Multigrid Methods for Semiconductor Device Simulation

Multigrid Methods for Semiconductor Device Simulation

ACADEMISCH PROEFSCHRIFT

ter verkrijging van de graad van doctor
aan de Universiteit van Amsterdam,
op gezag van de Rector Magnificus,
prof.dr. P.W.M. de Meijer,
in het openbaar te verdedigen in de Aula der Universiteit
(Oude Lutherse Kerk, ingang Singel 411, hoek Spui),
op woensdag 4 maart 1992, te 15.00 uur

door

Johannes Molenaar

geboren te Zwanenburg

Centrum voor Wiskunde en Informatica
Amsterdam
1992

Multirooster methoden voor halfgeleider device simulatie

Promotor : prof.dr. P.W. Hemker
Faculteit : Wiskunde en Informatica

This work has been supported by the Dutch Innovative Research Program (IOP-IC Technology)

Contents

Chapter 1

Introduction

1.1. General introduction

During the last three decades the development of semiconductor devices has been very fast. The first integrated circuits, that became commercially available in the early 1960s, contained just a few devices, whereas today it is possible to produce integrated circuits that contain tens of milliones of devices per single chip. This progress was possible by reducing the dimensions of the individual devices. In the development of new devices increasing use has been made of simulations. As the costs of computer resources are going down (thanks to the same miniaturization!) these simulations have become much cheaper than experimental investigations. Moreover simulations are more flexible, so the use of simulations may yield a better end product, because it is feasible to consider many more options. We distinguish two types of simulations: process simulation and device simulation.

In process simulation the various processing steps in the fabrication of a device are studied. The result of a process simulation is the doping profile and the geometry of the device, which both are used as input data for a device simulation. The objective of device simulation is to predict the electric behavior of the device, e.g. the electric field and the current densities within the device, and the current voltage characteristics of the device. In this thesis we only consider computations related with device simulation; for an introduction to process simulation the reader is referred to [42].

Basically there are three approaches to device simulation. Early device modeling was based on the division of the interior of the device into a few different regions, in which closed form solutions are obtained by making some (very) restrictive assumptions. The solutions in the different regions are then matched at the boundaries to produce a global solution (cf. [45, 47]). This classical approach may give an understanding of the operation of the device, but it has limited applicability and is not particularly suited for engineering purposes.

The statistical Monte Carlo technique makes it possible to include details of virtually any physical process (cf. [22]). However, the main disadvantage of the Monte Carlo method is that it requires enormous amounts of computing time, which makes it, at present, unsuitable for an engineering environment.

We consider the approach of numerically solving the "basic" semiconductor equations, that consist of the Poisson equation for the electric field, the continuity equations for electrons and holes, and the drift-diffusion approximation for the electron and hole current densities. This set of equations was first proposed by Van Roosbroeck [37] in 1950.

The first computational solution of these equations, for a one dimensional bipolar transistor, was presented in 1964 by Gummel [15]. Soon it became clear that standard discretizations are inappropriate for the semiconductor equations because of the stiffness of the equations. The problem was overcome by Scharfetter and Gummel [39], who developed a special purpose discretization, that is used up to now. A survey of papers about the computational solution of the semiconductor equations is found in [42].

Today many programs are available for two dimensional device simulation; we only mention MINIMOS [43], BAMBI [10], CURRY [23], PISCES [31] and TRENDY [40, 52]. Also some three dimensional device simulators have been reported (cf. [17, 18, 30, 38]), however for accurate computations these require extreme amounts of computing power.

In order to reduce the computing time for accurate and complex simulations we consider in this thesis the solution of the stationary two-dimensional semiconductor equations by the nonlinear multigrid method. Multigrid methods were developed in the late 1970s by Brandt, Hackbusch and others; for a multigrid bibliography see [4]. The major advantage of multigrid over other solution methods is that it has optimal complexity with respeet to both the amount of computational work and to the memory usage. Several attempts have already been made to explore the possibilities of multigrid techniques for the solution of the semiconductor equations (cf. [14, 20, 44, 46, 53]). However, up to now the question of whether the multigrid technique is feasible for practical applications was still open. As the semiconductor equations are strongly nonlinear and badly scaled, it is not at all straightforward to apply the multigrid method for these equations. In this thesis we show that it is indeed possible to use multigrid for practical semiconductor device simulation.

1.2. The semiconductor device equations

In this Section we formulate the system of partial differential equations that make up the basic device equations, and we equip the system with boundary conditions that represent the interaction of the device and the outside world. Then we discuss various possible choices of the dependent variables and their scaling.

The device equations can be derived from the Maxwell equations, some relations from solid state physics and many simplifying assumptions. For a derivation the reader is referred to [42]. Much freedom is left in the assumptions, so the material properties can be modeled at different levels of sophistication. We study the following system of partial differential equations that still contain most of the essential difficulties for numerical simulation:

$$\begin{array}{lll}
\text{one of the Maxwell equations:} & \operatorname{div}\mathbf{D} = q(\,p - n + D\,), & (1.1\text{a}) \\
\text{electron continuity equation:} & \operatorname{div}\mathbf{j}_n - q\dfrac{\partial n}{\partial t} = +qR\,, & (1.1\text{b}) \\
\text{hole continuity equation:} & \operatorname{div}\mathbf{j}_p + q\dfrac{\partial p}{\partial t} = -qR\,, & (1.1\text{c}) \\
\text{electric displacement current:} & \mathbf{D} = -\epsilon\,\operatorname{grad}\psi, & (1.1\text{d}) \\
\text{electron current relation:} & \mathbf{j}_n = +q\mu_n(U_T\operatorname{grad}n - n\operatorname{grad}\psi), & (1.1\text{e}) \\
\text{hole current relation:} & \mathbf{j}_p = -q\mu_p(U_T\operatorname{grad}p + p\operatorname{grad}\psi). & (1.1\text{f})
\end{array}$$

The dependent variables are the electrostatic potential ψ, and the electron and hole concentrations n and p, respectively, $\mathbf{D}$ denotes the electric displacement current and $\mathbf{j}_n$, $\mathbf{j}_p$ are the electron and hole current densities, respectively. D is the given dope function and R represents the net recombination-generation rate of electrons and holes; q, U_T, μ_n, μ_p are the elementary charge, the thermal voltage and the mobilities of electrons and holes, respectively. The permittivity ϵ of a medium is given by $\epsilon = \epsilon_R\epsilon_0$, with ϵ_0 the permittivity of vacuum and ϵ_R the relative permittivity of the medium. For simplicity we assume that the mobilities μ_n, μ_p are constant, and we treat ϵ as a piecewise constant scalar function, i.e. we assume that the medium is isotropic. (Numerical values for the physical constants are given in Appendix B.) Moreover, we only consider the stationary problem, so we set

$$\frac{\partial n}{\partial t} = \frac{\partial p}{\partial t} = 0 \tag{1.2}$$

in the continuity equations (1.1b) and (1.1c).

Next we consider the boundary and interface conditions. In general there are semiconductor/conductor interfaces (contacts), semiconductor/isolator interfaces, and outside boundaries. At the contacts we assume (cf. [42]) thermal equilibrium

$$np = n_i^2 \tag{1.3a}$$

and a vanishing space charge

$$p - n + D = 0, \tag{1.3b}$$

with n_i the intrinsic density of free charge carriers. Dirichlet boundary conditions for n and p follow from (1.3):

$$n = \tfrac{1}{2}(+D + \sqrt{D^2 + 4n_i^2}\,), \tag{1.4a}$$

$$p = \tfrac{1}{2}(-D + \sqrt{D^2 + 4n_i^2}\,). \tag{1.4b}$$

The boundary potential ψ at the contacts is the sum of the applied voltage V_{appl} and the so called built-in voltage V_{bi}, which is produced by the doping:

$$\psi = V_{\text{appl}} + V_{\text{bi}}, \tag{1.4c}$$

with

$$V_{\text{bi}} = U_T \operatorname{arsinh}\left[\frac{D}{2n_i}\right]. \tag{1.5}$$

In our model, at the interfaces between semiconductor and isolator regions possible surface charges and surface recombination effects are neglected. Hence the electric potential and displacement current in the direction orthogonal to the interface are assumed to be continuous,

$$[\psi]_{\delta\Omega_I} = [\mathbf{D}\cdot\mathbf{n}]_{\delta\Omega_I} = 0, \tag{1.6a}$$

with $\mathbf{n}$ the unit vector normal to the interface and $[f]_{\delta\Omega_I}$ the jump in the function f across the interface $\delta\Omega_I$. Furthermore we assume that no currents flow through the interface,

$$\mathbf{j}_n \cdot \mathbf{n} = \mathbf{j}_p \cdot \mathbf{n} = 0. \tag{1.6b}$$

At the outside boundaries we always assume a vanishing outward electric field and vanishing outward current densities,

$$\operatorname{grad}\psi \cdot \mathbf{n} = \mathbf{j}_n \cdot \mathbf{n} = \mathbf{j}_p \cdot \mathbf{n} = 0. \tag{1.7}$$

So far we used ψ, n and p as the dependent variables. If we assume that the Boltzmann statistics hold, we can write the carrier concentrations as (cf. [42])

$$n = n_i\, e^{\frac{\psi-\phi_n}{U_T}}, \tag{1.8a}$$

$$p = n_i\, e^{\frac{\phi_p-\psi}{U_T}}, \tag{1.8b}$$

with ϕ_n and ϕ_p the electron and hole quasi-Fermi potentials. In principle we can regard (1.8) as a simple change of dependent variables ((ϕ_n, ϕ_p) instead of (n, p)) that preserves the non-negativity of (n, p); the validity of the Boltzmann statistics is only necessary to interpret ϕ_n and ϕ_p as the quasi-Fermi potentials. We notice that the applied voltages V_{appl} at the contacts are precisely the Dirichlet boundary conditions for the variables ϕ_n and ϕ_p. Expressed in terms of (ψ, ϕ_n, ϕ_p) the current relations (1.1e-f) are written as

$$\mathbf{j}_n = -q\mu_n n \operatorname{grad}\phi_n = -q\mu_n n_i e^{\frac{\psi-\phi_n}{U_T}} \operatorname{grad}\phi_n, \tag{1.9a}$$

$$\mathbf{j}_p = -q\mu_p p \operatorname{grad}\phi_p = -q\mu_p n_i e^{\frac{\phi_p-\psi}{U_T}} \operatorname{grad}\phi_p. \tag{1.9b}$$

Another possible set of variables are the so called Slotboom variables (ψ, Φ_n, Φ_p) that are defined by

$$\Phi_n = e^{-\frac{\phi_n}{U_T}}, \tag{1.10a}$$

$$\Phi_p = e^{+\frac{\phi_p}{U_T}}. \tag{1.10b}$$

Expressed in these variables the semiconductor equations (1.1) appear in symmetric positive definite form:

$$-\operatorname{div}(\epsilon \operatorname{grad} \psi) + qn_i e^{+\psi}\Phi_n - qn_i e^{-\psi}\Phi_p = qD, \tag{1.11a}$$

$$-\operatorname{div}(\mu_n U_T n_i q e^{+\frac{\psi}{U_T}} \operatorname{grad} \Phi_n) + qR = 0, \tag{1.11b}$$

$$-\operatorname{div}(\mu_p U_T n_i q e^{-\frac{\psi}{U_T}} \operatorname{grad} \Phi_p) + qR = 0. \tag{1.11c}$$

We proceed by discussing some of the advantages and disadvantages for the different choices of the dependent variables. There is a tradeoff between the range of the values assumed by the variables and the nonlinearity of the equations. Typical ranges of the different variables are (cf. [32], Table VII)

Variable	Range for $V_{\mathrm{appl}} = 5$	Range for $V_{\mathrm{appl}} = 20$
ψ	$[-0.5, 5.5]$	$[-0.5, 20.5]$
ϕ_n, ϕ_p	$[0, 5]$	$[0, 20]$
n, p	$[10^{+1}, 10^{+21}]$	$[10^{0}, 10^{+21}]$
Φ_n, Φ_p	$[10^{-84}, 10^{+84}]$	$[10^{-336}, 10^{+336}]$

For constant μ and R each of the continuity equations expressed in n and p or in the Slotboom variables Φ_n and Φ_p are linear in the associated variable. Mathematically it is attractive to use the Slotboom variables (ψ, Φ_n, Φ_p) because the equations appear in symmetric positive definite form, but for numerical purposes they are quite useless because of the range of possible values assumed by Φ_n, Φ_p. On the other hand, the values assumed by the variable set (ψ, ϕ_n, ϕ_p) is of the same order as the applied voltage, but the semiconductor equations expressed in (ψ, ϕ_n, ϕ_p) are strongly nonlinear. So we have a favorite set of variables for the operator, i.e. the variable set (ψ, Φ_n, Φ_p), and a favorite set of variables for doing practical calculations, i.e. the variable set (ψ, ϕ_n, ϕ_p). For an elaborate discussion of the choice of variables see [32].

To simplify the notation we use the following scaling in the sequel:

Symbol	Scaling factor
ψ, ϕ_n, ϕ_p	U_T
n, p, D	n_i
R	q^{-1}
μ_n, μ_p	$(U_T n_i q)^{-1}$

By this scaling the equations, expressed in Slotboom variables, read

$$\operatorname{div} \mathbf{j}_\psi = e^{-\psi}\Phi_p - e^{+\psi}\Phi_n + D, \tag{1.12a}$$

$$\operatorname{div} \mathbf{j}_n = +R\,, \tag{1.12b}$$

$$\operatorname{div} \mathbf{j}_p = -R\,, \tag{1.12c}$$

$$\mathbf{j}_\psi = -\mu_\psi \operatorname{grad} \psi, \tag{1.12d}$$

$$\mathbf{j}_n = +\mu_n e^{+\psi} \operatorname{grad} \Phi_n, \tag{1.12e}$$

$$\mathbf{j}_p = -\mu_p e^{-\psi} \operatorname{grad} \Phi_p, \tag{1.12f}$$

with

$$\mathbf{j}_\psi = \frac{\mathbf{D}}{n_i q}, \tag{1.13}$$

$$\mu_\psi = \frac{U_T}{n_i q} \epsilon. \tag{1.14}$$

1.3. Discretization

In any numerical approach for the solution of partial differential equations we can distinguish two steps. First the continuous equations are replaced by a "discrete" system of (nonlinear) algebraic equations, whose solution represents in one way or another the approximate solution of the continuous problem. Then, because it is usually impossible to solve exactly the system of nonlinear equations obtained by discretization, an iterative scheme is used to approximate the solution of the discrete system of equations. In this Section we discuss some discretization methods for the semiconductor equations, and in the next Section we mention some of the iterative schemes that are being used for the numerical solution of the discretized semiconductor equations.

Basically three discretization methods are used for the semiconductor equations: the finite difference method, the finite volume (box) method and the finite element method. Although these three approaches are different in their origins, they often lead to equivalent systems of discrete equations. In all cases measures should be taken to take care of possible dominating convection terms in the elliptic equations. This is usually done by a scheme of Scharfetter-Gummel type.

In the method of finite differences the domain is covered by a regular rectangular grid and all derivatives in the strong formulation of the differential equations are replaced by difference quotients at the grid points. In order to introduce the Scharfetter-Gummel discretization we follow their approach for the one dimensional case (cf. [39]). We consider the semiconductor equations expressed in the variable set (ψ, n, p) and assume a uniform grid with mesh width h. For a grid point $\mathbf{x}^M$, with nearest neighbors $\mathbf{x}^L$ and $\mathbf{x}^R$, we obtain

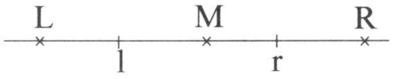


Figure 1.1. Numbering of cells for 1D Scharfetter-Gummel discretization.

$$\frac{j_\psi^r - j_\psi^l}{h} = p - n + D, \tag{1.15a}$$

$$\frac{j_n^r - j_n^l}{h} = +R\,, \tag{1.15b}$$

$$\frac{j_p^r - j_p^l}{h} = -R\,, \tag{1.15c}$$

with $\mathbf{x}^l$, $\mathbf{x}^r$ midway between the major grid points (cf. Figure 1.1). For Poisson's equation we replace j_ψ^r, j_ψ^l by standard central differences

$$j_\psi^r = -\mu_\psi \frac{\psi^R - \psi^M}{h}, \qquad j_\psi^l = -\mu_\psi \frac{\psi^M - \psi^L}{h}. \tag{1.16}$$

For the continuity equations of the holes,

$$\mathbf{j}_p = -\mu_p(p\,\mathrm{grad}\,\psi + \mathrm{grad}\,p), \tag{1.17}$$

we can not use the standard difference approximation

$$\begin{aligned} j_p^r &= -\mu_p\Big(\frac{p^M + p^R}{2}\frac{\psi^R - \psi^M}{h} + \frac{p^R - p^M}{h}\Big) \\ &= \frac{\mu_p}{2h}(p^M(2-\psi^R+\psi^M) - p^R(2+\psi^R-\psi^M)). \end{aligned}$$

If ψ differs more than 2 between two adjacent grid points, there is loss of diagonal dominance, so we may expect numerical instability.

To obtain a stable scheme, Scharfetter and Gummel assume that $\mathbf{j}_\psi$ and $\mathbf{j}_p$ are constant between the grid points, and treat (1.17) as an ordinary differential equation for p; $\mathbf{j}_p$ between two grid points is determined by using the values of p at the grid points as boundary conditions. Multiplication of (1.17) by $\exp((\mathbf{x}-\mathbf{x}^M)\mathrm{grad}\,\psi)$, and integration from $\mathbf{x}^M$ to $\mathbf{x}^R$ yields

$$p^R e^{\psi^R - \psi^M} - p^M + \frac{j_p^r}{\mu_p}\left(\frac{e^{\psi^R - \psi^M} - 1}{\dfrac{\psi^R - \psi^M}{h}}\right) = 0,$$

so

$$\begin{aligned} j_p^r &= \mu_p \frac{\psi^R - \psi^M}{h}\left[p^M \frac{1}{e^{\psi^R - \psi^M} - 1} - p^R \frac{e^{\psi^R - \psi^M}}{e^{\psi^R - \psi^M} - 1}\right] \\ &= \frac{\mu_p}{h}\left[p^M B(\psi^R - \psi^M) - p^R B(\psi^M - \psi^R)\right], \end{aligned} \tag{1.18}$$

with

$$B(x) = \frac{x}{e^x - 1}, \tag{1.19}$$

the so called Bernoulli function. Analogously we obtain for the continuity equations for electrons

$$j_n^r = \frac{\mu_n}{h}\left[-n^M B(\psi^M - \psi^R) + n^R B(\psi^R - \psi^M)\right]. \tag{1.20}$$

As $B(x) > 0$ there is no loss of diagonal dominance if big jumps in ψ occur. In fact, the Scharfetter-Gummel scheme can be considered as an exponentially fitted upwind discretization. The finite difference Scharfetter-Gummel discretization in two space dimensions is a direct application of the one-dimensional scheme along the two coordinate directions (cf. [42]).

To apply the finite volume discretization for the semiconductor equations we notice that all three equations (1.1a-c) are in divergence form:

$$\operatorname{div}\mathbf{j} = S. \tag{1.21}$$

The domain is divided in a number of boxes (that are not necessarily rectangular) and integration of (1.21) over a box Ω^i yields

$$\int_{\Omega^i} \operatorname{div}\mathbf{j}\, d\Omega = \oint_{\delta\Omega^i} \mathbf{j}\cdot\mathbf{n}\, ds = \int_{\Omega^i} S\, d\Omega,$$

with $\mathbf{n}$ the outward unit normal vector at the boundary $\delta\Omega^i$ of Ω^i. Next the integrals are approximated by quadrature and the Scharfetter-Gummel scheme is used to discretize the fluxes. We distinguish two types of box schemes. In cell-centered finite volume discretizations we choose the nodes, where the dependent variables are approximated, at the centres of each cell. In vertex-centered schemes we first cover the domain by a grid, and then we construct the boxes around these grid points. The main difference between these two approaches is that nodes are located at the boundary of the domain in the vertex-centered schemes. For examples of the finite volume discretization see [3, 20, 32, 42].

As can be expected from the failure of the standard finite difference discretization, standard finite element discretizations for the continuity equations are also prone to numerical instabilities. Therefore finite element discretizations have been proposed that yield exponentially fitted schemes like the Scharfetter-Gummel discretization (cf. [24, 51, 55]). This is done e.g. by using appropriate quadrature rules in these finite element methods. For the finite difference method the domain is generally partitioned in rectangles, whereas in the finite volume method and the finite element method also triangular partitionings are used.

Brezzi et al. [6, 7] introduced a two-dimensional exponentially fitted method for the semiconductor equations using a (hybrid) mixed finite element method. In the mixed finite element method the semiconductor equations are not considered a system of three second order equations, but as a system of six first order equations that are discretized separately. The advantage of the mixed finite element method is that it offers a systematical way to extend the one dimensional Scharfetter-Gummel scheme to more dimensions, and that standard error estimates are available. Two types of mixed finite element discretizations are used for the semiconductor equations: the dual version (cf.

[6, 33, 35, 50]) and the primal version (cf. [11]). The main difference between these two discretization methods is the a priori assumption about the smoothness of the dependent variables. Opposite to the primal version of the mixed finite element method, it is assumed in the dual version that the fluxes $(\mathbf{j}_\psi, \mathbf{j}_n, \mathbf{j}_p)$ are much smoother functions than the potentials (ψ, Φ_n, Φ_p). In this thesis we will consider both the primal and the dual mixed finite element discretization for the semiconductor equations. The last Chapter is devoted to the comparison of multigrid solution methods for these two possible approaches.

1.4. Solution procedures

For comparison, here we discuss some of the methods -other than multigrid- that are used for the iterative solution of the system of nonlinear equations obtained by the discretization. Basically there are two approaches: either the equations are decoupled or the equations are solved simultaneously. The standard way of decoupling the semiconductor equations was proposed by Gummel [15], whereas usually Newton's method is used for the simultaneous solution of the equations. We first consider the classical Gummel method. For variants of this decoupling method see [13, 36].

Let the superscript i denote the iteration index in the Gummel iteration. Starting from functions $(\psi^{(i)}, \phi_n^{(i)}, \phi_p^{(i)})$, new functions $(\psi^{(i+1)}, \phi_n^{(i+1)}, \phi_p^{(i+1)})$ are obtained by successively solving the linear systems of equations that are obtained by linearizing the semiconductor equations with respect to the associated variable,

$$-\operatorname{div}(\mu_\psi \operatorname{grad}\psi^{(i+1)}) + \\ (1+\psi^{(i+1)}-\psi^{(i)})n^{(i)} - (1+\psi^{(i)}-\psi^{(i+1)})p^{(i)} = D, \tag{1.22a}$$

$$-\operatorname{div}\mu_n(\operatorname{grad} n^{(i+1)} - n^{(i+1)}\operatorname{grad}\psi^{(i+1)}) = -R\,, \tag{1.22b}$$

$$-\operatorname{div}\mu_p(\operatorname{grad} p^{(i+1)} + p^{(i+1)}\operatorname{grad}\psi^{(i+1)}) = -R\,. \tag{1.22c}$$

Gummel's iteration has the advantage that the discrete systems involved are linear and that the iteration is more robust than standard Newton methods, i.e. it often converges even if only a poor initial guess is available. In Gummel iteration the matrix of the Poisson equation (1.22a) is symmetric and positive definite, whereas the matrices of the continuity equations (1.22b-c) are in general asymmetric. The solution of these linear systems of equations is often approximated by iterative methods. For the Poisson equation the coefficient matrices are symmetric so the classical conjugate gradient (CG) method can be used, e.g. with an incomplete Cholesky factorization (cf. [32]) as the preconditioner. The coefficient matrices for the continuity equations are nonsymmetric, and different types of iterative methods are proposed, like the Conjugate Gradient Squared (CGS) [18, 32], the generalized minimum residual algorithm (GMRES) [18] and the bi-conjugate gradient squared (BiCGS) method [17]. Recently good results have been reported for a stabilized version of the last algorithm (BiCGSTAB) [17, 49]. Often incomplete LU decompositions (ILU) are used for preconditioning, but good results are also obtained by e.g. ILLU (cf. [54]). Linear multigrid algorithms for solving the discretized continuity equations in Gummel's iteration have been proposed e.g. by Fuhrmann [12] and Reusken [35].

However, when the equations are strongly coupled, Gummel's iteration converges slowly, so coupled approaches, like the classical Newton's method, are more attractive. It is well known that Newton's method may converge extremely slowly, or even diverge, if the initial iterate is not sufficiently close to the solution. Therefore Newton's method is modified in two ways. To avoid overshoot a global damping parameter can be introduced in Newton's method (cf. [2, 9]), and if the Jacobian is nearly singular a multiple of the unit matrix can be added to the Jacobian matrix (cf. [1]); in fact these modifications may be combined, but still there is no guarantee that the iteration will converge if the problem is very nonlinear, because the Jacobian may change too much during the iteration. As before, the solution of the linear equations in Newton's method can be approximated by preconditioned conjugated gradient methods (cf. [32]).

Another approach to the iterative solution of the coupled semiconductor equations is to apply the nonlinear multigrid method (cf. [5, 16]). Hemker [20] introduced a nonlinear multigrid method for a cell-centered finite volume discretization for the one-dimensional device equations. A special feature of this multigrid method is that it employs a current conserving prolongation. Constaple and Berger [8] elaborated on this approach and presented a nonlinear multigrid method for a vertex-centered discretization of the two-dimensional problem.

In these two multigrid methods the grids are constructed in different ways. For a cell-centered finite volume discretization it is natural to refine the cells, and a cell-centered multigrid algorithm is obtained. On the other hand, for the vertex-centered scheme it is more natural to refine the mesh by adding grid lines, and a vertex-centered multigrid algorithm is obtained. The important difference between these two multigrid approaches is that in vertex-centered multigrid the nodes of the coarse grid coincide with nodes on the fine grid, whereas this is not the case in cell-centered multigrid. For cell-centered multigrid the coarse and fine grid cells are nested. We will compare both these approaches in Section 7.

1.5. Outline of the thesis

This thesis is based on a number of reports and papers [21, 25-29] that appeared or that are forthcoming. An outline of the thesis is as follows. In Chapter 2 we consider a dual mixed finite element discretization of the semiconductor equations, based on lowest order Raviart-Thomas elements on rectangles (cf. [34]). The discretization is changed by applying a quadrature rule to the integrals that appear in the discretization. By this quadrature rule we are able to retain the Scharfetter-Gummel discretization of the fluxes; the resulting scheme is equivalent to a cell-centered finite volume discretization. We show that the use of the quadrature rule does not affect the accuracy of the original mixed finite element discretization.

For the efficiency of any multigrid method the choice of a proper relaxation procedure is of prime importance. In Chapter 3 we consider relaxation methods for the systems of equations obtained by the dual mixed finite element discretization. By showing the equivalence of the weak formulation of the mixed finite element discretization and two constrained optimization problems we are able to prove convergence for two relaxation methods: a Vanka-type relaxation (cf. [48]) and a superbox relaxation (cf. [41]). By local mode analysis we study the feasibility of these relaxation methods as smoothers in the multigrid algorithm. It turns out that the Vanka-type relaxation is the most efficient.

In Chapter 4 we carry out a two-grid analysis for the combination of Vanka-type relaxation and the canonical grid transfer operators, that are induced by the Raviart-Thomas elements. It appears that for the *one-dimensional* Poisson equation the canonical grid transfer operators can only be used in combination with Vanka-type relaxation if the grid points are ordered red-black. Convergence is not guaranteed if lexicographically ordered Vanka-type relaxation is used. Surprisingly, in the *two-dimensional* case the Vanka-type relaxation can be used with either of the two types of ordering.

In Chapter 5 we present a basic multigrid method for the dual mixed finite element discretization of the semiconductor equations as developed in Chapter 2. The multigrid method is based on the canonical grid transfer operators and a collective Vanka-type relaxation. As in the one-dimensional case (cf. [53]) it appears to be necessary to apply a local damping of the restricted residual in the coarse grid correction of this cell-centered multigrid method. Under these conditions we observe a fast and nearly grid independent convergence behavior of the nonlinear multigrid iteration for a simple diode model problem.

As the semiconductor equations are singularly perturbed (cf. [24]), we may expect that the dependent variables vary rapidly in small parts of the domain, so it is attractive and efficient to use adaptive local mesh refinements. In Chapter 6 we present a dual mixed finite element discretization on an adaptive grid. The discrete equations thus obtained are solved iteratively by means of the dual version of the full multigrid algorithm (cf. [19]). Consistent with this algorithm we use the relative truncation errors between the coarse and fine grids as the refinement criterion for the automatic, adaptive mesh refinement scheme. The effectiveness of the scheme is demonstrated for a realistic bipolar transistor problem.

In the final Chapter we also consider the primal mixed finite element discretization of the semiconductor equations. By this discretization we obtain a scheme that is equivalent to the vertex-centered finite volume discretization; this system of discretized equations is solved by a vertex-centered multigrid algorithm. Here it is shown that it is not necessary to apply the local damping of the restricted residual in vertex-centered multigrid, provided that injection is used for the restriction of the residual. It is well known that injection is usually too inaccurate a grid transfer operator for second order differential equations. However, by means of Fourier two-grid analysis we show that it is possible to construct a smoothing operator that indeed yields well-behaved two-grid algorithms. To compare the cell-centered and vertex-centered multigrid algorithms in practice we consider two test-problems: a MOS-transistor and an LDDMOS-transistor. In numerical experiments it appears that the vertex-centered multigrid approach is more efficient and robust than the cell-centered multigrid technique. It is shown that in typical situations the multigrid method is an efficient and robust solution method for the large system of nonlinear equations that arise from the discretization of the semiconductor equations. Moreover, it is shown that the computing time needed is indeed proportional to the number of discretization cells/points in the grid.

REFERENCES

1. R.E. BANK and D.J. ROSE (1980). Parameter selection for Newton-like methods applicable to nonlinear partial differential equations, *SIAM J. Num. Anal.*, 17, 806-822.
2. R.E. BANK and D.J. ROSE (1981). Global approximate Newton methods, *Numer. Math.*, 37, 279-295.
3. R.E. BANK, D.J. ROSE, and W. FICHTNER (1983). Semiconductor device simulation, *SIAM J.Sci.Stat.Comput.*, 4, 391-435.
4. K. BRAND, M. LEMKE, and J. LINDEN (1987). Multigrid Bibliography, in *Multigrid Methods, Frontiers in Applied Mathematics, Vol. 3*, 189-230, ed. S.F. MCCORMICK, SIAM, Philadelphia.
5. A. BRANDT (1982). Guide to multigrid development, in *Multigrid Methods*, 220-312, ed. W. HACKBUSCH AND U. TROTTENBERG, Springer-Verlag, Lecture Notes in Mathematics 960.
6. F. BREZZI, L.D. MARINI, and P. PIETRA (1989). Numerical simulation of semiconductor devices, *Comp. Meths. Appl. Mech. and Engr.*, 75, 493-513.
7. F. BREZZI, L.D. MARINI, and P. PIETRA (1989). Two-dimensional exponential fitting and applications to drift-diffusion models, *SIAM J.Num.Anal.*, 26, 1342-1355.
8. R. CONSTAPEL and M. BERGER (1989). A Multigrid Approach for Device Simulation Using Local Linearization, in *Proceedings NASECODE VI*, 355-359, ed. J.J.H. MILLER, Boole Press Ltd., Dublin.
9. P. DEUFLHARD (1974). A Modified Newton Method for the Solution of Ill-conditioned Systems of Nonlinear Equations with Application to Multiple Shooting, *Numer. Math.*, 22, 289-315.
10. A.F. FRANZ and G.A. FRANZ (1985). BAMBI - A Design Model for Power MOSFETs, *IEEE Trans. Computer Aided Design*, CAD-7, 177-189.
11. J. FUHRMANN (1990). An interpretation of the Scharfetter-Gummel scheme as a mixed finite element discretization, in *Fourth Multigrid Seminar*, 1-7, ed. G. TELSCHOW, Karl-Weierstrass-Institut fur Mathematik, Berlin.
12. J. FUHRMANN and K. GÄRTNER (1990). Incomplete factorization and linear multigrid algorithms for the semiconductor device equations, in *Proceedings IMACS conference*, Brusssels.
13. H. GAJEWSKI and K. GÄRTNER. On the iterative solution of van Roosbroeck's equations, *ZAMM*, To appear.

14. S.P. GAUR and A. BRANDT (1977). Numerical solution of semiconductor transport equations in two dimensions by multi-grid method, in *Advances in Computer Methods for Partial Differential Equations II*, 327-329, ed. R. VICHNEVETSKY.

15. H.K. GUMMEL (1964). A Self-Consistent Iterative Scheme for One-Dimensional Steady State Transistor Calculations, *IEEE Trans. Electron Devices*, ED-11, 455-465.

16. W. HACKBUSCH (1985). *Multigrid Methods and Applications*, Springer-Verlag, Berlin, Series in Computational Mathematics 4.

17. O. HEINREICHSBERGER, S. SELBERHERR, and M. STIFTINGER (1991). Massively Parallel Solution of the Three-Dimensional van Roosbroeck Equations, in *Proceedings of NUMSIN'91*, Berlin.

18. O. HEINREICHSBERGER, S. SELBERHERR, M. STIFTINGER, and K.P. TRAAR. Fast iterative solution of carrier continuity equations for 3D device simulation, *SIAM J.Sci.Stat.Comput.*, To appear.

19. P.W. HEMKER (1980). On the structure of an adaptive multi-level algorithm, *BIT*, 20, 289-301.

20. P.W. HEMKER (1988). A nonlinear multigrid method for one-dimensional semiconductor device simulation, in *BAIL V*, ed. GUO BEN YU, J.J.H. MILLER AND SHI ZHONG-CI, Boole Press, Dublin.

21. P.W. HEMKER and J. MOLENAAR (1991). An adaptive multigrid approach for the solution of the 2D semiconductor equations, in *International Series of Numerical Mathematics 98*, ed. W. HACKBUSCH AND U. TROTTENBERG, Birkhauser Verlag, Basel.

22. C. JACOBONI and P. LUGLI (1989). *The Monte Carlo method for semiconductor device simulation*, Springer-Verlag, New York.

23. C. LEPOETER (1987). *CURRY example set,* Technical Report No. 4322.271.6005, Philips, Corp. CAD Centre, Eindhoven.

24. P.A. MARKOWICH (1986). *The Stationary Semiconductor Device Equations*, Springer-Verlag, Wien, New York.

25. J. MOLENAAR. *Multigrid for semiconductor device simulation: cell-centered or vertex-centered multigrid ?,* Report NM-R9118, Centre for Mathematics and Computer Science, Amsterdam, To appear.

26. J. MOLENAAR (1990). Non-linear multigrid in 2-D semiconductor device simulation: the zero current case, in *Second GMD Seminar on Semiconductor problems*, ed. W. JOPPICH, Gesellschaft für Mathematik und Datenverarbeitung, Bonn.

27. J. MOLENAAR (1991). A two-grid analysis of the combination of mixed finite elements and Vanka-type relaxation, in *International Series of Numerical Mathematics 98*, ed. W. HACKBUSCH AND U. TROTTENBERG, Birkhauser Verlag, Basel.

28. J. MOLENAAR (1991). *Adaptive multigrid applied to a bipolar transistor problem,* Report NM-R9115, Centre for Mathematics and Computer Science, Amsterdam.

29. J. MOLENAAR and P.W. HEMKER (1990). A multigrid approach for the solution of the 2D semiconductor equations, *IMPACT*, 2, 219-243.

30. S. Odanaka, A. Hiroki, K. Ohe, H. Umimoto, and K. Moriyama (1989). SMART-II: A Three Dimensional CAD Model for Submicrometer MOSFETs, in *Proceedings NASECODE VI*, 303-310, ed. J.J.H. Miller, Boole Press Ltd., Dublin.
31. M.R. Pinto, C.S. Rafferty, and R.W. Dutton (1984). *PISCES-II: Poisson and Continuity Equation Solver,* Tech. Rep., Stanford Electronics Lab..
32. S.J. Polak, C. den Heijer, W.H.A. Schilders, and P. Markowich (1987). Semiconductor device modelling from the numerical point of view, *Int.J.Num.Meth.Engng.*, 24, 763-838.
33. S.J. Polak, W.H.A. Schilders, and H.D. Couperus (1988). A finite element method with current conservation, in *Proc. SISDEP-88*, 453-462, ed. G. Baccarani and M. Rudan, Bologna.
34. P.A. Raviart and J.M. Thomas (1977). A mixed finite element method for second order elliptic problems, in *Mathematical aspects of the finite element method*, Springer-Verlag, Lecture Notes in Mathematics 606.
35. A. Reusken (1991). Multigrid applied to two-dimensional exponential fitting for drift-diffusion models, in *Proceedings of the International Symposium on Iterative Methods in Linear Algebra*, Brusssels, To appear.
36. C. Ringhofer and C. Schmeiser (1989). An approximate Newton method for the solution of the basic semiconductor device equations, *SIAM J. Numer. Anal.*, 26, 507-516.
37. W.V. Van Roosbroeck (1950). Theory of Flow of Electrons and Holes in Germanium and Other Semiconductors., *Bell Syst. Techn. J.*, 19, 560-607.
38. S. Satoh, S. Ando, and N. Nakayama (1989). Study of DRAM cell structure using a three-dimensional device simulator, in *Proceedings NASECODE VI*, 311-316, ed. J.J.H. Miller, Boole Press Ltd., Dublin.
39. D.L. Scharfetter and H.K. Gummel (1969). Large-Signal Analysis of a Silicon Read Diode Oscillator, *IEEE Trans.E.D.*, ED-16, 64-77.
40. E. van Schie (1990). *TRENDY: an integrated program for IC process and device simulation,* Ph.D. Thesis, T.U. Twente, Enschede.
41. G.H. Schmidt and F.J. Jacobs (1988). Adaptive Local Grid Refinement and Multi-grid in Numerical Reservoir Simulation, *J.Comput.Phys.*, 77, 140-165.
42. S. Selberherr (1984). *Analysis and Simulation of Semiconductor Devices*, Springer-Verlag, Wien.
43. S. Selberherr, A. Schütz, and H.W. Pötzl (1980). MINIMOS - A Two-Dimensional MOST Transistor Analyzer, *IEEE Trans. Electron Devices*, ED-27, 1540-1550.
44. A.S.L. Shieh (1984). Solution of coupled systems of PDE by the transistorized multi-grid method, in *Procs of a Conference on Numerical Solution of VLSI devices*, Boston.
45. W. Shockley (1949). The theory of p-n Junctions in Semiconductors and p-n Junction transistors, *Bell Syst. Tech. J.*, 28, 435-489.
46. S. Slamet (1983). *Quasi-Newton and multigrid methods for semiconductor device simulation,* Report UIUCDCS-R83-1154, Urbana.
47. S.M. Sze (1981). *Physics of semiconductor devices*, J. Wiley, New York.

48. S.P. VANKA (1986). Block-Implicit Multigrid Solution of Navier-Stokes Equations in Primitive Variables, *J.Comput.Phys.*, 65, 138-158.

49. H.A. VAN DER VORST. Bi-CGSTAB: A Fast and Smoothly Converging Variant of BiCG for the Solution of Nonsymmetric Linear Systems, *SIAM J.Sci.Stat.Comput.*, To appear.

50. S. WANG and C. WU (1988). Mixed finite element approximation of the stationary semiconductor continuity equations, in *Proc. SISDEP-88*, 457-484, ed. G.BACCARANI AND M.RUDAN, Bologna.

51. J.S. VAN WELIJ (1986). Basis functions matching tangential components on element edges, in *Proc. Sec. Int. Conf. on Simul. of Semi. Dev. and Proc.*, Pineridge Press, Swansea.

52. P. WOLBERT (1991). *Modeling and Simulation of Semiconductor Devices in TRENDY,* Ph.D. Thesis, T.U. Twente, Enschede.

53. P.M. DE ZEEUW (1991). Nonlinear multigrid applied to a 1D stationary semiconductor model, *SIAM J.Sci.Stat.Comput.*, To appear.

54. P.M. DE ZEEUW (1991). Private communication

55. M. ZLAMAL (1984). *Finite Element Solution of the Fundamental Equations of Semiconductor Devices I,* Report, Technical University Brünn, CSSR.

Chapter 2

Dual mixed finite element discretization

2.1. Introduction

In this Chapter we consider the dual mixed finite element discretization in the form we use it for the semiconductor device equations. The primal mixed finite element discretization is presented in Chapter 7. The mixed finite element method offers a way to generalize the essentially one-dimensional Scharfetter-Gummel scheme to two dimensions, while a well developed convergence analysis is available. To derive the discretization of the semiconductor equations, we study the standard second order elliptic boundary value problem

$$\begin{aligned} \operatorname{div}(A \operatorname{grad} u) &= f, && \text{on } \Omega, \\ u &= g, && \text{on } \delta\Omega_D, \\ \mathbf{n} \cdot A \operatorname{grad} u &= 0, && \text{on } \delta\Omega_N, \\ A &> 0. \end{aligned} \tag{2.1}$$

The domain Ω is an open, bounded region in $\mathbb{R}^n$, $n=1,2$, with a piecewise smooth boundary $\delta\Omega$, $\delta\Omega_D$ and $\delta\Omega_N$ denote the parts of the boundary with Dirichlet or homogeneous Neumann boundary conditions, respectively, and $\mathbf{n}$ is the outward normal unit vector at the boundary. A is in principle an $n \times n$-matrix, but for our purposes it suffices to take A a scalar function. For the semiconductor equations written in this form (cf. (1.11)) we observe that A and f are nonlinear functions of the independent variables (ψ, Φ_n, Φ_p).

In Section 2.2 we introduce some notation and develop the weak formulation of problem (2.1). As an example of the dual mixed finite element method we discretize a one-dimensional linear source problem in Section 2.3. By means of a simple numerical example we show that the discretization is not stable, but by using a quadrature rule in the evaluation of the integrals appearing in the discretization we are able to obtain a stable scheme; this can also be considered as lumping (cf. [8]). In Section 2.4 we derive a dual mixed final element discretization of the model problem (2.1) in two dimensions that is based on the lowest order Raviart-Thomas elements on rectangles (cf. [9]). To obtain a stable scheme we again use a quadrature rule to lump the discrete system. In Section 2.5 we present a standard error estimate for the mixed finite element discretization and we study the influence of the aforementioned quadrature rule on the accuracy of the discrete approximations. In Section 2.6 we finally apply the the mixed finite element method (with lumping) to the semiconductor equations in two dimensions. We obtain a scheme that is equivalent to a cell-centered finite volume discretization, where the fluxes between the control volumes are approximated by the Scharfetter-Gummel scheme. The advantage of this way of deriving the discrete equations is that in the multigrid method the Raviart-Thomas elements give rise to a sequence of nested approximating subspaces, so natural prolongation and restriction operators are suggested by the discretization.

2.2. Preliminaries

We start by introducing some notation. As usual the Sobolev-spaces $W^{m,p}(\Omega)$ are defined by

$$W^{m,p}(\Omega) = \{ u \mid D^\alpha u \in L^p(\Omega),\ \ 0 \leqslant |\alpha| \leqslant m \}, \tag{2.2}$$

$$H^m(\Omega) = W^{m,2}(\Omega), \tag{2.3}$$

with seminorm

$$|u|_{m,p,\Omega} = (\sum_{|\alpha|=m} \int_\Omega |D^\alpha u|^p \, d\Omega)^{\frac{1}{p}} \tag{2.4}$$

and norm

$$\|u\|_{m,p,\Omega} = (\sum_{0\leqslant l \leqslant m} |u|^p_{l,p,\Omega})^{\frac{1}{p}}, \tag{2.5}$$

$$\|u\|_{m,\infty,\Omega} = \max_{0\leqslant|\alpha|\leqslant m} \sup_\Omega |D^\alpha u|, \tag{2.6}$$

$$\|u\|_m = \|u\|_{m,2,\Omega}, \tag{2.7}$$

where D^α denotes the distributional derivative of order α. $L^2(\Omega)$ is the Hilbert space of square integrable functions on Ω with inner product $(\cdot,\cdot)$ and $H^{BC}(\text{div},\Omega)$ is the Hilbert space of vector functions with Lebesgue integrable divergence,

$$H^{BC}(\text{div},\Omega) = \{ \boldsymbol{\sigma} \mid \boldsymbol{\sigma} \in (L^2(\Omega))^2,\ \text{div}\,\boldsymbol{\sigma} \in L^2(\Omega),\ \mathbf{n}\cdot\boldsymbol{\sigma} = 0 \text{ on } \delta\Omega_N \}, \tag{2.8}$$

with inner product

$$(\boldsymbol{\sigma},\boldsymbol{\tau})_H = (\boldsymbol{\sigma},\boldsymbol{\tau}) + (\text{div}\,\boldsymbol{\sigma}, \text{div}\,\boldsymbol{\tau}).$$

To discretize (2.1) we introduce a new independent variable σ and split the second order differential equation (2.1) in a system of two first order differential equations:

$$\boldsymbol{\sigma} - A\,\text{grad}\,u = 0, \qquad \text{on } \Omega, \tag{2.9a}$$

$$\text{div}\,\boldsymbol{\sigma} = f, \qquad \text{on } \Omega, \tag{2.9b}$$

$$u = g, \qquad \text{on } \delta\Omega_D, \tag{2.9c}$$

$$\mathbf{n}\cdot\boldsymbol{\sigma} = 0, \qquad \text{on } \delta\Omega_N. \tag{2.9d}$$

In the dual version of the mixed finite element method it is assumed that $\boldsymbol{\sigma}$ is a much smoother quantity than u because A may be a rapidly varying function, so one assumes that $\boldsymbol{\sigma} \in H^{BC}(\text{div}, \Omega)$ and $u \in L^2(\Omega)$. This is in contrast with the primal version of the mixed finite element method where we take $\boldsymbol{\sigma} \in (L^2(\Omega))^2$ and $u \in H^1(\Omega)$. In this Chapter we only consider the dual version of the mixed finite element method, the primal version is discussed in Chapter 7.

For ease of notation we write $V = H^{BC}(\text{div}, \Omega)$ and $W = L^2(\Omega)$, and we introduce the bilinear forms $a: V \times V \to \mathbb{R}$ and $b: V \times W \to \mathbb{R}$ that are defined by

$$a(\boldsymbol{\sigma}, \boldsymbol{\tau}) = \int_\Omega A^{-1} \boldsymbol{\sigma} \cdot \boldsymbol{\tau} \, d\Omega, \tag{2.10}$$

$$b(\boldsymbol{\sigma}, t) = \int_\Omega t \, \text{div}\, \boldsymbol{\sigma} \, d\Omega. \tag{2.11}$$

Using this notation we write (2.9) in its weak form: find $(\boldsymbol{\sigma}, u) \in V \times W$ such that

$$a(\boldsymbol{\sigma}, \boldsymbol{\tau}) + b(\boldsymbol{\tau}, u) = \langle g, \boldsymbol{\tau} \rangle_{\delta\Omega_D}, \quad \forall \boldsymbol{\tau} \in V, \tag{2.12a}$$

$$b(\boldsymbol{\sigma}, t) \qquad\qquad = (f, t), \qquad \forall t \in W, \tag{2.12b}$$

with

$$\langle g, \boldsymbol{\tau} \rangle_{\delta\Omega_D} = \int_{\delta\Omega_D} g \boldsymbol{\tau} \cdot \mathbf{n} \, ds, \tag{2.13}$$

$$(f, t) = \int_\Omega f t \, d\Omega. \tag{2.14}$$

The boundary term $\langle g, \cdot \rangle_{\delta\Omega_D}$ stems from the application of Green's theorem.

2.3. Discretization of the 1D model problem

As an example of the dual mixed finite element method we consider the linear source problem on the unit interval, $\Omega = [0, 1] \subset \mathbb{R}^1$, with homogeneous boundary conditions,

$$\boldsymbol{\sigma} - \text{grad}\, u = 0, \tag{2.15a}$$

$$\text{div}\, \boldsymbol{\sigma} - cu = F(x), \tag{2.15b}$$

$$u(0) = u(1) = 0, \tag{2.15c}$$

c is a nonnegative constant. This is a special case of the standard problem (2.9) with $A = 1$, $g = 0$ and $f(u, x) = cu + F(x)$.

We decompose the domain Ω into a set of N uniform cells Ω_h^i,

$$\Omega_h^i = \left[\frac{(i-1)}{N}, \frac{i}{N} \right], \qquad i = 1, \cdots, N, \tag{2.16}$$

of size $h = \dfrac{1}{N}$. On this mesh we introduce the lowest order Raviart-Thomas elements. On each cell Ω_h^i the characteristic function $e_h^i \in L^2(\Omega)$ is

$$e_h^i(x) = \begin{cases} 1, & x \in \Omega_h^i, \\ 0, & x \notin \Omega_h^i. \end{cases} \tag{2.17}$$

For every edge E_h^j at $x = jh$ of a cell Ω_h^i we introduce the piecewise linear function $\boldsymbol{\epsilon}_h^j \in H^{BC}(\text{div}, \Omega)$,

$$\boldsymbol{\epsilon}_h^j(E_h^k) = \delta_{jk}, \quad k = 0, \cdots, N, \tag{2.18}$$

where δ_{jk} denotes the Kronecker delta. Our discrete approximation spaces are defined by

$$\begin{aligned} V_h &= \text{span}\,(\boldsymbol{\epsilon}_h^j) \subset V, \\ W_h &= \text{span}\,(e_h^i) \subset W, \end{aligned} \tag{2.19}$$

and the discrete approximation $(\boldsymbol{\sigma}_h, u_h)$ of solution $(\boldsymbol{\sigma}, u)$ is

$$\begin{aligned} \boldsymbol{\sigma}_h &= \sum_{j=0,N} \sigma_h^j \boldsymbol{\epsilon}_h^j, \\ u_h &= \sum_{i=1,N} u_h^i e_h^i. \end{aligned} \tag{2.20}$$

To discretize (2.12) we replace V and W in (2.12) by V_h and W_h, respectively, and use $\boldsymbol{\epsilon}_h^j$ and e_h^i as the test functions. After division by h for proper scaling, the resulting algebraic system for $(\underline{\sigma}_h, \underline{u}_h)^{\mathrm{T}}$, i.e. the column vector of the coefficients $\{\sigma_h^j, u_h^i\}$, is

$$\begin{bmatrix} \mathbf{a}_h & \mathbf{b}_h \\ \mathbf{b}_h^{\mathrm{T}} & \mathbf{c}_h \end{bmatrix} \begin{bmatrix} \underline{\sigma}_h \\ \underline{u}_h \end{bmatrix} = \begin{bmatrix} 0 \\ \underline{F}_h \end{bmatrix}. \tag{2.21}$$

The matrix elements in this system are

$$(\mathbf{a}_h)_{jk} = \frac{1}{h} \int_\Omega \boldsymbol{\epsilon}_h^j \boldsymbol{\epsilon}_h^k \, d\Omega = \begin{cases} \frac{1}{6}, & |j-k| = 1, \\ \frac{2}{3}, & j = k, \quad j = 1, \cdots, N-1, \\ \frac{1}{3}, & j = k, \quad j \in \{0, N\}, \\ 0, & \text{otherwise}, \end{cases} \tag{2.22a}$$

$$(\mathbf{b}_h)_{ji} = \frac{1}{h} \int_\Omega e_h^i \,\text{div}\, \boldsymbol{\epsilon}_h^j \, d\Omega = \begin{cases} -\frac{1}{h}, & j = i-1, \\ +\frac{1}{h}, & j = i, \\ 0, & \text{otherwise}, \end{cases} \tag{2.22b}$$

$$(\mathbf{c}_h)_{il} = -\frac{1}{h} \int_\Omega e_h^i e_h^l \, d\Omega = -\frac{c}{h^2} \delta_{il} \tag{2.22c}$$

and

$$(F_h)_i = -\frac{1}{h} \int_\Omega e_h^i F \, d\Omega. \tag{2.22d}$$

This discretization is not stable in the sense that the matrix $\mathbf{b}_h^{\mathrm{T}}\mathbf{a}_h^{-1}\mathbf{b}_h - \mathbf{c}_h$, obtained after elimination of σ_h, is not always an M-matrix. We show this by means of a simple example (cf. [8]), but first we recall the definition of an M-matrix.

Definition 2.1. A matrix a is an M-matrix if its off-diagonal elements are non-positive, $a_{i,j} \leqslant 0$ for $i \neq j$, and if it has a positive inverse.

Example 2.1. For a mesh that consists of three cells, i.e. $N=3$, we have

$$\mathbf{b}_h^{\mathrm{T}}\mathbf{a}_h^{-1}\mathbf{b}_h - \mathbf{c}_h = \frac{54}{5}\begin{bmatrix} 6 & -3 & 1 \\ -3 & 4 & -3 \\ 1 & -3 & 6 \end{bmatrix} + 9c\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

so $\mathbf{b}_h^{\mathrm{T}}\mathbf{a}_h^{-1}\mathbf{b}_h - \mathbf{c}_h$ is not an M-matrix. For Poisson's equation ($c=0$) we find that $\mathbf{b}_h^{\mathrm{T}}\mathbf{a}_h^{-1}\mathbf{b}_h$ has a positive inverse, but in the more general case $c>0$ we have

$$(\mathbf{b}_h^{\mathrm{T}}\mathbf{a}_h^{-1}\mathbf{b}_h - \mathbf{c}_h)_{1,3}^{-1} = \frac{5}{54}\frac{5-\tilde{c}}{\tilde{c}^3 + 16\tilde{c}^2 + 65\tilde{c} + 50},$$

with

$$\tilde{c} = \frac{5}{6}c.$$

For $c>6$ we observe that the inverse of $\mathbf{b}_h^{\mathrm{T}}\mathbf{a}_h^{-1}\mathbf{b}_h - \mathbf{c}_h$ is no longer a positive matrix, and the discretization is unstable in the sense that the matrix obtained after elimination of σ_h is not an M-matrix.

In literature two possible remedies to this stability problem can be found. For Poisson's equation ($c=0$) we can use use the hybrid mixed finite element method (cf. [11]): the flux $\boldsymbol{\sigma}_h$ is assumed to be piecewise linear, and the continuity of $\boldsymbol{\sigma}_h$ at the edges is enforced by Lagrange multipliers. The other possibility was proposed by Polak c.s. [8], viz. to lump the matrix $\mathbf{a}_h$. We proceed by working out these two approaches for problem (2.15).

To define the Lagrange multipliers we introduce on each internal edge E_h^j, $j \notin \{0,N\}$, the function $\bar{e}_h^j$,

$$\bar{e}_h^j(x) = \begin{cases} 1, & x = jh, \\ 0, & x \neq jh, \end{cases} \tag{2.23}$$

and the 'half tent' function $\boldsymbol{\epsilon}_h^{i,j} = e_h^i\boldsymbol{\epsilon}_h^j \in (L^2(\Omega))$ for $E_h^j \in \bar{\Omega}_h^j$. In addition to the approximating subspaces V_h and W_h in (2.19) we define

$$\begin{aligned} M_h &= \operatorname{span}(\bar{e}_h^j), \\ H_h &= \operatorname{span}(\boldsymbol{\epsilon}_h^{i,j}). \end{aligned} \tag{2.24}$$

The functions $\boldsymbol{\sigma}_h^* \in H_h$ are not necessarily continuous at the edges E_h^j; we denote the left and right limit values of $\boldsymbol{\sigma}_h^*$ at E_h^j by $\boldsymbol{\sigma}_h^*|_j^-$ and $\boldsymbol{\sigma}_h^*|_j^+$, respectively. The hybrid mixed finite element discretization of (2.15) reads: find $(\boldsymbol{\sigma}_h^*, u_h^*, \lambda_h) \in H_h \times W_h \times M_h$ such that

$$\begin{gathered}
\int_\Omega \boldsymbol{\sigma}_h^* \cdot \boldsymbol{\tau}_h \, d\Omega + \sum_i \int_{\Omega_h^i} u_h^* \operatorname{div} \boldsymbol{\tau}_h \, d\Omega = \sum_j \lambda_h(jh)(\boldsymbol{\tau}_h |_j^+ - \boldsymbol{\tau}_h |_j^-), \quad \forall \boldsymbol{\tau}_h \in H_h, \\
\sum_i \int_{\Omega_h^i} t_h \operatorname{div} \boldsymbol{\sigma}_h^* \, d\Omega = 0, \quad \forall t_h \in W_h, \qquad (2.25) \\
\sum_{j=1,N-1} \mu_h(jh)(\boldsymbol{\sigma}_h^* |_j^+ - \boldsymbol{\sigma}_h^* |_j^-) = 0, \quad \forall \mu_h \in M_h.
\end{gathered}$$

The third equation guarantees that $\boldsymbol{\sigma}_h^* \in H^{BC}(\operatorname{div}, \Omega)$. The algebraic system, that results from (2.25), is

$$\begin{pmatrix} \mathbf{a}_h & \mathbf{b}_h & \mathbf{l}_h \\ \mathbf{b}_h^{\mathrm{T}} & \mathbf{c}_h & 0 \\ \mathbf{l}_h^{\mathrm{T}} & 0 & 0 \end{pmatrix} \begin{pmatrix} \underline{\sigma_h} \\ \underline{u_h} \\ \underline{\lambda_h} \end{pmatrix} = \begin{pmatrix} \underline{0} \\ \underline{F_h} \\ \underline{0} \end{pmatrix}. \qquad (2.26a)$$

The matrix elements in this system are

$$(\mathbf{a}_h)_{jk} = \begin{cases} \frac{1}{3}, & j = k, \quad \text{for } j \in 1 \cdots 2N, \\ \frac{1}{6}, & j = k + 1 = 2i, \quad \text{for } i \in 1 \cdots N, \\ \frac{1}{6}, & j = k - 1 = 2i - 1, \quad \text{for } i \in 1 \cdots N, \\ 0, & \text{otherwise}, \end{cases} \qquad (2.26b)$$

$$(\mathbf{b}_h)_{ji} = \begin{cases} +\frac{1}{h}, & j = 2i, \quad \text{for } i \in 1 \cdots N, \\ -\frac{1}{h}, & j = 2i - 1, \quad \text{for } i \in 1 \cdots N, \\ 0, & \text{otherwise}, \end{cases} \qquad (2.26c)$$

$$(\mathbf{c}_h)_{il} = -\frac{c}{h^2} \delta_{il}, \quad \text{for } i \in 1 \cdots N, \qquad (2.26d)$$

and

$$(\mathbf{l}_h)_{jl} = \begin{cases} +\frac{1}{h}, & j = 2l, \quad \text{for } l \in 1 \cdots N - 1, \\ -\frac{1}{h}, & j = 2l + 1, \quad \text{for } l \in 1 \cdots N - 1, \\ 0, & \text{otherwise}. \end{cases} \qquad (2.26e)$$

If $\underline{\sigma_h}$ and $\underline{u_h}$ are eliminated from (2.26) we get a system of equations for the Lagrange multipliers only,

$$\mathbf{g}_h \, \underline{\lambda_h} = -\mathbf{l}_h^{\mathrm{T}} \mathbf{a}_h^{-1} \mathbf{b}_h \, (\mathbf{b}_h^{\mathrm{T}} \mathbf{a}_h^{-1} \mathbf{b}_h - \mathbf{c}_h)^{-1} \, \underline{F_h}, \qquad (2.27)$$

with

$$\mathbf{g}_h = \mathbf{l}_h^{\mathrm{T}} \left[\mathbf{a}_h^{-1} \mathbf{b}_h (\mathbf{b}_h^{\mathrm{T}} \mathbf{a}_h^{-1} \mathbf{b}_h - \mathbf{c}_h)^{-1} \mathbf{b}_h^{\mathrm{T}} \mathbf{a}_h^{-1} - \mathbf{a}_h^{-1} \right] \mathbf{l}_h. \qquad (2.28)$$

For Poisson's equation ($c=0$) the matrix $\mathbf{g}_h$ is always an M-matrix (cf. [2]), but for $c>0$ this is not always the case. We show this by means of an example.

EXAMPLE 2.2. For $N=3$ we have

$$\mathbf{g}_h = \frac{18}{c+12}\begin{bmatrix} 12+4c & c-6 \\ c-6 & 12+4c \end{bmatrix}$$

with inverse

$$\mathbf{g}_h^{-1} = \frac{c+12}{54(5c+6)(c+6)}\begin{bmatrix} 12+4c & 6-c \\ 6-c & 12+4c \end{bmatrix},$$

so for $c>6$ we find that $\mathbf{g}_h$ is no longer an M-matrix.

We conclude that in the general case $c>0$ the introduction of Lagrange multipliers does not solve the stability problem. Therefore we consider the other possibility: lumping of the matrix $\mathbf{a}_h$.

We change the discretization a little by approximating the integral in (2.22a) by a repeated trapezoidal rule,

$$(\mathbf{a}_h)_{j,k} \approx (\tilde{\mathbf{a}}_h)_{j,k} = \frac{1}{2}\sum_{i=1,N} (\boldsymbol{\epsilon}_h^j(ih)\boldsymbol{\epsilon}_h^k(ih) + \boldsymbol{\epsilon}_h^j((i-1)h)\boldsymbol{\epsilon}_h^k((i-1)h)). \tag{2.29}$$

By this quadrature $\mathbf{a}_h$ is approximated by a diagonal matrix; effectively the matrix $\tilde{\mathbf{a}}_h$ is lumped:

$$(\tilde{\mathbf{a}}_h)_{jk} = \begin{cases} 1, & j=k, \quad j \notin \{0,N\}, \\ \frac{1}{2}, & j=k, \quad j \in \{0,N\}, \\ 0, & \text{otherwise.} \end{cases} \tag{2.30}$$

When $\underline{\sigma}_h$ is eliminated from the lumped system, we obtain

$$(\mathbf{b}_h^{\mathrm{T}}\tilde{\mathbf{a}}_h^{-1}\mathbf{b}_h - \mathbf{c}_h) = \frac{1}{h^2}\begin{bmatrix} 3+ch^2 & -1 & & & \\ -1 & 2+ch^2 & -1 & & \\ & & \cdots & & \\ & & -1 & 2+ch^2 & -1 \\ & & & -1 & 3+ch^2 \end{bmatrix}, \tag{2.31}$$

which is equivalent to the system one obtains by a finite volume discretization. The matrix $\mathbf{b}_h^{\mathrm{T}}\tilde{\mathbf{a}}_h^{-1}\mathbf{b}_h - \mathbf{c}_h$ in (2.31) is an M-matrix for all values $c \geqslant 0$, so we conclude that by using the quadrature rule (2.29) we have obtained a stable discretization of the linear source problem (2.15). Moreover in Section 2.5 we show that the order of convergence of the approximations is not influenced by the use of the quadrature rule.

2.4. Discretization of the 2D model problem

For the discretization of the model problem (2.9) in two dimensions we assume that Ω can be divided by a regular partitioning in open disjoint, rectangular cells Ω^i, $\overline{\Omega} = \cup \overline{\Omega}^i$. (When no confusion arises we drop the subscript h.) On this rectangular grid we define lowest order Raviart-Thomas elements (cf. [9]). On each cell Ω^i we have the characteristic function e^i and for each edge E^j, $E^j \not\subset \delta\Omega_N$, of a cell Ω^i we define the 'tent function' $\boldsymbol{\epsilon}^j$, i.e. the vector function of which each component is piecewise linear on each Ω^i and which satisfies $\boldsymbol{\epsilon}^j \cdot \mathbf{n}^k = \delta_{jk}$, where $\mathbf{n}^k$ denotes the unit vector normal on the edge E^k in the positive coordinate direction; δ_{jk} is the Kronecker delta. Furthermore we introduce the function $\overline{e}^j \in L^2(\Omega)$ for the cell edges E^j, $E^j \not\subset \delta\Omega_D$,

$$\overline{e}^j(\mathbf{x}) = \begin{cases} 1, & \mathbf{x} \in E^j, \\ 0, & \mathbf{x} \notin E^j \end{cases}$$

and the 'half tent' function $\boldsymbol{\epsilon}^{i,j} = e^i \boldsymbol{\epsilon}^j \in (L^2(\Omega))^2$ for $E^j \subset \overline{\Omega}^i$ $(E^j \not\subset \delta\Omega_N)$. As in the one-dimensional case our discrete approximation spaces are defined by

$$\begin{aligned} V_h &= \text{span } (\boldsymbol{\epsilon}^j) \subset V, \qquad (2.32) \\ W_h &= \text{span } (e^i) \subset W, \\ M_h &= \text{span } (\overline{e}^j), \\ H_h &= \text{span } (\boldsymbol{\epsilon}^{i,j}). \end{aligned}$$

The spaces M_h and H_h are needed in the hybrid mixed finite element discretization of (2.9). The mixed finite element discretization of (2.9) formulates as follows: find $(\boldsymbol{\sigma}_h, u_h) \in V_h \times W_h$ such that

$$a(\boldsymbol{\sigma}_h, \boldsymbol{\tau}_h) + b(\boldsymbol{\tau}_h, u_h) = \langle g, \boldsymbol{\tau}_h \rangle_{\delta\Omega_D}, \quad \forall \boldsymbol{\tau}_h \in V_h, \qquad (2.33a)$$

$$b(\boldsymbol{\sigma}_h, t_h) \qquad = (f, t_h), \qquad \forall t_h \in W_h. \qquad (2.33b)$$

The integrals $\langle g, \boldsymbol{\tau}_h \rangle$ and $b(\boldsymbol{\tau}_h, t_h)$ are easily evaluated. As was shown in Section 2.3 , it is advantageous to use a quadrature rule for the evaluation of the integrals $a(\boldsymbol{\epsilon}^j, \boldsymbol{\epsilon}^k)$ that lumps the matrix $\mathbf{a}_h$. Moreover, we also need a quadrature rule for the evaluation of the integrals (f, t_h). We approximate these integrals by a repeated, weighted trapezoidal rule for rectangles,

$$\begin{aligned} \int_\Omega w(\mathbf{x}) z(\mathbf{x})\, d\Omega &= \sum_i \int_{\Omega^i} w(\mathbf{x}) z(\mathbf{x})\, d\Omega \\ &= \sum_i \sum_{\nu=1,2,3,4} z(\mathbf{x}^{i,\nu}) \int_{\Omega^{i,\nu}} w(\mathbf{x})\, d\Omega + \mathfrak{R}(w, z), \qquad (2.34) \end{aligned}$$

where $\mathbf{x}^{i,\nu}$ are the four vertices of Ω^i, and $\Omega^{i,\nu}$ the four quarter rectangles, parts of Ω^i, associated with these vertices, respectively (cf. Figure 2.1). In the next Section we give an estimation of the error term $\mathfrak{R}(w, z)$.

For the approximation of $a(\boldsymbol{\epsilon}^j, \boldsymbol{\epsilon}^k)$ we use (2.34) with $w = A^{-1}$ and $z = \boldsymbol{\epsilon}^j \cdot \boldsymbol{\epsilon}^k$. As in the one-dimensional case the bilinear form $a(\cdot, \cdot)$ is approximated by a diagonal bilinear form $\tilde{a}(\cdot, \cdot)$:

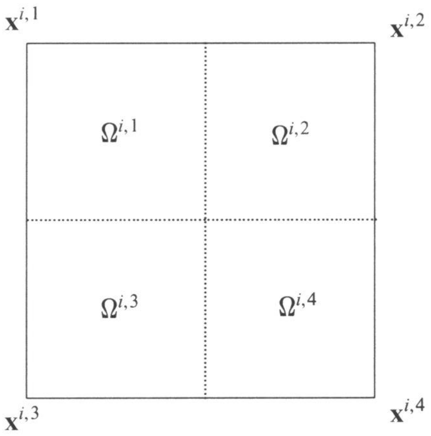


FIGURE 2.1. Subdivision of cell Ω^i for quadrature rule.

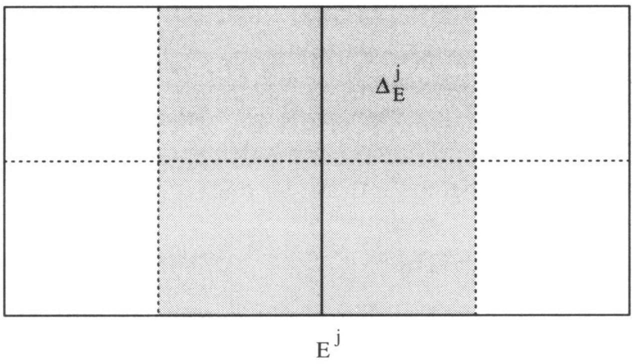


FIGURE 2.2. Dual cell Δ_E^j related to edge E^j.

$$\tilde{a}(\boldsymbol{\epsilon}^j, \boldsymbol{\epsilon}^k) = \delta_{jk} \int_{\Delta_E^k} A^{-1}\, d\Omega. \tag{2.35}$$

Here $\Delta_E^k = \cup\{\Omega^{i,\nu} \mid \overline{\Omega}^{i,\nu} \cap E^k \neq \varnothing\}$, i.e. Δ_E^k is the dual box related with the edge E^k (cf. Figure 2.2).

If (f, t_h) is also approximated by (2.34), now with $w = t_h$ and $z = f$, we obtain from (2.33b)

$$\forall\Omega^i\colon \sum_j h^j d^{i,j} \sigma^j = \text{area}\,(\Omega^i)\, \frac{1}{4} \sum_{\nu=1,2,3,4} f(\mathbf{x}^{i,\nu}), \tag{2.36a}$$

with

$$d^{i,j} = \begin{cases} +1, & \text{if } E^j \text{ is a N- or E- edge of } \Omega^i, \\ -1, & \text{if } E^j \text{ is a S- or W- edge of } \Omega^i, \\ \;\;0, & \text{otherwise,} \end{cases} \tag{2.36b}$$

and h^j the length of edge E^j. We notice that (2.36) implies discrete current conservation.

In our calculations on adaptive grids (Chapter 6) we will need approximations for the potentials u at edges of cells. It is known that the Lagrange multipliers that appear in the hybrid mixed finite element discretization are a good approximation for these values (cf. [1]). As in Section 2.3 we use the discontinuous tent functions $\boldsymbol{\epsilon}^{i,j}$ as test and trial functions in equation (2.33), and introduce Lagrange multipliers λ_h to enforce sufficient continuity of $\boldsymbol{\sigma}_h$.

The augmented variational equations read: find $(\boldsymbol{\sigma}_h^*, u_h^*, \lambda_h) \in H_h \times W_h \times M_h$ such that

$$\begin{aligned} \sum_i \int_{\Omega^i} A^{-1} \boldsymbol{\sigma}_h^* \cdot \boldsymbol{\tau}_h \, d\Omega + \sum_i \int_{\Omega^i} u_h^* \operatorname{div} \boldsymbol{\tau}_h \, d\Omega &= \sum_i \oint_{\delta\Omega^i} \lambda_h \boldsymbol{\tau}_h \cdot \mathbf{n}\, ds, && \forall \boldsymbol{\tau}_h \in H_h, \\ \sum_i \int_{\Omega^i} t_h \operatorname{div} \boldsymbol{\sigma}_h^* \, d\Omega &= \sum_i \int_{\Omega^i} f t_h \, d\Omega, && \forall t_h \in W_h, \\ \sum_j \int_{E^j} \mu_h \boldsymbol{\sigma}_h^* \cdot \mathbf{n}^j \, ds &= 0, && \forall \mu_h \in M_h. \end{aligned} \tag{2.37}$$

The third equation guarantees that $\boldsymbol{\sigma}_h^* \in H^{BC}(\text{div}, \Omega)$, hence in the interior of the domain the solution of (2.37) coincides with the solution of (2.33). The approximations of u at the edges E^j are the coefficients λ^j in $\lambda_h = \Sigma_j \lambda^j \overline{e}^j \in M_h$. If lumping is used, the Lagrange multipliers λ^j on the interior edges E^j can be expressed in the values of u^i in the adjacent cells Ω^i, $i = R, L$. Using the half tent functions $\boldsymbol{\epsilon}^{L,j}$ and $\boldsymbol{\epsilon}^{R,j}$ as test functions in (2.37) we obtain

$$\begin{aligned} \sigma^j \int_{\Omega^L \cap \Delta_E^j} A^{-1}\, d\Omega + u^L &= +\lambda^j, \\ \sigma^j \int_{\Omega^R \cap \Delta_E^j} A^{-1}\, d\Omega - u^R &= -\lambda^j, \end{aligned}$$

so

$$\lambda^j = u^L \frac{\int\limits_{\Omega^R \cap \Delta_E^j} A^{-1}\, d\Omega}{\int\limits_{\Delta_E^j} A^{-1}\, d\Omega} + u^R \frac{\int\limits_{\Omega^L \cap \Delta_E^j} A^{-1}\, d\Omega}{\int\limits_{\Delta_E^j} A^{-1}\, d\Omega}. \tag{2.38}$$

For A constant this comes down to linear interpolation.

2.5. Error estimates

In this Section we give a standard error estimate (cf. [4]) for the mixed finite element discretization (2.33) and we study the influence of the quadrature rule (2.34) on the accuracy of the discrete solution. We assume that A and f are given functions and for simplicity we assume homogeneous boundary conditions ($g = 0$).

On the partitioning of the domain Ω we introduce a norm

$$\|u\|_{m,p,\Delta} = \Big(\sum_i \|u\|^p_{m,p,\Omega^i}\Big)^{\frac{1}{p}}, \tag{2.39}$$

$$\|u\|_{m,\Delta} = \|u\|_{m,2,\Delta}. \tag{2.40}$$

The projection operators $\Pi_h^\sigma : V \rightarrow V_h$ and $\Pi_h^u : W \rightarrow W_h$ are defined by (cf. [9])

$$\langle \overline{e}^j, \boldsymbol{\sigma} \rangle_{E^j} = \langle \overline{e}^j, \Pi_h^\sigma \boldsymbol{\sigma} \rangle_{E^j}, \quad \forall E^j, \tag{2.41}$$

$$(e^i, u) = (e^i, \Pi_h^u u), \qquad \forall \Omega^i, \tag{2.42}$$

with

$$\langle \overline{e}^j, \boldsymbol{\sigma} \rangle_{E^j} = \int\limits_{E^j} \overline{e}^j \boldsymbol{\sigma} \cdot \mathbf{n}^j \, ds. \tag{2.43}$$

For these projections on the lowest order Raviart-Thomas elements the divergence operator and the projection operator commute (cf. [3]):

$$\operatorname{div} \Pi_h^\sigma = \Pi_h^u \operatorname{div}. \tag{2.44}$$

Furthermore, we have (cf. [5])

$$\|\Pi_h^\sigma \boldsymbol{\sigma}\|_V \leqslant C \|\boldsymbol{\sigma}\|_V, \quad \forall \boldsymbol{\sigma} \in V. \tag{2.45}$$

For the dual mixed finite element discretization of (2.33) the following result is known ([3]).

Theorem 2.1. *If the domain Ω is such that the problem* $\operatorname{div}(\operatorname{grad} u) = w$ *with homogeneous Dirichlet boundary conditions is regular, i.e. for each $w \in L^2(\Omega)$ there exists a unique solution $u \in H^2(\Omega)$ with $\|u\|_2 \leqslant C \|w\|_0$, and if $a(\cdot,\cdot)$ is L^2-coercive, i.e.*

$$\exists\, C > 0 : a(\boldsymbol{\sigma},\boldsymbol{\tau}) \geqslant C \|\boldsymbol{\sigma}\|_0 \, \|\boldsymbol{\tau}\|_0, \quad \forall \boldsymbol{\sigma},\boldsymbol{\tau} \in V, \tag{2.46}$$

then problem (2.12) has a unique solution $(\boldsymbol{\sigma}, u)$ and problem (2.33) has a unique *solution* $(\boldsymbol{\sigma}_h, u_h)$. *Moreover there exists a* $C > 0$, *independent if* h, *such that*

$$\begin{aligned} \|\boldsymbol{\sigma} - \boldsymbol{\sigma}_h\|_0 &\leqslant Ch\|u\|_2, \\ \|\operatorname{div}(\boldsymbol{\sigma} - \boldsymbol{\sigma}_h)\|_0 &\leqslant C\|u\|_2, \\ \|u - u_h\|_0 &\leqslant Ch\|u\|_2. \end{aligned}$$

PROOF. For a proof see [3]. □

The accuracy of the quadrature rule (2.34) is stated in the following lemma.

LEMMA 2.1. *For functions* w, $z \in C^2(\overline{\Omega}^i)$ *we have*

$$\Big| \int_{\Omega} wz\, d\Omega - \sum_i \sum_\nu z(\mathbf{x}^\nu) \int_{\Omega^{i,\nu}} w\, d\Omega \Big| \leqslant Ch^2 \|wz\|_{2,1,\Delta}. \tag{2.47}$$

PROOF. The quadrature is exact on Ω^i for all constant functions z, and also for constant functions w and linear functions z, so by using Taylor expansions around the centre $\mathbf{x}^i$ of Ω^i we obtain

$$\begin{aligned} &\int_{\Omega^i} wz\, d\mathbf{x} - \sum_\nu z(\mathbf{x}^\nu) \int_{\Omega^{i,\nu}} w(\mathbf{x})\, d\mathbf{x} \,| \\ &\qquad \leqslant \Big| \int_{\Omega^i} [z(\mathbf{x}^i) + (\mathbf{x} - \mathbf{x}^i) \cdot \operatorname{grad} z(\mathbf{x}^i)] w(\mathbf{x})\, d\mathbf{x} \,- \\ &\qquad\quad \sum_\nu [z(\mathbf{x}^i) + (\mathbf{x}^\nu - \mathbf{x}^i) \cdot \operatorname{grad} z(\mathbf{x}^i)] \int_{\Omega^{i,\nu}} w(\mathbf{x})\, d\mathbf{x} \Big| + Ch^2 \|wz\|_{2,1,\Omega^i} \\ &\qquad = \Big| \int_{\Omega^i} [(\mathbf{x} - \mathbf{x}^i) \cdot \operatorname{grad} z(\mathbf{x}^i)] w(\mathbf{x})\, d\mathbf{x} \,- \\ &\qquad\quad \sum_\nu [(\mathbf{x}^\nu - \mathbf{x}^i) \cdot \operatorname{grad} z(\mathbf{x}^i)] \int_{\Omega^{i,\nu}} w(\mathbf{x})\, d\mathbf{x} \Big| + Ch^2 \|wz\|_{2,1,\Omega^i} \\ &\qquad \leqslant \Big| \int_{\Omega^i} [(\mathbf{x} - \mathbf{x}^i) \cdot \operatorname{grad} z(\mathbf{x}^i)][w(\mathbf{x}^i) + (\mathbf{x} - \mathbf{x}^i) \cdot \operatorname{grad} w(\mathbf{x}^i)]\, d\mathbf{x} \,- \\ &\qquad\quad \sum_\nu [(\mathbf{x}^\nu - \mathbf{x}^i) \cdot \operatorname{grad} z(\mathbf{x}^i)] \int_{\Omega^{i,\nu}} [w(\mathbf{x}^i) + (\mathbf{x} - \mathbf{x}^i) \cdot \operatorname{grad} w(\mathbf{x}^i)]\, d\mathbf{x} \Big| \\ &\qquad\quad + Ch^2 \|wz\|_{2,1,\Omega^i} \\ &\qquad = \Big| \int_{\Omega^i} [(\mathbf{x} - \mathbf{x}^i) \cdot \operatorname{grad} z(\mathbf{x}^i)][(\mathbf{x} - \mathbf{x}^i) \cdot \operatorname{grad} w(\mathbf{x}^i)]\, d\mathbf{x} \,- \\ &\qquad\quad \sum_\nu [(\mathbf{x}^\nu - \mathbf{x}^i) \cdot \operatorname{grad} z(\mathbf{x}^i)] \int_{\Omega^{i,\nu}} [(\mathbf{x} - \mathbf{x}^i) \cdot \operatorname{grad} w(\mathbf{x}^i)]\, d\mathbf{x} \Big| \\ &\qquad\quad + Ch^2 \|wz\|_{2,1,\Omega^i} \end{aligned}$$

$$\leqslant Ch^2 \|wz\|_{2,1,\Omega^i},$$

so

$$\Big|\int_{\Omega} wz\, d\Omega - \sum_i \sum_{\nu} z(\mathbf{x}^{\nu}) \int_{\Omega^{i,\nu}} w(\mathbf{x})\, d\Omega\Big|$$
$$\leqslant C\,h^2 \sum_i \|wz\|_{2,1,\Omega^i} = C\,h^2 \|wz\|_{2,1,\Delta}. \quad \square$$

We are now ready to estimate the influence of the use of quadrature in the discretization. We split this influence in errors that are induced by the use of quadrature for the right hand side (f, t_h) (cf. Theorem 2.2), and errors that are induced by the use of quadrature for $a(\boldsymbol{\epsilon}_h^j, \boldsymbol{\epsilon}_h^k)$ (cf. Theorem 2.3). As t_h is piecewise constant, the use of (2.34) is equivalent to replacing f by a piecewise bilinear interpolation function f_h and using exact integration. Therefore we write the approximation of (f, t_h) by the quadrature rule (2.34) as (f_h, t_h).

THEOREM 2.2. *Let $(\boldsymbol{\sigma}_h, u_h)$ be the solution of*

$$\begin{aligned} a(\boldsymbol{\sigma}_h, \boldsymbol{\tau}_h) + b(\boldsymbol{\tau}_h, u_h) &= 0, & \forall \boldsymbol{\tau}_h \in V_h,\\ b(\boldsymbol{\sigma}_h, t_h) &= (f, t_h), & \forall t_h \in W_h, \end{aligned} \tag{2.48}$$

and let $(\hat{\boldsymbol{\sigma}}_h, \hat{u}_h)$ be the solution of

$$\begin{aligned} a(\hat{\boldsymbol{\sigma}}_h, \boldsymbol{\tau}_h) + b(\boldsymbol{\tau}_h, \hat{u}_h) &= 0, & \forall \boldsymbol{\tau}_h \in V_h,\\ b(\hat{\boldsymbol{\sigma}}_h, t_h) &= (f_h, t_h), & \forall t_h \in W_h. \end{aligned} \tag{2.49}$$

Under the conditions of theorem 2.1 we have

$$\begin{aligned} \|\boldsymbol{\sigma}_h - \hat{\boldsymbol{\sigma}}_h\|_0 &\leqslant Ch^2\ \|f\|_{2,\Delta},\\ \|u_h - \hat{u}_h\|_0 &\leqslant Ch^2\ \|f\|_{2,\Delta}. \end{aligned} \tag{2.50}$$

PROOF. First we prove that $\|\boldsymbol{\sigma}_h - \hat{\boldsymbol{\sigma}}_h\|_0^2 \leqslant Ch^2 \|f\|_{2,\Delta} \|u_h - \hat{u}_h\|_0$:

$$\begin{aligned} \|\boldsymbol{\sigma}_h - \hat{\boldsymbol{\sigma}}_h\|_0^2 &\leqslant C|\,a(\boldsymbol{\sigma}_h - \hat{\boldsymbol{\sigma}}_h, \boldsymbol{\sigma}_h - \hat{\boldsymbol{\sigma}}_h)|\\ &= C|\,b(\boldsymbol{\sigma}_h - \hat{\boldsymbol{\sigma}}_h, u_h - \hat{u}_h)|\\ &= C|(f - f_h, u_h - \hat{u}_h)|\\ &\leqslant Ch^2 \|f(u_h - \hat{u}_h)\|_{2,1,\Delta}\\ &\leqslant Ch^2 \|f\|_{2,\Delta} \|u_h - \hat{u}_h\|_0. \end{aligned} \tag{2.51}$$

Suppose that ψ is the solution of $\operatorname{div}(\operatorname{grad}\psi) = u_h - \hat{u}_h$. Let $\boldsymbol{\tau}_h = \Pi_h^{\sigma} \operatorname{grad}\psi$, then

$$\operatorname{div}\boldsymbol{\tau}_h = \operatorname{div}\Pi_h^{\sigma}\ \operatorname{grad}\psi = \Pi_h^{u}\ \operatorname{div}\operatorname{grad}\psi = u_h - \hat{u}_h,$$

so

$$\begin{aligned}\|u_h - \hat{u}_h\|_0^2 &= |\,b(\boldsymbol{\tau}_h, u_h - \hat{u}_h)\,| \\ &= |\,a(\boldsymbol{\sigma}_h - \hat{\boldsymbol{\sigma}}_h, \boldsymbol{\tau}_h)\,| \\ &\leqslant C\|\boldsymbol{\sigma}_h - \hat{\boldsymbol{\sigma}}_h\|_0 \ \|\Pi_h^\sigma \operatorname{grad}\psi\|_0 \\ &\leqslant C\|\boldsymbol{\sigma}_h - \hat{\boldsymbol{\sigma}}_h\|_0 \ \|\operatorname{grad}\psi\|_V \\ &\leqslant C\|\boldsymbol{\sigma}_h - \hat{\boldsymbol{\sigma}}_h\|_0 \ \|u_h - \hat{u}_h\|_0.\end{aligned}$$

Hence

$$\|u_h - \hat{u}_h\|_0 \leqslant C\|\boldsymbol{\sigma}_h - \hat{\boldsymbol{\sigma}}_h\|_0. \tag{2.52}$$

Combining (2.51) and (2.52) proves the theorem. □

Before we state a theorem on the influence of the use of quadrature in the evaluation of $a(\cdot,\cdot)$, we remark that the space V_h has the following inverse property:

$$h\|\boldsymbol{\sigma}_h\|_V \leqslant C\|\boldsymbol{\sigma}_h\|_0, \quad \forall \boldsymbol{\sigma}_h \in V_h. \tag{2.53}$$

In the proof of the theorem we need the following estimate for the norm $\|\cdot\|_{2,1,\Delta}$ that appears in the right hand side of (2.47).

LEMMA 2.2. *Let $\boldsymbol{\sigma}_h, \boldsymbol{\tau}_h \in V_h$ then*

$$\|\boldsymbol{\sigma}_h \cdot \boldsymbol{\tau}_h\|_{2,1,\Delta} \leqslant C\|\boldsymbol{\sigma}_h\|_V \, \|\boldsymbol{\tau}_h\|_V, \tag{2.54}$$

with C independent of h.

PROOF. By repeated use of Cauchy's inequality we obtain

$$\begin{aligned}\|\boldsymbol{\sigma}_h \cdot \boldsymbol{\tau}_h\|_{2,1,\Delta} &= \sum_i \sum_{|\alpha|\leqslant 2} \int_{\Omega^i} |D^\alpha(\boldsymbol{\sigma}_h \cdot \boldsymbol{\tau}_h)|\, d\Omega \\ &\leqslant C \sum_i \sum_{\substack{|\alpha|\leqslant 2\\ \alpha_1+\alpha_2=\alpha}} \sum_{d=x,y} \int_{\Omega^i} |D^{\alpha_1}\sigma_{h,d}|\,|D^{\alpha_2}\tau_{h,d}|\, d\Omega \\ &\leqslant C \sum_i \sum_{\substack{|\alpha|\leqslant 2\\ \alpha_1+\alpha_2=\alpha}} \sum_{d=x,y} [\int_{\Omega^i} |D^{\alpha_1}\sigma_{h,d}|^2\, d\Omega]^{\frac{1}{2}} [\int_{\Omega^i} |D^{\alpha_2}\tau_{h,d}|^2\, d\Omega]^{\frac{1}{2}} \\ &\leqslant C [\sum_i \sum_{|\alpha|\leqslant 2} \sum_{d=x,y} \int_{\Omega^i} |D^\alpha\sigma_{h,d}|^2\, d\Omega]^{\frac{1}{2}} [\sum_i \sum_{|\alpha|\leqslant 2} \sum_{d=x,y} \int_{\Omega^i} |D^\alpha\tau_{h,d}|^2\, d\Omega]^{\frac{1}{2}}.\end{aligned}$$

For $\boldsymbol{\sigma}_h \in V_h$ we have

$$\begin{aligned}&\sum_i \sum_{|\alpha|\leqslant 2} \sum_{d=x,y} \int_{\Omega^i} |D^\alpha\sigma_{h,d}|^2\, d\Omega \\ &\qquad = \sum_i \int_{\Omega^i} (|\sigma_{h,x}|^2 + |\sigma_{h,y}|^2 + |\partial_x\sigma_{h,x}|^2 + |\partial_y\sigma_{h,y}|^2)\, d\Omega\end{aligned}$$

$$= \|\boldsymbol{\sigma}_h\|_V^2. \quad \square$$

THEOREM 2.3. *Let $(\hat{\boldsymbol{\sigma}}_h, \hat{u}_h)$ be as in Theorem 2.2, and let $(\tilde{\boldsymbol{\sigma}}_h, \tilde{u}_h)$ be the solution of*

$$\begin{aligned} \tilde{a}(\tilde{\boldsymbol{\sigma}}_h, \boldsymbol{\tau}_h) + b(\boldsymbol{\tau}_h, \tilde{u}_h) &= 0, &\forall\, \boldsymbol{\tau}_h \in V_h, \\ b(\tilde{\boldsymbol{\sigma}}_h, t_h) &= (f_h, t_h), &\forall\, t_h \in W_h. \end{aligned} \tag{2.55}$$

If $\tilde{a}(\cdot\,,\cdot)$ is such that

$$\exists\ a_h > 0\colon\ \tilde{a}(\boldsymbol{\sigma}_h, \boldsymbol{\sigma}_h) \geqslant a_h \|\boldsymbol{\sigma}_h\|_0^2, \quad \forall\, \boldsymbol{\sigma} \in V_h \tag{2.56}$$

and the conditions of theorem 2.1 hold, then

$$\|\tilde{\boldsymbol{\sigma}}_h - \hat{\boldsymbol{\sigma}}_h\|_0 \leqslant Ch^2 \|A^{-1}\|_{2,\infty,\Omega}\ a_h{}^{-1}\ \|\hat{\boldsymbol{\sigma}}_h\|_V, \tag{2.57}$$

$$\|\tilde{u}_h - \hat{u}_h\|_0 \leqslant Ch^2 \|A^{-1}\|_{2,\infty,\Omega} (a_h{}^{-1} \|\hat{\boldsymbol{\sigma}}_h\|_V + \|\tilde{\boldsymbol{\sigma}}_h\|_V). \tag{2.58}$$

Moreover, if $\|\boldsymbol{\sigma}\|_V$ is bounded then $\|\hat{\boldsymbol{\sigma}}_h\|_V$ and $\|\tilde{\boldsymbol{\sigma}}_h\|_V$ are bounded.

PROOF. Subtracting (2.55) and (2.49) yields

$$b(\hat{\boldsymbol{\sigma}}_h - \tilde{\boldsymbol{\sigma}}_h, t_h) = 0, \quad \forall\, t_h \in W_h, \tag{2.59}$$

so

$$b(\hat{\boldsymbol{\sigma}}_h - \tilde{\boldsymbol{\sigma}}_h, \operatorname{div}(\hat{\boldsymbol{\sigma}}_h - \tilde{\boldsymbol{\sigma}}_h)) = \|\operatorname{div}(\hat{\boldsymbol{\sigma}}_h - \tilde{\boldsymbol{\sigma}}_h)\|_0^2 = 0. \tag{2.60}$$

By using (2.47), (2.54), (2.56) and (2.60) we obtain

$$\begin{aligned} a_h \|\tilde{\boldsymbol{\sigma}}_h - \hat{\boldsymbol{\sigma}}_h\|_0^2 &\leqslant \tilde{a}(\tilde{\boldsymbol{\sigma}}_h - \hat{\boldsymbol{\sigma}}_h, \tilde{\boldsymbol{\sigma}}_h - \hat{\boldsymbol{\sigma}}_h) \\ &\leqslant |\,\tilde{a}(\tilde{\boldsymbol{\sigma}}_h, \tilde{\boldsymbol{\sigma}}_h - \hat{\boldsymbol{\sigma}}_h) - a(\hat{\boldsymbol{\sigma}}_h, \tilde{\boldsymbol{\sigma}}_h - \hat{\boldsymbol{\sigma}}_h)\,| \\ &\qquad + |\,a(\hat{\boldsymbol{\sigma}}_h, \tilde{\boldsymbol{\sigma}}_h - \hat{\boldsymbol{\sigma}}_h) - \tilde{a}(\hat{\boldsymbol{\sigma}}_h, \tilde{\boldsymbol{\sigma}}_h - \hat{\boldsymbol{\sigma}}_h)\,| \\ &= |\,a(\hat{\boldsymbol{\sigma}}_h, \tilde{\boldsymbol{\sigma}}_h - \hat{\boldsymbol{\sigma}}_h) - \tilde{a}(\hat{\boldsymbol{\sigma}}_h, \tilde{\boldsymbol{\sigma}}_h - \hat{\boldsymbol{\sigma}}_h)\,| \\ &\leqslant C\,h^2 \|A^{-1} \hat{\boldsymbol{\sigma}}_h \cdot (\tilde{\boldsymbol{\sigma}}_h - \hat{\boldsymbol{\sigma}}_h)\|_{2,1,\Delta} \\ &\leqslant C\,h^2 \|A^{-1}\|_{2,\infty,\Omega}\ \|\hat{\boldsymbol{\sigma}}_h \cdot (\tilde{\boldsymbol{\sigma}}_h - \hat{\boldsymbol{\sigma}}_h)\|_{2,1,\Delta} \\ &\leqslant C\,h^2 \|A^{-1}\|_{2,\infty,\Omega}\ \|\hat{\boldsymbol{\sigma}}_h\|_V\ \|\tilde{\boldsymbol{\sigma}}_h - \hat{\boldsymbol{\sigma}}_h\|_V \\ &= C\,h^2 \|A^{-1}\|_{2,\infty,\Omega}\ \|\hat{\boldsymbol{\sigma}}_h\|_V\ \|\tilde{\boldsymbol{\sigma}}_h - \hat{\boldsymbol{\sigma}}_h\|_0. \end{aligned}$$

This proves the first part of the theorem. Next suppose that ψ is the solution of $\operatorname{div}(\operatorname{grad}\psi) = \tilde{u}_h - \hat{u}_h$. Let $\boldsymbol{\tau}_h = \Pi_h^{\sigma} \operatorname{grad}\psi$, then

$$\begin{aligned} \|\tilde{u}_h - \hat{u}_h\|_0^2 &= |\,b(\boldsymbol{\tau}_h, \tilde{u}_h - \hat{u}_h)\,| = |\,a(\hat{\boldsymbol{\sigma}}_h, \boldsymbol{\tau}_h) - \tilde{a}(\tilde{\boldsymbol{\sigma}}_h, \boldsymbol{\tau}_h)\,| \\ &\leqslant |\,a(\hat{\boldsymbol{\sigma}}_h - \tilde{\boldsymbol{\sigma}}_h, \boldsymbol{\tau}_h)\,| + |\,a(\tilde{\boldsymbol{\sigma}}_h, \boldsymbol{\tau}_h) - \tilde{a}(\tilde{\boldsymbol{\sigma}}_h, \boldsymbol{\tau}_h)\,| \\ &\leqslant C(\|\hat{\boldsymbol{\sigma}}_h - \tilde{\boldsymbol{\sigma}}_h\|_0\ \|\boldsymbol{\tau}_h\|_0 + h^2 \|A^{-1} \tilde{\boldsymbol{\sigma}}_h \cdot \boldsymbol{\tau}_h\|_{2,1,\Delta}) \end{aligned}$$

$$\leqslant C\, h^2 \|A^{-1}\|_{2,\infty,\Omega}(a_h{}^{-1}\, \|\hat{\boldsymbol{\sigma}}_h\|_V \|\boldsymbol{\tau}_h\|_0 + \|\tilde{\boldsymbol{\sigma}}_h\|_V\, \|\boldsymbol{\tau}_h\|_V)$$
$$\leqslant C\, h^2 \|A^{-1}\|_{2,\infty,\Omega}(a_h{}^{-1}\|\hat{\boldsymbol{\sigma}}_h\|_V + \|\tilde{\boldsymbol{\sigma}}_h\|_V)\, \|\operatorname{grad}\psi\|_V$$
$$\leqslant C\, h^2 \|A^{-1}\|_{2,\infty,\Omega}(a_h{}^{-1}\|\hat{\boldsymbol{\sigma}}_h\|_V + \|\tilde{\boldsymbol{\sigma}}_h\|_V)\, \|\tilde{u}_h - \hat{u}_h\|_0.$$

Now it remains to prove that $\|\hat{\boldsymbol{\sigma}}_h\|_V$ and $\|\tilde{\boldsymbol{\sigma}}_h\|_V$ are bounded. It follows from Theorem 2.1 that $\|\boldsymbol{\sigma}_h\|_V$ is bounded; from (2.50), (2.57) and (2.53) we obtain

$$\|\boldsymbol{\sigma}_h - \hat{\boldsymbol{\sigma}}_h\|_V \leqslant Ch\|f\|_{2,\Delta}$$

and

$$\|\tilde{\boldsymbol{\sigma}}_h - \hat{\boldsymbol{\sigma}}_h\|_V \leqslant Ch\|A^{-1}\|_{2,\infty,\Omega}a_h{}^{-1}\|\hat{\boldsymbol{\sigma}}_h\|_V,$$

therefore we conclude that $\|\hat{\boldsymbol{\sigma}}_h\|_V$ and $\|\tilde{\boldsymbol{\sigma}}_h\|_V$ are bounded. □

Combining the theorems 2.1, 2.2 and 2.3 we conclude that the quadrature rule (2.34) does not spoil the order of accuracy of the discretization, because the errors introduced by the quadrature are of the same or lower order than the discretization errors.

2.6. Discretization of the 2D semiconductor equations

To discretize the semiconductor equations we apply the discretization scheme presented in Section 2.5 to the system of equations written in Slotboom variables

$$\operatorname{div}\mathbf{j}_\psi = e^{-\psi}\Phi_p - e^{+\psi}\Phi_n + D, \tag{2.61a}$$
$$\operatorname{div}\mathbf{j}_n = +R\,, \tag{2.61b}$$
$$\operatorname{div}\mathbf{j}_p = -R\,, \tag{2.61c}$$
$$\mathbf{j}_\psi = -\mu_\psi \operatorname{grad}\psi, \tag{2.61d}$$
$$\mathbf{j}_n = +\mu_n e^{+\psi}\operatorname{grad}\Phi_n, \tag{2.61e}$$
$$\mathbf{j}_p = -\mu_p e^{-\psi}\operatorname{grad}\Phi_p, \tag{2.61f}$$

i.e. we apply the discretization procedure with $u = (\psi, \Phi_n, \Phi_p)$, $\boldsymbol{\sigma} = (\mathbf{j}_\psi, \mathbf{j}_n, \mathbf{j}_p)$ and $A = (-\mu_\psi, +\mu_n\exp(+\psi), -\mu_p\exp(-\psi))$, respectively. In order to obtain the usual definition of $(\mathbf{j}_\psi, \mathbf{j}_n, \mathbf{j}_p)$ (cf. [7]) we allow negative values for A. It appears that if lumping is used, we retain the Scharfetter-Gummel discretization [10] of the fluxes. For the continuity equations A^{-1} is the exponentially varying function $A^{-1} = e^{\pm\psi}$. If we approximate ψ in Δ_E^k by a linear function, interpolating ψ from its values ψ^R and ψ^L in the neighboring cells Ω^i, $i = R, L$ we obtain

$$\int_{\Delta_E^k} e^{\psi}\, d\Omega = \text{area}\,(\Delta_E^k)\ \text{Bexp}^{-1}\,(\psi^R, \psi^L), \tag{2.62}$$

with

$$\text{Bexp}\,(x, y) = \frac{x - y}{e^x - e^y}\,. \tag{2.63}$$

So for an edge E^j with adjacent cells Ω^i, $i=R,L$, we obtain from (2.61) and (2.62)

$$\begin{aligned} j_\psi^j &= -\frac{h^j}{a_E^j}\mu_\psi(\psi^R-\psi^L), \\ j_n^j &= +\frac{h^j}{a_E^j}\mu_n \operatorname{Bexp}(-\psi^R,-\psi^L)(e^{-\phi_n^R}-e^{-\phi_n^L}), \\ j_p^j &= -\frac{h^j}{a_E^j}\mu_p \operatorname{Bexp}(+\psi^R,+\psi^L)(e^{+\phi_p^R}-e^{+\phi_p^L}), \end{aligned} \tag{2.64}$$

with h^j the length of E^j and $a_E^j = \text{area}(\Delta_E^j)$. Moreover for a cell Ω^i with edges E^j, $j=n,e,s,w$ we have

$$h^n j_\psi^n + h^e j_\psi^e - h^s j_\psi^s - h^w j_\psi^w = a^i(e^{\phi_p^i-\psi^i} - e^{\psi^i-\phi_n^i} + \tilde{D}), \tag{2.65a}$$

$$h^n j_n^n + h^e j_n^e - h^s j_n^s - h^w j_n^w = +a^i R(\psi^i, \phi_n^i, \phi_p^i), \tag{2.65b}$$

$$h^n j_p^n + h^e j_p^e - h^s j_p^s - h^w j_p^w = -a^i R(\psi^i, \phi_n^i, \phi_p^i), \tag{2.65c}$$

with

$$\tilde{D} = \frac{1}{4}\sum_{\nu=1,4} D(\mathbf{x}^{i,\nu}),$$

a^i the area of cell Ω^i, h^j the length of the edge E^j, D the given dope function and R the recombination rate of electrons and holes.

We observe that the variables j_ψ^j, j_n^j and j_p^j may be eliminated to yield a scheme that is equivalent to the usual box scheme (see e.g. [7]) in the interior of the domain Ω. However the geometry of our discretization is cell-centered as opposed to the usual box scheme that is vertex-centered.

Finally, to obtain values of the potentials at the edges, we calculate the Lagrange multipliers λ_ψ^j, λ_n^j and λ_p^j for the semiconductor equations. For an edge E^j with adjacent cells Ω^L and Ω^R we find for Poisson's equation (cf. 2.38)

$$\lambda_\psi^j = \frac{a^R\psi^L + a^L\psi^R}{a^L + a^R}, \tag{2.66a}$$

and for the continuity eqations

$$\begin{aligned} e^{\lambda_p^j} &= \frac{\frac{1}{2}a^R e^{\phi_p^L}\operatorname{Bexp}^{-1}(\psi^R,\lambda_\psi^j) + \frac{1}{2}a^L e^{\phi_p^R}\operatorname{Bexp}^{-1}(\lambda_\psi^j,\psi^L)}{\frac{1}{2}(a^L+a^R)\operatorname{Bexp}^{-1}(\psi^R,\psi^L)} \\ &= \frac{e^{\phi_p^L}(e^{\psi^R}-e^{\lambda_\psi^j}) + e^{\phi_p^R}(e^{\lambda_\psi^j}-e^{\psi^L})}{e^{\psi^R}-e^{\psi^L}} \end{aligned} \tag{2.66b}$$

and

$$e^{-\lambda_n^j} = \frac{e^{-\phi_n^L}(e^{-\psi^R}-e^{-\lambda_\psi^j}) + e^{-\phi_n^R}(e^{-\lambda_\psi^j}-e^{-\psi^L})}{e^{-\psi^R}-e^{-\psi^L}}. \tag{2.66c}$$

We remark that (2.66) comes down to linear interpolation for Poisson's equation and exponential interpolation for the continuity equations, as was used for the one-dimensional case by Hemker [6].

2.7. Concluding remarks

We have derived a dual mixed finite element discretization of a model problem on a rectangular grid. The straightforward discretization is not stable in the sense that the system of eqations obtained after elimination of the variable $\boldsymbol{\sigma}$ is an M-matrix. Therefore we use a quadrature rule that lumps the discrete system; after the elimination of $\boldsymbol{\sigma}$ we do obtain an M-matrix. We prove that the use of this quadrature rule does not influence the accuracy of the discrete approximation. When applied to the stationary semiconductor equations, we obtain a generalization in two dimensions of the classical one-dimensional Scharfetter-Gummel scheme.

References

1. D.N. Arnold and F. Brezzi (1985). Mixed and non-conforming finite element methods: implementation, postprocessing and error estimates, *MMAN*, 19, 7-32.
2. F. Brezzi, L.D. Marini, and P. Pietra (1989). Two-dimensional exponential fitting and applications to drift-diffusion models, *SIAM J.Num.Anal.*, 26, 1342-1355.
3. J. Douglas and J.E. Roberts (1982). Mixed finite element methods for second order elliptic problems, *Mat.Aplic.Comp.*, 1, 91-103.
4. J. Douglas and J.E. Roberts (1985). Global Estimates for Mixed Methods for Second Order Elliptic Equations, *Math. of Comp.*, 44, 39-52.
5. M. Fortin (1977). An analysis of the convergence of mixed finite element methods, *RAIRO Num.Anal.*, 11, 341-354.
6. P.W. Hemker (1990). A nonlinear multigrid method for one-dimensional semiconductor device simulation: results for the diode, *J.Comp.Appl.Math.*, 30, 117-126.
7. S.J. Polak, C. den Heijer, W.H.A. Schilders, and P. Markowich (1987). Semiconductor device modelling from the numerical point of view, *Int.J.Num.Meth.Engng.*, 24, 763-838.
8. S.J. Polak, W.H.A. Schilders, and H.D. Couperus (1988). A finite element method with current conservation, in *Proc. SISDEP-88*, 453-462, ed. G. Baccarani and M. Rudan, Bologna.
9. P.A. Raviart and J.M. Thomas (1977). A mixed finite element method for second order elliptic problems, in *Mathematical aspects of the finite element method*, Springer-Verlag, Lecture Notes in Mathematics 606.
10. D.L. Scharfetter and H.K. Gummel (1969). Large-Signal Analysis of a Silicon Read Diode Oscillator, *IEEE Trans.E.D.*, ED-16, 64-77.
11. B.X. Fraeijs de Veubeke (1965). Displacement and equilibrium models in the finite element method, in *Stress analysis*, ed. O.C. Zienkiewicz and G. Hollister, John Wiley, New York.

Chapter 3

Optimization problems and relaxation methods

3.1. Introduction

In this Chapter we show that the variational formulation of the mixed finite element method is equivalent to a constrained optimization problem for a class of nonlinear problems. Moreover, we present two different relaxation methods for those problems that minimize appropriate functionals. In this way their convergence can be proved. By local mode analysis we further study the feasibility of these relaxation methods as smoothers in a multigrid algorithm.

As a starting point for our discussion we take the variational formulation (2.12): find $(\boldsymbol{\sigma}, u) \in V \times W$ such that

$$a(\boldsymbol{\sigma}, \boldsymbol{\tau}) + b(\boldsymbol{\tau}, u) = \langle g, \boldsymbol{\tau} \rangle_{\delta\Omega_D}, \quad \forall \boldsymbol{\tau} \in V, \tag{3.1a}$$

$$b(\boldsymbol{\sigma}, t) \qquad\qquad = (f, t), \qquad \forall t \in W. \tag{3.1b}$$

It is well known that for $f = F(\mathbf{x})$ problem (3.1) can be considered as a constrained minimization problem of the energy functional with (3.1b) as the constraint (cf. [2]). Schmidt and Jacobs [4] showed that also for linear sources $f = cu + F(\mathbf{x})$, with $c \geqslant 0$, problem (3.1) can be reformulated as a constrained minimization problem. We consider the class of nonlinear sources $f = f(\mathbf{x}, u)$, with

$$f(\mathbf{x}, u) = F(\mathbf{x}) + \mathcal{F}(u(\mathbf{x})), \tag{3.2a}$$

and $\mathcal{F}: \mathbb{R} \rightarrow \mathbb{R}$ strongly (positive) monotone (cf. e.g. [3], appendix A), i.e. $\mathcal{F} \in C^1(\mathbb{R})$ and F' bounded away from zero, i.e. there is a $\beta > 0$, such that

$$\mathcal{F}'(x) \geqslant \beta, \quad \forall\, x \in \mathbb{R}. \tag{3.2b}$$

Also for this kind of nonlinear sources f it is possible to reformulate problem (3.1) as a constrained minimization problem with (3.1b) as the constraint. We remark that the class of nonlinear problems that we consider includes the nonlinear Poisson equation in the semiconductor equations (2.61a), if ϕ_n and ϕ_p are given:

$$f(\mathbf{x}, \psi) = e^{\psi - \phi_n} - e^{\phi_p - \psi} - D(\mathbf{x}) \tag{3.3a}$$

and

$$\frac{d(e^{\psi - \phi_n} - e^{\phi_p - \psi})}{d\psi} = e^{\psi - \phi_n} + e^{\phi_p - \psi} \geqslant 2e^{\frac{1}{2}(\phi_p - \phi_n)}. \tag{3.3b}$$

Instead of using (3.1b) as the constraint, it is also possible to formulate the variational problem (3.1) as an optimization problem with (3.1a) as the constraint. In Section 3.2 we present these two different optimization problems and prove that they are both equivalent to problem (3.1).

For the discretized version of (3.1), i.e. with $V = V_h$ and $W = W_h$ (cf. Section 2.4), we consider two relaxation methods. In [4] Schmidt and Jacobs present a superbox relaxation for the linear source problem with homogeneous Neumann boundary conditions. As the relaxation minimizes the energy functional and all iterates satisfy the constraint (3.1b), they are able to prove convergence of the superbox relaxation method. We extend this approach to the nonlinear case and to more general boundary conditions in Section 3.4. Before that, we present in Section 3.3 a relaxation method proposed by Vanka [6] for the solution of the incompressible Navier-Stokes equations, that is easily adapted for the mixed finite element discretization. When the discrete equations are lumped, the iterates in this Vanka-type relaxation satisfy the constraint (3.1a). Therefore we can also consider Vanka-type relaxation as the minimization of a functional and prove its convergence for the nonlinear problem.

Guaranteed convergence is of course a nice property for a relaxation operator, but it does not imply that the relaxation operator is useful as a smoother in a multigrid algorithm, where the relaxation operator should remove high frequency Fourier components in the error. For both superbox relaxation and Vanka-type relaxation we carry out a Fourier local mode analysis for the linear source problem $f = cu + F(\mathbf{x})$ with $c \geqslant 0$. It is found that Vanka-type relaxation is the more efficient smoother.

3.2. Optimization problems

In this Section we show the equivalence of the variational problem (3.1) and two constrained optimization problems. We start by defining the subspaces $\Lambda^V, \Lambda^S \subset V \times W$ in which one of the constraints (3.1a) or (3.1b) hold:

$$\Lambda^V = \{ (\boldsymbol{\sigma}, u) \in V \times W \mid a(\boldsymbol{\sigma}, \boldsymbol{\tau}) + b(\boldsymbol{\tau}, u) = \langle g, \boldsymbol{\tau} \rangle_{\delta\Omega_D}, \ \forall \boldsymbol{\tau} \in V \} \tag{3.4a}$$

and

$$\Lambda^S = \{ (\boldsymbol{\sigma}, u) \in V \times W \mid b(\boldsymbol{\sigma}, t) = (f, t), \ \forall t \in W \}. \tag{3.4b}$$

We notice that the vector space Λ^V is convex, i.e. for $(\boldsymbol{\sigma}^1, u^1)$, $(\boldsymbol{\sigma}^2, u^2) \in \Lambda^V$ we have

$$\lambda(\boldsymbol{\sigma}^1, u^1) + (1-\lambda)(\boldsymbol{\sigma}^2, u^2) \in \Lambda^V, \qquad \text{for } \lambda \in [0,1].$$

The minimizing functionals are defined by

$$F^V(\boldsymbol{\sigma}, u) = \frac{1}{2} a(\boldsymbol{\sigma}, \boldsymbol{\sigma}) + \int_\Omega R^V(\mathbf{x}, u(\mathbf{x}))\, d\mathbf{x} \tag{3.5a}$$

and

$$F^S(\boldsymbol{\sigma}, u) = \frac{1}{2}a(\boldsymbol{\sigma},\boldsymbol{\sigma}) - \langle g, \boldsymbol{\sigma}\rangle_{\delta\Omega_D} + \int_\Omega R^S(u(\mathbf{x}))\,d\mathbf{x}, \tag{3.5b}$$

with $R^V : \Omega \times \mathbb{R} \to \mathbb{R}$ defined by

$$\frac{\partial R^V(\mathbf{x}, u)}{\partial u} = f(\mathbf{x}, u) \tag{3.6a}$$

and $R^S : \mathbb{R} \to \mathbb{R}$ defined by

$$\frac{dR^S(u)}{du} = u\,\mathfrak{F}'(u). \tag{3.6b}$$

In the following two examples we give these functions R for the linear source problem and the nonlinear Poisson equation in the semiconductor equations.

Example 3.1. In the linear source problem we have $f = F(\mathbf{x}) + cu$, so

$$R^V(\mathbf{x}, u) = uF(\mathbf{x}) + \frac{1}{2}cu^2, \tag{3.7a}$$

$$R^S(u) = \frac{1}{2}cu^2. \tag{3.7b}$$

Example 3.2. In the nonlinear Poisson equation (2.61a) f is as in (3.3a), so

$$R^V(\mathbf{x}, \psi) = e^{\psi-\phi_n} + e^{\phi_p-\psi} - D\psi, \tag{3.8a}$$

$$R^S(\psi) = (\psi-1)e^{\psi-\phi_n} - (\psi+1)e^{\phi_p-\psi}. \tag{3.8b}$$

The optimization problems for the continuous problem (3.1) are stated in the Theorems 3.1 and 3.2; for the discrete problem the corresponding optimization problems are given in the Theorems 3.3 and 3.4. From (3.5a), (3.6a) and (3.2) we see that the functional F^V is convex. As Λ^V is a convex space and F^V is a convex functional we can prove the uniqueness of the solution of problem (3.1) with f as in (3.2). In the proofs we make use of the following Taylor expansion: if $R \in C^3(\mathbb{R})$ then

$$\int_\Omega (R(u(\mathbf{x})+t(\mathbf{x})) - R(u(\mathbf{x})))\,d\mathbf{x}$$
$$= (R'(u), t) + \frac{1}{2}(R''(u), t^2) + \frac{1}{6}(R'''(u+\tilde{t}), t^3),$$

with $\tilde{t}(\mathbf{x}) = u(\mathbf{x}) + \mu(\mathbf{x})t(\mathbf{x})$ and $0 \leqslant \mu \leqslant 1$.

Theorem 3.1. *Suppose that problem (3.1) has a solution* $(\boldsymbol{\sigma}, u)$. *If* $a(\cdot\,,\cdot)$ *is* L^2*-coercive and* $\mathfrak{F} \in C^3(\mathbb{R})$ *is strongly monotone then the solution* $(\boldsymbol{\sigma}, u)$ *is unique and* F^V *has a unique global minimum in* Λ^V *for* $(\boldsymbol{\sigma}, u)$.

Proof. Let $(\boldsymbol{\sigma}, u)$ be a solution of problem (3.1) then $(\boldsymbol{\sigma}, u) \in \Lambda^V$. If $(\boldsymbol{\sigma}+\lambda\boldsymbol{\tau}, u+\lambda t) \in \Lambda^V$, with $\lambda \in \mathbb{R}$, then (3.4a) with $\boldsymbol{\sigma}$ as the test function implies

$$a(\lambda\boldsymbol{\tau},\boldsymbol{\sigma}) + b(\boldsymbol{\sigma}, \lambda t) = 0. \tag{3.9}$$

Using (3.5a), (3.9) and a Taylor expansion of R^V we obtain for λ small enough

$$\begin{aligned}
&F^V(\boldsymbol{\sigma}+\lambda\boldsymbol{\tau}, u+\lambda t) - F^V(\boldsymbol{\sigma}, u)\\
&\quad= \lambda a(\boldsymbol{\sigma}, \boldsymbol{\tau}) + \frac{\lambda^2}{2} a(\boldsymbol{\tau}, \boldsymbol{\tau}) + \int_\Omega (R^V(\mathbf{x}, u+\lambda t) - R^V(\mathbf{x}, u))\,d\mathbf{x}\\
&\quad= \lambda a(\boldsymbol{\sigma}, \boldsymbol{\tau}) + \frac{\lambda^2}{2} a(\boldsymbol{\tau}, \boldsymbol{\tau}) + \lambda(\frac{\partial R^V}{\partial u}, t) + \frac{\lambda^2}{2}(\frac{\partial^2 R^V}{\partial u^2}, t^2) + \mathcal{O}(\lambda^3)\\
&\quad= -\lambda b(\boldsymbol{\sigma}, t) + \frac{\lambda^2}{2} a(\boldsymbol{\tau}, \boldsymbol{\tau}) + \lambda(f, t) + \frac{\lambda^2}{2}(\mathcal{F}'(u), t^2) + \mathcal{O}(\lambda^3)\\
&\quad= \frac{\lambda^2}{2} a(\boldsymbol{\tau}, \boldsymbol{\tau}) + \frac{\lambda^2}{2}(\mathcal{F}'(u), t^2) + \mathcal{O}(\lambda^3).
\end{aligned}$$

This shows that F^V has a local minimum in Λ^V for $(\boldsymbol{\sigma}, u)$.

As Λ^V is a convex space and F^V is a convex functional, F^V has at most one global minimum (see e.g. [1]), so if (3.1) has a solution then this solution is unique. □

THEOREM 3.2. *Let $a(\cdot,\cdot)$ be L^2-coercive and $\mathcal{F} \in C^3(\mathbb{R})$ strongly monotone. If $\mathcal{F}\colon \mathbb{R} \to \mathbb{R}$ is invertible and $\mathcal{F}^{-1} \in C^0(\mathbb{R})$ then $(\boldsymbol{\sigma}, u)$ is the unique solution of (3.1) if and only if F^S has a unique local minimum in Λ^S for $(\boldsymbol{\sigma}, u)$. Moreover, any local minimum of F^S in Λ^S is also the global minimum.*

PROOF. Let $(\boldsymbol{\sigma}, u) \in \Lambda^S$ be the unique solution of (3.1). For $(\boldsymbol{\sigma}+\lambda\boldsymbol{\tau}, u+\lambda t) \in \Lambda^S$ we obtain from (3.4b) with u as the test function and λ small enough

$$\begin{aligned}
b(\lambda\boldsymbol{\tau}, u) &= (f(\mathbf{x}, u+\lambda t) - f(\mathbf{x}, u), u)\\
&= (\lambda t\mathcal{F}'(u) + \frac{\lambda^2}{2} t^2 \mathcal{F}''(u), u) + \mathcal{O}(\lambda^3).
\end{aligned} \tag{3.10}$$

Using (3.5b), (3.6b), (3.10) and a Taylor expansion of R^S we obtain

$$\begin{aligned}
&F^S(\boldsymbol{\sigma}+\lambda\boldsymbol{\tau}, u+\lambda t) - F^S(\boldsymbol{\sigma}, u)\\
&\quad= \lambda a(\boldsymbol{\sigma},\boldsymbol{\tau}) + \frac{\lambda^2}{2} a(\boldsymbol{\tau},\boldsymbol{\tau}) - \lambda\langle g, \boldsymbol{\tau}\rangle_{\delta\Omega_D} + \int_\Omega (R^S(u+\lambda t) - R^S(u))\,d\mathbf{x}\\
&\quad= \lambda a(\boldsymbol{\sigma},\boldsymbol{\tau}) + \frac{\lambda^2}{2} a(\boldsymbol{\tau},\boldsymbol{\tau}) - \lambda\langle g, \boldsymbol{\tau}\rangle_{\delta\Omega_D} + \lambda(\frac{dR^S}{du}, t) + \frac{\lambda^2}{2}(\frac{d^2R^S}{du^2}, t^2) + \mathcal{O}(\lambda^3)\\
&\quad= \lambda a(\boldsymbol{\sigma},\boldsymbol{\tau}) + \frac{\lambda^2}{2} a(\boldsymbol{\tau},\boldsymbol{\tau}) - \langle\lambda g, \boldsymbol{\tau}\rangle_{\delta\Omega_D} + \lambda(u\mathcal{F}'(u), t) + \frac{\lambda^2}{2}(u\mathcal{F}''(u), t^2) +\\
&\qquad \frac{\lambda^2}{2}(\mathcal{F}'(u), t^2) + \mathcal{O}(\lambda^3)
\end{aligned}$$

$$= \lambda a(\boldsymbol{\sigma},\boldsymbol{\tau}) + \lambda b(\boldsymbol{\tau}, u) - \lambda\langle g, \boldsymbol{\tau}\rangle_{\delta\Omega_D} + \frac{\lambda^2}{2}a(\boldsymbol{\tau},\boldsymbol{\tau}) + \frac{\lambda^2}{2}(\mathcal{F}'(u), t^2) + \mathcal{O}(\lambda^3)$$

$$= \frac{\lambda^2}{2}a(\boldsymbol{\tau},\boldsymbol{\tau}) + \frac{\lambda^2}{2}(\mathcal{F}'(u), t^2) + \mathcal{O}(\lambda^3).$$

This shows that F^S has a local minimum in Λ^S for $(\boldsymbol{\sigma}, u)$.

Conversely, suppose that F^S has a local minimum in Λ^S for $(\boldsymbol{\sigma}, u)$. Due to the fact that $\mathcal{F}$ has a continuous inverse we can find for all $\boldsymbol{\tau} \in V$ and λ, $0 \leqslant |\lambda| \ll 1$, a $t \in W$ such that $(\boldsymbol{\sigma}+\lambda\boldsymbol{\tau}, u+\lambda t) \in \Lambda^S$. This implies

$$a(\boldsymbol{\sigma}, \lambda\boldsymbol{\tau}) + b(\lambda\boldsymbol{\tau}, u) - \langle g, \lambda\boldsymbol{\tau}\rangle_{\delta\Omega_D} = 0,$$

so $(\boldsymbol{\sigma}, u)$ is the solution of (3.1). As (3.1) has at most one solution (cf. Theorem 3.1), we conclude that any local minimum of F^S in Λ^S is unique. Finally, if $(\boldsymbol{\sigma}, u)$ is the solution of (3.1) and $(\boldsymbol{\sigma}+\boldsymbol{\tau}, u+t) \in \Lambda^S$ then using (3.10) we obtain

$$\begin{aligned} &F^S(\boldsymbol{\sigma}+\boldsymbol{\tau}, u+t) - F^S(\boldsymbol{\sigma}, u) \\ &\quad = \frac{1}{2}a(\boldsymbol{\tau},\boldsymbol{\tau}) + a(\boldsymbol{\sigma},\boldsymbol{\tau}) - \langle g, \boldsymbol{\tau}\rangle_{\delta\Omega_D} + \int_\Omega (R^S(u+t) - R^S(u))\,d\mathbf{x} \\ &\quad = \frac{1}{2}a(\boldsymbol{\tau},\boldsymbol{\tau}) - b(\boldsymbol{\tau}, u) + \int_\Omega (R^S(u+t) - R^S(u))\,d\mathbf{x} \\ &\quad = \frac{1}{2}a(\boldsymbol{\tau},\boldsymbol{\tau}) - (f(\mathbf{x}, u+t) - f(\mathbf{x}, u), u) + \int_\Omega (R^S(u+t) - R^S(u))\,d\mathbf{x} \\ &\quad = \frac{1}{2}a(\boldsymbol{\tau},\boldsymbol{\tau}) + \int_\Omega (G_u(t(\mathbf{x})) - G_u(0))\,d\mathbf{x}, \end{aligned}$$

with $G_u : \mathbb{R} \to \mathbb{R}$ defined by

$$G_u(x) = R^S(u+x) - u\,\mathcal{F}(u+x).$$

For this G_u we have

$$G_u{}'(x) = (u+x)\mathcal{F}'(u+x) - u\mathcal{F}'(u+x) = x\mathcal{F}'(u+x),$$

so G_u has a global minimum for $x=0$, and therefore we conclude that F^S has a global minimum in Λ^S for $(\boldsymbol{\sigma}, u)$. □

The problem of solving the discretized equations, i.e. find $(\boldsymbol{\sigma}_h, u_h) \in V_h \times W_h$ such that

$$a(\boldsymbol{\sigma}_h, \boldsymbol{\tau}_h) + b(\boldsymbol{\tau}_h, u_h) = \langle g, \boldsymbol{\tau}_h\rangle_{\delta\Omega_D}, \quad \forall \boldsymbol{\tau}_h \in V_h, \tag{3.11a}$$

$$b(\boldsymbol{\sigma}_h, t_h) \qquad = (f, t_h), \qquad \forall t_h \in W_h, \tag{3.11b}$$

with V_h and W_h the spaces spanned by the lowest order Raviart-Thomas elements (cf. Section 2.4) can also be considered as a constrained minimization problem. The subspaces in which the minima are to be found, i.e. the constraints, are the discrete equivalents of Λ^V and Λ^S,

$$\Lambda_h^V = \{ (\boldsymbol{\sigma}_h, u_h) \in V_h \times W_h \mid a(\boldsymbol{\sigma}_h, \boldsymbol{\tau}_h) + b(\boldsymbol{\tau}_h, u_h) = \langle g, \boldsymbol{\tau}_h \rangle_{\delta\Omega_D}, \ \forall \boldsymbol{\tau}_h \in V_h \} \tag{3.12a}$$

and

$$\Lambda_h^S = \{ (\boldsymbol{\sigma}_h, u_h) \in V_h \times W_h \mid b(\boldsymbol{\sigma}_h, t_h) = (f, t_h), \ \forall t_h \in W_h \}. \tag{3.12b}$$

The variational formulations for the discretized problem (3.11) are now stated in the Theorems 3.3 and 3.4.

THEOREM 3.3. *Let the bilinear form* $a: V_h \times V_h \to \mathbb{R}$ *be* L^2*-coercive, and let* $\mathcal{F} \in C^3(\mathbb{R})$ *be strongly monotone then* $(\boldsymbol{\sigma}_h, u_h)$ *is the unique solution of (3.11) if and only if* F_h^V *has a unique minimum in* Λ_h^V *for* $(\boldsymbol{\sigma}_h, u_h)$.

PROOF. It follows from the proof of Theorem 3.1 that any solution $(\boldsymbol{\sigma}_h, u_h)$ of problem (3.11) is unique and that F_h^V has a global minimum in Λ_h^V for $(\boldsymbol{\sigma}_h, u_h)$. Conversely, let $(\boldsymbol{\sigma}_h, u_h)$ be a local minimum of F_h^V in Λ_h^V. In the finite dimensional case we can find for any $t_h \in W_h$ a $\boldsymbol{\tau}_h \in V_h$ such that $(\boldsymbol{\sigma}_h + \boldsymbol{\tau}_h, u_h + t_h) \in \Lambda_h^V$. As Λ_h^V is convex we observe that $(\boldsymbol{\sigma}_h + \lambda\boldsymbol{\tau}_h, u_h + \lambda t_h) \in \Lambda_h^V$ for $0 \leqslant \lambda \leqslant 1$. If F_h^V has a local minimum for $(\boldsymbol{\sigma}_h, u_h)$ then for $0 \leqslant \lambda \ll 1$ we have $b(\boldsymbol{\sigma}_h, \lambda t_h) = (f, \lambda t_h)$. So $(\boldsymbol{\sigma}_h, u_h)$ is also the unique solution of (3.11). □

THEOREM 3.4. *Let the bilinear form* $a: V_h \times V_h \to \mathbb{R}$ *be* L^2*-coercive, and let* $\mathcal{F} \in C^3(\mathbb{R})$ *be as in Theorem 3.2, then* $(\boldsymbol{\sigma}_h, u_h)$ *is the unique solution of (3.11) if and only if* F_h^S *has a unique minimum in* Λ_h^S *for* $(\boldsymbol{\sigma}_h, u_h)$.

PROOF. This is a consequence of Theorem 3.2. □

We notice that the equivalence of the variational formulations and the optimization problems also holds in the case that the discrete equations are lumped (cf. Section 2.4), because the bilinear form $\tilde{a}(\cdot, \cdot)$ in (2.35) is symmetric and L^2-coercive.

3.3. VANKA-TYPE RELAXATION

In [6] Vanka proposes a relaxation method for the discretization of the incompressible Navier-Stokes equations on a staggered grid with pressures at the cell centres and velocities at the cell edges. In the relaxation the cells in the grid are scanned in some pre-determined order. When a cell is visited the pressure related with that cell, and the velocities at its four edges are relaxed simultaneously. After updating the pressure and the velocities the next cell is visited.

This relaxation procedure can be seen as a block Gauss-Seidel relaxation, but we notice that during a single relaxation sweep the velocities are updated twice. Of course we can define a block Gauss-Seidel method in which the pressure in a cell and the velocities at only two of its edges are updated

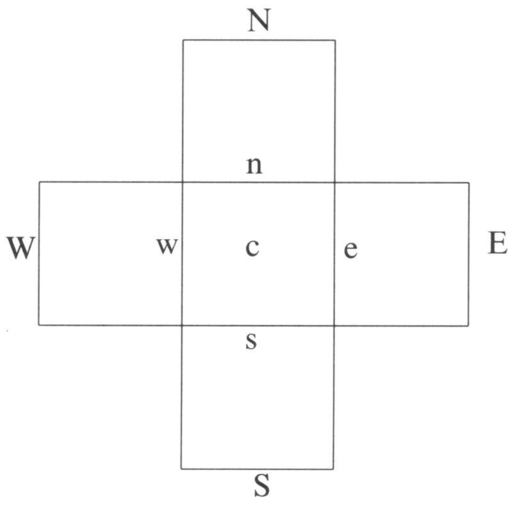


FIGURE 3.1. Relaxation subdomain for Vanka-type relaxation.

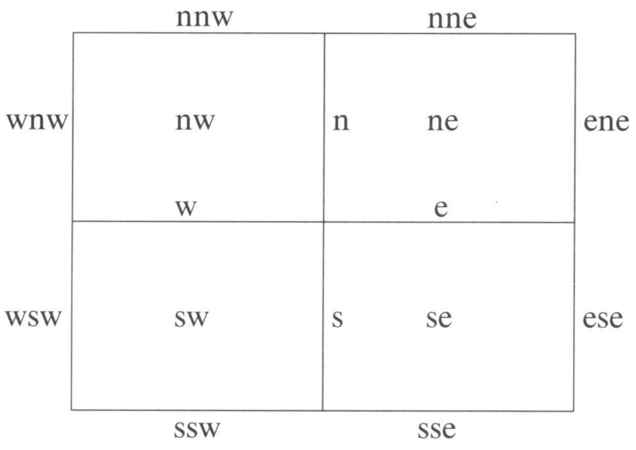


FIGURE 3.2. Relaxation subdomain for superbox relaxation.

simultaneously; this is sometimes called three-point Vanka relaxation. However, in Section 3.5 we show that three-point Vanka-type relaxation can not be used as a smoother for the mixed finite element discretization of Poisson's equation.

It is straightforward to generalize Vanka relaxation to the system of equations obtained by the mixed finite element discretization. During each visit to a cell Ω^c we relax the variable u^c and the variables σ^j, $j=n,e,s,w$, that are defined at the edges E^j of the cell Ω^c (cf. Figure 3.1). For a rectangular grid with mesh size (h_x, h_y) this means that we solve the system of equations

$$
\begin{aligned}
h_x\sigma^n + h_y\sigma^e - h_x\sigma^s - h_y\sigma^w - h_x h_y \mathcal{F}(u^c) &= F^c, \\
a^{n,n}\sigma^n + a^{n,s}\sigma^s - \frac{1}{h_y}(u^n - u^c) &= -a^{n,N}\sigma^N, \\
a^{e,e}\sigma^e + a^{e,w}\sigma^w - \frac{1}{h_x}(u^e - u^c) &= -a^{e,E}\sigma^E, \\
a^{s,s}\sigma^s + a^{s,n}\sigma^n - \frac{1}{h_y}(u^c - u^s) &= -a^{s,S}\sigma^S, \\
a^{w,w}\sigma^w + a^{w,e}\sigma^e - \frac{1}{h_x}(u^c - u^w) &= -a^{w,W}\sigma^W,
\end{aligned}
\tag{3.13}
$$

with

$$
a^{j,k} = \frac{1}{\text{area}(\Omega^c)} a(\boldsymbol{\epsilon}_h^j, \boldsymbol{\epsilon}_h^k), \tag{3.14}
$$

$$
F^c = \int_{\Omega^c} F(\mathbf{x})\, d\mathbf{x}. \tag{3.15}
$$

Dirichlet boundary conditions, $E^j \subset \delta\Omega_D$, are treated by replacing u^j in (3.13) by the Dirichlet boundary condition g^j, $g^j = \langle g, \boldsymbol{\epsilon}_h^j \rangle_{\delta\Omega_D}$, and in case of homogeneous Neumann boundary conditions, $E^j \subset \delta\Omega_N$, we replace the corresponding equation in (3.13) by the dummy equation $\sigma^j = 0$.

If the discrete equations are lumped, we trivially eliminate the fluxes σ^j from (3.13), in order to obtain

$$
\begin{aligned}
&\left(\frac{h_x}{h_y}\frac{1}{a^n} + \frac{h_y}{h_x}\frac{1}{a^e} + \frac{h_x}{h_y}\frac{1}{a^s} + \frac{h_y}{h_x}\frac{1}{a^w}\right) u^c + h_x h_y \mathcal{F}(u^c) = \\
&\qquad \frac{h_x}{h_y}\frac{1}{a^n}u^n + \frac{h_y}{h_x}\frac{1}{a^e}u^e + \frac{h_x}{h_y}\frac{1}{a^s}u^s + \frac{h_y}{h_x}\frac{1}{a^w}u^w - F^c,
\end{aligned}
\tag{3.16}
$$

with

$$
a^j = \frac{1}{\text{area}(\Omega^c)} \tilde{a}(\boldsymbol{\epsilon}_h^j, \boldsymbol{\epsilon}_h^j).
$$

We rewrite the equation (3.16) as

$$
\tilde{f}(u) = c_1 u + c_2 \mathcal{F}(u) = \tilde{c}, \tag{3.17}
$$

with $c_1, c_2 > 0$. From the fact that $\tilde{\mathcal{F}}$ is continuous and strictly (positive) monotone, we conclude that $\tilde{f}$ is continuous and strictly (positive) monotone. Moreover, we have

$$\lim_{u \to +\infty} \tilde{f}(u) = +\infty$$

and

$$\lim_{u \to -\infty} \tilde{f}(u) = -\infty,$$

therefore we conclude that (3.17) has a unique solution u for any $\tilde{c} \in \mathbb{R}$, thus problem (3.16) has a unique solution for any right hand side .

For the lumped equations the σ^j only depends on the u^i, $i = L,R$, in the cells Ω^i adjacent to the edge E^j. As u^i is always updated simultaneously with σ^j, it is a property of Vanka-type relaxation that all equations related to the the edges are satisfied as soon as a complete relaxation sweep has been performed. This makes it possible to consider Vanka-type relaxation, applied to the lumped equations, as the minimization of the functional F_h^V in the subspace Λ_h^V (cf.3.12a). Suppose that we have some iterate $(\boldsymbol{\sigma}_h, u_h) \in \Lambda_h^V$. We minimize $F_h^V(\boldsymbol{\sigma}_h + \boldsymbol{\tau}_h, u_h + t_h)$ in Λ_h^V for $(\boldsymbol{\tau}_h, t_h) \in \tilde{V}_h \times \tilde{W}_h$, with (cf. Figure 3.1)

$$\tilde{V}_h = \mathrm{span}(\boldsymbol{\epsilon}_h^n, \boldsymbol{\epsilon}_h^e, \boldsymbol{\epsilon}_h^s, \boldsymbol{\epsilon}_h^w),$$

$$\tilde{W}_h = \mathrm{span}(e_h^c).$$

It follows from the proof of Theorem 3.1 that $F_h^V(\boldsymbol{\sigma}_h + \boldsymbol{\tau}_h, u_h + t_h)$ has a unique minimum in Λ_h^V for the unique solution of (3.13).

The fact that this local minimization procedure has a unique solution can be used to prove that Vanka-type relaxation applied to the lumped equations, is convergent.

THEOREM 3.5. *Suppose that $a(\cdot,\cdot)$ is L^2-coercive, f is as in (3.2), and problem (3.11) with lumping has a unique solution $(\boldsymbol{\sigma}_h, u_h)$, then Vanka-type relaxation converges to $(\boldsymbol{\sigma}_h, u_h)$.*

PROOF. Suppose that the domain Ω is covered by N cells Ω^i. Let the operator $S^{V,n}$ denote the updating of cell Ω^i, $i = n \bmod N$ in the m^{th} relaxation sweep, $m = 1 + n \operatorname{div} N$, so

$$(\boldsymbol{\sigma}^{(n+1)}, u^{(n+1)}) = S^{V,n}\,(\boldsymbol{\sigma}^{(n)}, u^{(n)}).$$

The sequence $F_h^V(\boldsymbol{\sigma}^{(n)}, u^{(n)})$ is monotonically decreasing and bounded from below by $F_h^V(\boldsymbol{\sigma}_h, u_h)$, with $(\boldsymbol{\sigma}_h, u_h)$ the unique solution of (3.11). This implies that the sequence $F_h^V(\boldsymbol{\sigma}^{(n)}, u^{(n)})$ is convergent. Now suppose that the sequence $(\boldsymbol{\sigma}^{(n)}, u^{(n)})$ does not converge, then there is at least one Ω^i such that the sequence $(\boldsymbol{\sigma}^{(nN+i)}, u^{(nN+i)})$ does not converge. This contradicts the convergence of the sequence $F_h^V(\boldsymbol{\sigma}^{(n)}, u^{(n)})$ because the local minimization problem has a unique solution. Suppose that

$$\lim_{n \to \infty} (\boldsymbol{\sigma}^{(n)}, u^{(n)}) = (\boldsymbol{\sigma}^*, u^*),$$

then

$$\lim_{n\to\infty} (b(\boldsymbol{\sigma}^*, e_h^i) - (f, e_h^i)) = 0,$$

with $i = n \bmod N$; as we sweep over all cells Ω^i and $(\boldsymbol{\sigma}^*, u^*) \in \Lambda_h^V$ we conclude

$$(\boldsymbol{\sigma}^*, u^*) = (\boldsymbol{\sigma}_h, u_h). \quad \square$$

3.4. Superbox relaxation

In [4] Schmidt and Jacobs propose a relaxation for the linear source problem with homogeneous Neumann boundary conditions. During the relaxation we sweep over a number of relaxation subdomains, and minimize the functional F_h^S within those relaxation subdomains. Extending the approach of Schmidt and Jacobs to the case of Dirichlet boundary conditions, we define the following relaxations subdomains: superboxes that consist of four cells with a common vertex (cf. Figure 3.2), and single cells with at least one edge E^j part of the Dirichlet boundary, i.e. $E^j \subset \delta\Omega_D$.
In the first case F_h^S is minimized over the degrees of freedom inside the four cells, i.e. u^i, $i = ne, se, sw, nw$ and σ^j, $j = n, e, s, w$, while the fluxes σ^j at the external interfaces E^j, $j = nne, \cdots, nnw$, (cf. Figure 3.2) are kept fixed; thus ensuring that after relaxation the conservation law (3.11b) holds in all cells outside the superbox, provided that the conservation law holds there prior to relaxation. In the case of a single cell Ω^i we minimize the functional F_h^S over u^i and σ^j, with $E^j \subset \delta\Omega_D$. The fluxes σ^k at the edges E^k not part of the Dirichlet boundary are kept fixed.

We proceed by showing that these local minimization problems are uniquely solvable. From the proof of Theorem 3.2 we see that minimization of F_h^S with respect to the variables inside the superbox is equivalent to solving the system of equations (cf. Figure 3.2)

$$a^{n,n}\sigma^n - h_x{}^{-1}(u^{ne} - u^{nw}) = -a^{n,wnw}\sigma^{wnw} - a^{n,ene}\sigma^{ene}, \tag{3.18a}$$
$$a^{e,e}\sigma^e - h_y{}^{-1}(u^{ne} - u^{se}) = -a^{e,nne}\sigma^{nne} - a^{e,sse}\sigma^{sse}, \tag{3.18b}$$
$$a^{s,s}\sigma^s - h_x{}^{-1}(u^{se} - u^{sw}) = -a^{s,ese}\sigma^{ese} - a^{s,wsw}\sigma^{wsw}, \tag{3.18c}$$
$$a^{w,w}\sigma^w - h_y{}^{-1}(u^{nw} - u^{sw}) = -a^{w,nnw}\sigma^{nnw} - a^{w,ssw}\sigma^{ssw}, \tag{3.18d}$$
$$-h_x\sigma^e - h_y\sigma^n - h_xh_y\mathfrak{F}(u^{ne}) = -h_x\sigma^{nne} - h_y\sigma^{ene} - F^{ne}, \tag{3.18e}$$
$$+h_x\sigma^e - h_y\sigma^s - h_xh_y\mathfrak{F}(u^{se}) = +h_x\sigma^{sse} - h_y\sigma^{ese} - F^{se}, \tag{3.18f}$$
$$+h_x\sigma^w + h_y\sigma^s - h_xh_y\mathfrak{F}(u^{sw}) = +h_x\sigma^{ssw} + h_y\sigma^{wsw} - F^{sw}, \tag{3.18g}$$
$$-h_x\sigma^w + h_y\sigma^n - h_xh_y\mathfrak{F}(u^{nw}) = -h_x\sigma^{nnw} + h_y\sigma^{wnw} - F^{nw}, \tag{3.18h}$$

with $a^{j,k}$ and F^i as in (3.14) and (3.15), respectively. By adding the equations 3.18e to 3.18h, we observe that this system of equations is singular in the -so far excluded- case $\mathfrak{F}=0$. When $\mathfrak{F}=0$ and the sum of the right hand sides of 3.13e to 3.13h vanishes, the linear system (3.18) has a one-dimensional manifold of solutions. Schmidt and Jacobs treat this special case by replacing one of the equations 3.18e-3.18h by the requirement that the average of the u^i should not change during relaxation of the superbox, i.e. they require

$$u^{ne} + u^{se} + u^{sw} + u^{nw} = \text{constant}.$$

In Section 3.5 we study this treatment of the case $\mathcal{F}=0$ by means of local mode Fourier analysis. We remark that from the numerical point of view the system (3.18) always becomes singular in the limit case of vanishing mesh size.

The fact that the solution of (3.18) minimizes a functional is now used to prove that problem (3.18) has a unique solution.

LEMMA 3.1. *Let the bilinear form $a: V_h \times V_h \to \mathbb{R}$ be L^2-coercive, and let $\mathcal{F}$ be as in Theorem 3.2, then problem (3.18) has a unique solution.*

PROOF. First we prove uniqueness of the solution of (3.18) and then existence. Suppose that (3.18) has two solutions (σ^j, u^i) and $(\hat{\sigma}^j, \hat{u}^i)$. After elimination of σ^j and $\hat{\sigma}^j$ and by using the mean value theorem we can write (3.18e-h) as a system of linear equations for the differences $u^i - \hat{u}^i$:

$$\begin{bmatrix} c^n + c^e + \mathcal{F}'_{ne} & -c^e & 0 & -c^n \\ -c^e & c^e + c^s + \mathcal{F}'_{se} & -c^s & 0 \\ 0 & -c^s & c^s + c^w + \mathcal{F}'_{sw} & -c^w \\ -c^n & 0 & -c^w & c^w + c^n + \mathcal{F}'_{nw} \end{bmatrix} \begin{bmatrix} u^{ne} - \hat{u}^{ne} \\ u^{se} - \hat{u}^{se} \\ u^{sw} - \hat{u}^{sw} \\ u^{nw} - \hat{u}^{nw} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix},$$

with

$$c^n = \frac{1}{a^{n,n}} \frac{h_y}{h_x}, \quad \text{etc.,} \tag{3.19}$$

and

$$\mathcal{F}'_{ne} = \mathcal{F}'(\tilde{u}), \quad \text{with } \min(u^{ne}, \hat{u}^{ne}) \leqslant \tilde{u} \leqslant \max(u^{ne}, \hat{u}^{ne}), \text{ etc..}$$

From (3.2b) it follows that the matrix is an M-matrix, so $u^i - \hat{u}^i = 0$, and therefore any solution of (3.18) is unique.

To prove that a solution of (3.18) exists we first consider the problem of minimizing F_h^S in a relaxation subdomain that consists of two cells with a common edge. Suppose that we have some iterate $(\hat{\boldsymbol{\sigma}}_h, \hat{u}_h) \in \Lambda_h^S$. If we take, for example, the relaxation subdomain that consists of Ω^{ne} and Ω^{nw}, minimization of $F_h^S(\hat{\boldsymbol{\sigma}}_h + \boldsymbol{\tau}_h, \hat{u}_h + t_h)$ in Λ_h^S for $(\boldsymbol{\tau}_h, t_h) \in \hat{V}_h \times \hat{W}_h$, with (cf. Figure 3.2)

$$\hat{V}_h = \text{span}(\boldsymbol{\epsilon}_h^n),$$

$$\hat{W}_h = \text{span}(e_h^{ne}, e_h^{nw}),$$

is equivalent to solving the eqations (3.18a), (3.18e) and (3.18h) for σ^n, u^{ne} and u^{nw}. After elimination of σ^n we obtain

$$c^n(u^{ne} - u^{nw}) + h_x h_y \mathcal{F}(u^{ne}) = r^{ne}, \tag{3.20a}$$

$$c^n(u^{nw} - u^{ne}) + h_x h_y \mathcal{F}(u^{nw}) = r^{nw}, \tag{3.20b}$$

with c^n as in (3.19) and r^{ne}, r^{nw} the appropriate right hand sides. After elimination of u^{nw} from (3.20a) we obtain one nonlinear equation for u^{ne} that can be written as

$$\tilde{f}(u^{ne}) = c_1\mathcal{F}(u^{ne}) + c_2\mathcal{F}(u^{ne} + c_3\mathcal{F}(u^{ne}) + c_4) = \tilde{c}, \tag{3.21}$$

with $c_1, c_2, c_3 > 0$. As $\mathcal{F}$ is continuous and strictly monotone, we find that $\tilde{f}$ is continuous and strictly monotone. Moreover, because $\mathcal{F}$: $\mathbb{R} \to \mathbb{R}$ is invertible we have

$$\lim_{u\to+\infty} \mathcal{F}(u) = +\infty,$$

$$\lim_{u\to-\infty} \mathcal{F}(u) = -\infty,$$

and therefore

$$\lim_{u\to+\infty} \tilde{f}(u) = +\infty,$$

$$\lim_{u\to-\infty} \tilde{f}(u) = -\infty.$$

So (3.21) has a unique solution u^{ne} for any $\tilde{c} \in \mathbb{R}$. Therefore the minimization problem for the two adjacent cells has a unique solution.

Now we consider the following relaxation procedure for the solution of (3.18): sweep over the edges E^j, $j = n,e,s,w$, and minimize F_h^S in the relaxation subdomain that consists of both cells adjacent to E^j. As just shown this minimization problem has a unique solution, and therefore the sequence $F_h^S(\boldsymbol{\sigma}^{(n)}, u^{(n)})$, with $(\boldsymbol{\sigma}^{(n)}, u^{(n)})$ the iterate after n relaxation steps, is monotonically decreasing. From (3.6b) we obtain

$$R^S(u) = R^S(0) + \int_0^u \frac{dR^S(\tilde{u})}{d\tilde{u}}\, d\tilde{u} = R^S(0) + \int_0^u \tilde{u}\mathcal{F}'(\tilde{u})\, d\tilde{u} \geqslant R^S(0),$$

and because

$$\langle g, \boldsymbol{\epsilon}_h^j \rangle_{\delta\Omega_D} = 0, \qquad \text{for } j = n,e,s,w,$$

we conclude that for any iterate $(\hat{\boldsymbol{\sigma}}_h, \underset{\sim}{\hat{u}_h}) \in \underset{\sim}{\Lambda}_h^S$ the functional $F_h^S(\hat{\boldsymbol{\sigma}}_h + \boldsymbol{\tau}_h, \hat{u}_h + t_h)$ is bounded from below for $(\boldsymbol{\tau}_h, t_h) \in V_h \times \underset{\sim}{W}_h$, with (cf. Figure 3.2)

$$\tilde{V}_h = \operatorname{span}(\boldsymbol{\epsilon}_h^n, \boldsymbol{\epsilon}_h^e, \boldsymbol{\epsilon}_h^s, \boldsymbol{\epsilon}_h^w),$$

$$\tilde{W}_h = \operatorname{span}(e_h^{ne}, e_h^{se}, e_h^{sw}, e_h^{nw}).$$

The sequence $F_h^S(\boldsymbol{\sigma}^{(n)}, u^{(n)})$ is monotonically decreasing and bounded from below, so it converges. From the fact that the local minimization problem is uniquely solvable, and that all edges E^j, $j = n,e,s,w$ are visited during the relaxation, we conclude that $(\boldsymbol{\sigma}^{(n)}, u^{(n)})$ converges to the unique solution of (3.18). □

After having proved that the minimization problem for the superbox is uniquely solvable, we still need to prove that the minimization problem for single cells with at least one edge part of the Dirichlet boundary is uniquely solvable. In this case, after elimination of the fluxes, we obtain a nonlinear equation of the form(3.17), so it has a unique solution. Now we are able to prove that superbox relaxation applied to (3.11) converges.

THEOREM 3.6. *Suppose that the bilinear form* $a: V_h \times V_h \rightarrow \mathbb{R}$ *is* L^2*-coercive,* $\mathfrak{F}$ *is as in Theorem 3.2, and problem (3.11) has a unique solution* $(\boldsymbol{\sigma}_h, u_h)$*, then the superbox relaxation converges to* $(\boldsymbol{\sigma}_h, u_h)$*.*

PROOF. This proof is completely analogous to the proof of Theorem 3.5. □

3.5. LOCAL MODE ANALYSIS

In this Section we study the feasibility of Vanka-type relaxation and superbox relaxation as smoothers in a multigrid algorithm. For the sake of completeness we consider both the usual five-point Vanka-type relaxation and the three-point Vanka-type relaxation. As a model problem we take the two-dimensional linear source problem (cf. Example 3.1), with $f = su + F(\mathbf{x})$ and s a non-negative constant, which includes the case of Poisson's equation ($s = 0$). This problem is discretized on a uniform square grid by the mixed finite element discretization as described in Chapter 2. In all relaxation methods that we consider in this Section we assume that the cells are ordered row-wise (from left to right), with the rows being ordered from bottom to top. We start our discussion with the three-point Vanka-type relaxation, because it is the most simple to analyze.

In a three-point Vanka sweep all variables are updated only once, so starting from some approximation $\{\sigma_x^j, \sigma_y^k, u^i\}$ we obtain a new approximation $\{\overline{\sigma}_x^j, \overline{\sigma}_y^k, \overline{u}^i\}$. The relation between the corresponding error quantities is for convenience also denoted by σ_x^j, σ_y^k and u^i. When the cell Ω^c is visited, we solve the following algebraic system for $\overline{\sigma}_x^e$, $\overline{\sigma}_y^n$ and $\overline{u}^c$ (cf. (3.13) and Figure 3.1):

$$\text{n:} \quad \kappa\sigma_y^N + (1-2\kappa)\overline{\sigma}_y^n + \kappa\overline{\sigma}_y^s + \frac{1}{h}(\overline{u}^c - u^n) = 0, \tag{3.22a}$$

$$\text{e:} \quad \kappa\sigma_x^E + (1-2\kappa)\overline{\sigma}_x^e + \kappa\overline{\sigma}_x^w + \frac{1}{h}(\overline{u}^c - u^e) = 0, \tag{3.22b}$$

$$\text{c:} \quad \frac{1}{h}(\overline{\sigma}_y^n + \overline{\sigma}_x^e - \overline{\sigma}_y^s - \overline{\sigma}_x^w) - s\overline{u}^c = 0, \tag{3.22c}$$

where $\kappa = 0$ denotes the lumped case (cf. (2.30)) and $\kappa = \frac{1}{6}$ the case that the discrete equations are not lumped (cf. (2.22a)). Starting with a Fourier error mode $(\sigma_x, \sigma_y, u) = (a, b, c)e^{i\boldsymbol{\omega}\cdot\mathbf{x}}$ we obtain

$$\text{n:}\qquad \kappa e^{i\frac{3\theta_y}{2}} b + (1-2\kappa) e^{i\frac{\theta_y}{2}} \overline{b} + \kappa e^{-i\frac{\theta_y}{2}} \overline{b} - \frac{1}{h}(e^{i\theta_y} c - \overline{c}) = 0, \tag{3.23a}$$

$$\text{e:}\qquad \kappa e^{i\frac{3\theta_x}{2}} a + (1-2\kappa) e^{i\frac{\theta_x}{2}} \overline{a} + \kappa e^{-i\frac{\theta_x}{2}} \overline{a} - \frac{1}{h}(e^{i\theta_x} c - \overline{c}) = 0, \tag{3.23b}$$

$$\text{c:}\qquad \frac{1}{h}(e^{i\frac{\theta_x}{2}} - e^{-i\frac{\theta_x}{2}})\overline{a} + \frac{1}{h}(e^{i\frac{\theta_y}{2}} - e^{-i\frac{\theta_y}{2}})\overline{b} - s\overline{c} = 0, \tag{3.23c}$$

with $(\theta_x, \theta_y) = (h\omega_x, h\omega_y)$. Thus we obtain a relation between the error components before and after relaxation

$$\begin{bmatrix} \overline{a} \\ \overline{b} \\ \overline{c} \end{bmatrix} = \hat{S}_h^{V3} \begin{bmatrix} a \\ b \\ c \end{bmatrix}. \tag{3.24}$$

A good smoother for a multigrid algorithm should reduce the high frequency error components efficiently, therefore -as usual- we define the smoothing factor μ by (cf. [5])

$$\mu^{V3} = \sup_{T_h/T_H} \rho(\hat{S}_h^{V3}(\boldsymbol{\theta})), \tag{3.25}$$

with $T_H = (-\frac{\pi}{2}, \frac{\pi}{2}]^2$, $T_h = (-\pi, \pi]^2$, and $\rho(\cdot)$ the spectral radius. The first rows of the Tables 3.1 and 3.2 show the numerically evaluated smoothing factors μ^{V3} of three-point Vanka-type relaxation for the lumped ($\kappa=0$) and the non-lumped case ($\kappa=1/6$), respectively. We observe that three-point Vanka-type relaxation is not a good smoother, in fact it diverges for the non-lumped case.

	10^3	10^2	10^1	10^0	10^{-1}	10^{-2}	10^{-3}	0
Vanka-3	0.004	0.038	0.286	0.800	0.976	0.997	1.000	1.000
Vanka-5	0.002	0.019	0.143	0.400	0.488	0.499	0.500	0.500
superbox	0.001	0.010	0.083	0.316	0.470	0.497	0.499	0.500

TABLE 3.1. Smoothing factor μ depending on the source term sh^2 (with lumping).

	10^3	10^2	10^1	10^0	10^{-1}	10^{-2}	10^{-3}	0
Vanka-3	0.333	0.333	0.286	1.475	2.149	2.546	2.266	2.267
Vanka-5	0.091	0.085	0.276	0.561	0.622	0.629	0.629	0.629
superbox	0.329	0.301	0.211	0.298	0.430	0.445	0.447	0.447

TABLE 3.2. Smoothing factor μ depending on the source term sh^2 (no lumping).

Next we consider five-point Vanka-type relaxation. In a single relaxation sweep the fluxes $\{\sigma_x^j, \sigma_y^k\}$ are updated twice; so we obtain new values $\{\overline{\sigma}_x^j, \overline{\sigma}_y^k, \overline{u}^i\}$ by using intermediate values for $\{\sigma_x^j, \sigma_y^k\}$. The relation between the error quantities in a cell Ω^c is now (cf. (3.13) and Figure 3.1)

$$\text{n:} \quad \kappa e^{i\frac{3\theta_y}{2}} b + (1-2\kappa) e^{i\frac{\theta_y}{2}} b_1 + \kappa e^{-i\frac{\theta_y}{2}} \overline{b} - \frac{1}{h}(e^{i\theta_y} c - \overline{c}) = 0, \qquad (3.26a)$$

$$\text{e:} \quad \kappa e^{i\frac{3\theta_x}{2}} a + (1-2\kappa) e^{i\frac{\theta_x}{2}} a_1 + \kappa e^{-i\frac{\theta_x}{2}} \overline{a} - \frac{1}{h}(e^{i\theta_x} c - \overline{c}) = 0, \qquad (3.26b)$$

$$\text{s:} \quad \kappa e^{i\frac{\theta_y}{2}} b_1 + (1-2\kappa) e^{-i\frac{\theta_y}{2}} \overline{b} + \kappa e^{-i\frac{3\theta_y}{2}} \overline{b} - \frac{1}{h}(\overline{c} - e^{-i\theta_y} \overline{c}) = 0, \qquad (3.26c)$$

$$\text{w:} \quad \kappa e^{i\frac{\theta_x}{2}} a_1 + (1-2\kappa) e^{-i\frac{\theta_x}{2}} \overline{a} + \kappa e^{-i\frac{3\theta_x}{2}} \overline{a} - \frac{1}{h}(\overline{c} - e^{-i\theta_x} \overline{c}) = 0, \qquad (3.26d)$$

$$\text{c:} \quad \frac{1}{h}(e^{i\frac{\theta_x}{2}} a_1 - e^{-i\frac{\theta_x}{2}} \overline{a}) + \frac{1}{h}(e^{i\frac{\theta_y}{2}} b_1 - e^{-i\frac{\theta_y}{2}} \overline{b}) - s\overline{c} = 0, \qquad (3.26e)$$

with a_1 and b_1 the intermediate values of σ_x^j and σ_y^k, respectively. After elimination of these intermediate values we obtain a relation between the error components before and after relaxation as in (3.24). The second rows of the Tables 3.1 and 3.2 show numerical values for the smoothing factor μ^{V5} of five-point Vanka-type relaxation. We observe that the five-point Vanka-type relaxation is a good smoother not only for the lumped case, but also for the non-lumped case.

Finally we consider the superbox relaxation. Here the fluxes σ_x^j and σ_y^k are updated twice in a single sweep, whereas the u^i are updated four times. Figure 3.3 shows how many times the variables associated to the cells and edges have been updated after relaxation of the shaded superbox. In this case the relation between the error quantities before and after relaxation is (cf. (3.18) and the Figures 3.2 and 3.3)

$$\text{n:} \quad \kappa e^{i\theta_x} a + (1-2\kappa) a_1 + \kappa e^{-i\theta_x} a_1 - \frac{1}{h}(e^{i\frac{\theta_x}{2}} c_1 - e^{-i\frac{\theta_x}{2}} c_2) = 0, \qquad (3.27a)$$

$$\text{e:} \quad \kappa e^{i\theta_y} b + (1-2\kappa) b_1 + \kappa e^{-i\theta_y} \overline{b} - \frac{1}{h}(e^{i\frac{\theta_y}{2}} c_1 - e^{-i\frac{\theta_y}{2}} c_3) = 0, \qquad (3.27b)$$

$$\text{s:} \quad \kappa e^{i\theta_x} a_1 + (1-2\kappa) \overline{a} + \kappa e^{-i\theta_x} \overline{a} - \frac{1}{h}(e^{i\frac{\theta_x}{2}} c_3 - e^{-i\frac{\theta_x}{2}} \overline{c}) = 0, \qquad (3.27c)$$

$$\text{w:} \quad \kappa e^{i\theta_y} b + (1-2\kappa) \overline{b} + \kappa e^{-i\theta_y} \overline{b} - \frac{1}{h}(e^{i\frac{\theta_y}{2}} c_2 - e^{-i\frac{\theta_y}{2}} \overline{c}) = 0, \qquad (3.27d)$$

$$\text{ne:} \quad \frac{1}{h}(e^{i\frac{\theta_x}{2}} a - e^{-i\frac{\theta_x}{2}} a_1) + \frac{1}{h}(e^{i\frac{\theta_y}{2}} b - e^{-i\frac{\theta_y}{2}} b_1) - sc_1 = 0, \qquad (3.27e)$$

$$\text{se:} \quad \frac{1}{h}(e^{i\frac{\theta_x}{2}} a_1 - e^{-i\frac{\theta_x}{2}} \overline{a}) + \frac{1}{h}(e^{i\frac{\theta_y}{2}} b_1 - e^{-i\frac{\theta_y}{2}} \overline{b}) - sc_3 = 0, \qquad (3.27f)$$

$$\text{sw:} \quad \frac{1}{h}(e^{i\frac{\theta_x}{2}} a - e^{-i\frac{\theta_x}{2}} \overline{a}) + \frac{1}{h}(e^{i\frac{\theta_y}{2}} \overline{b} - e^{-i\frac{\theta_y}{2}} \overline{b}) - s\overline{c} = 0, \qquad (3.27g)$$

$$\text{nw:} \quad \frac{1}{h}(e^{i\frac{\theta_x}{2}} a_1 - e^{-i\frac{\theta_x}{2}} a_1) + \frac{1}{h}(e^{i\frac{\theta_y}{2}} b - e^{-i\frac{\theta_y}{2}} \overline{b}) - sc_2 = 0, \qquad (3.27h)$$

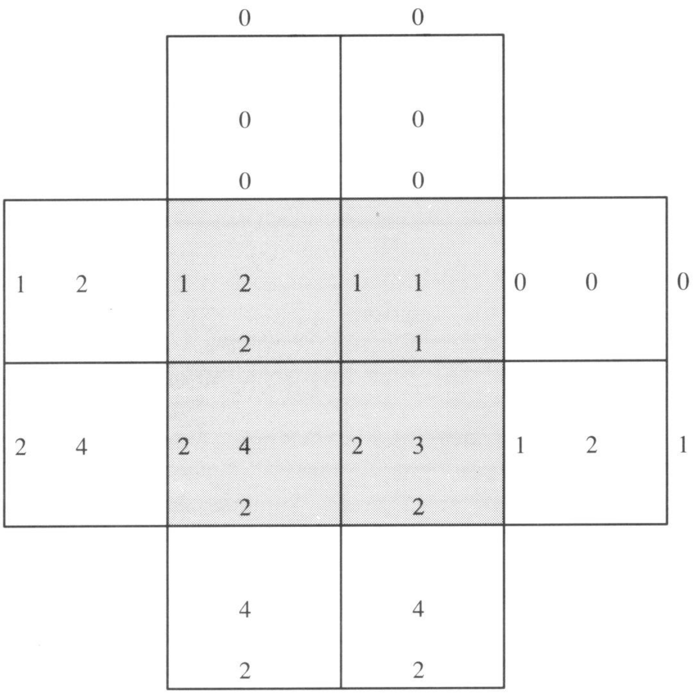


FIGURE 3.3. Number of updates of variables in superbox relaxation.

	$\kappa = 0$	$\kappa = \frac{1}{6}$
ne	0.500	0.447
se	0.500	0.447
sw	1.000	1.000
nw	0.500	0.447

TABLE 3.3. Smoothing factor μ^S of superbox relaxation for Poisson's equation.

with a_1, b_1, c_1, c_2 and c_3 the intermediate values. For the Poisson equation $(s=0)$ the system is linearly dependent, and we replace one of the equations (3.27e-h) by (cf. Section 3.4)

$$\begin{aligned}
&ce^{i(\frac{\theta_x}{2}+\frac{\theta_y}{2})}+c_1e^{i(-\frac{\theta_x}{2}+\frac{\theta_y}{2})}+c_2e^{i(\frac{\theta_x}{2}-\frac{\theta_y}{2})}+c_3e^{i(-\frac{\theta_x}{2}-\frac{\theta_y}{2})}= \\
&\qquad c_1e^{i(\frac{\theta_x}{2}+\frac{\theta_y}{2})}+c_2e^{i(-\frac{\theta_x}{2}+\frac{\theta_y}{2})}+c_3e^{i(\frac{\theta_x}{2}-\frac{\theta_y}{2})}+\bar{c}e^{i(-\frac{\theta_x}{2}-\frac{\theta_y}{2})}. \qquad (3.28)
\end{aligned}$$

The last rows of the Tables 3.1 and 3.2 show numerical values for the smoothing factor μ^S of superbox relaxation. We observe that the superbox relaxation is a good smoother just like five-point Vanka-type relaxation . We remark that the smoothing factor μ^S of superbox relaxation in the case of Poisson's equation depends on which of the equations (3.27e-h) is replaced by (3.28). This is shown in Table 3.3. If the equation for the se-cell is replaced, then superbox relaxation does not remove some high frequencies at all, and is thus useless as a smoother.

Although the smoothing properties of five-point Vanka-type relaxation and superbox relaxation are comparable we remark that the superbox relaxation is more expensive in arithmetic operations. Suppose that a grid consists of N cells and that in both relaxation methods the fluxes are eliminated analytically. In the case of Vanka-type relaxation we have to solve N equations in a single sweep, whereas in superbox relaxation we have to solve approximately N systems of 4 equations with 4 unknowns. This means that superbox relaxation requires at least 4 times as much work as Vanka-type relaxation.

3.6. Concluding remarks

We have shown that the variational formulation of our mixed finite element discretization can be considered in two ways as a constrained optimization problem. For these optimization problems we have shown that superbox relaxation and five-point Vanka-type relaxation minimize appropriate functionals, so their convergence can be proved for a class of nonlinear problems. By local mode analysis we have studied the feasibility of the two relaxation methods as smoothers for multigrid algorithms. It turns out that their smoothing power is comparable, but that Vanka-type relaxation is much more efficient.

References

1. L. Collatz and W. Wetterling (1975). *Optimization Problems*, Springer-Verlag, New York.
2. V. Girault and P. Raviart (1986). *Finite Element Methods for Navier-Stokes Equations*, Springer-Verlag, Berlin.
3. S.J. Polak, C. den Heijer, W.H.A. Schilders, and P. Markowich (1987). Semiconductor device modelling from the numerical point of view, *Int.J.Num.Meth.Engng.*, 24, 763-838.
4. G.H. Schmidt and F.J. Jacobs (1988). Adaptive Local Grid Refinement and Multi-grid in Numerical Reservoir Simulation, *J.Comput.Phys.*, 77,

140-165.

5. K. Stüben and U. Trottenberg (1982). Multigrid methods: fundamental algorithms, model problem analysis and applications, in *Multigrid Methods*, 220-312, ed. W. Hackbusch and U. Trottenberg, Springer-Verlag, Lecture Notes in Mathematics 960.
6. S.P. Vanka (1986). Block-Implicit Multigrid Solution of Navier-Stokes Equations in Primitive Variables, *J.Comput.Phys.*, 65, 138-158.

Chapter 4

Two-grid analysis

4.1. Introduction

In this Chapter we carry out a two-grid analysis of the combination of mixed finite elements and Vanka-type relaxation. Therefore we consider the Poisson equation

$$\begin{aligned} \operatorname{div}(\operatorname{grad} u) &= F, \qquad \text{on } \Omega, \\ u &= 0, \qquad \text{on } \delta\Omega, \end{aligned} \tag{4.1}$$

with $\Omega \subset \mathbb{R}^n$, $n = 1,2$. Problem (4.1) is discretized by the mixed finite element method based on lowest order Raviart-Thomas elements (see Chapter 2). To solve the linear system of equations resulting from the discretization we use a two-grid algorithm. For an introduction to two-grid algorithms the reader is referred to [1, 3]. The coarse grid Ω_H in the two-grid algorithm is constructed by cell-wise coarsening. The spaces spanned by the Raviart-Thomas elements defined on these two grids, are nested, i.e. $V_H \subset V_h$ and $W_H \subset W_h$, with V_h and W_h as in (2.19). Therefore canonical grid transfer operators are available; for the variable u_h this is a piecewise constant interpolation. As a smoothing procedure in the two-grid algorithm we use the Vanka-type relaxation presented in Chapter 3. When the discrete equations are lumped (cf. Chapter 2), Vanka-type relaxation is equivalent to point Gauss-Seidel applied to the system of equations from which the σ_h have been eliminated, i.e. a second order difference equation. However, it is well known that the piecewise constant interpolation is too inaccurate to be used as a grid transfer operator in multigrid algorithms for second order difference equations. This brings us to the central question of this Chapter: is it advantageous to combine Vanka-type relaxation and the canonical grid transfer operators?

To answer this question the two-grid analysis is carried out. In fact we do not only consider the case that the discrete equations are lumped, but also the case that they are not. In Section 4.2 we briefly review the mixed finite element discretization for the 1D problem and we introduce a two-grid algorithm for the iterative solution of the system of linear equations. The Fourier representations of the different operators in the coarse grid correction are derived in Section 4.3. It is shown that the canonical grid transfer operators are accurate enough to be used in the two-grid algorithm, if we do not take relaxation into account. In Section 4.4 we analyze Vanka-type relaxation for the 1D problem by means of a local mode Fourier analysis. In our analysis we include the use of a relaxation parameter as well as different orderings of the grid points: lexicographical and red-black. Next we include the relaxation operator in the two-grid algorithm (Section 4.5). As expected the two-grid algorithm does not converge properly when the Vanka-type relaxation with a lexicographical ordering of the grid points is used; however when a red-black ordering is used there are no convergence problems, independently of the fact that lumping is used or not. In Section 4.6 we extend our discussion to the two-dimensional case; for simplicity we only consider the case that the discrete equations are lumped. Here we do not find any convergence problems irrespectively of which ordering of the grid points is used in Vanka-type relaxation. Our conclusions are summarized in Section 4.7.

4.2. Preliminaries

To discretize the problem (4.1) on the unit interval $\Omega = [0, 1] \subset \mathbb{R}$, we decompose the domain Ω into a set of N uniform cells Ω_h^i of size $h = N^{-1}$ (cf. Section 2.3), N is even. The mixed finite element discretization by the lowest order Raviart-Thomas elements on this grid yields the linear system

$$\begin{bmatrix} \mathbf{a}_h & \mathbf{b}_h \\ \mathbf{b}_h^{\mathrm{T}} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \underline{\sigma_h} \\ \underline{u_h} \end{bmatrix} = \begin{bmatrix} 0 \\ \underline{F_h} \end{bmatrix}, \tag{4.2}$$

with

$$(\mathbf{b}_h)_{ji} = \begin{cases} -\dfrac{1}{h}, & j = i-1, \\ +\dfrac{1}{h}, & j = i, \\ 0, & \text{otherwise}, \end{cases} \tag{4.3}$$

and

$$(\mathbf{a}_h)_{jk} = \begin{cases} \kappa, & |j-k| = 1, \\ 1-2\kappa, & j=k, \;\; j = 1, \cdots, N-1, \\ \dfrac{1}{2} - \kappa, & j = k, \;\; j \in \{0, N\}, \\ 0, & \text{otherwise}, \end{cases} \tag{4.4}$$

which implies both the lumped case ($\kappa = 0$) as well as exact integration ($\kappa = \frac{1}{6}$). The discrete fine grid operator L_h: $\mathbb{R}^{2N+1} \rightarrow \mathbb{R}^{2N+1}$ is now defined by the system (4.2)

$$L_h = \begin{bmatrix} \mathbf{a}_h & \mathbf{b}_h \\ \mathbf{b}_h^{\mathrm{T}} & \mathbf{0} \end{bmatrix}. \tag{4.5}$$

The coarse grid is obtained by cell-wise coarsening, i.e. by taking $H = 2h$ instead of h. As noted before the approximating subspaces are nested, hence the canonical grid transfer operators are available. The canonical prolongation P_h:$\mathbb{R}^{N+1} \rightarrow \mathbb{R}^{2N+1}$ is defined on the space of coefficient vectors $(\underline{\sigma_H}, \underline{u_H})^{\mathrm{T}}$; it is a piecewise constant interpolation for $\underline{u_h}$ and a piecewise linear interpolation for σ_h. The canonical restriction $\overline{R}_H:\mathbb{R}^{2N+1}\rightarrow\mathbb{R}^{N+1}$ is the adjoint of P_h. The coarse grid operator is obtained by using the same discretization on the coarse grid Ω_H as on the fine grid Ω_h. If exact quadrature is used ($\kappa=\frac{1}{6}$), we find that L_H is the Galerkin approximation of L_h:

$$L_H = \overline{R}_H L_h P_h. \tag{4.6}$$

As smoothing operator $S_h:\mathbb{R}^{2N+1}\rightarrow\mathbb{R}^{2N+1}$ we use the Vanka-type relaxation, that is also called Symmetric Block Gauss-Seidel relaxation in the sequel. In 1D this means that in every cell Ω_h^i the σ_h^i, σ_h^{i-1} and u_h^i are relaxed simultaneously. Here we consider both the lexicographical (SBGS) and the red-black (SBRB) ordering of the cells in Vanka-type relaxation.

Finally we define the two-grid error amplification matrix

$$M_h^{\nu_2,\nu_1} = S_h^{\nu_2}(I_h - P_h(L_H)^{-1}\overline{R}_H L_h)S_h^{\nu_1}, \tag{4.7}$$

where $I_h : \mathbb{R}^{2N+1}\rightarrow\mathbb{R}^{2N+1}$ denotes the identity operator and ν_1,ν_2 the number of pre- and post relaxation sweeps, respectively.

4.3. Coarse grid correction

In order to derive Fourier representations of the different operators in the two-grid algorithm, we extend the domain to $\Omega=\mathbb{R}$ and omit the boundary conditions. The coefficient vectors $\underline{\sigma}_h$ and $\underline{u}_h$ are considered as grid functions defined on different discretization grids

$$\mathbb{Z}_{h,s} = \{(j-s)h, \text{ for } j\in\mathbb{Z}\}, \tag{4.8}$$

with

$$s = \begin{cases} 0, & \text{for } \underline{\sigma}_h, \\ \frac{1}{2}, & \text{for } \underline{u}_h. \end{cases}$$

The space of discrete L^2-functions on $\mathbb{Z}_{h,s}$, denoted by

$$L^{h,s}(\mathbb{Z}_{h,s}) = \{f_{h,s} \mid f_{h,s}\colon \mathbb{Z}_{h,s}\rightarrow\mathbb{C}\ ;\ h\sum_j |f_{h,s}((j-s)h)|^2<\infty\},$$

is a Hilbert space. The Fourier transform $FT(f_{h,s})=\hat{f}_{h,s}\colon T_h\rightarrow\mathbb{C}$ of a $L^{h,s}$-function is defined by

$$\hat{f}_{h,s}(\omega) = \sum_{j\in\mathbb{Z}} e^{-i(j-s)h\omega} f_{h,s}((j-s)h), \tag{4.9}$$

with $T_h=(-\frac{\pi}{h},\frac{\pi}{h}]$. The inverse transformation is given by

$$f_{h,s}((j-s)h) = \frac{h}{2\pi}\int_{\omega\in T_h} e^{i(j-s)h\omega}\hat{f}_{h,s}(\omega)d\omega. \tag{4.10}$$

By Parseval's equality the Fourier transformation operator FT: $L^{h,s}\rightarrow L^2(T_h)$ is a unitary operator.

Convolution or Toeplitz operators $B_{h,s}$: $L^{h,s}\rightarrow L^{h,s}$ are linear operators, generated by a grid function $b_{h,0} \in L^{h,0}$:

$$B_h f_{h,s}((j-s)h) = \sum_{k \in \mathbb{Z}} b_{h,0}(kh) f_{h,s}((j-k-s)h).$$

The Fourier transform $FT(B_h) = \hat{B}_h$ of a Toeplitz operator B_h is defined by

$$\hat{B}_h(\omega) = \hat{b}_{h,0}(\omega). \tag{4.11}$$

For example, the matrices $\mathbf{a}_h$ and $\mathbf{b}_h$ in (4.2) are Toeplitz operators with Fourier transforms

$$\hat{a}_h(\omega) = 1 - 4\kappa \sin^2(\frac{h\omega}{2}) \tag{4.12}$$

and

$$\hat{b}_h(\omega) = \frac{2i}{h}\sin(\frac{h\omega}{2}). \tag{4.13}$$

In order to obtain Fourier representations of the grid transfer operators we introduce the elementary prolongation $P^0_{h,s}$: $L^{H,s} \rightarrow L^{h,s}$,

$$(P^0_{h,s} f_{H,s})((j-s)h) = \begin{cases} f_{H,s}((\frac{j}{2}-s)h), & j \text{ even}, \\ 0, & j \text{ odd}, \end{cases} \tag{4.14}$$

and the elementary restriction $R^0_{H,s}$: $L^{h,s} \rightarrow L^{H,s}$,

$$(R^0_{H,s} f_{h,s})((j-s)H) = f_{h,s}((2j-s)h). \tag{4.15}$$

Using (4.9), (4.10) and (4.15) we find that the Fourier transforms of $f_{h,s}$ and $R^0_{H,s} f_{h,s}$ are related by

$$\widehat{(R^0_{H,s} f_{h,s})}(\omega) = e^{ish\omega} \sum_{p=0,1} e^{-ips\pi} \hat{f}_{h,s}(\omega + p\frac{\pi}{h}), \tag{4.16}$$

with $\omega \in T_H \subset T_h$. Notice that $R^0_{H,s}$ aliases the two frequencies $\{\omega, \omega + \frac{\pi}{h}\} \in T_h$ with one frequency $\omega \in T_H$. The frequencies $\omega \in T_H$ are called the low frequencies in T_h and the $\omega \in T_h / T_H$ are called the high frequencies. Now every $\omega \in T_h$ can be written as a 2-vector $(\omega + p\frac{\pi}{h})$, $p \in \{0,1\}$ on T_H. Hence for $\hat{f}_{h,s} \in L^2(T_h)$ we may also use the notation $\mathbf{f}_{h,s}$, where $\mathbf{f}_{h,s}$ is a 2-vector with entries $\hat{f}_{h,s}(\omega + p\frac{\pi}{h})$. Consistent with this notation, we write the Fourier transform $\hat{B}_{h,s}(\omega)$ of a Toeplitz operator as a 2×2-diagonal matrix $\hat{\mathbf{B}}_{h,s}(\omega)$ with entries $\hat{B}_{h,s}(\omega + p\frac{\pi}{h})$, $p = 0,1$. Any restriction operator $R_{H,s}$, that is defined by a unique stencil, can be written as the combination of $R^0_{H,s}$ and a Toeplitz operator B_h. Hence the Fourier transform of $R_{H,s}$ is given by

$$\hat{\mathbf{R}}_{H,s}(\omega) = \hat{\mathbf{R}}^0_{H,s}(\omega)\,\hat{\mathbf{B}}_h(\omega), \tag{4.17}$$

where $\hat{\mathbf{R}}_{H,s}(\omega)$ and $\hat{\mathbf{R}}^0_{H,s}(\omega)$ are 1×2 matrices.

Analogous to the restriction we may write any prolongation $P_{h,s}$ as the combination of $P^0_{h,s}$ and a Toeplitz operator B_h. The Fourier transforms of $f_{H,s}$ and $P^0_{h,s}f_{H,s}$ are related by

$$(\widehat{P^0_{h,s}f_{H,s}})(\omega+p\frac{\pi}{h}) = \frac{1}{2}e^{-ish(\omega-p\frac{\pi}{h})}\hat{f}_{H,s}(\omega). \tag{4.18}$$

The Fourier transform of $P_{h,s}=B_hP^0_{h,s}$ is in matrix notation

$$\hat{\mathbf{P}}_{h,s}(\omega) = \hat{\mathbf{B}}_h(\omega)\hat{\mathbf{P}}^0_{h,s}(\omega), \tag{4.19}$$

where $\hat{\mathbf{P}}_{h,s}$ and $\hat{\mathbf{P}}^0_{h,s}$ are 2×1-matrices. For the canonical prolongation P_h (cf. Section 4.2) the grid functions that generate the Toeplitz operators are

$$b^u_{h,0} = \begin{bmatrix} 1 & 1 \end{bmatrix}$$

and

$$b^\sigma_{h,0} = \begin{bmatrix} \frac{1}{2} & 1 & \frac{1}{2} \end{bmatrix}.$$

Using (4.18) and (4.19) we compute the Fourier transform of the canonical prolongation P_h as

$$\hat{\mathbf{P}}_h(\omega) = \begin{pmatrix} \hat{\mathbf{P}}^\sigma_h(\omega) & 0 \\ 0 & \hat{\mathbf{P}}^u_h(\omega) \end{pmatrix} = \begin{pmatrix} \cos^2\dfrac{\theta}{2} & 0 \\ \sin^2\dfrac{\theta}{2} & 0 \\ 0 & \cos\dfrac{\theta}{2} \\ 0 & \sin\dfrac{\theta}{2} \end{pmatrix}, \tag{4.20}$$

with $\theta=h\omega$. The Fourier transform $\hat{\bar{\mathbf{R}}}_H(\omega)$ of the canonical restriction $\bar{R}_H$ is the transpose of $\hat{\mathbf{P}}_h(\omega)$.

The components of the Fourier transform of a grid transfer operator are trigonometric functions of θ; this allows us to classify them according to their behavior in the case ω fixed and $h\rightarrow0$. Suppose that the Fourier transforms of a prolongation and its adjoint restriction are given by $\hat{\mathbf{P}}_{h,s}(\omega)$ and $\hat{\mathbf{P}}^T_{h,s}(\omega)$, respectively. The low frequency order m_L of this grid transfer operator $P_{h,s}$ is the largest number $m_L \geqslant 0$ such that

$$\hat{P}_{h,s}(\omega) = 1+\mathcal{O}(\theta^{m_L}), \quad \text{for} \quad h\rightarrow0,\ \omega\in T_H.$$

The high frequency order m_H of $P_{h,s}$ is the largest number $m_H \geqslant 0$ for which

$$\hat{P}_{h,s}(\omega+\frac{\pi}{h}) = \mathcal{O}(\theta^{m_H}), \quad \text{for} \quad h\rightarrow0,\ \omega\in T_H.$$

For the canonical grid transfer operators we have: $m^\sigma_L=2$, $m^\sigma_H=2$, $m^u_L=2$ and $m^u_H=1$. To avoid large amplification of high frequency errors, the high frequency order m_H should be at least equal to the order of the difference equation (cf. [1, 2]). As we are considering a system of first order equations, we conclude that the canonical grid transfer operators are sufficiently accurate. In fact, the Fourier transform of the coarse grid correction operator $M_h^{0,0}$ is given by

$$\hat{\mathbf{M}}_h^{0,0}(\omega) = \begin{bmatrix} +\sin^2\frac{\theta}{2} & -\cos^2\frac{\theta}{2} & 0 & 0 \\ -\sin^2\frac{\theta}{2} & +\cos^2\frac{\theta}{2} & 0 & 0 \\ \frac{h}{2i\sin\frac{\theta}{2}}\gamma_L(\frac{\theta}{2}) & \frac{h}{2i\sin\frac{\theta}{2}}\gamma_H(\frac{\theta}{2}) & \sin^2\frac{\theta}{2} & -\frac{1}{2}\sin\theta \\ \frac{h}{2i\cos\frac{\theta}{2}}\gamma_L(\frac{\theta}{2}) & \frac{h}{2i\cos\frac{\theta}{2}}\gamma_H(\frac{\theta}{2}) & -\frac{1}{2}\sin\theta & \cos^2\frac{\theta}{2} \end{bmatrix}, \tag{4.21}$$

with $\gamma_L(\theta) = \sin^2\theta(1-12\kappa\cos^2\theta)$, $\gamma_H(\theta) = \cos^2\theta(1-12\kappa\sin^2\theta)$, and κ as in (4.4).

So, if it is assumed that $\inf_{\omega \in T_H} |\omega|$ is bounded away from zero by the boundary conditions, we see that all elements of $M_h^{0,0}(\omega)$ remain bounded for $h \to 0$. This implies that errors in (σ_h, u_h) are not blown up by the coarse grid correction if $h \to 0$.

4.4. Vanka-type relaxation

In this Section we derive the Fourier representation of the smoothing operators SBGS and SBRB, i.e. the Vanka-type relaxation with a lexicographical, red-black, ordering of the cells, respectively. We start by treating the lexicographical ordering. In a single SBGS-sweep the σ_h^j are updated twice; so starting from initial values $\{\sigma_h^j, u_h^i\}$ SBGS yields new values $\{\overline{\sigma}_h^j, \overline{u}_h^i\}$, using intermediate values $\tilde{\sigma}_h^j$. If the cells are visited from left to right, the error quantities (also denoted by σ_h^j and u_h^i) in a cell Ω_h^i are related by

$$\begin{aligned} \frac{1}{h}(\tilde{\sigma}_h^i - \overline{\sigma}_h^{i-1}) &= 0, \\ \kappa\sigma_h^{i+1} + (1-2\kappa)\tilde{\sigma}_h^i + \kappa\overline{\sigma}_h^{i-1} + \frac{1}{h}(u_h^{i+1} - \overline{u}_h^i) &= 0, \\ \kappa\tilde{\sigma}_h^i + (1-2\kappa)\overline{\sigma}_h^{i-1} + \kappa\overline{\sigma}_h^{i-2} + \frac{1}{h}(\overline{u}_h^i - \overline{u}_h^{i-1}) &= 0. \end{aligned} \tag{4.22}$$

Starting with a Fourier error mode $\sigma_h^j = ae^{ijh\omega}$ and $u_h^j = be^{i(j-\frac{1}{2})h\omega}$ we see that $\tilde{\sigma}_h^j = \tilde{a}e^{ijh\omega}$, $\overline{\sigma}_h^j = \overline{a}e^{ijh\omega}$ and $\overline{u}_h^j = \overline{b}e^{i(j-\frac{1}{2})h\omega}$. After elimination of $\tilde{a}$ from (4.22) we obtain a relation between the error components before and after relaxation,

$$\begin{bmatrix} \overline{a} \\ \overline{b} \end{bmatrix} = \hat{S}_h^{GS}(\omega) \begin{bmatrix} a \\ b \end{bmatrix}, \tag{4.23}$$

with

$$\hat{S}_h^{GS}(\omega) = \frac{e^{i\frac{\theta}{2}}}{2-2\kappa-(1-2\kappa)e^{-i\theta}} \begin{bmatrix} \kappa e^{i\frac{\theta}{2}}(1-e^{i\theta}) & \dfrac{1-e^{i\theta}}{h} \\ \kappa h(\kappa+(1-\kappa)e^{i\theta}) & e^{-i\frac{\theta}{2}}(\kappa+(1-\kappa)e^{i\theta}) \end{bmatrix}. \tag{4.24}$$

The spectral radius $\rho(\cdot)$ and the spectral norm $\|\cdot\|_S$ of $\hat{S}_h^{GS}(\omega)$ are

$$\rho(\hat{S}_h^{GS}(\omega)) = \left| \frac{1-2i\kappa\sin\theta}{2-2\kappa-(1-2\kappa)e^{-i\theta}} \right| \tag{4.25a}$$

and

$$\|\hat{S}_h^{GS}(\omega)\|_S = (1+\kappa^2h^2)\frac{\left[\dfrac{4}{h^2}\sin^2\dfrac{\theta}{2}+1-4\kappa\sin\dfrac{\theta}{2}+4\kappa^2\sin^2\dfrac{\theta}{2}\right]^{\frac{1}{2}}}{|2-2\kappa-(1-2\kappa)e^{-i\theta})|}, \tag{4.25b}$$

respectively. By using (4.25a) the smoothing factor μ^{GS},

$$\mu^{GS} = \sup_{\frac{\pi}{2}\leqslant|\theta|\leqslant\pi} \rho(\hat{S}_h^{GS}(\omega)),$$

is readily calculated:

$$\mu^{GS} = \begin{cases} (\frac{1}{5})^{\frac{1}{2}}, & \kappa=0, \\ (\frac{10}{29})^{\frac{1}{2}}, & \kappa=\frac{1}{6}, \end{cases} \tag{4.26}$$

independent of h.

From (4.25b) we see that $\|\hat{S}_h^{GS}(\omega)\|_S$ becomes unbounded for $h\rightarrow 0$. This is a consequence of the fact that, starting from an initial iterand $\underline{\sigma}_h=0$, we find errors in $\overline{\sigma}_h$ of magnitude $\mathcal{O}(h^{-1})$ for $h\rightarrow 0$ (cf. 4.24). From the boundedness of $\rho(\hat{S}_h^{GS}(\omega))$ we conclude that only in the first relaxation sweeps the errors in $\underline{\sigma}_h$ are blown up by SBGS. In order to measure what happens in the first sweep, we introduce a norm $\|\cdot\|_H$ that takes into account the difference by an order of h between the components of $\underline{\sigma}_h$ and of $\underline{u}_h$. It is defined by

$$\|\hat{A}_h(\omega)\|_H = \|\hat{H}_h\hat{A}_h(\omega)\|_S, \tag{4.27}$$

with H_h: $\mathbb{R}^{2N+1}\rightarrow\mathbb{R}^{2N+1}$ a scaling operator,

$$H_h\begin{bmatrix}\underline{\sigma}_h \\ \underline{u}_h\end{bmatrix} = \begin{bmatrix}h\underline{\sigma}_h \\ \underline{u}_h\end{bmatrix}.$$

From (4.27) it follows that the scaled norm $\|\cdot\|_H$ is not submultiplicative.

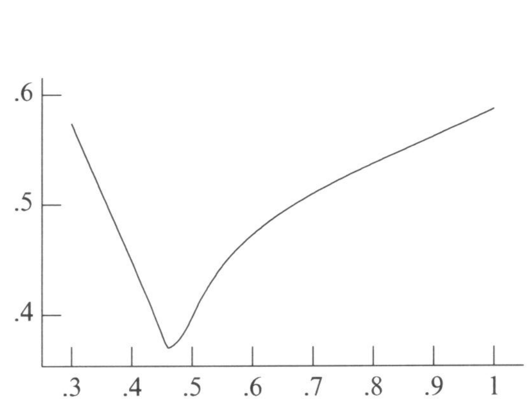


FIGURE 4.1. Smoothing factor μ^{GS} depending on the relaxation parameter α.

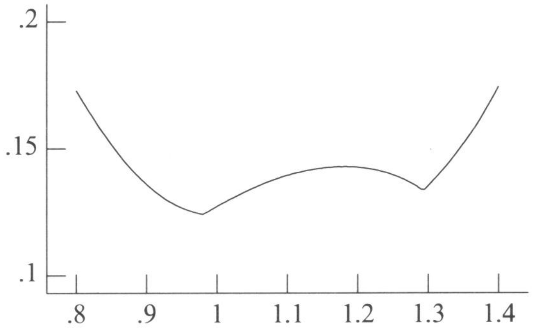


FIGURE 4.2. Smoothing factor μ^{RB} depending on the relaxation parameter α.

With respect to this norm we find for SBGS:

$$\|\hat{S}_h^{GS}(\omega)\|_H = (1+\kappa^2h^2)\frac{(4\sin^2\frac{\theta}{2}+1-4\kappa\sin\frac{\theta}{2}+4\kappa^2\sin^2\frac{\theta}{2})^{\frac{1}{2}}}{|2-2\kappa-(1-2\kappa)e^{-i\theta}|}, \tag{4.28}$$

which is indeed bounded for $h \rightarrow 0$.

Vanka proposes underrelaxation for σ_h^j (cf. [4]) to improve the smoothing properties of SBGS. This can be analyzed by replacing $\tilde{\sigma}_h^j$ and $\overline{\sigma}_h^{j-1}$ in (4.22) by $\frac{1}{\alpha}(\tilde{\sigma}_h^j-(1-\alpha)\sigma_h^j)$ and $\frac{1}{\alpha}(\overline{\sigma}_h^{j-1}-(1-\alpha)\tilde{\sigma}_h^{j-1})$, respectively, where α denotes the relaxation parameter ($\alpha = 1$ means no damping and $\alpha = 0$ total damping). For $\kappa = 0$ the smoothing factor of this damped relaxation is easily derived and it is (independent of h) given by

$$\mu^{GS}(\alpha) = \max((\frac{1}{5})^{\frac{1}{2}},(1-\alpha)^2). \tag{4.29}$$

So if the discrete equations are lumped it is useless to introduce a damping parameter in SBGS relaxation. Figure 4.1 shows a graph of $\mu^{GS}(\alpha)$ in the case $k=\frac{1}{6}$ (no lumping). Numerically we find an optimal smoothing rate $\mu^{GS}(\alpha_{\text{opt}})=0.369$ for $\alpha_{\text{opt}}=0.458$.

The Fourier representation of SBRB relaxation is obtained by a similar method. As usual, we write S_h^{RB} as the product of the partial step operators S_h^R and S_h^B,

$$\hat{\mathbf{S}}_h^{RB}(\omega) = \hat{\mathbf{S}}_h^R(\omega)\hat{\mathbf{S}}_h^B(\omega)$$

$$\hat{\mathbf{S}}_h^R(\omega) = \begin{bmatrix} s_1(\theta) & s_2(\theta+\pi) & s_3(\theta) & s_4(\theta+\pi) \\ s_2(\theta) & s_1(\theta+\pi) & s_4(\theta) & s_3(\theta+\pi) \\ s_5(\theta) & -is_5(\theta+\pi) & s_7(\theta) & -is_8(\theta+\pi) \\ is_5(\theta) & s_5(\theta+\pi) & is_8(\theta) & s_7(\theta+\pi) \end{bmatrix},$$

$$\left[\hat{\mathbf{S}}_h^B(\omega)\right]_{ij} = (-1)^{i+j}\left[\hat{\mathbf{S}}_h^R(\omega)\right]_{ij}, \tag{4.30}$$

with

$$s_1(\theta) = \kappa f(\theta)e^{i\frac{3\theta}{2}}\cos\frac{\theta}{2}, \qquad s_2(\theta) = -i\kappa f(\theta)e^{i\frac{3\theta}{2}}\sin\frac{\theta}{2},$$

$$s_3(\theta) = 2i\,\frac{f(\theta)}{h}\sin\theta\cos\frac{\theta}{2}, \qquad s_4(\theta) = 2\,\frac{f(\theta)}{h}\sin\theta\sin\frac{\theta}{2},$$

$$s_5(\theta) = \frac{h\kappa}{2}e^{i\frac{3\theta}{2}}(1+(1-\kappa)f(\theta)), \qquad s_8(\theta) = ie^{i\frac{\theta}{2}}\sin\frac{\theta}{2}+i(1-\kappa)f(\theta)\sin\theta,$$

$$s_7(\theta) = e^{i\frac{\theta}{2}}\cos\frac{\theta}{2}+i(1-\kappa)f(\theta)\sin\theta,$$

and

$$f(\theta) = \frac{-1}{2-2\kappa+\kappa e^{-2i\theta}}.$$

We see that all elements of $\hat{\mathbf{S}}_h^{RB}(\omega)$ remain bounded for $h\to 0$ and ω fixed. By doing so, we only consider the limit cases $|\theta|\to 0$, and $|\theta|\to\pi$. However, for $h\to 0$ and θ fixed $s_3(\theta)$ and $s_4(\theta)$ become unbounded. Numerical computation shows that the scaled norm of $\hat{\mathbf{S}}_h^{RB}(\omega)$ remains bounded:

$$\sup_{0\leq|h\omega|\leq\frac{\pi}{2}} \|\hat{\mathbf{S}}_h^{RB}(\omega)\|_H < \infty, \quad \text{for } h\to 0.$$

As SBRB relaxation mixes low and high frequencies the smoothing factor μ^{RB} is defined by

$$\mu^{RB} = \sup_{0\leq|h\omega|\leq\frac{\pi}{2}} \rho(\hat{\mathbf{Q}}\hat{\mathbf{S}}_h^{RB}(\omega)), \tag{4.31}$$

where $\hat{\mathbf{Q}}$ denotes the operator that annihilates all low frequencies

$$\hat{\mathbf{Q}} = \begin{bmatrix} 0 & & & \\ & 1 & & \\ & & 0 & \\ & & & 1 \end{bmatrix}. \tag{4.32}$$

If underrelaxation of σ_h^j is taken into account, we obtain for $\kappa=0$

$$\mu^{RB}(\alpha) = \max(\tfrac{1}{8},(1-\alpha)^2), \tag{4.33}$$

independent of h. So, again, it is not necessary to introduce a damping parameter if lumping is used. A plot of $\mu^{RB}(\alpha)$ for $\kappa=\frac{1}{6}$ is shown in Figure 4.2. In this case underrelaxation hardly improves the smoothing factor; numerically we find $\mu^{RB}(\alpha=1)=0.127$.

4.5. Two-grid algorithm for the 1D Poisson equation

In the Sections 4.3-4.4 we have shown that the scaled norm of the coarse grid correction operator and the relaxation operator remain bounded in the limit case of vanishing mesh size. As the scaled norm is not submultiplicative this does not imply that the scaled norm of the two-grid error amplification matrix (cf. 4.7) is bounded, and hence that the convergence rate of the two-grid algorithm is mesh-independent. If $\kappa=0$ both SBGS and SBRB relaxation without damping ($\alpha=1$) eliminate σ_h, so -in fact- we solve a second order difference equation for u_h. Therefore we may expect that the canonical grid transfer operators P_h^u and $\overline{R}_H^u$ are not accurate enough.

We show that this is indeed the case by studying the two-grid algorithm with SBGS relaxation for $\kappa=0$. If a single SBGS-sweep is used for pre- and post smoothing, the Fourier transform of $M_h^{1,1}$ after n cycles is given by

$$(\hat{\mathbf{M}}_h^{1,1}(\omega))^n = \left[\frac{e^{2i\theta}}{4-e^{-2i\theta}}\right]^n \begin{bmatrix} 0 & \mathbf{M}^{\sigma u} \\ 0 & \mathbf{M}^{uu} \end{bmatrix}, \tag{4.34}$$

with

$$n \text{ even} : \mathbf{M}^{\sigma u} = \begin{bmatrix} -\dfrac{2}{ih}\sin\dfrac{\theta}{2} & 0 \\ 0 & \dfrac{2}{ih}\cos\dfrac{\theta}{2} \end{bmatrix}, \qquad \mathbf{M}^{uu} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},$$

$$n \text{ odd} : \mathbf{M}^{\sigma u} = \begin{bmatrix} 0 & \dfrac{2}{ih}\cos\dfrac{\theta}{2} \\ \dfrac{2}{ih}\sin\dfrac{\theta}{2} & 0 \end{bmatrix}, \qquad \mathbf{M}^{uu} = \begin{bmatrix} 0 & \cot\dfrac{\theta}{2} \\ \tan\dfrac{\theta}{2} & 0 \end{bmatrix}. \tag{4.35}$$

The two-grid algorithm exhibits a typical alternating convergence behavior. An initial high frequency error mode u_h of amplitude b, causes a low frequency error mode of amplitude $b \cot\dfrac{\theta}{2}$ after a single two-grid cycle; so if $h \rightarrow 0$ initial high frequency error modes in u_h blow up. In the next cycle the large low frequency error mode u_h is nicely removed by the coarse grid correction, although a small high frequency error mode is introduced. This alternating behavior is reflected by the scaled norm of $(\hat{\mathbf{M}}_h^{1,1}(\omega))^n$; for $h \rightarrow 0$ we find

$$\sup_{0 \leqslant |h\omega| \leqslant \frac{\pi}{2}} \|(\hat{\mathbf{M}}_h^{1,1}(\omega))^n\|_H = \begin{cases} (\dfrac{5}{3^n})^{\frac{1}{2}}, & n \text{ even}, \\ \infty\ , & n \text{ odd}. \end{cases}$$

By numerical computation we observe a similar alternating convergence behavior for $\kappa = \dfrac{1}{6}$, even though the coarse grid operator L_H satisfies Galerkin's relation in this case.

The obvious remedy is to use more accurate grid transfer operators. We introduce $\tilde{P}_h^u$ the linear interpolation operator for u_h and $\tilde{R}_H^u$ its adjoint. The Fourier transforms of these more accurate grid transfer operators are

$$\hat{\hat{\mathbf{P}}}_h(\omega) = \begin{bmatrix} \hat{\hat{\mathbf{P}}}{}_h^{\sigma}(\omega) & 0 \\ 0 & \hat{\hat{\mathbf{P}}}{}_h^{u}(\omega) \end{bmatrix} = \begin{bmatrix} \cos^2\dfrac{\theta}{2} & 0 \\ \sin^2\dfrac{\theta}{2} & 0 \\ 0 & \cos^3\dfrac{\theta}{2} \\ 0 & \sin^3\dfrac{\theta}{2} \end{bmatrix}$$

and

$$\hat{\hat{\mathbf{R}}}_H(\omega) = \begin{bmatrix} \hat{\hat{\mathbf{R}}}{}_H^{\sigma}(\omega) & 0 \\ 0 & \hat{\hat{\mathbf{R}}}{}_H^{u}(\omega) \end{bmatrix} = \left[\hat{\hat{\mathbf{P}}}_h(\omega)\right]^{\mathrm{T}},$$

		$(P_h,\overline{R}_H)$	$(\tilde{P}_h,\tilde{R}_H)$
$\kappa=0$	SBGS	∞	0.346
	SBRB	0.590	0.099
$\kappa=1/6$	SBGS ($\alpha=1$)	∞	0.531
	SBGS ($\alpha=\alpha_{\text{opt}}$)	∞	0.368
	SBRB	0.628	0.339

TABLE 4.1. Norm of the two-grid error amplification matrix, $\sup_{0\leq|h\omega|\leq\frac{\pi}{2}} \|\hat{\mathbf{M}}_h^{1,1}(\omega)\|_H$.

	$(P_h,\overline{R}_H)$		$(\tilde{P}_h,\tilde{R}_H)$	
ν	SBGS	SBRB	SBGS	SBRB
1	0.577	0.500	0.343	0.096
2	0.333	0.325	0.145	0.046
3	0.192	0.259	0.088	0.031
4	0.111	0.221	0.058	0.023

TABLE 4.2. Two-level convergence factor λ_ρ^ν, $\kappa=0$.

	$(P_h,\overline{R}_H)$			$(\tilde{P}_h,\tilde{R}_H)$		
	SBGS	SBGS	SBRB	SBGS	SBGS	SBRB
ν	($\alpha=1$)	($\alpha=\alpha_{\text{opt}}$)		($\alpha=1$)	($\alpha=\alpha_{\text{opt}}$)	
1	0.657	0.477	0.457	0.476	0.391	0.264
2	0.429	0.211	0.367	0.311	0.277	0.178
3	0.281	0.147	0.314	0.236	0.212	0.136
4	0.184	0.093	0.278	0.177	0.166	0.111

TABLE 4.3. Two-level convergence factor λ_ρ^ν, $\kappa = \frac{1}{6}$.

so we have $m_L^u = 2$ and $m_H^u = 3$. Although it is not necessary to use $\tilde{P}_h^u$ for keeping the scaled norm of the error amplification matrix bounded, it is introduced to avoid similar problems with the amplification operator of the residuals. In Table 4.1 we show values for $\sup_{0\leqslant|h\omega|\leqslant\frac{\pi}{2}} \|\hat{\mathbf{M}}_h^{1,1}(\omega)\|_H$ for the different possible two-grid algorithms. If SBRB relaxation is used, the canonical grid transfer operators ($m_H^u = 1$) are sufficient: the high frequencies are so efficiently smoothed that they do not cause any problems. Here we see that the choice of the grid transfer operators is not only determined by the order of the difference equations, but is also influenced by the relaxation scheme.

The scaled norm of $\hat{\mathbf{M}}_h^{1,1}(\omega)$ only indicates what happens in a single two-grid cycle; the convergence rate after many cycles is estimated by the two-level convergence factor

$$\lambda_\rho^\nu = \sup_{0\leqslant|\theta|\leqslant\frac{\pi}{2}} \rho(\hat{\mathbf{M}}_h^{\nu_1,\nu_2}),$$

with $\nu = \nu_1 + \nu_2$. In Table 4.2 we show λ_ρ^ν for $\kappa = 0$ and for different values of ν. The combination of the transfer operators $\tilde{P}_h$ and $\tilde{R}_H$, and SBRB relaxation leads to a fast converging algorithm. In Table 4.3 we show λ_ρ^ν for $\kappa = \frac{1}{6}$. We see that the introduction of a damping parameter α in SBGS relaxation indeed leads to faster convergence, but the best convergence factors are again obtained by using the combination of SBRB relaxation and the transfer operators $\tilde{P}_h$ and $\tilde{R}_H$.

4.6. Two-grid algorithm for the 2D Poisson equation

So far we discussed the accuracy of the grid transfer operators for the 1D Poisson equation; in this Section we study the 2D case. The 2D Poisson equation is discretized on a uniform square mesh by means of lowest order Raviart-Thomas elements (cf. Section 2.4); for simplicity we only treat the case the equations are lumped ($\kappa = 0$).

If no underrelaxation is used ($\alpha = 1$), both SBGS and SBRB relaxation eliminate the variables $(\underline{\sigma}_{h,x}, \underline{\sigma}_{h,y})$ in a single sweep. From (4.2), (4.12) and (4.13) we see that, after a relaxation sweep, the Fourier transforms $(\hat{\sigma}_{h,x}, \hat{\sigma}_{h,y}, \hat{u}_h)$ of $(\underline{\sigma}_{h,x}, \underline{\sigma}_{h,y}, u_h)$ are related by

$$\hat{\sigma}_{h,x}(\omega_x, \omega_y) = -\frac{2i}{h_x} \sin\left[\frac{h_x\omega_x}{2}\right] \hat{u}_h(\omega_x, \omega_y) = -\hat{D}_{h,x}\hat{u}_h \tag{4.36a}$$

and

$$\hat{\sigma}_{h,y}(\omega_x, \omega_y) = -\frac{2i}{h_y} \sin\left[\frac{h_y\omega_y}{2}\right] \hat{u}_h(\omega_x, \omega_y) = -D_{h,y}\hat{u}_h, \tag{4.36b}$$

where $h = (h_x, h_y)$ denotes the mesh sizes in the two coordinate directions, and $(\omega_x, \omega_y) \in T_h{}^2 = (-\frac{\pi}{h}, \frac{\pi}{h}]^2$ the frequency. The matrix notation introduced in Section 4.3 is easily extended to the 2D case: every $(\omega_x, \omega_y) \in T_h{}^2$ can be written as a 4-vector on $T_H{}^2$ with entries $(\omega_x + p_x\frac{\pi}{h_y}, \omega_y + p_y\frac{\pi}{h_y})$, where $(p_x, p_y) \in \{(0,0), (1,0), (0,1), (1,1)\}$.

Using the techniques developed in the previous Sections we find that the Fourier transform of the two-grid error amplification matrix for the canonical grid transfer operators is given by

$$\hat{\mathbf{M}}_h^{1,1} = \begin{bmatrix} 0 & 0 & -\hat{\mathbf{D}}_{h,x}\hat{\mathbf{M}}_h^u \\ 0 & 0 & -\hat{\mathbf{D}}_{h,y}\hat{\mathbf{M}}_h^u \\ 0 & 0 & \hat{\mathbf{M}}_h^u \end{bmatrix}, \tag{4.37}$$

where

$$\hat{\mathbf{M}}_h^u = \hat{\mathbf{S}}_h(I - \hat{\mathbf{G}}_h^u)\hat{\mathbf{S}}_h, \tag{4.38}$$

$$(\hat{\mathbf{G}}_h^u)_{i,j} = \frac{4f_i f_j g_j}{\sin^2\theta_x + \sin^2\theta_y}, \qquad i,j = 1, \cdots, 4, \tag{4.39a}$$

$$\begin{aligned} f_1 &= \cos\frac{\theta_x}{2}\cos\frac{\theta_y}{2}, & g_1 &= \sin^2\frac{\theta_x}{2} + \sin^2\frac{\theta_y}{2}, \\ f_2 &= \sin\frac{\theta_x}{2}\cos\frac{\theta_y}{2}, & g_2 &= \cos^2\frac{\theta_x}{2} + \sin^2\frac{\theta_y}{2}, \\ f_3 &= \cos\frac{\theta_x}{2}\sin\frac{\theta_y}{2}, & g_3 &= \sin^2\frac{\theta_x}{2} + \cos^2\frac{\theta_y}{2}, \\ f_4 &= \sin\frac{\theta_x}{2}\sin\frac{\theta_y}{2}, & g_4 &= \cos^2\frac{\theta_x}{2} + \cos^2\frac{\theta_y}{2}, \end{aligned} \tag{4.39b}$$

I denotes the 4×4-identity matrix, $(\theta_x,\theta_y) = (h_x\omega_x, h_y\omega_y)$, and $\hat{\mathbf{S}}_h$ is the matrix representation of the smoothing operator for the equations from which $(\underline{\sigma}_{h,x}, \underline{\sigma}_{h,y})$ are eliminated. Using the trivial inequalities

$$\left|\frac{a^2}{a^2+b^2}\right| \leqslant 1 \quad \text{and} \quad \left|\frac{ab}{a^2+b^2}\right| \leqslant \frac{1}{2}, \quad \text{for} \quad (a,b) \in \mathbb{R}^2/\{0,0\},$$

we see from (4.39) that only error modes in $\underline{u}_h$ in the neighborhood of $(\theta_x,\theta_y) = (\pi,0)$ and $(\theta_x,\theta_y) = (0,\pi)$ are blown up in the coarse grid correction.

We proceed to show that these high frequency error modes are removed by the relaxation operator. The Fourier transform of SBGS relaxation is given by

$$\hat{S}_h^{GS}(\theta_x,\theta_y) = \frac{e^{i\theta_x} + e^{i\theta_y}}{4 - e^{-i\theta_x} - e^{-i\theta_y}}, \tag{4.40}$$

so the error modes $(\pi,0)$ and $(0,\pi)$ are eliminated by SBGS indeed. SBRB relaxation mixes low and high frequencies: its Fourier transform in matrix notation is

$$\hat{\mathbf{S}}_h^{RB}(\theta_x,\theta_y)=\frac{1}{2}\begin{bmatrix} s_0(1+s_0) & 0 & 0 & -s_0(1+s_0)\\ 0 & -s_1(1-s_1) & s_1(1-s_1) & 0\\ 0 & -s_1(1+s_1) & s_1(1+s_1) & 0\\ s_0(1-s_0) & 0 & 0 & -s_0(1-s_0)\end{bmatrix}, \qquad (4.41)$$

with

$$\begin{aligned} s_0(\theta_x,\theta_y) &= \frac{1}{2}(\cos\theta_x+\cos\theta_y),\\ s_1(\theta_x,\theta_y) &= \frac{1}{2}(\cos\theta_x-\cos\theta_y). \end{aligned}$$

As with SBGS, SBRB relaxation also eliminates the 'dangerous' error modes $(\pi,0)$ and $(0,\pi)$ without introducing them again.

Numerically we find that the scaled norm $\|\hat{\mathbf{M}}_h^{1,1}\|_H$ of the two-grid error amplification matrix is also bounded. In fact, we compute

$$\sup_{(\omega_x,\omega_y)\in T_H^2}\|\hat{\mathbf{M}}_h^{1,1}\|_H\approx 0.800, \quad \text{for SBGS relaxation}$$

and

$$\sup_{(\omega_x,\omega_y)\in T_H^2}\|\hat{\mathbf{M}}_h^{1,1}\|_H\approx 0.515, \quad \text{for SBRB relaxation.}$$

This guarantees that in the two-dimensional case neither for SBRB nor for SBGS relaxation are the error modes in u_h blown up by the two-grid algorithm, as happens in the 1D case when SB$\overline{\text{G}}$S relaxation is used. (Notice that 2D problems with line symmetry are essentially different from 1D problems.)

4.7. Concluding remarks

By local mode analysis we have shown that Vanka-type relaxation is an efficient smoother indeed for the mixed finite element discretization of Poisson's equation. Moreover, if the discrete equations are lumped it is useless to introduce a damping parameter in Vanka-type relaxation. Although lumping of the discrete equations spoils the Galerkin property of the coarse grid operator, it generally leads to faster converging two-grid algorithms. The Fourier transform of the two-grid error amplification operator shows that the canonical grid transfer operators are insufficiently accurate in the 1D case if a lexicographical ordering of the grid points is used in the relaxation procedure. However, they suffice if a red-black ordering is used. In the 2D case the canonical grid transfer operators can be used in combination with either of the relaxation patterns.

References

1. A. Brandt (1982). Guide to multigrid development, in *Multigrid Methods*, 220-312, ed. W. Hackbusch and U. Trottenberg, Springer-Verlag, Lecture Notes in Mathematics 960.
2. P.W. Hemker (1990). On the order of prolongations and restrictions in multigrid procedures, *J.Appl.Math.*, 32, 423-429.

3. K. Stüben and U. Trottenberg (1982). Multigrid methods: fundamental algorithms, model problem analysis and applications, in *Multigrid Methods*, 220-312, ed. W. Hackbusch and U. Trottenberg, Springer-Verlag, Lecture Notes in Mathematics 960.

4. S.P. Vanka (1986). Block-Implicit Multigrid Solution of Navier-Stokes Equations in Primitive Variables, *J.Comput.Phys.*, 65, 138-158.

Chapter 5

A multigrid method for the semiconductor equations

5.1. Introduction

In this Chapter we present a basic multigrid method for the solution of the two-dimensional semiconductor device equations. The equations are discretized by means of the lowest order Raviart-Thomas elements on a uniform rectangular grid (cf. Chapter 2). The resulting system of nonlinear equations is solved iteratively by means of a multigrid method. The multigrid method is based on the canonical grid transfer operators and a Vanka-type relaxation (cf. Chapter 4). In order to deal with the strong nonlinearity of the problem it appears necessary to apply a local damping of the restricted residual (cf. [9]). For a simple diode model problem we observe under these conditions a fast convergence that appears to be independent of the grid size. Hence, in combination with nested iteration, an efficient procedure is obtained.

An outline of this Chapter is as follows. In Section 5.2 we show how the Vanka-type relaxation is extended to deal with the set of nonlinear semiconductor equations. In the relaxation of a cell we have to solve a small nonlinear system of equations. This is done by Gummel's iteration, which appears to be more robust than Newton's iteration. We present an analysis of the convergence of Gummel's decoupling method for the solution of the small nonlinear systems. This analysis shows that the convergence of the point-wise Gummel iteration only depends on the difference in the values of ψ in the neighboring cells, and not on the initial estimate. Moreover, we show how the nonlinear equations in point-wise Gummel iteration can be solved efficiently. The nonlinear coarse grid correction stage is discussed in Section 5.3. Here we discuss the local damping of the restricted residual that is necessary to obtain a converging algorithm. Possible alternatives to the canonical prolongation operator are discussed in Section 5.4; good alternatives seem hard to find. In the Sections 5.5 and 5.6 we show how the equations on the coarsest grid are solved, and we describe a continuation method for the applied voltage. The results of the numerical experiments are presented in Section 5.7 and in the final Section of this Chapter we summarize our conclusions.

5.2. Vanka-type relaxation

For the efficiency of the multigrid method the choice of a proper smoother is of prime importance. Previous experience with the Poisson equation (cf. Chapter 3) indicated that the 5-point Vanka-type relaxation is a good candidate for a smoother. In this Section we show how Vanka-type relaxation can be applied to the semiconductor device equations. To solve the small nonlinear system of equations appearing in the relaxation of a cell we use Gummel's iteration (cf. [3]), and in Theorem 5.1 we present an analysis of the convergence of the point-wise Gummel iteration. Next we show how the nonlinear equations, that appear in Gummel's iteration can be solved efficiently by means of Schilders' correction transformation [7].

In Vanka-type relaxation all cells on a given level are visited in a predetermined order. When a cell is visited the variables related with that cell and the fluxes over its four edges are relaxed simultaneously. For a cell Ω^c, with nearest neighbors Ω^k, $k=n,e,s,w$, this means that we solve the system of equations (cf. 2.64 and 2.65)

$$h^n j_\psi^n + h^e j_\psi^e - h^s j_\psi^s - h^w j_\psi^w = a^c (e^{\phi_p^c - \psi^c} - e^{\psi^c - \phi_n^c} + \tilde{D}), \tag{5.1a}$$

$$h^n j_n^n + h^e j_n^e - h^s j_n^s - h^w j_n^w = +a^c R(\psi^c, \phi_n^c, \phi_p^c), \tag{5.1b}$$

$$h^n j_p^n + h^e j_p^e - h^s j_p^s - h^w j_p^w = -a^c R(\psi^c, \phi_n^c, \phi_p^c), \tag{5.1c}$$

with

$$\tilde{D} = \frac{1}{4} \sum_{\nu=1,4} D(\mathbf{x}^{c,\nu}), \tag{5.2}$$

a^c the area of the cell Ω^c with vertices $\mathbf{x}^{c,\nu}$, h^k the length of the edge E^k, D the given dope function and R the recombination rate of electrons and holes. The fluxes over for example the n-edge are

$$j_\psi^n = -\frac{h^n}{a_E^n} \mu_\psi (\psi^n - \psi^c), \tag{5.3a}$$

$$j_n^n = +\frac{h^n}{a_E^n} \mu_n \operatorname{Bexp}(-\psi^n, -\psi^c)(e^{-\phi_n^n} - e^{-\phi_n^c}), \tag{5.3b}$$

$$j_p^n = -\frac{h^n}{a_E^n} \mu_p \operatorname{Bexp}(+\psi^n, +\psi^c)(e^{+\phi_p^n} - e^{+\phi_p^c}), \tag{5.3c}$$

with $a_E^k = \text{area}(\Delta_E^k)$, Δ_E^k is the dual cell related to the edge E^k (cf. Figure 2.2). From this 15×15-system of equations the fluxes j_ψ^k, j_n^k and j_p^k can be eliminated easily. The resulting nonlinear 3×3-system could be solved by Newton's method, but the Jacobian matrix is possibly ill conditioned, if the initial guess is too far from the solution. Gummel's iteration (where the 3 nonlinear equations are solved sequentially) appears to be a more robust method for solving the nonlinear system. Here we analyze the convergence of Gummel's decoupling method. The analysis shows that the convergence of Gummel's method only depends on the difference in the values of ψ in the neighboring cells, and not on the initial estimate or on the properties of the dope function $D(\mathbf{x})$. In the spirit of [6] we study Gummel's iteration as a fixed-point mapping $T: \mathbb{R}^2 \to \mathbb{R}^2$, that maps a pair (ϕ_n, ϕ_p) onto a pair $(\tilde{\phi}_n, \tilde{\phi}_p) = T(\phi_n, \phi_p)$. To compute $T(\phi_n, \phi_p)$, first the electric potential $\psi(\phi_n, \phi_p)$ is computed as an intermediate result by the solution of (5.1a). The values $\tilde{\phi}_n$ and $\tilde{\phi}_p$ are obtained from this $\psi(\phi_n, \phi_p)$ by the solution of (5.1b,c). Existence of a solution in $A \subset \mathbb{R}^2$ follows when T is a contraction mapping on A. Then Gummel's iteration converges and the contraction factor may give an indication of the convergence speed of the iteration. To measure the distance in $\mathbb{R}^2$ we use the sup-norm:

$$\|(\phi_n^2, \phi_p^2) - (\phi_n^1, \phi_p^1)\|_\infty = \max(|\phi_n^2 - \phi_n^1|, |\phi_p^2 - \phi_p^1|). \tag{5.4}$$

In order to be more specific, we restrict the analysis to the zero recombination case. This enables us to find explicit expressions for the iterates.

Theorem 5.1. *If the variation in the ψ-values in the four neighboring points is sufficiently small* ($\max_k \psi^k - \min_k \psi^k < 12$), *then the operator T for the pointwise Gummel iteration is a contraction, i.e.*

$$\|T(\phi_n^1, \phi_p^1) - T(\phi_n^2, \phi_p^2)\|_\infty \leqslant C\|(\phi_n^1, \phi_p^1) - (\phi_n^2, \phi_p^2)\|_\infty, \tag{5.5}$$

with $C = \frac{1}{12}(\max_k \psi^k - \min_k \psi^k)$, *for all* $(\phi_n^i, \phi_p^i) \in \mathbb{R}^2$, $i = 1,2$.

Proof. The proof is given in two parts. We consider the iteration sequence

$$(\phi_n^i, \phi_p^i) \to \psi^i \to (\tilde{\phi}_n^i, \tilde{\phi}_p^i), \quad i = 1,2, \tag{5.6}$$

so that $\psi^i = \psi(\phi_n^i, \phi_p^i)$ and $(\tilde{\phi}_n^i, \tilde{\phi}_p^i) = T(\phi_n^i, \phi_p^i)$. In the first part we prove

$$|\psi^1 - \psi^2| \leqslant \|(\phi_n^1, \phi_p^1) - (\phi_n^2, \phi_p^2)\|_\infty, \tag{5.7}$$

and in the second part we show

$$\|(\phi_n^1, \phi_p^1) - (\phi_n^2, \phi_p^2)\|_\infty \leqslant C|\psi^1 - \psi^2|. \tag{5.8}$$

In fact we show (5.8) only for ϕ_p,

$$|\phi_p^1 - \phi_p^2| \leqslant C|\psi^1 - \psi^2|, \tag{5.9}$$

because a similar result for ϕ_n follows by analogy, and both results together yield (5.8).

In order to prove equation (5.7) we consider (5.1a), which yields for $i = 1,2$,

$$\sum_k w^k \mu_\psi(\psi^k - \psi^i) + (e^{\phi_p^i - \psi^i} - e^{\psi^i - \phi_n^i}) + D(x) = 0,$$

with $w^k = (h^k)^2 / a_E^k$. By subtraction we obtain

$$\sum_k w^k \mu_\psi(\psi^2 - \psi^1) + (e^{\phi_p^1 - \psi^1} - e^{\psi^1 - \phi_n^1} - e^{\phi_p^2 - \psi^2} + e^{\psi^2 - \phi_n^2}) = 0,$$

or

$$
(\psi^1 - \psi^2)\mu_\psi \sum_k w^k = (e^{\psi^1 - \phi_n^1}(e^{(\phi_n^1 - \phi_n^2) - (\psi^1 - \psi^2)} - 1) + \\
e^{\phi_p^2 - \psi^2}(e^{(\phi_p^1 - \phi_p^2) - (\psi^1 - \psi^2)} - 1)). \qquad (5.10)
$$

From this equality, the inequality (5.7) follows for the following reason.

Assume that (5.7) is *not* true, then we consider two cases: either $\psi^1 - \psi^2 > 0$ or $\psi^1 - \psi^2 < 0$. In the former case from the negation of (5.7) follows that $\psi^1 - \psi^2 \geqslant \phi_n^1 - \phi_n^2$ and $\psi^1 - \psi^2 \geqslant \phi_p^1 - \phi_p^2$. It follows that the left-hand side of the equality (5.10) is positive and the right-hand side is negative. This is a contradiction. Similarly, if $\psi^1 - \psi^2 < 0$ it follows that $\psi^1 - \psi^2 \leqslant \phi_n^1 - \phi_n^2$ and $\psi^1 - \psi^2 \leqslant \phi_p^1 - \phi_p^2$. Now it follows that the left-hand side of the equality (5.10) is negative and the right-hand side is positive. This also yields a contradiction. Because (5.7) is trivially satisfied for $\psi^1 = \psi^2$, we may conclude that (5.7) holds.

In order to prove the second part (5.9), we consider (5.1c). With zero recombination this yields for $i = 1,2$, (dropping the subscript p)

$$
\sum_k w^k (\phi^k - \phi^i) \frac{\mathrm{Bexp}(\psi^k, \psi^i)}{\mathrm{Bexp}(\phi^k, \phi^i)} = 0,
$$

using the definition of Bexp for the denominators, we obtain

$$
e^{\phi^i} \sum_k w^k \, \mathrm{Bexp}(\psi^k, \psi^i) = \sum_k w^k \; e^{\phi^k} \; \mathrm{Bexp}(\psi^k, \psi^i). \qquad (5.11)
$$

First we notice that all factors and terms in this expression are positive, and hence $\min_k e^{\phi^k} \leqslant e^{\phi^i} \leqslant \max_k e^{\phi^k}$, for $i = 1,2$, which yields (without any restriction on ψ^k)

$$
\min_k \phi^k \leqslant \phi^i \leqslant \max_k \phi^k, \quad \text{for } i = 1,2,
$$

and

$$
\phi^1 - \phi^2 \leqslant |\max_k \phi^k - \min_k \phi^k|.
$$

Further, from (5.11) we derive

$$
e^{\phi^1 - \phi^2} = \frac{\sum_k w^k \, \mathrm{Bexp}(\psi^k, \psi^2)}{\sum_k w^k \, \mathrm{Bexp}(\psi^k, \psi^1)} \; \frac{\sum_k w^k \, \mathrm{Bexp}(\psi^k, \psi^1) e^{\phi^k}}{\sum_k w^k \, \mathrm{Bexp}(\psi^k, \psi^2) e^{\phi^k}}.
$$

Now we define ψ_A to be the value of ψ^k for which

$$
\frac{\mathrm{Bexp}(\psi_A, \psi^2)}{\mathrm{Bexp}(\psi_A, \psi^1)} \geqslant \frac{\mathrm{Bexp}(\psi^k, \psi^2)}{\mathrm{Bexp}(\psi^k, \psi^1)} \qquad (5.12)
$$

for all k, and similarly ψ_B such that

$$
\frac{\mathrm{Bexp}(\psi_B, \psi^1)}{\mathrm{Bexp}(\psi_B, \psi^2)} \geqslant \frac{\mathrm{Bexp}(\psi^k, \psi^1)}{\mathrm{Bexp}(\psi^k, \psi^2)}
$$

for all k, then

$$e^{\phi^1-\phi^2} \leqslant \frac{\mathrm{Bexp}(\psi_A,\psi^2)}{\mathrm{Bexp}(\psi_A,\psi^1)} \cdot \frac{\mathrm{Bexp}(\psi_B,\psi^1)}{\mathrm{Bexp}(\psi_B,\psi^2)} . \tag{5.13}$$

Taking the logarithm and introducing the function $g(x) = \log\left[\dfrac{x}{e^x - 1}\right]$, we may write (5.13) as

$$\phi^1 - \phi^2 \leqslant g(\psi^2 - \psi_A) - g(\psi^1 - \psi_A) - g(\psi^2 - \psi_B) + g(\psi^1 - \psi_B)$$

or

$$\phi^1 - \phi^2 \leqslant \int_{-\psi_B}^{-\psi_A} \int_{\psi^2}^{\psi^1} (-g''(x+y))\, dx\, dy \,.$$

Since

$$g''(x) = \frac{1}{2\cosh(x) - 2} - \frac{1}{x^2}$$

we know that $0 < -g''(x) \leqslant 1/12$ and

$$\phi^1 - \phi^2 \leqslant \tfrac{1}{12}(\psi_B - \psi_A)(\psi^1 - \psi^2).$$

To determine ψ_A and ψ_B we consider

$$\log\left[\frac{\mathrm{Bexp}(\psi, \psi^2)}{\mathrm{Bexp}(\psi, \psi^1)}\right] = \int_{\psi^1}^{\psi^2} g'(x-\psi)dx = (\psi^2 - \psi^1)g'(\psi^m - \psi)$$

for some $\psi^m \in (\psi^1,\psi^2)$. Because $g'(\psi^m - \psi)$ is a monotonically increasing function of ψ we find $\psi_A = \max_k \psi^k$ and $\psi_B = \min_k \psi^k$ if $\psi^2 > \psi^1$, and if $\psi^2 < \psi^1$ we have $\psi_A = \min_k \psi^k$ and $\psi_B = \max_k \psi^k$. It follows that

$$\phi^1 - \phi^2 \leqslant \tfrac{1}{12}(\max_k \psi^k - \min_k \psi^k)|\psi^1 - \psi^2|.$$

Because the superscripts 1 and 2 may be interchanged without changing the meaning of the right-hand side, this proves (5.9) and hence the theorem. □

The proof of the theorem, valid for zero recombination and zero source term, clearly shows that convergence may be slower if a source term for the continuity equations takes values that make the right-hand side of (5.11) smaller. No solution exists for the local nonlinear problem, if the source term makes the right-hand side of (5.11) negative. This means that large source terms with the wrong sign can cause the non-existence of a solution. Hence, we have to face the possibility that the correction equations in the multigrid process have no solution if the right-hand side of the equation gets too large.

After this analysis of point-wise Gummel iteration, we proceed by showing how the nonlinear equations in Gummel's iteration can be solved efficiently. The continuity equations to be solved are linear if expressed in Φ_n and Φ_p.

However, we want to avoid doing computations in these Slotboom variables. Therefore we use the correction transformation proposed by Schilders [7]. Suppose that we solve the equation

$$f(g(x)) = 0, \tag{5.14}$$

with $f, g \in C^1(\mathbb{R})$ by means of Newton's method. If we linearize (5.14) with respect to x we obtain the corrections

$$\Delta x^{(n)} = \frac{-f(g(x^{(n)}))}{f'(g(x^{(n)}))\, g'(x^{(n)})}. \tag{5.15}$$

However, sometimes we expect that it is better to linearize with respect to the variable $y = g(x)$, e.g. because f is a linear function; in this case we find corrections

$$\Delta y^{(n)} = -\frac{f(y^{(n)})}{f'(y^{(n)})}. \tag{5.16}$$

Without doing calculations in the variable y, we can do the Newton iteration as if the equation (5.14) were linearized with respect to y, by calculating corrections $\Delta x^{(n)}$ as in (5.15), and then updating $x^{(n+1)}$ as

$$g(x^{(n+1)}) = g(x^{(n)}) + \Delta y^{(n)} = g(x^{(n)}) + g'(x^{(n)})\Delta x^{(n)}. \tag{5.17}$$

In case of the continuity equations we linearize and calculate corrections $\Delta\phi_n^{(n)}$ and $\Delta\phi_p^{(n)}$ for the quasi-Fermi potentials $\phi_n^{(n)}$ and $\phi_p^{(n)}$, and then apply the correction transformation (5.17), with $g=\exp(-x)$ and $g=\exp(+x)$, respectively:

$$\phi_n^{(n+1)} = \phi_n^{(n)} - \log(1 - \Delta\phi_n^{(n)}), \tag{5.18a}$$

$$\phi_p^{(n+1)} = \phi_p^{(n)} + \log(1 + \Delta\phi_p^{(n)}). \tag{5.18b}$$

Large corrections may yield negative arguments for the logarithmic function. If this happens, we damp the correction by replacing the function $\log(1+x)$ in (5.18) by the $C^1(-\infty,\infty)$ function $\mathrm{logPlus1}(x)$, identical to $\log(1+x)$ for $x > -1+\epsilon$, that is defined by (cf. [5])

$$\mathrm{logPlus1}(x) = \begin{cases} \log(1+x), & \text{for } x > -1+\epsilon, \\ 2\log(\epsilon) - \log(2\epsilon - x - 1), & \text{for } x \leqslant -1+\epsilon. \end{cases}$$

The implementation of the function $\mathrm{logPlus1}(x)$ is given in Appendix A. Without rounding errors, this would solve the continuity equations in a single Newton step; in practice a small number of iterations may be necessary. So by using a correction transformation in Newton's method the local continuity equations are solved efficiently.

In the following it is described how the local nonlinear Poisson equation is solved efficiently by a modified Newton method. To simplify notation and without loss of generality, we write the Poisson equation, appearing in Gummel's iteration, as

$$\sinh \bar{\psi} + a\, \bar{\psi} = b, \tag{5.19}$$

with $a > 0$. In principle equation (5.19) is solved by Newton's method. However, if the Jacobian is dominated by the sinh-function, it is better to linearize the equation with respect to the variable $\sinh \psi$. A suitable correction transformation strategy for the iterands $\psi^{(n)}$ in Newton's method, that switches between the two linearizations, is

$$\psi^{(n+1)} = \begin{cases} \operatorname{arsinh}(\sinh \psi^{(n)} + \Delta \psi^{(n)} \cosh \psi^{(n)}), & \text{if } |\frac{\cosh \psi}{a}| > 1, \\ \psi^{(n)} + \Delta \psi^{(n)}, & \text{otherwise.} \end{cases} \tag{5.20}$$

The iteration stops if $|\Delta \psi^{(n)}|$ is sufficiently small.

Next we study the initial iterand for Newton's iteration. If the last available iterand is taken as initial guess, we observe that large, untransformed corrections $\Delta \psi^{(n)}$ may cause overflow. To avoid this situation the process is restarted with a better initial estimate as soon as an untransformed correction is too large ($|\Delta \psi^{(n)}| > 50$). We consider two initial estimates for (5.19): $\psi^{(0)} = \operatorname{arsinh}(b)$ and $\psi^{(0)} = \dfrac{b}{a+1}$. To judge the feasibility of these initial estimates, we use the fact that the solution $\bar{\psi}$ of (5.19) minimizes the convex function

$$F(\psi) = \cosh \psi + a\, \frac{\psi^2}{2} - b\, \psi. \tag{5.21}$$

If the initial estimate $\psi^{(0)}$, for which F attains a minimal value, is chosen as the initial iterand, Newton's method converges rapidly (at most 4 steps are needed in the cases we studied).

	Reverse bias		Forward bias	
	1 Newton step	Solve Exact	1 Newton step	Solve Exact
No. of processes	21.824	21.824	21.824	21.824
Mean no. of Gummel its.	2.8	2.8	4.2	4.1
Max no. of Gummel its.	9	8	9	9
Mean no. of steps for (5.19)	1.0	1.4	1.0	2.0
Max no. of steps for (5.19)	1	6	1	6
Divergent process	27	0	0	0

A 'process' is the solution of a 3×3 nonlinear system, by Gummel iteration. The 'number of steps for (5.19)' is the number of Newton steps to solve Poisson's equation in Gummel's iteration. A process is called divergent, if Gummel's iteration does not converge within 25 steps.

TABLE 5.1. Solution of small nonlinear systems by Gummel's iteration.

This concludes our description of the use of Gummel's iteration for the solution of the small nonlinear systems appearing in the five-point Vanka-type relaxation. To illustrate the robustness of this method, we use a two-dimensional diode test problem (cf. Section 5.7), with either a forward biased (-1.0 V) or a reverse biased ($+5.0$ V) applied voltage. The performance of the relaxation process is shown in Table 5.1. Starting from a 4×4 grid, we perform two symmetric relaxation sweeps on every grid, before we interpolate the solution to a next finer grid. (No coarse grid corrections are applied.) The finest grid used is a 64×64 grid. In Table 5.1 we show results for the cases that either Poisson's equation is solved accurately (i.e. the modified Newton iteration is stopped if $|\Delta\psi^{(n)}| < 1.0\times10^{-12}$) in each Gummel step, or that the solution of Poisson's equation is approximated by a single step from a Newton iteration, using the last available iterand as initial estimate. In both cases the Gummel iteration is stopped if

$$|\Delta\psi^{(n)}| + |\Delta\phi_n^{(n)}| + |\Delta\phi_p^{(n)}| < 10^{-12}.$$

From Table 5.1 we see that the efficiency of Gummel's iteration is good, even in the forward biased case, in which the equations are strongly coupled. Solving Poisson's equation exactly during each step does not improve the efficiency of Gummel's iteration, but robustness is enhanced indeed.

5.3. The coarse grid correction

When, for the solution of the nonlinear discretized equations on the fine grid,

$$N_h(\bar{q}_h) = f_h, \tag{5.22}$$

we consider the usual nonlinear coarse grid correction stage of a two-grid algorithm,

$$N_H(\tilde{q}_H) = N_H(q_H) + \bar{R}_H(f_h - N_h(q_h)), \tag{5.23}$$

$$\tilde{q}_h = q_h + P_h(\tilde{q}_H) - P_h(q_H), \tag{5.24}$$

we recognize four important components that influence the effect of this stage. In the first place, there are the three operators N_H, $\bar{R}_H$, P_h, and further the starting approximation on the coarser grid q_H. We construct the coarse grid operator N_H by the same method as is used for N_h, which is described in Chapter 2. In principle, the choice for the operators $\bar{R}_H$ and P_h is free (as long as they are accurate enough), but in the context of our mixed finite element discretization there exist natural prolongation and restriction operators associated with the discretization, viz. those induced by the relations $V_H \subset V_h$ and $W_H \subset W_h$; for the definitions of V_h and W_h see Section 2.4. These relations imply that the prolongation corresponds for the potentials with piecewise constant interpolation, and for both components of the fluxes with piecewise linear interpolation in one direction and piecewise constant interpolation in the other. The corresponding prolongation stencils are $\begin{bmatrix}1 & 1\\ 1 & 1\end{bmatrix}$ for the potentials (associated with a cell), and $\begin{bmatrix}1/2 & 1/2\\ 1 & 1\\ 1/2 & 1/2\end{bmatrix}$, $\begin{bmatrix}1/2 & 1 & 1/2\\ 1/2 & 1 & 1/2\end{bmatrix}$, for the fluxes (associated with a horizontal and a vertical edge, respectively). The natural restriction $\overline{R}_H$ is the transpose of the natural prolongation P_h because the discretization presented in Chapter 2 is a Galerkin method, i.e. the spaces of test and trial functions are equal.

As initial approximation q_H for the coarse grid equation (5.23) we do *not* use the restriction of a solution on a finer grid, as described in [1], but for simplicity we take the last available iterand on the coarse grid. Such iterands are always available, because initial approximations for a finer grid are produced by interpolation from some approximation earlier computed on a coarser grid. The choice for the initial approximation q_H will be discussed again in the Sections 6.3 and 7.3.

For the semiconductor equations without a row scaling, the residual for the continuity equations (2.61b-c) corresponds with the rate-of-change in the carrier concentrations. In this unscaled form, the natural restriction operator has a "physical meaning": the sum of the rate-of-change in four small sub-cells corresponds with the total rate-of-change in the father cell. We believe that this is an advantageous property of the equations in their unscaled form. However, without row-scaling the size of the residuals (as well as the size of the diagonal elements of the Jacobian matrix) may vary largely in magnitude. In numerical experiments we encounter the situation that the values for a proper row-scaling, i.e. the size of the diagonal elements of the Jacobian matrix, differ strongly for the equations related with a coarse grid cell and the corresponding equations on the finer grid. In this case a small residual on the fine grid may yield an improper large correction on the coarse grid. This effect is seen in regions where the character of the solution changes rapidly (transition between *n*- and *p*-region, depletion layer). The same effect was observed by De Zeeuw in [9] for a one-dimensional case. De Zeeuw proposed to damp the restricted residual in order to avoid such problems, so the modified coarse grid equation reads (cf. (5.23))

$$N_H(\tilde{q}_H) = N_H(q_H) + D_H \overline{R}_H (f_h - N_h(q_h)). \tag{5.25}$$

The damping operator D_H is a diagonal operator, which has entries in $[0,1]$, depending on the current coarse and fine grid solution. Here we apply a similar technique for the two-dimensional case.

Suppose that the Jacobian matrix of the discretized semiconductor equations is given by

$$J_h(\phi_h^i,\phi_h^l) = \frac{\partial (N_H(\phi_H))^i}{\partial \phi_h^l}, \qquad \phi = \psi, \phi_n, \phi_p. \tag{5.26}$$

For every cell Ω_H^I, which is split into four cells Ω_h^i, we determine the diagonal elements of D_H by locally comparing the diagonal elements of the coarse and fine grid Jacobian matrices (cf. [9]):

$$D_H^I = \min\left(1, \frac{2\,|J_H(\phi_H^I,\phi_H^I)|}{\max\limits_{i=1,4}|J_h(\phi_h^i,\phi_h^i)|}\right), \qquad \phi = \psi, \phi_n, \phi_p. \tag{5.27}$$

If the mesh becomes fine enough, sharp layers are well resolved, the coarse and fine grid Jacobians gain in similarity, and the damping disappears, as we see from (5.27).

However, only damping the restricted residual does not guarantee that there will be no locally spurious corrections to the fine grid solution, if the grids are relatively coarse. Therefore we also suppress the coarse grid correction locally, if layers are not properly resolved. In fact, we suppress the coarse grid correction from a cell Ω_H^I, split into four cells Ω_h^i, if

$$\max_{i=1,4} |(2\psi_H^I - \phi_{n,H}^I - \phi_{p,H}^I) - (2\psi_h^i - \phi_{n,h}^i - \phi_{p,h}^i)| > 50.0. \tag{5.28}$$

This means that the correction is suppressed if an n-region appears as a part of a p-region on the coarser grid, or conversely. In the context of the multigrid algorithm, the need for damping the restricted residuals and suppressing the coarse grid corrections can be understood as follows.

Locally the coarse grid solution is a bad representation of the fine grid solution, because the grids are too coarse. However it is known that even very coarse grids still may help to reduce the low frequency error components. By locally damping the interaction between the grids, we are still able to reduce these low frequency error components in some parts of the solution, without exciting high frequency error components in other parts. If necessary, additional local relaxation can reduce errors in regions where the interaction between the grids is affected by damping; in our numerical experiments, however, this does not influence the observed convergence behavior.

5.4. Other prolongation operators

A priori there is no reason to assume that the natural prolongation operator P_h is the best possible prolongation operator. Because of the asymmetric character of the convection operator, and in view of the successful use of an asymmetric prolongation in a multigrid method for the one-dimensional semiconductor problem in [4, 5, 9] it is interesting to consider the possibility of extending this approach to the two-dimensional case. In 1D such an interpolation was based on the form

$$\Phi(x) - \Phi(a) = \int_a^x \operatorname{grad} \Phi(x) = \int_a^x e^{\pm\psi(\xi)} \mathbf{J} d\,\xi, \tag{5.29}$$

with the assumption of a piecewise constant $\mathbf{J}$ and a piecewise linear ψ over the area of integration (the dual boxes). In our mixed finite element context, the same exponential interpolation formula is found in Chapter 2 as equation (2.66). The principle behind the construction of the prolongation in the one-dimensional case is the equal flux over corresponding coarse and fine grid edges. In two dimensions, however, such an explicit prolongation cannot be constructed. This is because in two dimensions the assumption of a piecewise constant $\mathbf{J}$ and the existence of a unique function Φ leads to an inconsistency. Independence of $\Phi(x)$ on the integration path means $\operatorname{grad} \Phi = \exp(\pm\psi)\mathbf{J}$. This relation only holds for ψ and $\mathbf{J}$ satisfying

$$0 = \operatorname{rot}\operatorname{grad}\Phi = \operatorname{rot}(e^{\pm\psi}\mathbf{J}) = e^{\pm\psi}(\operatorname{rot}\mathbf{J} \pm \mathbf{J}\times\operatorname{grad}\psi). \tag{5.30}$$

With the assumption of a constant **J**, this implies that **J** should be parallel to grad ψ. However, in the two-dimensional case, this is generally too restrictive a condition. Assuming that the dependence of the integration path has only a minor influence, we might overlook the non-uniqueness of Φ and select a path, e.g. select the shortest line segment from the coarse cell center (with the known potential) to the fine cell center (where the potential has to be computed); such a prolongation has been proposed by e.g. Stelter in [8].

We now show that for the Scharfetter-Gummel discretization this may lead to negative Slotboom variables, which is clearly unacceptable. Consider a coarse grid cell Ω^C with kid cell Ω^{ne} and with neighbors Ω^N and Ω^E as in Figure 5.1. When we assume that the fluxes are piecewise constant on the dual boxes related to edges (cf. Figure 2.2) and that the grid consists of square cells (with side length h), we obtain for Poisson's equation (cf. (2.61d))

$$\psi^{ne} = \psi^C - \int_{\mathbf{x}^C}^{\mathbf{x}^{ne}} \mu_\psi{}^{-1}\mathbf{j}_\psi \cdot \mathbf{s}\, ds,$$

with $\mathbf{j}_\psi = -\mu_\psi h^{-1}(\psi^E - \psi^C, \psi^N - \psi^C)$, and $\mathbf{s}$ the constant unit vector along the integration path depicted in Figure 5.1, i.e. the straight line segment from $\mathbf{x}^C$ to $\mathbf{x}^{ne}$.

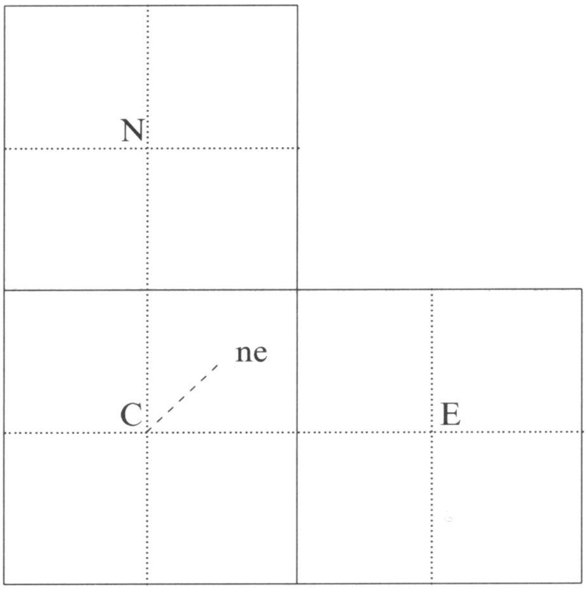


FIGURE 5.1. Numbering of cells for alternative prolongation.

This implies for the electric potential

$$\psi^{ne} = \frac{1}{2}\psi^{C} + \frac{1}{4}\psi^{N} + \frac{1}{4}\psi^{E}, \tag{5.31}$$

i.e. a bilinear interpolation. Analogously, we obtain for the Slotboom variables by using (2.61) and (2.62)

$$e^{\phi_p^{ne}} = e^{\phi_p^{C}} + \frac{1}{4}\mathrm{Bexp}^{-1}(\psi^{ne},\psi^{C})\big(\,\mathrm{Bexp}(\psi^{N},\psi^{C})(e^{\phi_p^{N}} - e^{\phi_p^{C}}) + \mathrm{Bexp}(\psi^{E},\psi^{C})(e^{\phi_p^{E}} - e^{\phi_p^{C}})\big). \tag{5.32}$$

Suppose that $\psi^C = \psi^E = 0$, $\psi^N = 20$, $\phi_p^C = \phi_p^N = 0$ and $\phi_p^E = -1$, then $\psi^{ne} = 5$ and numerically we find $\exp(\phi_p^{ne}) = -17.64$, so negative values for the Slotboom variables may occur in this prolongation. As there appears to be no good alternative to the canonical prolongation operator we use it in the numerical experiments that are presented in Section 5.7.

5.5. Solution method on coarsest grid

The solution procedure on the coarsest grid consists of a combination of Vanka-type relaxation sweeps and global Newton steps. To solve the linear systems in Newton's iteration we use the HARWELL sparse matrix solver. It analyzes the sparsity pattern of the Jacobian matrix, which needs only be done once as we always use the same discretization method on all grids. In the global Newton steps we use the correction transformation again. Now we transform the corrections point-wise with respect to the variables n and p, so (cf. 5.17)

$$\psi^{(n+1)} = \psi^{(n)} + \Delta\psi^{(n)}, \tag{5.33a}$$

$$\phi_n^{(n+1)} = \phi_n^{(n)} + \Delta\psi^{(n)} - \log(1 - (\Delta\phi_n^{(n)} - \Delta\psi^{(n)})), \tag{5.33b}$$

$$\phi_p^{(n+1)} = \phi_p^{(n)} + \Delta\psi^{(n)} + \log(1 + (\Delta\phi_p^{(n)} - \Delta\psi^{(n)})). \tag{5.33c}$$

The relaxation sweeps are introduced to make the solution procedure more robust (cf. [5]).

5.6. Continuation of the applied voltage

To start the multigrid algorithm, we first have to compute a solution on the coarsest grid. Initial estimates on the finer grids are obtained by interpolation from a coarser one. On the coarsest grid, we use a continuation strategy for the applied voltages at the contacts.

Starting at a voltage that yields a simple problem (e.g. zero voltage at all contacts), we change the boundary conditions stepwise to their final values. On the coarsest grid moving from one applied voltage to the next, we take the following steps: (i) change boundary conditions; (ii) find an initial approximation for these new boundary conditions; (iii) solve the problem on the coarsest grid.

The initial approximation for the new boundary conditions is obtained by a technique due to Mole c.s. [2]. Starting from a solution $(\psi^{(0)},\phi_n^{(0)},\phi_p^{(0)})$, we first assume that the carrier densities do not change during the continuation, and solve the following equations for the corrections $(\Delta\phi_n, \Delta\phi_p)$:

$$-\operatorname{div}\Delta\mathbf{j}_n = 0, \tag{5.34a}$$

$$-\operatorname{div}\Delta\mathbf{j}_p = 0, \tag{5.34b}$$

with

$$\Delta\mathbf{j}_n = -\mu_n e^{(\psi^{(0)} - \phi_n^{(0)})} \operatorname{grad}(\Delta\phi_n), \tag{5.34c}$$

$$\Delta\mathbf{j}_p = -\mu_p e^{(\phi_p^{(0)} - \psi^{(0)})} \operatorname{grad}(\Delta\phi_p). \tag{5.34d}$$

The Dirichlet boundary conditions are given by the change in the applied voltage. The linear equations (5.34) are discretized by the mixed finite element method as described in Chapter 2. The resulting system is solved iteratively by Gauss-Seidel relaxation; this iteration is stopped if the largest correction is a factor 10^{-2} less than the change in the applied voltage.

Next the initial approximation $(\psi^{(1)}, \phi_n^{(1)}, \phi_p^{(1)})$ is found by setting

$$\phi_n^{(1)} = \phi_n^{(0)} + \Delta\phi_n,$$

$$\phi_p^{(1)} = \phi_p^{(0)} + \Delta\phi_p,$$

and $\psi^{(0)}$ is updated in such a way that the density of the majority charge carries does not change, i.e.

$$\psi^{(1)} = \psi^{(0)} + \Delta\phi_n, \quad \text{in a } n\text{-region},$$

$$\psi^{(1)} = \psi^{(0)} + \Delta\phi_p, \quad \text{in a } p\text{-region}.$$

In exceptional cases, with a forward biased diode problem, we observed that the new minority level may temporarily become larger than the new majority level. However, this cause no problems, because of the robustness of our relaxation procedure.

5.7. Numerical experiment: diode problem

In order to test the basic multigrid algorithm presented in this Chapter we use a 2D quarter-circle diode problem . Figure 5.2 gives a schematic view of the geometry and the doping profile of this device, a detailed description can be found in Appendix B. For simplicity we assume a zero recombination rate, $R = 0$. At the two contacts the quasi-Fermi potentials ϕ_n and ϕ_p are given by the applied voltages and ψ is derived from these values, by assuming charge neutrality (cf. (1.4)),

$$p - n + D = 0. \tag{5.35}$$

At the remaining parts of the boundary homogeneous Neumann boundary conditions are assumed for all three equations. We consider two test cases: a reverse biased ($V_a = +5.0\,\mathrm{V}$) case and a forward biased case ($V_a = -1.0\,\mathrm{V}$). The coarsest grid used in the calculations is a uniform 4×4 grid. In all multigrid cycles a single symmetric Vanka-type relaxation sweep is made both before and after the coarse grid correction.

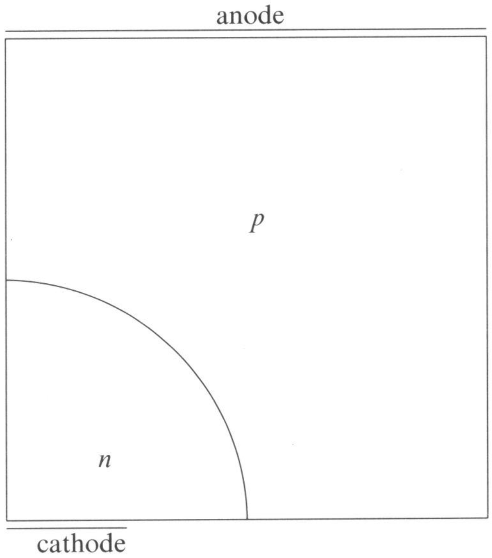
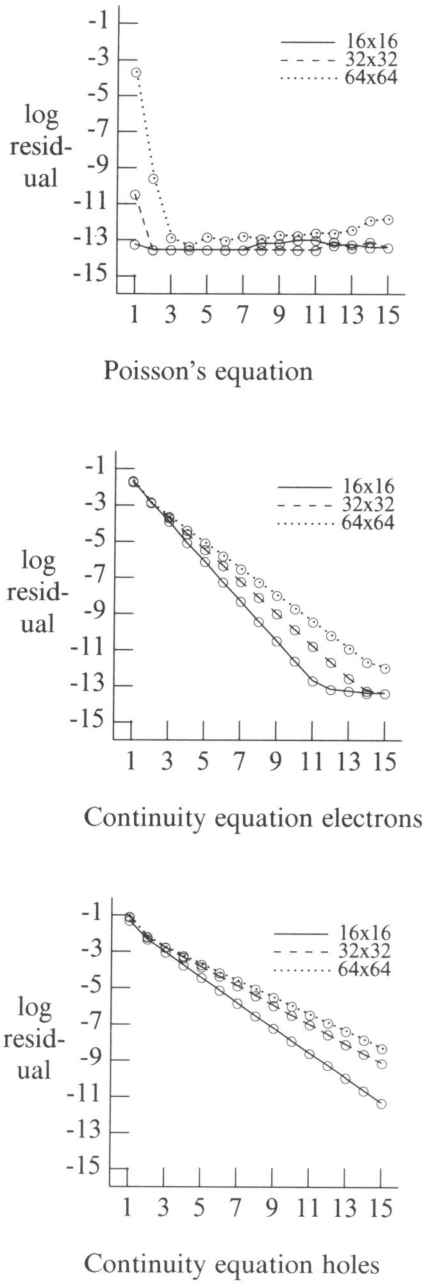


FIGURE 5.2. Configuration of quarter circle diode.

grid	cells with damping of the restricted residual	cells with suppressing of the correction.
4×4	6 (=38%)	1 (=6%)
8×8	10 (=16%)	4 (=6%)
16×16	16 (=6%)	7 (=3%)
32×32	28 (=2%)	15 (=1%)

TABLE 5.2. Damping of interaction between grids for the reverse biased diode.

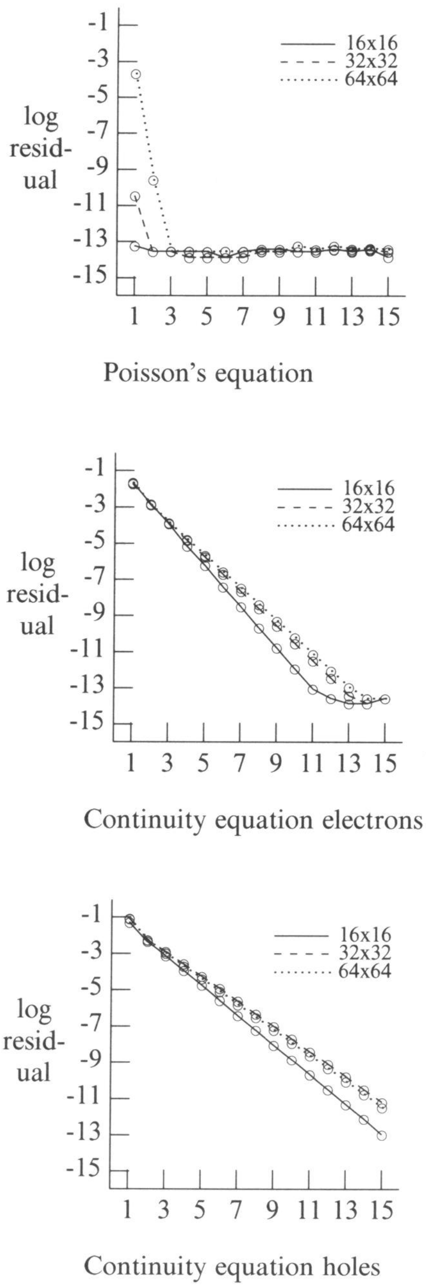


Poisson's equation

Continuity equation electrons

Continuity equation holes

FIGURE 5.3. Convergence behavior, reverse biased diode (V-cycles).

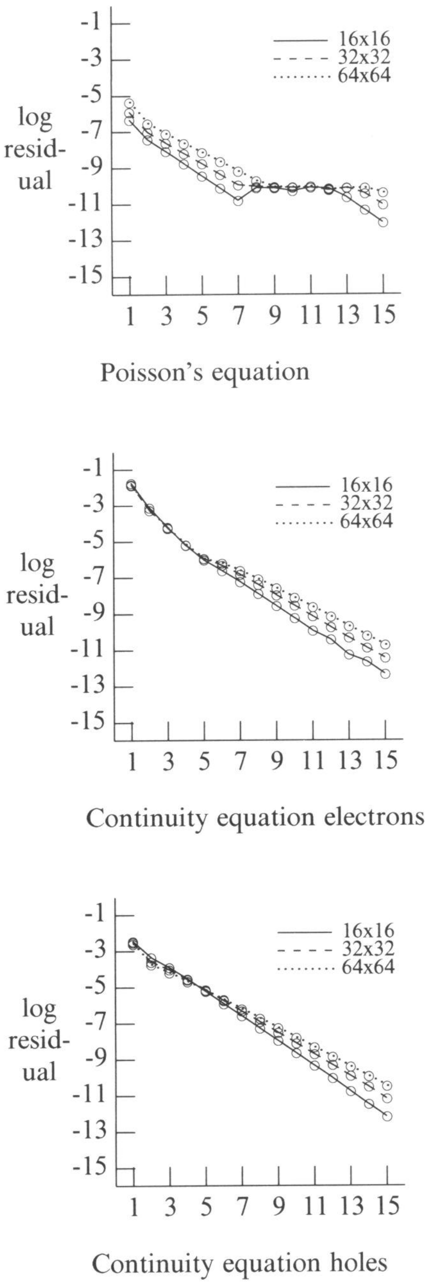


Poisson's equation

Continuity equation electrons

Continuity equation holes

FIGURE 5.4. Convergence behavior, reverse biased diode (W-cycles).

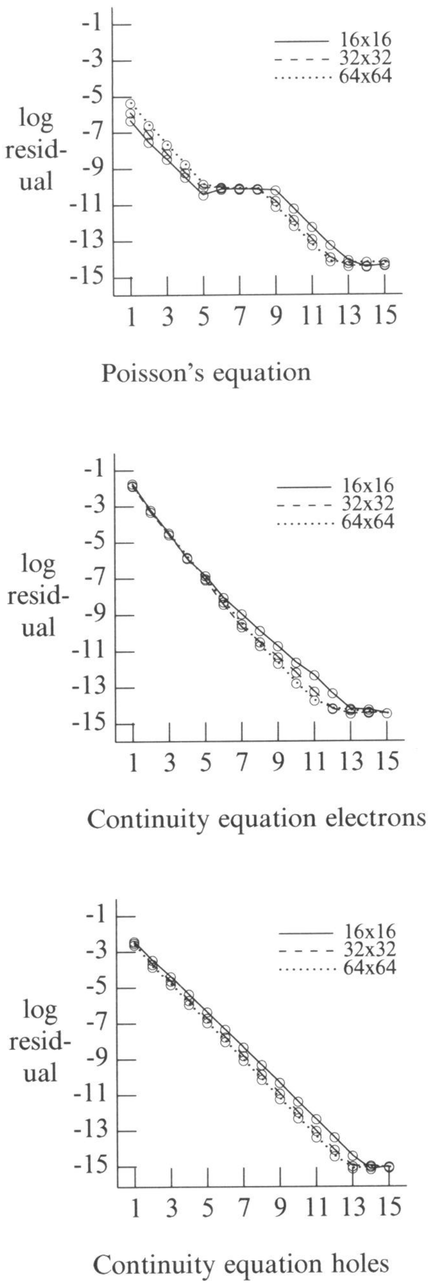


Poisson's equation

Continuity equation electrons

Continuity equation holes

FIGURE 5.5. Convergence behavior, forward biased diode (V-cycles).



Poisson's equation

Continuity equation electrons

Continuity equation holes

FIGURE 5.6. Convergence behavior, forward biased diode (W-cycles).

For the reverse biased problem, the convergence behavior for different meshes is shown in Figure 5.3 (V-cycles) and Figure 5.4 (W-cycles). The residual is scaled point-wise, by means of the diagonal elements of the Jacobian matrix: thus the scaled residual corresponds with corrections that would occur in a point-wise Jacobi relaxation. The maximum of this scaled residual is taken over the grid. In both cases it appears that Poisson's equation is solved up to machine precision in only a few cycles. Moreover, if W-cycles are used we find a nearly grid independent convergence behavior.

The Figures 5.5 and 5.6 show the convergence behavior for the forward biased problem, using V- and W-cycles. The convergence behavior for Poisson's equation looks irregular; it stalls until the continuity equations are solved sufficiently accurate. Again, we find a nearly grid independent convergence behavior for W-cycles.

Finally, in Table 5.2 we see that the interaction between the grids is damped only in a small percentage of the cells. This number decreases if the mesh gets finer. Damping only occurs in the reverse biased problem.

5.8. Concluding remarks

We have developed a basic multigrid method for the dual mixed finite element discretization of the semiconductor equations as described in Chapter 2. In the coarse grid correction it is necessary to apply a local damping of the restricted residual. If poor initial guesses are available Vanka-type relaxation with Gummel's iteration is robust, and the robustness is enhanced by solving exactly the nonlinear Poisson equation that appears in the point-wise Gummel iteration. Although the canonical prolongation operator is not very accurate, we do use it in our multigrid algorithm, as it appears that good alternatives are not available. In a numerical experiment this basic multigrid method yields good results for a simple diode problem. In the next Chapter we will develop more efficient multigrid algorithms.

References

1. A. Brandt (1982). Guide to multigrid development, in *Multigrid Methods*, 220-312, ed. W. Hackbusch and U. Trottenberg, Springer-Verlag, Lecture Notes in Mathematics 960.
2. S.P. Edwards, A.M. Howland, and P.J. Mole (1985). Initial guess strategy and linear algebra techniques for a coupled two-dimensional semiconductor equation solver, in *Proceedings NASECODE IV*, 272-280, Boole Press, Dublin.
3. H.K. Gummel (1964). A Self-Consistent Iterative Scheme for One-Dimensional Steady State Transistor Calculations, *IEEE Trans. Electron Devices*, ED-11, 455-465.
4. P.W. Hemker (1988). A nonlinear multigrid method for one-dimensional semiconductor device simulation, in *BAIL V*, ed. Guo Ben Yu, J.J.H. Miller and Shi Zhong-ci, Boole Press, Dublin.
5. P.W. Hemker (1990). A nonlinear multigrid method for one-dimensional semiconductor device simulation: results for the diode, *J.Comp.Appl.Math.*,

30, 117-126.

6. T. Kerkhoven (1988). On the effectiveness of Gummel's method, *SIAM J.Sci.Stat.Comput.*, 9, 48-60.
7. S.J. Polak, C. den Heijer, W.H.A. Schilders, and P. Markowich (1987). Semiconductor device modelling from the numerical point of view, *Int.J.Num.Meth.Engng.*, 24, 763-838.
8. A. Stelter (1990). *Gittertransferoperatoren für Mehrgitterverfahren zur Lösung der Kontinuitätsgleichungen in der Devicesimulation*, Diplomarbeit, Universität Hamburg, Hamburg.
9. P.M. De Zeeuw (1991). Nonlinear multigrid applied to a 1D stationary semiconductor model, *SIAM J.Sci.Stat.Comput.*, To appear.

Chapter 6

Adaptive multigrid applied to the semiconductor equations

6.1. Introduction

In this Chapter we present an adaptive multigrid method for the solution of the semiconductor equations. The discretization is made on an adaptive grid by means of the (hybrid) mixed finite element method on rectangles. The discrete equations thus obtained are solved iteratively by means of the dual version of a FAS-FMG algorithm. Contrary to the multigrid method presented in the previous Section, we now use a restriction of the fine grid solution as initial approximation on the coarse grid, and we do not damp the prolongation of the correction; again a collective Vanka-type relaxation is used as the smoother. As the semiconductor equations are strongly nonlinear and badly scaled, it is not straightforward to apply the multigrid method: special attention has to be paid to the formulation of the coarse grid problem. On the coarse grid we need some approximation of the solution. However, as we use the same discretization on all grids, this approximation implicitly determines the coefficients of the coarse grid problem. We discuss several possibilities of constructing a coarse grid approximation. In order to admit very coarse grids, it is still necessary to apply a local damping of the restricted residual on the coarse grids.

Another difficulty of the semiconductor equations is that they are singularly perturbed (cf. [7]), so we may expect that the dependent variables vary rapidly in small parts of the domain. Therefore it is desirable to have a fine mesh in parts of the domain where large variations of the solution occur. Several refinement criteria have been proposed for the semiconductor equations: estimates of the local truncation error, taking the singularly perturbed nature of the equations into account [7, 12], the second derivative of the electrostatic potential [10], or estimates of the error in the electric field and the current densities [3]. Our adaptive mesh refinement scheme is based on the equidistribution of the relative truncation errors between the coarse and fine grids. These relative truncation errors are approximations of the local truncation errors on the coarse grid. As there are three equations to be solved, we merge the different relative truncation errors into a single value for each cell in the mesh. The use of the relative truncation error as a refinement criterion is fully consistent with the dual version of a FAS-FMG algorithm (cf. [4]). We study the relative truncation errors for the semiconductor equations in detail and show how they can be incorporated into a practical grid adaptation scheme. The usefulness of this algorithm is demonstrated by means of a bipolar transistor test problem.

An outline of this Chapter is as follows. In Section 6.2 we describe the grids and the data-structure that is used for the adaptive calculations, and in Section 6.3 we present the multigrid method. The Vanka-type relaxation is discussed in Section 6.4. We give some implementational details, and introduce a line-wise version of Vanka-type relaxation. The relative truncation error and the adaptive mesh refinement strategy are discussed in Section 6.5, and the numerical experiments are presented in Section 6.6. In the final Section of this Chapter we summarize some conclusions.

6.2. Adaptive grids and data-structure

In this Section we present a method of grid generation that is very suitable for constructing adaptive grids, and that can handle a fairly wide range of geometries encountered in device simulation.

As before we assume that the domain $\Omega \subset \mathbb{R}^2$ can be divided by a regular partitioning in open disjoint, rectangular cells Ω_0^i, $\overline{\Omega} = \cup \overline{\Omega}_0^i$; these cells form the coarsest grid G_0 in a sequence of nested grids for the discretization.

On the set of cells a refinement operator $\mathfrak{R}$ is defined as the set-valued mapping, which splits one cell Ω_l^i of the grid into four smaller ones (cf. Figure 6.1). The class Q of admissible grids is specified recursively by two rules:

$$\text{i. } G_0 \in Q, \tag{6.1}$$

$$\text{ii. } G \in Q \Rightarrow \mathfrak{R}(G) \in Q.$$

The level l of a cell Ω_l^i, is defined as the minimum number of refinement steps between Ω_l^i and a cell of G_0. Using this definition we can classify the grids: a grid G_l of level l, is the set of all cells Ω_l^i. If a locally refined grid is used, there are interfaces between grids of a different level (cf. Figure 6.1). Following Schmidt and Jacobs [11] such interfaces are called 'green' interfaces.

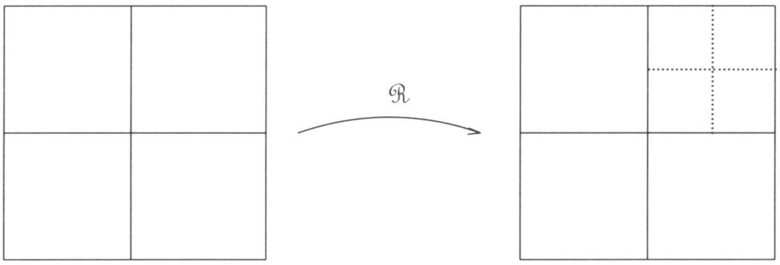


Figure 6.1. Refining the mesh by a refinement operator $\mathfrak{R}$.

In this way a nested sequence of partitionings of the domain Ω is obtained. Finer meshes may cover parts of Ω, but as soon as a fine level mesh exists in some area, also all coarser levels are available. The data-structure used for the implementation, a quad tree, reflects this structure of the grids. In every node of the tree (a cell or 'block') there are four pointers to possible offspring. The leaves of the tree correspond to unsplit cells. In addition, every node contains four pointers to interfaces, representing the sides of the cell. Neighboring cells on the same level are connected by their common interface. These interfaces are also used to distinguish between 'green' interfaces and physical boundaries.

To accommodate general geometries, the root of the tree needs not to represent G_0. So the first (negative) levels in the quad tree may contain entries, which are not necessarily related to a part of the domain. However, there must be a level in the tree which corresponds to G_0 exactly. The different numerical operations on data in the data-structure, are made by procedures that scan all cells, or all cells that satisfy a specific condition (e.g. all cells on a specified level), and which operate on each cell that is visited.

6.3. Multigrid on adaptive grids

Our algorithm for the iterative solution of the discretized equations on the adaptive grids is the dual version of the FAS-FMG algorithm (cf. [2, 4]). For the solution of the set of nonlinear equations, obtained after discretization,

$$N_h(\overline{q}_h) = f_h, \tag{6.2}$$

we consider the nonlinear coarse grid correction stage of a two-grid algorithm

$$N_H(\tilde{q}_H) = N_H(R_H q_h) + \overline{R}_H(f_h - N_h(q_h)), \tag{6.3}$$

$$\tilde{q}_h = q_h + P_h(\tilde{q}_H - R_H q_h), \tag{6.4}$$

where N_H denotes the nonlinear coarse grid operator, P_h the prolongation operator for the solution, and R_H and $\overline{R}_H$ the restriction operators for the solution and the residual, respectively. Contrary to the multigrid algorithm presented in Chapter 5 we now use the restriction of the fine grid approximation q_h as initial approximation on the coarse grid. In Chapter 5 we used the last available iterate as initial iterate, which is rather unsafe as q_H then depends on the history of the multigrid algorithm, so there is not guarantee that this iterate will remain in a neighborhood of the solution, and it may loose properties that are required for a proper approximate solution. The coarse grid operator N_H is constructed by discretization on the coarse grid, and on all grids we apply the same method of discretization.

In the dual version of FAS-FMG we rewrite (6.3) in the form

$$N_H(\tilde{q}_H) = \overline{R}_H f_h + \tau_{Hh}, \tag{6.5}$$

where

$$\tau_{Hh} = N_H(R_H q_h) - \overline{R}_H N_h(q_h) \tag{6.6}$$

denotes the relative truncation error that can be used as an approximation of the local truncation error of the coarse grid discretization. In this way, the fine grid is considered as a means of improving the right hand side of the coarse grid equations; therefore a grid needs not to be refined locally if the relative truncation error is sufficiently small. If this is the case we put $\tau_{Hh} = 0$. In Section 6.5 we describe how the relative truncation error is used as a refinement criterion.

As usual the coarse grid problem (6.5) is not solved exactly, but its solution is approximated by a combination of relaxation sweeps and coarse grid corrections on even coarser grids. Only on the coarsest grid the problem is solved accurately.

By using cell-wise refinement for all or part of the coarse grid cells, we get a nested sequence of approximating subspaces, $V_H \subset V_h$ and $W_H \subset W_h$. In this way, the mixed finite element method induces a natural set of grid transfer operators. For the operators $\bar{R}_H$ and P_h we use these natural prolongations and restrictions. This means that the prolongation for the potentials corresponds with piecewise constant interpolation, and for the fluxes with piecewise linear interpolation in one direction and piecewise constant in the other. The natural restriction $\bar{R}_H$ is the transpose of P_h because the spaces of test and trial functions are identical (cf. Chapter 2). We do not use a damping of the prolongation of the correction as proposed in Section 5.3, because it is rather arbitrary and it appears superfluous in numerical experiments.

It remains to specify the restriction operator R_H for the solution. At coarse grid edges E_H^j that are split into $E_h^{j_1}$ and $E_h^{j_2}$ we require current conservation

$$\sigma_H^j = (R_H \boldsymbol{\sigma}_h)_H^j = \frac{1}{2}(\sigma_h^{j_1} + \sigma_h^{j_2}). \tag{6.7}$$

This choice for R_H implies that we also have current conservation at the green edges: from (6.5) and (6.6) it follows that at convergence of the FAS-FMG algorithm we have $\boldsymbol{\sigma}_H = R_H \boldsymbol{\sigma}_h$. So all currents that flow out of the cells on the fine grid over a green edge, flow into the cells on the coarse grid.

The choice of a restriction operator for the potentials is less straightforward. In principle we could use the L^2-projection of any of the possible variable-sets (ψ, ϕ_n, ϕ_p), (ψ, Φ_n, Φ_p) or (ψ, n, p). The use of a restriction based on the Slotboom variables (ψ, Φ_n, Φ_p) is suggested by the discretization. However, in numerical experiments we observed that this may lead to coarse grid operators N_H of which the Jacobian matrix is ill-conditioned. Therefore we consider the other two possibilities.

For the semiconductor equations without any scaling, the residual of the conservation law (2.65b-c) for the continuity equations corresponds with the rate-of-change in the carrier concentrations. It may happen that the diagonal elements of the Jacobian matrices differ by orders of magnitude between a father cell and its four kid cells, especially if the transition between n- and p-region is not properly resolved on the coarse grid. In this case a small residual (after row scaling) on the fine grid may result in an unsuitable large correction on the coarse grid. For the 1D case De Zeeuw [13] proposed to apply a residual damping operator D_H in the coarse grid problem (6.3). This D_H is a diagonal matrix with entries in the interval [0, 1] that are determined by comparing the diagonal elements of the coarse and fine grid Jacobian matrices. The modified coarse grid equation then reads (cf. (6.3))

$$N_H(\tilde{q}_H) = N_H(R_H q_h) + D_H(R_H q_h, q_h)\,\overline{R}_H\,(f_h - N_h(q_h)). \qquad (6.8)$$

The elements of D_H differ from 1 only in small parts of the domain Ω (the transition regions), and the effect of damping is compensated in these regions by additional relaxation on the fine grid. The precise construction of D_H is described in Section 5.3.

This discussion makes clear that it is attractive to use a restriction that leads to coarse and fine grid Jacobians of which the corresponding diagonal entries are of comparable magnitude. The restriction based on the L^2-projection of the variables (ψ, n, p) appears to have this property: by rewriting eq. (2.61) in the form

$$\begin{aligned} -\operatorname{div}(\mu_n n \operatorname{grad}\phi_n) &= +R, \qquad (6.9)\\ -\operatorname{div}(\mu_p p \operatorname{grad}\phi_p) &= -R, \end{aligned}$$

we see that the diagonal elements of the Jacobian matrices with respect to the variable set (ψ, ϕ_n, ϕ_p) should be of comparable magnitude on the coarse and the fine grid, because the concentrations (n, p) are of comparable magnitude in the corresponding cells on the coarse and the fine grid. Indeed, in numerical experiments we observe that the diagonal elements of the damping operator D_H are all equal to 1. Unfortunately, we also observe that the coarse grid matrix tends to ill-conditioning in cases close to thermal equilibrium. We think that this is due to the following: for thermal equilibrium we have the trivial fine grid solution $\phi^i_{n,h} = \phi^i_{p,h} = 0$. In a coarse grid cell $\Omega^I_H = \bigcup_i \Omega^i_h$, $i = 1, \cdots, 4$, of which some kid cells are in n-region and others in p-region, we find (for a uniform grid)

$$n^I_H\, p^I_H = (\tfrac{1}{4}\sum_i n^i_h)(\tfrac{1}{4}\sum_i p^i_h) \gg 1,$$

which implies that $\phi^I_{n,H} \neq \phi^I_{p,H}$, so we get non-zero values for $\phi^I_{n,H}$ and/or $\phi^I_{p,H}$ on the coarse grid which is unphysical.

Therefore we propose a restriction that is based on the L^2-projection of the variables (ψ, ϕ_n, ϕ_p). In numerical experiments we observe (Section 6.6) that this choice yields a multigrid algorithm that is both robust and efficient, although the application of the residual damping operator D_H is necessary.

We complete this Section by describing the treatment of the 'green' edges, that appear on a partially refined grid. A straightforward approach is to impose inhomogeneous Neumann boundary conditions at the green edges on the fine mesh, as σ^j_H is given on the coarse mesh (cf. [11]). However, this can lead to patches on the fine grid that have only Neumann boundary conditions, so the solution u_h in such a patch is only determined up to an arbitrary constant. As there is no way to fix this constant for the semiconductor device equations, we have to impose Dirichlet boundary conditions at (at least some of) the green interfaces.

The Lagrange multipliers λ_h, as introduced in Section 2.6, are a good approximation of the potentials u_h at the edges (cf. [1]). So at a 'green' edge E_h^j, that is part of the coarse grid edge E_H^J, we use the Lagrange multiplier λ_H^J on the coarse grid as a Dirichlet boundary condition on the fine grid. As noticed before the flux σ_h^j at the green interface E_h^j is still a variable, so also on partially refined grids we have discrete current conservation due to the choice of the restriction operator for the fluxes $\boldsymbol{\sigma}_h$.

6.4. Vanka-type relaxation

Vanka-type relaxation applied to the semiconductor equations leads to a system of 15 equations (cf. Section 5.2). The fluxes appear linearly in these equations and they are easily eliminated, reducing the system to be solved for each cell to a set of three nonlinear equations. In this Section we describe how this nonlinear system can be solved by Newton's iteration, and we show how the different complex expressions involved are evaluated carefully. Moreover, we introduce a line-wise version of Vanka-type relaxation.

To solve the small system of nonlinear equations, we start using Newton's method combined with Schilders' correction transformation as in equation (5.33). The advantage of Newton iteration is that it converges quadratically in the neighborhood of the solution, whereas Gummel's iteration seems to converge only linearly (cf. Theorem 5.1). In the exceptional case that Newton's method fails we resort to point-wise Gummel iteration, which is less efficient but more robust (cf. Section 5.2). We now describe how to calculate the expressions needed in Vanka-type relaxation that have to be evaluated carefully because of otherwise possible roundoff errors.

The electron and hole currents over an edge E^j, with adjacent cells Ω^i, $i=R,L$, are (cf. eq. (2.64))

$$j_n^j = +\frac{h^j}{a_E^j}\mu_n \operatorname{Bexp}(-\psi^R, -\psi^L)(e^{-\phi_n^R} - e^{-\phi_n^L}), \tag{6.10a}$$

$$j_p^j = -\frac{h^j}{a_E^j}\mu_p \operatorname{Bexp}(+\psi^R, +\psi^L)(e^{+\phi_p^R} - e^{+\phi_p^L}), \tag{6.10b}$$

with Bexp(a, b) as in eq. (2.63), h^j the length of E^j and a_E^j the area of the dual edge related to the edge E^j. This is implemented as

$$j_n^j = -\frac{h^j}{a_E^j}\mu_n \operatorname{Dexp}(-\psi^R, -\psi^L, -\phi_n^R, -\phi_n^L)(\phi_n^R - \phi_n^L), \tag{6.11a}$$

$$j_p^j = -\frac{h^j}{a_E^j}\mu_p \operatorname{Dexp}(+\psi^R, +\psi^L, +\phi_p^R, +\phi_p^L)(\phi_p^R - \phi_p^L), \tag{6.11b}$$

with

$$\operatorname{Dexp}(a, b, c, d) = \frac{\operatorname{Bexp}(a, b)}{\operatorname{Bexp}(c, d)}. \tag{6.12}$$

The implementation of the functions Bexp(a, b) and Dexp(a, b, c, d) is given in Appendix A. The derivatives of the fluxes in (6.10) are calculated as

$$\frac{\partial j_n^j}{\partial \phi_n^L} = \frac{h^j}{a_E^j}\mu_n \operatorname{Bexp}(-\psi^R, -\psi^L)e^{-\phi_n^L}$$

$$= \frac{h^j}{a_E^j}\mu_n \operatorname{Bexp}(-\psi^R + \phi_n^L, -\psi^L + \phi_n^L), \tag{6.13a}$$

$$\frac{\partial j_n^j}{\partial \psi^L} = \frac{h^j}{a_E^j}\mu_n \operatorname{Cexp}(\psi^L - \psi^R)\operatorname{Bexp}(-\psi^R, -\psi^L)(e^{-\phi_n^R} - e^{-\phi_n^L})$$

$$= \operatorname{Cexp}(\psi^L - \psi^R)\, j_n^j, \tag{6.13b}$$

$$\frac{\partial j_p^j}{\partial \phi_p^L} = \frac{h^j}{a_E^j}\mu_p \operatorname{Bexp}(+\psi^R, +\psi^L)e^{+\phi_p^L}$$

$$= \frac{h^j}{a_E^j}\mu_p \operatorname{Bexp}(\psi^R - \phi_p^L, \psi^L - \phi_p^L) \tag{6.13c}$$

and

$$\frac{\partial j_p^j}{\partial \psi^L} = \frac{h^j}{a_E^j}\mu_p \operatorname{Cexp}(\psi^R - \psi^L)\operatorname{Bexp}(+\psi^R, +\psi^L)(e^{+\phi_p^R} - e^{+\phi_p^L})$$

$$= -\operatorname{Cexp}(\psi^R - \psi^L)\, j_p^j, \tag{6.13d}$$

with

$$\operatorname{Cexp}(x) = \frac{1}{x} + \frac{1}{1 - e^x}.$$

The implementation of the function $\operatorname{Cexp}(x)$ can also be found in Appendix A.

Finally we describe how the Lagrange multipliers λ_n^j and λ_p^j for the semiconductor equations (cf. eq. (2.66)) are evaluated. Both λ_n^j and λ_p^j can be written as λ^j in

$$e^{\lambda^j} = \frac{e^{\phi^L}(e^{\psi^R} - e^{\lambda_\psi^j}) + e^{\phi^R}(e^{\lambda_\psi^j} - e^{\psi^L})}{e^{\psi^R} - e^{\psi^L}}. \tag{6.14}$$

We take $\psi^R \geqslant \psi^L$ and calculate λ^j as follows:

$$\lambda^j = \tfrac{1}{2}(\phi^L + \phi^R) + \log\Big[e^{\frac{1}{2}(\phi^L - \phi^R)}\operatorname{Qexp}(\lambda_\psi^j - \psi^R, \psi^L - \psi^R) +$$

$$e^{\frac{1}{2}(\phi^R - \phi^L) + \lambda_\psi^j - \psi^R}\operatorname{Qexp}(\psi^L - \lambda_\psi^j, \psi^L - \psi^R)\Big], \tag{6.15}$$

with $\lambda_\psi^j = \frac{1}{2}(\psi^L + \psi^R)$ and

$$\operatorname{Qexp}(a,b) = \frac{1 - e^a}{1 - e^b}. \tag{6.16}$$

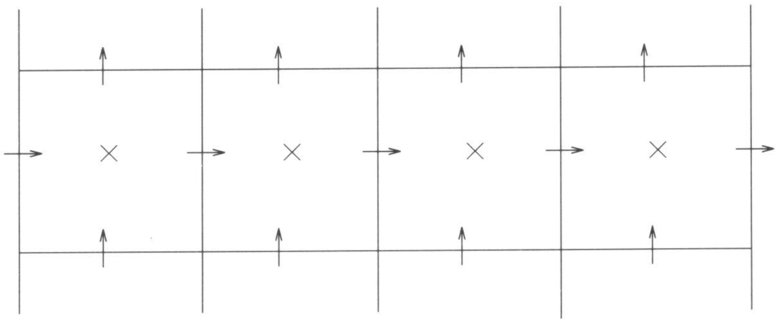


FIGURE 6.2. Line-wise Vanka-type relaxation.

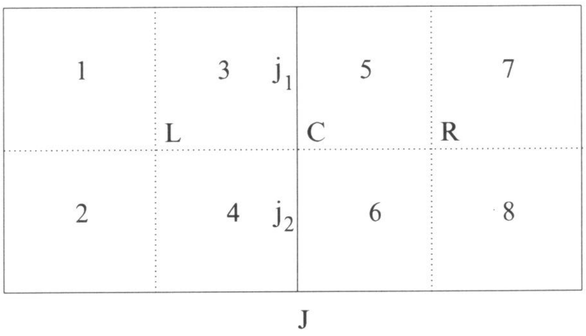


FIGURE 6.3. Numbering of cells used in calculation of relative truncation error.

So far we have only discussed the point-wise version of Vanka-type relaxation, but Vanka-type relaxation can also be used in a line-wise version. Then we take a line of cells and relax both the potentials in these cells and the fluxes at their edges simultaneously (cf. Figure 6.2). Again we can eliminate the fluxes to obtain a nonlinear system of which the Jacobian is a block tridiagonal matrix; each of the blocks is a 3×3-matrix. If sufficiently good initial estimates are available this system can be solved by Newton iteration. An experimental comparison of the efficiency of point- and line-wise Vanka relaxation is presented in Section 6.6.

6.5. Refinement criterion

As was seen in Section 6.3, it is fully consistent with the FAS-FMG method to use the relative truncation error as a refinement criterion in an adaptive mesh refinement strategy. Before describing the adaptive mesh generation we study the relative truncation error more closely.

Both for the equations related to the cells and to those related with the walls relative truncation errors can be defined. These are denoted by $\tau_{Hh}(\Omega_H^I)$ and $\tau_{Hh}(E_H^J)$, respectively. Using the definitions of the grid transfer operators given in Section 6.3, we find for the general problem (2.33), with a source f which is piecewise constant, that $\tau_{Hh}(\Omega_H^I)$ in a cell Ω_H^I, $\overline{\Omega}_H^I = \bigcup_{i=1,4} \overline{\Omega}_h^i$, is given by (cf. eq. (2.36))

$$\tau_{Hh}(\Omega_H^I) = f(\mathbf{x}_H^I) - \frac{1}{4} \sum_{i=1,4} f(\mathbf{x}_h^i), \tag{6.17}$$

where $\mathbf{x}_h^i$ denotes the centre of Ω_h^i. If lumping is used we obtain for $\tau_{Hh}(E_H^J)$ with $E_H^J = E_h^{j_1} \cup E_h^{j_2}$ (for the numbering of the cells see Figure 6.3)

$$\tau_{Hh}(E_H^J) = \tilde{a}_H^J \frac{u_H^R - u_H^L}{2h} - \frac{1}{2}(\tilde{a}_h^{j_1} \frac{u_h^5 - u_h^3}{h} + \tilde{a}_h^{j_2} \frac{u_h^6 - u_h^4}{h}), \tag{6.18}$$

with

$$u_H^L = \frac{1}{4} \sum_{i=1,4} u_h^i, \qquad u_H^R = \frac{1}{4} \sum_{i=5,8} u_h^i, \tag{6.19}$$

$2h$ the distance between $\mathbf{x}_H^L$ and $\mathbf{x}_H^R$, and (cf. (2.64)) $\tilde{a}_H^J = (-\mu_\psi, +\mu_n \mathrm{Bexp}(-\psi^R, -\psi^L), -\mu_p \mathrm{Bexp}(+\psi^R, +\psi^L))$, respectively. Due to the choice of the restriction operator R_H for the fluxes (6.7), $\boldsymbol{\sigma}_h$ does not appear in the relative truncation errors.

The following two theorems state the order behavior of the relative truncation errors on uniform grids in the limit case of vanishing mesh width for the three semiconductor equations. We assume that $(\psi, \phi_n, \phi_p) \in (C^3(\overline{\Omega}))^3$, and that $(\psi^i, \phi_n^i, \phi_p^i)$ are the averages of (ψ, ϕ_n, ϕ_p) over the cell Ω^i.

Theorem 6.1. *If the nonlinear source $f(\mathbf{x}, \psi, \phi_n, \phi_p)$ satisfies $f(\mathbf{x}, \psi, \phi_n, \phi_p) \in C^2(\overline{\Omega} \times \mathbb{R}^3)$, then for all three equations we have on uniform grids for $h \to 0$*

$$|\tau_{Hh}(\Omega_H^I)| \leqslant Ch^2 |f(\mathbf{x}, \psi, \phi_n, \phi_p)|_{2,\infty,\Omega_H^I}. \tag{6.20}$$

Proof. Using a Taylor expansion around $\mathbf{x}_H^I$, the centre of Ω_H^I, we obtain from (6.17)

$$\begin{aligned}|\tau_{Hh}(\Omega_H^I)| \leqslant \Big| & f(\mathbf{x}_H^I, \psi_H^I, \phi_{n,H}^I, \phi_{p,H}^I) \\ & - \Big(f(\mathbf{x}_H^I, \psi_H^I, \phi_{n,H}^I, \phi_{p,H}^I) + \tfrac{1}{4} \sum_{i=1,4} (\mathbf{x}_h^i - \mathbf{x}_H^I) \cdot \operatorname{grad} f(\mathbf{x}_H^I) \\ & + \tfrac{1}{4} \sum_{\phi=\psi,\phi_n,\phi_p} \frac{\partial f}{\partial \phi} \sum_{i=1,4} (\mathbf{x}_h^i - \mathbf{x}_H^I) \cdot \operatorname{grad} \phi(\mathbf{x}_H^I) \Big) \Big| \\ & + Ch^2 |f(\mathbf{x}, \psi, \phi_n, \phi_p)|_{2,\infty,\Omega_H^I} \\ = {} & Ch^2 |f(\mathbf{x}, \psi, \phi_n, \phi_p)|_{2,\infty,\Omega_H^I}.\end{aligned}$$

The last equality follows from a symmetry argument. □

We notice that the requirement for the source term f in Theorem 6.1 holds for the Shockley-Read-Hall recombination model, as well as for the Auger model, but it excludes the avalanche-generation term modeling impact ionization (cf. [12]).

In the next theorem we give the order behavior of the truncation error at the walls $\tau_{Hh}(E_H^J)$.

Theorem 6.2. *On uniform grids we have for Poisson's equation*

$$|\tau_{Hh}(E_H^J)| \leqslant Ch^2 |\psi|_{3,\infty,\Omega^L+\Omega^R}, \tag{6.21a}$$

and for the continuity equations for the electrons and holes

$$|\tau_{Hh}(E_H^J)| \leqslant \tilde{C} h^2 e^{\psi^C - \phi_n^C}, \tag{6.21b}$$

$$|\tau_{Hh}(E_H^J)| \leqslant \tilde{C} h^2 e^{\phi_p^C - \psi^C}, \tag{6.21c}$$

respectively, where $\tilde{C}$ is bounded by the supremum norm of the first, second and third order derivatives of (ψ, ϕ_n, ϕ_p).

Proof. The length of E_H^J is denoted by $2k$ and the coarse dual mesh size perpendicular to E_H^J by $2h$. By using a Taylor expansion of ψ around $\mathbf{x}^C$ (see Figure 6.3) we obtain in the case of Poisson's equation for h small enough

$$\begin{aligned}|\tau_{Hh}(E_H^J)| &= C \left| \frac{\psi_H^R - \psi_H^L}{2h} - \frac{1}{2} \left(\frac{\psi_h^5 - \psi_h^3}{h} + \frac{\psi_h^6 - \psi_h^4}{h} \right) \right| \\ &\leqslant \frac{C}{2h} | 2h\partial_x \psi^C - (\frac{k}{2} \partial_y \psi^C + h\partial_x \psi^C + \frac{hk}{4} \partial_{xy}^2 \psi^C)\end{aligned}$$

$$- (\frac{-k}{2}\partial_y\psi^C + h\partial_x\psi^C - \frac{hk}{4}\partial^2_{xy}\psi^C)| + Ch^2|\psi|_{3,\infty,\Omega^L+\Omega^R}$$

$$= Ch^2|\psi|_{3,\infty,\Omega^L+\Omega^R}.$$

For the continuity equation for holes (and likewise for electrons) we obtain by a Taylor expansion of ψ and ϕ_p around $\mathbf{x}^C$ and by using the equations (6.18) and (6.19)

$$|\tau_{Hh}(E^J_H)| =$$

$$C|\frac{\psi^R_H - \psi^L_H}{e^{\psi^R_H} - e^{\psi^L_H}} \frac{e^{\phi^R_{p,H}} - e^{\phi^L_{p,H}}}{h} -$$

$$(\frac{\psi^5_h - \psi^3_h}{e^{\psi^5_h} - e^{\psi^3_h}} \frac{e^{\phi^5_{p,h}} - e^{\phi^3_{p,h}}}{h} + \frac{\psi^6_h - \psi^4_h}{e^{\psi^6_h} - e^{\psi^4_h}} \frac{e^{\phi^6_{p,h}} - e^{\phi^4_{p,h}}}{h})|$$

$$\leqslant C\frac{e^{\phi^C_p - \psi^C}}{h}|(1+\tilde{C}h^2)(1+\tilde{C}h^2)(1+\tilde{C}h^2)(h\partial_x\phi^C_p + \tilde{C}h^3) -$$

$$(1+\frac{k}{4}\partial_y\phi^C_p + \tilde{C}h^2)(1-\frac{k}{4}\partial_y\psi^C + \tilde{C}h^2)(1+\tilde{C}h^2)(\frac{h}{2}\partial_x\phi^C_p + \frac{hk}{4}\partial^2_{xy}\phi^C_p + \tilde{C}h^3) -$$

$$(1-\frac{k}{4}\partial_y\phi^C_p + \tilde{C}h^2)(1+\frac{k}{4}\partial_y\psi^C + \tilde{C}h^2)(1+\tilde{C}h^2)(\frac{h}{2}\partial_x\phi^C_p - \frac{hk}{4}\partial^2_{xy}\phi^C_p + \tilde{C}h^3)|$$

$$\leqslant \tilde{C}h^2 e^{\phi^C_p - \psi^C}. \quad \square$$

Our adaptive mesh refinement scheme aims at equidistribution of the relative truncation errors; from Theorem 6.1 and 6.2 we conclude that it makes sense to refine the mesh in areas where the relative truncation errors are large.

In all finest cells and walls we define error indicators $\eta(\Omega^i_h, \phi)$ and $\eta(E^j_h, \phi)$, for $\phi = \psi, \phi_n, \phi_p$, by the relative truncation error in the parent cell and wall, respectively. Next, the three error indicators are merged by a summation of the normalized values, so we obtain single error indicators $\eta(\Omega^i_h), \eta(E^j_h) \in [0,1]$, for all cells and walls:

$$\eta(\Omega^i_h) = \frac{1}{3}\sum_{\phi} \frac{|\eta(\Omega^i_h,\phi)|}{\overline{\eta}_\Omega(\phi)}, \qquad \eta(E^j_h) = \frac{1}{3}\sum_{\phi} \frac{|\eta(E^j_h,\phi)|}{\overline{\eta}_E(\phi)}, \tag{6.22}$$

with, respectively,

$$\overline{\eta}_\Omega(\phi) = \max_i |\eta(\Omega^i_h, \phi)|, \qquad \overline{\eta}_E(\phi) = \max_j |\eta(E^j_h, \phi)|. \tag{6.23}$$

If the relative truncation errors are uniformly distributed then the error indicators are all equal to one and the grid is refined uniformly.

The actual mesh refinement procedure consists of two steps. In the first step we refine all cells and walls of which the error indicators $\eta(\Omega^i_h), \eta(E^j_h)$ are larger than user-defined parameters δ_Ω and δ_E, respectively; a wall is refined by refining both cells adjacent to it. In the second step we add some additional refinements in order to maintain a certain grid regularity: a cell is split if at least three of its neighbors are split, and the parent wall of a green wall is refined if it is also green.

In the next Section we use this grid adaptation scheme in practical calculations.

6.6. Numerical experiment: bipolar transistor

As a test problem for our multigrid algorithm we use the bipolar *npn*-transistor from the CURRY-example set [6]. Figure 6.4 gives a schematic view of the geometry of the transistor. The length of the device is 20 μm and the width is 8 μm; for a precise description of the device we refer to Appendix B. The generation-recombination rate is modeled by the usual Shockley-Read-Hall term (cf. [9]),

$$R^{SRH} = \frac{np - 1}{\tau_p(n+1) + \tau_n(p+1)}, \tag{6.24}$$

with carrier lifetimes $\tau_p = \tau_n = 10^{-6}$ s. The applied voltages at the collector and the base are kept constant at $V_c = 1.0$ V and $V_b = 0.0$ V, respectively. Starting from $V_e = -0.50$ V, the applied voltage at the emitter is lowered during the simulation to $V_e = -0.80$ V in steps of 0.05 V.

The coarsest grid used in our calculation consists of 4×10 squares. In fact, this mesh is even too coarse to resolve the contacts properly. An obvious generalization of the discretization is used to treat these parts of the boundary. It is assumed that the currents through such a boundary edge, which are determined by the Dirichlet boundary condition and the potential in the adjacent cell, only flow through that part of the edge that is covered by the contact. This approach makes it possible to use rather coarse and regular coarsest grids in our calculations, even if tiny contacts are present. These contacts are resolved properly on the finer grids, after a sufficient number of refinements has been made.

The continuation of the boundary conditions happens on the coarsest mesh. We start by solving the thermal equilibrium case, i.e. no applied voltages. After changing the boundary conditions we solve the problem on the coarsest grid using the previously obtained solution as initial iterate. The solution procedure on the coarsest grid consists of a combination of Vanka-type relaxation sweeps and Newton steps as described in Section 5.5. The new coarse grid solution is then interpolated to a next finer grid using the canonical prolongation operator described in Section 6.3. This fine grid approximation is improved iteratively by a few *W*-cycles. Due to the robustness of the solution procedure on the coarsest grid we are able to take large steps in the continuation process. For this problem we have also tried the continuation procedure described in Section 5.6, which aims at keeping the currents and the majority concentrations constant during the continuation of the applied voltage. This appeared to give no improvement, which -we believe- is due to the fact that the currents do change when the applied voltages are altered.

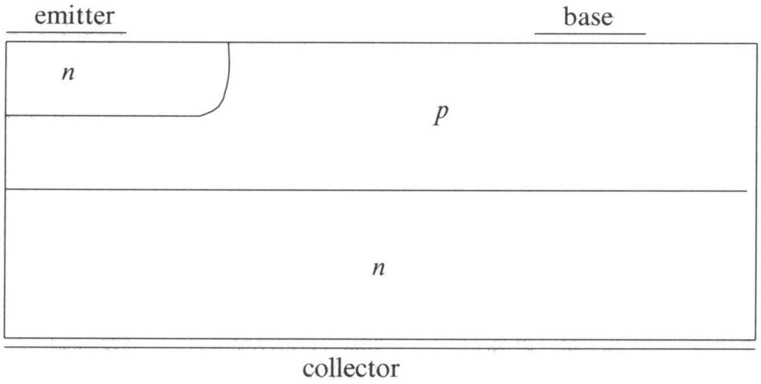


FIGURE 6.4. Configuration of a transistor.

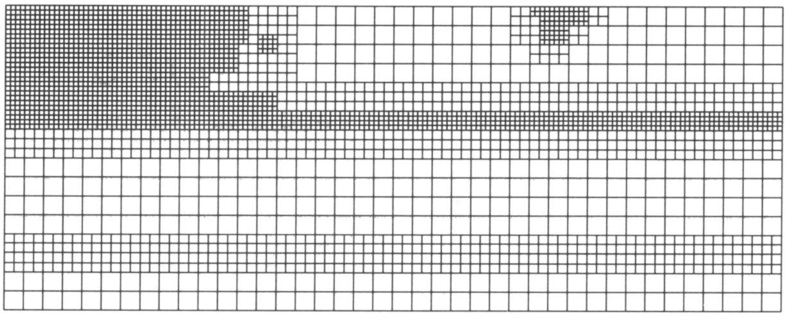


FIGURE 6.5. Example of adaptive grid.

	MFEM uniform mesh			reference solution
V_e	16×40	32×80	64×160	56×62
-0.50	9.5×10^{-5}	1.4×10^{-5}	1.0×10^{-5}	9.8×10^{-6}
-0.55	5.8×10^{-4}	9.5×10^{-5}	7.0×10^{-5}	6.7×10^{-5}
-0.60	3.4×10^{-4}	6.4×10^{-4}	4.8×10^{-4}	4.6×10^{-4}
-0.65	1.8×10^{-2}	4.3×10^{-3}	3.3×10^{-3}	3.1×10^{-3}
-0.70	8.4×10^{-2}	2.8×10^{-2}	2.2×10^{-2}	2.1×10^{-2}
-0.75	3.2×10^{-1}	1.7×10^{-1}	1.4×10^{-1}	1.3×10^{-1}
-0.80	$1.1\times10^{+0}$	7.9×10^{-1}	7.1×10^{-1}	6.9×10^{-1}

TABLE 6.1. Collector currents (A/cm).

	point-Vanka			line-Vanka		
V_e	16×40	32×80	64×160	16×40	32×80	64×160
-0.50	0.17	0.21	0.18	0.06	0.12	0.11
-0.55	0.18	0.20	0.18	0.07	0.13	0.12
-0.60	0.18	0.20	0.17	0.07	0.13	0.12
-0.65	0.17	0.20	0.17	0.08	0.13	0.12
-0.70	0.20	0.21	0.18	0.15	0.17	0.12
-0.75	0.31	0.24	0.17	0.21	0.21	0.12
-0.80	0.43	0.24	0.18	0.31	0.23	0.16

TABLE 6.2. Average residual reduction factor ρ for W-cycles.

$\delta_\Omega = \delta_E$	number of fine cells	relative error collector currents
0.01	5542	0.002
0.05	3007	0.012
0.10	1951	0.017
0.15	1909	0.073

TABLE 6.3. Results for adaptive grids, that correspond to a uniform 64×160 grid.

Table 6.1 shows the collector currents that are computed on the different grids, together with a reference solution computed with the CURRY package on a non-uniform 56×62 grid. It appears that the collector currents converge at least linearly (cf. Section 2.5) if the mesh size decreases. To estimate the convergence rate of the multigrid algorithm we introduce the average reduction factor ρ,

$$\rho = \left[\frac{d^{(10)}}{d^{(0)}} \right]^{\frac{1}{10}}, \tag{6.25}$$

where $d^{(i)}$ denotes the maximum of the scaled residual after i FAS-sweeps. The residual is scaled point-wise, by means of the diagonal 3×3 blocks of the Jacobian matrix: thus the scaled residual corresponds with corrections that would occur in a point-wise collective Jacobi relaxation. The maximum of this scaled residual is taken over the grid and over the three variables (ψ, ϕ_n, ϕ_p). Every FAS-sweep consists of a W-cycle: it appears that V-cycles are less robust for the semiconductor problem (cf. [5, 8]).

Table 6.2 shows the average reduction factor ρ for different grids both for symmetric point-Vanka and for alternating line-Vanka relaxation. We observe that the use of line-Vanka relaxation leads to a more efficient algorithm. The convergence behavior is not really grid independent (in some cases it appears that the convergence is faster on finer grids!), but in all cases the convergence is fast, and only a few iterations are necessary to attain truncation error accuracy.

Finally, we demonstrate the use of adaptive grids. Starting from the coarsest 4×10 grid, we add a single level of uniform refinement, which is necessary to estimate the relative truncation error on the coarsest grid. After solving the discrete equations on the first two grids, we refine the grid adaptively as described in the previous Section. Figure 6.5 shows an example of a grid generated by the adaptive procedure. The finest level corresponds to a uniform 64×160 grid. This adaptive grid is finally obtained for $V_e = -0.80\,\mathrm{V}$ with $\delta_\Omega = \delta_E = 0.1$.

It is clear that small values for δ_Ω and δ_E give rise to relatively fine grids, which means that the solution is more accurate at the expense of more computational work. Table 6.3 shows this tradeoff between the number of cells in the grid and the accuracy of the discrete solution calculated on the adaptive grid. The accuracy of the solution on the adaptive grid is measured by the relative error in the collector currents

$$\delta I_c(V_e) = \left| \frac{I_{c,U} - I_{c,A}}{I_{c,U}} \right|, \tag{6.26}$$

where $I_{c,A}$, $I_{c,U}$ denote the collector currents on the adaptive grid and the corresponding uniform grid, respectively. In fact, Table 6.3 shows the maximum value of $\delta I_c(V_e)$, and the maximum number of fine cells in the adaptive grid for the series of applied voltages at the emitter. In all cases we took $\delta_\Omega = \delta_E$. From Table 6.3 we conclude that it is indeed possible to save a substantial amount of work by using adaptive grids.

6.7. Concluding remarks

We have developed an adaptive multigrid algorithm for the solution of the semiconductor equations. Our adaptive grid refinement procedure is based on the relative truncation error, which is natural within the framework of the multigrid algorithm. The multigrid algorithm uses a Vanka-type relaxation. In numerical experiments it appears that the line-wise version of Vanka-type relaxation is more efficient than the point-wise version. The efficiency of the adaptive multigrid algorithm is demonstrated by means of a numerical experiment.

References

1. D.N. Arnold and F. Brezzi (1985). Mixed and non-conforming finite element methods: implementation, postprocessing and error estimates, *MMAN*, 19, 7-32.
2. A. Brandt (1982). Guide to multigrid development, in *Multigrid Methods*, 220-312, ed. W. Hackbusch and U. Trottenberg, Springer-Verlag, Lecture Notes in Mathematics 960.
3. K. Deljouie-Rakhshandeh (1988). A self-adaptive approach for numerical device simulation, in *Proc. SISDEP-88*, 519-527, ed. G.Baccarani and M.Rudan, Bologna.
4. P.W. Hemker (1980). On the structure of an adaptive multi-level algorithm, *BIT*, 20, 289-301.
5. P.W. Hemker (1990). A nonlinear multigrid method for one-dimensional semiconductor device simulation: results for the diode, *J.Comp.Appl.Math.*, 30, 117-126.
6. C. Lepoeter (1987). *CURRY example set,* Technical Report No. 4322.271.6005, Philips, Corp. CAD Centre, Eindhoven.
7. P.A. Markowich (1986). *The Stationary Semiconductor Device Equations*, Springer-Verlag, Wien, New York.
8. J. Molenaar and P.W. Hemker (1990). A multigrid approach for the solution of the 2D semiconductor equations, *IMPACT*, 2, 219-243.
9. S.J. Polak, C. den Heijer, W.H.A. Schilders, and P. Markowich (1987). Semiconductor device modelling from the numerical point of view, *Int.J.Num.Meth.Engng.*, 24, 763-838.
10. W.H.A. Schilders (1988). A novel approach to adaptive meshing for the semiconductor problem, in *Proc. SISDEP-88*, 519-527, ed. G.Baccarani and M.Rudan, Bologna.
11. G.H. Schmidt and F.J. Jacobs (1988). Adaptive Local Grid Refinement and Multi-grid in Numerical Reservoir Simulation, *J.Comput.Phys.*, 77, 140-165.
12. S. Selberherr (1984). *Analysis and Simulation of Semiconductor Devices*, Springer-Verlag, Wien.
13. P.M. De Zeeuw (1991). Nonlinear multigrid applied to a 1D stationary semiconductor model, *SIAM J.Sci.Stat.Comput.*, To appear.

Chapter 7

Cell-centered or vertex-centered multigrid ?

7.1. Introduction

So far we only studied the dual mixed finite element discretization of the semiconductor device equations. When suitable quadrature rules are used the dual mixed finite element discretization leads to a cell-centered finite volume discretization. The multigrid method that is used to solve these discretized equations can be classified as a cell-centered multigrid method. In this cell-centered multigrid method it appears necessary to apply local damping of the restricted residual in order to deal with the strong nonlinearity of the problem.

However we can also use the primal version of mixed finite element method to discretize the semiconductor equations. In this case we obtain a vertex-centered finite volume discretization. These equations can be solved by a vertex-centered multigrid algorithm. It is shown in Section 7.3 that it is not necessary to apply the local damping of the restricted residual in vertex-centered multigrid, provided that injection is used for the restriction of the residual. As is well known, injection is usually too inaccurate a grid transfer operator for second order differential equations: initial high frequency error modes are blown up in the coarse grid correction. Instead of using a more accurate restriction operator we construct a smoothing operator that effectively wipes out the 'dangerous' high frequency error modes. Then, we show by a two-grid analysis that the use of this smoothing operator leads to well-behaved two-grid algorithms indeed.

To compare the resulting cell-centered and vertex-centered multigrid algorithms in practice, we consider two test-problems: a MOS-transistor and an LDDMOS-transistor. In numerical experiments it appears that vertex-centered multigrid is more efficient and more robust than cell-centered multigrid.

An outline of this Chapter is as follows. In Section 7.2 we present the primal mixed finite element discretization for the steady semiconductor device equations. In Section 7.3 we discuss the formulation of the coarse grid correction in cell-centered and vertex-centered multigrid; we consider both the scaling problem of the coarse and fine grid matrices, and the stability of the coarse grid operator. The two-grid analysis is carried out in Section 7.4 and in Section 7.5 we present the results of the numerical experiments. In the last Section our conclusions are summarized.

7.2. Primal mixed finite element discretization

To derive a primal mixed finite element discretization for the semiconductor equations we again consider the standard second order elliptic boundary value problem (cf. (2.1))

$$\begin{aligned} \operatorname{div}(A \operatorname{grad} u) &= f, && \text{on}\,\Omega, \\ u &= g, && \text{on}\,\delta\Omega_D, \\ \mathbf{n}\cdot A \operatorname{grad} u &= 0, && \text{on}\,\delta\Omega_N, \end{aligned} \tag{7.1}$$

with $A > 0$, and $\delta\Omega_D$, $\delta\Omega_N$ the parts of the boundary with Dirichlet or homogeneous Neumann boundary conditions, respectively.

In the dual version of the mixed finite element method it is assumed that $(\boldsymbol{\sigma}, u) \in H^{BC}(\operatorname{div}, \Omega) \times L^2(\Omega)$. In the primal version, however, we take $\boldsymbol{\sigma} \in \hat{V} = (L^2(\Omega))^2$ and $u \in \hat{W} = H^1(\Omega)$. We introduce the bilinear forms $\hat{a} : \hat{V} \times \hat{V} \to \mathbb{R}$ and $\hat{b} : \hat{V} \times \hat{W} \to \mathbb{R}$ by

$$\hat{a}(\boldsymbol{\sigma}, \boldsymbol{\tau}) = a(\boldsymbol{\sigma}, \boldsymbol{\tau}) = \int_\Omega A^{-1} \boldsymbol{\sigma} \cdot \boldsymbol{\tau}\, d\Omega, \tag{7.2}$$

$$\hat{b}(\boldsymbol{\sigma}, t) = -\int_\Omega \boldsymbol{\sigma} \cdot \operatorname{grad} t\, d\Omega. \tag{7.3}$$

As before we assume that Ω can be divided in open disjoint, rectangular cells Ω^i, $\overline{\Omega} = \cup \overline{\Omega}^i$. To discretize (7.1) we define for each vertex $\mathbf{x}^l$, $\mathbf{x}^l \notin \delta\Omega_D$, of some cell Ω^i the piecewise bilinear function $\hat{e}^l$ by $\hat{e}^l(\mathbf{x}^k) = \delta_{lk}$, and for each edge E^j we define the piecewise constant vector-valued function $\hat{\boldsymbol{\epsilon}}^j$, with $\hat{\boldsymbol{\epsilon}}^j$ parallel to E^j and $\|\hat{\boldsymbol{\epsilon}}^j\|$ the characteristic function on Δ_E^j, the dual cell related to the E^j (cf. Figure 2.2). Our approximating subspaces are now defined by

$$\begin{aligned} \hat{V}_h &= \operatorname{span}(\hat{\boldsymbol{\epsilon}}^j) \subset \hat{V}, \\ \hat{W}_h &= \operatorname{span}(\hat{e}^i) \subset \hat{W}, \end{aligned} \tag{7.4}$$

and the primal mixed finite element discretization of (7.1) reads: find $(\hat{\boldsymbol{\sigma}}_h, \hat{u}_h) \in \hat{V}_h \times \hat{W}_h$, such that

$$\hat{a}(\hat{\boldsymbol{\sigma}}_h, \hat{\boldsymbol{\tau}}_h) + \hat{b}(\hat{\boldsymbol{\tau}}_h, \hat{u}_h) = 0, \qquad \forall \hat{\boldsymbol{\tau}}_h \in \hat{V}_h, \tag{7.5a}$$

$$\hat{b}(\hat{\boldsymbol{\sigma}}_h, \hat{t}_h) \qquad = (f, \hat{t}_h), \qquad \forall \hat{t}_h \in \hat{W}_h. \tag{7.5b}$$

Following Fuhrmann [2] we introduce a quadrature rule for integrals over the dual boxes Δ_E^k: let $g \in C^0(\Delta_E^k)$ be a continuous function, then

$$\int_{\Delta_E^k} g\, d\Omega \approx \frac{\operatorname{area}(\Delta_E^k)}{\operatorname{length}(E^k)} \int_{E^k} g\, ds. \tag{7.6}$$

Direct application of (7.6) to the integrals in (7.5a), with $u = (\psi, \Phi_n, \Phi_p)$, $\boldsymbol{\sigma} = (\mathbf{j}_\psi, \mathbf{j}_n, \mathbf{j}_p)$ and $A = (-\mu_\psi, +\mu_n \exp(+\psi), -\mu_p \exp(-\psi))$, respectively, yields the Scharfetter-Gummel discretization of the fluxes.

In order to interpret equation (7.5b) as a conservation law we introduce the dual boxes Δ_V^l that are related to the vertices $\mathbf{x}^l$ of cells (cf. Figure 7.1):

$\Delta_V^l = \cup\{\Omega^{i,\nu} \mid \mathbf{x}^l \in \overline{\Omega}^{i,\nu}\}$, with $\Omega^{i,\nu}$ the four quarter rectangles, parts of Ω^i, associated with these vertices. By taking $\hat{t}_h = \hat{e}^l$ and using (7.6) we see that the left hand side of (7.5b) indeed equals the net influx in the dual cell Δ_V^l. If the right hand side of (7.5b) is also approximated by quadrature then

$$(f,\hat{e}^l) \approx f^l \int_\Omega \hat{e}^l \, d\Omega = f^l \ \mathrm{area}(\Delta_V^l), \tag{7.7}$$

with $f^l = f(\mathbf{x}^l)$. We see that (7.5b) is the conservation law with respect to the dual box Δ_V^l, and we have regained a discretization that is equivalent to the usual vertex-centered box-scheme.

Finally, we discuss the treatment of the silicon/oxide interfaces that appear in MOS-devices. At these interfaces we have homogeneous Neumann boundary conditions for $\mathbf{j}_n$ and $\mathbf{j}_p$, whereas ψ and $\mathbf{j}_\psi \cdot \mathbf{n}$ are continuous, with $\mathbf{n}$ the normal unit vector at the interface (this means that we do not consider surface charges). We assume that the silicon/oxide interfaces are resolved by the edges E^j of the cells.

In the primal mixed finite element discretization the continuity of ψ follows from the choice of $\hat{V}_h$, but the continuity of the displacement current $\mathbf{j}_\psi$ does not hold. In the dual version the continuity of ψ does not hold, whereas the continuity of $\mathbf{j}_\psi$ in the direction normal to the interface is evident. For an edge E^j, with adjacent cells Ω^L and Ω^R that are in silicon and oxide region, we obtain for the dual mixed finite element discretization from (2.33a) and (2.34) (cf. (2.64))

$$j_\psi^j = -\frac{2\mu_\psi^L \mu_\psi^R}{\mu_\psi^L a^R + \mu_\psi^R a^L} h^j (\psi^R - \psi^L), \tag{7.8}$$

i.e. we take the harmonic average of the coefficients μ_ψ.

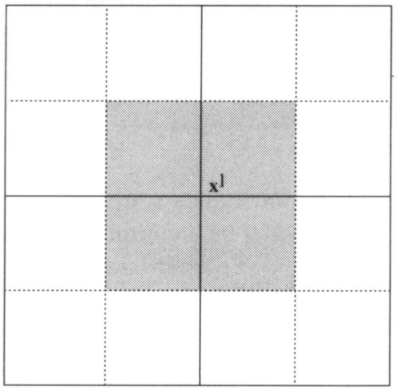


FIGURE 7.1. Dual box Δ_V^l related to the vertex $\mathbf{x}^l$.

7.3. Coarse grid correction

In this Section we compare the coarse grid correction stage of multigrid algorithms for the two different discretizations of the semiconductor equations: the cell-centered scheme and the vertex-centered scheme. Due to the strong nonlinearity and bad scaling of the equations the construction of the coarse grid correction operator is not trivial at all. We focus our discussion on two points: the stability and the proper diagonal scaling of the coarse grid operator. The choice of a smoothing operator is postponed until Section 7.4.

For both mixed finite element discretizations we obtain the fine grids by uniform refinement of the coarse grids: starting from a coarsest grid finer grids are constructed by cell-wise refinement, i.e. the cells Ω_H^l on the coarse grid are split into 4 equal, smaller ones. This means that the cell-centered discretization gives rise to a cell-centered multigrid method (cf. Figure 7.2), whereas the vertex-centered discretization brings about a vertex-centered multigrid method (cf. Figure 7.3). The important difference between these two multigrid methods is that in vertex-centered multigrid the nodes of the coarse grid coincide with nodes on the fine grid, which is not the case in the cell-centered multigrid.

The system of nonlinear equations on the fine grid can be written as

$$N_h(\overline{q}_h) = f_h. \tag{7.9}$$

The nonlinear coarse grid correction stage of a two-grid algorithm is then as usual given by ([1, 3])

$$N_H(\tilde{q}_H) = N_H(q_H) + \overline{R}_H(f_h - N_h(q_h)), \tag{7.10}$$

$$\tilde{q}_h = q_h + P_h(\tilde{q}_H) - P_h(q_H), \tag{7.11}$$

where N_H denotes the nonlinear coarse grid operator, P_h the (possibly nonlinear) prolongation operator for the solution, and $\overline{R}_H$ the restriction operator for the residual. As we only consider methods that solve the semiconductor equations simultaneously, it seems impossible to construct explicitly the coarse grid operator N_H as the Galerkin approximation of N_h, therefore we construct N_H by discretization on the coarse grid. As the problem is nonlinear, this implies that the choice of the initial iterand on the coarse grid q_H determines the entries of the Jacobian matrix of the coarse grid operator.

There are several approaches for the selection of q_H. One might simply take the last available iterate in the full multigrid process as in Chapter 5. This, however, is rather unsafe because q_H then depends on the history of the multigrid algorithm, and there is no guarantee that this iterate will remain in a sufficiently small neighborhood of the solution. Thus it may loose the properties that are required for a proper approximate solution. Other possibilities are to take $q_H = R_H q_h$, where R_H denotes a restriction operator for the solution (cf. Chapter 6), or to solve the problem on the coarse grid during the nested iteration, and to use that coarse grid solution as the initial iterate q_H each time that the grid is visited in the multigrid iteration. In this Chapter we only consider the last two approaches.

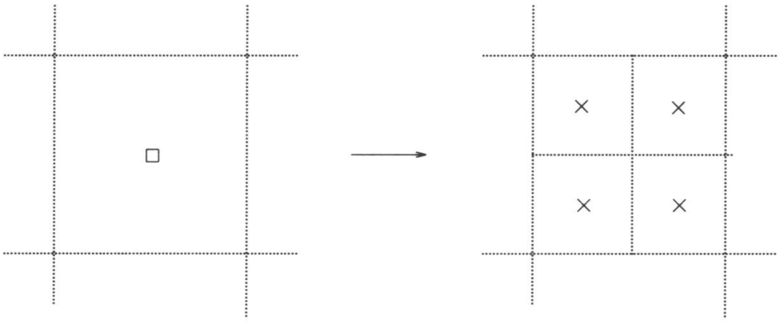


FIGURE 7.2. Coarse and fine grid cell-centres in cell-centered multigrid.

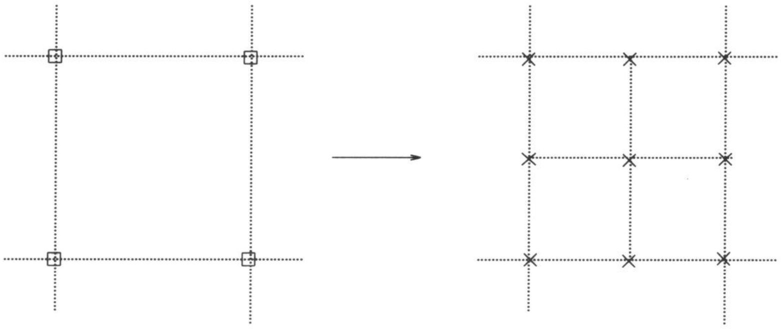


FIGURE 7.3. Coarse and fine grid nodes in vertex-centered multigrid.

A priori, it is not clear whether the problem on the coarse grid has a solution at all, or whether the coarse grid operator is stable. To get some insight in the last question we study the Jacobian matrices

$$J_H(\phi^i_H,\phi^l_H) = \frac{\partial(N_H(\phi_H))^i}{\partial\phi^l_H}, \qquad \phi = \psi, \phi_n, \phi_p, \tag{7.12}$$

that appear when global Gummel iteration is used to solve the coarse grid problem. For simplicity we assume square grids and neglect the recombination rate R.

For Poisson's equation this matrix is always strongly diagonally dominant (cf. [7]). For the continuity equation for holes related to the cell Ω^C_H, with nearest neighbors Ω^N_H, $N = n,e,s,w$, we have (cf. 6.13c)

$$J_H(\phi^C_{p,H},\phi^C_{p,H}) = +\mu_p\, e^{\phi^C_{p,H} - \psi^C_H} \sum_{N=n,e,s,w} B(\psi^N_H - \psi^C_H), \tag{7.13}$$

$$J_H(\phi^C_{p,H},\phi^N_{p,H}) = -\mu_p\, e^{\phi^C_{p,H} - \psi^C_H}\, B(\psi^N_H - \psi^C_H)\, e^{\phi^N_{p,H} - \phi^C_{p,H}}, \tag{7.14}$$

with

$$B(x) = \frac{x}{e^x - 1}, \tag{7.15}$$

the Bernoulli function. (The expression for the electron continuity equation is fully analogous.) It is known that the rowsum of the Jacobian matrix in Gummel's iteration is given by (cf. [6])

$$\sum_I J_H(\phi^C_{p,H},\phi^I_{p,H}) = H(j^n_{p,H} + j^e_{p,H} - j^s_{p,H} - j^w_{p,H}), \tag{7.16}$$

the summation is over all cells Ω^I_H in the grid (for the electron continuity equation a similar relation holds) and H denotes the mesh size. This means that for the solution of the discrete problem on the coarse grid with zero right hand side ($R = 0$), the matrix is weakly diagonally dominant, if there is a Dirichlet boundary value available (cf. [10]).

If we construct the coarse grid solution as some restriction of the fine grid solution, it is not guaranteed that the coarse grid Jacobian matrix for the continuity equations is still weakly diagonal. In the following Theorem we estimate the rowsum of the Jacobian matrix J_H that is scaled by the corresponding diagonal element.

THEOREM 7.1. *Let Ω^C_H be a cell with nearest neighbors Ω^N_H, $N = n,e,s,w$, then*

$$\frac{\sum_I J_H(\phi^C_{p,H},\phi^I_{p,H})}{J_H(\phi^C_{p,H},\phi^C_{p,H})} < 1$$

and

$$\left| \frac{\sum_I J_H(\phi^C_{p,H},\phi^I_{p,H})}{J_H(\phi^C_{p,H},\phi^C_{p,H})} \right| \leqslant \sum_{N=n,e,s,w} \left| 1 - e^{\phi^N_{p,H} - \phi^C_{p,H}} \right|.$$

Proof. Using (7.15), (7.16) and (6.10b) we find

$$\sum_I J_H(\phi_{p,H}^C, \phi_{p,H}^I) = \sum_{N=n,e,s,w} \mu_p e^{\phi_{p,H}^C - \psi_H^C} B(\psi_H^N - \psi_H^C)(1 - e^{\phi_{p,H}^N - \phi_{p,H}^C}),$$

so from (7.13) we obtain

$$\frac{\sum_I J_H(\phi_{p,H}^C, \phi_{p,H}^I)}{J_H(\phi_{p,H}^C, \phi_{p,H}^C)} = \frac{\sum_{N=n,e,s,w} B(\psi_H^N - \psi_H^C)(1 - e^{\phi_{p,H}^N - \phi_{p,H}^C})}{\sum_{N=n,e,s,w} B(\psi_H^N - \psi_H^C)}.$$

The fact that $B(x) > 0$ proves the theorem. □

Theorem 7.1 shows that if a restriction of the fine grid solution is used as initial iterate $q_H = q_H^R$ on the coarse grid, we may expect loss of diagonal dominance. However, if the solution of the coarse grid problem is fixed and used as initial iterate $q_H = q_H^F$ on the coarse grid, the matrices in Gummel's iteration are all weakly diagonally dominant.

For the semiconductor equations without any scaling, the residual of the continuity equations corresponds with the rate-of-change in the carrier concentrations. Without row scaling this means that the size of the residuals varies widely in magnitude throughout the domain. In the cell-centered multigrid algorithm obtained from the dual mixed finite element discretization it may also happen that the magnitude of the diagonal elements of the Jacobian matrices for a father cell differs by orders of magnitude from the corresponding elements for the four kid cells, especially if the transition between n- and p-region is not properly resolved on the coarse grid. In this case a small residual (after row scaling) on the fine grid may result in a large correction on the coarse grid. Therefore it may be necessary to apply a damping operator D_H for the restricted residual (cf. Chapter 5 and 6). The modified coarse grid equation then reads

$$N_H(\tilde{q}_H) = N_H(q_H) + D_H \overline{R}_H(f_h - N_h(q_h)). \tag{7.17}$$

This D_H is a diagonal matrix with entries in [0, 1] that are determined by comparing the diagonal elements of the coarse and fine grid Jacobian matrices: for every cell Ω_H^I, which is split into four cells Ω_h^i, we have (cf. 5.27))

$$D_H^I = \min\left(1, \frac{2\,|J_H(\phi_H^I, \phi_H^I)|}{\sup_{i=1,4} |J_h(\phi_h^i, \phi_h^i)|}\right), \qquad \phi = \psi, \phi_n, \phi_p.$$

In actual calculations we observe that damping is not necessary for Poisson's equation, and for the continuity equations the elements of D_H differ from 1 only in small parts of the domain (the transition regions), but there extremely small values ($< 10^{-10}$) for the diagonal elements appear. With this cell-centered multigrid algorithm good results were obtained for both one- and two-dimensional test problems (cf. [11] and the Chapters 5, 6).

To understand the necessity of damping in cell-centered multigrid we consider the Jacobian matrix again. If a transition between p- and n-region is not properly resolved on the coarse grid, the hole concentration $e^{\phi_p^c - \psi}$, that appears in (7.13), explains the large variations in magnitude of $J(\phi_p^c, \phi_p^c)$ between coarse and fine grids in the cell-centered multigrid method. A possible solution for this problem might be to construct q_H, by means of the L^2-projection of the variables (ψ, n, p); unfortunately, this choice may lead to ill-conditioning of the coarse grid matrix (cf. Section 6.3).

This scaling problem can be avoided by using a vertex-centered multigrid method: if we use injection for the restriction of the solution, the electron and hole concentrations are equal in the coinciding coarse and fine grid points. In this case, if we assume a kind of monotonicity for ψ, we can prove that the corresponding elements of the Jacobian matrix are of the same order of magnitude on the coarse and fine grid.

THEOREM 7.2. *Let Ω_h^c be a cell of the fine grid with nearest neighbors Ω_h^l, $l = n,e,s,w$, and let Ω_H^C be the corresponding cell of the coarse grid, with nearest neighbors Ω_H^L, $L = N,E,S,W$ (see Figure 7.4). If vertex-centered multigrid is used, with injection for the restriction of the solution , and if, furthermore,*

$$\min_{l=n,e,s,w} \psi_h^l \leqslant \psi_h^c \leqslant \max_{l=n,e,s,w} \psi_h^l, \tag{7.18a}$$

and

$$\min\,(\psi_h^c, \psi_H^L) \leqslant \psi_h^l \leqslant \max\,(\psi_h^c, \psi_H^L), \tag{7.18b}$$

for $(l,L) = ((n,N), (e,E), (s,S), (w,W))$, then we have for the ratio of the corresponding diagonal elements of the fine and coarse grid Jacobian matrices

$$\frac{J_H(\phi_{p,H}^C, \phi_{p,H}^C)}{J_h(\phi_{p,h}^c, \phi_{p,h}^c)} \geqslant \frac{1}{4}.$$

PROOF. Suppose that $(\psi_h^l - \psi_h^c)$ is minimal for some $l = k$. From (7.18a) it follows that $\psi_h^k \leqslant \psi_h^c$, so from (7.18b) we conclude $\psi_H^K \leqslant \psi_h^k$. Using the fact that $B(x)$ is monotonically decreasing we obtain

$$\begin{aligned}\frac{J_H(\phi_{p,H}^C, \phi_{p,H}^C)}{J_h(\phi_{p,h}^c, \phi_{p,h}^c)} &= \frac{\sum\limits_{L=N,E,S,W} \mu_p e^{\phi_{p,H}^C - \psi_H^C} B(\psi_H^L - \psi_H^C)}{\sum\limits_{l=n,e,s,w} \mu_p e^{\phi_{p,h}^c - \psi_h^c} B(\psi_h^l - \psi_h^c)} \\ &= \frac{\sum\limits_{L=N,E,S,W} B(\psi_H^L - \psi_H^C)}{\sum\limits_{l=n,e,s,w} B(\psi_h^l - \psi_h^c)} \\ &\geqslant \frac{B(\psi_H^K - \psi_H^C)}{4\,B(\psi_h^k - \psi_h^c)} \geqslant \frac{1}{4}. \quad \square\end{aligned}$$

Theorem 7.2 shows that in a vertex-centered multigrid method local damping of the restricted residual is not necessary, provided that injection is used for the restriction of the residual $\overline{R}_H$. The use of e.g. full-weighting brings back the scaling problem. Moreover, if we assume that in vertex-centered multigrid the concentrations n and p on the coarse grid are a good point-wise approximation of the concentrations on the fine grid, we expect that the coarse grid operator, with q_H^F as the initial iterate, is both stable and properly scaled.

This brings us to the point of the choice for the grid transfer operators P_h and $\overline{R}_H$ for the vertex-centered multigrid method. We have just shown that it is attractive to use simple injection $\overline{R}_H^{I}$ for the restriction of the residual. For the prolongation of the solution in vertex-centered multigrid we define a nonlinear interpolation operator P_h^{NL}. For this interpolation, injection is used for the fine grid points, that also appear as coarse grid points (see Figure 7.3). Next we use the one-dimensional, current-conserving interpolation proposed by Hemker [4], to obtain values at the midpoints of the edges. Finally, we locally solve the equations at the middle of the cell, using the interpolated values at the midpoints of the edges as boundary conditions. In this way we construct our nonlinear interpolation for the two-dimensional case.

In the dual mixed finite element discretization the approximating subspaces are nested, $V_H \subset V_h$ and $W_H \subset W_h$, so a natural set of grid transfer operators is available for the cell-centered multigrid algorithm. The prolongation P_h for the scalar quantities u_H is a piecewise constant interpolation, whereas the restriction of the residual is its transpose; these operators are denoted by P_h^C and $\overline{R}_H^C$, respectively. As before we have to apply the damping operator D_H in cell-centered multigrid.

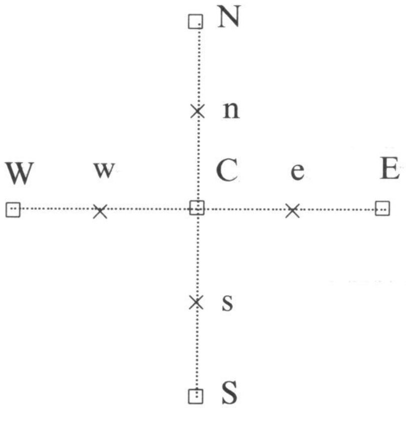


FIGURE 7.4. Numbering of nodes in vertex-centered multigrid.

7.4. Two-grid analysis

In this Section we carry out two-grid analyses for the various multigrid algorithms proposed in the previous Section. This is done because it is well known that the grid transfer operators, proposed for the two multigrid algorithms, are too inaccurate to be used in multigrid algorithms for solving second order differential equations (cf. ([1, 5]). Our strategy to circumvent this possible source of problems is to take smoothing operators that can be used in combination with the inaccurate grid transfer operators, as we did before for Poisson's equation on a square grid in Chapter 4.

We consider the anisotropic model problem

$$L\,u = -(A\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2})u = f, \qquad A > 0, \tag{7.19}$$

on the infinite domain $\Omega = \mathbb{R}^2$. This model problem can be considered as Poisson's equation on a rectangular, not necessarily square, grid. For both the cell-centered and the vertex-centered discretization the Fourier transform $\hat{L}_h : T_h \rightarrow \mathbb{C}$, $T_h = [-\pi,\pi)^2$, of the discretized operator L_h is given by

$$\hat{L}_h(\theta_x, \theta_y) = \frac{4}{h^2}(A\sin^2\frac{\theta_x}{2} + \sin^2\frac{\theta_y}{2}). \tag{7.20}$$

We introduce a matrix notation for T_h as in Section 4.3: every $\boldsymbol{\theta} \in T_h$ is written as a 4-vector on T_H with entries $(\boldsymbol{\theta} + \pi\mathbf{p})$, where $\boldsymbol{\theta} \in T_H = [-\frac{\pi}{2},\frac{\pi}{2})^2$ and $\mathbf{p} \in \{(0,0), (1,0), (0,1), (1,1)\}$. The accuracy of a restriction operator $\bar{R}_H$ is measured by the high frequency order m_H, i.e. the largest number m_H for which

$$\hat{\bar{R}}_H(\boldsymbol{\theta} + \pi\mathbf{p}) = \mathcal{O}(|\boldsymbol{\theta}|^{m_H}), \qquad \text{for } |\boldsymbol{\theta}| \rightarrow 0,\ \mathbf{p} \neq (0,0).$$

The high frequency order should at least be equal to the order of the differential equation being solved in order to avoid blow-up of high frequency error components in the coarse grid correction (cf. [1, 5]).

The two-grid error amplification matrix $M_h^{\nu_1,\nu_2}$ for a two-grid algorithm is defined by

$$M_h^{\nu_1,\nu_2} = S_h^{\nu_2}(I_h - P_h(L_H)^{-1}\bar{R}_H L_h)S_h^{\nu_1}, \tag{7.21}$$

where I_h denotes the identity operator and ν_1,ν_2 the number of pre- and post relaxation sweeps, respectively. Using the techniques developed in Chapter 4 we find for the cell-centered multigrid method, described in Section 7.3, that the Fourier transform $\hat{\mathbf{M}}_h^{0,0}$ of the coarse grid correction operator $M_h^{0,0}$ is in matrix representation given by

$$(\hat{\mathbf{M}}_h^{0,0})_{i,j} = \delta_{ij} - \frac{4 f_i f_j g_j}{A\sin^2\theta_x + \sin^2\theta_y}, \qquad i,j = 1, \cdots, 4, \tag{7.22a}$$

with

$$\begin{aligned}
f_1 &= \cos\frac{\theta_x}{2}\cos\frac{\theta_y}{2}, & g_1 &= A\sin^2\frac{\theta_x}{2} + \sin^2\frac{\theta_y}{2},\\
f_2 &= \sin\frac{\theta_x}{2}\cos\frac{\theta_y}{2}, & g_2 &= A\cos^2\frac{\theta_x}{2} + \sin^2\frac{\theta_y}{2},\\
f_3 &= \cos\frac{\theta_x}{2}\sin\frac{\theta_y}{2}, & g_3 &= A\sin^2\frac{\theta_x}{2} + \cos^2\frac{\theta_y}{2},\\
f_4 &= \sin\frac{\theta_x}{2}\sin\frac{\theta_y}{2}, & g_4 &= A\cos^2\frac{\theta_x}{2} + \cos^2\frac{\theta_y}{2}.
\end{aligned} \tag{7.22b}$$

From (7.22) we see that initial high frequency error modes in the neighborhood of $(0,\pi)$ and $(\pi,0)$ are blown up by the coarse grid correction, due to the inaccuracy of the restriction operator $\bar{R}_H^C$. The Fourier representation of $\bar{R}_H^C$ is

$$\hat{\bar{\mathbf{R}}}_H^C = \left[f_1 \quad f_2 \quad f_3 \quad f_4 \right],$$

so its high frequency order m_H is only 1, whereas for a second order differential equation it should be 2.

The same problem occurs in vertex-centered multigrid when straight injection is used for the restriction and bilinear interpolation for the prolongation. In this case the coarse grid correction matrix is given by

$$(\hat{\mathbf{M}}_h^{0,0})_{i,j} = \delta_{ij} - \frac{4\tilde{f}_i g_j}{A\sin^2\theta_x + \sin^2\theta_y}, \qquad i,j = 1, \cdots, 4, \tag{7.23a}$$

with

$$\begin{aligned}
\tilde{f}_1 &= \cos^2\frac{\theta_x}{2}\cos^2\frac{\theta_y}{2},\\
\tilde{f}_2 &= \sin^2\frac{\theta_x}{2}\cos^2\frac{\theta_y}{2},\\
\tilde{f}_3 &= \cos^2\frac{\theta_x}{2}\sin^2\frac{\theta_y}{2},\\
\tilde{f}_4 &= \sin^2\frac{\theta_x}{2}\sin^2\frac{\theta_y}{2},
\end{aligned} \tag{7.23b}$$

and g_j as in (7.22b). Now all high frequencies (θ_x,θ_y), with $\theta_x \to \pi$ or $\theta_y \to \pi$, are blown up by the coarse grid correction, because the high frequency order m_H of $\bar{R}_H^I$ is zero.

The obvious remedy seems to be the use of more accurate restriction operators, but these have larger stencils which is undesirable for the semiconductor equations (see Section 7.3). Therefore we look for relaxation operators that effectively eliminate the dangerous high frequency error modes. As an example we consider the damped Jacobi relaxation.

Relaxation	$A=1$	$A\neq 1$	Coupling
damped Jacobi (0.5)	(π,π)	(π,π)	-
point Gauss-Seidel	$(0,\pi)$, $(\pi,0)$	-	-
line Gauss-Seidel	-	-	-
red-black	$(\pi,0)$, $(0,\pi)$, (π,π)	(π,π)	$(\pi,0) \rightleftarrows (0,\pi)$
x-line zebra	$(\pi,0)$	$(\pi,0)$	$(\pi,\pi) \rightleftarrows (0,\pi)$
y-line zebra	$(0,\pi)$	$(0,\pi)$	$(\pi,\pi) \rightleftarrows (\pi,0)$

TABLE 7.1. Elimination and coupling of high frequency error modes for the model problem (7.19) by some standard smoothing operators.

A	μ^{yJx}
10^{-3}	0.125
10^{-2}	0.121
10^{-1}	0.095
10^{+0}	0.025
10^{+1}	0.095
10^{+2}	0.121
10^{+3}	0.125

TABLE 7.2. Smoothing factor μ^{yJx} of zebra-JOR relaxation for the model problem.

	Cell-centered MG		Vertex-centered MG				
	$\overline{R}_H$		$\overline{R}_H^I$		$\overline{R}_H^Z$		$\overline{R}_H^Z$
A	λ_ρ^1	$\lambda_S^{1,0}$	λ_ρ^1	$\lambda_S^{1,0}$	λ_ρ^1	$\lambda_S^{1,0}$	$\tilde{\lambda}_\rho^1$
10^{-3}	0.133	0.706	1.000	1.996	0.124	0.543	0.271
10^{-2}	0.144	0.693	1.000	1.962	0.116	0.534	0.260
10^{-1}	0.157	0.597	1.000	1.722	0.061	0.456	0.179
10^{+0}	0.202	0.357	1.000	1.423	0.111	0.212	0.111
10^{+1}	0.157	0.431	1.000	1.414	0.694	0.982	0.179
10^{+2}	0.144	0.490	1.000	1.414	0.961	1.359	0.260
10^{+3}	0.133	0.499	1.000	1.414	0.996	1.408	0.271

TABLE 7.3. Spectral norm $\lambda_S^{1,0}$ and radius λ_ρ^1 of two-grid error amplification matrix with a single zebra-JOR pre-relaxation sweep.

EXAMPLE 7.1. The Fourier representation of Jacobi relaxation with damping parameter α ($\alpha = 1$ means no damping, and $\alpha = 0$ total damping) for problem (7.19) is

$$\hat{S}_h^{JOR}(\theta_x, \theta_y) = 1 - 2\alpha \frac{A \sin^2 \frac{\theta_x}{2} + \sin^2 \frac{\theta_y}{2}}{A + 1},$$

so the high frequency (π,π) is eliminated for $\alpha = \frac{1}{2}$.

Table 7.1 shows which of the high frequencies are eliminated by some standard smoothing operators for problem (7.19). For Poisson's equation discretized on a square grid ($A = 1$), we observe that point Gauss-Seidel can be used as the smoother in our cell-centered multigrid algorithm, as was found earlier in Chapter 4. For the more general case ($A \neq 1$), we see that of the smoothers that do not mix frequencies only damped Jacobi relaxation, with damping parameter 0.5, eliminates one of the highest frequencies: $(\theta_x,\theta_y) = (\pi,\pi)$. On the other hand red-black and zebra relaxation are more powerful smoothers, but they have the disadvantage of coupling frequencies; this is also shown in Table 7.1. Due to the coupling between high frequencies the alternating zebra relaxation, i.e. the combination of x-line zebra and y-line zebra relaxation, does not eliminate both the frequencies $(\pi,0)$ and $(0,\pi)$.

In order to eliminate all the highest frequencies we introduce the zebra-JOR relaxation S_h^{yJx} that consist of the sequence of a x-line zebra sweep, a damped Jacobi sweep (with damping factor 0.5) and a y-line zebra sweep. After the x-line zebra sweep and the JOR sweep both the high frequencies $(\pi,0)$ and (π,π) are eliminated; in the final y-line zebra sweep the high frequency $(0,\pi)$ is eliminated, while the two others are not reintroduced again. We notice that in the Jacobi sweep only half of the points need to be relaxed, as it follows the x-line zebra sweep.

As usual for relaxation operators that mix frequencies, we define the smoothing factor μ^{yJx} of zebra-JOR relaxation by

$$\mu^{yJx} = \sup_{\boldsymbol{\theta} \in T_H} \rho(\hat{\mathbf{Q}}\hat{\mathbf{S}}_h^{yJx}(\boldsymbol{\theta})),$$

where $\rho(\cdot)$ denotes the spectral radius, $\hat{\mathbf{S}}_h^{yJx}$ the Fourier transform of the iteration matrix of zebra-JOR relaxation and $\hat{\mathbf{Q}}$ the operator that annihilates all low frequencies

$$\hat{\mathbf{Q}} = \begin{bmatrix} 0 & & & \\ & 1 & & \\ & & 1 & \\ & & & 1 \end{bmatrix}.$$

Table 7.2 shows μ^{yJx} for different values of A; we conclude that zebra-JOR is a robust smoother.

As the last relaxation sweep of zebra-JOR is a zebra sweep, we can introduce a more accurate restriction operator $\overline{R}_H^Z$ for the vertex-centered multigrid algorithm, that also uses the residual from only one point, provided that in the last partial relaxation sweep of S_h^{yJx} (on the fine grid) the lines are relaxed that do not contain coarse grid points; its stencil is given by

$$\overline{R}_H^Z \simeq \frac{1}{8}\begin{bmatrix}1 & 0 & 1\\ 0 & 4 & 0\\ 1 & 0 & 1\end{bmatrix}. \tag{7.24}$$

The use of $\overline{R}_H^Z$ in combination with zebra relaxation is equivalent to the use of half weighting in combination with red-black relaxation (cf. [9]). The Fourier representation of $\overline{R}_H^Z$ is

$$\hat{\mathbf{R}}_H^Z = \frac{1}{2}\left[1+\cos\theta_x\cos\theta_y \quad 1-\cos\theta_x\cos\theta_y \quad 1-\cos\theta_x\cos\theta_y \quad 1+\cos\theta_x\cos\theta_y\right],$$

so its high frequency order m_H is still zero, as with injection, but now there is no aliasing of the high frequencies $(\pi,0)$ and $(0,\pi)$ with $(0,0)$.

Besides the spectral radius $\rho(\cdot)$ of a matrix, we study the spectral norm $\|\cdot\|_S$ of the amplification operator. The spectral norm $\|\hat{\mathbf{M}}_h^{\nu_1,\nu_2}\|_S$ of the two-grid error amplification matrix indicates what happens in a single two-grid cycle, whereas the spectral norm $\rho(\hat{\mathbf{M}}_h^{\nu_1,\nu_2})$ describes the convergence behavior after many cycles. We are interested in the supremum of these quantities with respect to $\boldsymbol{\theta}$ (the worst case behavior), so we define

$$\lambda_S^{\nu_1,\nu_2} = \sup_{\boldsymbol{\theta}\in T_H} \|\hat{\mathbf{M}}_h^{\nu_1,\nu_2}\|_S,$$

$$\lambda_\rho^{\nu} = \sup_{\boldsymbol{\theta}\in T_H} \rho(\hat{\mathbf{M}}_h^{\nu_1,\nu_2}),$$

with $\nu = \nu_1 + \nu_2$. Table 7.3 shows the values of $\lambda_S^{1,0}$ and λ_ρ^1 for three different multigrid algorithms: cell-centered multigrid, vertex-centered multigrid with straight injection $\overline{R}_H^I$ and vertex-centered multigrid with the more accurate restriction $\overline{R}_H^Z$. In all cases $\lambda_S^{1,0}$ is bounded so initial high frequency error modes are not blown up in a single two-grid cycle. The vertex-centered multigrid algorithm with straight injection $\overline{R}_H^I$ fails to converge in all cases ($\lambda_\rho^1 = 1$), whereas the use of $\overline{R}_H^Z$ yields an algorithm that does not converge for $A \gg 1$. The problem is that the low frequency (0,0) is not removed in the two-grid cycle, in fact we have

$$\lim_{|\boldsymbol{\theta}|\to 0} (\hat{\mathbf{M}}_h^{1,0})_{1,1} = \frac{-A^2}{(2+A)^2}.$$

When we interchange the x-zebra and the y-zebra sweep in the zebra-JOR relaxation, we get a two-grid algorithm that fails to converge for $A \ll 1$; in this case we have

$$\lim_{|\boldsymbol{\theta}|\to 0} (\hat{\mathbf{M}}_h^{1,0})_{1,1} = \frac{-1}{(1+2A)^2}.$$

Therefore we alternately use S_h^{xJy} and S_h^{yJx} in a series of two-grid cycles; the two-level convergence factor $\tilde{\lambda}_\rho^1$ for the two-grid cycle with a single relaxation sweep is now defined by

$$\tilde{\lambda}_{\rho}^{1} = (\sup_{\boldsymbol{\theta} \in T_H} \rho(\hat{\mathbf{M}}_h^{0,0}\, \hat{\mathbf{S}}_h^{xJy}\, \hat{\mathbf{M}}_h^{0,0}\, \hat{\mathbf{S}}_h^{yJx}))^{\frac{1}{2}}.$$

The last column of Table 7.3 shows $\tilde{\lambda}_{\rho}^{1}$; the two-grid algorithm now converges for all values of A.

7.5. Numerical experiments

To compare the efficiency and the robustness of the multigrid algorithms that we have developed, we consider two test problems: a MOS-transistor and an LDDMOS-transistor.

In order to represent the geometry of these devices properly on the coarsest grid, we use non-uniform grids. The problem on the coarsest grid is solved by a combination of relaxation sweeps and Newton steps as described in Section 5.5.

The continuation process to find a proper initial estimate is applied on the coarsest grid only. We start by solving the thermal equilibrium case (no applied voltages). Then we change the applied voltages and solve the problem on the coarsest grid using the previously obtained solution as initial iterate; due to the robustness of the solution procedure we are able to take large steps. The coarse grid solution is interpolated to a next finer grid, and multigrid is used to solve the problem on the fine grid (nested iteration).

In our numerical experiments we applied both cell-centered and vertex-centered multigrid with two possibilities for constructing the coarse grid initial iterate q_H as described in Section 7.3: either we 'freeze' the coarse grid solution q_H^F, or we use a restriction $q_H^R = R_H q_h$ of the fine grid solution. In vertex-centered multigrid we take injection for R_H, and in cell-centered multigrid we use the L^2-projection of the variables (ψ, ϕ_n, ϕ_p), which works successfully in the case of a bipolar transistor problem (cf. Section 6.6). We only consider W-cycles, as it appeared that V-cycles are not sufficiently robust for the semiconductor problem (cf. Section 5.7). In all cases a single zebra-JOR sweep is used both for pre- and post smoothing; in vertex-centered multigrid the x-zebra and y-zebra sweeps are interchanged in the subsequent V-cycles that make up the W-cycle as indicated in Section 7.4. For details about the non-linear relaxation operators we refer to the Sections 5.2 and 6.4.

To estimate the convergence rate of the multigrid algorithms we use the average reduction factor ρ (cf. (6.25)),

$$\rho = \left[\frac{d^{(10)}}{d^{(0)}} \right]^{\frac{1}{10}}, \tag{7.25}$$

where $d^{(i)}$ denotes the maximum of the scaled residual after i FAS-sweeps. The residual is scaled point-wise, by means of the diagonal 3×3 blocks of the Jacobian matrix: thus the scaled residual corresponds with corrections that would occur in a point-wise collective Jacobi relaxation. The maximum of this scaled residual is taken over the grid and over the three variables (ψ, ϕ_n, ϕ_p).

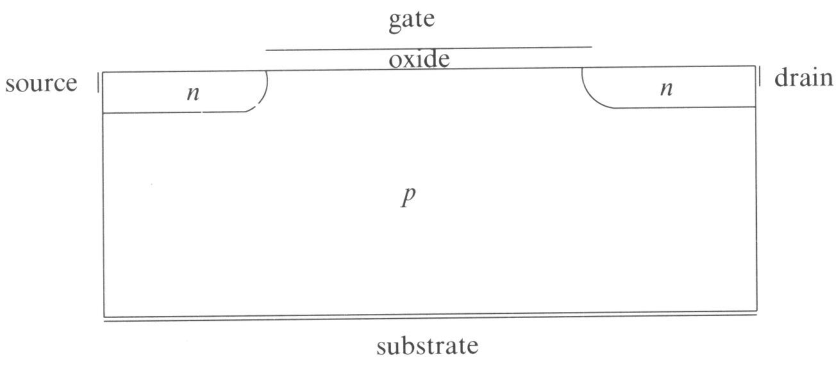


FIGURE 7.5. Configuration of MOS-transistor.

	q_H	0.0	1.0	2.0	3.0	4.0	5.0
Cell-centered MG	q_H^R	0.18	0.20	0.23	0.25	0.27	0.31
Cell-centered MG	q_H^F	0.19	0.20	0.35	0.37	0.40	0.41
Vertex-centered MG	q_H^R	0.15	0.15	0.17	0.17	0.16	0.16
Vertex-centered MG	q_H^F	0.16	0.16	0.16	0.15	0.16	0.19

TABLE 7.4. Average convergence factor ρ for different gate voltages V_g on 64×80 grid (MOS-transistor).

7.5.1. MOS-transistor

Figure 7.5 gives a schematic view of the geometry and the doping profile of the MOS-transistor. The length of the device is 4.0 μm, the width is 1.5 μm and the oxide-layer is 0.05 μm thick. The recombination rate R is given by the Shockley-Read-Hall model (6.24) with carrier lifetimes $\tau_n = \tau_p = 10^{-6}$ s. More details about the device can be found in Appendix B. The applied voltages at the source, drain and substrate are kept constant at $V_S = 0.0$ V, $V_d = 0.1$ V and$V_s = 0.0$ V, respectively. During the simulation the applied voltage at the gate is raised from $V_g = 0.0$ V to $V_g = 5.0$ V in steps of 1.0 V. Table 7.4 show the average residual factor ρ for the different multigrid algorithms on a 64×80 grid, the coarsest grid is an 8×10 grid, so 3 levels of uniform refinement are used. In all cases the multigrid algorithms converge rapidly; we observe that the vertex-centered algorithms are more efficient than the cell-centered algorithms.

7.5.2. LDDMOS-transistor

A harder problem is the LDDMOS-transistor of which a plot is shown in Figure 7.6; again a precise description of the device is found in the Appendix B. We solve this problem only for the electrons, and assume that ϕ_p is piecewise constant; the recombination rate R is zero. We keep the applied voltages at the gate, substrate and source constant at $V_g = 2.0$ V, $V_s = 0.0$ V and $V_S = 0.0$ V, respectively, while the drain voltage V_d is raised from 0.0 V to 5.0 V in steps of 1.0 V.

The coarsest grid used, a non-uniform 10×10 grid, is shown in Figure 7.7. The first grid-line in the silicon region lies on the coarsest grid 0.01 μm underneath the silicon/oxide interface. Table 7.5 shows the average reduction factor ρ for the different multigrid algorithms. For this test problem the relaxation procedure sometimes fails on one of the coarse grids. If this happens we do not use the correction calculated on that grid, and return to the finer grid immediately; in Table 7.5 these cases are indicated by an asterisk. Notice that for vertex-centered multigrid with a frozen solution q_H^F on the coarse grids, we are able to use all grids; as shown in Section 7.3 in this case the coarse grid problems are properly scaled and stable. In Figure 7.8 we show the convergence behavior for much finer grids (the finest grid contains 320×320 cells) of this multigrid algorithm for $V_d = 5.0$ V; we observe that the convergence behavior is grid independent indeed. Finally, we give a rough estimate for the execution times of the vertex-centered multigrid algorithm in Table 7.6; these results are obtained on a SUN SPARC station 1 for a non optimized PASCAL-code. As the convergence behavior is grid independent and the amount of work is proportional to the number of grid-points we conclude that multigrid has optimal complexity also for this test problem.

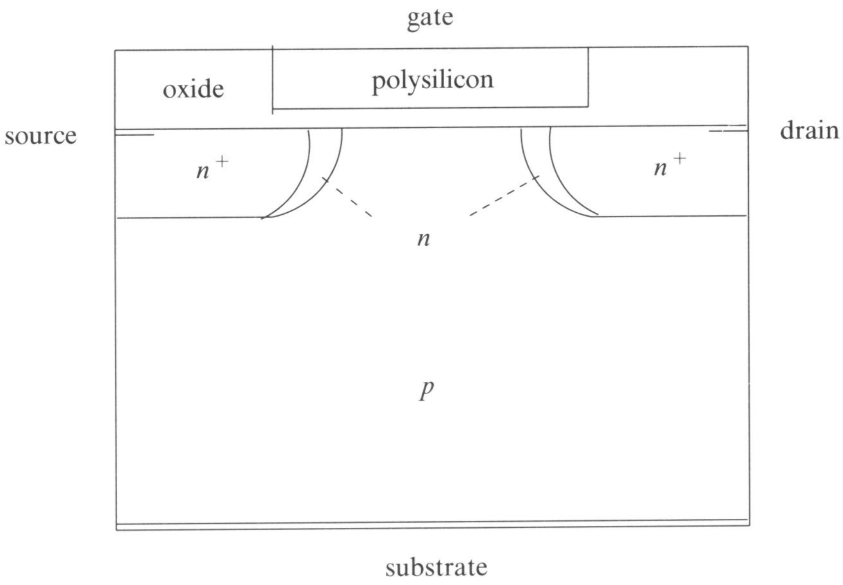


FIGURE 7.6. Configuration of LDDMOS-transistor.

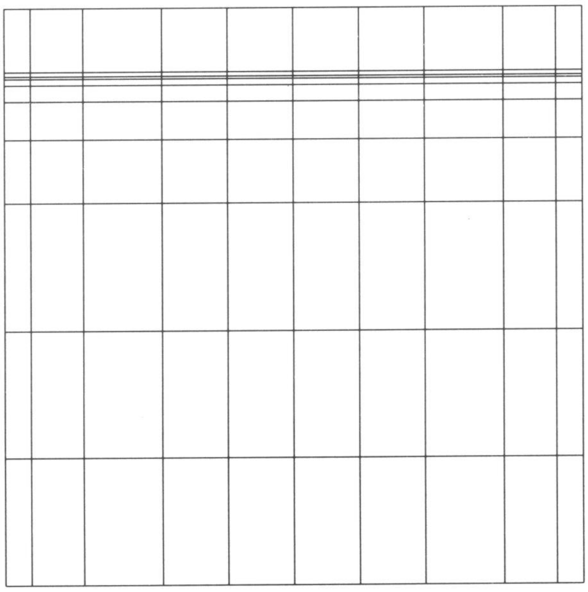


FIGURE 7.7. Coarsest grid for LDDMOS-transistor.

	q_H	0.0	1.0	2.0	3.0	4.0	5.0
Cell-centered MG	q_H^R	0.20	0.52	0.63	0.74^*	0.65^*	0.74^*
Cell-centered MG	q_H^F	0.23	0.47	0.44	0.74^*	0.65^*	0.74^*
Vertex-centered MG	q_H^R	0.13	0.21	0.40^*	0.80^*	0.90^*	0.91^*
Vertex-centered MG	q_H^F	0.14	0.34	0.24	0.23	0.21	0.22

TABLE 7.5. Average convergence factor ρ for different drain voltages V_d on 80×80 grid (LDDMOS-transistor).

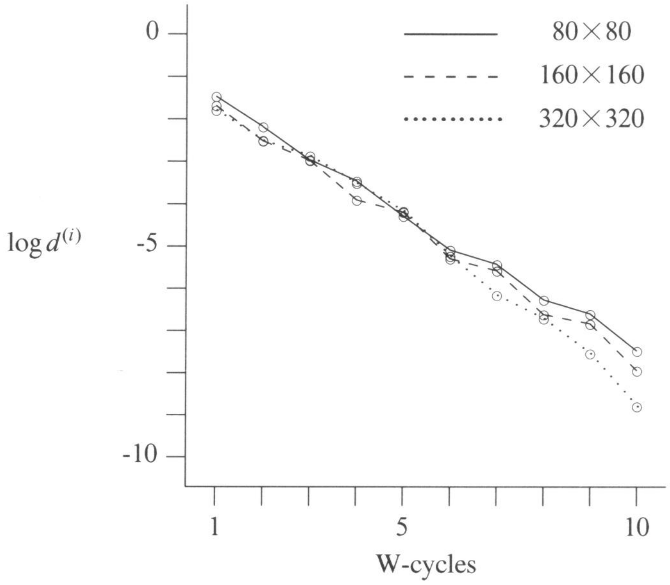


FIGURE 7.8. Convergence history of vertex-centered multigrid with $q_H = q_H^F$ for LDDMOS-transistor, $V_d = 5.0\,\mathrm{V}$.

grid	time in s
40×40	33
80×80	129
160×160	508
320×320	1959

TABLE 7.6. Estimation of time per W-cycle on a SPARC station 1.

7.6. Concluding remarks

We have introduced and compared two different mixed finite element discretizations of the stationary semiconductor equations, both equivalent to finite volume discretizations. To solve the systems of nonlinear equations obtained after discretization, we have analyzed a cell-centered and a vertex-centered multigrid algorithm. By studying the Jacobian matrices of the coarse grid problems it appears that vertex-centered multigrid avoids the scaling problem that is inherent to the cell-centered multigrid method. Moreover, it is shown that the use of a restriction of the fine grid solution as initial iterate on the coarse grid may lead to ill-conditioned coarse grid problems; it is better to calculate the solutions on the coarse grids during the nested iteration, and to use these solutions as starting iterates on the coarse grids during the multigrid cycling. In both cell-centered and vertex-centered multigrid we use inaccurate grid transfer operators for the restriction of the residual; by Fourier analysis it is shown that the choice of a suitable relaxation operator may lead to a well-behaved two-grid algorithm for an anisotropic model problem.

These findings are confirmed by our numerical experiments. It appears that the vertex-centered multigrid method is more robust than the cell-centered multigrid method. Moreover, the vertex-centered multigrid method has a fast and grid independent convergence behavior for practical test problems.

In this Thesis we assumed very simple models for the recombination-generation term R and the carrier mobilities μ_n, μ_p. Here, we like to make some remarks about possible problems that might arise in the multigrid algorithm, if more sophisticated models are used.

The recombination-generation term R is usually modeled as the sum of Shockley-Read-Hall and Auger recombination, and a generation term modeling impact ionization. The inclusion of Auger recombination is straightforward, and no difficulties are expected. However, it is still an open question how to deal with impact ionization in multigrid . The impact ionization term depends exponentially on the electric field (cf. [8]) and it is not clear whether it is necessary, or possible, to resolve this electric field accurately on the coarser grids. On the other hand, the modeling of impact ionization is also problematic in conventional device simulation programs.

There are several models for the mobilities μ_n, μ_p (cf. [8]); these models imply a nonlinear dependence of μ_n and μ_p on the solution (ψ, n, p). However, it appears that μ_n and μ_p are smoothly varying functions, so no serious problems are expected with regard to the multigrid algorithm.

We complete this Thesis by concluding that it is attractive indeed to use multigrid for semiconductor device simulation problems.

References

1. A. Brandt (1982). Guide to multigrid development, in *Multigrid Methods*, 220-312, ed. W. Hackbusch and U. Trottenberg, Springer-Verlag, Lecture Notes in Mathematics 960.
2. J. Fuhrmann (1990). An interpretation of the Scharfetter-Gummel scheme as a mixed finite element discretization, in *Fourth Multigrid Seminar*, 1-7, ed. G. Telschow, Karl-Weierstrass-Institut fur Mathematik, Berlin.
3. W. Hackbusch (1985). *Multigrid Methods and Applications*, Springer-Verlag, Berlin, Series in Computational Mathematics 4.
4. P.W. Hemker (1990). A nonlinear multigrid method for one-dimensional semiconductor device simulation: results for the diode, *J.Comp.Appl.Math.*, 30, 117-126.
5. P.W. Hemker (1990). On the order of prolongations and restrictions in multigrid procedures, *J.Appl.Math.*, 32, 423-429.
6. P.W. Hemker and J. Molenaar (1991). An adaptive multigrid approach for the solution of the 2D semiconductor equations, in *International Series of Numerical Mathematics 98*, ed. W. Hackbusch and U. Trottenberg, Birkhauser Verlag, Basel.
7. S.J. Polak, C. den Heijer, W.H.A. Schilders, and P. Markowich (1987). Semiconductor device modelling from the numerical point of view, *Int.J.Num.Meth.Engng.*, 24, 763-838.
8. S. Selberherr (1984). *Analysis and Simulation of Semiconductor Devices*, Springer-Verlag, Wien.
9. K. Stüben and U. Trottenberg (1982). Multigrid methods: fundamental algorithms, model problem analysis and applications, in *Multigrid Methods*, 220-312, ed. W. Hackbusch and U. Trottenberg, Springer-Verlag, Lecture Notes in Mathematics 960.
10. D.M. Young (1971). *Iterative solution of large linear systems*, Academic Press, New York.
11. P.M. De Zeeuw (1991). Nonlinear multigrid applied to a 1D stationary semiconductor model, *SIAM J.Sci.Stat.Comput.*, To appear.

Appendix A

Numerical evaluation of some functions

In this Appendix we give the numerical evaluation of some functions.

```
$\mathrm{logPlus1}\,(x) = \log(1 + x)$
{ use: correction transformation (cf. (5.18) and (5.33)) }
if $|x| < 0.001$ then $\mathrm{logPlus1} = x(1-x(\frac{1}{2}-x(\frac{1}{3}-x(\frac{1}{4}-x\frac{1}{5}))))$
else
        $\epsilon = 10^{-7}$
        if $(1+x) > \epsilon$ then    $\mathrm{logPlus1} = \log(1+x)$
        else                          $\mathrm{logPlus1} = 2\log(\epsilon)-\log(2\epsilon-x-1)$
```

```
$\mathrm{Cexp}(x) = \frac{1}{x} + \frac{1}{1-e^x}$
{ use: calculation Jacobian maxtrix elements (cf. (6.13)) }
if $(x < -30.0)$       then $\mathrm{Cexp} = 1 + \frac{1}{x}$
elseif $(x > 30.0)$    then $\mathrm{Cexp} = \frac{1}{x}$
elseif $(|x| > 0.2)$   then $\mathrm{Cexp} = \frac{1}{x} + \frac{1}{1-e^x}$
else
        $t = ((((\frac{1}{5040}x + \frac{1}{720})x + \frac{1}{120})x + \frac{1}{24})x + \frac{1}{6})x + \frac{1}{2}$
        $\mathrm{Cexp} = \frac{t}{1+xt}$
```

$$\mathrm{Zexp}(x) = \frac{e^x - 1}{x}$$

{ use: calculation Bexp (x) }

```
if $|x| < 0.01$ then $\mathrm{Zexp} = ((((\frac{1}{720}x + \frac{1}{120})x + \frac{1}{24})x + \frac{1}{6})x + \frac{1}{2})x + 1$
elseif $|x| < 30$ then $\mathrm{Zexp} = \frac{e^x - 1}{x}$
elseif $x < 0$ then $\mathrm{Zexp} = -\frac{1}{x}$
else $\mathrm{Zexp} = \frac{e^x}{x}$
```

$$\mathrm{Qexp}(x, y) = \frac{1 - e^x}{1 - e^y}$$

{ use: calculation Lagrange multipliers (cf. (6.15)) }

```
if $x \geqslant 0$ or $x < y$ then ERROR
elseif $x < -0.01$ then $\mathrm{Qexp} = \frac{1 - e^x}{1 - e^y}$
elseif $y < -0.01$ then $\mathrm{Qexp} = \frac{x\,\mathrm{Zexp}(x)}{e^y - 1}$
else $\mathrm{Qexp} = \frac{x\,\mathrm{Zexp}(x)}{y\,\mathrm{Zexp}(y)}$
```

$$\mathrm{Bexp}(x, y) = \frac{x - y}{e^x - e^y}$$

{ use: calculation Jacobian matrix elements (cf. (6.13)) and Dexp (x) }

```
if $x > y$ then $\mathrm{Bexp} = \mathrm{Bexp}(y,x)$
else $\mathrm{Bexp} = \frac{e^{-y}}{\mathrm{Zexp}(x-y)}$
```

```
$\mathrm{Dexp}(a, b, c, d) = \dfrac{\mathrm{Bexp}(a, b)}{\mathrm{Bexp}(c, d)}$
{ use: calculation current densities (cf. (6.11)) }
$amb = a - b$
$cmd = c - d$
if $amb \leqslant 0$ then
        $abe = b$
else
        $abe = a$
        $amb = -amb$
if $cmd \leqslant 0$ then
        $cde = d$
else
        $cde = c$
        $cmd = -cmd$
$\mathrm{Dexp} = e^{cde - abe} \dfrac{\mathrm{Zexp}(cmd)}{\mathrm{Zexp}(amb)}$
```

Appendix B

Description of test problems

In this Appendix we give a detailed description of the test problems that have been used in this thesis, i.e. the quarter circle diode (Chapter 5), the bipolar transistor (Chapter 6), the MOS-transistor (Chapter 7) and the LDDMOS-transistor (Chapter 7). The bipolar transistor problem and the LDDMOS problem are taken from the CURRY example set [1], whereas the MOS problem has been suggested to us by Dr. W.H.A. Schilders (Philips Natlab). Table B lists some values of physical parameters that we have used in our numerical experiments and on the following pages a detailed description of the test devices is given. All lengths are expressed in cm and the scaling, discussed in Section 1, is not used.

Symbol	Meaning	Value
n_i	intrinsic concentration	$1.22 \times 10^{+10}\ \mathrm{cm}^{-3}$
τ_n	electron lifetime	$1.0 \times 10^{-6}\ \mathrm{s}$
τ_p	hole lifetime	$1.0 \times 10^{-6}\ \mathrm{s}$
ϵ_0	permittivity of vacuum	$8.854187818 \times 10^{-14}\ \mathrm{A\,s\,V^{-1}\,cm^{-1}}$
ϵ_R^S	relative permittivity silicon	11.7
ϵ_R^O	relative permittivity oxide	4.0
q	elementary charge	$1.6021 \times 10^{-19}\ \mathrm{A\,s}$
U_T^{-1}	inverse of thermal voltage	$38.68293\ \mathrm{V}^{-1}$

TABLE B. Numerical values of physical parameters.

Quarter circle diode

configuration	Figure 5.2
dimensions	$(0, 10\times10^{-4})\times(0, 10\times10^{-4})$
μ_n	500
μ_p	500
contacts	
anode	$(0, 10\times10^{-4}) - (10\times10^{-4}, 10\times10^{-4})$
cathode	$(0, 0) - (2.5\times10^{-4}, 0)$

```
dope(x, y)
r = sqrt(sqr(x) + sqr(y))
if      $r \leqslant 5\times10^{-4}$   then   dope = $+1.0\times10^{+18}$
elseif  $r = 5\times10^{-4}$   then   dope =    0.0
else                              dope = $-1.0\times10^{+18}$
```

Bipolar transistor

configuration	Figure 6.4
dimensions	$(0, 20\times 10^{-4})\times(-8\times 10^{-4}, 0)$
μ_n	500
μ_p	500
contacts	
emitter	$(0, 0) - (3\times 10^{-4}, 0)$
base	$(14\times 10^{-4}, 0) - (18\times 10^{-4}, 0)$
collector	$(0, -8\times 10^{-4}) - (20\times 10^{-4}, -8\times 10^{-4})$

dope(x, y)

bgdope $= +6.00\times 10^{+15}$

dptop1	$= +6.00\times 10^{+19}$	dptop2	$= -2.15\times 10^{+18}$	dptop3	$= +1.10\times 10^{+19}$
ydisp1	$= +0.00\times 10^{+00}$	ydisp2	$= +0.00\times 10^{+00}$	ydisp3	$= +8.00\times 10^{-04}$
d1	$= +7.10\times 10^{-05}$	d2	$= +1.15\times 10^{-04}$	d3	$= +1.30\times 10^{-04}$
xmskl1	$= -5.00\times 10^{-04}$	xmskl2	$= -2.10\times 10^{-03}$	xmskl3	$= -2.50\times 10^{-03}$
xmskr1	$= +5.00\times 10^{-04}$	xmskr2	$= +2.10\times 10^{-03}$	xmskr3	$= +5.00\times 10^{-03}$

```
{ n-type background doping }
dope = bgdope
{ n-type emitter doping }
fx   = 0.5*(erf((x−xmskl1)/d1) − erf((x−xmskr1)/d1))
fy   = dptop1*exp(−sqr((y+ydisp1)/d1))
dope = dope + fx*fy
{ p-type base doping }
fx   = 0.5*(erf((x−xmskl2)/d2) − erf((x−xmskr2)/d2))
fy   = dptop2*exp(−sqr((y+ydisp2)/d2))
dope = dope + fx*fy
{ n-type collector doping }
fx   = 0.5*(erf((x−xmskl3)/d3) − erf((x−xmskr3)/d3))
fy   = dptop3*exp(−sqr((y+ydisp3)/d3))
dope = dope + fx*fy
```

MOS-TRANSISTOR

configuration	Figure 7.5
dimensions	$(-2\times10^{-4}, +2\times10^{-4})\times(-1.5\times10^{-4}, +0.05\times10^{-04})$
μ_n	300
μ_p	200
contacts	
source	$(-2\times10^{-4}, -0.2\times10^{-4})-(-2\times10^{-4}, 0)$
gate	$(-1\times10^{-4}, 0.05\times10^{-4})-(+1\times10^{-4}, 0.05\times10^{-4})$
drain	$(+2\times10^{-4}, -0.2\times10^{-4})-(+2\times10^{-4}, 0)$
substrate	$(-2\times10^{-4}, -1.5\times10^{-4})-(+2\times10^{-4}, -1.5\times10^{-4})$

```
dope(x, y)
bgdope  = $-1.00\times10^{+16}$
dnbox   = $+1.00\times10^{+15}$
ydisp1  = $+0.10\times10^{-04}$
drp     = $+4.00\times10^{-06}$
dope    = bgdope
**if** $x \leqslant -1.01\times10^{-04}$ **or** $x \geqslant 0.99\times10^{-04}$ **then**
    **if** $y \geqslant -0.201\times10^{-04}$ **then**
            d1      = drp*sqrt(2)
            dptop1  = dnbox/(sqrt($\pi$)*d1)
            fy      = dptop1*exp($-$sqr((y+ydisp1)/d1))
            dope    = fy
```

LDDMOS-TRANSISTOR

configuration	Figure 7.6
dimensions	$(0, 2.2\times10^{-4})\times(-2.0\times10^{-4}, 0.2675\times10^{-4})$
polysilicon	$(0.6\times10^{-4}, 1.6\times10^{-4})\times(0.0175\times10^{-4}, 0.2675\times10^{-4})$
μ_n	300
contacts	
source	$(0, 0) - (0.1\times10^{-4}, 0)$
drain	$(2.1\times10^{-4}, 0) - (2.2\times10^{-4}, 0)$
gate	$(0.6\times10^{-4}, 0.2675\times10^{-4}) - (1.6\times10^{-4}, 0.2675\times10^{-4})$
substrate	$(0, -2\times10^{-4}) - (2.2\times10^{-4}, -2\times10^{-4})$

dope (x, y)

bgdope $= -1.00\times10^{+15}$

dptop0	$= -1.0500\times10^{+17}$	dptop1	$= -8.0000\times10^{+15}$
ydisp0	$= +0.0500\times10^{-04}$	ydisp1	$= +0.4000\times10^{-04}$
d0	$= +0.1500\times10^{-04}$	d1	$= +0.1600\times10^{-04}$
		xmskl1	$= -1.0000\times10^{-03}$
		xmskr1	$= +1.0000\times10^{-03}$
dptop2	$= +2.0000\times10^{+20}$	dptop3	$= +2.0000\times10^{+20}$
ydisp2	$= +0.0650\times10^{-04}$	ydisp3	$= +0.0650\times10^{-04}$
d2	$= +0.0557\times10^{-04}$	d3	$= +0.0557\times10^{-04}$
xmskl2	$= -0.2500\times10^{-04}$	xmskl3	$= +1.9500\times10^{-04}$
xmskr2	$= +0.2500\times10^{-04}$	xmskr3	$= +2.4500\times10^{-03}$
dptop4	$= +2.4000\times10^{+18}$	dptop5	$= +2.4000\times10^{+18}$
ydisp4	$= +0.0000\times10^{+00}$	ydisp5	$= +0.0000\times10^{+00}$
d4	$= +0.1875\times10^{-04}$	d5	$= +0.1875\times10^{-04}$
xmskl4	$= -0.5750\times10^{-04}$	xmskl5	$= +1.6250\times10^{-04}$
xmskr4	$= +0.5750\times10^{-04}$	xmskr5	$= +2.7750\times10^{-04}$

```
dope = bgdope
{ treshold implantation }
if y > -ydisp0 then   fy = dptop0
else                  fy = dptop0*exp(-sqr((y+ydisp0)/d0))
dope = dope + fy
{ anti punch through implantation }
fy     = dptop1*exp(-sqr((y+ydisp1)/d1))
fx     = 0.5*(erf((x-xmskl1)/d1) - erf((x-xmskr1)/d1))
dope   = dope + fx*fy
```

```
{ n+  source }
fy    = dptop2*exp(-sqr((y+ydisp2)/d2))
fx    = 0.5*(erf((x-xmskl2)/d2) - erf((x-xmskr2)/d2))
dope  = dope + fx*fy
{ n+  drain }
fy    = dptop3*exp(-sqr((y+ydisp3)/d3))
fx    = 0.5*(erf((x-xmskl3)/d3) - erf((x-xmskr3)/d3))
dope  = dope + fx*fy
{ n-  source }
fy    = dptop4*exp(-sqr((y+ydisp4)/d4))
fx    = 0.5*(erf((x-xmskl4)/d4) - erf((x-xmskr4)/d4))
dope  = dope + fx*fy
{ n-  drain }
fy    = dptop5*exp(-sqr((y+ydisp5)/d5))
fx    = 0.5*(erf((x-xmskl5)/d5) - erf((x-xmskr5)/d5))
dope  = dope + fx*fy
{ polysilicon }
if  $y \geqslant 0.01\times10^{-04}$  then  dope = $1.0\times10^{+20}$
```

REFERENCES

1. C. LEPOETER (1987). *CURRY example set,* Technical Report No. 4322.271.6005, Philips, Corp. CAD Centre, Eindhoven.

Samenvatting

In de ontwikkelfase van halfgeleider devices wordt veelvuldig gebruik gemaakt van numerieke simulaties. Numerieke simulaties zijn niet alleen goedkoper, maar ook sneller en flexibeler dan experimentele onderzoekingen. In dit proefschrift beschouwen wij de device simulatie. Het doel van een device simulatie is het voorspellen van het electrische gedrag van een halfgeleider device, zoals bijvoorbeeld het electrische veld in het device en de IV-karakteristiek. Dit gedrag wordt beschreven met behulp van een stelsel van partiële differentiaal vergelijkingen, de halfgeleider vergelijkingen, dat bestaat uit de Poisson vergelijking voor het electrische veld, continuiteits vergelijkingen voor gaten en electronen, en de drift-diffusie benadering voor de electronen en gaten stroomdichtheden. De moeilijkheden die zich voordoen bij het numeriek oplossen van de halfgeleider vergelijkingen zijn o.a.:

- de enorme variatie in orde-van-grootte die optreedt in de te berekenen grootheden,
- de sterke niet-lineariteit van het probleem,
- het singulier gestoorde karakter van de vergelijkingen waardoor het gebruik van adaptief gegenereerde roosters wenselijk is,
- en de zeer grote lineaire en niet-lineaire stelsels die opgelost moeten worden voor gedetailleerde simulaties; dit vereist een efficiente oplosmethode.

Het is bekend dat multirooster methoden zeer efficient zijn voor diverse probleem klassen, en daarom wordt de toepasbaarheid van multirooster methoden voor het halfgeleider probleem onderzocht .

Wij beschouwen twee gemengde eindige elementen discretisaties van de halfgeleider vergelijkingen: de duale en de primale versie. Door het gebruik van geschikte kwadratuur-regels in de discretisatie verkrijgen wij schema's, die, respectievelijk, equivalent zijn met cell-centered en vertex-centered eindige volume discretisaties. Voor het oplossen van deze stelsels van niet-lineaire vergelijkingen wordt, respectievelijk, een cell-centered en een vertex-centered multirooster methode gebruikt. De duale gemengde eindige elementen discretisatie wordt uitgevoerd op adaptieve roosters.

Echter, in de cell-centered multirooster methode doet zich het probleem voor dat de schaling van de vergelijkingen op de grove en fijne roosters enorm kan verschillen, waardoor het onmogelijk is om het grof-rooster probleem op te lossen. Om dit probleem te ondervangen wordt een lokale demping van het residu gebruikt. In de vertex-centered multirooster methode kan het probleem vermeden worden door het gebruik van een zeer eenvoudige restrictie voor het residu: de injectie.

Uit de literatuur is bekend dat injectie ongeschikt is als restrictie van het residu in multirooster methoden voor het oplossen van tweede orde differentie vergelijkingen: hoog frequente fout-componenten worden opgeblazen in de grof-rooster correctie. Om dit probleem te ondervangen construeren wij een speciale smoother, die deze hoog frequente fout-componenten elimineert.

Wij demonstreren de bruikbaarheid van multirooster methoden voor halfgeleider simulatie aan de hand van diverse praktische test problemen. Het blijkt dat de vertex-centered multirooster methode een robuust en optimaal efficient algorithme is voor het numeriek oplossen van de halfgeleider vergelijkingen.

Stellingen behorende bij het proefschrift

Multigrid Methods for Semiconductor Device Simulation

van H. Molenaar

1. Cell-centered multirooster is minder geschikt voor het numeriek oplossen van de halfgeleider vergelijkingen dan vertex-centered multirooster vanwege de enorme variatie die optreedt in de orde van grootte van het residu van de continuïteitsvergelijkingen.

   Hoofdstuk 7 van dit proefschrift.

2. Voor het niet-lineaire multirooster-algoritme is telkens een beginschatting op de grove roosters nodig. Voor de halfgeleider vergelijkingen verdient het de voorkeur hiervoor de oplossing van het grofrooster-probleem te nemen in plaats van een restrictie van de fijnrooster-oplossing.

   Hoofdstuk 7 van dit proefschrift.

3. Het grofste rooster in een multirooster-algoritme moet niet zo grof mogelijk gekozen worden, maar eerder zo fijn dat de vergelijkingen op dat rooster nog steeds snel opgelost kunnen worden.

4. De Vanka-type relaxatie is een efficiëntere smoother voor de duale gemengde eindige elementen discretisatie van de Poisson vergelijking dan de superbox-relaxatie die door Schmidt en Jacobs geïntroduceerd is.

   Hoofdstuk 3 van dit proefschrift.

   S.P. Vanka, *Block-Implicit Multigrid Solution of Navier-Stokes Equations in Primitive Variables*, J.Comput.Phys., 1986, 65, 138-158.
   G.H. Schmidt and F.J. Jacobs, *Adaptive Local Grid Refinement and Multi-grid in Numerical Reservoir Simulation*, J.Comput.Phys., 1988, 77, 140-165.

5. Het gebruik van onnauwkeurige rooster-transferoperatoren in multirooster-algoritmen kan ondervangen worden door een geschikte keuze van de smoother.

   Hoofdstuk 4 en 7 van dit proefschrift.

6. Beschouw de volgende partiële differentiaalvergelijking: $-\phi_{xx} - \varepsilon\phi_{yy} = f$. De stelling van Khalil, dat symmetrisch lijn Gauss-Seidel relaxatie een perfecte smoother voor dit probleem is in het limietgeval $\varepsilon \rightarrow 0$, is gebaseerd op een analyse van het geval dat uitsluitend Dirichlet randvoorwaarden gegeven zijn. Indien echter ook Neumann randvoorwaarden gegeven zijn dan is de bewering niet waar.

Stellingen behorende bij het proefschrift *Analysis of Linear Multigrid Methods for Elliptic Differential Equations with Discontinuous and Anisotropic Coefficients* van M. Khalil.

7. Landschapsfotografie in Nederland is een vorm van architectuurfotografie.

8. De wandelsport in Nederland is meer gediend met een verbod op de verkoop van artikel 461-bordjes, dan met het uitzetten van meer Lange-Afstand-Wandelpaden.

9. In de discussie rond de beperkte toegang tot de gezondheidszorg voor mensen met een ongezonde levenswijze wordt er ten onrechte van uitgegaan dat zij gedurende hun *leven* vaker ziek zijn.

10. Bezitters van oliekachels vertonen in de wintermaanden een vergrote sociale activiteit.