Creation of 3D Surface Models From 2D Layouts For BEM Analysis

Mirko Spasojevic, Per Ljung, Martin Bächtold
Coyote Systems Inc.

ABSTRACT

In this paper we present a new approach to creation of 3D surface models of MEMS devices from 2D layout masks suitable for analysis using the Boundary Element Method (BEM) [1].

The algorithm is implemented as a part of the commercially available BEM based multi-physics solver AutoMEMS [2] and is responsible for automatically generating surface 3D models from a layout (GDSII or CIF) and a user supplied process description. The surface model is then automatically meshed for numerical analysis with 3D BEM solver.

Keywords: 3D surface modeling, BEM, boundary element method

1 INTRODUCTION

The most accurate numerical analysis of complex MEMS devices can be obtained by solving partial differential equations describing 3D physical fields. There are several widely accepted numerical techniques for solving electrical, mechanical and thermal fields, such as Finite Difference Method (FDM), Finite Element Method (FEM) and Boundary Element Method (BEM).

For each of these numerical techniques it is essential to produce a geometry model that accurately approximates the real device made in silicon. The geometry model can be produced by numerically simulating the physics that describes the fabrication process [3][4][5]. Such approaches of process simulation typically require significant computational effort and yield geometries with very detailed approximation of the real manufactured device. Because of the very high computational cost of process simulation and since the generated level of detail causes additional effort during the numerical analysis, this technique is typically limited to small portions of the device.

The alternative is to describe the effects of the fabrication steps using a set of geometric operations, thereby emulating the manufacturing process [6]. Such techniques can produce geometries very quickly, with resulting geometries in very good agreement with the actual manufactured device. Unlike FDM and FEM which require volumetric tessellation, BEM needs only surface representation of the device, which greatly reduces the complexity of the geometry generation and the construction of a computational mesh.

2 GEOMETRY MODELING TOOLS

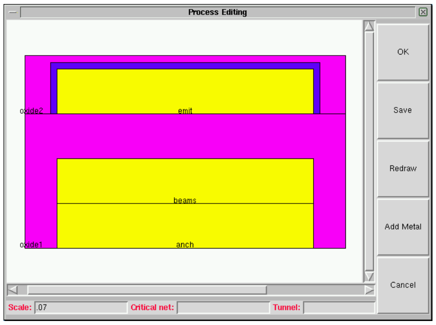
There are a number of commercial solid modeling engines available [7] [8] and we have used some of them to prepare surface meshes for our BEM tools [1]. Geometry generation with these tools is not computationally feasible for realistic MEMS devices. Further, some of the required functionality, such as ability to model conformal depositions is either missing in these tools or is very difficult to use. We found that the resulting triangular surface meshes were frequently unnecessarily fine for use with our tool, causing large computational cost in the field solves without any gain in accuracy of the solution.

In most of the cases we observed that creating a surface mesh was a task as challenging as solving the field problem. To address these shortcomings we have developed our own surface model generator. Our goal was to provide a fast and robust geometry generation engine which is fully automated and needs only minimal user interaction.

3 GEOMETRY GENERATION

The tool reads the mask layout in CIF or GDSII format and combines this information with the parameters of the fabrication process to produce a mesh suitable for analysis with our BEM solver.

.



Scale: .07

Figure 1: Graphical process description.

In order to generate a 3D model from the mask layout, the user must supply at least the heights and thicknesses of each metal layer. For models with multiple dielectric interfaces the user also has to specify the heights and thicknesses of the dielectrics and the values of the relative dielectric constants. Our tool supports both planar and conformal deposition of dielectric materials. The process parameters can be described in a file using simple table based format or by using a graphical editor as shown in Figure 1.

Generating a proper mesh discretization is essential to obtaining an effective solution using the BEM. A very coarse mesh may yield inaccurate results, whereas a too fine mesh requires additional solving time that may not be necessary for a desired solving accuracy. Generally, it is not possible to produce an optimal mesh by relying only on the geometry of the device and without any information about the physics involved in the problem. A mesh would be considered an optimal if it yields a solution within desired accuracy goal with minimal computational cost.

Our BEM solver employs an error indicator based adaptive mesh refinement which automatically converges to a good mesh [9] [10]. Thus we moved the effort of trying to produce an optimal mesh from the geometry generation stage to the BEM solver stage.

The adaptive refinement strategy is performed in several steps. First, the integrity of the input geometry is tested to ensure a legal layout. In the next step, an initial mesh is created based on a set of heuristic rules, to ensure that there are no elements with large aspect ratios or excessive size. After solving for field distribution on the initial mesh, the error indicators are computed on all elements. The elements that contribute most to the overall error are refined in successive steps until the solution with the desired accuracy is reached. This approach greatly simplifies the geometry generation process, because the task of finding a suitable computational mesh is performed by the BEM solver.

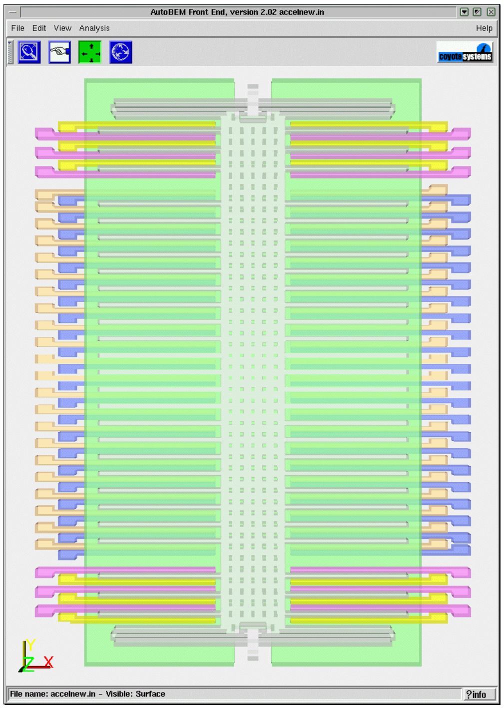
Instead of attempting to produce an optimal mesh, our geometry generator creates a near minimal mesh, i. e. a mesh containing the minimum number of surface panels required to describe the device geometry. The initial meshing produces a set of triangle strips which are then merged into quadrilaterals whenever possible, thereby further reducing the number of elements.

3.1 Layer Operations

In order to allow all processing steps and maintain computational efficiency in model generation, we moved away from traditional 3D solid modeling techniques. The new idea in our algorithm is to exploit the layered structure of the models and perform the major part of computational work in 2D instead of 3D.

The layers are read from a CIF or GDSII input file in the order specified by the process description. In the first step the polygons on all the layers are merged using 2D boolean unions [11] [12] [13] [14]. This ensures that there are no overlapping parts of the model and it also reduces the number of surfaces in the model. This step can be performed very quickly and it does not require any modifications of the input geometry. When using commercial 3D solid modeling tools to generate MEMS geometries, we often found it necessary to reduce the geometrical complexity of our devices, otherwise the models would become too complex and require prohibitively long run times.

An example of a typical MEMS device is the accelerometer in the Figure 2 courtesy of ADI [15]. When using commercial solid modeling tools we had to remove the etch holes; with our algorithm that was not necessary. To merge the 820 polygons in this example on a 233 MHz Intel/Linux computer takes less than a second.



File name: accelnew.in – Visible: Surface

Figure 2: Accelerometer geometry

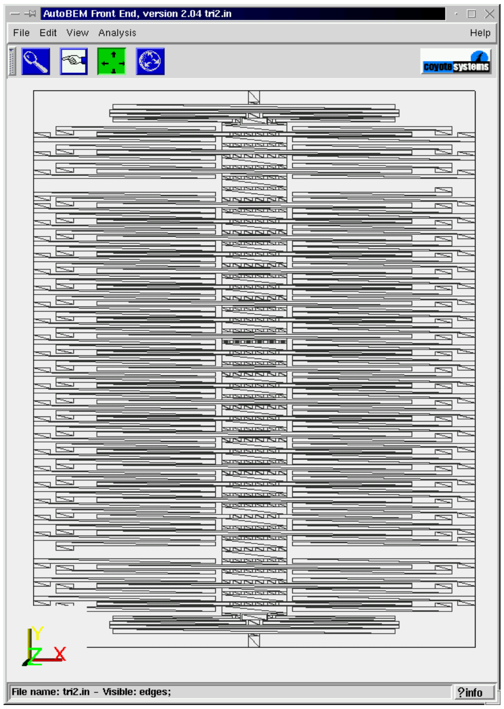


Figure 3: Triangular mesh of accelerometer geometry

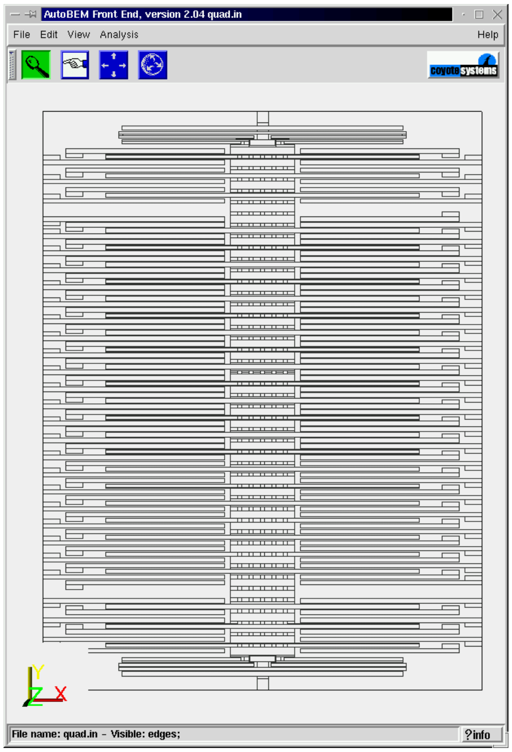


Figure 4: Mesh repaired using quadrilateral elements

3.2 Inter-Layer Operations

Once all of the layers are cleaned up in 2D, we can extrude the masks in the z direction using the user supplied heights and thicknesses of the metal layers. The side walls created in this step are assigned proper materials on both sides. Special care has to be taken if there are multiple dielectric interfaces in the model.

Our BEM solver employs the Constrained Boundary Element Method (CBEM) [10] and can use general surface meshes for simulations. This enables us to generate the side walls of a layer and its top and bottom surfaces independently of each other. Note that the vertices of the boundary elements from two parts of the mesh do not have to coincide using CBEM. To properly identify the interfaces between neighboring layers, we compare the layers whose top and bottom are touching according to the process description.

The areas of intersection between two mask layers do not require any interface elements if both layers are of the same material. In the case of different materials on different layers, these areas can be identified using boolean polygon intersection operation and meshed as interfaces. Otherwise, they are removed from the geometry by employing boolean polygon difference operation [14].

The resulting interfaces are meshed into triangle strips and the mesh is shown in Figure 3. The produced mesh is usually not suitable for immediate use by a BEM solver, because of the very sharp corners and high aspect ratios of the triangles. Typically, it is not easy to repair such a mesh consisting only of triangular elements. However, by merging the neighboring triangles into quadrilateral elements, an improved mesh may be created as seen in Figure 4. This model contains only quadrilateral elements that can easily be refined in the solving process. For the example in Figure 2. the entire process of identifying the interfaces and meshing takes 6 seconds on a 233 MHz Intel/Linux computer.

3.3 Modeling Dielectrics

All of the panels produced during the geometry generation form an interface between exactly two different volumes. In cases where only one metal and one dielectric are used, there are only two volumes which can be identify as the “inside” (metal) and the "outside" (dielectric) volumes. In the case of multiple dielectrics each of the dielectric materials is a separate volume, and different parts of the geometry may have different inside and outside volumes. Some parts of the dielectric volumes are identified during the processing of the metal layers, and the parts that remain to be modeled are the interfaces between different dielectrics.

In order to model interfaces between dielectrics, we find the complement of each of the 2D layer masks within a predetermined bounding polygon on each layer. The resulting polygonal structure is meshed and the volumes on both sides of the interface are identified from the process description. Again, this meshing step is completely independent of the meshes produced in earlier steps, even if they share some edges.

To model conformal dielectrics we create a dielectric volume around metal layers that emulate a conformal deposition. This is accomplished by enlarging each of the 2D masks by the thickness of the conformal dielectric. Similarly to the generation of the metal layers, the horizontal interfaces are determined by the polygons resulting from the boolean polygonal operations and assigning the proper dielectric tags, and the vertical interfaces are obtained by extrusion.

4 VALIDATION OF MODELS

The validity of the produced models is verified through the results of numerical simulations performed by the BEM engine [9][10]. Although the described approach cannot handle arbitrary geometries, for most of the practical designs it is sufficiently fast, robust and simple to use, and the focus of attention can be shifted from creating the model to using the solver to optimize complex MEMS designs.

REFERENCES

[1] Coyote Systems Inc. , Users Guide http://www. coyotesystems. com/pdf/autobem. pdf, 1999

[2] M. Bachtold, M. Spasojevic, C. Lage, P. B. Ljung, "A System for Full-Chip and Critical Net Parasitic Extraction for ULSI, Interconnects using a Fast 3D Field Solver", IEEE Trans. on CAD, to appear

[3] E. W. Scheckler, A. R. Neureuther, "Coupling Model and Algorithms for 3D Topography Simulation: Plasma Etching, Ion Milling and Deposition in SAMPLE-3D", Proc. Workshop on Num. Mod. of Processes and Devices for Integrated Circuits, 1992, pp. 9-14

[4] J. Pelka "Three-Dimensional Simulation of Ion-Enhanced Dry-Etch Process", Microelectronic Engineering, vol 14, 1991, pp. 269-281

[5] E. Straser, S. Selberherr "Algorithms and Models for Cellular Based Topography Simulation", IEEE Trans. on CAD of integrated Circuits and systems, vol. 14(9), 1995, pp. 1104-1114

[6] M. Bachtold, S. Taschini, J. G. Corvink, H. Baltes, “Automated Extraction of Capacitances and Electrostatic forces in MEMS and ULSI Interconnects from Mask Layout", Proc. IEDM, IEEE, 1997, pp. 129-132

[7] ACIS, Spatial Technology http://www.spatial.com

[8] Parasolid, Unigraphics Solutions Inc, http://www. parasolid.com

[9] P. Ljung, M. Bachtold and M. Spasojevic, “Analysis of Realistic Large MEMS Devices”, Proc. MSM, San Diego, March 2000

[10] M. Bachtold, P. B. Ljung, "The Constrained Boundary Element Method, a Technique Allowing General surface Meshes in MEMS Simulations", Tech. Digest Solid state Sensor and Actuator workshop, Hilton Head Island, 1998, pp. 201-204

[11] J. O’Rourke, Computational Geometry in C (2nd Ed.) Cambridge University Press, 1998

[12] F. P. Preparata and M. I. Shamos, “Computational Geometry: An Introduction”, Springer-Verlag, NY, 1985

[13] B. R. Vatti, “A Generic Solution to Polygon Clipping”, Communications of the ACM, 35(7), July 1992, pp. 56-63

[14] A. Murta. “A Generic Polygon Clipping Library”, http://www.cs.man.ac.uk/aig/staff/alan/software/gpc.html, 1998

[15] Analog Devices Inc., 1999