table 9.8 shows that 10 relaxation steps are needed to achieve a bound for $\|M_h^{2h}\|_S$ of this size.

v	1	2	3	4	...	10
$\rho^*(v)$	0.250	0.074	0.053	0.041	...	0.018
$\ M_h^{2h}\ _S \leq$	0.770	0.372	0.247	0.185	...	0.074

Table 9.8: Comparison of $\rho^*(v)$ with bounds of $\|M_h^{2h}\|_S$ obtained by Hackbusch's estimation

If in the above two-grid method the FW restriction operator is replaced by the HW restriction operator, we obtain norm values σ_S^* as shown in Table 8.1b. Property (1), however, is not at all satisfied (for $\|\cdot\|=\|\cdot\|_S$ and $\delta=2$).

We finally remark that the above smoothing property (2) (and the factor $n(v)$ occurring there) is not directly related to the definition of the smoothing factor given in Section 7.5.

9.3.2 Splitting of the two-grid operator norm into a sum

Another theoretical approach starts from a splitting

$$G(\Omega_h) = G^{\text{low}}(\Omega_h) \oplus G^{\text{high}}(\Omega_h) \quad (9.39)$$

and a corresponding splitting of I_h into a sum of projection operators

$$I_h = Q_h^{\text{low}} + Q_h^{\text{high}}. \quad (9.40)$$

(Here "low" and "high" do not need to have the same meaning as in the model problem or in the local mode analysis.)

In order to derive norm estimates for $M_h^{2h} = K_h^{2h} S_h^v$, the above splitting is used in the following way:

$$M_h^{2h} = K_h^{2h} (Q_h^{\text{low}} + Q_h^{\text{high}}) S_h^v \quad (9.41)$$

which is estimated by

$$\|M_h^{2h}\| \leq \|K_h^{2h} Q_h^{\text{low}}\| \cdot \|S_h^v\| + \|K_h^{2h}\| \cdot \|Q_h^{\text{high}} S_h^v\|. \quad (9.42)$$

This estimate can be regarded as the basis of Wesseling's approach [108]. Under reasonable circumstances, $\|S_h^v\|$ and $\|K_h^{2h}\|$ can be assumed to be bounded by constants

C and C' (independent of h). The idea then is to make - again independently of h - $\|K_h^{2h} Q_h^{\text{low}}\|$ small by a suitable definition of "low" and to make $\|Q_h^{\text{high}} S_h^v\|$ small by choosing a sufficiently large number v of relaxation steps.

One obvious disadvantage of this approach is the following. Once the term "low" (i.e. Q_h^{low}) is defined, the first summand in (9.42) (more precisely: its supremum over h for fixed meshsize ratio q^*) is bounded from below, independently of v . In particular, an h -independent bound for $\|M_h^{2h}\|$ derived from (9.42) then is bounded from below, independently of v , also. This is, however, not realistic: typically $\|M_h^{2h}\|$ is proportional to $1/v$ for $v \rightarrow \infty$ (see the corresponding remark in the previous section).

Example: Consider model problem (P) and the sample method in Section 3.1. If we use the natural definition of "low" given in (3.15) one can compute explicitly (using the Fourier representations given in Chapter 3):

$$\sup_h \|K_h^{2h} Q_h^{\text{low}}\|_S = 0.901\dots, \quad \sup_h \|S_h^v\|_S = 1 \quad (v=1,2,\dots). \quad (9.43)$$

Wesseling gives bounds for the single terms in (9.42) which are valid for rather general situations. For special situations, however, these bounds turn out to be very unrealistic. Starting from (9.42), Wesseling estimates

$$\|K_h^{2h} Q_h^{\text{low}}\| \leq \|K_h^{2h} L_h^{-1}\| \cdot \|L_h Q_h^{\text{low}}\|, \quad \|K_h^{2h}\| \leq \|K_h^{2h} L_h^{-1}\| \cdot \|L_h\|. \quad (9.44)$$

By this, the approximation property (9.37) enters again. The further estimates of Wesseling's proof - which correspond to a very restrictive definition of "low" - are aimed at obtaining a small bound for the right hand side of the first inequality in (9.44). Applied to model problem (P) and the sample method in Section 3.1 (with the exception that the Galerkin operator (2.21) is used on the coarser grid), this definition means that only frequencies (7.4) with - roughly - $|\theta| < 10^{-3}$ are regarded as low frequencies. As a consequence, millions of relaxation steps have to be carried out per cycle in order to make $\|Q_h^{\text{high}} S_h^v\|$ sufficiently small (cf. the corresponding remark in [108]).

This extremely unsatisfactory result is due to the fact that the definition of "low" frequencies is not natural here: The definition used is not oriented to the ratio of the meshsizes h and $2h$, but rather to the technique of the proof.

9.3.3 Further remarks on the definition of "low" and "high"

(1) Another interesting way to measure low and high frequencies is contained in Braess' approach [13]. This approach refers to red-black coarsening ($H=\sqrt{2}h$). Apart

from the energy norm $\|\cdot\|_E$ on $\mathbb{G}(\Omega_h)$, Braess introduces a certain semi-norm $|\cdot|$ on $\mathbb{G}(\Omega_h)$ which takes only grid points of Ω_H into account (see [13], Section 2). Then the quotient $\lambda := |\cdot|/\|\cdot\|_E$ turns out to be a measure for the smoothness of Ω_h -grid functions with respect to Ω_H ($\lambda \sim 0$ corresponds to "high" and $\lambda \sim 1$ corresponds to "low" frequencies).

Using this measure in connection with a strengthened Cauchy inequality, Braess obtains (not optimal, but) very good h -independent norm bounds for an MGR-method for Poisson equation on certain convex domains. These results were extended to methods using standard coarsening by Verfürth [106].

(2) Finally, we mention that in McCormick's algebraic approach to multigrid methods [70], the elements of the nullspace of I_h^{2h} can essentially be regarded as "high" frequencies.

10. Multigrid programs for standard applications

On the basis of the model problem analysis, two collections of programs for certain second order elliptic equations have been developed. We refer to these collections as MGØØ and MGØ1.

The MGØØ programs are restricted to rectangular domains (with different types of boundary conditions). The sample program listed in the Appendix (for Poisson's equation with Dirichlet boundary conditions in the unit square) is a very special program of the MGØØ collection. More information about MGØØ and, in particular, a systematic comparison with several direct Fast Elliptic Solvers and related methods is given by Foerster, Witsch [36].

The MGØ1 programs refer to second order equations with Dirichlet boundary conditions on non-rectangular bounded domains. The main objective in the development of these programs was to study the influence of the shape of the domain on the behavior of several multigrid algorithms. Because of the complexity of programs for problems on general domains and with respect to possible generalizations, special emphasis was laid upon the structural transparency of these programs; efficiency was only our second objective. (The technical structure of the MGØ1 programs is partially adopted from a similar program contained on MUGTAPE, see [25]).

Current versions of the MGØ1 collection refer to the equations

$$(1) \quad -\Delta u + c(x)u = f(x),$$

$$(2) \quad -a_1(x)u_{x_1 x_1} - a_2(x)u_{x_2 x_2} + c(x)u = f(x),$$

$$(3) \quad -a_1(x)u_{x_1 x_1} - a_2(x)u_{x_2 x_2} + b_1(x)u_{x_1} + b_2(x)u_{x_2} + c(x)u = f(x),$$

(4) generalizations to $f(x,u)$.

In Section 10.1, we give a short description of admissible domains and the discretization. Sections 10.2 and 10.3 refer to the cases (1) and (2), respectively. As for (2), we are especially interested in the case of strongly differing coefficients a_1 and a_2 . In Section 10.4, only a few remarks are made with respect to the technically more complicated case (3). Nonlinear applications (4) have been reported upon already in Chapter 5.

10.1 Description of domains and discretization

For technical reasons only, MGØ1 has, up to now, been restricted to domains Ω which can be described by two values A, B and two functions $FL(x_1)$ (F-low) and

$FH(x_1)$ (F-high) in the following way:

$$\Omega = \{(x_1, x_2) : A < x_1 < B, FL(x_1) < x_2 < FH(x_1)\}.$$

Here the functions FL and FH may have a finite number of jumps (cf. Figure 10.1). For the discretization, a grid $\Omega_h \cup \Gamma_h$ is used consisting of the set Ω_h of *interior grid points* and the set Γ_h of *boundary grid points*. Here Ω_h is defined by

$$\Omega_h := \Omega \cap G_h(x^*), \quad G_h(x^*) := \{x = x^* + h\kappa : \kappa \in \mathbb{Z}^2\}, \quad h = h_{x_1} = h_{x_2}$$

with grid origin $x^* \in \mathbb{R}^2$. The boundary grid points are just the intersection of Γ with grid lines of $G_h(x^*)$.

We denote by

- *regular interior grid points*: all points of Ω_h , whose neighboring grid points in northern, southern, western and eastern direction are also points of Ω_h ;
- *irregular interior grid points*: all interior grid points which are not regular.

At regular interior grid points, the differential operator L is approximated by the usual 5-point discretization (order of consistency = 2), replacing e.g.

$$u_{x_1 x_1} \text{ by } \frac{1}{h^2} [1 \ -2 \ 1]_h, \quad u_{x_1} \text{ by } \frac{1}{2h} [-1 \ 0 \ 1]_h.$$

Near the boundary, i.e. in the irregular grid points, a 5-point Shortley-Weller approximation [89] is used. This means, for example, that

$$a_1(x)u_{x_1 x_1} + a_2(x)u_{x_2 x_2}$$

is replaced by

$$2 \begin{bmatrix} & & a_2(x) \\ & \frac{a_1(x)}{h_N(h_N+h_S)} & - \frac{a_1(x)}{h_W h_E} + \frac{a_2(x)}{h_N h_S} & \frac{a_1(x)}{h_E(h_W+h_E)} \\ & . & \frac{a_2(x)}{h_S(h_N+h_S)} & \end{bmatrix} u(x).$$

Here h_N, h_S, h_W , and h_E denote the distances to the neighboring grid points in northern, southern, western and eastern direction, respectively.

So far, we have not yet made any assumptions about the coefficients a_1, a_2, b_1, b_2, c . Since only elementary multigrid techniques will be considered in the following sections, certain degeneracies have to be excluded. We want to mention these restrictions here only qualitatively:

- the problem is assumed to be elliptic and "nicely definite";
- all coefficients should have some "smoothness", e.g. considerable jumps in size are not admitted;
- the ratios of a_1, b_1 and of a_2, b_2 , respectively, should be such that a singularly perturbed behavior (of the discrete problem) is not likely and that the central differencing of the first derivatives leads to no instability.

If one of these assumptions is not fulfilled, in general a more sophisticated multigrid approach will be necessary (which is adapted to the particular situation at hand, see Section 10.4).

10.2 Helmholtz equation (with variable c)

The discrete problem

$$\begin{aligned} -\Delta_h u_h + c(x)u_h &= f^\Omega(x) \quad (x \in \Omega_h) \\ u_h &= f^\Gamma(x) \quad (x \in \Gamma_h) \end{aligned} \tag{10.1}$$

is solved by MG01 [97]

- (1) either by using the underlying multigrid method as an iterative solver (which is especially suitable to investigate the influence of the shape of the domain Ω on the convergence behavior),
- (2) or in the full multigrid mode as explained in Chapter 6 (which is the usual way to apply the program in practice).

The main components of MG01 used in solving (10.1) iteratively are the following:

- Coarsening: The grid-coarsening is done by doubling the (interior) meshsize in both directions, i.e. $\Omega_{2h} = \Omega \cap G_{2h}(x^*)$. On all grids the discrete operators are constructed in the same way as on the finest grid. The coarsest grid may be "very coarse" (but should be geometrically reasonable, see Figure 10.1).
- Smoothing: For smoothing RB relaxation is used (usually with $v_1=2, v_2=1$).
- Grid transfer: For the fine-to-coarse and coarse-to-fine transfers we use half

injection (HI, see Section 8.1) and linear interpolation, respectively.

- Structure of cycles: Alternatively, V- or W-cycles (see Section 4.1) may be used.

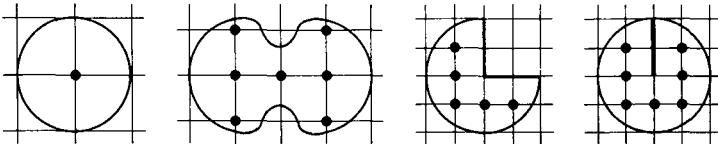


Figure 10.1: "Reasonable" coarsest grids

All following quantitative results refer to Poisson's equation and the MG01 version described above with

$$v_1 = 2, v_2 = 1. \quad (10.2)$$

If N denotes the number of grid points of Ω_h , the total computational work for one iteration step of the corresponding method is less than

15 N additions, 5 N multiplications (for V-cycles),	(10.3)
23 N additions, 7.5 N multiplications (for W-cycles),	

neglecting lower order terms. These numbers are independent of the shape of the domain.

Table 10.1 shows some numerically calculated asymptotic convergence factors of the multigrid iteration (for both V- and W-cycles) for several domains. All domains are comparable in size with the unit square. The convergence factors have been computed by a v. Mises vector iteration.

We recall that on rectangular domains for $h_{x_1} = h_{x_2}$ the corresponding asymptotic two-grid convergence factor is given by $\rho^*(3) \approx 0.034$ (see Section 8.2). As far as W-cycles ($\gamma=2$) are concerned, Table 10.1 shows that the multigrid convergence factors (for $h=1/128$) are nearly the same as ρ^* . This is also true for general domains, as long as there are no reentrant corners. The worst convergence factor, namely 0.097 instead of ~ 0.03 , can be observed for the domain with a cut. Here a singular behavior like \sqrt{R} is typical for the solution $u(x)$ near the singular point (where R denotes the distance of x to the singular point). Clearly, for such problems Poisson's equation on a rectangle is no longer a model case. Nevertheless, the W-cycle convergence turns out to be very satisfactory even in such cases. We point out that the given convergence factors remain essentially unchanged

if γ is increased ($\gamma \geq 3$). This means that these factors can practically be regarded as the asymptotic two-grid convergence factors.

As expected, the multigrid convergence factors for V-cycles ($\gamma=1$) are worse than those for W-cycles. For domains without reentrant corners, the convergence factors are nevertheless so good that the V-cycle efficiency (convergence factor versus computational work) is slightly better than the W-cycle efficiency. In the cases with reentrant corners, however, the V-cycle convergence deteriorates considerably. Altogether, V-cycles may be used just as well as W-cycles at least for "harmless" problems. To be on the safe side, however, we usually prefer to use the more robust W-cycles.

domain	convergence factor	domain	convergence factor
	V: 0.059 W: 0.033		V: 0.056 W: 0.033
	V: 0.059 W: 0.033		V: 0.122 W: 0.049
	V: 0.063 W: 0.032		V: 0.182 W: 0.075
	V: 0.058 W: 0.033		V: 0.230 W: 0.097
	V: 0.088 W: 0.033		V: 0.221 W: 0.095

Table 10.1: Numerically computed asymptotic convergence factors per multigrid iteration step for different domains ($v_1=2$, $v_2=1$, $h=1/128$) and both V- and W-cycles.

For the full multigrid version we used $r=1$ W-cycles per level and cubic FMG-interpolation (cf. Section 6.3). In Table 10.2, we give FMG results for various domains. For some values of h , both the error $\|\tilde{u}_h - u_h\|_2$ of the computed FMG solution \tilde{u}_h compared to the solution u_h of the discrete problem (upper values) and the exact discretization error $\|u_h - u\|_2$ (lower values) are shown.

For the examples treated in Table 10.2, the continuous solution is given by

$$u(x) = \sin\pi(x_1+x_2) \quad \text{for the upper 5 domains}$$

$$u(R, \phi) = \sqrt{R} \sin(\phi/\alpha) \quad \text{for the lower 4 domains}$$

In the latter case, u reflects the singular behavior of solutions which is typical

for respective reentrant corners. Here (R, ϕ) denote polar coordinates with respect to the singular point; $\alpha\pi$ is the inner angle of the domain at this point (see Table 10.2).

As we have seen in Table 10.1, the convergence factors of the multigrid iteration deteriorate with increasing α for the domains with reentrant corners ($1 < \alpha \leq 2$). The question arises, whether the corresponding convergence speed is still sufficient for the satisfactory performance of FMG (cf. the influence of n^r in the estimation (6.10)). For this, one should notice that the discretization error becomes larger for increasing α also. This means that the value of κ_1 in (6.7) is smaller than 2 in these cases. More precisely, the following estimate is valid [66]: For any $1 \leq \alpha \leq 2$ and any $\epsilon > 0$ there exists a constant C such that

$$|u(R, \phi) - u_h(R, \phi)| \leq C h^\alpha \frac{\frac{2}{R} - 2\epsilon}{R^{\frac{1}{\alpha}}} + \epsilon = \begin{cases} O(h^\alpha) & \text{if } R \text{ fixed} \\ O(h^{\alpha - \frac{1}{\alpha}}) & \text{if } R = O(h). \end{cases} \quad (10.4)$$

Therefore, the loss of MG convergence speed is - so to say - compensated by a loss of discretization accuracy. The errors given in Table 10.2 show indeed that the main objective of the FMG method, namely to obtain approximate solutions \tilde{u}_h with $\|\tilde{u}_h - u_h\|_2 \leq \|u_h - u\|_2$, is achieved for all examples considered. The same is true for highly oscillatory solutions, see Table 10.3.

All results in Tables 10.2 and 10.3 refer to W-cycles. We have computed corresponding errors $\|\tilde{u}_h - u_h\|_2$ for V-cycles also (maintaining $r = 1$). The ratio $\|\tilde{u}_h - u_h\|_2 / \|u_h - u\|_2$ is larger then, but in all cases still ≤ 1 . This means, that V-cycles may also be employed in the cases considered.

The total computational work of MG01 in the FMG version ($r=1$) is less than

22 \mathcal{N} additions, 8 \mathcal{N} multiplications (if V-cycles are used),	
32.5 \mathcal{N} additions, 11.5 \mathcal{N} multiplications (if W-cycles are used)	(10.5)

(neglecting lower order terms), where \mathcal{N} is the number of grid points on the finest grid. These numbers are independent of the shape of the domain. In particular, they are the same as for a corresponding special program for rectangular domains. Concerning the real computing times, this special program is, of course, faster than MG01 (for a given reasonable \mathcal{N}), as in MG01 additional work has to be performed due to the more complicated grid structure. (As for computing times concerning programs on rectangular domains, see MG00 [36].)

domain	$h=1/32$	$h=1/64$	$h=1/128$
	0.157(-5) 0.194(-3)	0.114(-6) 0.486(-4)	0.789(-8) 0.121(-4)
	0.119(-5) 0.145(-3)	0.264(-6) 0.368(-4)	0.300(-7) 0.922(-5)
	0.562(-6) 0.274(-4)	0.606(-7) 0.672(-5)	0.945(-8) 0.166(-5)
	0.445(-5) 0.339(-3)	0.520(-6) 0.849(-4)	0.853(-7) 0.213(-4)
	0.246(-5) 0.129(-3)	0.453(-6) 0.331(-4)	0.701(-7) 0.841(-5)
	0.123(-4) 0.198(-3)	0.168(-4) 0.694(-4)	0.174(-5) 0.239(-4)
	0.792(-4) 0.816(-3)	0.363(-4) 0.337(-3)	0.156(-4) 0.137(-3)
	0.234(-3) 0.201(-2)	0.122(-3) 0.938(-3)	0.590(-4) 0.432(-3)
	0.471(-3) 0.384(-2)	0.269(-3) 0.196(-2)	0.144(-3) 0.989(-3)

solution:

$$u(x) = \sin\pi(x_1 + x_2)$$

$$u(x) = \sqrt{R} \sin(\phi/\alpha)$$

Table 10.2: FMG results by MG01 ($v_1=2$, $v_2=1$, W-cycles, $r=1$) for smooth solutions and singular solutions (upper value: $\|\tilde{u}_h - u_h\|_2$, lower value: $\|u_h - u\|_2$)

domain	$n_1=9, n_2=6$	$n_1=18, n_2=12$	$n_1=30, n_2=20$
	0.634(-5) 0.169(-2)	0.723(-4) 0.679(-2)	0.290(-3) 0.191(-1)

$$u(x) = \sin(n_1\pi x_1)\sin(n_2\pi x_2)$$

Table 10.3: Method as in Table 10.2 for highly oscillatory solutions ($h = 1/128$)

10.3 Anisotropic operators

The structure of the MG \emptyset 1-version for anisotropic problems

$$\begin{aligned} -a_1(x)u_{x_1x_1} - a_2(x)u_{x_2x_2} + c(x)u &= f^\Omega(x) \quad (x \in \Omega) \\ u &= f^\Gamma(x) \quad (x \in \Gamma) \end{aligned} \quad (10.6)$$

($a_1(x) > 0, a_2(x) > 0$) is the same as for the Helmholtz equation with the following exceptions:

- RB relaxation is replaced by ZEBRA or - if necessary - by alternating ZEBRA relaxation;
- The HI restriction operator is replaced by the FW operator (see Section 3.1) which becomes particularly simple in connection with ZEBRA relaxation.

If $a_1(x)-a_2(x)$ does not change its sign, ZEBRA relaxation is used with lines in a fixed direction (with $v_1=1, v_2=0$ or $v_1=v_2=1$; cf. Section 8.3), namely

$$\begin{aligned} x_1\text{-ZEBRA} &\quad \text{if } a_1(x) \geq a_2(x), \\ x_2\text{-ZEBRA} &\quad \text{if } a_1(x) \leq a_2(x). \end{aligned} \quad (10.7)$$

If $a_1(x)-a_2(x)$ changes its sign on Ω , alternating ZEBRA relaxation of type (8.23) is used instead (with $v_1=1, v_2=0$ or $v_1=v_2=1$). (Notice that one smoothing step of alternating ZEBRA costs twice the work of one single x_2 - or x_1 -ZEBRA step.)

In the following we give some convergence factors for MG \emptyset 1 using alternating ZEBRA relaxation: The convergence behavior of this method is very good for any size of a_1 and a_2 and independent of the domain.

Table 10.4 shows some numerically calculated W-cycle convergence factors for (10.6) with constant coefficients

$$a_1(x) \equiv \epsilon, \quad a_2(x) \equiv 1, \quad c(x) \equiv 0 \quad (10.8)$$

on the unit square $\Omega=(0,1)^2$. For comparison, also the corresponding smoothing factors $(\mu^*)^v$ and two-grid convergence factors ρ^* are given in Table 10.4 (cf. Section 8.3). We see that in particular ρ^* predicts the behavior of the multigrid method very well. For $\epsilon \rightarrow \infty$ or $\epsilon \rightarrow 0$ all three values approach 0.125 (in the case $v_1=1, v_2=0$) and 0.052 (in the case $v_1=1, v_2=1$).

ϵ	$v_1 = 1, v_2 = 0$			$v_1 = 1, v_2 = 1$		
	$\mu^*(1)$	$\rho^*(1)$	W-cycle	$(\mu^*(2))^2$	$\rho^*(2)$	W-cycle
100	0.122	0.119	0.117	0.051	0.051	0.047
10	0.102	0.082	0.079	0.041	0.038	0.036
2	0.061	0.019	0.018	0.020	0.013	0.012
1	0.048	0.023	0.019	0.014	0.009	0.008
0.5	0.061	0.019	0.018	0.020	0.013	0.012
0.1	0.102	0.082	0.079	0.041	0.038	0.036
0.01	0.122	0.119	0.117	0.051	0.051	0.047

Table 10.4: Comparison of smoothing factors, two-grid convergence factors ρ^* and numerically computed W-cycle convergence factor for alternating ZEBRA relaxation and constant coefficients (10.8) ($\Omega = (0,1)^2$, $h = 1/64$).

Table 10.5 shows convergence factors for

$$a_1(x) = K^{x_1+x_2-1}, \quad a_2(x) \equiv 1, \quad c(x) \equiv 0 \quad (10.9)$$

instead of (10.8). Obviously, $1/K \leq a_1(x) \leq K$ ($x \in \Omega$) holds. The given results for $K = 1, 2, 10$ and 100 show that even for variable $a_1(x)$ the convergence behavior is well predicted by the two-grid convergence factors for constant a_1 , if the respective "worst cases" (namely: $\max a_1(x)$ or $\min a_1(x)$ ($x \in \Omega$)) are considered: For example, in the case $K = 100$ the corresponding convergence factor 0.097 (for $v_1=1, v_2=0$) is smaller than the two-grid convergence factor corresponding to the "worst" constant a_1 -values (namely $a_1 \equiv 100$ and $a_1 \equiv 1/100$) which is 0.117 (see Table 10.4).

K	$v_1 = 1, v_2 = 0$	$v_1 = 1, v_2 = 1$
1	0.019	0.008
2	0.014	0.008
10	0.058	0.025
100	0.097	0.038

Table 10.5: Numerically computed W-cycle convergence factors for alternating ZEBRA relaxation and variable coefficients (10.9) ($\Omega = (0,1)^2$, $h = 1/64$)

Finally, Table 10.6 shows W-cycle convergence factors for some non-rectangular domains (again for $a_2(x) \equiv 1$, $c(x) \equiv 0$). The convergence factors given for the upper three domains behave as in the case of the unit square treated in Table 10.5. The differences in the convergence factors are due to the fact that $a_1(x)$ varies over

different scales for the different domains. Again, in all cases the convergence factors listed are smaller than the two-grid convergence factors ρ^* corresponding to the worst constant a_1 . The situation for the lower three domains (reentrant corners) is quite similar in spite of the singularity. Only for the last domain (with a cut) does the convergence behavior slow down slightly in comparison to the first domain.

domain			
$a_1(x)$	$10^2(x_1+x_2)$	$10^2(x_1+x_2)$	$10^4(x_1+x_2)/3$
$v_1=1, v_2=0$	0.073	0.063	0.093
$v_1=1, v_2=1$	0.031	0.017	0.039
domain			
$a_1(x)$	$10^2(x_1+x_2)$	$10^2(x_1+x_2)$	$10^2(x_1+x_2)$
$v_1=1, v_2=0$	0.065	0.069	0.091
$v_1=1, v_2=1$	0.035	0.041	0.056

Table 10.6: Numerically computed W-cycle convergence factors for alternating ZEBRA relaxation, variable coefficients and different domains

Table 10.7 shows the amount of computational work needed per MG iteration step in the case of arbitrary 5-point difference approximations. The numbers given do not include the computational work required for the decomposition of the occurring tri-diagonal systems. These decompositions are regarded as precomputations which have to be performed only once before starting the MG iteration. They amount to

$$2\frac{2}{3}\mathcal{N} \text{ additions}, 5\frac{1}{3}\mathcal{N} \text{ multiplications}, 2\frac{2}{3}\mathcal{N} \text{ divisions}. \quad (10.10)$$

	$v_1 = 1, v_2 = 0$	$v_1 = v_2 = 1$	
	$+/-$	*	$+/-$
V-cycle	$16\frac{1}{3}N$	$18N$	$27\frac{1}{3}N$
W-cycle	$24\frac{1}{2}N$	$27N$	$41N$

Table 10.7: Operation count for one step of MG iteration with alternating ZEBRA relaxation for general 5-point discretizations with variable coefficients (neglecting lower order terms).

The FMG version of the iterative MG method described above behaves similarly as shown in Tables 10.2 and 10.3 for Poisson's equation. Thus we do not give detailed results here. In the FMG mode, it is in practice sufficient for most cases to use

$$\text{V-cycles; } r = 1; \quad v_1 = 1, v_2 = 0.$$

To be on the safe side, however, one could use W-cycles instead (with $v_1=1, v_2=0$ or even $v_1=v_2=1$).

The total operation count for the FMG versions (corresponding to Table 10.7) is given in Table 10.8, assuming cubic FMG-interpolation. (Here, of course, we take the set-up operations (10.10) into account.)

	$v_1 = 1, v_2 = 0$			$v_1 = v_2 = 1$		
	$+/-$	*	\div	$+/-$	*	\div
V-cycle, $r = 1$	$25\frac{4}{9}N$	$30N$	$2\frac{2}{3}N$	$40\frac{1}{9}N$	$47\frac{5}{9}N$	$2\frac{2}{3}N$
W-cycle, $r = 1$	$36\frac{1}{3}N$	$42N$	$2\frac{2}{3}N$	$58\frac{1}{3}N$	$68\frac{2}{3}N$	$2\frac{2}{3}N$

Table 10.8: Operation count for the FMG method (corresponding to Table 10.7).

10.4 More general situations

For the numerical treatment of problems involving the more general differential operator

$$-a_1(x)u_{x_1 x_1} - a_2(x)u_{x_2 x_2} + b_1(x)u_{x_1} + b_2(x)u_{x_2} + c(x)u = f(x),$$

one has to take the following complications into account.

For fixed h , the discretization of first derivatives by central differencing leads to instabilities if the coefficients a_1, a_2 are very small compared to b_1, b_2 roughly if

$$|a_1| \leq |b_1|h/2, \quad |a_2| \leq |b_2|h/2. \quad (10.11)$$

But even if such *singularly perturbed* cases are excluded, one must be aware of similar difficulties which may still occur on the coarser grids used in the MG process.

In many cases these difficulties may be overcome, e.g., by introducing some *artificial viscosity* on the respective grids, i.e. the actual values of a_1 and/or a_2 are increased in a suitable way. This possibility (and its relevance for multigrid methods) has been described by Brandt in several papers [23],[26]. A systematic study for a model equation was made by Börgers [12].

In the singularly perturbed case (10.11), artificial viscosity is already needed on the finest grid. Through this, the order of discretization accuracy is also reduced. The MG treatment of such problems has been studied, e.g., in [23],[12],[56], [3].

If, on the other hand,

$$|a_1| > |b_1|h_0/2, \quad |a_2| > |b_2|h_0/2 \quad (10.12)$$

holds (where h_0 denotes the meshsize of the coarsest grid), the multigrid methods described above may also be used without any change for these more general problems. Exceptions are given by *indefinite* and *highly indefinite problems* (for which we refer to [28]) and problems with (*strongly*) *discontinuous coefficients*. Problems of the latter kind are treated in [1],[60].

11. Multigrid methods on composite meshes

11.1 Composite mesh discretization and a "naive" multigrid approach

The numerical treatment of elliptic equations with general boundary conditions on general domains is known to be technically rather complicated. One approach is to use different coordinate systems in the "main part" of the interior of the given domain and near the boundary. Advantages of this approach are due to the fact that suitably chosen local coordinates (with the boundary line being a grid line) allow the use of regular discretizations of the boundary conditions as well as higher order discretizations near the boundary. Furthermore, mesh refinement (orthogonally or tangentially to the boundary) can be performed in a technically simple way, for example for the treatment of boundary layers.

In the following description, we assume for simplicity that the given domain Ω is bounded and simply connected and that it has a smooth boundary Γ . For such a situation, in [95], a *composite mesh discretization method* has been considered. Here the given domain Ω is divided into two overlapping parts, Ω_I and Ω_0 : Ω_I is an *interior domain* ($\overline{\Omega}_I \subset \Omega$) and $\Omega_0 \subset \Omega$ is an "annulus-shaped" region along the boundary Γ (*boundary domain*). Ω_0 is assumed to be the image of a rectangular domain Ω_R under an orthogonal transformation Φ (see Figure 11.1).

The given problem on Ω is now discretized in both Ω_I and Ω_R (using the transformed equations in Ω_R), by use of, for example, rectangular grids. The grids are connected to each other by a suitable interpolation scheme. We use the notation *composite mesh system* for the resulting system which consists of discrete problems on Ω_I , Ω_R and of the interpolation relations.

This composite mesh system may be solved iteratively using a discrete analogue of Schwarz' alternating method. In each step of this method, the two discrete elliptic problems on Ω_I and Ω_R are solved in an alternating manner. Clearly, for each of these problems, multigrid methods may be used separately. This possibility is straightforward and has been studied in [68] for a model problem. Though the efficiency of this method is much better than, e.g., that of the corresponding SOR application, the total efficiency is limited by the convergence properties of Schwarz' method. The convergence of Schwarz' method, however, depends on the geometrical situation, e.g. on the overlapping of Ω_I and Ω_0 (roughly: the smaller the overlapping, the slower the convergence). On the other hand, a large region of overlap involves many extra grid points resulting in more computational work.

Instead of the "naive" combination of Schwarz' method with multigrid techniques, we propose a more direct multigrid approach to the composite mesh system. In this

method, which has been investigated systematically in [68] for a model problem, a multigrid hierarchy of composite meshes is used: The principle of Schwarz' alternating method is applied here only within the relaxation process for smoothing. It turns out that the efficiency of this smoothing does not depend sensitively on the geometrical situation as, e.g., the overlapping.

Clearly, the composite mesh idea can be used not only in such simple geometrical situations as assumed above. In general, one will have to compose not only two, but several meshes (for example, if Ω has a boundary which is only piecewise smooth).

11.2 A "direct" multigrid method for composite meshes

Let a linear elliptic boundary value problem (1.1) be given on a simply connected bounded domain Ω with smooth boundary Γ . As described in the previous section, we assume

$$\Omega = \Omega_I \cup \Omega_0, \quad \overline{\Omega}_I \subset \Omega, \quad \Omega_I \cap \Omega_0 \neq \emptyset. \quad (11.1)$$

Here Ω_0 denotes an annulus-shaped "boundary domain" with "outer" boundary Γ and "inner" boundary Γ_0 . Following the lines of Schwarz' alternating method, the original problem (1.1) is replaced by the two boundary value problems (0) and (I):

(0) *boundary problem* (on Ω_0):

$$\begin{aligned} L^\Omega u_0 &= f^\Omega(x) \quad (x \in \Omega_0), \\ L^\Gamma u_0 &= f^\Gamma(x) \quad (x \in \Gamma), \quad u_0 = f^{\Gamma_0}(x) \quad (x \in \Gamma_0); \end{aligned} \quad (11.2)$$

(I) *interior problem* (on Ω_I):

$$\begin{aligned} L^\Omega u_I &= f^\Omega(x) \quad (x \in \Omega_I), \\ u_I &= f^{\Gamma_I}(x) \quad (x \in \Gamma_I := \partial\Omega_I), \end{aligned} \quad (11.3)$$

where the connection to (1.1) is given by

$$f^{\Gamma_0}(x) = u_I(x) \quad (x \in \Gamma_0), \quad f^{\Gamma_I}(x) = u_0(x) \quad (x \in \Gamma_I). \quad (11.4)$$

We assume that Ω_0 is the image of a rectangular domain

$$\Omega_R := \{(s, t) : 0 < s \leq S, 0 < t < T\} \quad (11.5)$$

under a suitable orthogonal transformation ϕ (see Figure 11.1). For example, if Γ is parametrized with respect to arclength, i.e.

$$\Gamma = \{(x_1^0(s), x_2^0(s)) : 0 \leq s \leq S\} \quad (\dot{x}_1^0(s))^2 + (\dot{x}_2^0(s))^2 = 1,$$

a suitable domain Ω_0 can be defined using the mapping [18]

$$\phi(s, t) = \begin{bmatrix} x_1(s, t) \\ x_2(s, t) \end{bmatrix} := \begin{bmatrix} x_1^0(s) - t \dot{x}_2^0(s) \\ x_2^0(s) + t \dot{x}_1^0(s) \end{bmatrix}. \quad (11.6)$$

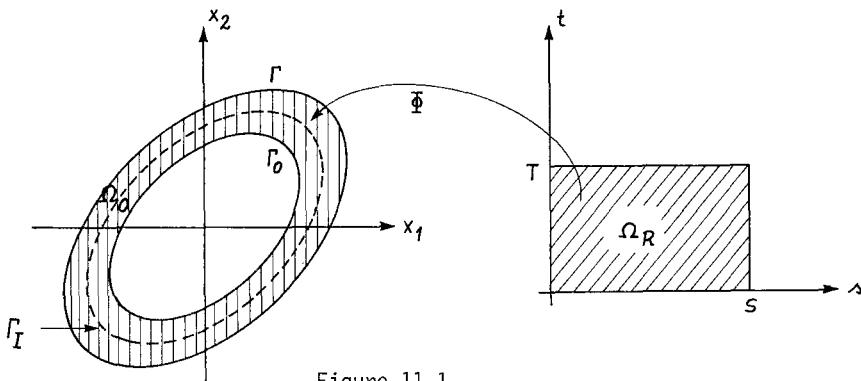


Figure 11.1

By (R) we denote problem (0) in terms of the local coordinates (s, t) . Clearly, periodic boundary conditions are prescribed for $s=0$ and $s=S$ ($0 \leq t \leq T$).

We now assume that appropriate discretizations for both problems (R) and (I) are given. We use the formal parameter h to characterize this composite mesh discretization. In particular, (R_h) , (I_h) denote the discrete problems and $\Omega_{R,h}$, $\Omega_{I,h}$ the corresponding grids. In the following, we do not distinguish between $\Omega_{R,h}$ -gridfunctions $u_{R,h}$ and the corresponding transformed gridfunctions $u_{0,h}$ on $\Omega_{0,h}$. The *composite mesh multigrid method* now is applied to composite mesh grid functions U_h (which consist of both, $u_{I,h}$ and $u_{R,h}$). In principle, this method proceeds as follows.

Apart from the h -discretization, a sequence of coarser composite meshes and corresponding discretizations has to be given. For the fine-to-coarse and the coarse-to-fine transfer, the usual operators are applied, but now individually to $\Omega_{I,h}$ and $\Omega_{R,h}$. The only essential difference compared to the usual MG methods consists in the smoothing part, in which the idea of Schwarz' alternating method is used. One smoothing step, e.g. for the h -grid, consists of the following four parts (assuming a first approximation \tilde{U}_h to be given):

- (1) interpolate the grid values $\tilde{u}_{I,h}$ to the discrete boundary points of Γ_0 ;
- (2) apply one relaxation step to $\tilde{u}_{R,h}$ with respect to the discrete problem (R_h) ;
- (3) interpolate the grid values $\tilde{u}_{R,h}$ to the discrete boundary points of Γ_I ;
- (4) apply one relaxation step to $\tilde{u}_{I,h}$ with respect to the discrete problem (I_h) .

Here the relaxation methods used in (2) and (4) for the discrete problems (R_h) and (I_h) , respectively, have to be chosen suitably. In particular, they should have comparable smoothing properties (smoothing factors). Otherwise one should allow a variable number of relaxation steps, which may be different in (2) and (4). The interpolation procedure used in (1) and (3) should be of a sufficiently high order. For a concrete example, see the following section.

11.3 Some results for a model problem

We want to specify the composite mesh MG method as described in the previous section to the very simple situation of Poisson's equation

$$-\Delta u = f^\Omega(x) \quad (x \in \Omega), \quad u = f^\Gamma(x) \quad (x \in \Gamma)$$

on the unit disk Ω . Clearly, for this utterly simple problem, one would not use the composite mesh approach in practice. For the purpose of demonstrating typical properties of the composite mesh MG method, however, this problem is quite suitable. In particular, one can discuss the question of overlapping and its influence on the convergence speed, the question of how to interpolate between the grids and the smoothing techniques. The results of these considerations are of a more general relevance. They are not restricted to the above problem.

We define Ω_0 using the orthogonal transformation Φ given by (11.6), i.e.

$$\Phi(s,t) = \begin{bmatrix} x_1(s,t) \\ x_2(s,t) \end{bmatrix} := \begin{bmatrix} (1-t)\cos(s) \\ (1-t)\sin(s) \end{bmatrix}$$

and the domain Ω_I to be an octagon. (An octagon has been chosen because it can easily be matched by a rectangular grid.) A composite mesh (for given $h=(h_I, h_S, h_T)$) is defined by the two grids $\Omega_{I,h}$ and $\Omega_{R,h}$ as shown in Figure 11.2. On $\Omega_{I,h}$ the Laplace-operator is discretized using the standard 5-point formula. We have Dirichlet boundary conditions along Γ_I . On $\Omega_{R,h}$ the transformed Laplace-operator

$$-\Delta^* := -\frac{1}{1-t} \left(\frac{1}{1-t} \frac{\partial^2}{\partial s^2} + \frac{\partial}{\partial t} ((1-t) \frac{\partial}{\partial t}) \right)$$

is discretized at a point $P = (s_i, t_j) = (ih_s, jh_t) \in \Omega_{R,h}$ by

$$\frac{1}{h_s^2(1-jh_t)} \begin{bmatrix} -q^2(1-jh_t-h_t/2) \\ -\frac{1}{1-jh_t} & -\Sigma & \frac{-1}{1-jh_t} \\ -q^2(1-jh_t+h_t/2) \end{bmatrix}$$

where $q := h_s/h_t$ and Σ denotes the sum of the four neighboring coefficients. We have Dirichlet boundary conditions along ∂_N , ∂_S and periodic boundary conditions at ∂_W and ∂_E (see Figure 11.2).

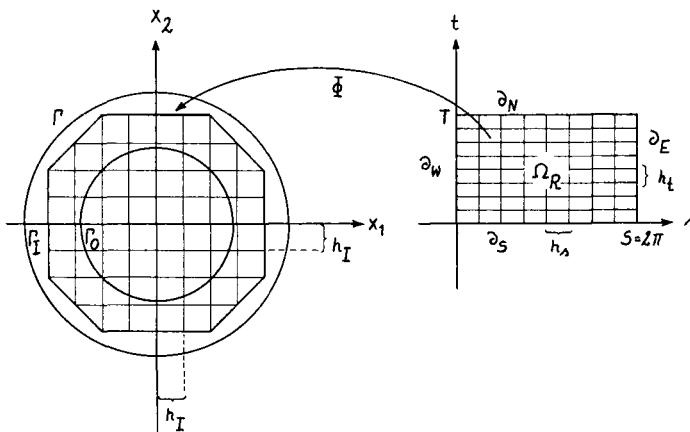


Figure 11.2: Composite mesh

The components of the MG algorithm used are the following:

- Coarsening, grid transfer, type of cycle: We apply standard coarsening (for both grids) and coarse-grid operators using the same discretization as on the fine grids. The transfer are done by full weighting and linear interpolation, respectively. The results given below are based on W-cycles.
- Smoothing: On smoothing step is performed as was described in the previous section. For (R_h) ZEBRA relaxation is used, with lines in the appropriate direction (depending on T and h_s/h_t). In the results given below, for (I_h) ZEBRA relaxation was used, also. (Here one could apply RB relaxation as well.) To connect the grids $\Omega_{R,h}$ and $\Omega_{I,h}$ (steps (1) and (3) in the previous section), cubic interpolation is used. This turns out to be necessary: using, for example, linear interpolation results in a much slower multigrid convergence.

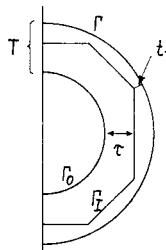


Figure 11.3:

In Table 11.1 we show the dependence of the multigrid algorithm on the size τ of the overlapping and on t_1 , the distance of Γ_I to the boundary Γ (see Figure 11.3). The second and third column show numerically computed convergence factors (for $v_1=v_2=1$). The main result is that these convergence factors are practically independent of τ and t_1 even for very small τ and large t_1 . We have already mentioned that this convergence behavior cannot be expected for the "naive" multigrid method which is closer related to Schwarz' alternating method. The corresponding convergence factors for this naive method are given in column 4 and 5 of Table 11.1: they do indeed show a high sensitivity with respect to τ and t_1 .

τ	composite mesh MG		"naive" MG	
	$t_1=0.0$	$t_1=0.5$	$t_1=0.0$	$t_1=0.5$
0.30	0.057	0.060	0.067	0.385
0.26	0.057	0.059	0.086	0.411
0.22	0.057	0.059	0.119	0.462
0.18	0.057	0.059	0.159	0.527
0.14	0.055	0.059	0.215	0.602
0.10	0.057	0.059	0.303	0.687
0.08	0.058	0.059	0.368	0.734
0.06	0.067	0.062	0.448	0.783

Table 11.1: Numerically computed convergence factors ($v_1=v_2=1$)

In Table 11.2 we finally compare the convergence of the above composite mesh multigrid method with the convergence of the SOR method (applied in a straightforward alternating manner following the lines of Schwarz' alternating method). The results given refer to the case $t_1 = 0$ and $T_2 = 0.4$ (cf. Figure 11.3) and to different meshsizes $h_I = 1/N$, $h_S = 2\pi/N_S$, $h_t = T/N_t$. To compare the numbers given, one has to take into account that the total computational work for one multigrid iteration step is larger than that of one SOR step by a factor of about 4.

N, N_t, N_s	composite mesh MG	SOR
8,16,16	0.057	0.837
16,32,32	0.057	0.906
32,64,64	0.057	0.968
64,128,128	0.057	0.988

Table 11.2: Numerically computed multigrid convergence factors compared with SOR convergence factors (per iteration step) for $t_1 = 0$, $T_2 = 0.4$.

Remark: The composite mesh approach may also be used for the multigrid treatment of boundary value problems on unbounded domains. For example, let a differential equation be given in $\Omega = \mathbb{R}^2 \setminus \bar{D}$, where D is some bounded domain; on ∂D and at infinity boundary conditions are assumed to be prescribed.

Clearly, there are several well-known ways to handle the unbounded domain and the boundary condition at infinity numerically (transformation techniques, e.g. of conformal mapping type, replacement of the unbounded domain by a bounded one etc.). In the composite mesh approach, the unbounded domain Ω is divided into two parts Ω_I and Ω_0 with (11.1). Here Ω_I is some bounded domain "around \bar{D} " and Ω_0 is some geometrically simple unbounded domain. For example, Ω_0 may be the exterior of a circle (which can immediately be represented as the image of a rectangle), see Figure 11.4. A composite mesh MG method can then be applied to this combined system on Ω_I and Ω_0 as described above.

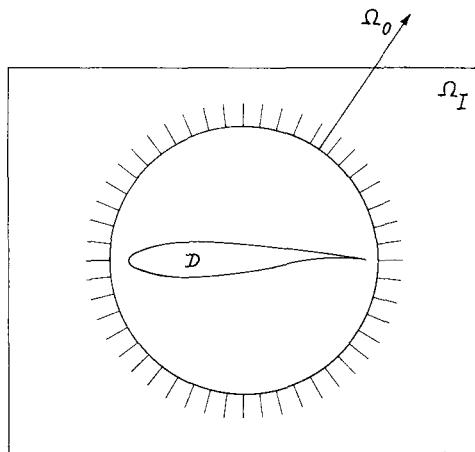


Figure 11.4:

APPENDIX

In this appendix we list a very specialized program MG $\emptyset\emptyset$ D from the MG $\emptyset\emptyset$ program collection (see Foerster, Witsch [36]). It solves model problem (P) by using an efficient multigrid method (see Section 8.1), which can be applied either as an iterative solver or in the FMG mode.

The following program is an exemplary driving routine for calling the multigrid subroutine MG $\emptyset\emptyset$ D:

```
C+++++MAIN PROGRAM FOR DEMONSTRATION OF MGGOOD
C
C      MAIN PROGRAM FOR DEMONSTRATION OF MGGOOD
C
C+++++DOUBLE PRECISION F, G, DIFMX, DEF MX, DIF, DEF
C      INTEGER IDIM, IER, M, N, NP, NY1, NY2, NCYCLE, NFMG, IGAM, INITF
C      DOUBLE PRECISION W(12000)
C      EXTERNAL F, G
C
C      IDIM = 12000
C      M = 6
C      NY1 = 2
C      NY2 = 1
C      NCYCLE = 1
C      NFMG = 1
C      IGAM = 1
C
C      CALL MGGOOD(M,NY1,NY2,NFMG,NCYCLE,IGAM,F,G,W,IDIM,INITF,IER)
C
C      IF (IER.EQ.0) GOTO 10
C      WRITE (6,9000) IER, IDIM
C      STOP
C 10      N = 2**M
C          NP = N+1
C          DIF = DIFMX(W,NP,G)
C          DEF = DEF MX(W,W(INITF+1),NP)
C          WRITE (6,9100)N, DIF, DEF
C          STOP
C 9000 FORMAT (16H *** ERROR, IER=, I3, 7H, IDIM=, I6, 4H ***)
C 9100 FORMAT (30H NUMBER OF INTERVALS = , I4 //
C *          30H MAXIMUM NORM OF THE ERROR = , D12.4 //
C *          30H MAXIMUM NORM OF THE DEFECT = , D12.4)
C      END
C
C.....*
C      DOUBLE PRECISION FUNCTION F(X,Y)
C      DOUBLE PRECISION X, Y, DSIN
C      F = 10.0D0*DSIN(3.0D0*X+Y)
C      RETURN
C      END
C
C.....*
C      DOUBLE PRECISION FUNCTION G(X,Y)
C      DOUBLE PRECISION X, Y, DSIN
C      G = DSIN(3.0D0*X+Y)
C      RETURN
C      END
```

```

*****
C      M G O O D      VERSION 20/04/82
*****
C
C      SUBROUTINE MGOOD(M, NY1, NY2, NFMG, NCYCLE, IGAM, F, G, W, IDIM,
*                      INITF, IER)
C
C      MULTIGRID MODULE FOR THE FAST SOLUTION OF POISSON'S EQUATION
C      WITH DIRICHLET BOUNDARY CONDITIONS ON THE UNIT SQUARE
C
C      DIFFERENTIAL EQUATION : -DELTA U(X,Y) = F(X,Y)
C      BOUNDARY CONDITION :          U(X,Y) = G(X,Y)
C
C      CYCLE STRUCTURE:
C
C      COARSENING: H, 2H, 4H
C      DIFFERENCE OPERATORS: USUAL 5-POINT STARS ON ALL GRIDS
C      RELAXATION: RED-BLACK
C      FINE-TO-COARSE: HALF INJECTION
C      COARSE-TO-FINE: BILINEAR INTERPOLATION
C
C      FULL MULTIGRID MODE:
C
C      FULL MULTIGRID INTERPOLATION USES GRID EQUATION
C      (4-TH ORDER)
C
C      INPUT:
C
C      M      NUMBER OF GRIDS (0 < M < 11). FOR GIVEN M:
C              NUMBER OF POINTS ON THE FINEST GRID (INCLUDING BOUNDARY
C              POINTS) = 2**M + 1 IN BOTH DIRECTIONS
C      NY1     NUMBER OF RELAXATIONS BEFORE COARSE-GRID CORRECTION
C              (NY1 > 0)
C      NY2     NUMBER OF RELAXATIONS AFTER COARSE-GRID CORRECTION
C              (NY2 > 0)
C      NCYCLE   NUMBER OF MULTIGRID ITERATIONS (NCYCLE > 0)
C      NFMG    .EQ. 0 : NCYCLE MULTIGRID ITERATIONS ARE PERFORMED
C              .NE. 0 : FMG-VERSION IS PERFORMED PLUS NCYCLE-1
C              ADDITIONAL MG ITERATIONS AFTERWARDS
C      IGAM    TYPE OF CYCLING (IGAM > 0).
C              E.G.: IGAM=1 FOR V-CYCLES, IGAM=2 FOR W-CYCLES
C      W       DOUBLE PRECISION WORK ARRAY OF DIMENSION IDIM
C      IDIM    DIMENSION OF W. APPROXIMATELY: IDIM > 2.8*4**M
C
C      EXTERNALS:
C
C      F      DOUBLE PRECISION FUNCTION F(X,Y), RIGHT HAND SIDE OF
C              THE DIFFERENTIAL EQUATION
C      G      DOUBLE PRECISION FUNCTION G(X,Y), BOUNDARY VALUES
C
C      REMARK: GRID #1 AND #M ARE THE FINEST AND COARSEST GRID USED,
C              RESPECTIVELY
C
C      OUTPUT:
C
C      IER    ERROR INDICATOR
C              = 0 NO ERRORS
C              = 1 INSUFFICIENT MEMORY, I.E. IDIM TOO SMALL.
C                  IN THIS CASE IDIM IS USED AS OUTPUT PARAMETER
C                  TO SHOW THE MINIMAL DIMENSION
C              = 2 M, NY1, NY2, NCYCLE OR IGAM WRONG
C              ONLY IN CASE IER=1 : MINIMAL LENGTH OF W
C              CF. DESCRIPTION OF W
C      W       W CONTAINS THE DISCRETE APPROXIMATION TO THE GIVEN
C              BOUNDARY VALUE PROBLEM ON THE FINEST GRID. THE GRID
C              VALUES ARE STORED ROWWISE FROM LEFT TO RIGHT AND
C              FROM BOTTOM TO TOP.
C              I.E. THE GRID VALUE CORRESPONDING TO THE GRID POINT
C                  (XI,YJ) :
C
C                  XI = (I-1)*H, YJ = (J-1)*H (H=1/N, N=2**M)
C
C      IS STORED AT
C
C                  W((J-1)*(N+1)+I) (0 < I,J < N+2)
C
C      THE CORRESPONDING VALUES OF THE RIGHT HAND SIDE ARE
C      STORED IN THE SAME MANNER AT
C
C                  W((J-1)*(N+1)+I+INITF).
C
C      THE REMAINING STORAGE CONTAINS COARSE GRID VALUES.

```

```

INTEGER IDC, IDF, IDR, IT, ITM, INITF, L, L1, LEV, LIN, NP,
* NY1, NY2, IGAM, IDIM, IER, NCYCLE, NFMG, M, ID(11), NPK(10),
DOUBLE PRÉCISIÓN W( 1), HK(10), DFLOAT
EXTERNAL F, G
DFLOAT(L1)=DBLE(FLOAT(L1))
IER = 0
C
* IF (M.LE.0 .OR. M.GE.11 .OR. NY1.LE.0 .OR. NY2.LE.0 .OR.
*      NCYCLE.LE.0 .OR. IGAM.LE.0 ) IER=2
* IF (IER.GT.0) GOTO 50
C
      DETERMINATION OF COARSER GRIDS
C
NP = 2**M + 1
HK(1) = 1.000/DFLOAT(NP-1)
ID(1) = 1
ID(2) = NP*NP + 1
NPK(1) = NP
DO 10 L=2,M
      NPK(L) = (NPK(L-1)+1)/2
      HK(L) = HK(L-1) + HK(L-1)
      ID(L+1) = ID(L) + NPK(L)**2
10 CONTINUE
C
      CHECK OF DIMENSIONS
C
INITF = ID(M+1)
IF (2*INITF.LE.IDIM) GOTO 20
IER = 1
IDIM = 2*INITF
GO TO 50
20 LIN = 1
IF (NFMG.NE.0) LIN = M
C
      SET UP ALL GRID VALUES NEEDED
C
CALL INIT1(NP, HK(1), F, G, W(1), W(INITF+1))
CALL INIT2(M, LIN, NPK, ID, W(1), W(INITF+1), INITF)
C
      (FULL) MULTIGRID PROCEDURE
C
DO 40 L=1,LIN
      LEV = LIN - L + 1
      ITM = 1
      IF (LEV.EQ.1) ITM = NCYCLE
      DO 30 IT=1,ITM
      CALL MG1(LEV, M, NY1, NY2, IGAM, NPK, ID, W, W(INITF+1), INITF)
30 CONTINUE
      IF (L.GE.LIN) GO TO 50
      IDF = ID(LEV-1)
      IDC = ID(LEV)
      IDR = IDF+INITF
      CALL INT4(NPK(LEV), NPK(LEV-1), W(IDC), W(IDF), W(IDR))
      CALL PUTZB(NPK(LEV), W(IDC))
40 CONTINUE
50 RETURN
END
C
C-----.
C
      SUBROUTINE INIT1(NP, H, F, G, U, FR)
C
      COMPUTES INITIAL VALUES ON THE FINEST GRID
C
INTEGER I, J, N, NP
DOUBLE PRÉCISIÓN FR(NP,NP), U(NP,NP), F, G, DFLOAT, H, H2, X, Y
DFLOAT(K) = DBLE(FLOAT(K))
C
H2 = H*H
N = NP - 1
DO 20 J=2,N
      Y = DFLOAT(J-1)*H
      U(1,J) = G(0.000,Y)
      DO 10 I=2,N
          X = DFLOAT(I-1)*H
          U(I,J) = 0.000
          FR(I,J) = H2*F(X,Y)
10   CONTINUE
      U(NP,J) = G(1.000,Y)
20 CONTINUE
      DO 30 I=1,NP
          X = DFLOAT(I-1)*H
          U(I,1) = G(X,0.000)
          U(I,NP) = G(X,1.000)
30 CONTINUE

```

```

30 CONTINUE
RETURN
END

C
C-----.
C
SUBROUTINE INIT2(M, LIN, NPK, ID, U, FR, IDIM)
      COMPUTES INITIAL VALUES ON THE COARSER GRIDS
      INTEGER IDIM, LIN, M, IDC, IDF, L, LH, ID(11), NPK(10)
      DOUBLE PRECISION FR(IDIM), U(IDIM)
      IF (LIN.EQ.1) GO TO 20
      TRANSFER OF F AND U TO COARSER GRIDS
      DO 10 L=2,LIN
         IDC = ID(L)
         IDF = ID(L-1)
         CALL TRANS(NPK(L), NPK(L-1), U(IDC), FR(IDC), U(IDF), FR(IDF))
10 CONTINUE
      PUT ZERO TO BOUNDARY VALUES FOR MULTIGRID CORRECTIONS
20 IF (LIN.EQ.M) GO TO 40
      LH = LIN + 1
      DO 30 L=LH,M
         IDC = ID(L)
         CALL PUTZB(NPK(L), U(IDC))
30 CONTINUE
40 RETURN
END

C
C-----.
C
SUBROUTINE TRANS(NPC, NPF, UC, FC, UF, FF)
      TRANSFER OF F AND U FROM GRID NPF TO NPC
      INTEGER NC, NPC, NPF, I, IF, J, JF
      DOUBLE PRECISION FC(NPC,NPC), FF(NPF,NPF), UC(NPC,NPC),
      *                   UF(NPF,NPF)
      NC = NPC - 1
      DO 20 J=2,NC
         JF = J + J - 1
         UC(1,J) = UF(1,JF)
         DO 10 I=2,NC
            IF = I + I - 1
            UC(I,J) = UF(IF,JF)
            FC(I,J) = 4.0D0*FF(IF,JF)
10 CONTINUE
         UC(NPC,J) = UF(NPF,JF)
20 CONTINUE
      DO 30 I=1,NPC
         IF = I + I - 1
         UC(I,1) = UF(IF,1)
         UC(I,NPC) = UF(IF,NPF)
30 CONTINUE
RETURN
END

C
C-----.
C
SUBROUTINE PUTZB(NPC, UC)
      PUTS ZERO TO BOUNDARY OF COARSER GRIDS
      INTEGER NC, NPC, I, J
      DOUBLE PRECISION UC(NPC,NPC)
      NC = NPC - 1
      DO 10 J=2,NC
         UC(1,J) = 0.0D0
         UC(NPC,J) = 0.0D0
10 CONTINUE
      DO 20 I=1,NPC
         UC(I,1) = 0.0D0
         UC(I,NPC) = 0.0D0
20 CONTINUE
RETURN
END

```

```

C SUBROUTINE MGI(LEV, M, NY1, NY2, IGAM, NPK, ID, U, FR, IDIM)
C          ONE MULTIGRID ITERATION STEP (ON ACTUAL FINEST GRID LEV)
C
C * INTEGER IDIM, LEV, M, NY1, NY2, ID(11), NPK(10), IDC, IDF, IZ,
C   K, IGAM, ICGAM(10)
C * DOUBLE PRÉCISION FR(IDIM), U(IDIM)
C
C      IZ = 1
C      DO 10 K=LEV,M
C         ICGAM(K) = 0
C 10 CONTINUE
C      K = LEV
C      IF (K.EQ.M) GO TO 30
C 20      IDF = ID(K)

CCC      RELAXATIONS BEFORE CGC

C      IZ = 1
C      IF (K.GT.LEV .AND. ICGAM(K).EQ.0) IZ = 0
C      CALL RELAX(NY1+NY1, IZ, NPK(K), U(IDF), FR(IDF), FR(1))
C      ICGAM(K) = ICGAM(K) + 1

CCC      RESIDUAL TRANSFER TO NEXT COARSER GRID

C      IDC = ID(K+1)
C      CALL RESTR(NPK(K+1), NPK(K), FR(IDC), U(IDF), FR(IDF))
C      K = K + 1
C      IF (K.LT.M) GOTO 20
C 30      IDC = ID(M)

CCC      EXACT SOLUTION ON COARSEST GRID

C      CALL RELAX(1, IZ, NPK(M), U(IDC), FR(IDC), FR(1))
C      IF (K.EQ.LEV) GOTO 50
C 40      K = K - 1
C      IDF = ID(K)
C      IDC = ID(K+1)

CCC      LINEAR INTERPOLATION TO NEXT FINER GRID

C      CALL INT2A(NPK(K+1), NPK(K), U(IDC), U(IDF))

CCC      RELAXATION AFTER CGC

C      CALL RELAX(NY2+NY2, 1, NPK(K), U(IDF), FR(IDF), FR(1))
C      IF (K.EQ.LEV) GOTO 50
C      IF (ICGAM(K).LT.IGAM) GOTO 20
C      ICGAM(K) = 0
C      GOTO 40
C 50      RETURN
C      END

C.....SUBROUTINE RELAX(ITM, IZ, NPF, UF, FF, W)
C
C      RED-BLACK RELAXATION ON GRID NPF
C
C      INTEGER ITM, IZ, NF, NPF, I, IS, IT, ITMAX, J
C      DOUBLE PRÉCISION FF(NPF,NPF), UF(NPF,NPF), W(NPF)
C
C      NF = NPF - 1
C      ITMAX = IABS(ITM)
C      IS = 2
C      IF (ITM.LT.0) IS = 3
C
C      DO 70 IT=1,ITMAX

CCC      RELAXATION OF EVEN POINTS FOR IS=2, OF ODD POINTS FOR IS=3

C      IF (IZ.NE.0 .OR. IT.GT.1) GO TO 30
C
C      ZERO STARTING VALUES

C      DO 20 J=2,NF
C         DO 10 I=IS,NF/2
C            UF(I,J) = 0.25DO*FF(I,J)
C 10      CONTINUE
C 20      CONTINUE

```

```

      IS = 5 - IS
20    CONTINUE
      GO TO 70
C
C      NON-ZERO STARTING VALUES
30    DO 40 I=IS,NF,2
      W(I) = UF(I,1) + UF(I-1,2)
40    CONTINUE
      DO 60 J=2,NF
        IF (IS.EQ.3) W(2) = UF(2,J) + UF(1,J+1)
        DO 50 I=IS,NF/2
          W(I+1) = UF(I+1,J) + UF(I,J+1)
          UF(I,J) = 0.25D0*(UF(I,J)+W(I)+W(I+1))
50    CONTINUE
      IS = 5 - IS
60    CONTINUE
70    CONTINUE
C
C      RETURN
      END
C
C.....SUBROUTINE RESTR(NPC, NPF, FC, UF, FF)
C
C      COMPUTATION OF THE DEFECT AND FINE-TO-COARSE TRANSFER
C      (HALF-INJECTION)
C
C      INTEGER NC, NPC, NPF, I, IF, J, JF
C      DOUBLE PRECISION FC(NPC,NPC), FF(NPF,NPF), UF(NPF,NPF), H
C
NC = NPC - 1
DO 20 J=2,NC
  JF = J + J - 1
  DO 10 I=2,NC
    IF = I + I - 1
    *   H = FF(IF,JF) - 4.0D0*UF(IF,JF) + UF(IF,JF-1) + UF(IF-1,JF)
    *     + UF(IF+1,JF) + UF(IF,JF+1)
    FC(I,J) = H + H
10  CONTINUE
20  CONTINUE
      RETURN
      END
C
C.....SUBROUTINE INT2A(NPC, NPF, UC, UF)
C
C      COARSE-TO-FINE TRANSFER (BILINEAR INTERPOLATION) AND CORRECTION
C
C      INTEGER NC, NPC, NPF, I, IF, J, JF
C      DOUBLE PRECISION UC(NPC,NPC), UF(NPF,NPF)
C
NC = NPC - 1
DO 20 J=2,NC
  JF = J + J - 1
  DO 10 I=1,NC
    IF = I + I
    UF(IF,JF) = UF(IF,JF) + 0.5D0*(UC(I,J)+UC(I+1,J))
10  CONTINUE
20  CONTINUE
  DO 40 J=1,NC
    JF = J + J
    DO 30 I=1,NC
      IF = I + I - 1
      UF(IF,JF) = UF(IF,JF) + 0.5D0*(UC(I,J)+UC(I,J+1))
30  CONTINUE
40  CONTINUE
      RETURN
      END
C
C.....SUBROUTINE INT4(NPC, NPF, UC, UF, FRF)
C
C      4-TH ORDER FMG-INTERPOLATION
C
C      INTEGER NC, NPC, NPF, I, IF, J, JF, JM
C      DOUBLE PRECISION FRF(NPF,NPF), UC(NPC,NPC), UF(NPF,NPF), HA, HN

```

```

C TRANSFER OF COARSE GRID VALUES TO THE FINE GRID
NC = NPC - 1
JF = 3
DO 20 J=2,NC
  IF = 3
  DO 10 I=2,NC
    UF(IF,JF) = UC(I,J)
    IF = IF + 2
  10 CONTINUE
  JF = JF + 2
20 CONTINUE

C COMPUTATION OF THE REMAINING EVEN POINTS BY USING THE
C ROTATED 5-POINT STAR
JF = 2
JM = 1
DO 40 J=2,NPC
  IF = 2
  HA = UC(1,JM) + UC(1,J)
  DO 30 I=2,NPC
    HN = UC(I,JM) + UC(I,J)
    UF(IF,JF) = 0.25D0*(FRF(IF,JF)+FRF(IF,JF)+HA+HN)
    HA = HN
    IF = IF + 2
  30 CONTINUE
  JM = JM + 1
  JF = JF + 2
40 CONTINUE

C COMPUTATION OF THE ODD POINTS BY ONE HALF (ODD) RELAXATION STEP
CALL RELAX(-1, 1, NPF, UF, FRF, FRF)
RETURN
END

C.....  

C DOUBLE PRECISION FUNCTION DIFMX(UC, NP, SOL)
C COMPUTES THE MAXIMUM NORM OF THE DIFFERENCE BETWEEN
C SOL (=SOLUTION OF THE BVP) AND THE VALUES IN UC
C
C INTEGER NP, I, J
C DOUBLE PRECISION UC(NP,NP), SOL, DFLOAT, H, X, Y
C DFLOAT(K) = DBLE(FLOAT(K))
C
C H = 1.0D0/DFLOAT(NP-1)
C DIFMX = 0.0D0
C DO 20 J=1,NP
C   Y = DFLOAT(J-1)*H
C   DO 10 I=1,NP
C     X = DFLOAT(I-1)*H
C     DIFMX = DMAX1(DIFMX,DABS(SOL(X,Y)-UC(I,J)))
C 10 CONTINUE
C 20 CONTINUE
C RETURN
C END

C.....  

C DOUBLE PRECISION FUNCTION DEFMX(UC, FC, NP)
C COMPUTES THE MAXIMUM NORM OF THE DEFECT
C
C INTEGER N, NP, I, J
C DOUBLE PRECISION UC(NP,NP), FC(NP,NP), D, HSQR
C
C N = NP - 1
C HSQR = DBLE(FLOAT(N*N))
C DEFMX = 0.0D0
C DO 20 J=2,N
C   DO 10 I=2,N
C     D = (FC(I,J)-4.0D0*UC(I,J)+UC(I-1,J)+UC(I,J-1)+UC(I,J+1)
C *           +UC(I+1,J))*HSQR
C     DEFMX = DMAX1(DEFMX,DABS(D))
C 10 CONTINUE
C 20 CONTINUE
C RETURN
C END

```

References:

1. Alcouffe, R.E.; Brandt, A.; Dendy, J.E.(Jr.); Painter, J.W.: *The multi-grid methods for the diffusion equation with strongly discontinuous coefficients.* Siam J. Sci. Stat. Comput., 2, pp. 430-454, 1981.
2. Allgower, E.L.; Böhmer, K.; McCormick, S.F.: *Discrete correction methods for operator equations.* Numerical Solution of Nonlinear Equations. Proceedings, Bremen 1980 (E.L. Allgower, K. Glashoff, H.O. Peitgen, eds.). Lecture Notes in Mathematics, 878, pp. 30-97. Springer-Verlag, Berlin, 1981.
3. Asselt, E.J. van: *The multi grid method and artificial viscosity.* This Proceedings.
4. Astrakhantsev, G.P.: *An iterative method of solving elliptic net problems.* U.S.S.R. Computational Math. and Math. Phys., 11 no. 2, pp. 171-182, 1971.
5. Auzinger, W.; Stetter, H.J.: *Defect correction and multigrid iterations.* This Proceedings.
6. Bakhvalov, N.S.: *On the convergence of a relaxation method with natural constraints on the elliptic operator.* U.S.S.R. Computational Math. and Math. Phys., 6 no. 5, pp. 101-135, 1966.
7. Bank, R.E.: *A multi-level iterative method for nonlinear elliptic equations.* Elliptic Problem Solvers (M.H. Schultz, ed.), pp. 1-16. Academic Press, New York, NY, 1981.
8. Bank, R.E.; Dupont, T.F.: *Analysis of a two-level scheme for solving finite element equations.* Report CNA-159, Center for Numerical Analysis, University of Texas at Austin, 1980.
9. Bank, R.E.; Sherman, A.H.: *An adaptive multi-level method for elliptic boundary value problems.* Computing, 26, pp. 91-105, 1981.
10. Becker, K.: *Mehrgitterverfahren zur Lösung der Helmholtz-Gleichung im Rechteck mit Neumannschen Randbedingungen.* Diplomarbeit, Institut für Angewandte Mathematik, Universität Bonn, 1981.

11. Beyn, W.J.; Lorenz, J.: *Spurious solution for discrete superlinear boundary value problems.* Computing, 28, pp. 43-51, 1982.
12. Börgers, C.: *Mehrgitterverfahren für eine Mehrstellendiskretisierung der Poisson-Gleichung und für eine zweidimensionale singulär gestörte Aufgabe.* Diplomarbeit, Institut für Angewandte Mathematik, Universität Bonn, 1981.
13. Braess, D.: *The convergence rate of a multigrid method with Gauss-Seidel relaxation for the Poisson equation.* This Proceedings.
14. Brand, K.: *Multigrid bibliography.* Gesellschaft für Mathematik und Datenverarbeitung, St. Augustin, 1982.
15. Brandt, A.: *Multi-level adaptive technique (MLAT) for fast numerical solution to boundary value problems.* Proceedings Third International Conference on Numerical Methods in Fluid Mechanics, Paris 1972 (H. Cabannes, R. Teman, eds.). Lecture Notes in Physics, 18, pp. 82-89. Springer-Verlag, Berlin, 1973.
16. Brandt, A.: *Multi-level adaptive techniques (MLAT). I. The multi-grid method.* Research Report RC 6026, IBM T.J. Watson Research Center, Yorktown Heights, NY, 1976.
17. Brandt, A.: *Multi-level adaptive solutions to boundary-value problems.* Math. Comp., 31, pp. 333-390, 1977.
18. Brandt, A.: *Multi-level adaptive techniques (MLAT) for partial differential equations: ideas and software.* Mathematical Software III. (J.R. Rice, ed.), pp. 277-318. Academic Press, New York, NY, 1977.
19. Brandt, A.: *Multi-level adaptive finite-element methods. I. Variational problems.* Special Topics of Applied Mathematics (J. Frehse, D. Pallaschke, U. Trottenberg, eds.), pp. 91-128. North-Holland Publishing Company, Amsterdam, 1979.
20. Brandt, A.: *Multi-level adaptive techniques (MLAT) for singular-perturbation problems.* Numerical Analysis of Singular Perturbation Problems (P.W. Hemker, J.J.H. Miller, eds.), pp. 53-142. Academic Press, London 1979.
21. Brandt, A.: *Numerical stability and fast solutions to boundary value problems.* Boundary and Interior Layers - Computational and Asymptotic Methods (J.J.H. Miller, ed.), pp. 29-49. Boole Press, Dublin, 1980.

22. Brandt, A.: *Multigrid solutions to steady-state compressible Navier-Stokes equations. I.* Preprint no. 492, Sonderforschungsbereich 72, Universität Bonn, 1981.
23. Brandt, A.: *Multigrid solvers for non-elliptic and singular-perturbation steady-state problems.* Research Report, Dept. of Applied Mathematics, Weizmann Institute of Science, Rehovot, 1981.
24. Brandt, A.: *Multigrid solvers on parallel computers.* Elliptic Problem Solvers (M.H. Schultz, ed.), pp. 39-84. Academic Press, New York, NY, 1981.
25. Brandt, A.: *Guide to multigrid development.* This Proceedings.
26. Brandt, A.; Dinar, N.: *Multi-grid solutions to elliptic flow problems.* Numerical Methods for Partial Differential Equations (S.V. Parter, ed.), pp. 53-147. Academic Press, New York, NY, 1979.
27. Brandt, A.; McCormick, S.F.; Ruge, J.: *Multigrid methods for differential eigenproblems.* Report, Dept. of Mathematics, Colorado State University, Ft. Collins, CO, 1981.
28. Brandt, A.; Ta'asan, S.: *Multi-grid methods for highly oscillatory problems.* Research Report, Dept. of Applied Mathematics, Weizmann Institute of Science, Rehovot, 1981.
29. Buzbee, B.L.; Golub, G.H.; Nielson, C.W.: *On direct methods for solving Poisson's equation.* SIAM J. Numer. Anal., 7, pp. 627-656, 1970.
30. Chan, T.F.; Keller, H.B.: *Arc-length continuation and multi-grid techniques for nonlinear elliptic eigenvalue problems.* Technical Report no. 197, Computer Science Dept., Yale University, New Haven, CT, 1981.
31. Collatz, L.: *The Numerical Treatment of Differential Equations.* Springer Verlag, Berlin, 1966.
32. Deconinck, H.; Hirsch, C.: *A multigrid finite element method for the transonic potential equation.* This Proceedings.
33. Dinar, N.: *Fast methods for the numerical solution of boundary-value problems.* PH.D. Thesis, Dept. of Applied Mathematics, Weizmann Institute of Science, Rehovot, 1978.

34. Fedorenko, R.P.: *A relaxation method for solving elliptic difference equations.* U.S.S.R. Computational Math. and Math. Phys., 1 no. 5, pp. 1092-1096, 1962.
35. Fedorenko, R.P.: *The speed of convergence of an iterative process.* U.S.S.R. Computational Math. and Math. Phys., 4 no. 3, pp. 227-235, 1964.
36. Foerster, H.; Witsch, K.: *Multigrid software for the solution of elliptic problems on rectangular domains: MGOO (Release 1).* This Proceedings.
37. Foerster, H.; Stüben, K.; Trottenberg, U.: *Non standard multigrid techniques using checkered relaxation and intermediate grids.* Elliptic Problems Solvers (M.H. Schultz, ed.), pp. 285-300. Academic Press, New York, NY, 1981.
38. Forsythe, G.E.; Wasow, W.R.: *Finite Difference methods for Partial Differential Equations.* John Wiley, New York-London, 1960.
39. Frederickson, P.O.: *Fast approximate inversion of large sparse linear systems.* Mathematics Report no. 7-75, Dept. of Mathematical Sciences, Lakehead University, Ontario, 1975.
40. Fuchs, L.: *Transonic flow computation by a multi-grid method.* Numerical Methods for the Computation of Inviscid Transonic Flows with Shock Waves (A. Rizzi, H. Viviand, eds.), pp. 58-65. Vieweg, Braunschweig, 1981.
41. Gary, J.: *The multigrid method applied to the collocation method.* SIAM J. Numer. Anal., 18, pp. 211-224, 1981.
42. Hackbusch, W.: *Ein iteratives Verfahren zur schnellen Auflösung elliptischer Randwertprobleme.* Report 76-12, Institut für Angewandte Mathematik, Universität Köln, 1976.
43. Hackbusch, W.: *On the convergence of a multi-grid iteration applied to finite element equations.* Report 77-8, Institut für Angewandte Mathematik, Universität Köln, 1977.
44. Hackbusch, W.: *On the computation of approximate eigenvalues and eigenfunctions of elliptic operators by means of a multi-grid method.* SIAM J. Numer. Anal., 16, pp. 201-215, 1979.

45. Hackbusch, W.: *Convergence of multi-grid iterations applied to difference equations.* Math. Comp., 34, pp. 425-440, 1980.
46. Hackbusch, W.: *Survey of convergence proofs for multigrid iterations.* Special Topics of Applied Mathematics (J. Frehse, D. Pallaschke, U. Trottenberg, eds.), pp. 151-164. North-Holland Publishing Company, Amsterdam, 1980.
47. Hackbusch, W.: *Bemerkungen zur iterierten Defektkorrektur und zu ihrer Kombination mit Mehrgitterverfahren.* Rev. Roumaine Math. Pures Appl., 26, pp. 1319-1329, 1981.
48. Hackbusch, W.: *Die schnelle Auflösung der Fredholmschen Integralgleichung zweiter Art.* Beiträge Numer. Math., 9, pp. 47-62, 1981.
49. Hackbusch, W.: *On the convergence of multi-grid iterations.* Beiträge Numer. Math., 9, pp. 213-239, 1981.
50. Hackbusch, W.: *Multi-grid convergence theory.* This Proceedings.
51. Hackbusch, W.: *On multi-grid iterations with defect correction.* This Proceedings.
52. Hackbusch, W.: *Introduction to multi-grid methods for the numerical solution of boundary value problems.* Computational Methods for Turbulent, Transonic and Viscous Flows (J.A. Essers, ed.). Hemisphere, to appear.
53. Hemker, P.W.: *Fourier analysis of gridfunctions, prolongations and restrictions.* Preprint NW 93/80, Dept. of Numerical Mathematics, Mathematical Centre, Amsterdam, 1980.
54. Hemker, P.W.: *The incomplete LU-decomposition as a relaxation method in multigrid algorithms.* Boundary and Interior Layers - Computational and Asymptotic Methods (J.J.H. Miller, ed.), pp. 306-311. Boole Press, Dublin, 1980.
55. Hemker, P.W.: *Introduction to multigrid methods.* Nieuw Archief voor Wiskunde (3), 29, pp. 71-101, 1981.
56. Hemker, P.W.: *Mixed defect correction iteration for the accurate solution of the convection diffusion equation.* This Proceedings.

57. Hemker, P.W.; Schippers, H.: *Multiple grid methods for the solution of Fredholm integral equations of the second kind.* Math. Comp., 36, pp. 215-232, 1981.
58. Hockney, R.W.; Eastwood, J.W.: *Computer Simulation Using Particles.* McGraw-Hill, New York, 1981.
59. Jameson, A.: *Acceleration of transonic potential flow calculations on arbitrary meshes by the multiple grid method.* Paper AIAA-79-1458, AIAA Fourth Computational Fluid Dynamics Conference, New York, NY, 1979.
60. Kettler, R.: *Analysis and comparison of relaxation schemes in robust multigrid and preconditioned conjugate gradient methods.* This proceedings.
61. Kettler, R.; Meijerink, J.A.: *A multigrid method and a combined multigrid-conjugate gradient method for elliptic problems with strongly discontinuous coefficients in general domains.* Shell Publication 604, KSEPL, Rijswijk, 1981.
62. Klunker, E.B.: *Contribution to methods for calculating the flow about thin lifting wings at transonic speeds - analytic expressions for the far field.* NASA Technical Note D-6530, Langley Research Center, 1971.
63. Kroll, N.: *Direkte Anwendungen von Mehrgittertechniken auf parabolische Anfangsrandwertaufgaben.* Diplomarbeit, Institut für Angewandte Mathematik, Universität Bonn, 1981.
64. Kronsjö, L.: *A note on the "nested iterations" method.* BIT, 15, pp. 107-110, 1975.
65. Kronsjö, L.; Dahlquist, G.: *On the design of nested iterations for elliptic difference equations.* BIT, 11, pp. 63-71, 1972.
66. Laasonen, P.: *On the discretization error of the Dirichlet problem in a plane region with corners.* Ann. Acad. Sci. Fenn. Ser. AI Math. Dissertationes, 408, pp. 1-16, 1967.
67. Linden, J.; Trottenberg, U.; Witsch, K.: *Multigrid computation of the pressure of an incompressible fluid in a rotating spherical gap.* Proceedings of the Fourth GAMM-Conference on Numerical Methods in Fluid Mechanics (H. Viviand, ed.), pp. 183-193. Vieweg, Braunschweig, 1982.

68. Linden, J.: *Mehrgitterverfahren für die Poisson-Gleichung in Kreis und Ringgebiet unter Verwendung lokaler Koordinaten.* Diplomarbeit, Institut für Angewandte Mathematik, Universität Bonn, 1981.
69. Magnus, R.J.; Gallaher, W.H.: *Flow over airfoils in the transonic regime - computer programs.* AFFDL-TR-70-16, Vol. II, U.S. Air Force, 1970.
70. McCormick, S.F.; Ruge, J.: *Multigrid methods for variational problems.* Preprint, Dept. of Mathematics, Colorado State University, Ft. Collins, CO, 1981 (submitted to: SIAM J. Num. Anal.).
71. Meis, T.; Marcowitz, U.: *Numerical Solution of Partial Differential Equations.* New York-Heidelberg-Berlin, Springer, 1981.
72. Meis, T.; Branca, H.W.: *Schnelle Lösung von Randwertaufgaben.* Z. Angew. Math. Mech., 62, 1982.
73. Meis, T.; Lehmann, H.; Michael, H.: *Application of the multigrid method to a nonlinear indefinite problem.* This Proceedings.
74. Mittelmann, H.D.: *Multi-grid methods for simple bifurcation problems.* This Proceedings.
75. Nicolaides, R.A.: *On multiple grid and related techniques for solving discrete elliptic systems.* J. Comput. Phys., 19, pp. 418-431, 1975.
76. Nicolaides, R.A.: *On the ℓ^2 convergence of an algorithm for solving finite element equation.* Math. Comp., 31, pp. 892-906, 1977.
77. Nicolaides, R.A.: *On multi-grid convergence in the indefinite case.* Math. Comp., 32, pp. 1082-1086, 1978.
78. Nicolaides, R.A.: *On some theoretical and practical aspects of multigrid methods.* Math. Comp., 33, pp. 933-952, 1979.
79. Ophir, D.: *Language for processes of numerical solutions to differential equations.* PH.D. Thesis, Dept. of Applied Mathematics, Weizmann Institute of Science, Rehovot, 1978.
80. Ortega, J.M.; Rheinboldt, W.C.: *Iterative Solution of Nonlinear Equations in Several Variables.* New York-London, Academic Press, 1970.

81. Proskurowski, W.; Widlund, O.: *On the numerical solution of Helmholtz' equation by the capacitance matrix method.* Math. Comp., 30, pp. 433-468, 1976.
82. Ries, M.; Trottenberg, U.; Winter, G.: *A note on MGR methods.* Linear Algebra Appl., to appear, 1982.
83. Ruge, J.W.: *Multigrid methods for differential eigenvalue and variational problems and multigrid simulation.* PH.D. Thesis, Dept. of Mathematics, Colorado State University, Ft. Collins, CO, 1981.
84. Schmidt, W.; Jameson, A.: *Applications of multi-grid methods for transonic flow calculations.* This Proceedings.
85. Schröder, J.: *Zur Lösung von Potentialaufgaben mit Hilfe des Differenzenverfahrens.* ZAMM, 34, pp. 241-253, 1954.
86. Schröder, J.: *Beiträge zum Differenzenverfahren bei Randwertaufgaben.* Habilitationsschrift, Hannover 1955.
87. Schröder, J.; Trottenberg, U.: *Reduktionsverfahren für Differenzengleichungen bei Randwertaufgaben I.* Numer. Math., 22, pp. 37-68, 1973.
88. Schröder, J.; Trottenberg, U.; Witsch, K.: *On fast Poisson solvers and applications.* Numerical Treatment of Partial Differential Equations (R. Bulirsch, R.D. Grigorieff, J. Schröder, eds.). Proceedings of a Conference held at Oberwolfach, July 4-10, 1976. Lecture Notes in Mathematics, 631, pp. 153-187, Springer-Verlag, Berlin, 1978.
89. Shortley, G.H.; Weller, R.: *Numerical solution of Laplace's equation.* J. Appl. Phys., 9, pp. 334-348, 1938.
90. Solchenbach, K.; Stüben, K.; Trottenberg, U.; Witsch, K.: *Efficient solution of a nonlinear heat conduction problem by use of fast reduction and multigrid methods.* Preprint no. 421, Sonderforschungsbereich 72, Universität Bonn, 1980.
91. South, J.C.(Jr.); Brandt, A.: *Application of a multi-level grid method to transonic flow calculations.* Transonic Flow Problems in Turbomachinery (T.C. Adamson, M.F. Platzer, eds.). Hemisphere, Washington, DC, 1977.

92. Southwell, R.V.: *Stress calculation in frameworks by the method of systematic relaxation of constraints. I, II*, Proc. Roy. Soc. London Ser. A, 151, pp. 56-95, 1935.
93. Southwell, R.V.: *Relaxation Methods in Theoretical Physics*. Clarendon Press, Oxford, 1946.
94. Stiefel, E.: *Über einige Methoden der Relaxationsrechnung*. Z. Angew. Math. Phys., 3, pp. 1-33, 1952.
95. Starius, G.C.: *Composite mesh difference methods for elliptic boundary value problems*. Numer. Math., 28, pp. 242-258, 1977.
96. Stetter, H.J.: *The defect correction principle and discretization methods*. Numer. Math., 29, pp. 425-443, 1978.
97. Stüben, K.: *MGØ1: A multi-grid program to solve $\Delta U - c(x,y)U = f(x,y)$ (on Ω), $U = g(x,y)$ (on $\partial\Omega$)*, on nonrectangular bounded domains Ω . IMA-Report no. 82.02.02, Gesellschaft für Mathematik und Datenverarbeitung, St. Augustin, 1982.
98. Stüben, K.; Trottenberg, U.: *On the construction of fast solvers for elliptic equations*. Computational Fluid Dynamics. Lecture Series 1982-04, von Karman Institute for Fluid Dynamics, Rhode-Saint-Genese, 1982.
99. Temperton, C.: *Algorithms for the solution of cyclic tridiagonal systems*. J. Comput. Phys., 19, pp. 317-323, 1975.
100. Thole, C.A.: *Beiträge zur Fourieranalyse von Mehrgittermethoden: V-cycle, ILU-GLättung, anisotrope Operatoren*. Diplomarbeit, Institut für Angewandte Mathematik, Universität Bonn, to appear.
101. Törnig, W.: *Numerische Mathematik für Ingenieure und Physiker*, Band 1, Springer-Verlag, Berlin, 1979.
102. Trottenberg, U.: *Reduction methods for solving discrete elliptic boundary value problems - an approach in matrix terminology*. Fast Elliptic Solvers (U. Schumann, ed.). Advance Publications, London, 1977.
103. Trottenberg, U.: *Schnelle Lösung partieller Differentialgleichungen - Idee und Bedeutung des Mehrgitterprinzips*. Jahresbericht 1980/81, Gesellschaft für Mathematik und Datenverarbeitung, pp. 85-95, Bonn, 1981.

104. Trottenberg, U.; Witsch, K.: *Zur Kondition diskreter elliptischer Randwertaufgaben.* GMD-Studien no. 60. Gesellschaft für Mathematik und Datenverarbeitung, St. Augustin, 1981.
105. Varga, R.S.: *Matrix Iterative Analysis.* Englewood Cliffs, Prentice Hall, 1962.
106. Verfürth, R.: *The contraction number of a multigrid method with mesh ratio 2 for solving Poisson's equation.* Report, Institut für Angewandte Mathematik, Ruhr-Universität Bochum, 1982.
107. Wesseling, P.: *Numerical solution of stationary Navier-Stokes equation by means of a multiple grid method and Newton iteration.* Report NA-18, Dept. of Mathematics, Delft University of Technology, Delft, 1977.
108. Wesseling, P.: *The rate of convergence of a multiple grid method.* Numerical Analysis. Proceedings, Dundee 1979 (G.A. Watson, ed.). Lecture Notes in Mathematics, 773, pp. 164-184. Springer-Verlag, Berlin, 1980.
109. Wesseling, P.: *Theoretical and practical aspects of a multigrid method.* Report NA-37, Dept. of Mathematics, Delft University of Technology, Delft, 1980.
110. Wesseling, P.: *A robust and efficient multigrid method.* This Proceedings.
111. Winter, G.: *Fourieranalyse zur Konstruktion schneller MGR-Verfahren.* Dissertation, Institut für Angewandte Mathematik, Universität Bonn, to appear.