DEVELOPMENT OF 3D FINITE-ELEMENT CHARGED-PARTICLE CODE WITH ADAPTIVE MESHING

L. Ives, T. Bui, W. Vogler, Calabazas Creek Research, Inc., Saratoga, USA
Mark Shephard, Debyendu Datta, Scientific Computational Research Center, Rensselaer Polytechnic Institute, NY

Abstract

A number of new RF sources are in development that require complex analysis of steady-state, charged particle trajectories. A few codes are currently available; however, they are difficult to use and place severe demands on computational resources. A new program is nearing completion that addresses issues of problem setup and significantly reduce the computational requirements. In particular, the code accepts geometrical input from standard CAD of solid modeling programs, provides an intuitive graphical user interface for attribute assignment, and requires no user setup of the computational mesh. Meshing is fully automatic and adaptive. The mesh adaptation dramatically reduces the size of the computational problems, often by three orders of magnitude, while maintaining or increasing the accuracy over conventional fixed mesh codes. The program includes both electrostatic and magnetostatic field solvers and a fully relativistic particle pusher. The code is intended for engineers and scientist involved in design of the next generation of complex, 3D, charged particle devices.

INTRODUCTION

Calabazas Creek Research, Inc. (CCR), in cooperation with the Scientific Computational Research Center (SCOREC) at Rensselaer Polytechnic Institute, and Simmetrix, Inc., is completing development of a 3D, finite element, adaptive mesh program for designing electron guns, collectors, and other charged particle devices. The program is called *Beam Optics Analysis* (BOA) and includes an intuitive, user-friendly, graphical user interface (GUI) and integral, 3D, magnetostatic solver. The adaptive meshing eliminates requirements for user input of mesh information, dramatically simplifying problem setup, especially for 3D problems. The adaptive meshing also reduces the number of nodes to the minimum necessary for problem solution, significantly reducing the computational requirements and problem execution time.

PROGRAM STRUCTURE

The GUI provides input of geometrical information from any program that outputs ACIS or ParaSolid formatted files, which includes most all commercially available CAD programs. The geometrical information is transferred to a display program for assignment of attributes to objects and surfaces. Typical attributes include voltages, emission surfaces, and material properties. Material properties include conductivity, dielectric properties, work function, secondary emission properties, and magnetic properties. For emitters, the user will be able to input work function and temperature information. Future versions will allow variation of these parameters across the emitting surface. Properties for commonly used materials will be built into the GUI. Prototype GUI screens are shown in Figure 1.

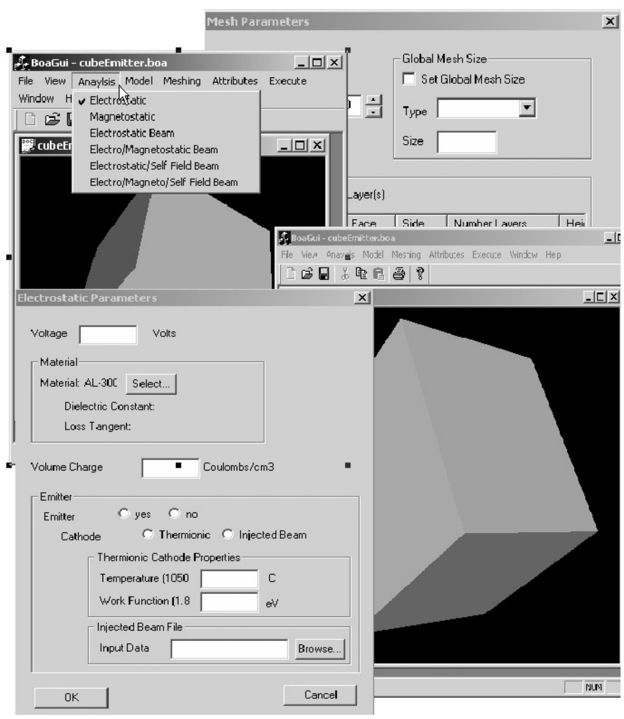

Figure 1. Prototype GUI screens in BOA

Once the attribute information is assigned, the user inputs programmatic information, including the accuracy requirements, number of iterations, and convergence criteria. The program will also have a restart capability. Upon initiation of program execution, the geometrical information is transferred to *MeshSim,* the mesh generator, which creates the initial mesh based on user and geometrical information. A simple example is a charged cylinder in a square box, shown Figures 1 & 2.

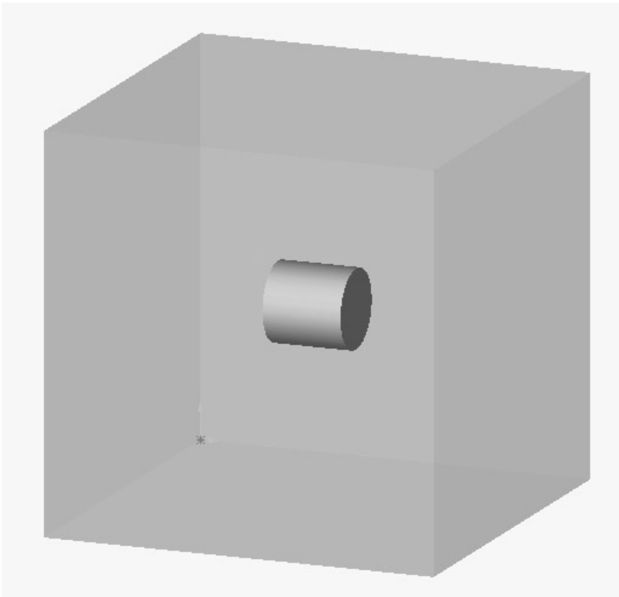

Figure 2. Single cylinder in a box

The initial mesh is used to solve Laplace's Equation throughout the region preparatory to emitting electrons and solving Poisson's Equation. For the example problem, electrons are emitted from the left boundary and accelerated to the right. As the problem evolves, the mesh is adapted based on the electric field values and the space charge of the electrons. Figure 3 displays the initial mesh and the final mesh for the sample problem. In addition to refining the mesh in regions with high electric field gradients and space charge, the element density is reduced where the accuracy criteria is exceeded, such as at the problem boundaries. Consequently, the number of elements is reduced to the minimum necessary to solve the problem. For the sample problem, the number of initial elements was 1641, and the final iteration contained 2690, despite the fact that the element density at the central cylinder increased by more than one order of magnitude. Equivalent accuracy with a fixed mesh code would require more than 55,000 elements, approximately 20 times the number required by BOA.

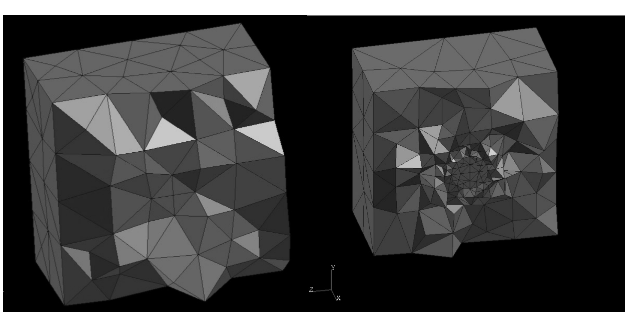

Figure 3. For the example problem, the initial mesh is shown on the left and the final mesh is shown on the right. These are slices through the tetrahedral mesh perpendicular to the direction of electron emission.

The trajectories for the sample problem are plotted in Figure 4. The BOA results are being compared with analytical problems to verify the accuracy and optimize the performance. The GUI is also being completed to facilitate generation of input data and provide post processing. Problem setup is being tested with AutoCAD and SolidWorks.

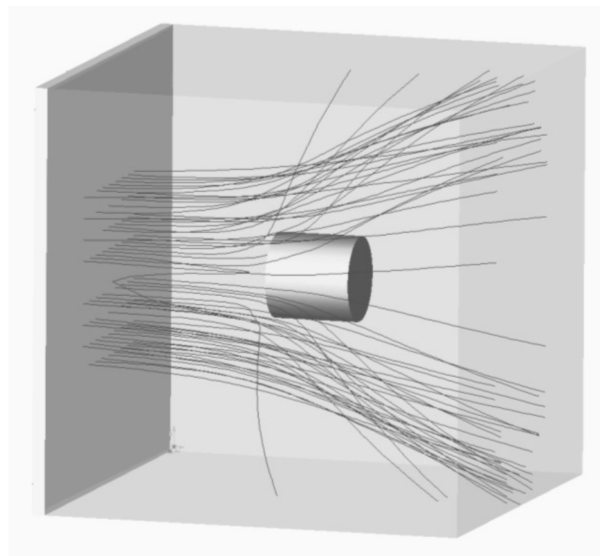

Figure 4. Electron trajectories for sample problem.

The program includes an integral, 3D, magnetostatic solver. This allows design of the magnetic circuit in parallel with the electrostatic configuration and eliminates import of magnetic field information from other 3D programs, though the code will have capability to import field information from Maxwell 3D and MAFIA. Calculation of self magnetic fields is included.

Development of the magnetostatic solver required considerable effort, including development of a new class of tetrahedral element to allow precise matching of boundary conditions across element faces. This was a joint development between CCR and researchers at SCOREC. Information on the components and performance of BOA will be presented. The magnetostatic solver is currently undergoing final testing. A comparison with theoretical results is shown in Figure 5 for a sample case.

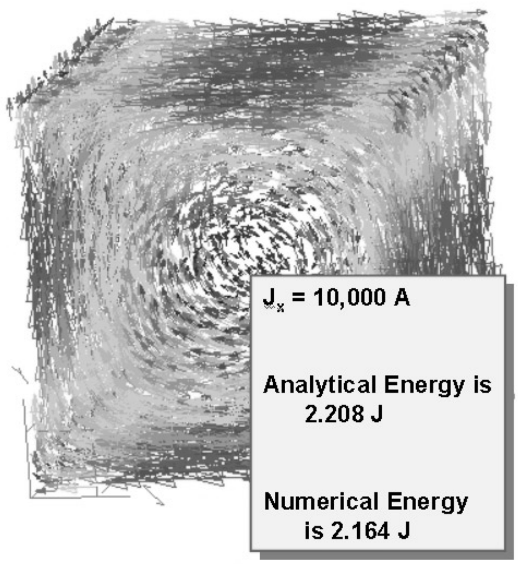

Figure 5. Sample test results for simple magnetostatic problem

FUTURE DEVELOPMENTS

Following completion of the basic code, a number of enhancements are planned. These include:

- parallel processing,
- secondary electron emission,
- performance optimization
- enhanced post processing,
- template generation.

Additional features will be incorporated as suggested by users.

SUMMARY

An advanced 3D trajectory code with adaptive meshing is undergoing final testing. The program accepts geometrical input from standard CAD programs with parametric input from a user-friendly, intuitive, graphical user interface. With adaptive meshing, no user input is required for the mesh. The initial mesh is generated automatically, then coarsened or refined based on the computational results. This minimizes the computational requirement and allows analysis of complex, 3D problems on common personal computers.

ACKNOWLEDGEMENTS

This research is funded by U.S. Department of Energy Grant No. DE-FG03-00ER82966.