Ralf Kornhuber   Rainer Roitzsch

Self Adaptive Finite Element Simulation of Reverse Biased pn–Junctions

Ralf Kornhuber   Rainer Roitzsch

Self Adaptive Finite Element Simulation
of Reverse Biased pn–Junctions

Abstract. The potential distribution of reverse biased pn–junctions can be described by a double obstacle problem for the Laplacian. This problem is solved by a self adaptive Finite Element Method involving automatic termination criteria for the iterative solver, local error estimation and local mesh refinement. Special attention is paid to the efficient resolution of the geometries typically arising in semiconductor device simulation. The algorithm is applied to a reverse biased pn–junction with multi–step field plate and stop–electrode to illustrate its efficiency and reliability.

Introduction

Traditionally Finite Difference Methods are very popular in the field of semi-conductor device simulation. This may be due to the fact that the usual geometries have an approximately rectangular structure and that the initial amount of programming work is comparably small. Nevertheless in recent years the built in inflexibility of Finite Difference Methods turned out to be increasingly disadvantageous. Especially the automatic reduction of unknowns based on self adaptive methods is difficult to perform in usual finite difference framework. Hence Finite Element Methods based on triangular or quadrangular elements have become more and more attractive.

In the present paper the self adaptive approach derived in [11] is applied to the simulation of reverse biased pn–junctions which mathematically turns out to be a double obstacle problem for the Laplacian. The main idea is to approximate the solution ψ not only by means of one fixed triangulation but to construct a suitable sequence $\mathcal{T}_0$, $\mathcal{T}_1$, ... of triangulations on which ψ is approximated. Here the information contained in $\mathcal{T}_k$ and the approximation Ψ_k is used inductively to design the new triangulation $\mathcal{T}_{k+1}$ on the next level. Naturally the initial triangulation $\mathcal{T}_0$ is assumed to be comparatively coarse.

The main ingredients of the presented self adaptive algorithm are as follows

- Resolution of the geometry with a minimal number of nodes by allowing acute angles for initial triangles $t \in \mathcal{T}_0$.

- Acceleration of the iterative solver by good start iterates.

- Automatic choice of termination criteria for the iterative solver.

- Local mesh refinement based on local error estimation involving the improvement of acute angles.

The paper is organized as follows. In the first chapter we state the physical problem which is then transformed in a suitable weak formulation to which Finite Element Methods can be applied. We further recall a simple iterative scheme for the solution of the discrete problem. The second chapter contains a detailed description of the self adaptive algorithm. Numerical results are reported in Chapter 3.

1. Preliminaries

1.1 Physical Modelling

We consider a device occupying a bounded polygonal domain $\Omega \subset \mathrm{IR}^2$ whose stationary behavior is ruled by the drift–diffusion equations [19]

$$
\begin{aligned}
-\operatorname{div}(\varepsilon \nabla \psi) &= q(D - n + p) \\
\operatorname{div} J_n &= qR , \quad J_n = q(D_n \nabla n - \mu_n n \nabla \psi) \\
\operatorname{div} J_p &= -qR , \quad J_p = -q(D_p \nabla p + \mu_p p \nabla \psi)
\end{aligned}
\tag{1.1}
$$

where usually the electric potential ψ and the carrier concentrations n and p for holes and electrons are unknown while the permittivity ε, the doping profile D, the elementary charge q, the electron and hole diffusivities D_n and D_p, the electron and hole mobilities μ_n and μ_p, and the recombination–generation rate R are given parameters of the problem.

The boundary $\partial\Omega$ of Ω is split into (ohmic) contacts $\partial\Omega_C$ and insulating segments $\partial\Omega_I$. This leads to Dirichlet boundary conditions for ψ, n and p on $\partial\Omega_C$ and vanishing electric field $\nabla\psi$ and current densities J_n, J_p on $\partial\Omega_I$. This model whose advantages and limits are thoroughly discussed in [20] can be considerably simplified under strongly reverse bias conditions.

Let us consider a pn–junction γ separating a p–region Ω_p from the remaining device as shown in Figure 1.1.
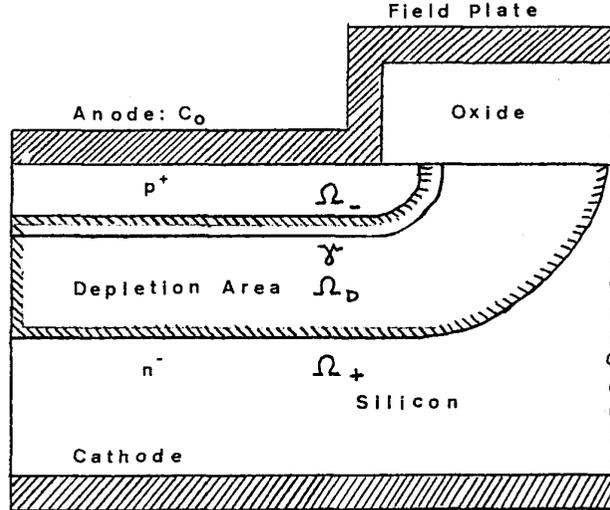

Figure 1.1 Planar pn–junction with single–step field plate

Assume that a constant negative voltage $-\psi_0$ is applied to the anode $C_0 \subset \partial\Omega_C$ attached to Ω_p, while the voltage at all other contacts is kept zero. Then

3

the carriers are sucked away from the junction γ leaving a depletion region Ω_D where ideally no carriers are present and the potential is bounded by $-\psi_0$ and zero.

$$n = p = 0 \ , \quad -\psi_0 < \psi < 0 \quad \text{on } \Omega_D \ . \tag{1.2}$$

The depletion region Ω_D separates the remaining parts Ω_+ and Ω_- of the domain Ω. For large ψ_0 the total depletion assumption

$$\begin{aligned} n = 0 \ , \quad & p = -D \ , \quad & \nabla\psi = 0 \quad \text{on } \Omega_- \\ n = D \ , \quad & p = 0 \ , \quad & \nabla\psi = 0 \quad \text{on } \Omega_+ \end{aligned} \tag{1.3}$$

holds. Substituting (1.2) and (1.3) in (1.1) we obtain

$$\begin{aligned} \psi &= -\psi_0 \quad & \text{on } \Omega_- \\ -\operatorname{div}(\varepsilon\nabla\psi) &= qD \quad & \text{on } \Omega_D \\ \psi &= 0 \quad & \text{on } \Omega_+ \ . \end{aligned} \tag{1.4.a}$$

We additionally assume that the potential ψ and the electric field $\nabla\psi$ are continuous across the inner boundaries $\Gamma_- = \bar{\Omega}_- \cap \bar{\Omega}_D$ and $\Gamma_+ = \bar{\Omega}_+ \cap \bar{\Omega}_D$,

$$\begin{aligned} \psi &= -\psi_0 \ , \quad & \nabla\psi \cdot n_- = 0 \quad \text{on } \Gamma_- \\ \psi &= 0 \ , \quad & \nabla\psi \cdot n_+ = 0 \quad \text{on } \Gamma_+ \end{aligned} \tag{1.4.b}$$

with n_-, n_+ denoting the unit normals on Γ_-, Γ_+ respectively. Note that the free boundaries Γ_-, Γ_+ together with ψ are unknowns of problem (1.4) which is completed by the boundary conditions.

$$\begin{aligned} \psi|_{\partial\Omega_C} &= \begin{cases} -\psi_0 \quad & \text{on} \quad & C_0 \\ 0 \quad & \text{on} \quad & \partial\Omega_C \backslash C_0 \end{cases} \\ \left. \frac{\partial}{\partial n}\psi \right|_{\partial\Omega_I} &= 0 \ . \end{aligned} \tag{1.4.c}$$

Remark 1.1

Using an appropriate scaling introduced by BREZZI [9] the drift diffusion equations (1.1) become singularly perturbed as $\psi_0 \longrightarrow \infty$. Then (1.4) is recovered as the corresponding reduced problem. See [17] for details.

Remark 1.2

Note that the Laplace equation in (1.4.a) only holds where the permittivity ε is smooth. It has to be replaced by a suitable flux balance condition at jump discontinuities of ε due to different materials.

4

1.2 Finite Element Approximation

In order to apply Finite Element Methods to (1.4) we need a suitable weak formulation of the problem allowing for solutions which are not twice differentiable. Now it is well known that in the simple case of the pure Laplacian

$$-\operatorname{div}(\varepsilon\nabla\psi) = qD \text{ on } \Omega \qquad (1.5)$$

with boundary conditions (1.4.c), the appropriate weak problem turns out to be the minimization of the energy functional

$$F(\varphi) = \tfrac{1}{2}a(\varphi,\varphi) - l(\varphi) \qquad (1.6)$$

over all $\varphi \in H$ and

$$H = \left\{\varphi \in H^1(\Omega) \mid \varphi = -\psi_0 \text{ on } C_0, \ \varphi = 0 \text{ on } \partial\Omega_C\backslash C_0\right\}.$$

Here $H^1(\Omega)$ denotes the space of all quadratic summable functions with quadratic summable derivative provided with the norm

$$\|\varphi\| = \left(\int_\Omega \varphi^2(x,y) + \varphi_x^2(x,y) + \varphi_y^2(x,y)d(x,y)\right)^{1/2}. \qquad (1.7)$$

The bilinear form $a(\cdot,\cdot)$ and the functional l are defined by

$$a(\varphi_1,\varphi_2) = \int_\Omega \varepsilon\nabla\varphi_1 \cdot \nabla\varphi_2 d(x,y), \ \varphi_1,\varphi_2 \in H \qquad (1.8)$$

$$l(\varphi) = q \int_\Omega D\varphi d(x,y), \ \varphi \in H. \qquad (1.9)$$

Now our actual problem (1.4) amounts to the solution of the Lacplace equation (1.5) under the additional constraints that

$$-\psi_0 \leq \psi \leq 0 \text{ on } \Omega. \qquad (1.10)$$

Hence it is straightforward to choose the following problem as a weak formulation of (1.4).

Find $\psi \in K$ with $K = \{\varphi \in H \mid -\psi_0 \leq \varphi \leq 0 \ a.e. \ on \ \Omega\}$ so that

$$F(\psi) \leq F(\varphi), \ \varphi \in K. \qquad (1.11)$$

5

Remark 1.3

Let ε be piecewise constant and strictly positive. Then the weak problem (1.11) has a unique solution $\psi \in K$. Further any smooth solution of (1.11) is a solution of the classical problem(1.4). See [4] for details.

Let $\mathcal{T}$ be a triangulation of Ω. More precisely, $\mathcal{T}$ is a set of triangles such that Ω is the union of all triangles $t \in \mathcal{T}$ and such that the intersection of two triangles $t, t' \in \mathcal{T}$ either consists of a common edge or a common vertex or is empty. We further require that all points separating $\partial\Omega_C$ from $\partial\Omega_I$ are vertices of $\mathcal{T}$. The set of all continuous functions on Ω which are linear on each $t \in \mathcal{T}$ is called $S(\mathcal{T})$.

We want to approximate the exact solution ψ by a finite element solution $\Psi \in S(\mathcal{T})$. This corresponds to the minimization of the functional F over the subset of finite element functions $K_{\mathcal{T}} = K \cap S(\mathcal{T})$, i.e.

Find $\Psi \in K_{\mathcal{T}}$ with $K_{\mathcal{T}} = K \cap S(\mathcal{T})$ so that

$$F(\Psi) \leq F(\varphi) \,, \quad \varphi \in K_{\mathcal{T}} \,. \tag{1.12}$$

Remark 1.4

It is easily seen that (1.12) has a unique solution $\Psi \in K_{\mathcal{T}} \subset S(\mathcal{T})$. Further it is shown in [14] for quasiuniform triangulations that

$$\|\psi - \Psi\| \longrightarrow 0$$

if the maximal length h of the edges of the triangulations tends to zero. Though first order convergence is shown in a variety of cases (see [14] for further references) there is no paper known to the authors treating the case of a general polygonal domain. In this case a pure elliptic model problem is considered in [2]. Using weighted Besov norms it is shown that in the case of properly distributed nodes the error measured in $H^1(\Omega)$ behaves like $\mathcal{O}(N^{1/2})$ with N denoting the number of unknowns.

1.3 Solution of the Resulting Discrete Problem

Before we state a simple iterative method to compute the finite element approximation Ψ, we give a more convenient reformulation of the discrete problem (1.12).

Let $\mathcal{P}$ be the set of nodes of the triangulation $\mathcal{T}$. Then for all $p \in \mathcal{P}$ we

define basis functions $\lambda_p \in S(\mathcal{T})$ with the property

$$\lambda_p(p') = \begin{cases} 1 & p = p' \\ 0 & p \neq p' \end{cases}, \quad p' \in \mathcal{P} .$$

Then each $\varphi \in S(\mathcal{T})$ has the representation

$$\varphi = \sum_{p \in \mathcal{P}} \varphi_p \lambda_p ,$$

where $\varphi_p = \varphi(p)$ are the values of φ at the nodes $p \in \mathcal{P}$. As boundary conditions are imposed on the solution Ψ of (1.12) we have

$$\Psi_p = \begin{cases} -\psi_0 & , \quad p \in C_0 \\ 0 & , \quad p \in \partial\Omega_C \backslash C_0 \end{cases}, \quad p \in \mathcal{P}_C$$

for all Dirichlet boundary points $\mathcal{P}_C = \mathcal{P} \cap \partial\Omega_C$. Hence Ψ can be written as

$$\Psi = \Psi_C + \sum_{p \in \mathcal{P} \backslash \mathcal{P}_C} \Psi_p \lambda_p , \tag{1.13}$$

with the known function

$$\Psi_C = \sum_{p \in \mathcal{P}_C} \Psi_p \lambda_p \in S(\mathcal{T}) \tag{1.14}$$

representing the Dirichlet data and the unknown values Ψ_p at the remaining nodes $p \in \mathcal{P}_0 = \mathcal{P} \backslash \mathcal{P}_C$. For convenience we assume that

$$\mathcal{P}_0 = \{p_1, \ldots, p_N\} \tag{1.15}$$

is ordered in some suitable way and $\lambda_i = \lambda_{p_i}, i = 1, \ldots, N$. Then substituting (1.13) in problem (1.12) we obtain the equivalent formulation.

Find a vector $\underline{\Psi} = (\Psi_i)_{i=1}^N \in K_N$ *with*

$$K_N = \{\underline{\varphi} = (\varphi_i)_{i=1}^N \in \mathbb{R}^N \mid -\psi_0 \leq \varphi_i \leq 0 , \quad i = 1, \ldots, N\} \tag{1.16.a}$$

such that

$$\underline{\Psi}^T(A\underline{\Psi} - \underline{b}) = \min_{\underline{\varphi} \in K_N} \underline{\varphi}^T(A\underline{\varphi} - \underline{b}) \tag{1.16.b}$$

In (1.16.b) the matrix A and the vector $\underline{b}$ are given by

$$A = (a_{ij})_{i,j=1}^N , \quad \underline{b} = (b_i)_{i=1}^N \text{ and}$$
$$a_{ij} = \tfrac{1}{2}a(\lambda_j, \lambda_i) , \quad i,j = 1, \ldots, N \tag{1.17}$$
$$b_i = l(\lambda_i) - a(\Psi_C, \lambda_i) , \quad i = 1, \ldots, N .$$

We have obtained an N dimensional minimization problem (1.16.b) on the bounded set $K_N \subset \mathbb{R}^N$. The following iterative algorithm is replacing this difficult N–dimensional constraint minimization by N simple onedimensional constraint minimizations in each iteration step.

We start with some initial guess $\underline{\Psi}^0 = (\Psi_i^0)_{i=1}^N$. Then in the computation of a component Ψ_i^1 of the next iterate $\underline{\Psi}^1$ all other components are kept fixed (either taken from $\underline{\Psi}^0$ or already computed). This is reducing (1.16.b) to a one dimensional quadratic minimization problem on the interval $[-\psi_0, 0]$ whose solution Ψ_i^1 can explicitly be computed. The method may be improved by using not Ψ_i^1 but the weighted average of Ψ_i^0 and Ψ_i^1 as the new iterate. We obtain the following iterative method for the solution of (1.16).

For a given ν–th iterate $\underline{\Psi}^\nu = (\Psi_i^\nu)_{i=1}^N$
the next iterate $\underline{\Psi}^{\nu+1} = (\Psi_i^{\nu+1})_{i=1}^N$ is computed according to

$$\Psi_i^{\nu+1} = \omega\Psi_i^\nu + (1-\omega)\Psi_i^{\nu+1/2} \qquad (1.18.\text{a})$$

with

$$\Psi_i^{\nu+1/2} = \begin{cases} 0\,, & \Psi_i^* > 0 \\ \Psi_i^*\,, & -\psi_0 \leq \Psi_i^* \leq 0 \\ -\psi_0\,, & \Psi_i^* < -\psi_0 \end{cases} \qquad (1.18.\text{b})$$

and

$$\Psi_i^* = (b_i - \sum_{j<i} a_{ij}\Psi_j^{\nu+1} - \sum_{j>i} a_{ij}\Psi_j^\nu)/a_{ii}\,. \qquad (1.18.\text{c})$$

Note, that in the case $K_N = \mathbb{R}^N$ the well–known Gauß–Seidel method with relaxation is recovered. Hence (1.18) is frequently called projected Gauß–Seidel method or relaxation method. It is shown for instance in [10] that the sequence $\underline{\Psi}^\nu, \nu = 0, 1, \ldots.$ produced by (1.18) is convergent to $\underline{\Psi}$ for all $\omega = [0, 2]$. See [14] for an optimal choice of ω. Finally it should be mentioned that this method has been applied in the field of semiconductors by ADLER et al. [1] and FALCK et al. [12].

2. The Self Adaptive Finite Element Algorithm

To produce an approximate finite element solution Ψ with given tolerance TOL,

$$\| \Psi - \psi \| \leq \text{TOL} \tag{2.1}$$

we may choose some triangulation $\mathcal{T}$ and hope that (2.1) is fulfilled. A more sophisticated way amounts to regard not only Ψ but also $\mathcal{T}$ as unknown of the problem. Ideally we desire a triangulation $\mathcal{T}$ providing an accurate approximation in the sense of (2.1) at least computational cost, i.e. with the smallest possible number of nodes. Such a triangulation may be sought by try and error but this may become a quiet unefficient procedure. The self adaptive multilevel approach presented in this section automatically determines a suitable triangulation together with the corresponding approximate solution up to the desired accuracy.

Starting with an intentionally coarse triangulation $\mathcal{T}_0$ let us assume that a triangulation $\mathcal{T}_k$, $k \geq 0$, is given. Then the algorithm reads as follows.

Algorithm 2.1 Self Adaptive Finite Element Algorithm

Step 1: Compute the finite element approximation Ψ_k with respect to the triangulation $\mathcal{T}_k$.

Step 2: Determine a subset $\mathcal{T}'_k \subset \mathcal{T}_k$ where Ψ_k is deemed too inaccurate.

Step 3: Refine the triangles $t \in \mathcal{T}'_k$ to obtain a finer triangulation $\mathcal{T}_{k+1}$.

Step 4: If $\mathcal{T}_{k+1} \neq \mathcal{T}_k$ go to Step 1 with $\mathcal{T}_k$ replaced by $\mathcal{T}_{k+1}$ else stop.

This approach is widely used in self adaptive methods. See for instance [6] or [11]. In general the efficiency increases with the ratio N_K/N_0 where K denotes the level on which the desired accuracy is reached and N_k is the number of unknowns on level k.

To apply the concept to our actual problem (1.11) we have to describe the three first steps of Algorithm 2.1 in detail. This is done in the following three sections.

2.1 Solution of the Discrete Problem

In Section 1.3 we have stated the projected Gauß–Seidel method with relaxation (1.18) providing a sequence Ψ_k^0, Ψ_k^1, ... converging to the desired solution Ψ_k of the discrete problem (1.12) with $\mathcal{T} = \mathcal{T}_k$. To complete Step 1

of Algorithm 2.1 we have to choose a startiterate Ψ_k^0, and a stopping criterion deciding which iterate Ψ_k^ν gives a sufficiently accurate approximation $\tilde{\Psi}_k$ of Ψ_k.

Let us first consider the choice of a startiterate Ψ_k^0. On the coarsest triangulation $\mathcal{T}_0$ where each iteration step is comparably cheap we choose the quite crude initial guess

$$\Psi_k^0(p) := -\psi_0/2 \, , \quad k = 0 \, , \tag{2.2}$$

for the unknown values at the non–Dirichlet points p. For $k > 0$ the approximation $\tilde{\Psi}_{k-1} \in S(\mathcal{T}_{k-1})$ on the previous level is expected to be a good guess for Ψ_k. Hence we choose

$$\Psi_k^0 := I_{k-1}^k \tilde{\Psi}_{k-1} \, , \quad k > 0 \, , \tag{2.3}$$

with I_{k-1}^k denoting the prolongation from $\mathcal{T}_{k-1}$ to $\mathcal{T}_k$.

Obviously we cannot achieve the exact solution Ψ_k of the discrete problem by the projected Gauß–Seidel method or any other iterative scheme. But in general Ψ_k itself is only an approximation for the exact solution ψ. Hence we only need to compute an approximation $\tilde{\Psi}_k$ of Ψ_k which is preserving the order of discretization accuracy. More precisely we want $\tilde{\Psi}_k$ to satisfy the relation

$$\|\tilde{\Psi}_k - \Psi_k\| \le c\|\psi - \Psi_k\| \tag{2.4}$$

with some constant $c \approx 1$. Of course the unavailable terms in (2.4) have to be replaced by suitable approximations. Using an approach developed by DEUFLHARD et al. [11] we first consider the discretization error

$$\varepsilon_k = \|\psi - \Psi_k\|$$

Assume that $k \ge 1$. Then some guess $\tilde{\varepsilon}_{k-1}$ for ε_{k-1} is available from the error estimation that has been carried out in Step 2 of Algorithm 2.1 on the previous level. Hence ε_k is replaced by ε_{k-1} according to

$$\varepsilon_k = \rho_k \varepsilon_{k-1} \tag{2.5}$$

with ρ_k reflecting the variation of the discretization error if $\mathcal{T}_{k-1}$ is replaced by $\mathcal{T}_k$. In view of Remark 1.4 we assume

$$\|\psi - \Psi_k\| = \mathcal{O}(N_k^{-1/2}) \, , \quad k = 0, 1 \dots \tag{2.6}$$

with N_k denoting the number of unknowns on level k. Hence

$$\tilde{\rho}_k = s(N_{k-1}/N_k)^{1/2} \tag{2.7}$$

with some damping factor s, $0 < s < 1$ is a natural choice for an approximation $\tilde{\rho}_k$ of ρ_k. In the numerical example presented in Chapter 3 the value $s = 0.5$ is selected.

10

Remark 2.1

Following [11] it is easily shown that the order of discretization (2.6) is preserved by $\tilde{\Psi}_k$, $k = 0, 1, \ldots$ if

$$\|\tilde{\Psi}_k - \Psi_k\| \leq \tilde{\rho}_k \|\psi - \Psi_{k-1}\|, \quad k = 1, 2, \ldots \tag{2.8}$$

holds with $\tilde{\rho}_k$ defined in (2.7) and (2.4) is satisfied for $k = 0$.

As a consequence of Remark 2.1 we have to make sure that (2.4) holds for $k = 0$. Because of the lack of a posteriori information the error estimation is carried out after a certain number of steps of the iteration (1.16) to provide a new guess $\tilde{\varepsilon}_0$ for the discretization error ε_0. The iteration is stopped if

$$\|\Psi_0^\nu - \Psi_0\| \leq s_0 \tilde{\varepsilon}_0$$

with some safety factor s_0, $0 < s_0 < 1$. In the present version $s_0 = 0.1$ is selected. Though this procedure increases the amount of work on the initial level it does not deteriorate the over all efficiency of the algorithm as long as the initial triangulation is comparably coarse.

In the next step we have to find a guess $\tilde{\delta}_k^\nu$ for the iteration error

$$\delta_k^\nu = \|\Psi_k^\nu - \Psi_k\|$$

appearing on the left hand side of (2.4). First recall the general error estimate

$$\delta_k^\nu \leq \frac{L_k}{1 - L_k} \|\Psi_k^{\nu+1} - \Psi_k^\nu\| \tag{2.9}$$

for fixpoint iterations like the projected Gauß–Seidel scheme with $L_k < 1$ denoting the contraction number of the algorithm. It is well known that for large ν the ratio

$$L_k^\nu = \frac{\|\Psi_k^{\nu+1} - \Psi_k^\nu\|}{\|\Psi_k^\nu - \Psi_k^{\nu-1}\|}$$

gives a good approximation of the convergence rate L_k. Hence after a certain minimal number ν_0 of iterations δ_k^ν is approximated by

$$\tilde{\delta}_k^\nu = \frac{L_k^\nu}{1 - L_k^\nu} \|\Psi_k^{\nu+1} - \Psi_k^\nu\|, \ \nu \geq \nu_0 \, . \tag{2.10}$$

In the actual version of the algorithm $\nu_0 = 10$ is selected.

Remark 2.2

Unfortunately in the case of the projected Gauß–Seidel scheme L_k tends to one for an increasing number of unknowns. Hence the use of this method and

11

the corresponding error estimate (2.10) is limited to reasonable grid sizes. As illustrated in Chapter 3 this is sufficient for technical accuracy.

Finally the explicit evaluation of the H^1-norm $\| \cdot \|$ is too expensive for numerical purposes. Hence $\| \cdot \|$ is replaced by $| \cdot |$ with

$$|\varphi| = (\underline{\varphi}^T A \underline{\varphi})^{\frac{1}{2}}, \quad \underline{\varphi} = (\underline{\varphi}_i)_{i=1}^{N_k} \tag{2.11}$$

for each $\varphi \in S(\mathcal{T}_k)$ represented by

$$\varphi = \Psi_C + \sum_{i=1}^{N_k} \varphi_i \lambda_i$$

corresponding to (1.13). Recall that A is the stiffness matrix defined in (1.17).

Remark 2.3

The norm $| \cdot |$ may be interpreted as a discrete analogue of the energy norm $\| | \cdot | \|$,

$$\| |\varphi| \| = (a(\varphi, \varphi))^{\frac{1}{2}}, \quad \varphi \in H^1(\Omega) \tag{2.12}$$

which is equivalent to $\| \cdot \|$ on the subspace H.

2.2 Local Error Estimation

In Step 2 of Algorithm 2.1 we have to determine a subset $\mathcal{T}_k'$ of $\mathcal{T}_k$ for refinement. This will be done on the basis of a local error estimate which is presented in this section. As the exact solution ψ is out of reach, the approximation $\tilde{\Psi}_k$ computed in Step 1 will be compared with another approximation Φ of ψ of higher order. For this reason we first derive the discrete problem resulting from a piecewise quadratic ansatz. Then this problem is simplified to obtain a less expansive guess of the local error. For ease of presentation the subscript k denoting the actual level is suppressed in the sequel.
Let

$$\mathcal{E}_0 = \{e_1, \ldots, e_M\}$$

denote the set $\mathcal{E}_0$ of edges of triangles of $\mathcal{T} = \mathcal{T}_k$ which are not part of $\partial \Omega_C$. We define

$$\mathcal{P}_0^Q = \{p_1^Q, \ldots, p_M^Q\}$$

as the set of midpoints p_i^Q of e_i, $i = 1, \ldots, M$. Then following the lines of Section 1.3 for each edge e_i, $i = 1, \ldots, M$, we define the piecewise quadratic, continuous function μ_i satisfying

$$
\begin{aligned}
\mu_i(p) &= 0 \,, \quad p \in \mathcal{P}_0 \\
\mu_i(p_j^Q) &= \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases} \,, \quad p \in \mathcal{P}_0^Q \,.
\end{aligned}
\tag{2.13}
$$

Recall that $\mathcal{P}_0$ denotes all nodes of $\mathcal{T}$ which are not lying on $\partial \Omega_C$. Then any continuous, piecewise quadratic function φ on Ω satisfying the boundary conditions has the representation

$$
\varphi = \Psi_C + \sum_{i=1}^{N} \varphi_i^L \lambda_i + \sum_{j=1}^{M} \varphi_j^Q \mu_j = \Psi_C + \varphi^L + \varphi^Q
\tag{2.14}
$$

with Ψ_C and λ_k, $i = 1, \ldots, N$, taken from the preceeding section. Note, that the coefficients φ_j^Q are not the nodal values of φ at the midpoints p_j^Q, $j = 1, \ldots, M$. In fact

$$
\varphi(p_j^Q) = \varphi_j^Q + \tfrac{1}{2}(\varphi_{j-}^L + \varphi_{j+}^L)
\tag{2.15}
$$

holds with $p_{j-}, p_{j+} \in \mathcal{P}$ denoting the vertices of the edge $e_j = (p_{j-}, p_{j+})$ with midpoint p_j^Q.

Let $Q(\mathcal{T})$ denote all functions with a representation (2.14). Then the piecewise quadratic approximation $\Phi \in Q(\mathcal{T})$ of the exact solution ψ of problem (1.11) is determined as follows.

Find $\Phi \in K_{\mathcal{T}}^Q$ with the property

$$
F(\Phi) \leq F(\varphi) \,, \quad \varphi \in K_{\mathcal{T}}^Q \,,
\tag{2.16}
$$

and

$$
K_{\mathcal{T}}^Q \{ \varphi \in \mathcal{Q}(\mathcal{T}) | -\psi_0 \leq \varphi(p) \leq 0, \ p \in \mathcal{P}_0 \cup \mathcal{P}_0^Q \} \,.
$$

Note that $K_{\mathcal{T}}^Q \not\subset K$ as the constraints may be violated between the nodal points $p \in \mathcal{P}_0 \cup \mathcal{P}_0^Q$. Substituting the representations

$$
\Phi = \Psi_C + \sum_{i=1}^{N} \Psi_i^L \lambda_i + \sum_{j=1}^{M} \Psi_j^Q \mu_j = \Psi_C + \Psi^L + \Psi^Q
\tag{2.17}
$$

and (2.14) in (2.16) we obtain an equivalent version corresponding to (1.14).

Find $\underline{\Psi}^L = (\Psi_i^L)_{i=1}^N$ and $\underline{\Psi}^Q = (\Psi_j^Q)_{j=1}^M$ with $(\underline{\Psi}^L, \underline{\Psi}^Q) \in K_{N,M}$,

$$
\begin{aligned}
K_{N,M} = \{ (\underline{\varphi}^L, \underline{\varphi}^Q) = (\varphi_1^L, \ldots, \varphi_N^L, \varphi_1^Q, \ldots, \varphi_M^Q) \in \mathrm{I\!R}^{N+M} | \\
-\psi_0 \leq \varphi_i^L \leq 0 \,, \quad i = 1, \ldots, N, \\
-\psi_0 \leq \varphi_j^Q + \tfrac{1}{2}(\varphi_{j+}^L + \varphi_{j-}^L) \leq 0 \,, \quad j = 1, \ldots, M \}
\end{aligned}
\tag{2.18.a}
$$

13

and

$$\left(\begin{array}{c} \underline{\Psi}^L \\ \underline{\Psi}^Q \end{array}\right)\left[\left(\begin{array}{cc} A_{LL} A_{LQ} \\ A_{QL} A_{QQ} \end{array}\right)\left(\begin{array}{c} \underline{\Psi}^L \\ \underline{\Psi}^Q \end{array}\right) - \left(\begin{array}{c} \underline{b}^L \\ \underline{b}^Q \end{array}\right)\right] = \min_{(\underline{\varphi}^L,\underline{\varphi}^Q)\in K_{N,M}} \quad (2.18.b)$$

Here $A_{LL} := A$ and $\underline{b}^L := \underline{b}$ are taken from (1.17) and $A_{QL} := A_{LQ}^T$ with

$$A_{LQ} = (a_{ij}^{LQ})_{\substack{(i=1,\dots,N \\ j=1,\dots,M)}}, A_{QQ} = (a_{ij}^{QQ})_{i,j=1}^M, \quad \underline{b}^Q = (b_j^Q)_{j=1}^M$$

are given by

$$\begin{aligned} a_{ij}^{LQ} &= \tfrac{1}{2}a(\mu_j,\lambda_i), \quad i=1,\dots,N, \ j=1,\dots,M \\ a_{ij}^{QQ} &= \tfrac{1}{2}a(\mu_j,\mu_i), \quad i,j=1,\dots,M \\ b_j^Q &= l(\mu_j) - a(\Psi_C,\mu_j), \quad j=1,\dots,M \end{aligned} \quad (2.19)$$

where $a(\cdot,\cdot)$ and $l(\cdot)$ are defined in (1.8) and (1.9). Note that as an outcome of (2.15) the unknowns Ψ_i^L, $i=1,\dots,N$ and Ψ_j^Q, $j=1,\dots,M$ are coupled by the set of constraints $K_{N,M}$.

Recall that we have computed an approximation $\tilde{\Psi}$ of ψ and that

$$\|\Phi - \tilde{\Psi}\| = \|\Psi^L - \tilde{\Psi} + \Psi^Q\| \quad (2.20)$$

is intended to be a guess for the discretization error. As it is much too expensive to compute Ψ^L and Ψ^Q explicitly from (2.16) or (2.18) we replace Ψ^L, Ψ^Q by approximations $\tilde{\Psi}^L$, $\tilde{\Psi}^Q$ which are obtained from a suitable simplification of (2.18). First it is easily seen that (2.18.b) can be rewritten as

$$(\underline{\Psi}^L)^T(A_{LL}\underline{\Psi}^L - \underline{b}^L) + (\underline{\Psi}^Q)^T(A_{QQ}\underline{\Psi}^Q - \underline{r}^Q) = \min_{(\underline{\varphi}^L,\underline{\varphi}^Q)\in K_{N,M}} \quad (2.21)$$

with $\underline{r}^Q \in \mathbb{R}^M$ defined by

$$\underline{r}^Q = \underline{r}^Q(\underline{\Psi}^L) = \underline{b}^Q - 2A_{QL}\underline{\Psi}^L \quad (2.22)$$

Now (2.21) is approximated by decoupling the linear and the quadratic part

$$(\underline{\tilde{\Psi}}^L)^T(A_{LL}\underline{\tilde{\Psi}}^L - \underline{b}^L) = \min_{\underline{\varphi}^L\in K_N} \quad (2.23)$$

and

$$(\underline{\tilde{\Psi}}^Q)^T(A_{QQ}\underline{\tilde{\Psi}}^Q - \underline{r}^Q(\underline{\tilde{\Psi}}^L)) = \min_{\underline{\varphi}^Q\in K_M(\tilde{\Psi}^L)} \quad (2.24)$$

with the set of constraints $K_M \subset \mathbb{R}^M$ defined by

$$K_M = K_M(\tilde{\Psi}^L) = \{ \ \underline{\varphi}^Q = (\varphi_j)_{j=1}^M \in \mathbb{R}^M \ |$$
$$-\psi_0 \le \varphi_j^Q + \tfrac{1}{2}(\tilde{\Psi}_{j+}^L + \tilde{\Psi}_{j+}^L) \le 0 \ j=1,\dots,M\} \quad (2.25)$$

with $e_j = (p_{j-}, p_{j+})$, $j=1,\dots,M$.

14

Remark 2.5

Note that the splitting of (2.21) in (2.23) and (2.24) results from one step of a Block–Gauß–Seidel method for (2.21) applied to the startiterate $(0,0)$.

Recall that $\Psi = \Psi_k$ is the exact solution of the linear part (2.23). Hence we choose

$$\tilde{\Psi}^L = \Psi \ . \tag{2.26}$$

The efficient treatment of the quadratic part requires further simplification. The problem is localized substituting the matrix A_{QQ} by the diagonal

$$D_{QQ} = (a_{jj}^{QQ})_{j=1}^M \ . \tag{2.27}$$

so that we end up with

$$\left(\underline{\tilde{\Psi}}^L\right)^T \left(D_{QQ}\underline{\tilde{\Psi}}^Q - \underline{r}^Q(\underline{\tilde{\Psi}})\right) = \min_{\underline{\varphi}^Q \in K_M(\tilde{\Psi})} \ . \tag{2.28}$$

Now the solution $\underline{\tilde{\Psi}}^Q$ of (2.28) can be easily computed. We obtain

$$\tilde{\Psi}_j^Q = \begin{cases} \Psi_j^+, & \Psi_j^* \geq \Psi_j^+ \\ \Psi_j^*, & \Psi_j^- < \Psi_j^* < \Psi_j^+ \ , \quad j = 1,\dots,M \\ \Psi_j^-, & \Psi_j^- < \Psi_j^* \end{cases} \tag{2.29}$$

with

$$\Psi_j^* = r_j^Q/a_{jj}^{QQ}$$
$$\Psi_j^+ = 0 - \tfrac{1}{2}(\tilde{\Psi}_{j+} + \tilde{\Psi}_{j-}) \ , \quad \Psi_j^- = -\psi_0 - \tfrac{1}{2}(\tilde{\Psi}_{j+} + \tilde{\Psi}_{j-})$$

where again $p_{j-}, p_{j+} \in \mathcal{P}$ are denoting the vertices of the edge e_j with midpoint $p_j \in \mathcal{P}_0, j = 1,\dots,M$.

Remark 2.6

A corresponding approach has been introduced by DEUFLHARD et al. [11] in the case of purely elliptic problems where also a mathematical justification is given. The present case will be further investigated in a forthcoming paper.

Having determined $\tilde{\Psi}^L$ and $\tilde{\Psi}^Q$ from (2.26) and (2.29) the expensive H^1–norm $\|\cdot\|$ is again replaced by a discrete energy norm defined by

$$|\varphi^L + \varphi^Q| = \left((\underline{\varphi})^T A\underline{\varphi}^L + (\underline{\varphi}^Q)^T D_{QQ}\underline{\varphi}^Q\right)^{1/2} \tag{2.30}$$

where we have used the representation (2.14). Using the iteration error estimate $\tilde{\delta} = \tilde{\delta}_k$ computed in the preceeding section this finally leads to the discretization error estimate

$$\tilde{\varepsilon} = \left(\tilde{\delta} + (\tilde{\Psi}^Q)^T D_{QQ} \tilde{\Psi}^Q \right)^{1/2} . \tag{2.31}$$

Based on the defect $\tilde{\Psi}^Q$ we now determine a set of edges $\mathcal{E}'_k \subset \mathcal{E}_k$ which is marked for refinement. Recall that $\mathcal{E}_0 = \{e_1, \ldots, e_M\} \subset \mathcal{E}$ denotes the subset of edges which are not part of $\partial\Omega_C$. Of course we want an edge $e_j \in \mathcal{E}_0$ to lie in $\mathcal{E}'$ if

$$\eta(e_j) = a^{QQ}_{jj} (\tilde{\Psi}^Q_j)^2$$

is comparably large. More precisely the elements of $\mathcal{E}'$ are selected as follows. Assume that $k \geq 1$ and an edge $e \in \mathcal{E}_0$ results from subdividing an edge E being part of a triangulation on a lower level. Then an extrapolation of $\eta(e)$ on the next level yields

$$\eta_e = \frac{\eta(e)}{(\eta(E))^2} . \tag{2.32.a}$$

If a father E of e is not present we set

$$\eta_e = 0 . \tag{2.32.b}$$

Finally $\mathcal{E}'$ is defined by

$$\mathcal{E}' = \{e \in \mathcal{E}_0 | \eta(e) \geq \eta_{\max}\}$$

using the maximum $\eta_{\max}$ of all η_e, $e \in \mathcal{E}_0$. This strategy has been originally proposed in [3] and is used in a way suggested in [8].

If $k = 0$ then $\eta_{\max}$ is replaced by the arithmetic mean of all $\eta(e)$, $e \in \mathcal{E}_0$.

2.3 Refinement techniques

In the preceding section we have determined a subset of edges $\mathcal{E}'_k \in \mathcal{E}_k$ which are marked for refinement. Now the subset $\mathcal{T}'_k \subset \mathcal{T}_k$ contains all triangles with at least one edge lying in $\mathcal{E}'_k$. Before we describe how the refinement of the triangles $t \in \mathcal{T}'_k$ is actual performed we briefly discuss the problems resulting from typical geometries of semiconductor devices.

Due to the fabrication process a device usually exhibits a horizontal layer structure as shown in Figure 2.1. Here the width of different layers may vary over 3 or 4 orders of magnitude. For this reason a corresponding coarse triangulation $\mathcal{T}_0$ consisting of "nice" triangles (with interior angles bounded from above and below by about $\pi/2$ and $\pi/4$) may not be very coarse at all so that the multi–level efficiency decreases significantly.
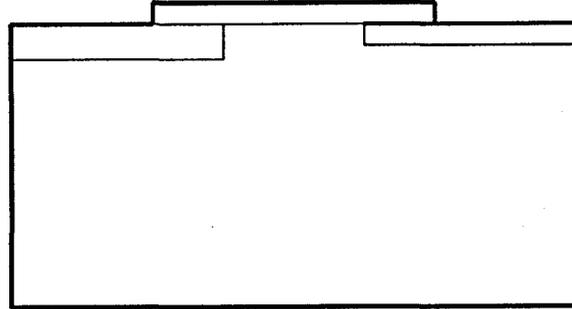
Figure 2.1 Typical geometry

Here one remedy also used in the presence of curved boundaries is to start with a less awkward geometry, and to approximate simultaneously not only the solution but also the geometry. The other possibility which is considered here is to allow "nasty" triangles for $\mathcal{T}_0$. If necessary these triangles may be improved by so-called blue refinements as shown in Figure 2.2.
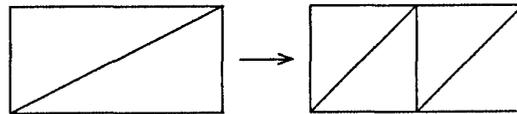

Figure 2.2 Blue refinement

More precisely, blue refinement is performed if the following three conditions hold. For the denotation we refer to Figure 2.3.

(B 1) The quadrangle (t, t') is a candidate for blue refinement.

(B 2) No bisection of e, e' is intended but at least one of the edges E, E' or D is marked for refinement.

(B 3) The angles α, α' are the minimal angles in the triangles t, t'.

Of course the nasty triangles $t \in \mathcal{T}_0$ are candidates for blue refinement on the initial level. This property is expressed by a mark on the corresponding diagonal and is inherited in the refinement process.

The conditions (B 2) and (B 3) make sure that the intended bisections match with blue refinement and that the minimal interior angles are really improved.

17
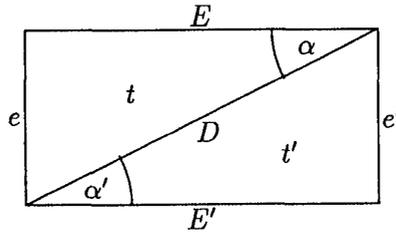
Figure 2.3 Denotation

Remark 2.7

Note that blue refinements have been introduced in [16] under quite different circumstances. Here the situation is less complicated, as the candidates for blue refinement are known a priori.

We are left with triangles which do not satisfy the conditions (B) but have at least one edge that is marked for refinement. In this case a triangle is subdivided in four similar subtriangles according to Figure 2.4. Obviously this so-called red refinement does not affect the interior angles.
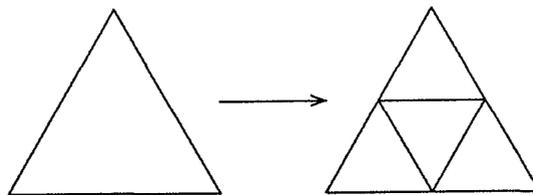
Figure 2.4 Red refinement
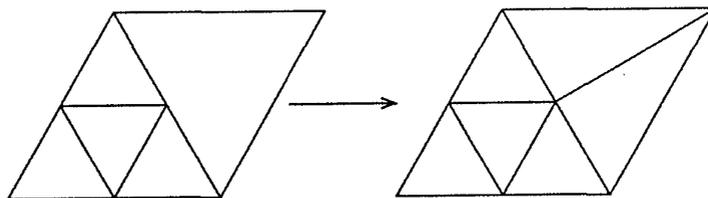
Figure 2.5 Green closure

18

To remedy irregular nodes resulting from blue or red refinement an additional green closure is used as shown in Figure 2.5. Note that the interior angles may deteriorate if green refined triangles are further subdivided. Hence all green refinements are skipped in advance of each new refinement step.

Remark 2.8

After subdivision of all edges $e \in \mathcal{E}_k'$ by red or blue refinement further refinement may be necessary or blue refinements may have to be skipped for structural reasons. We will not go into details here but refer to the pioneering work of BANK et al. [7] and [5] or LEINEN [16] for recent results. Structural problems of blue refinements are discussed in [18].

3. Numerical Results

To illustrate the behavior of the self–adaptive algorithm presented in the preceding chapter let us consider a planar pn–junction with multi–step field plates. The example is due to [13].

The geometry Ω is given in Figure 3.1. Note that the representation is not in scale. Otherwise the different steps of the field plates ranging from $0.08\mu m$ to $8.68\mu m$ would not be visible. The permittivity ε is given by $\varepsilon = \varepsilon_0\ \varepsilon_r$ with the permittivity in vacuum

$$\varepsilon_0 = 8.854 \cdot 10^{-14}\frac{As}{Vcm} \quad \text{modified by} \quad \varepsilon_r = \begin{cases} 1 & \text{air} \\ 3.9 & \text{oxide} \\ 11.7 & \text{silicon} \end{cases} \qquad (3.1)$$

in the different materials. The doping function D has the values

$$D = \begin{cases} -10^{17} & cm^{-3} & \text{in } p^+ \\ 10^{19} & cm^{-3} & \text{in } n^+ \\ 8 \cdot 10^{13} & cm^{-3} & \text{in } n^- \end{cases} . \qquad (3.2)$$

Finally the elementary charge q is given by

$$q = 1.602 \cdot 10^{-19} As . \qquad (3.3)$$

The boundary conditions are of Dirichlet–type at the contacts C_0, C_1, C_2 and at the boundary $ABCDE$ of the considered region. At the contacts we apply the following voltage

$$\psi|_{\partial\Omega_C} = \begin{cases} -\psi_0 & \text{on } C_0 \\ 0 & \text{on } C_1 \\ 0 & \text{on } C_2 \end{cases} \qquad (3.4)$$

with $\psi_0 = 1000$ V. Along AB a linear increase of the potential is assumed which then is kept zero along $BCDE$. Finally the device is assumed to be isolated along $\partial\Omega_I = FG$ so that

$$\varepsilon\frac{\partial}{\partial n}\psi\bigg|_{\partial\Omega_I} = 0 . \qquad (3.5)$$

Using the program BOXES described in [18] we obtain the initial triangulation $\mathcal{T}_0$ depicted in Figure 3.2. Note that $\mathcal{T}_0$ contains triangles with very acute angles but requires only the comparatively small number of 267 nodes. With the help of BOXES we may remove these "nasty" triangles by successive blue refinement to produce a different triangulation $\tilde{\mathcal{T}}_0$ with reasonable triangles but about 2500 nodes. Hence "nice" initial triangles have to be paid by numerical efficiency.

The level curves of the initial solution Ψ_0 are shown in Figure 3.3.
Now we start the self adaptive algorithm 2.1 with the desired accuracy

$$\|\Psi - \psi\| \leq \text{TOL} \qquad (3.6)$$

and $\text{TOL} = 10^{-6}$. Of course other termination criteria as for instance the variation of the computed breakdown voltage may be reasonable. Using the criterion (3.6) the algorithm stops after 9 refinement steps. The corresponding triangulation $\mathcal{T}_9$ is depicted in Figure 3.4. Note that the acute angles have disappeared in the regions of strong refinement and that the nodes are concentrated at the reentrant corners. Figure 3.5 showing the corresponding approximation Ψ_9 confirms the high resolution in this most important area. The final figures illustrate the situation along the silicon/oxide interface. The potential Ψ_9 is depicted in Figure 3.6 while Figure 3.7 shows the norm of the electric field $|\nabla \Psi_9|$ which is of special importance for the blocking capability of the device. Note that the concentrations of points on the right hand side – where the potential is equal to zero – does not result from adaptive refinement but from the resolution of the geometry by the initial triangulation (compare Figure 3.2). All computations have been carried out on a SPARC 1+ workstation requiring 67 seconds of computation time.
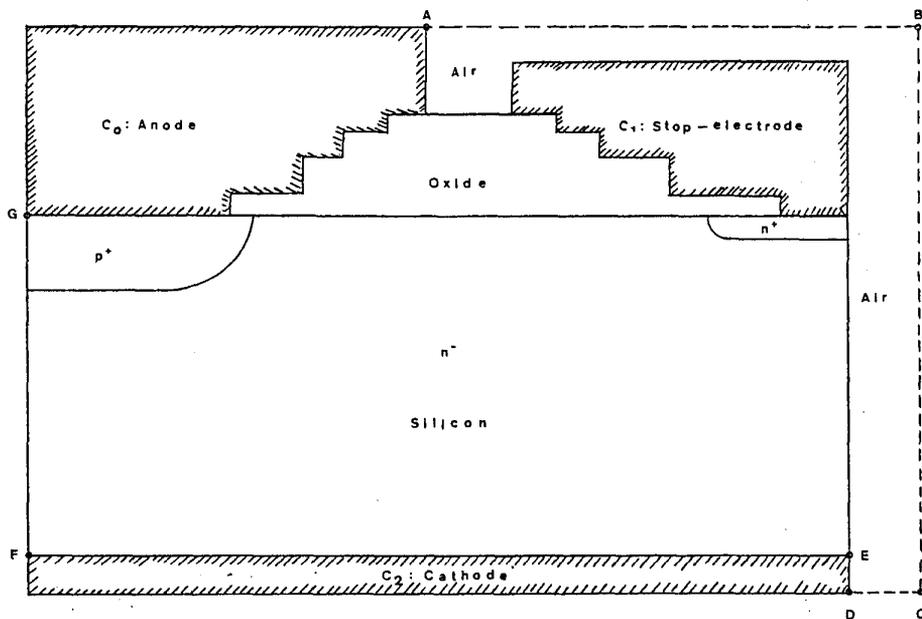
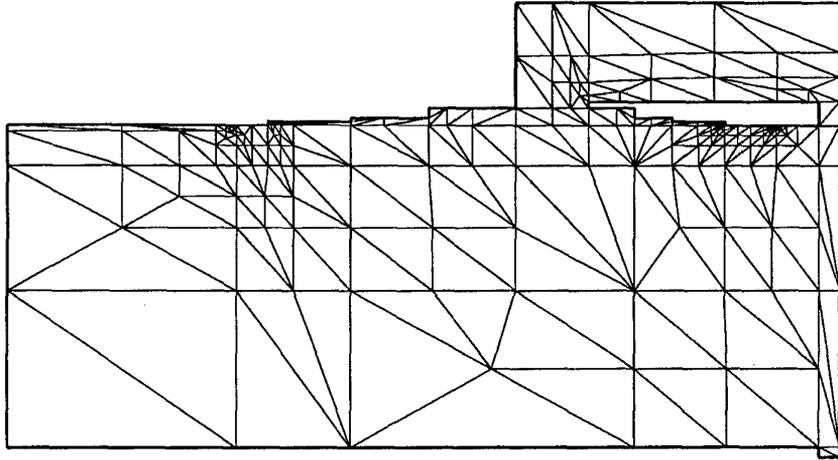Figure 3.1 Geometry of the computational domain

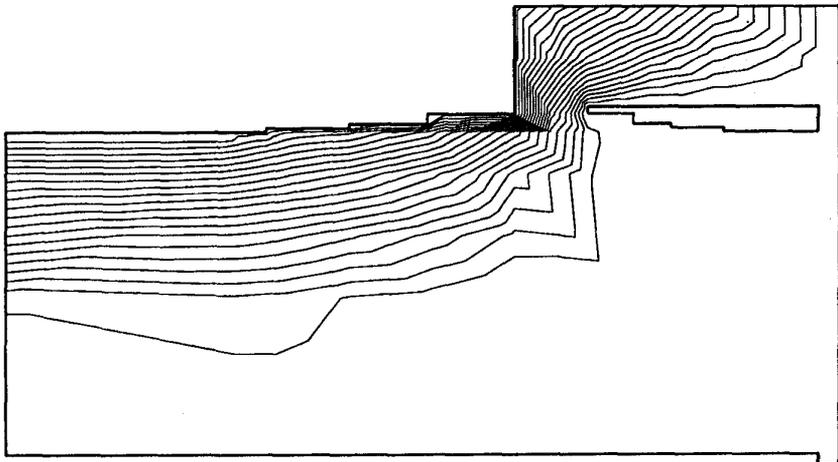**Figure 3.2** Initial triangulation generated by BOXES

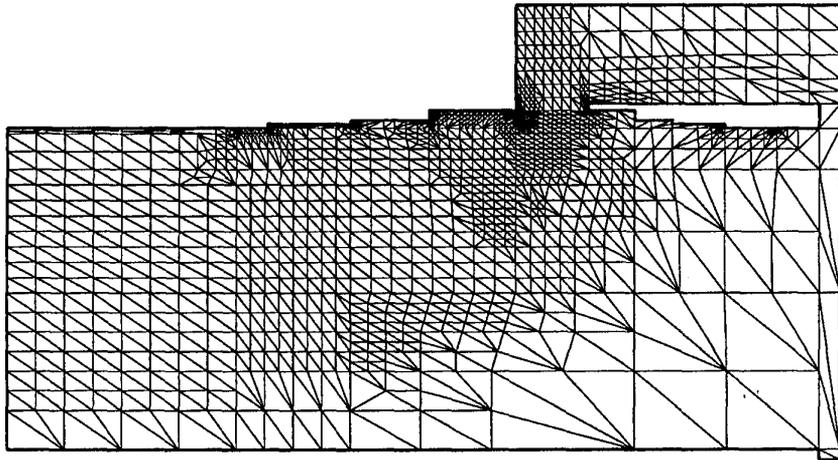**Figure 3.3** Level curves of initial solution Ψ_0 corresponding to $\mathcal{T}_0$

22

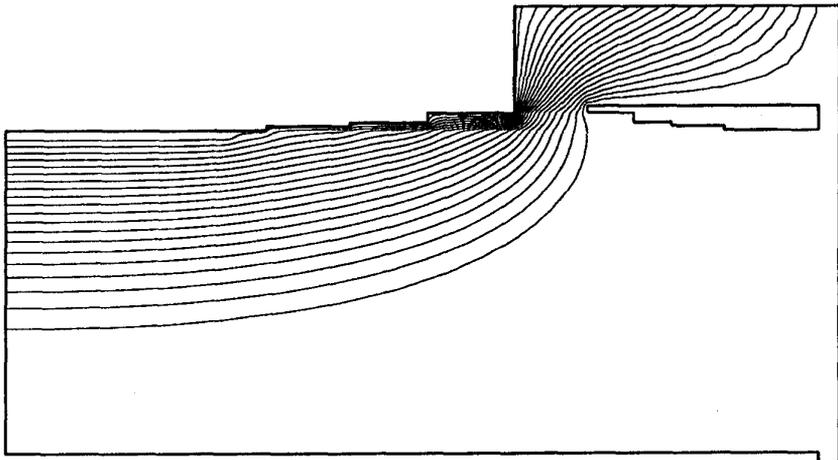Figure 3.4 Final triangulation $\mathcal{T}_9$ obtained by self–adaption refinement

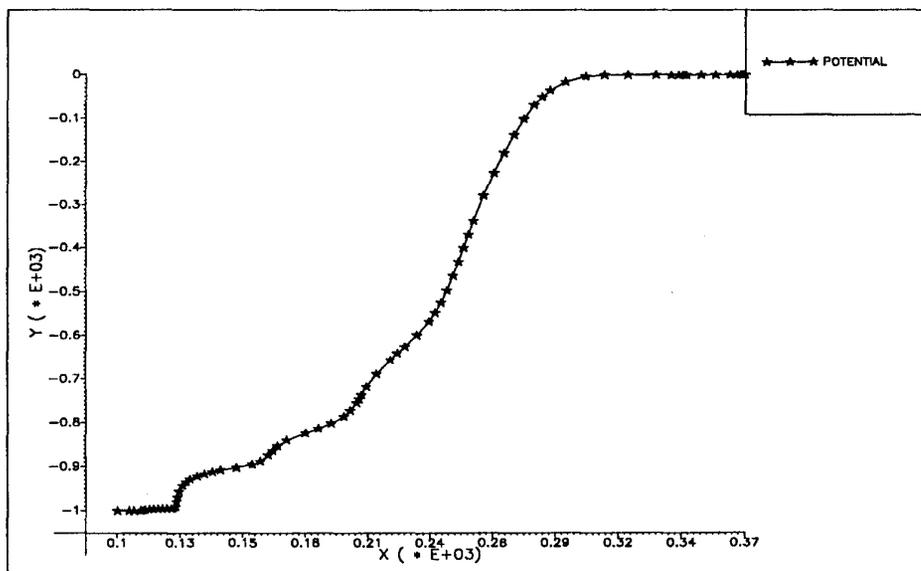Figure 3.5 Level curves of final solution Ψ_9 corresponding to $\mathcal{T}_9$

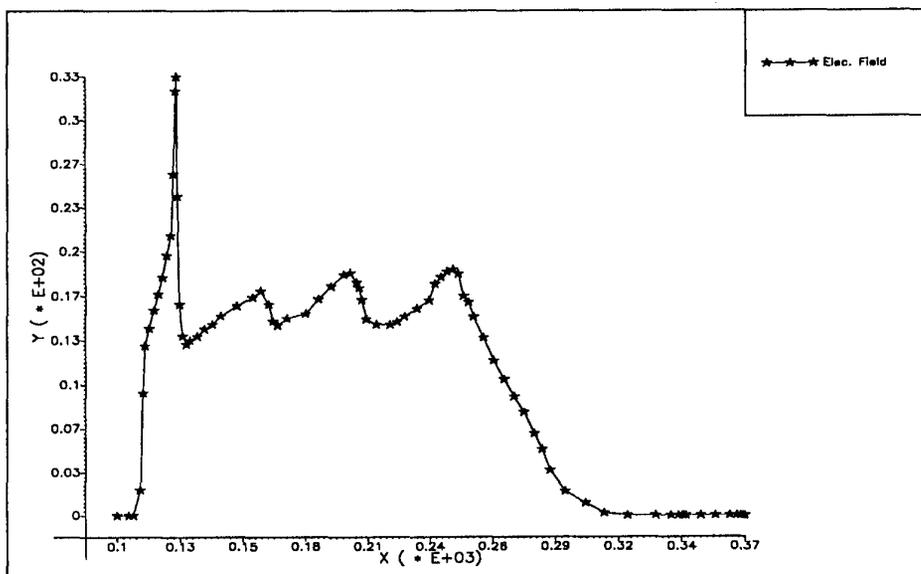**Figure 3.6** Potential Ψ_9 along the silicon/oxide interface

Figure 3.7 Norm of the electric field $|\nabla\Psi_9|$ along the silicon/oxide interface

24

References

[1] M.S. Adler, V. Temple, R.C. Rustay: *Theoretical Basis for Field Calculations on Multi–Dimensional Reverse Biased Semiconductor Devices.* Solid–State Electron **25**, p. 1179–1186 (1982)

[2] I. Babuška, R.B. Kellogg, J. Pitkäranta: *Direct and Inverse Error Estimates for Finite Elements with Mesh Refinements.* Num. Math. **33**, p. 447–471 (1979)

[3] I. Babuška, W.C. Rheinboldt: *Error Estimates for Adaptive Finite Element Computations.* SIAM J. Num. Anal. **15**, p. 736–754 (1978)

[4] C. Baiochi, A. Capelo: *Variational and Quasivariational Inequalities.* Wiley, Chichester, New York (1984)

[5] R.E. Bank: *PLTMG Users Guide.* Version 6.0 (1990)

[6] R.E. Bank, T.F. Dupont, H. Yserentant: *The Hierarchical Basis Multigrid Method.* Num. Math. **52**, p. 427–458 (1988)

[7] R.E. Bank, A.H. Sherman, H. Weiser: *Refinement Algorithm and Data Structures for Regular Local Mesh Refinement.* Scientific Computing, R. Stepleman et al. (eds.), Amsterdam: IMACS North–Holland, p. 3–17 (1983)

[8] F.A. Bornemann: *private communication* (1990)

[9] F. Brezzi: *Theoretical and Numerical Problems in Reverse Biased Semiconductor Devices.* Proc. 7th Intern. Conf. on Comput. Meth. in Applied Science and Engineering. INRIA, Paris (1985)

[10] J. Céa, R. Glowinski: *Sur des méthods d'optimisation par relaxation.* Rev. Fr. Anatom. Inf. Rech. Oper. R3, p. 5–32 (1973)

[11] P. Deuflhard, P. Leinen, H. Yserentant: *Concepts of an Adaptive Hierarchical Finite Element Code.* IMPACT 1, p. 3–35 (1989)

[12] E. Falck, W. Gerlach: *Berechnung der Durchbruchsspannung von planaren pn–Übergängen mit mehrstufigen Feldplatten.* AEÜ **43**, p. 328–334 (1989)

[13] W. Feiler, E. Falck, W. Gerlach: *private communication*

[14] R. Glowinski: *Numerical Methods for Nonlinear Variational Problems.* Springer, New York, Berlin, Heidelberg, Tokyo (1984)

[15] R. Kornhuber, R. Roitzsch: *On Adaptive Grid Refinement in the Presence of Internal or Boudary Layers*. IMPACT 2, p. 40–72 (1990)

[16] P. Leinen: *Ein schnell adaptiver Löser für elliptische Randwertprobleme auf Seriell– und Parallelrechnern*. Dissertation, Universität Dortmund (1990)

[17] P.A. Markowich, C.A. Ringhofer, C. Schmeiser: *Semiconductor Equations*. Springer Verlag Wien, New York (1990)

[18] R. Roitzsch, R. Kornhuber: *BOXES – a Program to Generate Triangulations from a Rectangular Domain Description.*, Technical Report TR90–9, Konrad–Zuse–Zentrum (ZIB) (1990)

[19] W.V. van Roosbroeck: *Theory of Flow of Electrons and Holes in Germanium and other Semiconductors*. Bell Syst. Tech. J. **29**, p. 560–607 (1950)

[20] S. Selberherr: *Analysis and Simulation of Semiconductor Devices*. Springer Verlag Wien, New York (1984)