Efficient Algorithms for Three-Dimensional Etching and Deposition Simulation

W. Pyka, R. Martins, and S. Selberherr

Institute for Microelectronics, TU Vienna
Gusshausstr. 27–29, A-1040 Vienna, Austria

Abstract

The reduction of computing time without loss of accuracy is a very important task for three-dimensional process simulators. We present new approaches for fast and stable simulation of etching and deposition processes by introducing nonspherical structuring element algorithms.

1. Introduction

For the simulation we use a cell-based structuring element algorithm derived from image processing [1]. The propagation of the surface is calculated by moving a structuring element – in the general three-dimensional case it is an ellipsoid – along the surface and by switching the material index of cells hit by this element. The two-dimensional analogy with circles can be seen on the left hand side of Fig. 1. The process step to be simulated determines the shape of the structuring element. For isotropic etching or deposition the structuring element is a sphere for all cells of the surface. The radius of the sphere is determined by the etch/deposition rate and the time step. For reactive ion etching, sputter deposition or similar particle flux determined processes, the visibility of the particle source seen from the actual position has to be determined and the orientation and the size of the ellipsoid are calculated by vector integration of a given distribution function over the visible solid angle. After each time step, the indices of the material array are updated, the new surface is extracted and again size and orientation of the structuring elements are calculated and applied to the new surface until the overall simulation time is reached.

2. Isotropic Deposition

For the isotropic simulation a sphere is used as structuring element. As can be seen on the left of Fig. 1, this introduces a lot of redundant operations. Especially the cubes near the surface are hit several times by spheres originating from different neighboring surface cells but their material index needs to be changed only once. For larger simulation domains this algorithm turns out to be slow, because of the dependence of the calculation time on the number of cells. If n is the number of cells per micron, the number of operations on the three-dimensional material array depends on n^3. The same applies for the scanning operation of the region surrounding the structuring element. The calculation time of this second operation additionally increases with

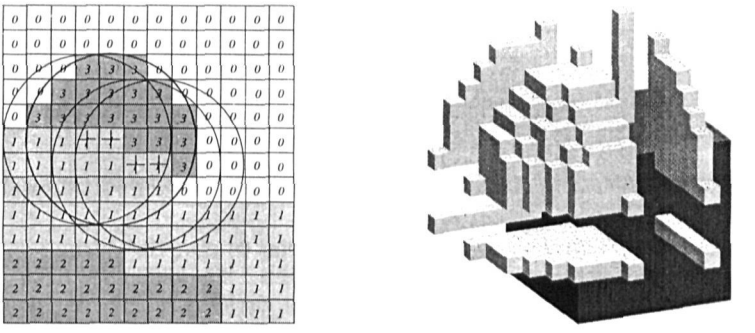

Figure 1: Structuring elements: spheres on the left (here represented by circles for the two-dimensional case), and a combination of spherical segments, circles and lines on the right.

the deposition rate r, because the rate determines the radius of the sphere. Our goal was to increase simulation speed by eliminating redundant operations.

The first optimization step is to limit the spheres to appropriate spherical segments as depicted on the right hand side of Fig. 1. At corners of the topography the sphere is reduced to an eighth, at edges to a one cell thick quarter circle and within planes to a one cell thick line. With this measures the CPU time is up to 100 times lower than for the complete spherical algorithm, depending on the complexity of the structure.

A further approach to CPU time reduction is the restriction to exclusive application of lines as structuring elements. The length of the lines is determined by deposition rate and time. In case of isotropic deposition the direction is perpendicular to the surface. Special care has to be taken at corners or edges where the linear structuring element has to be applied repeatedly with directions interpolated between the surface orientation of the adjacent cells for assuring a smooth and continuous surface. It turned out, that this algorithm leads to results slightly differing from the spherical method because the continuous results of the orientation calculation do not match exactly with the discontinuously represented geometry. The line algorithm is suited better for a triangular based surface representation. In this case the triangles implicitly contain the orientation information and an orientation extraction from the cell array, which is less accurate and time consuming is not necessary. Moreover the number of interpolations between the directions of the surface normals of cells at corners and edges is reduced by this more accurate orientation information.

Table 1 summarizes the computation time for an isotropic aluminum deposition into a 0.5 μm diameter hole for the three different algorithms, namely the original sphere algorithm (a), the spherical segments algorithm (b), and the line algorithm (c). The deposition rate is 0.07 μm/s and the deposition time is 120 s. In total $160 \times 160 \times 100$ cells are used. The simulation was performed with two time steps of 60 s. Fig. 2 shows the initial structure and the topography after each of the two time steps. All calculations are done on a 200 MHz DEC Alpha workstation.

As expected the spherical segment algorithm (b) is several decades faster than the original spherical one (a). The drawback of (b) is, that the second time step (2) takes about twice the time of the first time step (1). The reason is that due to the more complex structure after the first time step even the spherical segments algorithm (b) introduces some redundant information. The figures for the line algorithm (c) are misleading, because they include also the time for the orientation calculation required

	Time step	
Algorithm	(1) 0–60 s	(2) 60–120 s
(*a*) Original sphere algorithm	1361.2	1219.9
(*b*) Spherical segments algorithm	4.7	9.8
(*c*) Line algorithm	23.3	21.6

Table 1: CPU time [s] for the deposition routine of different deposition algorithms.

for this algorithm. The advantage of this algorithm is, that the calculation time is quite independent of the complexity of the structure. The difference to (*b*) is, that the interpolation is always performed with quarter circles parallel to the polar angle of the surface orientation of the considered cell and not with eighths of spheres or circles. Therefore the CPU time for the second time step (2) of (*c*) is comparable to the first time step (1).

3. Sputter Deposition

Another improvement is achieved by an efficient interpolation technique, as shown for magnetron sputter deposition. We approximate the sputter reactor flux by using an exponential function $F(\vartheta) = a\vartheta^3 e^{-b\vartheta}$, which is fitted to the angular distributions resulting from Monte Carlo simulations of sputtering particle transport [2]. In this case the orientation of the deposition vector is not determined by the surface normal but by the flux integral. The interpolation can be performed between the rate vectors of two neighboring cells. In general there is only a small difference in the directions and the azimuthal interpolation needs not to be evaluated over 90° as for the isotropic deposition. Fig. 3 shows a cross-section of calculated deposition rates for a three-dimensional simulation of TiN magnetron sputter deposition into a circular contact hole. The dark area at the topmost corner shows the interpolation of the rates similar to the interpolation for the isotropic simulation.

4. Applications

The improvements in the surface moving algorithms allow fast simulation of three-dimensional structures. As an example Fig. 4 depicts a 5.2 μm×2.8 μm×3.2 μm struc-

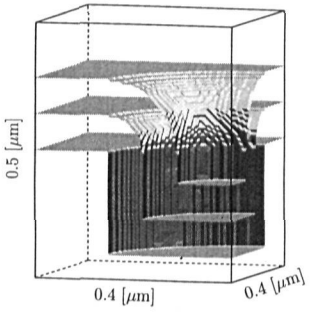

Figure 2: Isotropic deposition of Aluminum into a circular hole – initial geometry, topography after 60 s and final structure after 120 s deposition time.

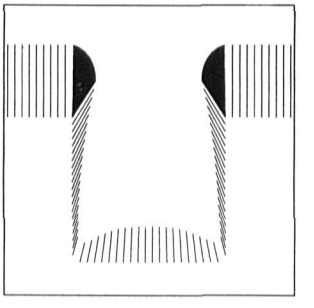

Figure 3: Magnetron sputter deposition: linear structuring elements resulting from flux integration.

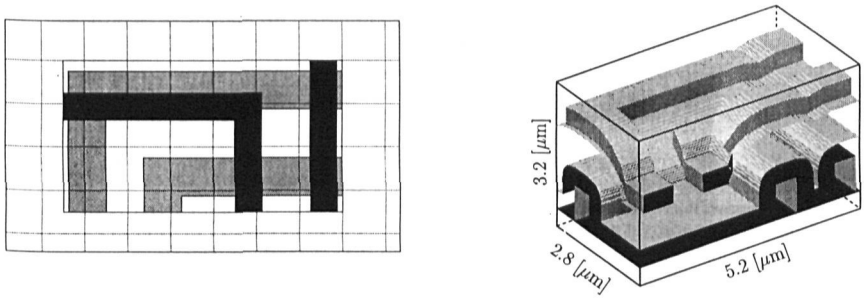

Figure 4: 5.2 μm $\times$ 2.8 μm $\times$ 3.2 μm structure: the layout of the two metal layers on the left, and the topography after masking of the second metal layer on the right.

ture with two metal layers. A metal layer is deposited onto a silicon substrate. The following resist layer is exposed using information from a layout file. After the etch back of the metal, the resist is stripped and a silicon dioxide isolation is deposited. Next a second aluminum layer is deposited and structured with a second resist mask. Fig. 4 shows the topography with the second resist film, before the etch back of the second metal layer.

With the original algorithm the computation time for the complete structure was 218 minutes, the two isotropic deposition steps for the dioxide and the second metal layer took 210 minutes. With the new algorithm the structure was simulated in 10 minutes, 2 minutes thereof were needed for the deposition.

5. Conclusions

With the reduction from three-dimensional ellipsoidal or spherical elements to two-dimensional circular or one-dimensional linear structuring elements, the time used for the deposition routine can be reduced drastically. With the accelerations gained the deposition routine itself can be neglected concerning the computational effort and the calculation of the visibility limits becomes the critical time factor. For a more accurate implementation of the linear structuring element and for a even faster simulation a combination of the cellular structure with additional triangle based surface information might be successful.

6. Acknowledgments

This research project is supported by the "Christian Doppler Forschungsgesellschaft", Vienna, Austria.

References

[1] E. Strasser and S. Selberherr, "Algorithms and Models for Cellular Based Topography Simulation," *IEEE Trans.Computer-Aided Design*, vol. 14, no. 9, pp. 1104–1114, 1995.

[2] A. Myers, J. Doyle, and J. Abelson, "Monte Carlo Simulations of Magnetron Sputtering Particle Transport," *J. Vac. Sci. Technol. A*, vol. 9, no. 3, pp. 614–618, 1991.