A Novel Surface Mesh Coarsening Method for Flux-Dependent Topography Simulations of Semiconductor Fabrication Processes

Christoph Lenz[1], Alexander Scharinger[1], Andreas Hössinger[2], and Josef Weinbub[1]

[1] Christian Doppler Laboratory for High Performance TCAD, Institute for Microelectronics, TU Wien
lenz@iue.tuwien.ac.at, scharinger@iue.tuwien.ac.at, weinbub@iue.tuwien.ac.at
[2] Silvaco Europe Ltd., St Ives, UK andreas.hoessinger@silvaco.com

Summary. We propose a new mesh coarsening method for performance-critical topography simulations of semiconductor fabrication processes. The underlying algorithm takes advantage of important domain-specific surface properties. Our method is well suited for Monte Carlo flux calculations and geometric distances are considerable improved compared to references.

1 Introduction

Process technology computer-aided design (TCAD) tools are used to simulate fabrication processes of semiconductor devices whereby one of the aspects is to simulate the evolution of the topography. In such simulations the surface of a semiconductor device is typically represented implicitly by a level set function on a grid. The surface evolution is described by solving the level set equation, which is a partial differential equation [8]. Simulating the surface evolution by etching or deposition using physically based models consists of the following repeated steps: surface flux calculation, surface velocity calculation, and surface advection. The flux calculation is the computationally most expensive step. There exist efficient algorithms originating from the field of computer graphics (e.g. ray tracing), which, however, require an explicit representation of the surface. Therefore, using an explicit surface for flux calculations is an attractive alternative [6].

The explicit representation of a surface is usually realized via a so-called *triangle surface mesh*, henceforth referred to as mesh. Such meshes are made of *elements* (triangles), *vertices* (points), and *edges* (lines between vertices). The run-time of simulations using these meshes are often bound by the number of vertices and hence by the number of elements in the mesh. Therefore, to improve the performance of the flux calculation, reducing the number of mesh elements is required whilst simultaneously keeping the geometric features of the simplified mesh as close to the original geometry as possible.

1.1 Related Work

Several methods for surface mesh simplification can be found in literature [1,3–5]. However, some of these methods try to simplify the geometry as equally as possible [3,5] or use very computationally expensive metrics [1,4]; both approaches are unfit for the problem investigated in this work. Since the mesh simplification has to be conducted at every time step of a simulation it is important that the simplification process offers high performance but also, as already hinted, maintains the features of the geometry as detailed as possible.

2 Surface Mesh Simplification

The simplification method presented in this work is based on the Lindstrom-Turk algorithm. Our new method uses the mean curvature of each vertex to modify the amount of edges the simplification algorithm can remove. To calculate the mean curvature of each vertex we use the method proposed in [7]. The mean curvature is calculated by approximating the *Laplace-Beltrami operator*

$$\frac{1}{2}\sum_{j\in N_1(i)}(\cot\alpha_{ij}-\cot\beta_{ij})(\mathbf{x}_i-\mathbf{x}_j),$$

which is calculated for every vertex and averaged with the area around a given vertex. Furthermore, as the calculation of the mean curvature is a computationally expensive operation, the curvature of each vertex is calculated only once during each simplification step.

Our method divides the mesh into several regions. The *feature region*, the *transition region*, and the *flat region*. These are identified with the help of user-defined parameters: threshold, range, and step length. The threshold specifies the minimum mean curvature value which is considered to contain information about the geometry. The feature region is simplified with the minimal value of the provided range. Subsequently, in the transition region the boundary conditions are gradually increased by the provided step length until they reach the maximum value of the provided range, which is the flat region. This iterative approach allows the mesh to retain a higher resolution near parts of the mesh that contain more information about the geometry and assures that the quality of the triangles doesn't deteriorate.

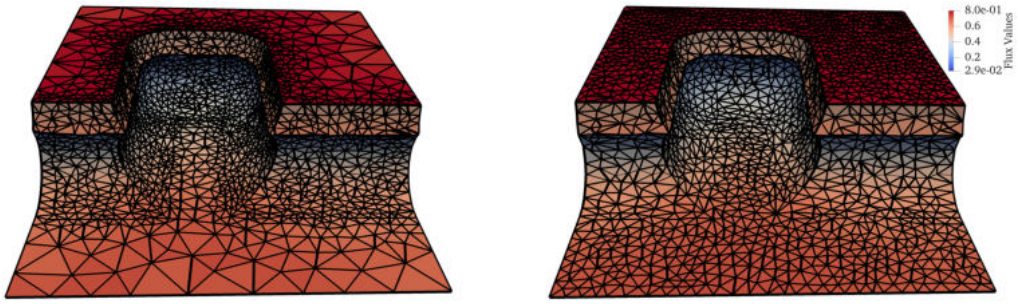

Fig. 1. Simplified mesh created by our algorithm (left) and the Lindstom-Turk algorithm (right). Both meshes where simplified to have the same number of vertices. The colors represent the calculated flux values (in arbitrary units).

3 Results

Figure 1 shows exemplary meshes created by our method and by the reference Linstrom-Turk algorithm as well as the respective flux values (as computed by a Monte Carlo flux algorithm [2]). Our simplification method maintains a lower Hausdorff distance [9] to the original mesh than the Lindstrom-Turk simplification, see Fig. 2. Furthermore, the additional computations that have to be done to calculate the curvatures and regions only increase the required simplification time by 12%. Figure 3 shows a comparison of convergence measures of the Monte Carlo estimates of the flux. Low relative error of the estimates signifies fast convergence. Our method provides a reduced number of mesh elements, a higher resolution in curved areas, and fast convergence.

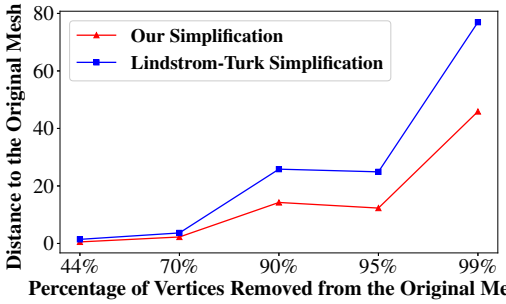

Fig. 2. Accumulated Hausdorff distance to the original geometry for our method and the Lindstrom-Turk algorithm

4 Conclusion

A surface mesh simplification method tailored to the needs of topography simulations for process TCAD has been introduced that selectively simplifies areas of a mesh with high and low geometric information. We investigate the impact on the flux calculation and show that the convergence of the Monte Carlo estimates is similar to a mesh simplified with the reference Lindstrom-Turk algorithm while at the same time maintaining a good resolution of flux values in areas of high curvature. Future work will investigate automatisms for determining the three region-defining parameters.

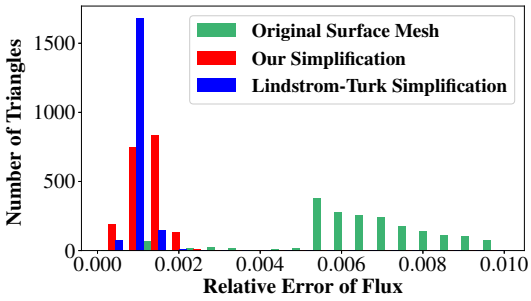

Fig. 3. Histogram of relative errors of the Monte Carlo estimates after surface flux calculation with 2^{24} samples distributed uniformly over the whole source plain of the simulation domain.

Acknowledgement. The financial support by the Austrian Federal Ministry for Digital and Economic Affairs and the National Foundation for Research, Technology and Development is gratefully acknowledged.

References

1. H. Borouchaki and P. J. Frey. Simplification of Surface Mesh Using Hausdorff Envelope. *Computer Methods in Applied Mechanics and Engineering*, 2005.
2. O. Ertl. *Numerical Methods for Topography Simulation*. Doctoral dissertation, TU Wien, 2010.
3. M. Garland and P. S. Heckbert. Surface Simplification Using Quadric Error Metrics. *Proc. SIGGRAPH*, pages 209–216, 1997.
4. S. J. Kim, C. H. Kim, and D. Levin. Surface Simplification Using a Discrete Curvature Norm. *Computers and Graphics*, 26(5):657–663, 2002.
5. P. Lindstrom and G. Turk. Fast and Memory Efficient Polygonal Simplification. *Proc. VIS*, pages 279–286, 1998.
6. P. Manstetten. *Efficient Flux Calculations for Topography Simulation*. Doctoral dissertation, TU Wien, 2018.
7. M. Meyer, M. Desbrun, P. Schröder, and A. H. Barr. Discrete Differential-Geometry Operators for Triangulated 2-Manifolds. In *Visualization and Mathematics III*, pages 35–57, 2003.
8. J.A. Sethian. *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science*. Cambridge University Press, 1999.
9. R. Straub. Exact Computation of the Hausdorff Distance between Triangular Meshes. In *EG Short Papers*, 2007.