APPLICATION OF THE ARGUS CODE TO ACCELERATOR DESIGN CALCULATIONS*

A. Mondelli, C. Chang, A. Drobot, K. Ko, A. Mankofsky, and J. Petillo
Science Applications International Corporation
1710 Goodridge Drive
Mc Lean, Virginia 22102

Abstract

ARGUS is a system of three-dimensional codes which share the same utilities for structure input and grid generation, memory management, data handling, and diagnostics. The field module of ARGUS can solve for electrostatic fields or for the complete electromagnetic fields in either the frequency or time domain. This module can be used as a stand-alone code for impedance and other cold-test calculations. In addition, ARGUS includes a multi-species particle-in-cell (PIC) simulation module. This module can push particles in specified fields, and can function with the field module to generate self-consistent calculations of collective effects in particle beams. The module runs in a transient mode as well as in a steady-state mode (as a gun code).

Description of the ARGUS Code

The ARGUS system of codes has been developed at SAIC over the past six years. The individual codes within the ARGUS model can function together as modules of a large code, or they can function as stand-alone three-dimensional codes to treat specific types of problems.

The ARGUS model has not been built from a two-dimensional code, but is a completely new numerical model whose architecture has been designed to handle three-dimensional problems. ARGUS uses a sophisticated domain-decomposition algorithm[1], coupled with memory-management and data-handling techniques to optimize the use of core memory for each problem and to efficiently move data between core and disk memory during the calculation. A large problem is divided into blocks which are independently processed in core. The data-handling module moves each block from disk to core as it is needed by various code modules. The ARGUS modules are designed for compatibility with this data structure. The disk I/O is table driven, so that the disk I/O sequence for each algorithm can be independently optimized. With these techniques the maximum problem size in ARGUS is limited by the available disk storage and by the overhead of disk I/O, rather than by the size of core memory. The data-handling module attempts to minimize disk I/O by using available data in core whenever possible. Small problems usually remain core-resident.

The structure input in ARGUS is carried out with a combinatorial geometry scheme. The code stores a library of basic three-dimensional objects (*e.g.* a rectangular solid, an elliptical cylinder, an ellipsoid, a paraboloid, *etc.*). These objects are combined by the user with Boolean logical operations *(and, not, or)* to produce structures of arbitrary shape. The structures so specified are represented on the computational grid by a "structure mask" array, which stores the material and electrical properties of each cell on the grid.

Each ARGUS module shares the grid and structure set-up, memory management, data-handling, and diagnostics modules already in the code. New modules are easily implemented in ARGUS as long as they are compatible with the overall code architecture. In addition, ARGUS contains an *operator* module that computes the vector differential operators (**grad**, **div**, **curl**) on the ARGUS grid. Since the grid can be nonuniform in all three directions, the availability of these operators as stand-alone entities greatly facilitates the construction of new algorithms on the ARGUS mesh.

The individual physics modules in ARGUS include a direct FFT electrostatic solver using capacitance matrices for embedded structures, a Chebychev-polynomial electrostatic solver, a leap-frog time-domain electromagnetic field solver, a frequency-domain electromagnetic eigenvalue solver, a steady-state iterative solver for equilibrium particle flows, and a full transient PIC simulation module. These modules can operate individually, as stand-alone codes, or in combination with other modules to produce models of increased complexity and generality.

Applications of ARGUS in accelerator problems are described in a review paper by A. Mankofsky[2]. The present paper describes some calculations undertaken since that paper was completed.

The Four-Finger Cross-Bar RFQ Linac

The electromagnetic field solvers in PIC codes can be operated without particles to provide a powerful tool for cold-test calculations. Such calculations are needed in accelerator design to analyze the higher-order mode content of accelerating cavities or of traveling-wave structures, and to compute the wake fields and impedances associated with various elements or structures in a beam line. In most instances the structures and modes are three-dimensional in nature.

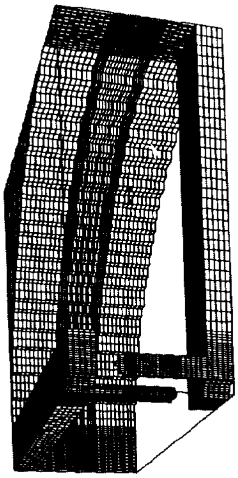

Figure 1. The ARGUS structure mask for the four-finger cross-bar RFQ linac.

Cold-test calculations can be carried out in either the time domain or the frequency domain. The correct choice between these two types of solver is problem-dependent, and modern cold-test modules include both options. In the time domain, the user prescribes a driving field or a driving source, and allows the fields to evolve in time through the solution of Maxwell's equations. In the frequency domain, the user solves an eigenvalue equation for normal electromagnetic modes of the structure. In each case the fields obtained in the calculation are further processed to obtain physics and engineering quantities, such as the stored energy, the Q, the impedance, and the wake field. Although the time-domain and frequency-domain prescriptions are formally equivalent, the time-domain method as implemented in ARGUS offers greater flexibility for treating transient problems and problems having nonlinear materials. The frequency-domain solver, on the other hand, is usually faster-running for problems where the user needs to find the properties associated with the lowest several modes of the structure.

An example of this capability is given by the four-finger cross-bar RFQ linac structure, which is being developed by D. Swenson at SAIC[3]. It provides stronger focusing and allows acceleration to higher energies than is possible with conventional RFQ's. The ARGUS structure mask for this accelerator is shown on

CH2669-0/89/0000-0877$01.00©1989 IEEE

Figure 1. Only one quarter of the structure is displayed. Symmetry boundary conditions are used to represent the parts of the structure that are not explicitly shown. The electric field pattern obtained in the accelerating mode is shown on Figure 2. Two cuts through the structure are shown, the upper one through the curved side wall, and the lower one, orthogonal to the upper, through the RFQ fingers. These calculations have been carried out with ARGUS, which also computes the mode frequency, the shunt impedance and the Q associated with this mode.

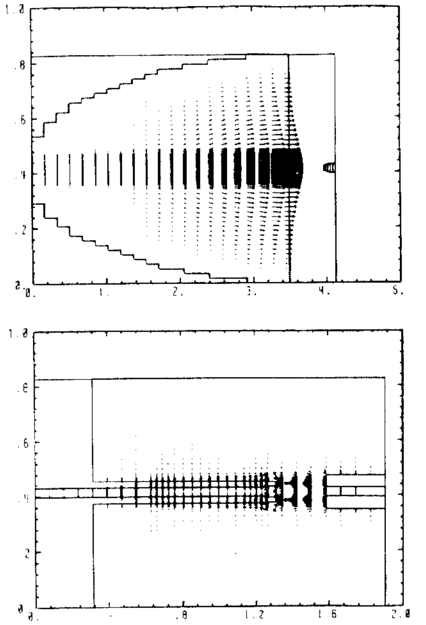

Figure 2. The electric field vectors in the fundamental mode for the structure shown in Fig. 1.

The CCVV ESQ Accelerator and Pierce Gun

The lowest self-consistent level of treating particles is a steady-state model, where the code follows successive generations of test particles across the grid in fixed fields. As each generation of particles traverses the system, the code accumulates the sources needed to update the fields for the next generation of particles. A Pierce gun has been modeled for Lawrence Berkeley Laboratory (LBL) as a preliminary study for the actual H⁻ gun on their CCVV accelerator.[4]

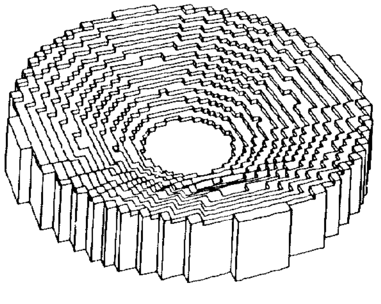

Figure 3. The ARGUS structure mask for the Pierce gun.

Figure 3 shows the curved electrode structure. The gun carries a voltage of 39 kV and draws a current of 30 mA/cm^2. The simulation has used a 60x60x32 grid, and the injected particles have 1 eV of axially-directed kinetic energy in a beam of 1 cm radius. Figure 4 displays the intersection between the equipotential surfaces and the plane through the x-z axes, where z is the directiön of beam propagation. The figure also shows the projections of the particle trajectories onto the x-z plane after ten iterations, i.e. for a well-converged solution.

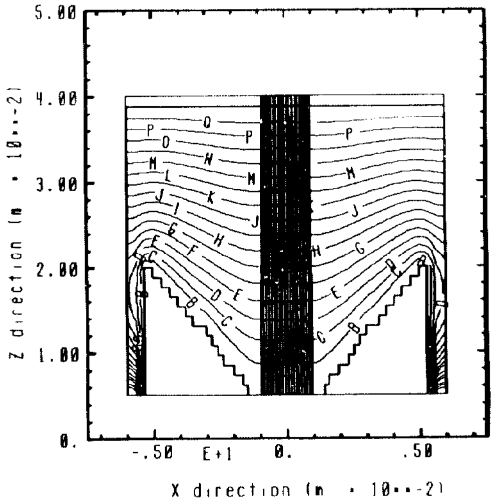

Figure 4. Equipotentials and particle trajectories from the solution for the Pierce gun.

An attempt to carry out simulations of the various components of the CCVV accelerator[5] is currently in progress with LBL. The accelerator consists of 100-kV electrostatic quadrupole (ESQ) accelerator modules, preceded by an extractor and beam matching section. Figure 5 shows the ARGUS structure representation for the matching section and one ESQ module. The matching section is the lower structure in this figure. The side walls of the box containing the accelerator are conducting boundaries, and the top and bottom walls of the box are Neumann symmetry boundaries.

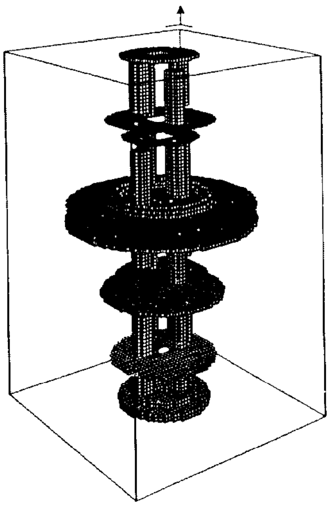

Figure 5. The ARGUS structure mask for the CCVV matching section and first ESQ module.

Voltages are imposed on the various electrodes, but the actual voltage feeds and other geometrical features have been omitted in this calculation. The numerical grid is 63x63x105. The particles are injected into the matching section at an energy of 100 keV, directed axially in a beam of 0.6 cm radius. Figure 6 displays the intersection between the equipotential surfaces and a plane cutting through the ESQ fingers, which appear cross-shaped because they are only marginally resolved on the grid. The particle trajectories after five iterations are shown superimposed on the equipotentials in Figure 7. Figure 8 shows the phase space of axial momentum vs. axial position, which shows the energy variation (without net acceleration) in the matching section, followed by acceleration through the ESQ accelerator module.

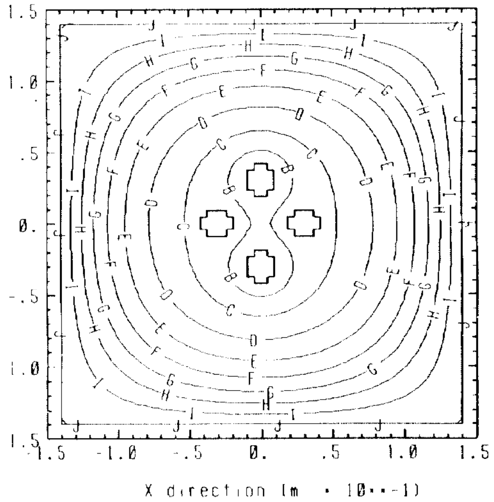

Figure 6. Equipotential lines on a plane cutting through the ESQ fingers.

The Role for ARGUS in Accelerator Physics and Design

Multi-dimensional simulations of charged-particle beam transport, equilibrium, and stability are perhaps the most traditional applications of PIC codes. The potential impact of these techniques on accelerator design is vast, and has begun to be explored only over the last few years. As powerful as PIC simulation codes are, they do not replace the standard accelerator design codes, but rather they augment the capabilities of these codes. They are typically incapable of handling very long, thin systems because the need for radial resolution demands that the cell size be very small, which leads to a very large number of cells to describe the problem. Whole-accelerator PIC simulations, when possible, will be very expensive calculations. Instead, PIC codes can augment other accelerator design tools by allowing detailed calculations of the collective beam behavior in specific regions of interest.

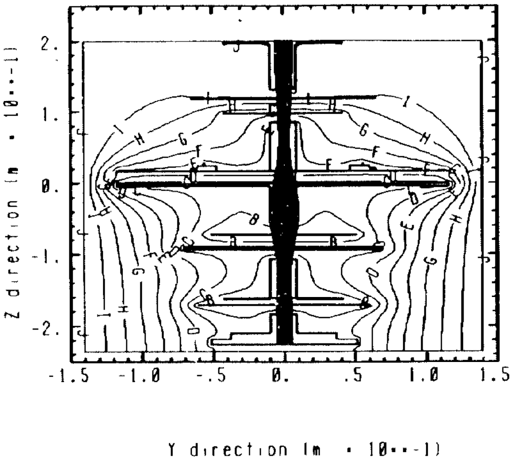

Figure 7. Equipotentials and particle trajectories in a plane containing the CCVV machine axis.

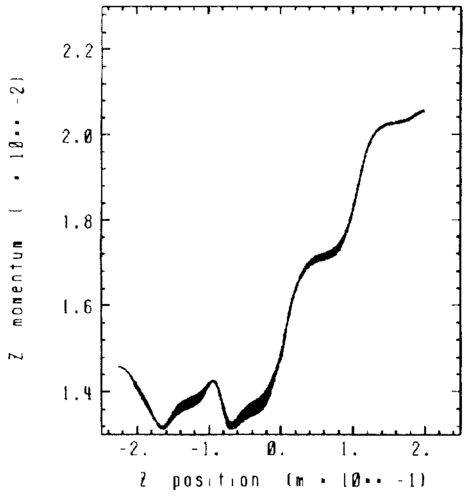

Figure 8. Axial momentum vs. axial position in the solution for the CCVV system.

---

* This work has been supported by the SAIC Independent R&D Program, Lawrence Livermore National Laboratory, Stanford Linear Accelerator Center, Lawrence Berkeley Laboratory, and the Tri-Service/NASA Advanced Numerical Modeling Initiative.

References

1 A. Mankofsky, J. Seftor, C.-L. Chang, K.Ko, A. Mondelli, A. Drobot, J. Moura, W. Aimonetti, S. Brandon, D. Nielsen, and K. Dyer, *Comp. Phys. Comm.* **48**, 155 (1988).

2 A. Mankofsky, *Three-Dimensional Electromagnetic Particle Codes and Applications to Accelerators*, in **Linear Accelerator and Beam Optics Codes** (C.R. Eminhizer, ed., A.I.P. Conf. Proc. No. 177, 1988), pp.137-160.

3 D.A. Swenson, private communication.

4 O.A. Anderson, C.F. Chan, W.S. Cooper, W.B. Kunkel, J.W. Kwan, A.F. Lietzke, C.A. Matuk, P. Purgalis, and L. Soroka, *Proc. 1987 IEEE Part. Accel. Conf.* (IEEE Catalog No. 87CH2387-9), Vol. 1, p. 289.

5 O.A. Anderson, L. Soroka, C.H. Kim, R.P. Wells, C.A. Matuk, P. Purgalis, W.S. Cooper, and W.B. Kunkel, *Constant-Current Variable-Voltage (CCVV) Accelerators* (LBL-25339), presented at the First European Part. Accel. Conf. (Rome, June 7-11, 1988).