

ADVANCED ETCHING ALGORITHMS FOR PROCESS SIMULATION OF 3D MEMS-TUNABLE LASERS

A. El Boukili

Al Akhawayn University in Ifrane, Morocco

[Email:a.elboukili@au.ma](mailto:a.elboukili@au.ma)

ABSTRACT

We are proposing new and advanced etching algorithms for the process simulation of three dimensional (3D) micro electro mechanical systems (MEMS)-tunable vertical cavity semiconductor optical amplifiers (VCISOAs). These algorithms are based on advanced domain decomposition methods, Delaunay meshing algorithms, and surface re-meshing and smoothing techniques. These algorithms are simple, robust, and significantly reduce the overall run time of the process simulation of 3D MEMS-tunable laser devices. The description of the proposed etching algorithms will be presented. Numerical simulation results showing the performances of these algorithms will be given and analyzed

for realistic 3D MEMS devices and MEMS-tunable laser devices.

Keywords: advanced etching algorithms, domain decomposition, process simulation, MEMS-tunable lasers

1. INTRODUCTION

Vertical cavity semiconductor optical amplifiers (SOAs) represent a low-cost alternative to existing amplifier technologies. They could be used in fiber-optic communication systems such as metro and access networks (Garrett et al. 2005; Garrett 2005).

The surface-normal operation of vertical-cavity (SOAs) gives rise to many advantages as high coupling efficiency to optical fibers, polarization insensitive gain, the potential to fabricate high fill-factor two-dimensional arrays, and the ability to test devices on wafer (Garrett et al. 2005; Garrett 2005; Wu et al. 1995; Larsson, Massengale, and Harris 1996; Tayebati et al. 1998; Christensen et al. 1997; Rabinovich et al. 2003).

Understanding MEMS-tunable device fabrications is essential in optimizing design and enabling a more rapid production of mature devices. The 3D MEMS process simulation is a complex and challenging issue. The structures are geometrically complicated and inherently three-dimensional.

The main goal of this paper is to improve, enhance, extend from two dimensions (2D) to three dimensions, and optimize the existing geometrical etching algorithms for process simulation of 3D MEMS and 3D MEMS-tunable vertical cavity

semiconductor optical amplifiers (MT-VCISOAs) as given in Figure 1. Most of commercial process simulators used worldwide are based on the industry standard two dimensional (2D) process simulator SUPREM-IV (Hansen and Deal 1993) or 3D process simulator FLOOPS (Law 1994). SUPREM-IV has been developed by Stanford University in California. In this paper, firstly, we are focusing on the improvements of the 2D etching algorithms used in SUPREM-IV. And, secondly, we did extend these 2D etching algorithms to 3D which was another challenging task.

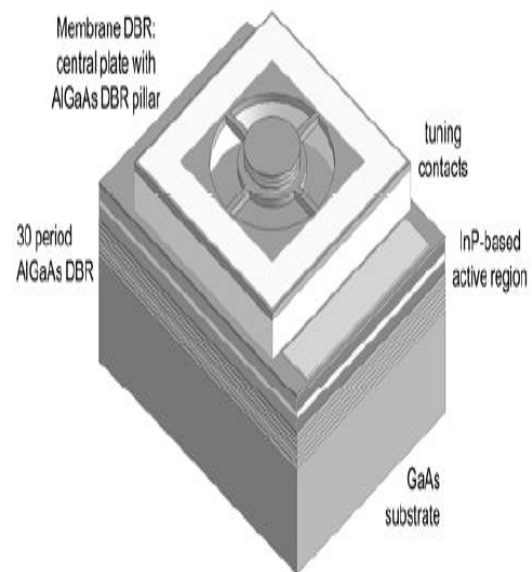


Figure 1: Schematic representation of MT-VCISOAs

We were, exactly, motivated by solving the following three difficulties:

1. In case of a 2D refined mesh using SUPREM-IV algorithms, a strange ripped nylon mesh is generated after etching as seen in Figure 2.
2. Extending the 2D geometrical etching algorithms of SUPREM-IV to 3D for process simulations of complex geometries of MEMS and MT-VCISOAs, while keeping the trade-off

between computational efficiency and the simulation's run time.

3. Creating an accurate and suitable 3D geometry and mesh for fabrication, mechanical, and electrical simulations of MT-VCSOAs and MEMS.

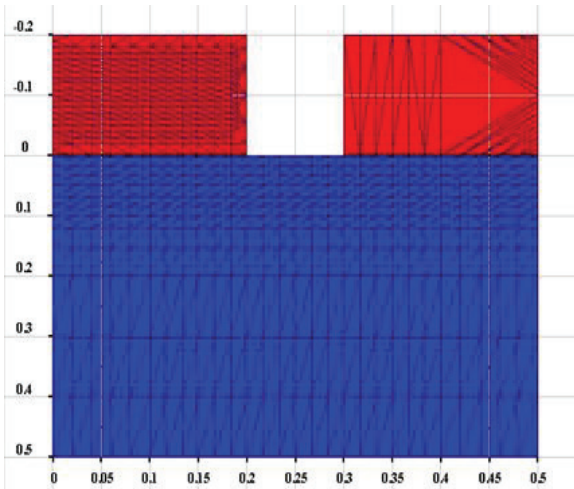


Figure 2: Ripped nylon mesh after SUPREM-IV etching

The traditional geometrical etching algorithms (Sahul 1996; Hansen and Deal 1993; Law 1994) are based on Boolean operations as region subtractions, region intersections, and region unions. They also incorporate additional supporting algorithms as, geometry validation, mesh quality control after etching, de-looping, removing holes, and making regions convex. In case of moving boundaries, as in physical etching, moving boundaries algorithms (Oldham et al. 1980; Glenn 2000) are used to calculate the etching rate (or the velocity of the boundary). In our case, we are not dealing with moving boundaries.

The main step in these etching algorithms is the Boolean subtraction operation. Our contribution is firstly to solve the difficulty 1. By applying boundary smoothing and re-meshing idea after subtraction operation as can be seen in Figure 3. Secondly, we used domain decomposition method (DDM) together with geometrical etching algorithms for the etching simulation of the complex 3D MEMS and MT-VCSOAs. The use of DDM is an excellent remedy to the difficulty in 2. The application of DDM significantly increases the efficiency and reduces the run time of etching simulation and the overall run time of the other process simulations including deposition, implantation, oxidation, and diffusion.

The DDM consists in decomposing the whole 3D structure into different 3D blocks. We could then apply etching algorithms to each block and then merge all the blocks to get the whole 3D structure. Advanced merging algorithms have also been used after etching.

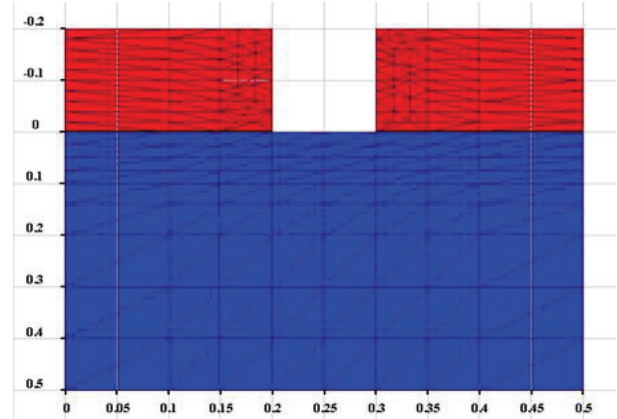


Figure 3: Boundary smoothing helped repair the ripped mesh

They are based on region unions and intersections. The DDM is also very convenient for parallelization.

This paper is organized as follows. The second section describes in details the new etching algorithms. The third section presents the 2D and 3D numerical results that validate qualitatively and quantitatively the introduced geometrical etching algorithms. The performance of these etching algorithms are also discussed and analyzed. A comparison of the obtained 3D numerical results with other results found in literature is also presented.

Section 4 contains conclusions, future work, and recommendations stemming from the work presented in this paper.

2. DESCRIPTION OF THE NEW ETCHING ALGORITHMS

Every 2D or 3D semiconductor structure could be defined in terms of geometry, surface mesh, and volume mesh. Geometry represents the gross outline of the structure and the different materials in the structure. We could look at the geometry as a set of regions. And each region is defined by a material. Surface or boundary mesh refers to a set of planar elements (points, edges, or triangles) whose union form the boundary of the geometry. The volume mesh refers to a set of volumetric elements (as tetrahedral elements) whose union defines the interior and the exterior of the whole device. The main focus of this paper is to smooth and re-mesh properly the surface of each region of the geometry after a geometrical etching. By doing so, we were able to solve the difficulty 1. in 2D and in 3D.

In this section, we will describe in details the algorithms developed and used to perform the geometrical etching simulations in 2D. The extension of these algorithms to 3D has also been achieved and used.

The geometrical etching algorithm first defines the geometry of the region to be etched. This region is then subtracted from each region of the whole structure. The pseudo-code of the algorithm is given in the Algorithm 1 given below. The main idea behind the etching

algorithm is as follows: the intersection i between the etch region e and a region r in the geometry is first computed (line 3 in Algorithm 1). If the etch region is fully enclosed within the geometry region, then, the etch is completed by defining the geometry region to have two boundaries (lines 4-5). The outer boundary is the existing boundary of the geometry region and the inner boundary is the etch region's boundary. Our new implementation of the etching algorithm consists in adding a new procedure **SmoothBoundary()** that will smooth properly these inner and outer boundaries according to some given data from the user. If the etch region completely contains the geometry region, then, the geometry region is completely deleted (lines 6-7). Otherwise, the intersection between the etch region and the geometry region is considered. The geometry region r is to be replaced with the regions that are outside the etch region e . Let I be the set of all the points of the intersection i . Let R be the set of all the points of the geometry region r . Let E be the set of all the points of the etch region e and let $P = R - I$. The Algorithm 1 below, then, works as follows.

First a point p_0 of the region r 's boundary that is not part of the intersection (member of the set P) is found (line 13). Starting from this point p_0 , all the other subsequent points of R that are member of P are collected to form a new boundary B until the intersection with the etch region is found (lines 15-19). The collection continuous along the intersection from the set I until the region r is found (lines 20-25). The points of the region r in the set R are then collected until the first point is encountered (lines 26-29). This closed loop then generates a new boundary. This new boundary is then smoothed and re-meshed by calling the new procedure **SmoothBoundary()** and finally added to the geometry data structure. The procedure continues until all the points of the geometry region r 's boundary that belong to P are part of the new boundary (lines 14, 31-32).

The pseudo-code of the new geometrical etching algorithm (Algorithm 1) is given in the following subsection.

2.1. Pseudo-code of the new etching algorithm

Algorithm 1: new geometrical etching algorithm

Procedure Etch (e, r)

Inputs: e : etch region, r : geometry region

Outputs: new smoothed boundaries and new regions

begin

1. $e :=$ Etch Region;
2. $r :=$ Geometry Region;
3. $i :=$ RegionIntersection(e, r);
4. **if** ($i=e$) **then** (etch region inside region)
5. add boundary of e as internal boundary of r
- SmoothBoundary()**; (new improvement to etching algorithms)
6. **elseif** ($i=r$) **then** (geometry region inside etch region)

7. delete r ;
 - else**
 8. $E :=$ Points of e ;
 9. $R :=$ Points of r ;
 10. $I :=$ Points of i ;
 11. $P := R - I$;
 12. $B := 0$;
 13. first:=p:= MEMBER($0, P$);
 14. **while** ($P \neq 0$) **do** (find the points that are outside)
 15. INSERT(p, B);
 16. np:=FIND(R , point next to p);

 17. **if** ($np \in P$) **then**

 18. DELETE(p, P);
 19. p:=np;
 - else** (find the points that belong to i)
 20. np:=FIND(I , point on edge from p to np);
 21. INSERT(np, B);
 22. **repeat**
 23. np:=FIND(I , point next to np);
 24. INSERT(np, B);
 25. **until** np is on r ;
 - (loop back to get all the other points of r)
 26. p:=FIND(R , point next to np);

 27. **while** ($p \neq$ first) **do**

 28. INSERT(p, B);
 29. DELETE(p, P)
 30. add new region formed by points in the set B ;
 - SmoothBoundary()**; (new improvement to etching algorithms)
 31. $B := 0$;
 32. first:=p:=MEMBER($0, P$);
 33. delete region r ;
 - SmoothAllBoundaries()**; (new improvement to etching algorithms).
- This procedure is optional. It could be called to smooth globally all the boundaries.
- end**

We should note that if the boundary smoothing procedures **SmoothBoundary()** and **SmoothAllBoundaries()** are called outside the geometrical etching Algorithm 1, then, the analysis of the complexity in time of the Algorithm 1 shows that the Algorithm 1 is $O(N+K)$. The number N represents the number of points of the region r . And K is the number of the points of the etch region e . Then, the complexity of the Algorithm 1 has the advantage to be linear.

2.2. Pseudo-code of the new smoothing boundary procedures

During etching algorithm, we call the smoothing boundary procedure **SmoothBoundary()** to smooth and re-mesh the new boundary that we just get. This procedure will split up some or all the long edges of the new boundary according to some criteria and to the user defined parameter α . The shape of the geometry is not altered if the long edges are split up. However, what constitutes a long edge? In this algorithm, long edges are judged according to the original perimeter of the boundary under hand. If an edge of length l_s is bigger than the user specified percentage α of the perimeter or bigger than 3 times the smallest edge of length l_{min} , then, this edge will be split up. The algorithm 2, given bellow, first finds edges in the boundary and checks to see if it is bigger than α times perimeter or bigger than 3 times l_{min} . If it is, then, the procedure **SplitEdge()** is called to split up the edge.

Algorithm 2: Boundary Smoothing

Procedure SmoothBoundary ()

Inputs: **B**: region boundary to be smoothed according to user α : user defined parameter used as a smoothing factor

Outputs: new smoothed and re-meshed boundary **B**

begin

1. $p :=$ perimeter of boundary **B**;
2. $l_{min} =$ length of the smallest edge of boundary **B**;
3. **for** each edge **e** of **B** **do**
4. $l_s =$ length of **e**;
5. **if** ($l_s > 3 \text{ times } l_{min}$ or $l_s > \alpha \text{ times } p$) **then**

6. **SplitEdge(e)**;

end

The etching and smoothing algorithms rely on many other geometry utility algorithms such as bulk and boundary mesh generation algorithms, algorithms to remove holes and make all the regions convex, algorithms assuring boundary orientation in counter-clockwise manner and other similar utilities.

3. 3D NUMERICAL RESULTS AND ANALYSIS

Using the proposed new geometrical etching algorithms, we were able to solve the 3 difficulties 1., 2. and 3. described above. On the other hand, we were successful to etch efficiently all the complicated 3D structures under hand. Figure 4 shows the mesh of a realistic MEMS Tunable VCSOAs after improved

etching. This Figure 4 shows also the mesh quality after improved etching. The Figure 5 shows the structure of a realistic Radio Frequency MEMS device before etching.

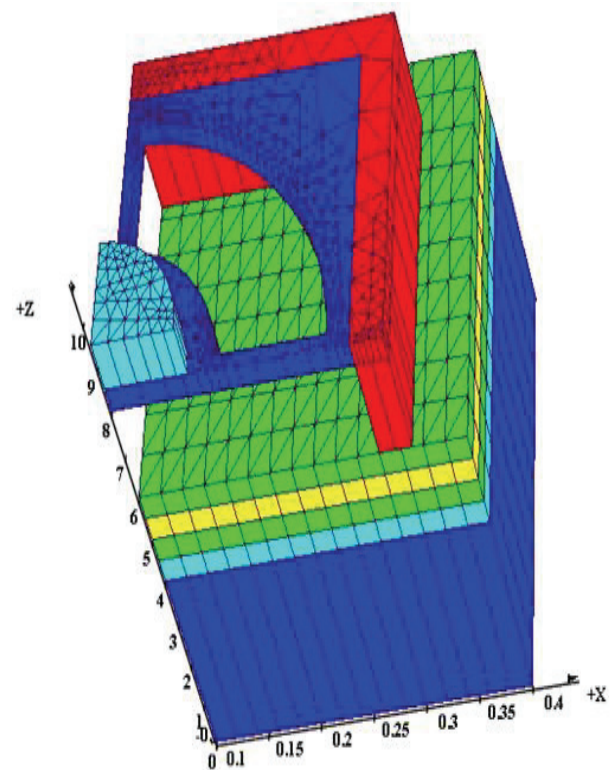


Figure 4: 3D MEMS Tunable VCSOAs after improved etching

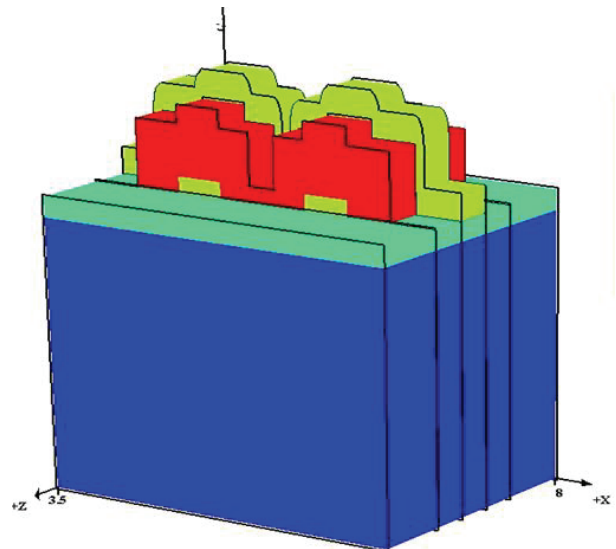


Figure 5: Radio frequency MEMS before 3D etching

The Figure 6 shows the mesh and the geometry of a Radio Frequency MEMS device after improved etching. The quality of the mesh and the geometry shown in all these Figures (Figure 3 to 6) do validate the excellent

performance and the qualitative and the quantitative behavior of the proposed etching algorithms. These performances are comparable with those found in (Glenn 2000) for different 3D structures. The size of our obtained meshes is even smaller and better optimized than those in (Glenn 2000). This is due to the use of the domain decomposition method and the local mesh adaptation.

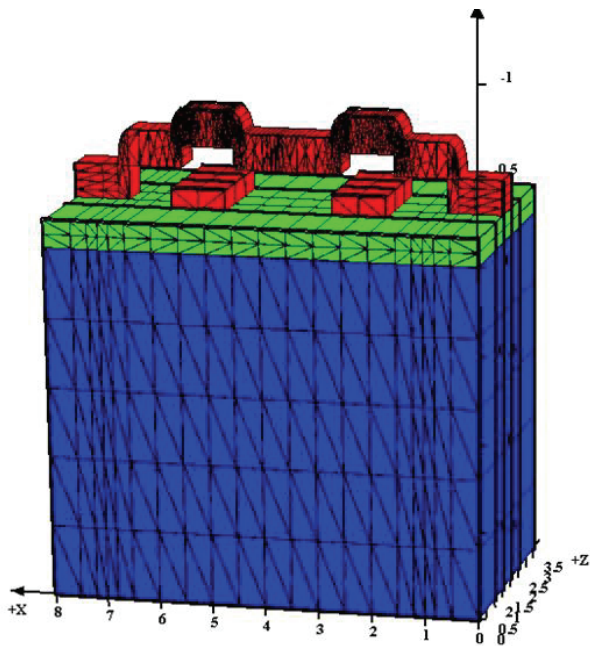


Figure 6: Radio frequency MEMS after 3D improved etching

4. CONCLUSION

In conclusion, by using these new geometrical etching algorithms, it is possible to generate accurately complex geometries and meshes for mechanical and electrical simulations of 3D MEMS and MEMS-tunable laser devices. Parallelization of these algorithms could be investigated in future work.

REFERENCES

- Christensen, G.L., Tran, A.T.T., Zhu, Z.H., Lo, Y.H., Hong, M., Mannaerts, J.P., Bhat, R., 1997. Half Symetric Cavity Tunable Microelectromechanical VCSEL with Single Mode. *IEEE Photon Technology Letter*, Vol.10, 1679-1681.
- Garett, D.C., Qi, C., Chaung-Yeng, C., Shaomin, W., Chad, S.W., Noel, C.M., Bowers, J.E., 2005. MEMS-Tunable Vertical Cavity SOAs. *IEEE Journal of quantum electronics*, Vol. 41, No 3, March.
- Garett, D.C., 2005. *MEMS-Tunable Vertical Cavity SOAs*. P.H.D. Thesis, University of California, Santa Barbara, CA, USA.
- Garett, D.C., 2000. *A Three-Dimensional Front*

tracking Algorithm for Etching and Deposition processes. P.H.D. Thesis, State University of New York, NY, USA.

- Hansen, S.E., Deal, M.D., 1993. *User's Manual of SUPREM-IV.GS: Two dimentional Process Simulation for Silicon and Gallium Arsenide*. Integrated Circuit Laboratory, Stanford Unievrsity, Stanford, California, USA.
- Larson, M.C., Massengale, A.R., Harris, J.S., 1996. Continuously Tunable Micromachined Vertical-Cavity Surface Emitting Laser. *Electron Letter*, Vol. 32,330-332.
- Law, M.E., 1994. *User's Manual of FLOOPS: Florida Objected Oriented Process Simulator Manual*. Dept. of Elect. Eng., University of Florida, Florida, USA.
- Oldham, W.G., Neureuther, A.R., Sung, C., Reynolds, J.L., Nandgaonakar, S.N., 1980. A General Simulator for VLSI Lithograpy and Eteching Process: PartII-Application to Deposition and Etching. *IEEE Trans Electron Devices*, Vol.ED-27, 1455-1459.
- Rabinovich, W.S., Stievater, T.H., Papanicolaou, N.A., Katzer, D.S., Goetz, P.G., 2003. Half Symetric Cavity Tunable Microelectromechanical VCSEL with Single Mode. *IEEE Photon Technology Letter*, Vol.10, 1679-1681.
- Sahul, Z.H., 1996. *Grid and Geometry Servers for Semiconductor Process and Simulation*. P.H.D. Thesis, Stanford University, Stanford, CA, USA.
- Tayebati, P., Wang, P.D., Vakhshoori, D., Lu, C.C., Azimi, M., Sacks, R.N., 1998. Half Symetric Cavity Tunable Microelectromechanical VCSEL with Single Mode. *IEEE Photon Technology Letter*, Vol.10, 1679-1681.
- Wu, M.S., Vail, E.C., Li, G.S., Yuen, W., Chang-Hasnain, C.J., 1995. Tunable Micromachined Vertical Cavity Surface Emitting Laser. *Electron Letter*, Vol. 31,1671-1672.

AUTHORS BIOGRAPHY

Abderrazzak El Boukili received both the PhD degree in Applied Mathematics in 1995, and the MSc degree in Numerical Analysis, Scientific Computing and Nonlinear Analysis in 1991 at Pierre et Marie Curie University in Paris-France. He received the BSc degree in Applied Mathematics and Computer Science at Picardie University in Amiens-France. In 1996 he had an industrial Post-Doctoral position at Thomson-LCR company in Orsay-France where he worked as software engineer on Drift-Diffusion model to simulate hetero-junction bipolar transistors for radar applications. In 1997, he had European Post-Doctoral position at University of Pavia-Italy where he worked as research engineer on software development for simulation and modeling of quantum effects in hetero-junction bipolar transistors for mobile phones and high frequency applications. In 2000, he was Assistant Professor and Research Engineer at the University of Ottawa-Canada.

Through 2001-2002 he was working at Silvaco Software Inc. in Santa Clara, California-USA as Senior Software Developer on mathematical modeling and simulations of vertical cavity surface emitting lasers. Between 2002-2008, he was working at Crosslight Software Inc. in Vancouver-Canada as Senior Software Developer on 3D Process simulation and Modeling. Since Fall 2008, he is working as Assistant Professor of Applied Mathematics at Al Akhawayn University in Ifrane-Morocco. His main research interests are in industrial TCAD software development for simulations and modeling of opto-electronic devices and processes.
<http://www.aui.ma/perosnal/~A.Elboukili>.