A Delaunay Mesh Generation Approach without the Use of Convex Hulls

F. Stimpfl, R. Heinzl, P. Schwaha, and S. Selberherr

Institute for Microelectronics, TU Wien
Gußhausstrasse 27-29/E360, A-1040 Vienna, Austria
E-mail: stimpfl@iue.tuwien.ac.at

1 Introduction

Modeling, generation, and adaptation of unstructured meshes based on e.g. simplicies is of utmost importance for scientific computing, especially in the area of Technology Computer Aided Design (TCAD). Not only does the quality of the results depend on the quality of the mesh, a solution is even impossible, if a mesh cannot be successfully generated. The simulation of microelectronic devices such as transistors is an area of TCAD, which mostly makes use of finite volume schemes for discretization, due to its inherently flux preserving nature, which also implies the fundamental requirement of a Delaunay conform mesh. So far most of the Delaunay mesh generation algorithms utilize convex hulls.

A drawback of these approaches is the fact, that a convex hull is initially constructed and meshed. The final mesh is contained in this convex hull and has to be obtained by recreating the given boundary of the modeled structure. This issue may also result in overhead, due to the construction of convex hull parts, which can be of substantial size and also have to be meshed just to be removed at the end of the mesh generation. This uneccessarily complicates and slows down the whole Delaunay mesh generation process.

Another issue results from a computer's finite numerical accuracy becoming especially apparent when performing geometrical tests. Problems with geometrical prediactes can even yield topological inconsistencies. While topological issues do not suffer from numerical instability by themselves, they can result in failure of the whole meshing algorithm.

Our approach focuses on theoretical results as well as practical techniques to overcome the complex treatment of Delaunay conform mesh generation. The area of TCAD is chosen for application, because the successful generation of a mesh is of fundamental importance here. The introduced approach is based on a surface treatment algorithm in order to satisfy the constrained Delaunay property for the hull first. A subsequent advancing front algorithm is specially adapted to comply with the consistent Delaunay property and to circumvent colliding fronts.

2 The Meshing Approach

Our approach splits the description of the Delaunayzation step into a formal and an algorithmic part. The formal part contains already existing proofs which guarantee the consistency and the Delaunay conformity. The algorithmic part is based on the use of formal theories with the help of an advancing front algorithm. The formal part is presented first.

Lemma 1. *Given a domain D containing the vertices V and the boundary B, then* $\forall\, b \in B$ *there is no vertex* $v \in V$*, which encroaches b, if b is Delaunay.*

In the case an encroaching element exists, an orthogonal, azimuthal projection or a rotation of the encroaching element onto the boundary element is applied [3].

Lemma 2. *If* $b \in B$ *is Delaunay, the* $v \in V$ *closest to b, which does not encroach b due to Lemma 1, is used to create a Delaunay volume element [3].*

Lemma 3. *Let T be the set of volume elements of a tesselation of D. If* $\forall\, t \in T$ *is locally Delaunay* $\Rightarrow$ *T is globally Delauany [3].*

In the algorithmic part of our approach, the first step insures, that all boundary elements conform to the Delaunay property according to Lemma 1. In the subsequent step the advancing front algorithm traverses all existing boundary elements and creates new volume elements according to Lemma 3. It has been shown that this algorithm terminates [3]. If no additional vertices are inserted while advancing the front, the resulting tesselation is Delaunay according to Lemma 3.

3 Result

Taking our approach results in several very important benefits which simplify mesh generation tasks. The fact, that the surface is preserved during the whole volume meshing process, results in inherent parallelizability of the meshing process, which enables to utilize the full potential of upcoming multi-core CPUs as well as the distribution on a computing cluster.

Additionally, the advancing front algorithm does not distinguish between the actual mesh generation and adaptation. Therefore, mesh adaptation can be treated in the same manner as mesh generation, by simply removing bad elements and a subsequent local meshing step. The absense of a convex hull greatly simplifies these steps and also prevents unnecessary meshing steps of areas which are removed at the end, because they are not needed. Only the truly required parts are meshed.

Carefully choosing the programming paradigms and the programming language for the implementation results in additional benefits for multi-dimensional Delaunay mesh generation. Geometrical issues can be treated by using generic programming and utilizing numerical libraries, e.g., intervall arithmetic or exact numerical kernels, e.g. CGAL [1] or Mauch [2].

References

1. A. Fabri. CGAL - The Computational Geometry Algorithm Library, 2001.
2. S. Mauch. *Efficient Algorithms for Solving Static Hamilton-Jacobi Equations*. Dissertation, California Institute of Technology, Purdue, CA, 2003.
3. J. R. Shewchuk. *Delaunay Refinement Mesh Generation*. Dissertation, School of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania, May 1997.