

A Strategy to Enforce the Discrete Minimax Principle on Finite Element Meshes

T. Binder, H. Ceric, A. Hössinger, and S. Selberherr

Institute for Microelectronics, TU Vienna
Gusshausstrasse 27-29, Vienna, Austria
Email: Binder@iue.tuwien.ac.at

Abstract – Low quality meshes in three-dimensional finite element diffusion simulations often violate the discrete Minimax principle. An indication of this shortcoming is the occurrence of negative concentrations. We present an a posteriori refinement algorithm to enforce the discrete Minimax principle by locally refining the mesh based on the last time-step. Two examples of diffusion simulations where negative concentrations could effectively be avoided by applying our refinement algorithm are presented. A final example is given to demonstrate the locality of the algorithm.

I. INTRODUCTION

The physical diffusion process follows the entropy principle, where the flow always takes place from regions of high concentration to regions of lower concentration. Over time an equilibrium concentration must appear. This behavior is expressed as the (continuous) maximum principle for parabolic differential operators. A physical interpretation thereof states that the maximum must occur either at the initial time or at the boundary in case mass flows from the outside. The dual minimum principle states that also the minimum occurs at the initial time or at the boundary. A compliance with the maximum and minimum principle (Minimax principle) guarantees that the maximum and the minimum will stay below and above the initial values, respectively. Thus, positive concentrations at all time-steps during the simulation are a necessary condition that the Minimax principle is fulfilled. In a diffusion simulation with homogeneous Neumann boundary conditions (no mass flux through the boundary of the simulation domain) the mass (i.e. dopant dose) is conserved, thus the maximum and minimum must occur at the initial time.

It was shown that traditional Delaunay meshes (necessary for two-dimensional diffusion simulations) are neither sufficient nor necessary for three-dimensional diffusion simulations based on the finite element method [1, 2]. A mesh that is obtuse angle free satisfies the Minimax principle, however, this is a too strong crite-

riion. A weaker mesh criterion was derived by Xu and Zikatanov [3]

$$\sum_{k=1}^n l_k \cdot \cot \theta_k \geq 0.0 \quad (1)$$

Please refer to [4] for a comparison of different discretization mechanisms with respect to the mesh criterion.

In a solution obtained from simulation the quality of the mesh has a very strong influence on the discretization error that is introduced. For the case of three-dimensional finite element diffusion simulations a low quality mesh can violate the discrete Minimax principle. A violation of the Minimax principle always results in an error in the dopant concentration dose. Sometimes this is observed as negative concentrations that emerge during the simulation. A fulfilled angle criterion of an edge of the mesh guarantees a positive contribution to the respective entry in the system matrix. We present an a posteriori refinement algorithm to enforce the discrete Minimax principle by locally refining the mesh based on the last time-step. Two examples of meshes that result in negative concentrations without applying our refinement algorithm are given.

II. MESH REFINEMENT

The most important metrics of a mesh refinement algorithm are locality and preservation of the quality of the elements. Recursive and iterative refinement algorithms for three-dimensional tetrahedral meshes were introduced in [5] and [6], respectively. Both algorithms are based on the intersection of an edge of a tetrahedron and result in the very same (refined) tetrahedralization. They take into account the history of refinement steps and work in the following way:

- In a first step all tetrahedrons for which an estimated error exceeds a certain limit are marked for refinement.
- Second, all marked tetrahedrons are split along their refinement edge. Depending on the used algorithm

this split is either performed recursively or iteratively. Note that also elements that originally were not marked for refinement can be refined.

The selection criterion for selecting the refinement edge depends on the history of a given tetrahedron. The algorithm defines three generations of refined elements, where each generation is refined differently. The generations are numbered in ascending order with the children of generation three belonging again to generation one. The algorithm results in choosing the longest edge in many cases. The above mentioned algorithms keep the refinement local by refraining from refining the longest edge in some cases. The iterative algorithm is more complicated to implement, the recursive algorithm defines a hard criterion on the initial tetrahedralization.

Our refinement algorithm is a simplification of the above mentioned algorithms. It always recursively refines the longest edge of the tetrahedron under consideration. Figure 1, Figure 2, and Figure 3 illustrate the algorithm. Figure 1 depicts the initial input mesh with

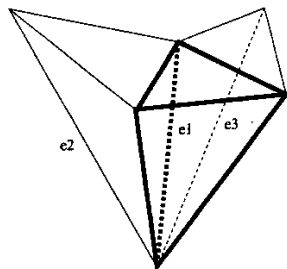


Figure 1: Initial tetrahedralization.

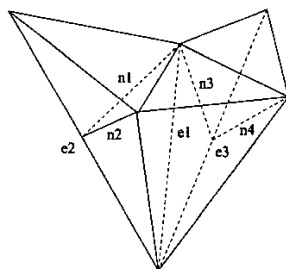


Figure 2: Mesh after first two recursion steps.

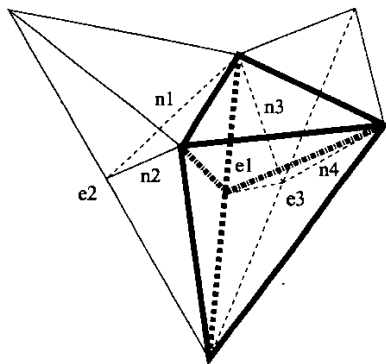


Figure 3: Final refinement result.

one tetrahedron marked for refinement (drawn in thick lines). The edges $e1$, $e2$, and $e3$ depict the longest edge of every tetrahedron respectively. Both edges $e2$ and $e3$ are longer as edge $e1$, thus the neighboring elements are refined recursively. Figure 2 depicts the case after two recursion steps. New tetrahedrons and edges ($n1$ to $n4$) were introduced by splitting along edges $e2$ and $e3$ of the

original mesh. Figure 3 depicts the final result of the refinement step. The tetrahedrons that result by splitting the originally marked element are drawn in thick lines. In this example a total of 6 new tetrahedrons was generated. The recursion is guaranteed to stop, although in the worst case every tetrahedron of the mesh will be refined. However, the application of the algorithm in two and three dimensions showed very good locality which justifies the usage of this simple refinement mechanism.

Since the longest edge of a tetrahedron implies the biggest dihedral angle, one could suspect that a bisection of the longest edge of any tetrahedron allows to make the resulting mesh conforming to the angle criterion by an iterative application. Unfortunately, this is not the case, since geometrical similarities can not be generally avoided for successive refinement steps even if all edges violating the angle criterion are refined. Figure 4 depicts a simple



Figure 4: Two-dimensional example of geometrical similarities occurring during recursive mesh refinement.

two-dimensional case for such a geometrical similarity. Although by bisecting the edge its length l_k is reduced and thus it improves the angle criterion (1), the sum will never become positive if it was originally negative for that edge. Additionally, new edges might be introduced that have an even larger negative sum as the worst edge in the initial mesh. Thus, the bisection algorithms in conjunction with the angle criterion can not generally be applied as an operation to improve mesh quality.

III. A POSTERIORI MESH REFINEMENT

Since an edge refinement algorithm can not be used to enforce the angle criterion (1) we use a different approach to enforce the Minimax principle. Starting at time step t_0 the simulation is performed as follows.

- (1) Compute the actual time step. Keep values of previous time step.
- (2) The concentration of all dopant species is investigated. If no negative concentrations are found save the computed concentrations for later reference, increase the time step, and continue at (1).
- (3) Recursively refine 10% of the edges with the biggest gradient.
- (4) Take over values of previous time step and go to (1).

Although the algorithm is not (theoretically) guaranteed to terminate since it can not repair a violated angle criterion, the practical application shows that it effectively avoids negative concentrations.

IV. EXAMPLES

In the following examples we simulate the well-known model equation for diffusion

$$\frac{\partial C}{\partial t} = \nabla \cdot D \nabla C \quad (2)$$

with homogeneous Neumann boundary conditions (mass conservation, no flux to/from the outside), a backward Euler time discretization and no lumping. For these examples in case of negative concentrations the Minimax principle as pointed out in [4] is violated. As initial profile a spherical Gaussian distribution with a high gradient was chosen.

The first example (Figure 5 and Figure 6) depicts a cube with 650 elements which was refined to ≈ 3100 elements during the simulation. Both figures depict the result after the same time. Figure 5 depicts the unrefined case. Negative concentrations emerged and the symmetry of the Gaussian profile is violated. Figure 6 depicts the result of the simulation with the refinement algorithm turned on. Negative concentrations are not visible. The

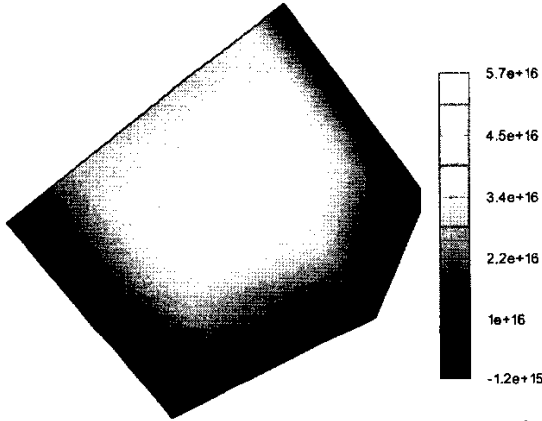


Figure 5: Original mesh - negative concentrations emerge during the simulation. The symmetry of the spherical Gaussian profile is violated.

second example was performed on a different geometry (Figure 7 and Figure 8). The simulation started with an initial number of tetrahedrons of ≈ 1900 which were refined to ≈ 8000 tetrahedrons. The non-refined simulation depicted in Figure 7 was stopped after 5s due to the emergence of negative concentrations. The simulation shown in Figure 8 depicts the case after 13s with the refinement algorithm turned on. The last example demonstrates the locality of the refinement algorithm. Figure 9 depicts a part of a schematic transistor cell. We aimed at refining the gate region. Figure 10 depicts the final mesh after the refinement was performed.

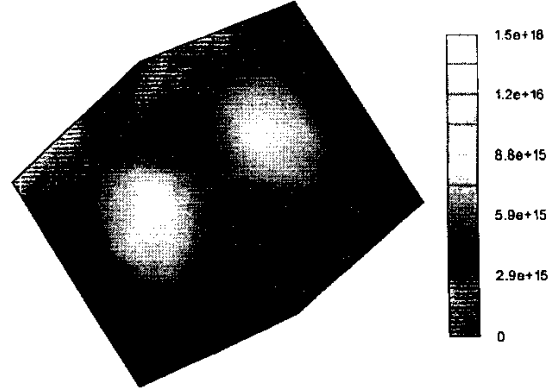


Figure 6: Adaptively refined mesh - negative concentrations have not occurred. The symmetry of the spherical Gaussian profile is reproduced.

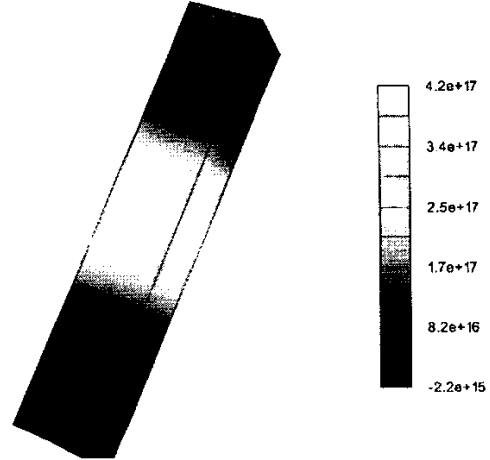


Figure 7: Negative concentrations occur already after 5s.

V. CONCLUSION

We developed a mesh refinement algorithm that is based on the recursive bisection of the longest edge of a tetrahedron. The refinement mechanism was used to perform an adaptive a posteriori mesh refinement that aims at avoiding negative concentrations in three-dimensional finite element diffusion simulations. The algorithm effectively avoided negative concentrations which indicates that the discrete Minimax principle is fulfilled at any time step during the simulation. In traditional "brute force" refinement algorithms the mesh is refined such that the maximum gradient is a priori kept at a certain maximum value. Compared to these refinement strategies that do not take into account the result of a time-step the mesh size was kept comparably small. We successfully applied our refinement algorithm to several examples, two

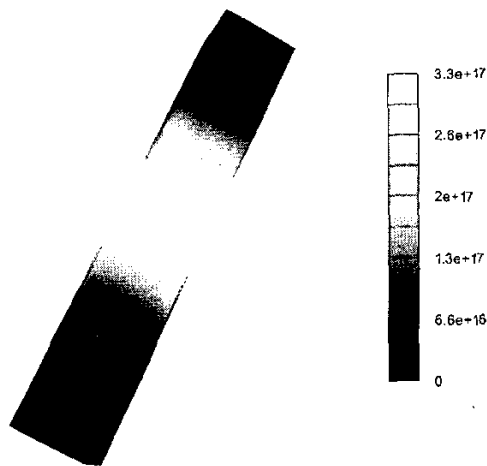


Figure 8: Adaptively refined cuboid – negative concentrations have not occurred.

of which were presented. The developed refinement algorithm is not limited to the presented a posteriori method. Any error estimator can be used to select the tetrahedrons that need to be refined.

Mesh refinement that is based on the bisection of edges can not generally be applied to improve the mesh quality (angle criterion). Future research in this area will thus be directed into mesh relaxation, where the points of the mesh are moved around their initial position [7]. The goal thereby is to find a position of the points where the angle criterion is improved or even fulfilled. The movement of the points is controlled by an optimization algorithm which uses the angle criterion as target. Such a relaxation step can then be used as an a priori quality improvement step before the actual simulation is started.

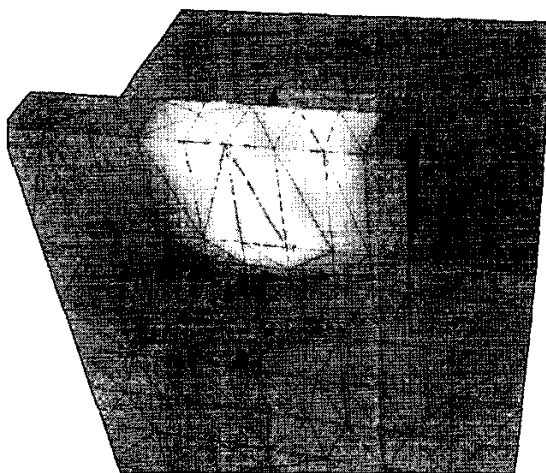


Figure 9: Original coarse mesh.

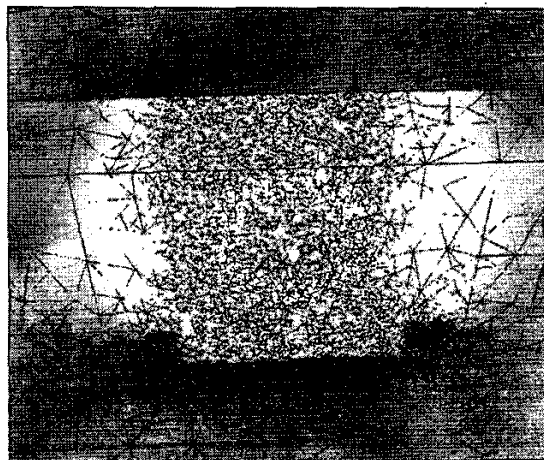


Figure 10: Final mesh. Refinement is restricted to the gate region.

REFERENCES

- [1] P. Fleischmann, B. Haindl, R. Kosik, and S. Selberherr, in *Simulation of Semiconductor Processes and Devices* (Kyoto, Japan, 1999), pp. 71–74. See Ref. [8].
- [2] B. Haindl, R. Kosik, P. Fleischmann, and S. Selberherr, in *Simulation of Semiconductor Processes and Devices* (Kyoto, Japan, 1999), pp. 131–134. See Ref. [8].
- [3] J. Xu and L. Zikatanov, *Math. Computation* **68**, 1429 (1999).
- [4] R. Kosik, P. Fleischmann, B. Haindl, P. Pietra, and S. Selberherr, *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems* **19** no. 11, 1233 (2000).
- [5] I. Kossack, *J. Comput. Appl. Math.* **55**, 275 (1994).
- [6] E. Baensch, *IMPACT of Computing in Science and Engineering* **3**, 181 (1991).
- [7] M. Radi and S. Selberherr, in *Simulation of Semiconductor Processes and Devices*, edited by K. D. Meyer and S. Biesemans (Springer, Leuven, Belgium, 1998), pp. 193–196.
- [8] in *Simulation of Semiconductor Processes and Devices* (Kyoto, Japan, 1999).