Generalized Comprehensive Approach for Robust Three-Dimensional Mesh Generation for TCAD

R. Heinzl and T. Grasser
Christian Doppler Laboratory for TCAD in Microelectronics at the Institute for Microelectronics, TU Vienna
Phone: +43-1-58801/36053, Fax: +43-1-58801/36099, E-mail: heinzl@iue.tuwien.ac.at
Gußhausstraße 27–29, A–1040 Vienna, Austria

Abstract— **To overcome the difficulties in three-dimensional mesh generation for TCAD the advancing front Delaunay mesh generation method [1] was generalized by means of a set of meshing rules. A solid modeling language based on the needs of modern TCAD applications, and a mesh optimization based on a fuzzy classification scheme for the degree of degeneracy of elements have been developed and are ready for use. The applicability and results obtained from our generalized comprehensive approach are presented.**

I. INTRODUCTION

Robust mesh generation for TCAD is one of the most exciting applications within the meshing area and extremely challenging due to the occurrence of all difficult mesh generation tasks like boundary conformity, thin layers [1], complex surface representations [1], small angles [2], the need for surface aligned volume layers, and additionally, the requirements from the ITRS [3]. In particular, process simulation requires boundary integrity, has to handle all kinds of degeneracy in topography simulation, and has to generate surface and interface aligned elements for ion implantation and diffusion simulation. The finite-element method needs well shaped elements but up to now it is not clear, how to measure this criterion. Device simulation, on the other hand, is mostly based on the finite-volume method, which means that the elements must be Delaunay conform. In addition highly non-linear models require directional dependent mesh densities.

The most promising mesh generation technique for three dimensions, the enhancement of the incremental Delaunay refinement algorithm to three-dimensions [4] (used in Tetgen [5]), has not yet found its way to all engineering applications due to the fact that boundary integrity is the major drawback of this method.

Tree-based mesh generation methods cannot incorporate non-planar surfaces and in general produce a larger number of points than necessary. Inherently the generation of surface aligned refinement layers is not possible. Boundary integrity is still not easily guaranteed, although very sophisticated techniques were developed [6]. The quality of the elements is mostly predetermined by the tree discretization method and therefore limited to tetrahedra generated from a cuboid. On that account it has been proven for the two-dimensional [7] and the three-dimensional case [8] that no degenerated elements are generated.

In the area of TCAD, mesh generation and optimization cannot be seen as several or single problems within a domain. In contrary the interaction between all steps like input specification, mesh generation, number and quality of elements, possibly the Delaunay conformity, and adaptive refinement capabilities must play together perfectly in each domain. For this reason we present a *comprehensive approach* to robust mesh generation for all domains of TCAD. Robust means that numerics do not impose a restriction to the mesh generation process.

II. OUR APPROACH

Because of the critical issues summarized above we work with an advancing front technique, which has the ability to produce high-quality elements and nicely graded meshes. Moreover, in contrast to the other methods, boundary integrity is always preserved. On the other hand side, it is very difficult to guarantee the convergence of this technique, especially in three dimensions, because of regions which are not easy to fill up with elements. The slow and complicated geometrical intersection tests to avoid collisions of the fronts are also a major drawback.

Hence our approach [1] is only based on the concept of the advancing front technique combined with a Delaunay point location strategy, similar to the gift-wrapping algorithm [9], in order to avoid the expensive geometrical intersection tests. To overcome the difficulties always present in the enhancement of something as complex as mesh generation we have *generalized* the mechanisms from deLink [1] to a set of abstract meshing rules [10] where the rule set describes the topological connection for the elements. With this generalization we have generated a powerful extension possibility within our mesh generator *deLink2*. This mesh generator is the main part of the comprehensive approach and is tightly coupled to our language approach and the mesh optimization, presented in the next sections. All different kinds of mesh optimization strategies can also be handled by geometrical and topological adaptation rules if the classification of degeneracy is chosen suitably.

Another immediate gain of the advancing front technique is the inherent capability to use the mesh generation technique for mesh refinement and adaptation. This can be used for adaptive mesh generation coupled with some kind of error estimation [11]. In the following, however, the main focus is on the geometrical and topological criteria of robust mesh generation.

III. LANGUAGE APPROACH

A solid modeling language based on any type of extrusion mesh, for instance Laygrid from SAP [12], is not efficient for the generation of currently used non-orthogonal structures. Therefore a *Constructive Solid Geometry* (CSG) [10] based language was developed to meet the present requirements in TCAD like the rising importance of tapered shapes and slanted geometries. With this input language as the first step in our comprehensive approach, we can easily generate non-trivial input structures for all areas of TCAD. To illustrate the practicability of this language all structures in the following sections haven been specified with this language.

IV. MESH OPTIMIZATION

To guarantee the last step in our comprehensive approach a geometrical and topological optimization step with respect to the quality of the elements has to be performed. The definition of a quality measure for elements in three-dimensions is a relatively tedious task because of the consideration of quality with regard to the further use of the mesh. As an example a suitable mesh for topography simulation can never be useful for device simulation. Additionally a lot of different and partially conflicting quality measures for tetrahedra have been established [2, 7, 13, 14]. Most of the classification methods use only one of these quality measures like surface area, volume area, radius ratio, mean ratio, solid angle, dihedral angle, or edge ratio to classify the elements. This is not without difficulty, as on the one hand, most degenerated elements are not identifiable by a single quality measure. The *needle* cannot be identified by the dihedral angle criterion or the *sliver* cannot be identified by the edge ratio. Especially in the area of TCAD some kind of degeneracy can be allowed for special applications, for instance topography or interconnect simulation, to reduce the number of points. In interconnect structures a lot of *wedges* are mostly used for coating elements (Figure 4).

To allow this kind of freedom in the classification of quality for all areas of TCAD we use a non-straight forward classification scheme and subdivide it into two main parts based on the scheme of [2]. First, we identify four classes of quality defined by the number of small dihedral angles (Figure 1).

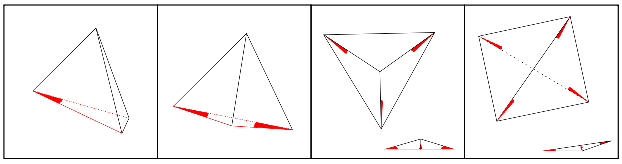


Figure 1: Four different classes of degenerated tetrahedra (wedge, spade, cap, sliver), sorted by the number of acute dihedral angles.

Then, the tetrahedra are classified by the number of degenerated triangles, like daggers and blades. The dagger has one short edge and at least one small angle, where the blade has no short edge and therefore one large and two small angles. Figure 2 shows the needle (or spire) with three daggers (the short edges are marked in the figure), the slat (or splinter) with two opposite short edges and therefore four daggers, and the spindle with no short edges and therefore four blades as triangles.

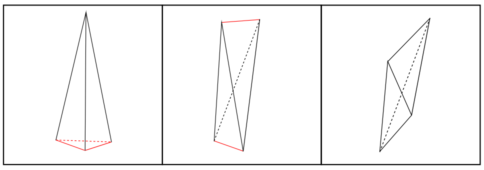
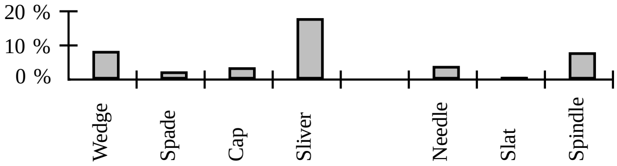


Figure 2: Three different types of degenerated tetrahedra (needle, slat, spindle).

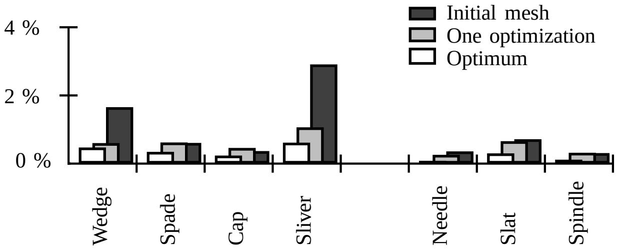
Based on these two parts a fuzzy classification scheme for tetrahedra is derived. Fuzzy means, that each classification part gets a threshold-value from the current application to classify each tetrahedron. The next diagram shows a typical example of a degenerated tetrahedron which belongs to more than one class of degeneracy (the percent value reads for the classification amount).



With the freedom in this classification scheme different types of optimization processes are customized to an application to achieve best results. The following improvement techniques are used in our comprehensive approach:

- Vertex relocation
  The vertices are moved to a geometrical optimum [15].
- Edge/Face swapping
  A simple topological operation which swaps the edge/face shared by a number of elements is applied.
- Edge collapsing
- Edge splitting

The next diagram shows the results from a combination of different improvement techniques. The initial mesh is generated without any geometrical or topological optimization strategy. The subsequent optimization steps are optimized with respect to the current application.



The reduction of all degenerated tetrahedra and particularly the most problematic *sliver* type within each optimization step is obvious (the percent values read for the relative number compared to the total number of tetrahedra).

V. RESULTS

With the CSG approach we are able to construct complicated *interconnect structures* easily by using simple operators on volume elements. Non-orthogonal structures are possible as well as pyramidal objects to model vias of interconnect structures (Figure 3). In general interconnect structure modeling results in a large number of comparatively simple geometrical structures with very different spatial dimensions.

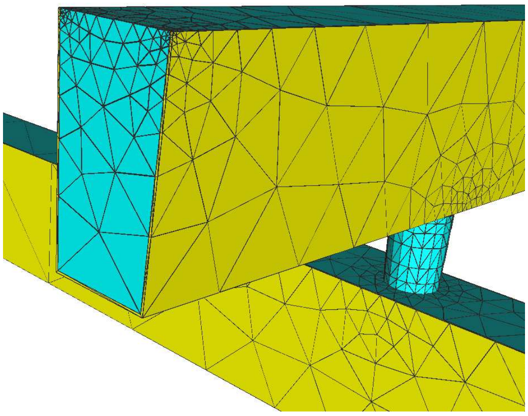


Figure 3: Interconnect structure modeling of lines and vias. Note the mesh in the extremely thin layer.

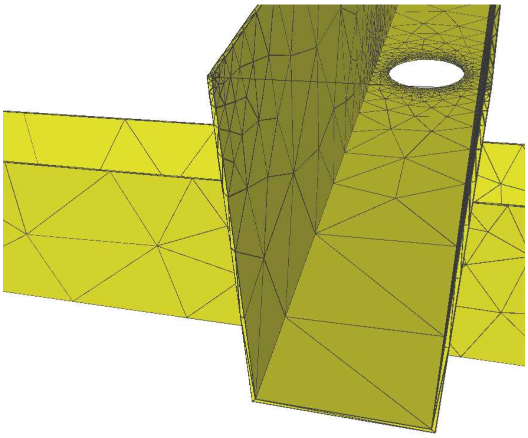
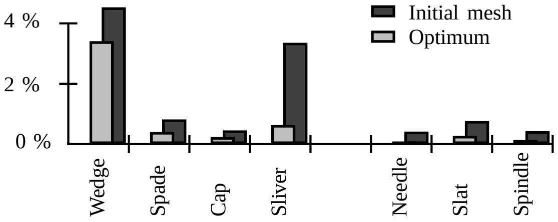


Figure 4: Masking of the metal line to present the hole of the via.

Next the result from our tetrahedra classification scheme and the improvement by the optimization techniques are presented.



After various optimization techniques nearly all *slivers* are eliminated. Only the *wedge* type remains in the mesh because of the very thin coating elements which are intended in this structure.

All steps in *process simulation* are calculated by finite elements or the level set method, and so the Delaunay property is not essential for the discretization of the simulation domain. Therefore a lot of mesh generation techniques cannot be used efficiently because they are optimized to Delaunay mesh generation. Process simulation steps need surface or interface aligned volume layers for ion implantation simulation or diffusion simulation and must provide the ability to handle surface elements of arbitrary complexity containing degenerated or even faulty elements.

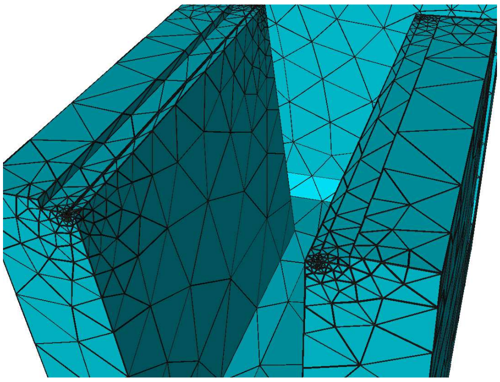
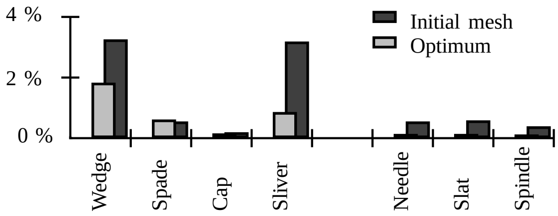


Figure 5: Structure modeling for topography simulation with difficult surface representations.

In the next diagram the results from our optimization strategy to reduce the degree of degeneracy are presented.



In Figure 6 we can see a fully three-dimensional meshed output of an isotropical deposition topography simulation calculated by a level set method [16, 17].

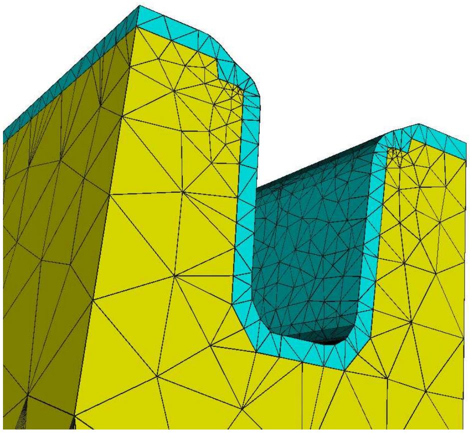


Figure 6: Topography simulation of the generated trench.

In strong contrast to process simulation, *device simulation* is normally based on the finite-volume method which needs mesh elements fulfilling the Delaunay property. Also direction dependent mesh densities in some special areas are needed to resolve highly non-linear quantities.

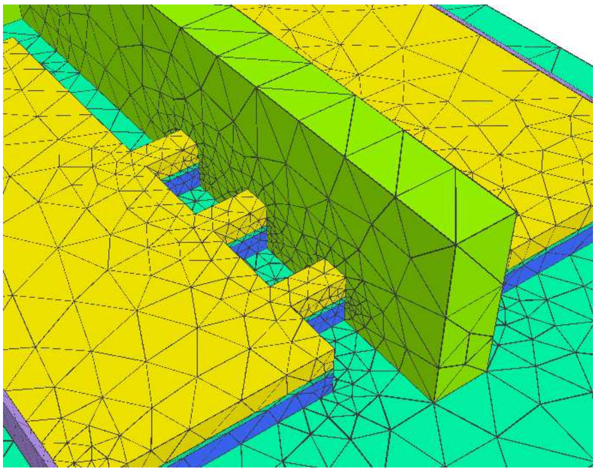


Figure 7: Mesh generation suitable for device simulation with the highly sensitive channel regions of a FinFET.

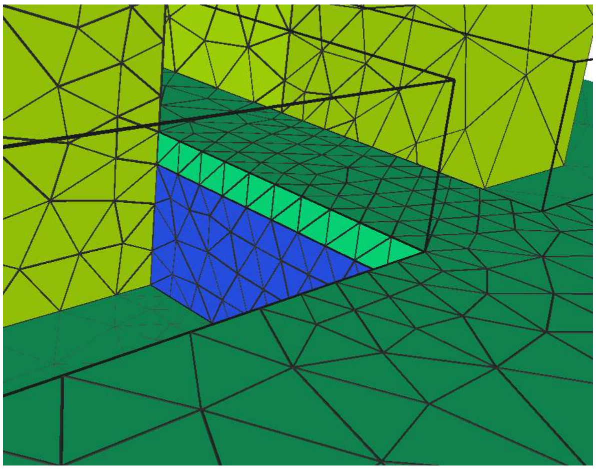
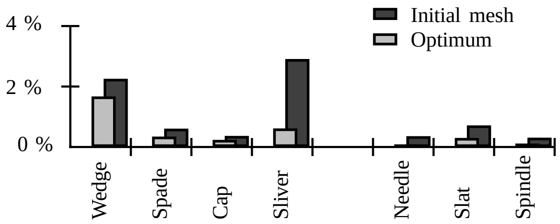


Figure 8: A closer look at the adapted channel region.

In the example depicted in Figure 8 small and flux aligned elements are required in the sensitive channel regions of the structure while the element count outside this domain should be kept to a minimum to reduce computation time (Figure 7). Therefore the classification and optimization strategies are of utmost importance in device simulation.



Therefore the degenerated elements, primarily the *sliver* type, have to be kept to a minimum, which is presented in the diagram.

VI. CONCLUSION

The new language approach gives us the ability to generate non-trivial structures for TCAD easily whereas the mesh generation and mesh optimization steps handle all difficult meshing tasks like surface meshing, volume meshing, and quality optimization. With this comprehensive approach we have shown that notoriously difficult TCAD examples can be generated, meshed, and optimized robustly, resulting in good adapted meshes for process and device simulation.

VII. ACKNOWLEDGMENT

The authors acknowledge the support from M. Spevak, P. Schwaha, E. Al-Ani, and Prof. S. Selberherr.

REFERENCES

[1] P. Fleischmann and S. Selberherr, in *Proc. SISPAD* (Kobe, Japan, 2002), pp. 99–102.
[2] S.-W. Cheng, T. K. Dey, E. A. Ramos, and T. Ray, in *SCG '04: Proc. 20th ASCG* (ACM Press, 2004), pp. 290–299.
[3] International Technology Roadmap for Semiconductors, 2003.
[4] J. R. Shewchuk, Ph.D. thesis, School of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania, 1997.
[5] H. Si, *TetGen 1.3 User's Guide*, Weierstrass Institute for Applied Analysis and Stochastics, Germany, 2005, http://tetgen.berlios.de/.
[6] G. Garretón, Dissertation, ETH, Zürich, 1999.
[7] Bern and Eppstein, *Computing in Euclidean Geometry, World Scientific, Lecture Notes Series on Computing – Vol. 1* (1992).
[8] S. A. Mitchell and S. A. Vavasis, in *Symposium on Computational Geometry* (1992), pp. 212–221.
[9] R. A. Dwyer, in *SCG '89: Proc. 6th CG* (ACM Press, New York, NY, USA, 1989), pp. 326–333.
[10] J. Schöberl, in *NETGEN - An Advancing Front 2D/3D-Mesh Generator Based on Abstract Rules* (1997), No. 1, pp. 41–52.
[11] P. Schwaha, R. Heinzl, M. Spevak, and T. Grasser, in *Proc. SISPAD* (Kobe, Japan, 2005).
[12] R. Sabelka, Dissertation, Technische Universität Wien, 2001.
[13] T. Baker, in *Proc. 7th Int. Conf. Finite Element Methods in Flow Problems* (1989), pp. 1018–1024.
[14] P. P. Pébay and T. J. Baker, Mathematics of Computation **72**, 1817 (2003).
[15] D. A. Field, Communications in Applied Numerical Methods, Wiley **4**, 709 (1988).
[16] A. Sheikholeslami *et al.*, in *Proc. ULIS* (IEEE, Bologna, Italy, 2005), pp. 139–142.
[17] A. Hössinger, J. Cervenka, and S. Selberherr, in *Proc. SISPAD* (Boston, USA, 2003), pp. 259–262.