

Challenges in 3D Process Simulation for Advanced Technology Understanding

S.M. Cea, A. Eremenko, P. Fleischmann*, M.D. Giles, S. Halama*, F.O. Heinz,
A. N. Ivanov, P. H. Keys, A.D. Lilak

Process Technology Modeling, Intel Corporation
* Core CAD Technologies, Intel Corporation
2501 NW 229th Ave., Hillsboro, OR 97124, USA
stephen.m.cea@intel.com

Abstract

This paper describes advances and remaining challenges in unstructured 3D meshing techniques for both process and device simulations and parallelization of the process simulator FLOOPS. The meshing is performed using a point cloud manager to create points and an unstructured tetrahedral mesher. Distributed parallel techniques are used to parallelize the sparse matrix assembly and solution for 3D process diffusion simulations.

1 Introduction

Continually shrinking device widths, the importance of 3D stress modeling [1] to help engineer stress in advanced logic technologies, and the interest in exploration of truly three dimensional devices such as Tri-Gates[2] have made 3D process and device modeling essential for advanced technology development. Routine 3D process and device simulations have been limited in the past due to a number of factors including poor robustness of 3D meshing and structure creation, and long simulations times. This paper describes additions to our internal version of the process simulator FLOOPS[3,4] to enable robust meshing for both process and device simulations, and enable parallel solution of process diffusion and reaction equations to decrease simulation time. Remaining challenges and opportunities for improvement to these areas are also highlighted.

2 Three dimensional meshing

Three dimensional meshing for process and device simulation is difficult because of the large range of feature sizes that need to be resolved, the requirement for both highly anisotropic and Delaunay meshes, and the need for automatic meshing. A typical problem when combining Delaunay techniques with pronounced mesh anisotropy is undesired mesh topology (the Delaunay property causing a point to attract all surrounding edges, thus destroying the desired anisotropy). In this work, an internally modified version of deLink[5] is used as a 3D Delaunay engine driven by polygonal geometry information and points supplied from FLOOPS. The separation into two modules, a point cloud manager and a robust unstructured engine, has proven to simplify the problem of making anisotropic, partially structured meshes for general

geometries. The point clouds have an associated priority. High priority regions, usually anisotropic structured point clouds, are protected against points from other less important point clouds using the concept of protective bubbles. The protective bubbles are the equatorial spheres of the elements on the boundary of the point cloud. A point encroaches a point cloud if it is inside either its bounding box or its protective bubbles. Low priority points that encroach a point cloud are excluded. Geometry points that encroach a structured point cloud are added to it. In this way the point clouds adapt easily to new geometry points added during the process simulation that could compromise the structured regions. Figure 1 shows meshes from a 3D narrow width transistor after gate formation and near the end of the simulation after a number of spacer depositions and etches. The structured mesh regions with anisotropic spacing are maintained throughout the simulation. Available point cloud types include anisotropic structured, semi structured point clouds that can relax the grid spacing in one direction, refinement point clouds that are used for device meshing to refine on doping or potential, and boundary fitted point clouds for deposited layers. DeLink first creates the surface mesh from a possibly coarse polygonal input and the given point clouds, which are not necessarily confined to the structure's regions and which may overlap polygons. Reliable and efficient surface meshing is a continuing challenge. It then creates the tetrahedral Delaunay mesh using a modified advancing front approach [5]. DeLink contains several mechanisms to increase robustness for meshing these manifold sets of co-spherical points. Robustness within meshing of one material region is gained through an elaborate point classification system to heuristically pick the right point for the next tetrahedron during the advancing front algorithm. If this fails, it corrects mesh overlaps in a post-processing step. To increase robustness across material interfaces and to prevent leaking of the advancing front across region interfaces, efficient intersection testing schemes have been added to deLink. Flat tetrahedra that deLink cannot remove by flipping are chosen to be kept rather than refined. These flat tetrahedra are dealt with by the finite volume discretization code which ignores them during the coupling coefficient calculations (only Delaunay flat elements can remain and these do not contribute to the coupling). This highlights the important link between meshing and discretization and how changes in the discretization can lessen the requirements on the mesher. Development of process and device discretization schemes that can further lessen the mesh requirements is a challenging area with the potential to improve the overall robustness of 3D simulations. The final resulting mesh is all tetrahedral but can contain very structured regions with a large number of zero coupling edges that can be excluded from the discretization. Figure 2a shows a 3D Tri-Gate device mesh created with a mix of structured and general point clouds and refinement on Net Doping. Figure 2b has a zoomed in view of the mesh, and highlights the regions in the channel and source where the mesh refinement is structured.

3 Parallel solution of diffusion equations

As mentioned above, one of the limitations for routine use of 3D process simulation is long simulation times. Figure 3a shows the timing breakdown of an example 3D full flow process simulation. The total time for serial process simulation is ~13 hours and the majority of the time is spent in solving the equations for dopant and defect diffusion and reaction. The diffusion time is dominated by repeatedly solving the

sparse linear system. The “other” category contains all time not spent in solving diffusions and includes meshing, ion implantation, and structure file output. Improving the solve time for 3D process simulation is difficult because many of the techniques used to optimize the solution time in 2D do not work well in 3D. Due to this fact, the assembly and solve portions of FLOOPS have been parallelized using a distributed memory model using MPI [6]. Parallelizing just the matrix assembly and solve sections of the code allows us to take advantage of the parallel speedup and distribute the memory required for the sparse matrix while not having to parallelize the entire code. There are a number of challenges that make parallelizing the PDE solve code for process simulations difficult. These include the large number of equations solved on each node and the variable number of equations solved that depend on the materials and species present for that diffusion. The linear system solution for both serial and parallel simulations in 3D is performed using preconditioned iterative methods from PETSc[7]. Partitioning of the problem is a critical step needed to reduce the amount of communications and to balance the load across the processors. This is performed using METIS[8]. Simulations are run in parallel on a pool of workstations using a fast Ethernet network. Figure 3b shows the total diffusion time and speedup vs. serial for a 3D full flow process simulation run on 2, 4 or 8 machines. Good speedups of 1.9 and 2.9 are seen for 2 and 4 machines. Running on 8 machines is less efficient, with a speedup of only 4.4 due to a loss of preconditioner effectiveness and increased communications overhead. Better parallel solution strategies are needed for parallelization over large numbers of machines. Figure 3a also shows the time breakdown for a 4 CPU simulation. The small speedup in “other” is due to parallelization of the implant. The overall simulation time has been reduced to a little more than 6 hours and the total diffusion time is now only 56% of the total time. This demonstrates that further parallel optimization should also focus on parallelizing other sections of the code like the meshing and fieldserver. Parallelizing the fieldserver would also result in large memory savings because the memory would be distributed across the parallel machines. The main challenges for full parallelization of a process simulator include having to partition the mesh without knowing the equations that will be solved on it at the time of partitioning or efficient repartitioning prior to every step, and development of robust parallel meshing algorithms

References

- [1] S. M. Cea, M. Armstrong, C. Auth, T. Ghani, M. D. Giles, T. Hoffmann, R. Kotlyar, P. Matagne, K. Mistry, R. Nagisetty, B. Obradovic, R. Shaheed, L. Shifren, M. Stettler, S. Tyagi, X. Wang, C. Weber, K. Zawadzki, “Front end stress modeling for advanced logic technologies,” 2004 IEDM Technical Digest, p. 963 (2004).
- [2] J. Kavalieros; B. Doyle; S. Datta; G. Dewey; M. Doczy; B. Jin; D. Lionberger; M. Metz; W. Rachmady; M. Radosavljevic; U. Shah; N. Zelick; R. Chau, “Tri-Gate Transistor Architecture with High-k Gate Dielectrics, Metal Gates and Strain Engineering,” 2006 Symposium on VLSI Technology, pp. 50-51.
- [3] M.E. Law and S.M. Cea, “Continuum based modeling of silicon integrated circuit processing: An object oriented approach,” Comp. Mat. Sci. 12, 289-308, (1998).
- [4] H.W. Kennel, S.M. Cea, A.D. Lilak, P.H. Keys, M.D. Giles, J. Hwang, J.S. Sandford, and S. Corcoran, “Modeling of Ultrahighly Doped Shallow Junctions for Aggressively Scaled CMOS,” IEDM Technical Digest, pp. 875 – 878, (2002)..

- [5] P. Fleischmann and S. Selberherr, "Enhanced advancing front Delaunay meshing in TCAD," *SISPAD 2002*, pp 99–102, Kobe, 2002.
- [6] M. Snir, S. Otto, S. Huss-Lederman, D. Walker, J. Dongarra, *MPI – The Complete Reference*. MIT Press, Cambridge, MA, 1996.
- [7] S. Balay, W. D. Gropp, L. C. McInnes, B. F. Smith, *PETSc Users Manual*, ANL-95/11-Revision 2.3.1, 2006.
- [8] G. Karypis and V. Kumar. *METIS 4.0: Unstructured Graph Partitioning and Sparse Matrix Ordering System*. Technical Report Dept. of Computer Science Univ. of Minnesota, Minneapolis, 1998.

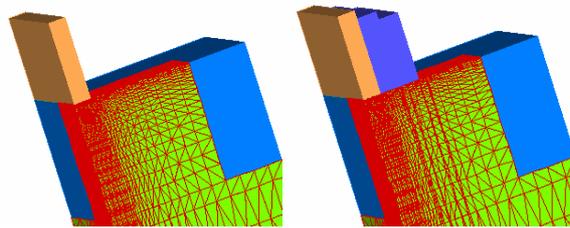


Figure 1: Narrow device process simulation meshes at two points in the flow.

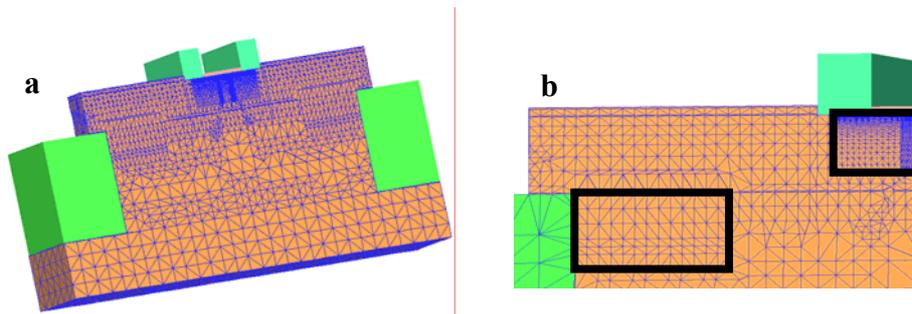


Figure 2: Three dimensional mesh for device simulation of a Tri-Gate device and zoom in on a device mesh highlighting two areas with structured point clouds.

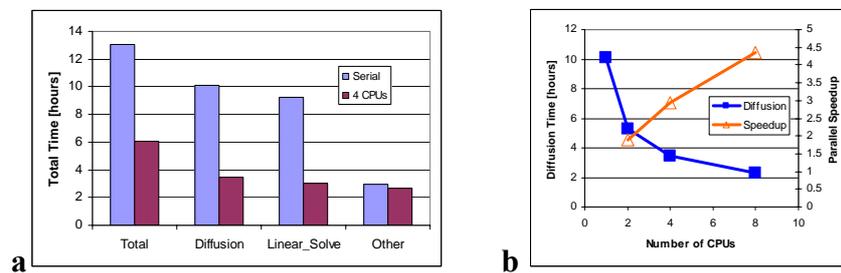


Figure 3: Time breakdown for example 3D full flow process simulation run serially and in parallel on 4 CPUs and total diffusion time and speedup vs. serial for a 3D fullflow simulation on 1, 2, 4 or 8 CPUs.