VISTA Status Report
June 2003

Ch. Hollauer, A. Sheikholeslami, V. Palankovski,
S. Wagner, R. Wittmann, S. Selberherr

Institute for Microelectronics
Technische Universität Wien
Gusshausstrasse 27-29
A-1040 Wien, Austria

Contents

1 3D Modeling of Thermal Oxidation with the Finite Element Method

A numerical model which is suitable to describe three-dimensional thermal oxidation of silicon is proposed. By oxidation the three material components silicon, silicon dioxide and oxidant molecules are involved. The model takes into account that the diffusion of oxidants, the chemical reaction, and the volume increase occur simultaneously in a so-called reactive layer. This reactive layer has a spatial finite width, in contrast to the sharp interface between silicon and dioxide in the convential formulation. The oxidation process is numerically described with a coupled system of equations for reaction, diffusion, and displacement. In order to solve the numerical formulation of the oxidation process the finite element scheme is applied.

1.1 Introduction

Thermal oxidation of silicon is one of the most important steps in fabrication of highly integrated electronic circuits, being mainly used for efficient isolation of adjacent devices from each other.
If a surface of a silicon body has contact with an oxiding atmosphere, the chemical reaction of the oxidant (oxygen or steam) with silicon results in silicon dioxide. This reaction consumes silicon and the newly formed silicon dioxide has more than twice the volume of the original silicon. If a silicon dioxide domain is already existing, the oxidants diffuse through the oxide domain and react at the interface of oxide and silicon to form new oxide so that the dioxide domain is penetrated.
Thermal oxidation is a complex process where the three subprocesses oxidant diffusion, chemical reaction, and volume increase occur simultaneously. The volume increase is the main source of mechanical stress and strain, and these cause displacement [1].
From the mathematical point of view the problem can be described by a coupled system of partial differential equations, one for the diffusion of the oxidant through the oxide, the second for the conversion of silicon into silicon dioxide at the interface, and a third for the mechanical problem of the Si–$\mathrm{SiO_2}$–body, which can be modeled as an elastic, viscoelastic, or viscous body.
All published approaches can be classified essentially into three groups. The first type of method [2] maps the silicon dioxide domain in each time step onto a a simple numerical domain. The second approach uses the boundary element method for diffusion and displacement [3]. The third one [4] models the domain of computation by finite elements.
For a realistic and accurate oxidation simulation the three subproblems should be coupled, however, most oxidation models decouple them into a sequence of quasi-stationary steps. In our model all subprocesses are coupled simultaneously and the oxidation process can be simulated in three dimensions.
We will restrict the following explanation to the most simple physical model of linear oxidant diffusion and linear elastic displacement of the Si–$\mathrm{SiO_2}$–body.

1.2 Model

We define a normalized silicon concentration

$$\eta(\vec{x},t) = \frac{C_{Si}(\vec{x},t)}{C_{0Si}} \tag{1}$$

where $C_{Si}(\vec{x},t)$ is the silicon concentration at time t and point $\vec{x}\,(\mathrm{x},\ \mathrm{y},\ \mathrm{z})$ and C_{0Si} is the concentration in pure silicon. So η is 1 in pure silicon and 0 in pure silicon dioxide.
The oxidant diffusion is described by

$$D\Delta C(\vec{x},t) = k(\eta)C(\vec{x},t)\,. \tag{2}$$

Here D is the diffusion coefficient and $k(\eta)$ is the strength of a spatial sink and not just a reaction coefficient at a sharp interface like in the standard model [5]. $k(\eta)C(\vec{x},t)$ defines how many particles of oxygen per unit volume react in a unit time interval to silicon dioxide.
The change of η is described by

$$\frac{\partial \eta(\vec{x},t)}{\partial t} = -\frac{1}{\lambda}k(\eta)C(\vec{x},t)/N_1 \tag{3}$$

where λ is the volume expansion factor (=2.25) for the reaction from silicon to silicon dioxide, and N_1 is the number of oxidant molecules incorporated into one unit volume of silicon dioxide.
Furthermore, we define in (4) that $k(\eta)$ is linear proportional to η.

$$k = \eta(\vec{x},t)k_{max} \tag{4}$$

The chemical reaction of silicon and oxygen causes a volume increase. The additional volume in a reference volume silicon ΔV, where we assume that the oxidant concentration C is constant, is given by

$$V^{add} = \frac{\lambda - 1}{\lambda}\Delta\, t\Delta\, Vk(\eta)C(\vec{x},t)/N_1\,. \tag{5}$$

We define the normalized additional volume V_{rel}^{add} as

$$V_{rel}^{add} = \frac{V^{add}}{\Delta\, V}\,. \tag{6}$$

For our model we assume, that the Si–SiO_2–body deforms elastically. In the theory of linear elasticity with small displacements $\vec{\theta}(x,y,z) = \{u(x,y,z)\, v(x,y,z)\, w(x,y,z)\}$ and strains ε_{ij} (i,j stands for x, y or z), the strain tensor $\tilde{\varepsilon}$ is defined as

$$\tilde{\varepsilon} = L_D\vec{\theta} \tag{7}$$

where $\vec{\theta}$ is the displacement vector and L_D is a differential operator, so that for example $\varepsilon_{xx} = \frac{\partial u}{\partial x}$ and $\varepsilon_{xy} = \frac{1}{2}(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x})$.
Assuming a linear material, the stress tensor $\tilde{\sigma}$ is given by

$$\tilde{\sigma} = \mathbf{D}\tilde{\varepsilon} \tag{8}$$

where $\mathbf{D}$ is a (6x6) material matrix of elastic constants. The elastic constants are linear functions of Young's Modulus E and Poisson's ratio μ of the materials.
The force vector $\vec{f}(x,y,z) = \{f_x, f_y, f_z\}$ is the gradient of the stress tensor $\tilde{\sigma}$.

$$\vec{f} = \nabla\tilde{\sigma} \tag{9}$$

The most important part is that the volume expansion causes displacement. The normalized additional volume from (6) can be written as

$$V_{rel}^{add} = \varepsilon_x + \varepsilon_y + \varepsilon_z\,. \tag{10}$$

For an isotropic material the strain components are equal so that

$$\varepsilon_x = \varepsilon_y = \varepsilon_z = \frac{1}{3}V_{rel}^{add}\,. \tag{11}$$

With (7) and (11) the relationship between the volume expansion and the displacement is fully determined.

1.3 Discretization

1.3.1 Weak Formulation

Before we start with the discretization we apply the weak formulation on (2) and (3).
So we apply the Galerkin method with linear test functions $N_k(\vec{x})$ on the diffusion equation from (2) and apply Green's theorem to remove the Laplace operator Δ as follows

$$\begin{aligned} k_{max} \int_V \eta\, C\, N_k\, dV = D \int_V \Delta C\, N_k\, dV = \\ D \int_\Gamma \frac{\partial C}{\partial \vec{n}}\, N_k\, d\Gamma - D \int_V \nabla C\, \nabla N_k\, dV \end{aligned} \tag{12}$$

where $\frac{\partial C}{\partial \vec{n}} = 0$ and so the term $\int_\Gamma \frac{\partial C}{\partial \vec{n}}\, N_k\, d\Gamma$ is also zero, and so (12) is simplified to

$$k_{max} \int_V \eta\, C\, N_k\, dV = -D \int_V \nabla C\, \nabla N_k\, dV\,. \tag{13}$$

The application of the Galerkin method with the same linear test functions $N_k(\vec{x})$ to the distribution function from (3) leads to

$$\int_V \frac{\partial \eta}{\partial t}\, N_k\, dV = k_{max} \int_V \eta\, C\, N_k\, dV\,. \tag{14}$$

1.3.2 Oxidant Diffusion

In order to solve (2) and (3) on a three-dimensional domain with the volume V_{global}, we split the domain up into tetrahedral elements with the volume V and perform a finite element discretization. The spatial discretization for $C(\vec{x})$ on a single tetrahedral element is

$$C(\vec{x}, t = t_n) = \sum_{i=1}^{4} c_i^{(t_n)} N_i(\vec{x}) \tag{15}$$

where $c_i^{(t_n)}$ is the oxidant concentration at node i and discrete time t_n. $N_i(\vec{x})$ is the form function on node i.
The spatial discretization for $\eta(\vec{x})$ on a single tetrahedral element is

$$\eta(\vec{x}, t = t_n) = \sum_{i=1}^{4} \eta_i^{(t_n)} N_i(\vec{x}) \tag{16}$$

where $\eta_i^{(t_n)}$ is the normalized silicon concentration at node i and discrete time t_n. $N_i(\vec{x})$ is the linear form function on a node i.
If we replace $C(\vec{x}, t)$ and $\eta(\vec{x}, t)$ in (13) with (15) and (16) we obtain

$$-D \int_V \Big(\sum_{i=1}^{4} c_i^{(t_n)} \nabla N_i\, \nabla N_k \Big) dV =$$

$$k_{max} \int\limits_V \Big(\big(\sum_{i=1}^{4} \eta_i^{(t_n)} N_i \sum_{i=1}^{4} c_i^{(t_n)} N_i \big)\, N_k \Big) dV = k_{max} \int\limits_V \Big(\big(\sum_{i=1}^{4} \eta_i^{(t_n)}\, c_i^{(t_n)}\, N_i \big)\, N_k \Big) dV\,. \tag{17}$$

With the following substitution

$$M_{ki} = \int\limits_V N_k(\vec{x})\, N_i(\vec{x}) dV \tag{18}$$

$$K_{ki} = \int\limits_V \nabla N_k(\vec{x})\, \nabla N_i(\vec{x}) dV \tag{19}$$

(17) is simplified to

$$\sum_{i=1}^{4} \left(D\, K_{ki}\, c_i^{(t_n)} + k_{max}\, M_{ki}\, c_i^{(t_n)}\, \eta_i^{(t_n)} \right) = 0 \tag{20}$$

which is a non-linear equation system (k is the equation index) with the constants D, K_{ki}, k_{max} and M_{ki} and with the unknown variables $c_i^{(t_n)}$ and $\eta_i^{(t_n)}$ for one finite element.

1.3.3 Change of η

The spatial discretization for $C(\vec{x})$ and $\eta(\vec{x})$ is the same like in the last subsection and is already described by (15) and (16). Because of the time dependence of (3) an additional time discretization of the partial differential term $\frac{\partial \eta(\vec{x},t)}{\partial t}$ is necessary. The time discretization is performed with the simple Backward-Euler method as

$$\frac{\partial \eta(\vec{x}, t = t_n)}{\partial t} = \frac{\eta(\vec{x}, t_n) - \eta(\vec{x}, t_{n-1})}{\Delta t} \tag{21}$$

where t_n and t_{n-1} are two successive discrete times.
If we replace $C(\vec{x}, t)$, $\eta(\vec{x}, t)$ and $\frac{\partial \eta(\vec{x},t)}{\partial t}$ in (14) with the discrete expressions (15), (16) and (21), we obtain

$$\frac{1}{\Delta t} \int\limits_V \Big(\sum_{i=1}^{4} \big(\eta_i^{(t_n)} - \eta_i^{(t_{n-1})} \big) N_i\, N_k \Big) dV = k_{max} \int\limits_V \Big(\big(\sum_{i=1}^{4} \big(\eta_i^{(t_n)}\, c_i^{(t_n)} \big) N_i\, N_k \Big) dV. \tag{22}$$

With the substitution (18) the last equation is simplified to a non-linear equations system (k is the equation index)

$$\sum_{i=1}^{4} \left(M_{ki} \left(\eta_i^{(t_n)} - \eta_i^{(t_{n-1})} - k_{max}\, c_i^{(t_n)}\, \eta_i^{(t_n)} \right) \frac{1}{\Delta t} \right) = 0 \tag{23}$$

with the unknown variables $c_i^{(t_n)}$ and $\eta_i^{(t_n)}$ and with the constants M_{ki}, k_{max} and $\frac{1}{\Delta t}$ for one finite element. The values for $\eta_i^{(t_{n-1})}$ are already determined at the previous time step.

If we combine the two equation systems (20) and (23), we obtain a non-linear equations system for one

finite element (with eight equations and the eight unknows $c_{1..4}^{(t_n)}$ and $\eta_{1..4}^{(t_n)}$). Now we are able to solve the system (for example with the Newton method) at each time point t_n and the values for $c_i^{(t_n)}$ and $\eta_i^{(t_n)}$ can be determined.

1.3.4 Mechanics

The finite element discretization for a mechnical system has been already often described, for example by [6]. Because of this fact we will restrict this subsection only to some steps which are important for the oxidation simulation.
After discretization of the continuum, the relationship between strain and displacement (7) can be written as

$$\tilde{\varepsilon}^e = \mathbf{B}\,\vec{d}^e = [\mathbf{B_i}, \mathbf{B_j}, \mathbf{B_m}, \mathbf{B_p}] \tag{24}$$

in which $\tilde{\varepsilon}^e$ is the strain tensor, $\vec{d}^e$ is the displacement vector and i, j, m and p are the four nodes on a single thetrahedron.
The element displacement is defined by the 12 displacement components of the nodes as

$$\vec{d}^e = \left\{ \begin{array}{c} \vec{d}_i \\ \vec{d}_j \\ \vec{d}_m \\ \vec{d}_p \end{array} \right\} \quad \text{with} \quad \vec{d}_i = \left\{ \begin{array}{c} u_i \\ v_i \\ w_i \end{array} \right\} \quad \text{etc.} \tag{25}$$

The submatrix $\mathbf{B_i}$ for the node i is

$$\mathbf{B_i} = \left[\begin{array}{ccc} \frac{\partial N_i}{\partial x}, & 0, & 0 \\ 0, & \frac{\partial N_i}{\partial y}, & 0 \\ 0, & 0, & \frac{\partial N_i}{\partial z} \\ \frac{\partial N_i}{\partial y}, & \frac{\partial N_i}{\partial x}, & 0 \\ 0, & \frac{\partial N_i}{\partial z}, & \frac{\partial N_i}{\partial y} \\ \frac{\partial N_i}{\partial z}, & 0, & \frac{\partial N_i}{\partial x} \end{array} \right] = \left[\begin{array}{ccc} b_i, & 0, & 0 \\ 0, & c_i, & 0 \\ 0, & 0, & d_i \\ c_i, & b_i, & 0 \\ 0, & d_i, & c_i \\ d_i, & 0, & b_i \end{array} \right] \tag{26}$$

with the linear form function $N_i(\vec{x})$ defined as

$$N_i(\vec{x}) = a_i + b_i\,x + c_i\,y + d_i\,z \tag{27}$$

in which a_i, b_i, c_i and d_i are constant geometrical coefficients for the finite element. For example b_i is

$$b_i = -\det \left| \begin{array}{ccc} 1, & y_j, & z_j \\ 1, & y_m, & z_m \\ 1, & y_p, & z_p \end{array} \right| . \tag{28}$$

The entire inner virtual work on a finite element is

$$W_{inner} = \int_V \{\tilde{\varepsilon}^e\}^T\,\sigma^e dV \tag{29}$$

in which the transposed strain tensor is

$$\{\tilde{\varepsilon}^e\}^T = \vec{d}^{e\,T}\,\mathbf{B^T} \tag{30}$$

and the stress tensor (8) can be written as

$$\sigma^e = \mathbf{D}\,\tilde{\varepsilon}^e = \mathbf{D}\,\mathbf{B}\,\vec{d}^e\,. \tag{31}$$

That leads us to the following equation for W_{inner}.

$$W_{inner} = \vec{d^e}^T \int_V \mathbf{B^T D B}\,\vec{d}^e\,dV \tag{32}$$

The outer virtual work on a finite element, caused by the node forces is

$$W_{outer} = \vec{d^e}^T \vec{f}^e\,. \tag{33}$$

On a element the inner work must be equal with the outer work.

$$W_{inner} = \vec{d^e}^T \int_V \mathbf{B^T D B}\,\vec{d}^e\,dV = \vec{d^e}^T \vec{f}^e = W_{outer} \tag{34}$$

With the substituation

$$\mathbf{K^e} = \int_V \mathbf{B^T D B}dV \tag{35}$$

where $\mathbf{K^e}$ is the so-called stiffnes matrix, we obtain a linear equation system for the mechanical problem.

$$\mathbf{K^e}\,\vec{d}^e = \vec{f}^e \tag{36}$$

The most important part is, how the volume increase (5), caused by the chemical reaction of silicon to silicon dioxide, loads the displacement problem.
Due to (11) we obtain the components ε_x^e, ε_y^e and ε_y^e for the strain tensor $\tilde{\varepsilon}^e$ and with

$$\vec{f}_i^e = -\mathbf{B_i^T D}\tilde{\varepsilon}^e\,V^e \tag{37}$$

the relationship between the volume expansion and the node forces is given, and with (36) and (37) the displacements on the nodes is fully determined.

By coupling (20), (23) and (36), a local equation system for one finite element is given, which is a complete numerical formulation of the oxidation process with its oxidant diffusion, chemical reaction and volume increase at any time.

1.4 Simulation Procedure

In the first step of the simulation procedure, we perform a finite element discretization. With this aim in view we split up the Si–$\mathrm{SiO_2}$–body into tetrahedral elements and that results in a tetrahedral grid on the domain. The size of the tetrahedrons and, as a result of that, the number of the finite elements can be influenced by the meshing module.
In the next step we set the initial values for the oxidant concentration C and the normalized silicon concentration η on the grid nodes. For example η must be 1 in a pure silicon domain.
Since the oxidation process is time dependent, the actual simulation time must be reset at the beginning of the simulation.
As shown in Fig. 2, we iterate over all finite elements and build the local equation system for one element for every actual discrete time.

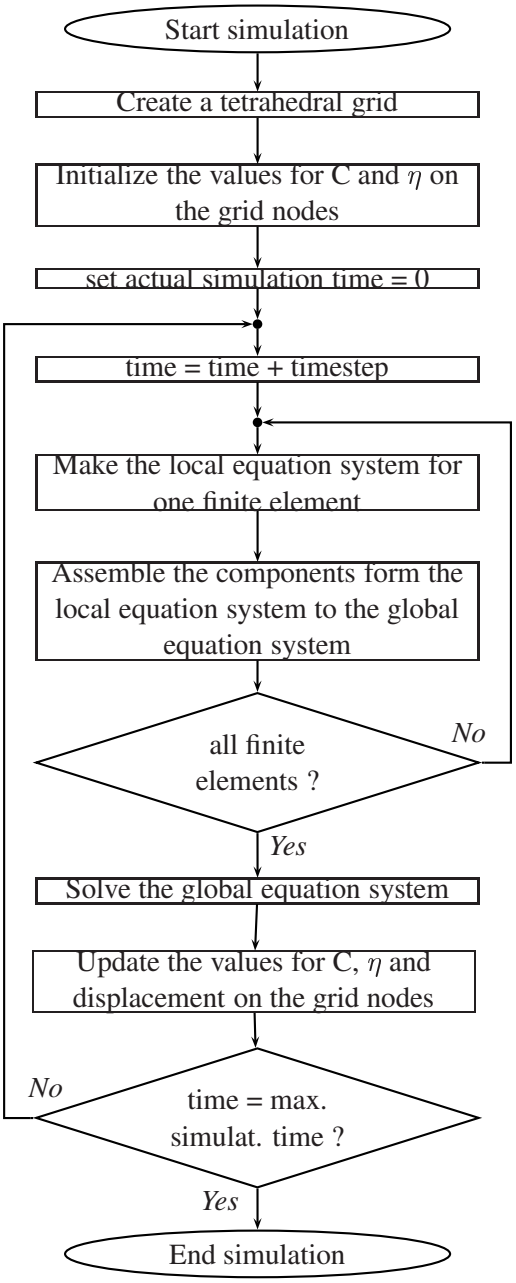

Figure 1: Simulation procedure

The local system describes the oxidation process numerically for one element with the coupled system for diffusion, chemical reaction and the displacement problem. Note, that it would be wrong to solve the relative simple local equation system for one element. The finite element method includes the superpostition principle but not in the way to add up locally calculated results in order to determine the global results. In our case "global" has a spatial meaning and stands for the whole discretized domain.

In order to describe the global oxidation process we need a global coupled equation system. The components of the global equation system are assembled from the local equation system by using the superpostition principle.

After the iteration over all elements is finished, the global assembled equation system is also finished. Now the global non-linear equation system can be solved and we obtain the results for the C, η and displacement values for the global discretized oxdidation process for the actual time step.

With these results we update the values for C, η and displacement on the grid nodes by adding up the new results to the already existing values on the grid nodes and so the values for C, η and displacement always keep pace with the actual simulation time.

If the above described procedure is finished, we increase the actual simulation time and start with the assembling loop again as long as the actual time is equal to the maximum simulation time.

1.5 A Demonstrative Example

As example a silicon body with the initial dimension (0.5 x 0.4 x 0.5) μm as shown in Fig. 2 is oxidized. For the simulation the parameters $C^* = 3 \cdot 10^7\,[\frac{\mathrm{part.}}{\mu\mathrm{m}^3}]$, $D = 0.08\,[\frac{\mu\mathrm{m}^2}{\mathrm{s}}]$, $k_{max} = 40\,[\frac{1}{\mathrm{s}}]$ were chosen, where C^* is the surface oxidant concentration.

In this example only the upper surface of the body has contact with the oxiding atmosphere. As shown in Figs. 2 – 5 the bottom surface is fixed and on the rest of the surfaces there are free mechanical boundary conditions applied.

In the Figs. 2 – 5 the angel of view is always the same and the proportions of the body geometry are also true, so that the displacement effects caused by the volume increase can be watched correct.

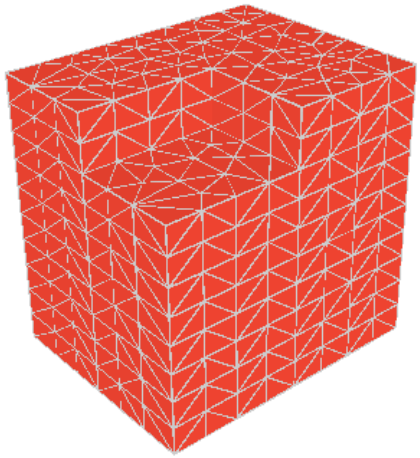

Figure 2: Initial silicon body before oxidation.

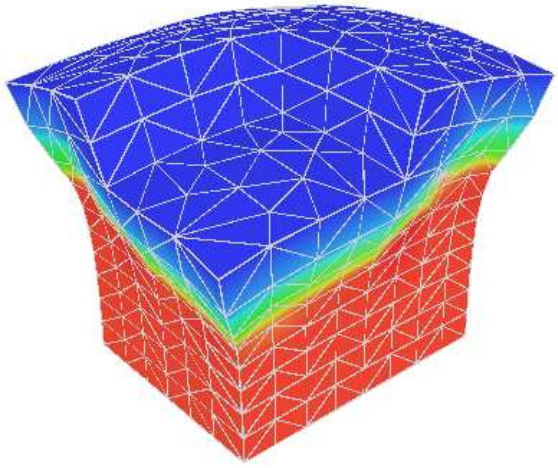

Figure 3: Deformation and silicon dioxide distribution (upper region) at some time $3 * t_1$.

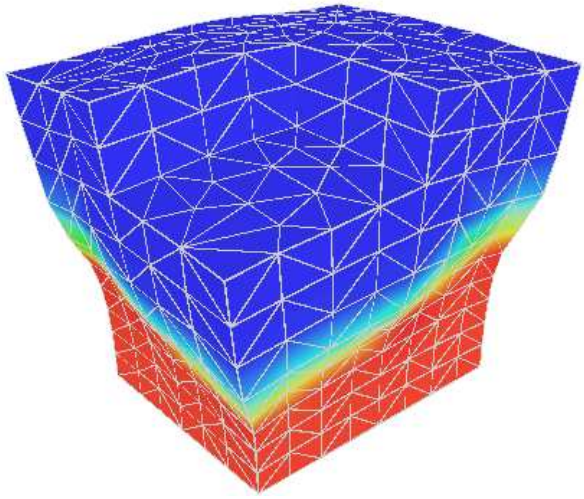

Figure 4: Deformation and silicon dioxide distribution (upper region) at time $7 * t_1$.

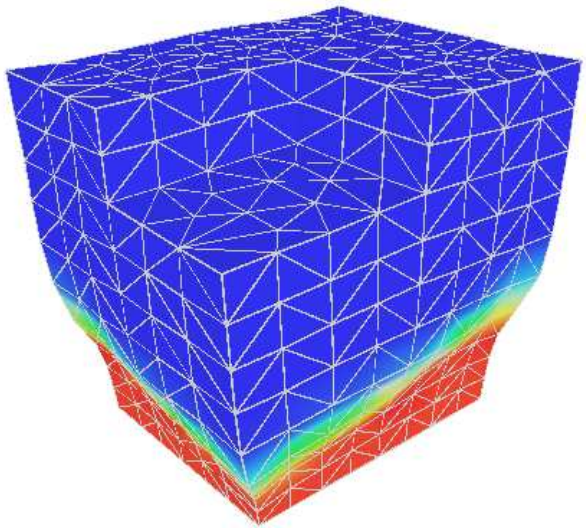

Figure 5: Deformation and silicon dioxide distribution (upper region) at time $13 * t_1$.

1.6 Conclusion

A three-dimensional oxidation model which is based on the finite element technique has been proposed. In this model it is assumed that the interface beween the silicon and oxide is a reaction layer with finite width instead of a sharp interface. In this layer there is a mixture of the three components silicon, oxidants, and oxide.

One of the advantages of this model is that the numerical formulation, consisting of a coupled differential equation system, describes the complete physical oxidation process in a very realistic way.

As demonstrated on a numerical example, this model is a powerful tool to simulate the whole oxidation process on three-dimensional semiconductor structures.

2 Generating Structurally Aligned Grids Using a Level Set Approach

We describe a technique to generate structurally aligned triangular grids. The main advantage of this method is the adjustable propagating speed of the front in different parts of the simulation domain in order to achieve different densities of triangles in each part of the simulation domain. This feature is usually needed in semiconductor device simulation. Other advantages of this technique are twofold: firstly, the grid can be very well adapted to the structures, and secondly, the grid elements fulfill desirable requirements like Delaunay triangulation and the minimum angle criterion. The technique is based on viewing the boundary of the simulation domain as a front which is propagated structurally at different speeds. A smooth propagation is achieved by the level set method by viewing the front as the zero level set of a higher dimensional function whose equation of motion is described by a partial differential equation.

2.1 Introduction

We describe a method to generate structurally aligned triangular grids and illustrate it in two examples. We use the level set method to propagate the boundary of the simulation domain as a front by viewing it as the zero level set of a higher dimensional function with an adjustable speed depending on how fine the triangular grid should be. The equation of motion of this higher dimensional function is given by a partial differential equation, which is approximated by techniques borrowed from the numerical solution of hyperbolic conservation laws which guarantee that the correct entropy satisfying solution will be produced. The evolving front is thus a hypersurface, e.g., a curve in two space dimensions and a surface in three space dimensions. The resulting algorithm can be used to generate two and three dimensional grids around complex bodies containing sharp corners and significant variations in curvatures. We use this technique to generate different grids around a variety of shapes for different device structures.
The most important advantage of this method is the adjustable propagating speed of the front which provides an automatic way for generating grids with different densities of grid cells in particular parts of its domain. The history of two-dimensional process and device simulation leads to the observation that a stable triangulation engine is one of the most important prerequisites for simulation purposes. In the second part of our algorithm the final grid elements are produced using the TRIANGLE program [7, 8]. Furthermore, thereby grids are very well adapted to the structures and are of high quality because we can enforce minimum angle criterion which guarantees that the triangles have angles which are equal or greater than a certain minimum angle and therefor we can well control the shape of the triangles.
Although the level set method has been used for generating structurally aligned grids [9], the method presented there cannot generate anisotropic grids and no condition concerning the quality of the grid, e.g., minimum angles, can be enforced.
The outline of this paper is as follows. Firstly, the basic ideas of the level set method are shortly explained. Secondly, the grid generation algorithm as a combination of the level set method and triangulation is presented. Thirdly, an algorithm for equalizing the length of segments is presented. Finally, examples for two simple initial structures and a real device structure are given.

2.2 The Level Set Method

The level set method [10] provides means for describing boundaries, i.e., curves, surfaces or hypersurfaces in arbitrary dimension, and their evolution in time, which is caused by forces or fluxes normal to the surface. The basic idea is to view the curve or surface in question at a certain time t as the zero level set (with respect to the space variables) of a certain function $u(t, \mathbf{x})$, the so called level set function. Thus the initial surface is the set $\{\mathbf{x} \mid u(0, \mathbf{x}) = 0\}$.

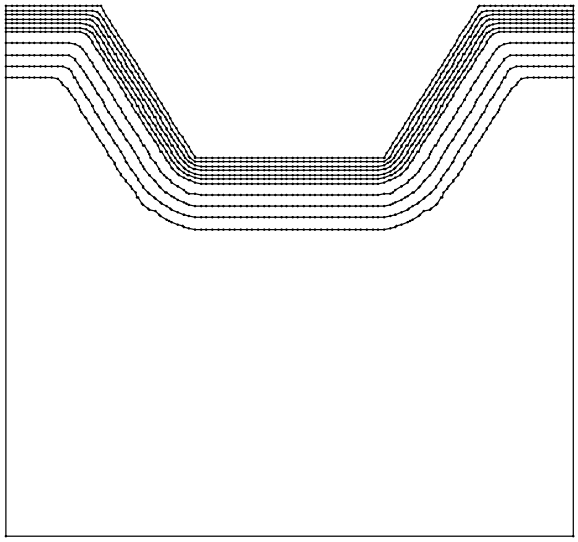

Figure 6: The extracted boundaries at 10 time steps.

Each point on the surface is moved with a certain speed normal to the surface which determines the time evolution of the surface. The speed function $F(t, \mathbf{x})$ generally depends on the time and space variables and we assume for now that it is defined on the whole simulation domain and for the time interval considered. The surface at a later time t_1 shall also be considered as the zero level set of the function $u(t, \mathbf{x})$, namely $\{\mathbf{x} \mid u(t_1, \mathbf{x}) = 0\}$. This leads to the level set equation

$$\begin{aligned} &u_t + F(t, \mathbf{x})\|\nabla_{\mathbf{x}} u\| = 0, \\ &u(0, \mathbf{x}) \quad \text{given} \end{aligned}$$

in the unknown variable u, where $u(0, \mathbf{x})$ determines the initial surface. Having solved this equation the zero level set of the solution is the seeked curve or surface at all later times.
Although in the numerical application the level set function is eventually calculated on a grid, the resolution achieved is in fact much higher than the resolution of the grid, and hence higher than the resolution achieved using a cellular format on a grid of same size.
In summary, first the initial level set grid is calculated as the signed distance function from a given initial surface. Then the speed function values on the whole grid are used to update the level set grid in a finite difference or finite element scheme. Usually the values of the speed function are not determined on the whole domain by the physical models and therefore have to extrapolated suitably from the values provided on the boundary, i.e., the zero level set. A fast and efficient level set algorithm combining extending the speed function and narrow banding was presented in [11, 12]. There a surface coarsening algorithm similar to the one used in this work was described as well.

2.3 Generating the Level Set Structured Triangulated Grid

Our basic philosophy is to advance the front through the simulation domain using different speed functions. Throughout this section we restrict ourselves to two-dimensional grids. At discrete chosen time intervals, zero level set functions are constructed using a boundary extraction algorithm. In our example we have assumed a constant speed for the first 6 time steps and 8/3 times this speed for the next 4 time steps. This is shown in Fig. 6. We can see that the whole simulation domain is now divided into three different parts according to three different grid resolutions depending on the application. An arbitrary number of segments and speed functions can be used if desired.

Based on the edges constructed in the first step the grid generator TRIANGLE is used to obtain a Delaunay triangulation. In this example we demanded that the produced triangles have no angles smaller than 20 degrees. Requiring minimum angles is important since it enables a priori error estimates and estimates of the order of convergence [13].
Fig. 7 shows the triangulated simulation domain. Because of different lengths of the segments which are obtained by each boundary extraction, we can clearly see that this triangulation contains triangles which are too small. An enlargement of this undesirable situation is shown in Fig. 9. We introduce an algorithm for overcoming this problem in the next section.

2.4 The Segment Length Equalizer

To find the origin of this problem we briefly describe the boundary extraction algorithm which uses an interpolation method to find the points of the boundary and represents these as a list of segments with different lengths. Fig. 11 shows a part of the last five steps of advancing the front on a larger scale to show more clearly the varying lengths of the segments. The segments may become arbitrarily small and are the cause of the areas of dense triangles. To overcome this problem we need to ensure that all segments of the boundary have about equal lengths.
We start the algorithm by choosing a certain common length d for all segments. In our example we chose the minimum value of the vertical or horizontal distance between the points of our original rectangular grid which is used in the level set step. The first point of the extracted boundary stays without any changes but to find the second point we have to discern two cases. The first one is that the distance between the second and first point of the originally extracted boundary is equal or greater than our d and in the second one this distance is smaller than d. In the first case we compute the second point of the new boundary in this manner that we get a point which fulfills two restrictions: first, the caused segment must be along the first segment of the originally extracted boundary and second, the length of the new segment must be equal to d. In this case the new segment is a part of the old segment but the length of the new segment is equal or smaller than the old one. In the second case we compute the second point of the new boundary along the next segment of the origin boundary and like the first case fulfilling the length requirement. In this case the new segment is parallel to the second segment of the origin boundary and the length of the new segment is greater than the old one. These steps are iterated until we reach the boundary of the domain. Fig. 12 and Fig. 8 show the resulting segments with the enlargement and triangulated grid after equalizing the lengths of the segments. Furthermore in Fig. 10 a part of Fig. 8 is shown on a larger scale. In Fig. 13, Fig. 14 and Fig. 15 we show a simulation domain with a rectangular advancing front as another example and the resulting grid also with the enlargement.

2.4.1 Grid Generation for a Real Device Structure

Fig. 16 shows the device structure of a trench gate UMOS transistor. This device is useful for power switching at high voltages [14, 15, 16]. Trench gate UMOS transistors also provide advantages because of their geometric layout, i.e., because their inversion and accumulation channel regions are perpendicular to the wafer surface. Hence they enable to maximize the ratio of cell perimeter to area and thus increase packing density. An analytical model for a typical trench gate UMOS transistor is given in [17].
The model is derived using the charge control analysis of the channel and drain drift regions and gradual channel approximation is assumed to be valid in modeling the channel region. The shape of the different junctions is obtained by the doping concentration profile which is modeled with a Gaussian distribution.
For the grid generation we used four boundaries which follow the three junctions. At the $n^+ - p$ junction we used three boundaries in each direction of the initial boundary which follow the junction with a distance of 0.02μm between any two adjacent boundaries.

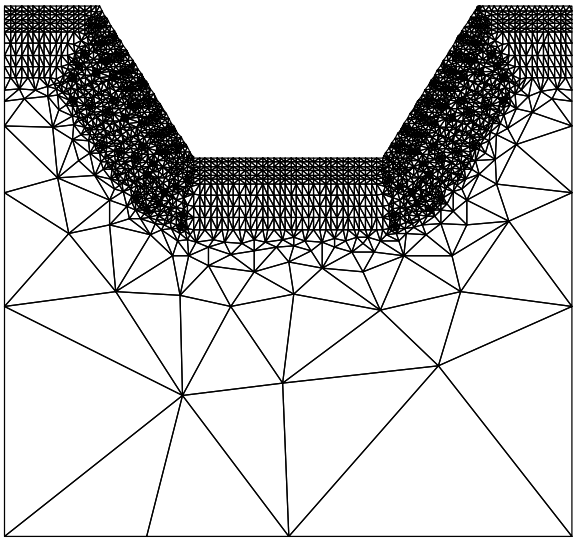

Figure 7: The triangulated grid without using the segment length equalizer.

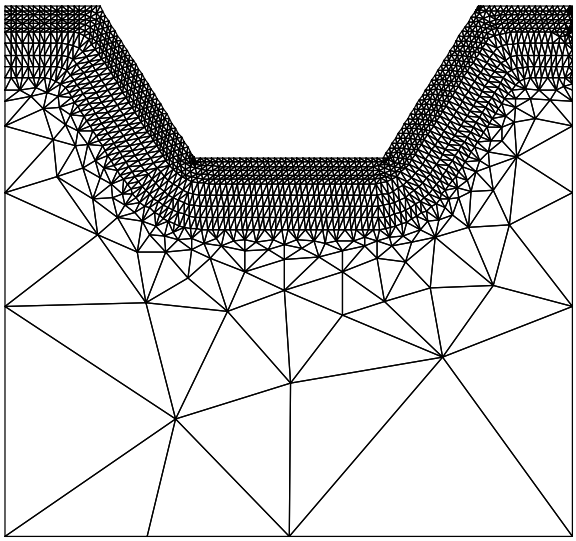

Figure 8: The triangulated grid is caused using the segment length equalizer.

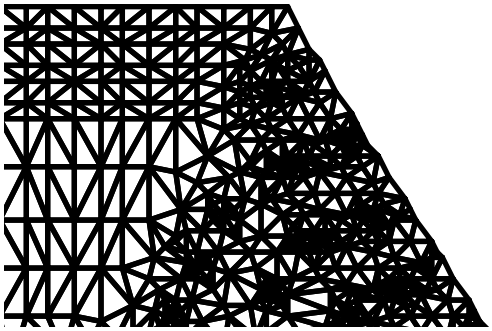

Figure 9: A part of the above grid on a larger scale.

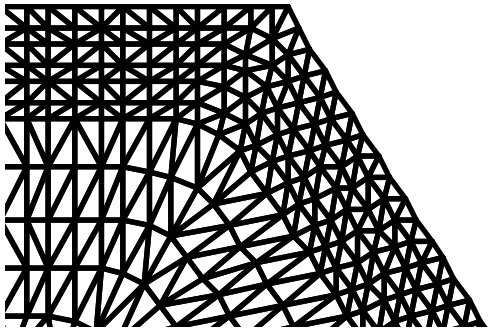

Figure 10: A part of the above grid on a larger scale.

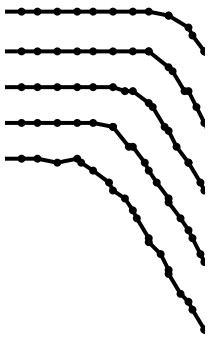

Figure 11: The last five steps of advancing the front is shown partly on a larger scale. The varying lengths of the segments are shown clearly.

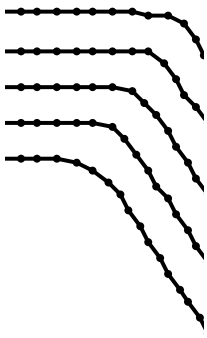

Figure 12: The last five steps of advancing the front is shown partly on a larger scale after equalizing the lengths of the segments. The length of the segments are not more different.

At the $p-n$ junction we used one boundary above and below the initial boundary and a distance of 0.02μm. At the $n-n^+$ junction in the lower part of the device we took into account two boundaries with a distance of 0.5μm going downwards from the initial boundary following the junction. For the last prescribed edges we started at the tight hand side of the p region and moved to the left using three boundaries at a distance of 0.005 μm.
Finally, we applied the TRIANGLE program requiring a minimum angle of 25^o with the prescribed edges as input. The grid produced is shown in Fig. 17, and it resolves very finely the junction areas as demanded.

2.5 Conclusion

A technique for generating structurally aligned triangulated grids using the level set method was described and implemented in two dimensions. In contrast to previously generated structurally aligned grids based on the level set method [9] the anisotropy of the grids and their quality can be controlled. The simulation domain can be divided into parts with different resolutions using adjustable speeds for advancing the front through the simulation domain with level set method. This adjustable grid resolution is essential in semiconductor device simulation where high resolutions are required in certain parts of the simulation domain. Furthermore the grid can very well adapted to different structures. Finally enforcing the minimum angle criterion is important for the numerical behavior of the subsequent finite element calculations and ensures high quality grids. At the same time, the diameter of the triangles may vary over several orders of magnitude within one simulation domain (cf. Fig. 8, Fig. 14, and Fig. 17). Our technique enables to produce triangulated grids for each form of semiconductor device structure with demanded resolution at different junctions (cf. Fig. 17).

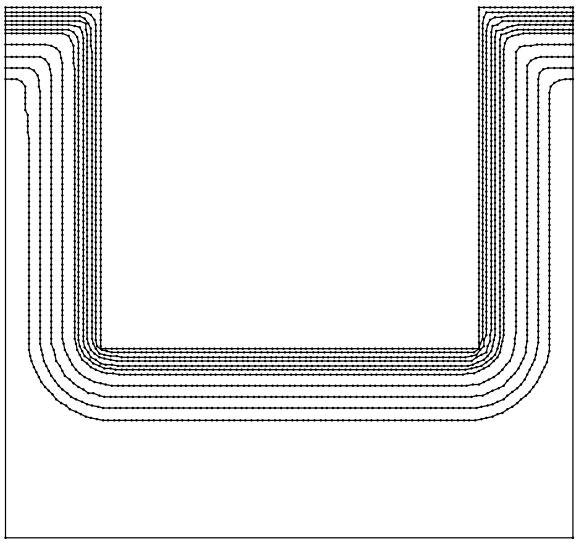

Figure 13: The advancing rectangular front after 10 time steps. As same as Fig. 8 the ratio of the speed in the first 6 steps to the last 4 steps is 3/8.

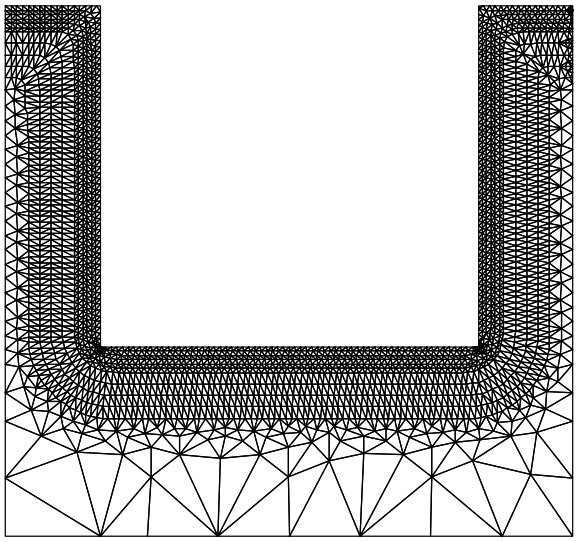

Figure 14: The triangulated grid of simulation domain in Fig. 13.

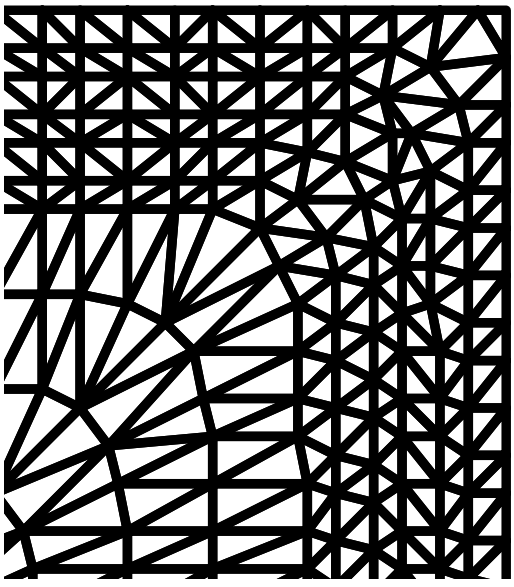

Figure 15: Fig. 14 is shown partly on a larger scale.

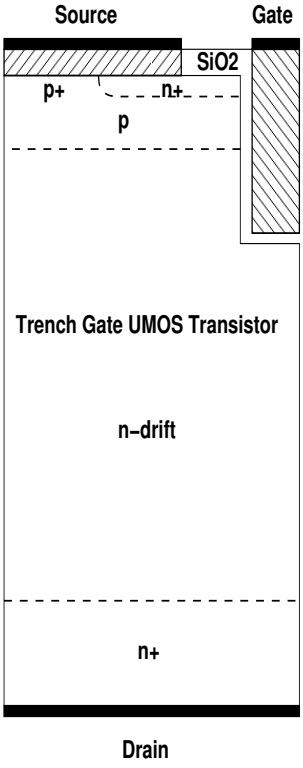

Figure 16: Structure of TMOSFET. The half cell pitch of the device is 2.5μm and its n drift length is about 9.5μm.

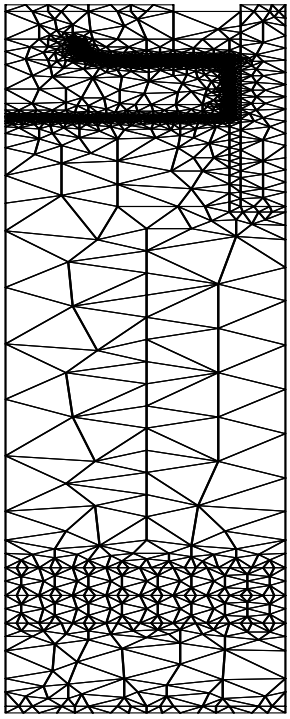

Figure 17: The grid generated for the device in Fig. 16.

3 Rigorous Modeling Approach to Numerical Simulation of SiGe-HBTs

We present results of fully two-dimensional numerical simulations of Silicon-Germanium (SiGe) Heterojunction Bipolar Transistors (HBTs) in comparison with experimental data. Among the critical modeling issues discussed in the paper, special attention is focused on the description of the anisotropic majority/minority electron mobility in strained SiGe grown in Si.

3.1 Introduction

Our SiGe HBT-CMOS integrated process is based on a 0.35 μm mixed-signal CMOS process and includes an additional high-performance analog-oriented HBT module. The applications reach is from circuits for mobile communication to high-speed networks. Using simulation in a predictive manner has been recognized as an integral part of any advanced technology development. In order to satisfy predictive capabilities the simulation tools must capture the process as well as the device physics.

3.2 Physical Modeling

The two-dimensional device simulator MINIMOS-NT [18] can deal with various semiconductor materials and complex geometrical structures. Previous experience gained in the area of III-V HBT modeling and simulation which lead to successful results [19] was a prerequisite to use MINIMOS-NT also for simulation of SiGe HBTs.

3.2.1 Bandgap and Bandgap Narrowing

Modeling of strained SiGe is not a trivial task, since special attention has to be focused on the stress-dependent change of the bandgap due to Ge content [20]. This effect must be separated from the dopant-dependent bandgap narrowing which for itself depends on the semiconductor material composition, the doping concentration, and the lattice temperature [21].

3.2.2 Carrier Mobility

As the minority carrier mobility is of considerable importance for bipolar transistors, an analytical low field mobility model which distinguishes between majority and minority electron mobilities has been developed [21] using Monte Carlo simulation data for electrons in Si. A similar expression is currently implemented in MINIMOS-NT:

$$\mu_n^{\mathrm{maj}} = \frac{\mu_n^{\mathrm{L}} - \mu_{mid}^{\mathrm{maj}}}{1 + \left(\frac{N_{\mathrm{D}}}{C_{mid}}\right)^{\alpha}} + \frac{\mu_{mid}^{\mathrm{maj}} - \mu_{hi}^{\mathrm{maj}}}{1 + \left(\frac{N_{\mathrm{D}}}{C_{hi}^{\mathrm{maj}}}\right)^{\beta}} + \mu_{hi}^{\mathrm{maj}} \tag{38}$$

$$\mu_n^{\mathrm{min}} = \frac{\mu_n^{\mathrm{L}} - \mu_{mid}^{\mathrm{min}}}{1 + \left(\frac{N_{\mathrm{A}}}{C_{mid}}\right)^{\alpha}} + \frac{\mu_{mid}^{\mathrm{min}} - \mu_{hi}^{\mathrm{min}}}{1 + \left(\frac{N_{\mathrm{A}}}{C_{hi}^{\mathrm{min}}}\right)^{\beta}} + \mu_{hi}^{\mathrm{min}} \tag{39}$$

$$\mu_n^{\mathrm{LI}} = \left(\frac{1}{\mu_n^{\mathrm{maj}}} + \frac{1}{\mu_n^{\mathrm{min}}} - \frac{1}{\mu_n^{\mathrm{L}}}\right)^{-1} \tag{40}$$

where μ^{L} is the mobility for undoped material, μ_{hi} is the mobility at the highest doping concentration. μ_{mid}^{maj}, μ_{hi}^{maj}, μ_{mid}^{min}, μ_{hi}^{min}, C_{mid}, C_{hi}^{maj}, C_{hi}^{min}, α, and β are used as fitting parameters. The final low-field electron mobility μ^{LI}, which accounts for a combination of both acceptor and donor doping is given by (40). Fig. 18 demonstrates a good match between the analytical model, our Monte Carlo simulation data, and measurements from [22]-[23] at 300 K for Si.

Monte Carlo simulation which accounts for alloy scattering and the splitting of the anisotropic conduction band valleys due to strain [24] in combination with an accurate ionized impurity scattering model [25], allowed us to obtain results for SiGe for the complete range of donor and acceptor concentrations and Ge contents x. We use the same functional form to fit the doping dependence of the in-plane mobility component for $x = 0$ and $x = 1$ (Si and strained Ge on Si). The material composition dependence is modeled by

$$\frac{1}{\mu(x)} = \frac{1-x}{\mu^{\mathrm{Si}}} + \frac{x}{\mu^{\mathrm{Ge}}} + \frac{(1-x)\cdot x}{C_{\mu}} \tag{41}$$

C_{μ} is a bowing parameter which equals 140 cm^2/Vs and 110 cm^2/Vs for doping levels below and above C_{mid}, respectively. Fig. 19 shows the in-plane minority electron mobility in $Si_{1-x}Ge_x$ as a function of x at 300 K for different acceptor doping concentrations. The model parameters used for SiGe at 300 K are summarized in Table 1.

The component of the mobility perpendicular to the surface is then obtained by a multiplication factor given by the ratio of the two mobility components. The good agreement of the model with the measured and the Monte Carlo simulation data, both for in-plane and perpendicular to the surface directions, is illustrated in Fig. 20.

3.3 Simulated Device Structure

The double-base SiGe HBT structures are CVD-grown with emitter areas of 12×0.4 μm^2. The base-emitter junction is formed by Rapid Thermal Processing which causes out-diffusion of Arsenic from the poly-Silicon emitter layer into the crystalline Silicon. The process simulation with DIOS [26] reflects real device fabrication as accurately as possible. The implant profiles as well as the annealing steps are calibrated to one-dimensional SIMS profiles. To save computational resources the simulation domain covers only one half of the real device which is symmetric and the collector-sinker and the base poly-Silicon contact layer are not included in the structure.

All important physical effects, such as surface recombination, impact ionization generation, and self-heating, are properly modeled and accounted for in the simulation in order to get good agreement with measured forward (Fig. 21) and output characteristics (Fig. 22) using a concise set of models and parameters. In contrast, simulation without including self-heating effects cannot reproduce the experimental data, especially at high power levels.

The only fitting parameters used in the simulation are the contribution of doping-dependent bandgap narrowing to the conduction band (here about 80% and 20% for donor and acceptor doping, respectively), the concentration of traps in the Shockley-Read-Hall recombination model (here 10^{14} cm^{-3}), the velocity recombination for holes (here 8200 cm/s) in the polysilicon contact model [27] used at the emitter contact, and the substrate thermal resistance.

A closer look at the increasing collector current $\mathrm{I_C}$ at high collector-to-emitter voltages $\mathrm{V_{CE}}$ and constant base current $\mathrm{I_B}$ stepped by 0.4 μA from 0.1 μA to 1.7 μA reveals the interplay between self-heating and impact ionization (see Fig. 23). While impact ionization leads to a strong increase of $\mathrm{I_C}$, self-heating decreases it. In fact, both $\mathrm{I_C}$ and $\mathrm{I_B}$ increase due to self-heating at a given bias condition. As the change is relatively higher for $\mathrm{I_B}$, in order to maintain it at the same level, $\mathrm{V_{BE}}$ and, therefore, $\mathrm{I_C}$ decrease.

3.4 Conclusion

Critical issues for numerical modeling of SiGe devices have been discussed including accurate models for bandgap narrowing and minority/majority electron mobility in strained SiGe. Good agreement was obtained between simulation and experimental DC-results (forward and output characteristics) of SiGe HBTs. The newly established models are beneficial for future process development.

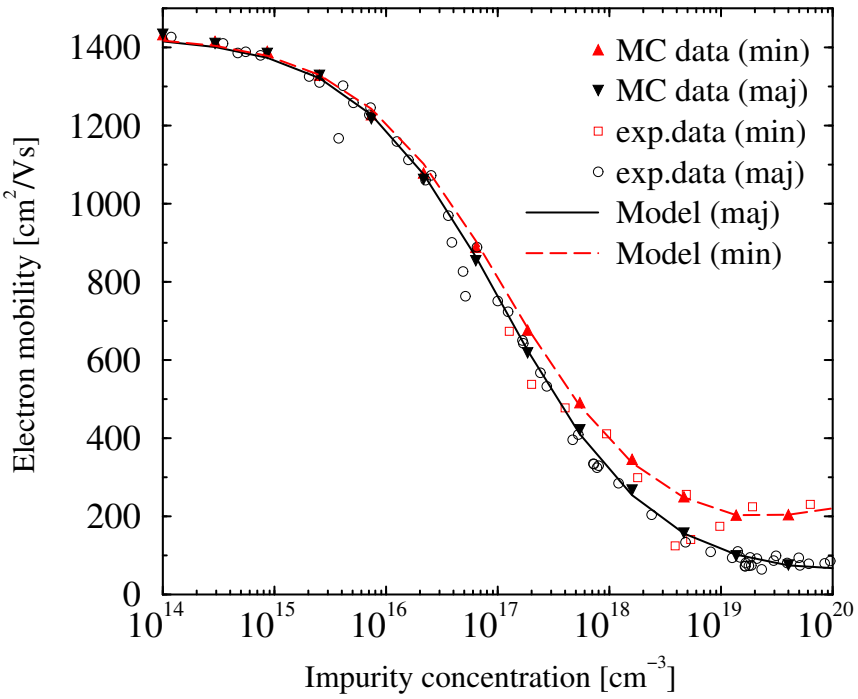

Figure 18: Majority and minority mobility in Si at 300 K: Comparison between Monte Carlo simulation data and experimental data.

Table 1: Parameter values for the majority/minority electron mobility at 300 K.

Parameter	Si	Ge(on Si)	Unit
μ_n^{L}	1430	560	$\mathrm{cm}^2/\mathrm{Vs}$
μ_{mid}^{maj}	44	80	$\mathrm{cm}^2/\mathrm{Vs}$
μ_{hi}^{maj}	58	59	$\mathrm{cm}^2/\mathrm{Vs}$
μ_{mid}^{min}	141	124	$\mathrm{cm}^2/\mathrm{Vs}$
μ_{hi}^{min}	218	158	$\mathrm{cm}^2/\mathrm{Vs}$
α	0.65	0.65	
β	2.0	2.0	
C_{mid}	1.12e17	4.0e17	cm^{-3}
C_{hi}^{maj}	1.18e20	4.9e18	cm^{-3}
C_{hi}^{min}	4.35e19	5.4e19	cm^{-3}

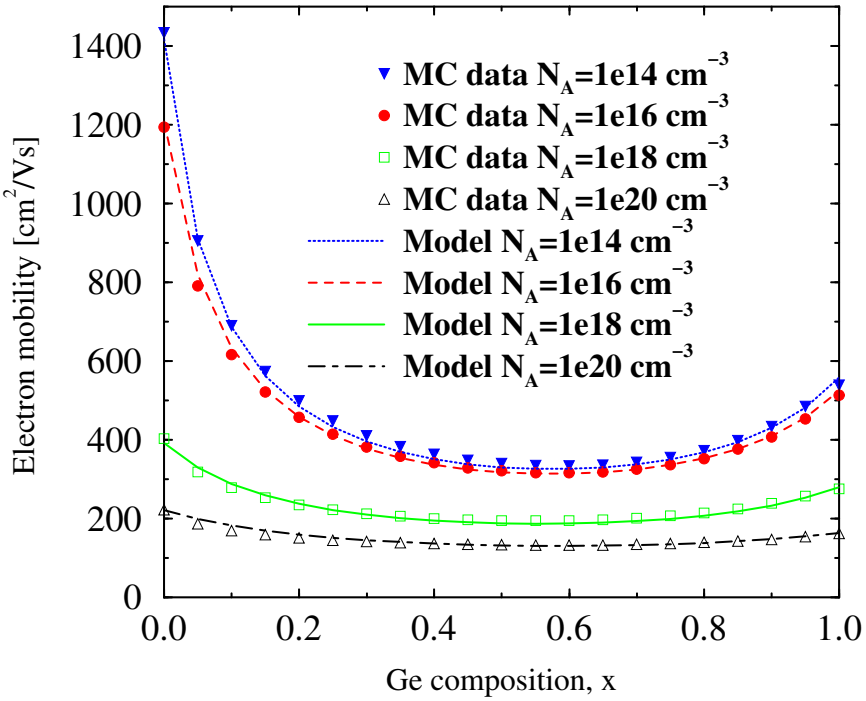

Figure 19: Minority electron mobility in $Si_{1-x}Ge_x$ as a function of x for in-plane direction: The model gives good agreement with Monte Carlo simulation data.

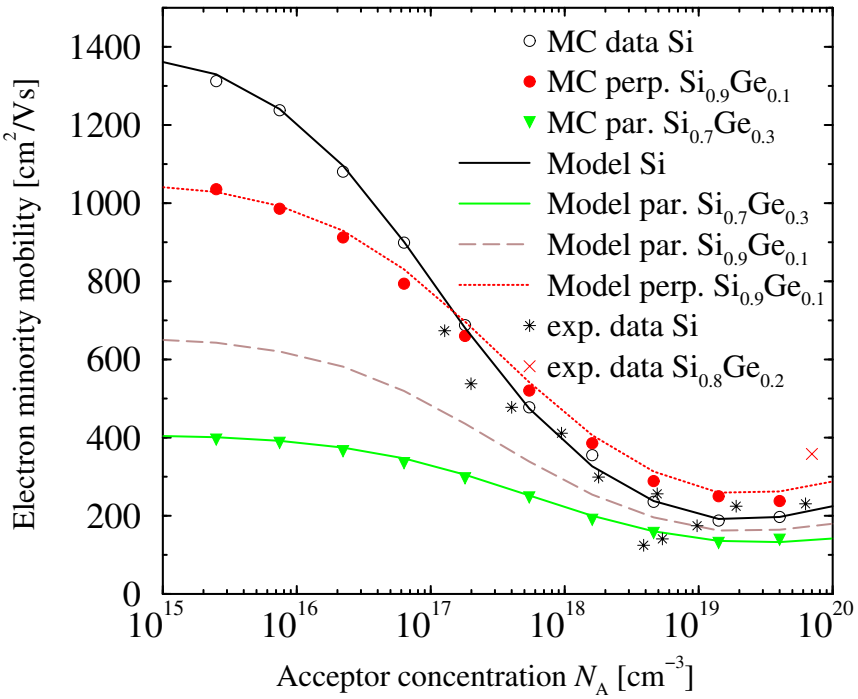

Figure 20: Minority electron mobility in $Si_{1-x}Ge_x$ as a function of N_A and x: The model gives good agreement with measurements and Monte Carlo simulation data both for in-plane and perpendicular to the surface directions.

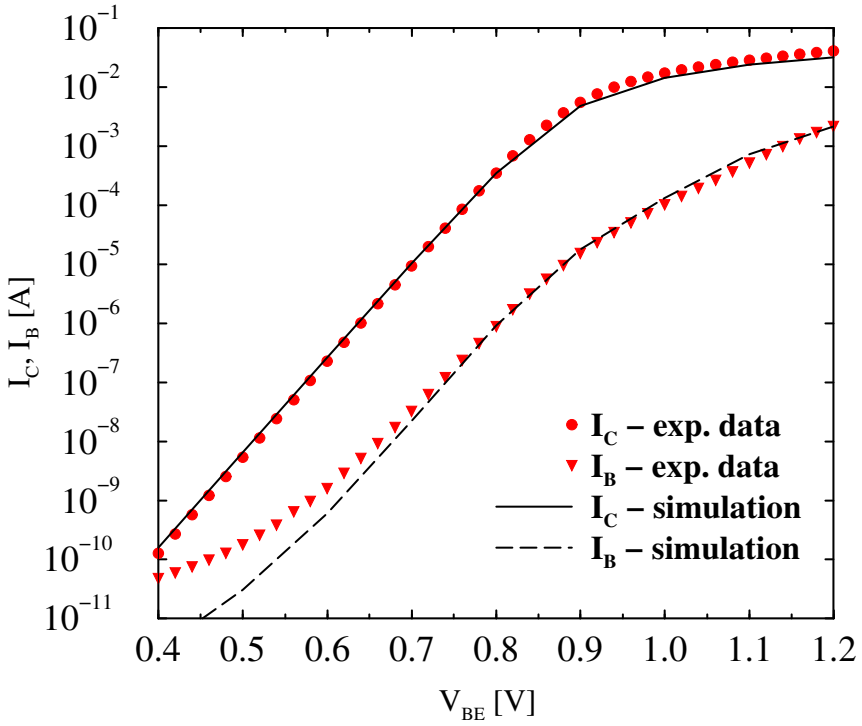

Figure 21: Forward Gummel plots at $V_{CB} = 0$ V: Comparison between measurement data and simulation at room temperature. The bandgap is one of the crucial modeling parameters.

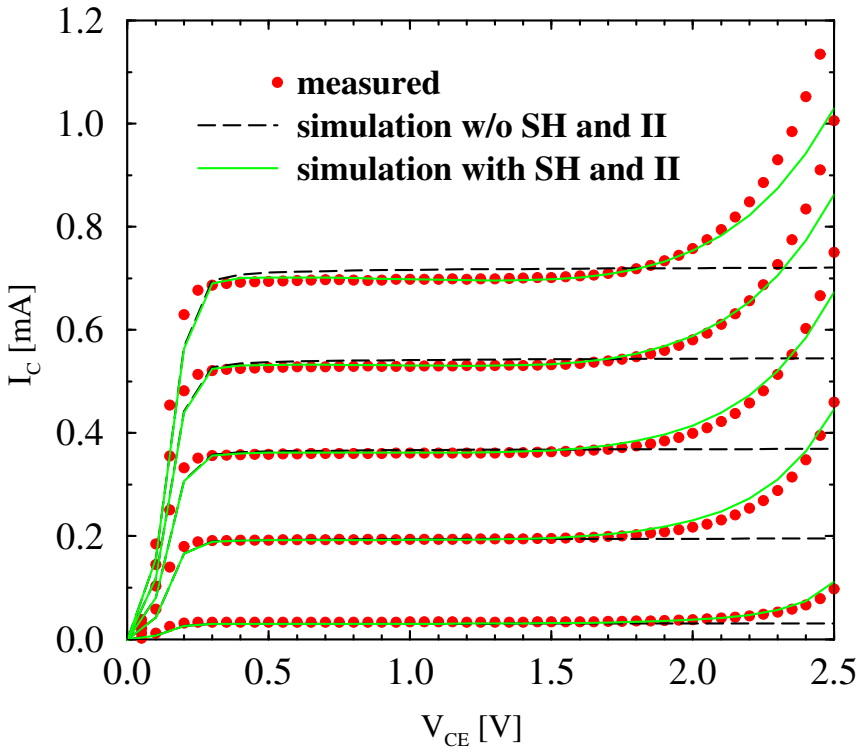

Figure 22: Output characteristics: Simulation with and without self-heating (SH) and impact ionization (II) compared to measurement data. I_B is stepped by 0.4 μA from 0.1 μA to 1.7 μA.

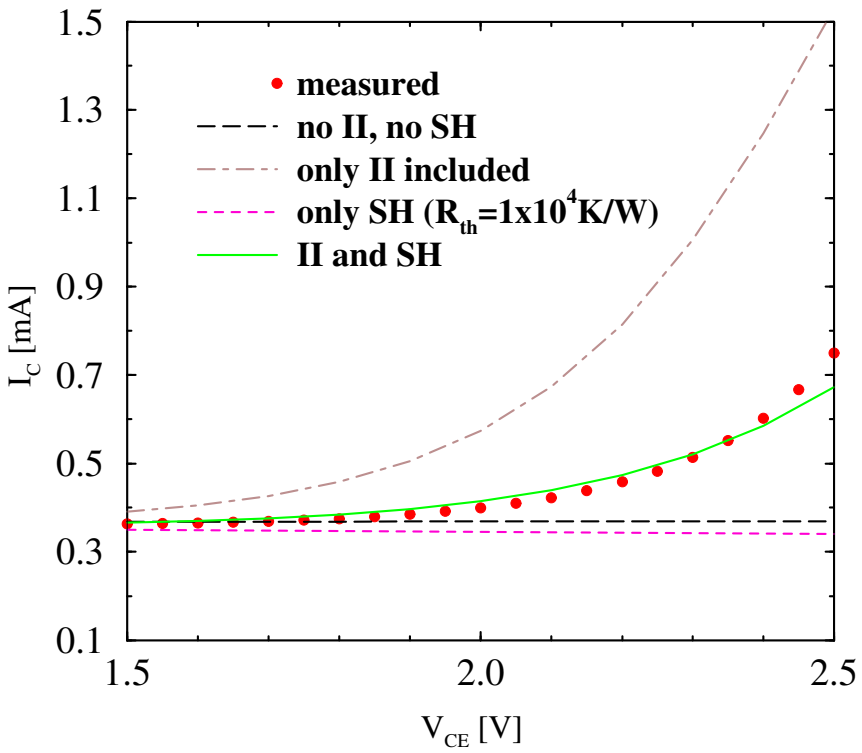

Figure 23: Output characteristics for $I_B = 0.9\ \mu A$: A closer look at the increasing I_C at high V_{CE} reveals the interplay between self-heating (SH) effect and impact ionization (II) generation.

4 Direct Extraction Feature for Scattering Parameters of SiGe-HBTs

We present a direct approach to obtain scattering parameters S-parameters and other derived figures of merit of SiGe-HBTs by means of small-signal (AC) analysis. Therefore, an additional simulation mode has been implemented in the three-dimensional device simulator Minimos-NT [18]. Several additional features are provided for efficiently obtaining various small-signal parameters. The accuracy of the results is proven by analytical methods and by comparison with measurements.

4.1 Introduction

Since advanced SiGe techniques allow competitive performance of high frequency devices in markets that were prior the object of other materials, small-signal analysis by means of simulation of these devices becomes more important. The idea of small-signal AC device characterization is to analyze the relationship between small (in terms of the limit of the amplitude to avoid harmonic generation) sinusoidal contact currents and voltages superimposed upon a DC device operating point obtained by a steady-state simulation mode. S-parameter sets which are widely used for RF circuit design, are one particular result of a small-signal analysis. The advantage over Y-parameters is that normalized incident and reflected waves are used to characterize the operation of the two-port network. Thus, no short circuit is required, which often cannot be achieved because the parasitics cause unstable devices and thus prevent measurements.
The current version of the device simulator Minimos-NT [18] has been equipped with an efficient feature for obtaining intrinsic admittance and scattering parameters, which can then easily be converted to other parameter sets, such as Z- or H-parameters. For example, it is common practice to use the parameter h_{21} to extract the cut-off frequency f_T. Hence, a direct small-signal analysis of complex structures can crucially ease device design and circuit development.

4.2 Physical Modeling

The physical models implemented in Minimos-NT allow advanced simulation of heterostructure devices [19], since all important physical effects such as bandgap narrowing, surface recombination, transient trap recombination, impact ionization, self-heating, and hot electron effects are taken into account. In addition, we use an anisotropic electron mobility model. The simulator deals with different complex structures and materials, such as Si, Ge, SiGe, GaAs, AlAs, InAs, GaP, InP, their alloys and non-ideal dielectrics.
The models are based on experimental or Monte Carlo simulation data and cover the whole material composition range. The carrier transport, generation-recombination, and the self-heating models take also the transient contributions into account and, therefore, the same models are used for the small-signal simulation mode in our approach.

4.3 The Small-Signal Simulation Mode

A small-signal simulation mode can be based on several approaches, e.g. Fourier decomposition and applying quasi-static or equivalent-circuit parameter models. These approaches commonly use a transient simulation mode as shown in Fig. 24. The time derivatives are usually discretized by a backward Euler discretization, and thus a high number of steps has to be performed to achieve sufficient accuracy. For that reason the time consumption is usually reduced by extracting an equivalent circuit using the information of only one frequency.
Our small-signal analysis mode is based on the S^3A approach presented in [28]. After a conventional

DC step at a given operating point the simulator is switched to the simulation mode in the frequency domain, where the device is excited by a complex sinusoidal perturbation of infinitesimal amplitude. For example, the electron current continuity equation can be symbolically given as $F(V, n, p) = \mathrm{d}G(n(t))/\mathrm{d}t$, with nonlinear functions F and G. The time-dependent vector function of electron concentration $n(t)$ is then substituted by $n(t) = n_0 + n \cdot e^{j\omega t}$. The system is thus Fourier transformed ($dt \rightarrow j\omega$) and the final small-signal approximation is obtained by terminating the Taylor series expansion after the linear part. In comparison to transient methods [29, 30] performance is better (only one equation system per frequency step has to be solved) and the results are more accurate, since approximations are not required. As was shown in [31] the speed-up can be up to 98%.
This approach requires the ability for solving complex-valued linear equation systems, for which several methods can be applied. One possibility is to reuse a real-valued assembly and solver system, split the real- and imaginary part and solve both systems separately. In terms of memory consumption this approach has, especially for three-dimensional simulations, severe disadvantages, since the dimension doubles causing a fourfold-sized system matrix. Thus, the computational effort for factorization can be excessive. In [28] iterative methods like block-Gauss-Seidel or block-SOR are suggested for reducing this effort. Another approach implemented in Minimos-NT is to provide a template-based assembly and solver system (BiCGStab and GMRES(m) iterative solvers) capable to handle both real- and complex-valued systems. The real-valued variant was kept due to performance considerations.
In addition to this already established small-signal analysis method, we have implemented a feature for direct extraction of intrinsic (de-embedded) Y- and S-parameters. As an optional feature these parameters can be transformed into extrinsic parameters in order to take parasitics introduced by the measurement set-up into account.

4.4 The Simulation Device and Results

The investigated $0.4 \times 12\mu\mathrm{m}^2$ SiGe-HBT device structure is obtained by process simulation [26]. For DC simulations usually only the active part (base and emitter area, collector contact was moved to the bottom) of the device is required. For that reason the collector area was cut to speed-up simulations due to the reduced grid size. Only half of the real structure was simulated because of symmetry. Fig. 25 shows a comparison of simulated and measured forward Gummel plots at $\mathrm{V_{CE}}$=1 V.
Note that it is absolutely necessary for AC simulations to take the complete device structure into account. Thus, for the reduced device structure the important capacitances between collector and substrate $\mathrm{C_{CS}}$ as well as between base and collector $\mathrm{C_{BC}}$ could not be reproduced. In addition, the correct base and collector resistances are missing. There are two possibilities to overcome this problem. Either the missing parts are approximated by introducing linear elements in a postprocessing step or a larger or even complete structure is used for AC simulations. The first option allows faster simulations but gives approximated results. The second one produces more accurate results and does not require a postprocessing step, but takes much more time: in the example the computational effort of device simulation is 2.5 times higher.
In Fig. 26 and Fig. 27 both options are compared: in the frequency range between 50 MHz and 31 GHz measured and simulated S-parameters at $\mathrm{V_{CE}}$ =1 V and current densities $\mathrm{J_C}$ = 28 kA/cm^2 and $\mathrm{J_C}$ = 76 kA/cm^2 are shown. For the first option we embedded the device structure in a circuit containing the following elements: $\mathrm{C_{CS}} = 50$ fF, $\mathrm{C_{BC}} = 20$ fF, $\mathrm{R_B} = 15\,\Omega$ and $\mathrm{R_C} = 27\,\Omega$. Their values were experimentally estimated. The results of the second option are the intrinsic parameters only.
The quality of the simulated (intrinsic) Y-parameters is proven by calculating the row and column sums of the admittance matrix, which have to be zero according to Kirchhoff's laws. The simulation yields errors of about 10^{-16} A/V for typical matrix entries of 10^{-3} A/V. The transformation to intrinsic S-parameters is completely analytical (also the accounting for the capacitances) and, thus, the results can be directly compared to the measurement data. Since the measurement environment accounts for the parasitics, no

transformation to extrinsic parameters is necessary.
For the same device we calculated the matched gain g_m and the short-circuit current gain h_{21} in order to extract the figures of merit f_T (short-circuit cut-off frequency) and f_{max} (maximum oscillation frequency) found at the intersection with 0 dB (unity gain point). Fig. 28 and Fig. 29 show the comparison of our results and the corresponding measurement data. While the measurement data ends at 31 GHz the simulation could be extended to frequencies beyond this intersection. The peak of the f_T-curve in Fig. 28 corresponds exactly to the frequency at the respective intersection in Fig. 29.
Fig. 28 shows also the effect of the introduction of anisotropic electron mobility. In addition, results obtained by a commercial device simulator (Dessis [32]) using default models and parameters are included for comparison.

4.5 Conclusion

The agreement in order of the typical curve characteristics with measured and transformed data proves the efficiency of our approach. In addition, the performance speed-up in comparison to alternatives is an important advantage. However, a general approach to match simulated results and measured data perfectly has to comprise a proper physical modeling of the complete device since there are no extrinsic fitting parameters available. We are able to extract various sets of small-signal parameters as well as related figures of merit by means of simulation with Minimos-NT.

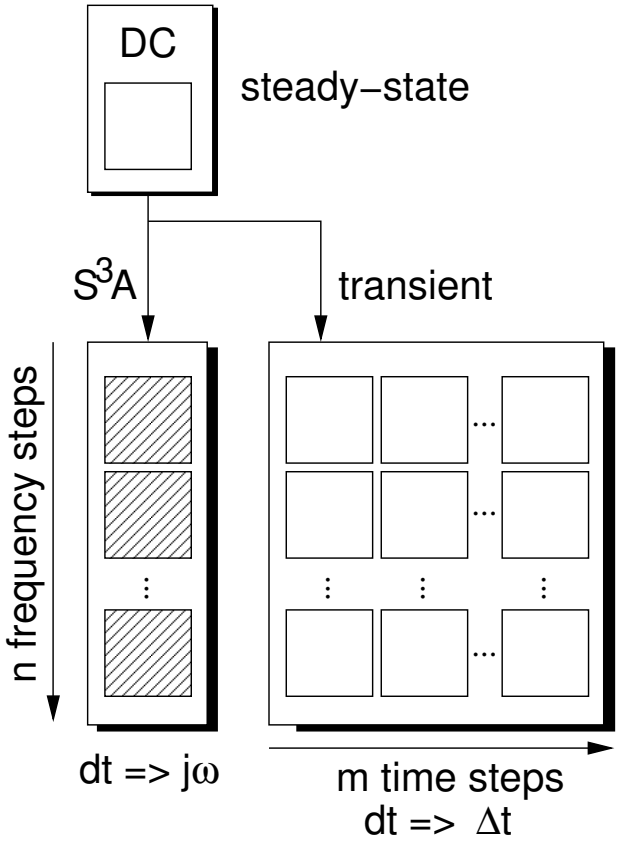

Figure 24: Comparison of small-signal and transient approaches. The dashed rectangles of the S^3A approach symbolize complex-valued equation systems, the other real-valued ones.

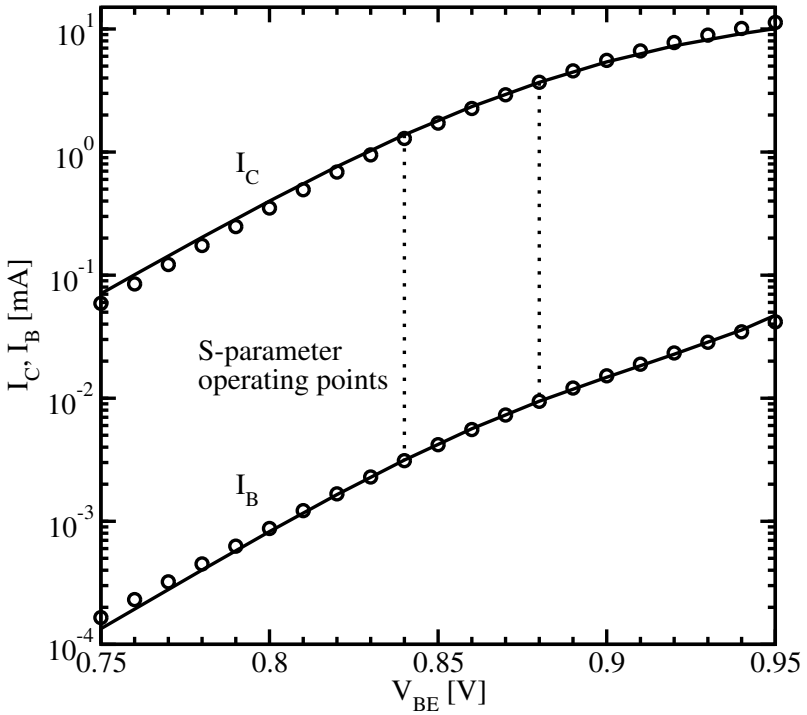

Figure 25: Comparison of simulated and measured forward Gummel plots at V_{CE}=1 V.

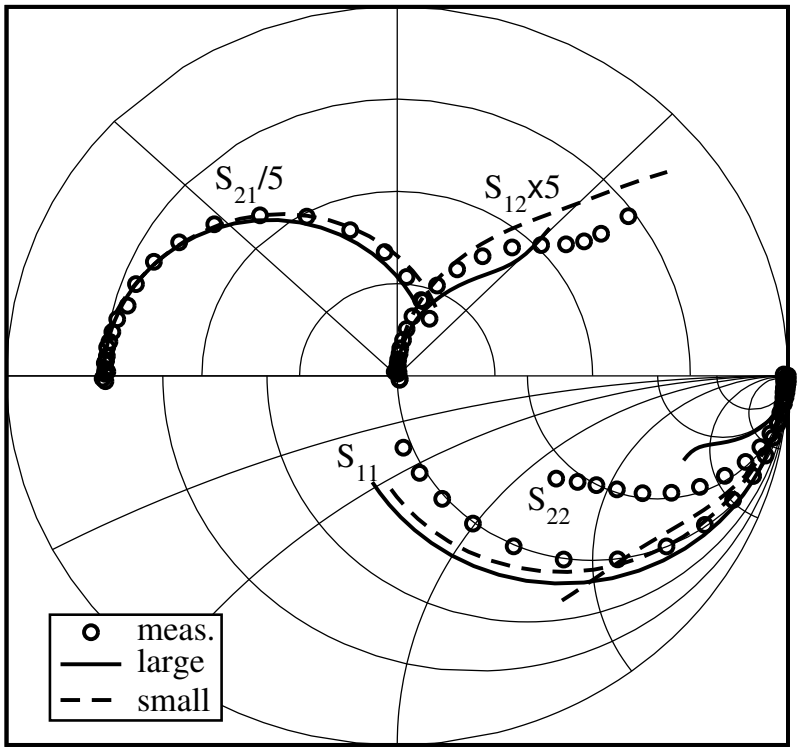

Figure 26: S-parameters in a combined Smith chart from 50 MHz to 31 GHz at V_{CE} = 1 V and current density J_C = 28 kA/cm^2. For simulations either a larger device structure or a small one embedded in a circuit is used.

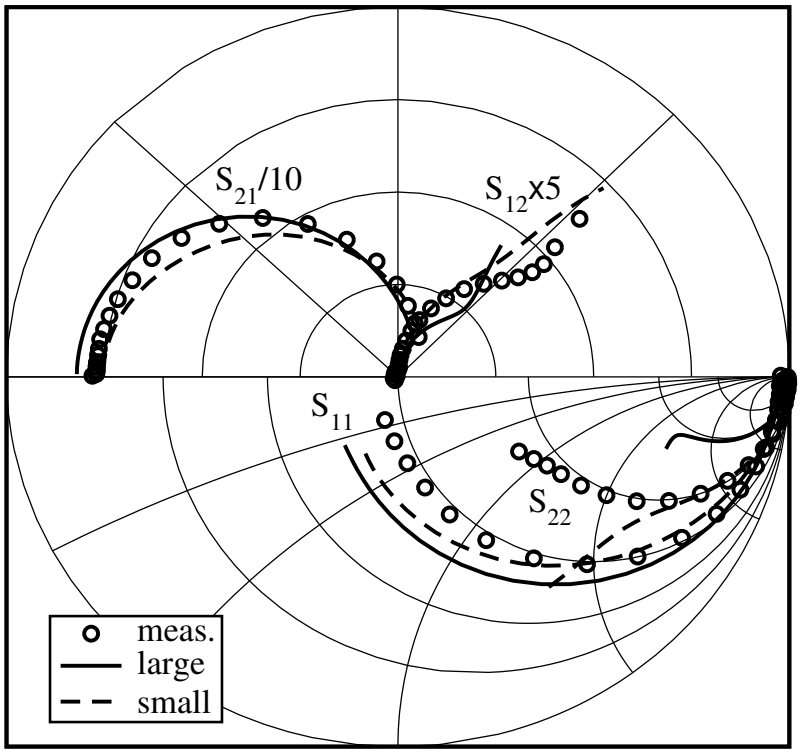

Figure 27: S-parameters in a combined Smith chart from 50 MHz to 31 GHz at V_{CE} = 1 V and current density J_C = 76 kA/cm^2.

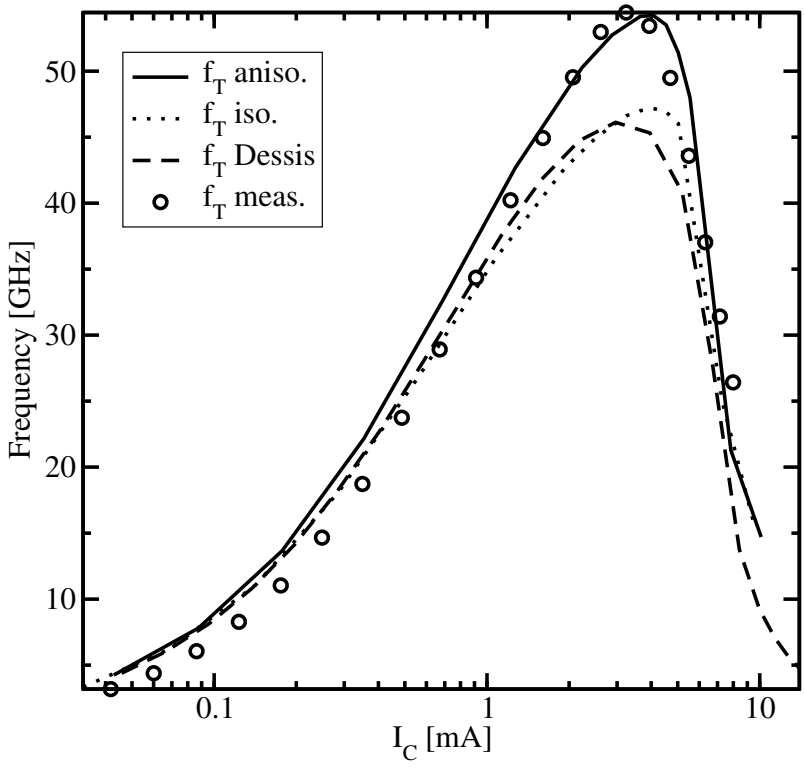

Figure 28: Cut-off frequency f_T versus collector current I_C at V_{CE} = 1 V.

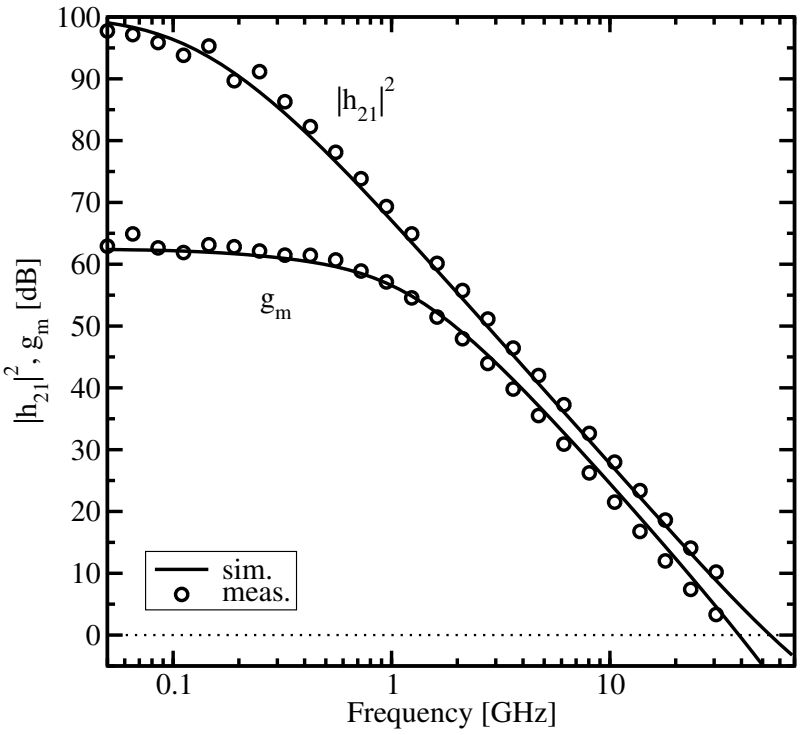

Figure 29: Short-circuit current gain h_{21} and matched gain g_m vs. frequency at V_{CE} = 1 V and current density J_C = 76 kA/cm^2.

5 Statistical Analysis for the 3D Ion Implantation Simulation

Without a proper statistical analysis of the simulation output data, it is not possible to assess the statistical accuracy of three-dimensional Monte Carlo simulation results. The Monte Carlo technique applied to the simulation of ion implantation produces a statistical fluctuation of the doping profile, in particular in the three-dimensional case. The statistical accuracy is determined basically by the number N of simulated ion trajectories. It depends also on the variation of the ion concentration up to several orders of magnitudes in the simulation domain. The theoretical simulation error of order $1/\sqrt{N}$ has been expectedly verified by several simulation experiments with different N. The paper describes the application of statistical methods in order to evaluate the accuracy of three-dimensional ion implantation results compared to one-dimensional results. We propose a method to determine the number of trajectories required to obtain a specified precision in a three-dimensional Monte Carlo simulation study.

5.1 Introduction

Ion implantation is the state-of-the-art method for doping semiconductors because of its high controllability. The small dimensions of modern semiconductor devices have led to simulation applications which require a high accurate and full three-dimensional treatment. Since the process of ion implantation has a statistical nature, it is straightforward to use statistical methods to simulate it on computers. The most important of such methods is the Monte Carlo method which is based on applying random behavior at an atomistic level [33] [34].
Particularly, the position where an ion hits the crystalline target is calculated using random numbers. Furthermore, the lattice atoms of the target are in permanent movement due to thermal vibrations. Thus, the actual positions of the vibrating atoms in the target are also simulated using random numbers. The trajectory of each implanted ion is determined by the interactions with the atoms and the electrons of the target material. The final position of an implanted ion is reached where it has lost its complete energy. The accuracy of the simulation is mainly determined by the complexity of the models that describe the physical behavior. These models are applicable for a wide range of implantation conditions without additional calibration. The number of simulated ions must be considerably increased in order to achieve the same statistical accuracy for three-dimensional simulations as in two dimensions. Therefore the computational effort grows approximately proportional to the surface area of the simulation domain.
A very common mode of operation is to simulate an arbitrary large number N of ion trajectories and then treat the resulting ion concentration estimates as the exact doping profile. In spite of the use of an expensive simulation model misleading results might be obtained, if the random nature of the output data is ignored. From our point of view no in-depth analysis of the simulation accuracy of Monte Carlo process simulations has been carried out so far, and in this work we will present the first comprehensive investigation of the statistical accuracy for three-dimensional Monte Carlo simulations of ion implantation.
The practitioner of a Monte Carlo simulation is always concerned with the computational time and the statistical accuracy of the simulation. Both are related to the simulation rate of convergence to the ”true” value. The standard error in the simulation can be viewed as the standard deviation of the random sample divided by an increasing function of N, the number of simulated ions. We assume that all simulated ions are statistically independent. One way to reduce the simulation error is by using a smart postprocessing of the row data. The statistical fluctuation can be reduced effectively by smoothing the Monte Carlo simulation results in a postprocessing step [35].
The other obvious way to reduce the error is by increasing the number N of simulated ions. The traditional Monte Carlo technique using pseudo random numbers has only a convergence rate of order $1/\sqrt{N}$, which follows from the Central Limit Theorem [36]. This rate is independent of the dimension and depends only on the number N of simulations.

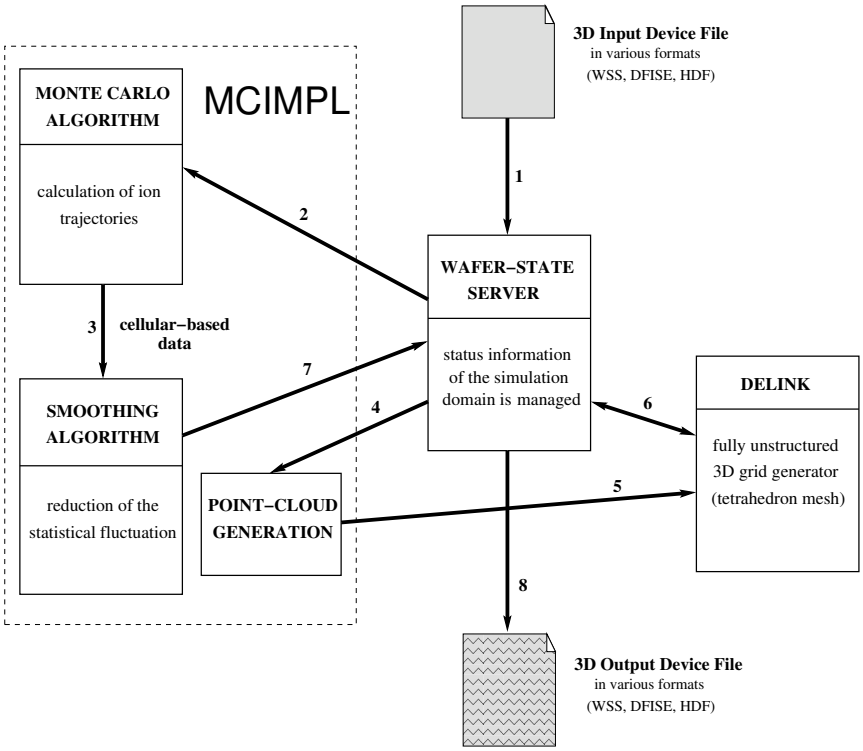

Figure 30: Data flow and involved process simulation tools.

However, there is always a trade-off between the computational effort and the simulation error. In particular with regard to three-dimensional Monte Carlo simulations additional speed-up techniques have to be used in order to get reasonably low statistical noise by practicable long simulation runs. Examples of such speed-up techniques are the trajectory split method and the trajectory reuse method.

5.2 The Simulator

All Monte Carlo simulation experiments were performed with the object-oriented, multi-dimensional ion implantation simulator MCIMPL. The simulator is based on a binary collision algorithm and can handle arbitrary three-dimensional device structures consisting of several amorphous materials and crystalline silicon. In order to optimize the performance, the simulator uses cells arranged on an ortho-grid to count the number of implanted ions and of generated point defects. The final concentration values are smoothed and translated from the internal ortho-grid to an unstructured grid suitable for subsequent process simulation steps, like finite element simulations for annealing processes.
Fig. 30 shows the data flow during the simulation of ion implantation. The simulator MCIMPL is embedded in a process simulation environment by using the object-oriented WAFER-STATE SERVER library [37].
The WAFER-STATE SERVER has been developed in order to integrate several three-dimensional process simulation tools used for topography, ion implantation, and annealing simulations. It holds the complete information describing the simulation domain in a volume mesh discretized format, and it provides convenient methods to access these data. The idea was that simulators make use of these access methods to initialize their internal data structure, and that the simulators report their modifications of the wafer structure to the WAFER-STATE SERVER. Thereby a consistent status of the wafer structure can be sustained during the whole process flow.
The meshing strategy of DELINK follows the concept of advancing front Delaunay methods and produces tetrahedral grid elements [38].

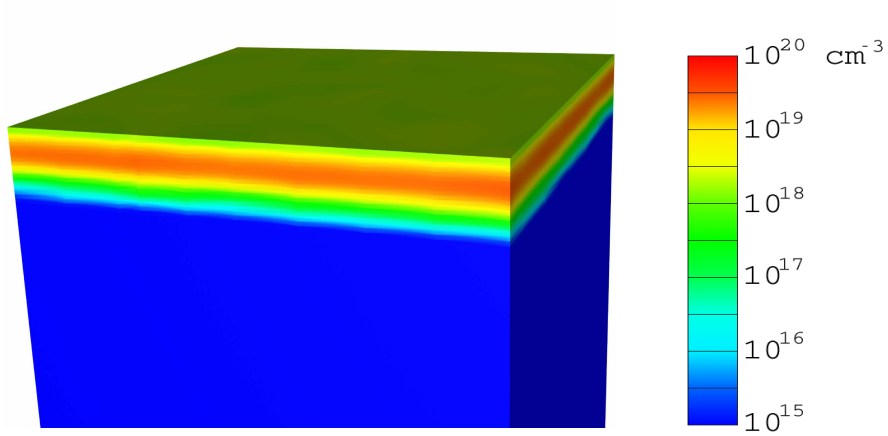

Figure 31: Accurate Monte Carlo simulation result of phosphorus implantation in silicon with $N = 10^7$ simulated ions, an energy of 25 keV, and a dose of 10^{14} cm^{-2}.

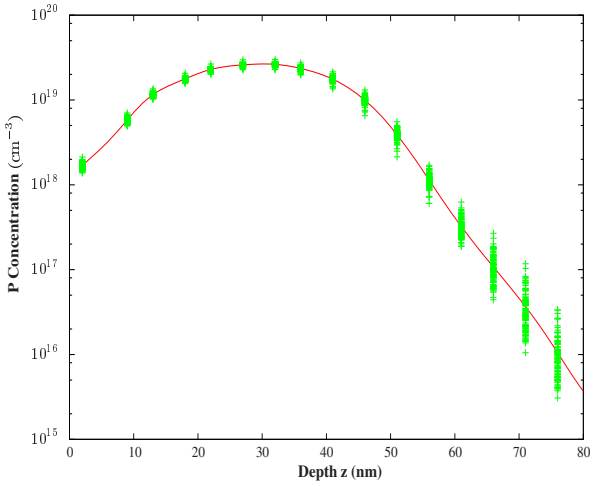

Figure 32: Variability of the 3D result.

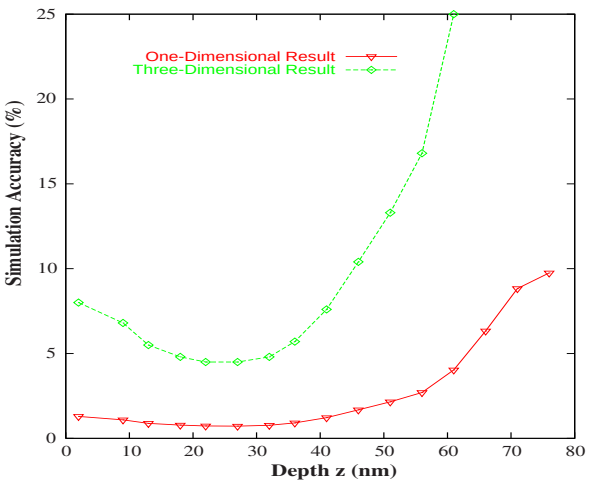

Figure 33: Result evaluation.

5.3 Analysis Method

For the analysis of three-dimensional simulation output, several numerical experiments were performed on a three-dimensional structure equivalent to a one-dimensional problem. In particular, implantations of phosphorus ions into a crystalline silicon substrate were simulated with different N. Fig. 31 shows the three-dimensional result for an accurate simulation with $N = 10^7$ ions. We extracted the z coordinates and the phosphorus concentration values C (vertical direction) from all 2972 grid points of the unstructured grid. This leads to Fig. 32 which demonstrates the statistical fluctuation of the impurity concentration at equal penetration depth z.
The relative standard deviation of the impurity concentration in a plane z = const is a measure for the simulation error of three-dimensional results compared to one-dimensional results. The mean impurity concentration $\overline{C}(n)$ of n grid points at equal location z forms the one-dimensional doping profile. The standard deviation $S(n)$ of a sample defined by the concentration values of n grid points in a plane z = const is given by

$$S(n) = \sqrt{\frac{\sum_{i=1}^{n}[C_i - \overline{C}(n)]^2}{n-1}} \tag{42}$$

$$\sigma = \frac{S(n)}{\overline{C}(n)} \tag{43}$$

The relative standard deviation σ according to (43) is calculated in order to evaluate the three-dimensional result. Fig. 33 demonstrates the statistical accuracy of the three-dimensional result related to the one-dimesional doping profile. Most of the simulated ions come to rest close to the mean projected range R_{p}, causing a small variance there. Due to the very low dopant concentration in deeper regions (typically more than 10^4 times lower than at the maximum), insufficient events lead to an increase of the statistical noise. Being based on random numbers, the results obtained with the Monte Carlo technique are never exact, but rigorous in a statistical sense. The results converge to the used model characteristics. A 90% confidence interval is constructed for the mean, in order to assess the relative error of the one-dimensional doping profile in relation to the model limit value. The half of the approximate 90% confidence interval, $\Delta(n)$, using the t-distribution [36] is given to

$$\Delta(n) = t_{n-1,0.95}\,\frac{S(n)}{\sqrt{n}} \tag{44}$$

The relative statistical error $\epsilon(n)$ for the one-dimensional doping profile can be defined as

$$\epsilon(n) = \frac{\Delta(n)}{\overline{C}(n)} \tag{45}$$

The assessed statistical accuracy of the one-dimensional doping profile according to (45) is also demonstrated in Fig. 33.

The accuracy of the Monte Carlo result is determined by the number of counted ions per cell. The distribution of N ions determines the one-dimensional doping profile by using a scaling factor α:

$$\alpha \int_0^\infty \overline{C}(z)dz = N \tag{46}$$

(46) can be used in order to calculate the factor α by means of numerical integration. For a small volume of the width Δz (cell dimension) the local number N_i of simulated ions is determined by

$$N = \sum_i N_i, \qquad N_i = \alpha\,\overline{C}_i\,\Delta z \tag{47}$$

The division by all cells of a z plane yields to the average ions per cell, which is demonstrated in Fig. 34 for N = 10^7 simulated ions. Each bar is located at the grid points of the internal ortho-grid. The histogram demonstrates that in deep regions only one simulated ion per cell is available in the mean. More and more empty cells at increasing penetration depth downgrade the statistics dramatically. An essential contribution to the accomplished accuracy of the final result is obtained through the reduction of the statistical fluctuation by an implemented smoothing algorithm [35]. This algorithm sweeps a small rectangular grid over the points of the new tetrahedral grid and uses an approximation by generalized Bernstein polynomials. The Bernstein approximation of a concentration value on a new grid point by using the values of cells located close to the new grid point reduces significantly the statistical noise. The bad statistics generated by empty cells can be attenuated by averaging the values of surrounding cells.

We extracted again z coordinates and phosphorus concentration values from all $120 \times 112 \times 20$ cells of the simulation area. Fig. 35 compares the relative standard deviation for $N = 10^6$ ions before and after smoothing. Thus a significant improvement of the statistical accuracy of Monte Carlo results can be achieved through the filter effect of the Bernstein polynomials, which eliminates high-frequency fluctuations from the original data.

Of great importance for the simulation is the weight of an ion, which is defined by the ratio of the number of real ions N_{real} to the number of simulated ions N.

$$Weight = \frac{N_{\mathrm{real}}}{N} \tag{48}$$

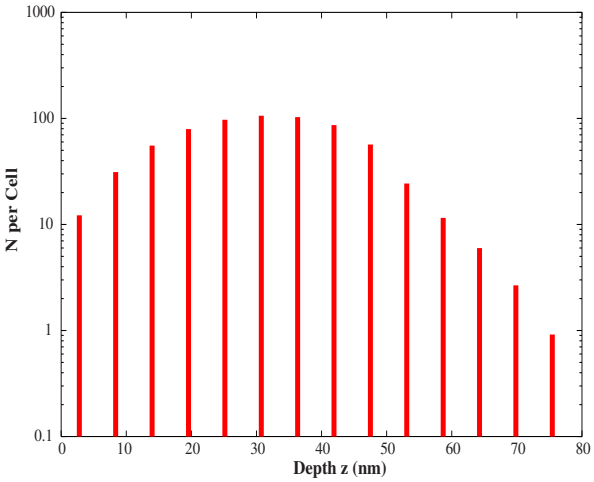

Figure 34: Estimated simulated ions per cell for a total number N = 10^7.

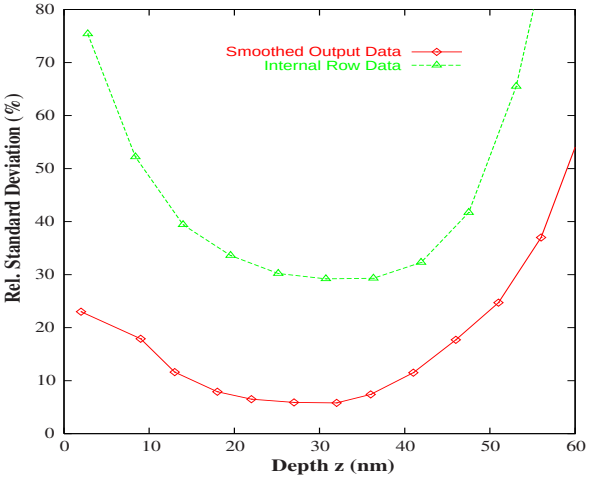

Figure 35: Improvement of the statistical accuracy by smoothing ($N = 10^6$).

In our simulation experiment shown in Fig. 31 the surface dimension is $0.7\mu\mathrm{m}$ x $0.65\mu\mathrm{m}$. With a dose of 10^{14} cm^{-2}, 455000 ions are implanted. With 10^7 simulated ions the weight of an ion results to 0.0455. In practice the real-world implanted doping profile has also a fluctuation due to the statistical nature of the implantation process. In our simulation example a real ion has only a very little weight. Thus the simulation result can be considered as a simulation of averaging over multiple real-world implantations.

5.4 Improvement of the Simulator

The crucial factor for the duration and accuracy of the simulation is the specified number N of ion trajectories as input data of the simulator. One drawback of the fixed-sample-size procedure based on N simulated ions is that the analyst has no immediate control over the precision of the output data. We suggest an improvement of the used fixed-sample-size procedure by determining the duration of the simulation also through a specified precision as input data of the simulator.
The simulation error of the Monte Carlo method is of order $1/\sqrt{N}$. The relationship between the standard deviation σ and the number N of ions is given by

$$\sigma = const \cdot \frac{1}{\sqrt{N}} \tag{49}$$

This relationship has been expectedly verified by simulation experiments with different N and is demonstrated in Fig. 36. It can also be used to assess the number of trajectories required to obtain a specific precision in a Monte Carlo simulation study.

As measure of the simulation accuracy, the desired maximum of the relative standard deviation $\sigma_{\max}$ within the range $2 \cdot \Delta R_\mathrm{p}$ (twice the straggling at the mean projected range) of the doping profile is used. In our experiment of Fig. 31, $2 \cdot \Delta R_\mathrm{p}$ = 22 nm at R_p = 30 nm.
For the calculation of the required N as function of the given standard deviation $\sigma_{\max}$, a parameter γ is used which takes the incident atom species and the implantation energy into account. The following formula can be used to assess the N for a specified surface area A and a desired precision $\sigma_{\max}$:

$$N = \gamma \, \frac{\mathrm{A}}{\mathrm{A}_0} \, \frac{1}{\sigma^2_{\max}}, \qquad A_0 = 0.455 \ \mu\mathrm{m}^2 \tag{50}$$

Fig. 36 demonstrates this relationship for a phosphorus implantation, an ion energy of 25keV, A = A_0, and parameter γ = 15992.

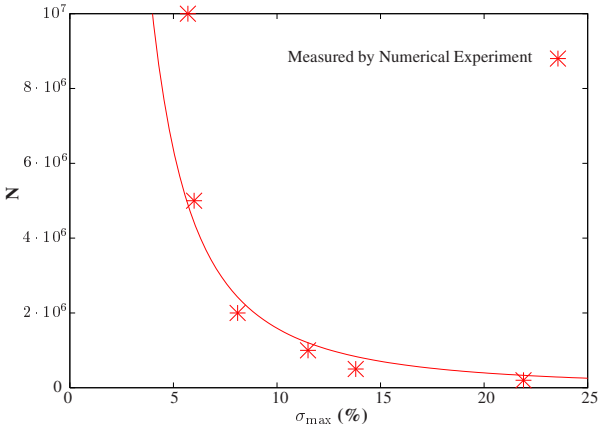

Figure 36: Required N as a function of the desired three-dimensional accuracy .

5.5 Conclusion

The functionality of the three-dimensional Monte Carlo simulator MCIMPL for ion implantation is demonstrated. The statistical fluctuation of the simulation result caused by the stochastic simulation method and the expensive three-dimensional treatment are analyzed. The evaluation of the statistical accuracy for three-dimensional results is performed by the use of statistical methods like calculating the standard deviation or the confidence interval of the output data. The gained insight into the relationships responsible for the statistical accuracy is used in order to achieve a better controllability of the simulator.

References

[1] S. Zelenka. *Stress Related Problems in Process Simulation*. PhD thesis, Federal Institute of Technology, Zurich, Switzerland, 2000.

[2] J. Lorenz, J. Pelka, A. Sachs, A. Seidl, and M. Svoboda. Composite – A complete Modeling Program for Silicon Technology. *IEEE Transactions on CAD*, 4(4):421–430, 1985.

[3] H. Matsumoto and M. Fukuma. Numerical Modeling of Nonuniform Si Thermal Oxidation. *IEEE Transactions on Electron Devices*, 32(2):132–140, 1985.

[4] E. Rank and U. Weinert. A Simulation System for Diffuse Oxidation of Silicon: A Two-Dimensional Finite Element Approach. *IEEE Transactions on CAD*, 9(5):543–550, 1990.

[5] B.E. Deal and A.S. Grove. General Relationship for the Thermal Oxidation of Silicon. *Journal Applied Physics*, 36(12):3770–3778, 1965.

[6] O.C. Zienkiewicz. *The Finite Element Method: Basic Formulation and Linear Problems*, volume 1. McGraw - Hill, Maidenhead, England, 4 edition, 1987.

[7] T.P. Fang and A. Piegl. Delaunay Triangulation Using a Uniform Grid. *IEEE Computer Graphics and Applications*, pp 36–46, 1993.

[8] J.R. Shewchuk. Triangle: Engineering a 2D Quality Mesh Generator and Delaunay Triangulator. *First Workshop on Applied Computational Geometry (Philadelphia, Pennsylvania)*, pp 124–133, 1996. http://www-2.cs.cmu.edu/quake/tripaper/triangle0.html.

[9] J.A. Sethian. Curvature Flow and Entropy Conditions Applied to Grid Generation. *J.Comput.Phys.*, pp 440–454, 1994.

[10] J.A. Sethian. *Level Set Methods and Fast Marching Methods*. Cambridge University Press, Cambridge, 1999.

[11] C. Heitzinger, J. Fugger, O. Häberlen, and S. Selberherr. On Increasing the Accuracy of Simulations of Deposition and Etching Processing Using Radiosity and the Level Set Method. In *European Solid-State Device Research Conference (ESSDERC 2002)*, pp 347–350, Florence, Italy, 2002.

[12] C. Heitzinger and S. Selberherr. On the Topography Simulation of Memory Cell Trenches for Semiconductor Manufacturing Deposition Processes Using the Level Set Method. In *16th European Simulation Multiconference (ESM 2002): Modelling and Simulation*, pp 653–660, Darmstadt, Germany, 2002.

[13] P. Knabner and L. Angermann. *Numerik partieller Differentialgleichungen*. Springer, Berlin, 2000.

[14] C. Bulucea and R. Rossen. Trench DMOS Transistor Technology for High Current (100A Range) Switching. *Solid-State Electron.*, 34(5):493–507, 1991.

[15] K. Shenai. Optimized Trench MOSFET Technologies for Power Devices. *IEEE Trans.Electron Devices*, 39(6):1435–1443, 1992.

[16] K. Dharmawardana and G. Amaratunga. Analytical Model for High Current Density Trench Gate MOSFET. In *Proc. of the 10th International Symposium on Power Semiconductor Devices and ICs (ISPSD 1998)*, pp 351–354, Kyoto, Japan, 1998.

[17] K. Dharmawardana and G. Amaratunga. Modeling of High Current Density Trench Gate MOSFET. *IEEE Trans.Electron Devices*, 47(12):2420–2428, 2000.

[18] Institut für Mikroelektronik, Technische Universität Wien, Austria. *Minimos-NT 2.0 User's Guide*. http://www.iue.tuwien.ac.at/software/minimos-nt.

[19] V. Palankovski, R. Schultheis, and S. Selberherr. Simulation of Power Heterojunction Bipolar Transistors on Gallium Arsenide. *IEEE Trans.Electron Devices*, 48(6):1264–1269, 2001.

[20] J. Eberhardt and E. Kasper. Bandgap Narrowing in Strained SiGe on the Basis of Electrical Measurements on Si/SiGe/Si Hetero Bipolar Transistors. *Materials Science and Engineering*, 89:93–96, 2002.

[21] V. Palankovski, G. Kaiblinger-Grujin, and S. Selberherr. Implications of Dopant-Dependent Low-Field Mobility and Band Gap Narrowing on the Bipolar Device Performance. *J.Phys.IV*, 8:91–94, 1998.

[22] G. Masetti, M. Severi, and S. Solmi. Modeling of Carrier Mobility Against Carrier Concentration in Arsenic-, Phosphorus- and Boron-Doped Silicon. *IEEE Trans.Electron Devices*, ED-30(7):764–769, 1983.

[23] I.Y. Leu and A. Neugroschel. Minority-Carrier Transport Parameters in Heavily Doped p-type Silicon at 296 and 77 K. *IEEE Trans.Electron Devices*, 40(10):1872–1875, 1993.

[24] S. Smirnov, H. Kosina, and S. Selberherr. Investigation of the Electron Mobility in Strained $Si_{1-x}Ge_x$ at High Ge Composition. In *Proc. Intl.Conf. on Simulation of Semiconductor Processes and Devices*, pp 29–32, 2002.

[25] H. Kosina and G. Kaiblinger-Grujin. Ionized-Impurity Scattering of Majority Electrons in Silicon. *Solid-State Electron.*, 42(3):331–338, 1998.

[26] ISE Integrated Systems Engineering AG, Zürich. *DIOS-ISE, ISE TCAD Release 8.0*, 2002.

[27] Z. Yu, B. Ricco, and R. Dutton. A Comprehensive Analytical and Numerical Model of Polysilicon Emitter Contacts in Bipolar Transistors. *IEEE Trans.Electron Devices*, 31(6):773–784, 1984.

[28] S. Laux. Techniques for Small-Signal Analysis of Semiconductor Devices. *IEEE Trans.Electron Devices*, 32(10):2028–2037, 1986.

[29] R. Anholt. HBT S-Parameter Computations Using G-PISCES-2B. *GaAs Simulation and Analysis News*, (4):1–4, 1999.

[30] R. Quay, R. Reuter, V. Palankovski, and S. Selberherr. S-Parameter Simulation of RF-HEMTs. In *Proc. 6th IEEE International Symposium on Electron Devices for Microwave and Optoelectronic Applications (EDMO)*, pp 13–18, Manchester,UK, November 1998.

[31] S. Wagner, V. Palankovski, T. Grasser, R. Schultheis, and S. Selberherr. Small-Signal Analysis and Direct S-Parameter Extraction. In *Proc. 10th IEEE International Symposium on Electron Devices for Microwave and Optoelectronic Applications (EDMO)*, pp 50–55, Manchester, UK, November 2002.

[32] ISE Integrated Systems Engineering AG, Zürich. *DESSIS-ISE, ISE TCAD Release 8.0*, 2002.

[33] G. Hobler and S. Selberherr. An Algorithm for Extracting and Smoothing Three-Dimensional Monte Carlo Simulation Results. *IEEE Transactions on CAD*, 8(5):450–489, 1989.

[34] J.F. Ziegler, J.P. Biersack, and U. Littmark. *The Stopping Range of Ions*. Pergamon Press, 1995.

[35] C. Heitzinger, A. Hössinger, and S. Selberherr. On Smoothing Three-Dimensional Monte Carlo Ion Implantation Simulation Results. *IEEE Transactions on CAD*, 22(7), 2003.

[36] A.M. Law and W.D. Kelton. *Simulation Modeling and Analysis*. McGraw-Hill, USA, 2000.

[37] T. Binder and S. Selberherr. Rigorous Integration of Semiconductor Process and Device Simulators. *IEEE Transactions on CAD*, 2003. in print.

[38] P. Fleischmann and S. Selberherr. Enhanced Advancing Front Delaunay Meshing in TCAD. In *Int. Conf. on Simulation of Semiconductor Processes and Devices (SISPAD 2002)*, pp 92–102, Kobe, Japan, 2002.