IEEE Transactions on Nuclear Science, Vol. NS-28, No. 3, June 1981

BEAM, AN IMPROVED BEAM EXTRACTION AND ACCELERATION MODELING CODE

M.R. Shubaly, R.A. Judd and R.W. Hamm*

Atomic Energy of Canada Limited
Chalk River Nuclear Laboratories
Chalk River, Ontario K0J 1J0

Summary

The desirable features of an ion beam extraction and acceleration modeling code are outlined, and features of the BEAM (Beam Extraction and Acceleration Modeling) code are described. Examples of typical calculations are given and plans for future development are presented.

Introduction

The development of high-perveance, high-quality ion beam extraction and acceleration systems has been paced by the development of calculational techniques available to aid in electrode design. Early analytical calculations[1,2], which treated the multi-gap system as a series of lenses, neglected electrode thickness, field distortion by the electrode apertures and distortion of the plasma surface. Pierce-type[3] calculations were limited to a beam with parallel sides and also neglected aberrations from electrode apertures. Thompson[4] outlined a design procedure for axially asymmetric beams where the defocusing effects of space charge and of the extraction electrode were balanced by convergence of the extracted beam. The electrode shapes were derived assuming the plasma boundary was spherical. Again this treatment did not consider aberrations from apertures or non-uniform plasma surfaces. Furthermore, none of the above treat non-uniform current density, non-zero ion temperature, or ions with a directed velocity.

A number of numerical codes to treat the problem have been developed. Some of these, for example, the "Herrmannsfeldt" code[5] and the "Sheffield" code[6], were originally electron-gun codes and as such do not truly treat extraction from a plasma but assume it can be modeled as ions being generated from an emitting surface. Another similar code is the Kirstein-Hornsby code[7] which was later modified by Bates[8]. Many later codes were based on this one, especially on its treatment of electrode boundaries. The assumption of a fixed emitting surface is especially inappropriate for high-current density, high-perveance sources. Two codes that the authors have used that do treat the problem correctly are the AXCEL[9,10] code developed at the Oak Ridge National Laboratory and the SNOW[11] code developed at the Sandia National Laboratory. It was limitations in these two codes that prompted the development of BEAM. The AXCEL code cannot handle non-uniform current density, or injection problems where the ions have a large directed velocity. Furthermore, space charge neutralization is not included. The SNOW code is more flexible in the types of problems it will handle, however the electrode boundaries are restricted to being on mesh lines. Furthermore, the user input is very awkward.

Desirable Features of a Simulation Code

In this section we list, in order of importance, the desirable features of an ion beam simulation code.

1. The code must provide an accurate solution for the ion trajectories and the electrostatic potentials. The simulation should start in the unperturbed plasma and calculate the ion trajectories, the potential field and the shape of the plasma surface. This is the only way to get a true estimate of the beam emittance and divergence.
2. The program must be easy to use. Any program, no matter how well written, will find little acceptance if the users, generally non-programmers, have to go through complicated mental gymnastics or involved hand-calculations to provide the electrode and beam input.
3. Electrode boundaries must not be restricted to being coincident with mesh lines. Straight line fits or simple arcs between input data points should be utilized. In codes where the electrode boundaries are restricted to mesh lines, the calculation can proceed much more rapidly. However, aberrations caused by the discrete nature of the boundaries can be generated if the mesh is not extremely fine.
4. The mesh density must be variable to permit finer resolution in critical regions (e.g. near the plasma surface), without unnecessarily increasing storage requirements or computational time.
5. The program should not restrict the type of plasma or ion beam input that it will handle. Both extraction and injection calculations should be possible. The code must be able to treat problems with variable current density in the source plasma, variable ion injection energy and angle, and finite ion temperature effects.
6. Space charge neutralization should be included. Behaviour of high-current density, high-perveance beams is strongly affected by space charge neutralization by ions and electrons generated by collisions in the residual gas. Although space charge neutralization is still not completely understood, a recent theoretical treatment[12] gives good agreement with most experimental results.
7. The output from the program must provide easily used information. One of the most important outputs is an overlaid equipotential and trajectory plot showing the boundaries of the electrodes and giving all dimensions and potentials in real unscaled values. Another required feature is phase space plots of the beam at chosen intervals along the axis. Values for the rms emittance, divergence and radius, maximum divergence and radius and total emittance should also be provided.
8. Rectangular (slit) as well as cylindrical geometry should be treated.
9. Axial magnetic fields should be included in the calculation. Many sources presently used (e.g. duoplasmatrons and duoPIGatrons) have weak axial magnetic fields that can perturb the beam trajectories. Sources with strong axial fields (e.g. ECR) cannot be treated by presently available codes.
10. A restart option would be useful. Since many runs feature only small changes in current-density or applied potentials, starting with a previous converged solution could greatly reduce the computational time.
11. Last, but by no means least, the program should have complete documentation including comment cards as well as user's and programmer's manuals. Many of the present codes have been changed and modified to the extent that the flow

* Los Alamos National Laboratory,
Los Alamos, New Mexico, USA 87545

of logic through the program is very difficult to follow. Because the codes were not general enough to treat all types of problems, patches have been put into programs that restrict them to specific types of calculations.

The development of BEAM started by considering the list of desirable features and comparing the two codes that the authors were familiar with (AXCEL and SNOW). Because of the boundary handling characteristics and the expanding mesh used in AXCEL, it was used as the starting point for BEAM. After an extensive structural re-organization, the features not available in AXCEL were added. The space charge update and neutralization calculations from SNOW were used.

Description of BEAM

BEAM solves the Poisson equation

$$\nabla^2\phi = \frac{e}{\varepsilon_o}(N-n)$$

where ϕ is the electrostatic potential, n is the ion density (including space charge neutralization if applicable), and N, the electron density is given by

$$N = N_o \exp[e(\phi_o-\phi)/kTe]$$

where N_o and T_e are the electron density and temperature and ϕ_o is the potential at the centre of the plasma. The Poisson equation is solved using the successive over-relaxation technique of Whitson, Smith and Whealton[9]. The ion density is found by tracing a large number of ion trajectories through the system numberically and calculating the resulting ion density. This density is then corrected for space charge neutralization.

BEAM starts by processing the user input. The first card defines whether the run is a restart or a new problem. The following two cards provide a descriptive title for printouts and plots. Following this are the mesh size, the axial expansion rate and the type of geometry - either rectangular or cylindrical. BEAM uses a rectangular mesh with an optional exponential axial expansion which provides a finer mesh in the plasma region and a coarser mesh downstream. Some care has to be taken in choosing the expansion parameter as, if it is taken too large, features at the bottom end of the column will be smeared out. Next is the electrode boundary input. Each card contains the electrode number, two data points (x,y or z,r) and the type of boundary point - either on the axis of symmetry, on the outer boundary, or an internal boundary point. The boundary is generated from a straight line fit between data points. The following cards give the potentials and an optional axial displacement for each electrode.

Next the parameters of the source plasma (potential with respect to the first system electrode, electron and ion temperatures and ion mass) and the distance from the first electrode to the "unperturbed plasma" are defined. The number of trajectories per cell, the current density, and the apparent focal distance of the trajectories are specified as a function of distance across the "emitting plane". The remaining cards provide space charge neutralization data, control data for the graphics output, and convergence criteria for the calculation. BEAM formats and prints the user's input and certain derived quantities. Next the electrode positions are adjusted for the axial expansion; boundary points are coded, and the mesh nodes are characterized.

After an initial calculation of the potential field without a beam, the initial ion trajectories are calculated. The trajectories start from either the input position of the emitting plane for injection problems, or from a plane a specified number of Debye lengths back for extraction problems. The emitting plane can be divided into regions. In each region, the number of rays per cell, the current density and the apparent focal point can be defined. Varying the number of rays provides better resolution in critical areas (for example at the edge of an extraction aperture). The "focal point" provides an easy way to vary convergence or divergence across the entrance plane to treat injected beams or plasma sources using expansion cups. The rays can then be distributed with a thermal spread in their momentum. The trajectories are then traced numerically through the system to yield the ion space charge. The space charge is corrected for plasma electron and space charge neutralization terms and updated in the Poisson equation using under-relaxation. The space charge neutralization term[12] includes electrons and slow ions generated by collisions in the background gas. Neutralization is calculated in the region downstream of the potential hill for electrons (if it exists). The potential, trajectories and space charge calculations are repeated cyclically until a satisfactory convergence is reached, or until a user-defined maximum number of cycles. Convergence is tested by comparing values for beam radius (height), divergence and emittance at the exit aperture for the present and previous iterations.

For each iteration BEAM prints out a formatted table of the potential matrix; phase space plots of the ion trajectories at selected points along the axis and at the exit of the system; rms and maximum diameters and divergences; rms emittance and the rms ellipse parameters at these positions. In addition, on the first cycle a table of initial ion orbit parameters is generated, and on cycles where the plots are to be done the ion charge matrix, corrected for space charge neutralization, is printed out.

Plots of overlaid trajectories, equipotentials and electrode boundaries with tables of system parameters are generated on user defined iteration cycles. Potential spacing between the most positive and least positive electrodes and between zero and the most negative electrodes are user defined. In addition, for better resolution near critical electrodes, closely-spaced equipotentials on either side of a chosen electrode can be defined.

Test Cases

Two typical cases have been chosen to illustrate some of the features of BEAM. The first case is a duoplasmatron source with an expansion cup in an accelerating column. This test case models the FINS[13] accelerator at CRNL. Only the top half of the column is considered. For this system, the deuterium ions have a directed energy of about 70 eV at the exit of the expansion cup, and the current density varies from 40 mA/cm^2 at the centre of the cup to 10 mA/cm^2 at the edge. Because of the shape of the expansion cup, ions appear to originate from a point 1.11 cm behind the first electrode. Figure 1 shows the equipotentials and trajectories for this problem after 9 iterations at which point the maximum variation in beam radius, emittance and divergence is less than 2%. For the sake of clarity in the reduced figure, the tables of input parameters and equipotentials have been deleted.

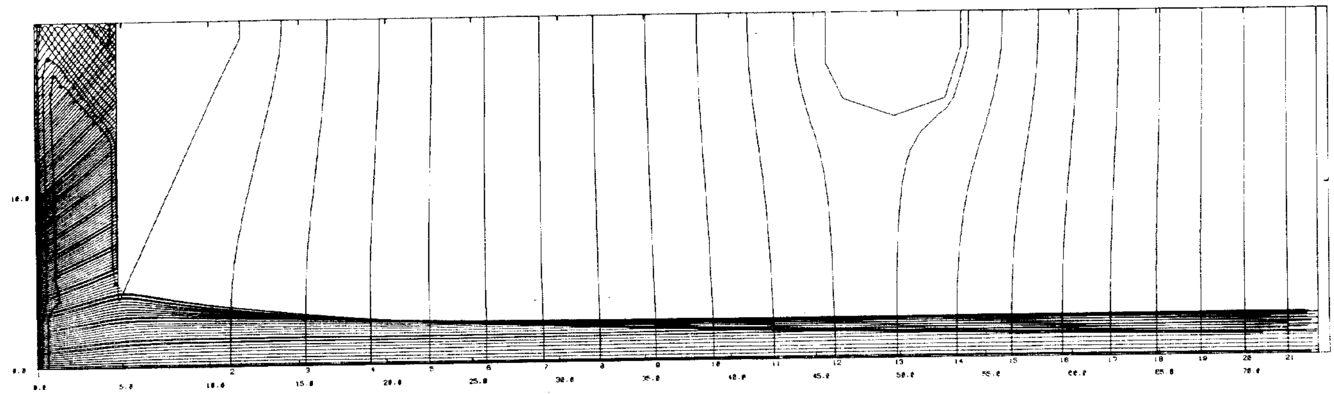

Fig. 1 Duoplasmatron Extraction - Variable Current Density and Injection Angle.

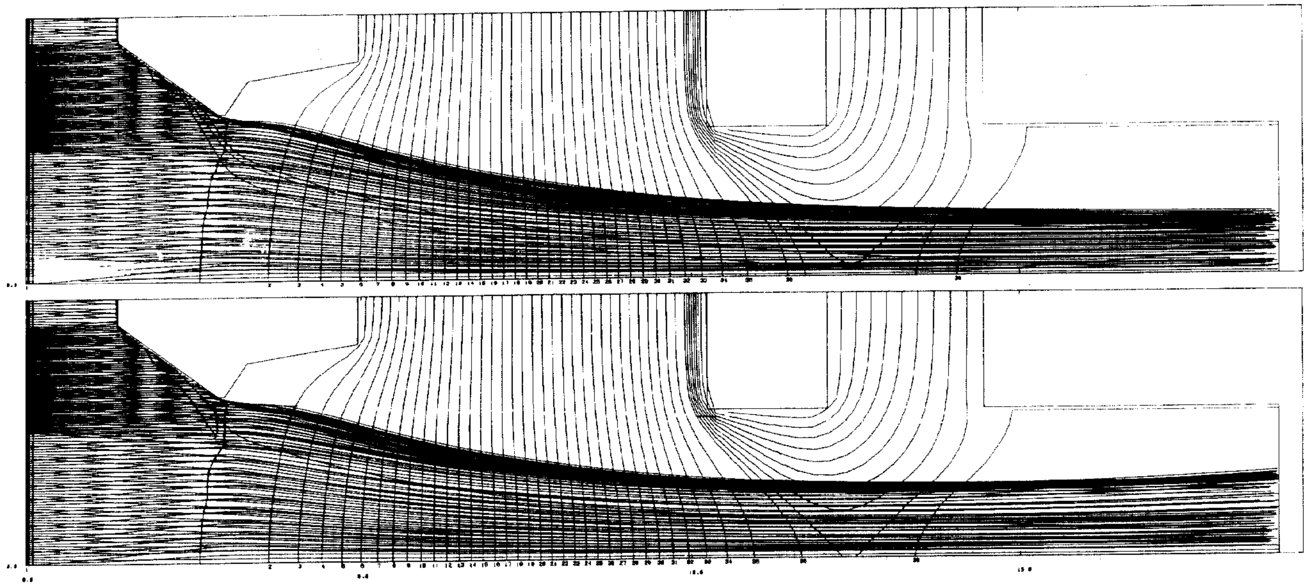

Fig. 2 High Perveance Extraction System - (top) without space charge neutralization, (bottom) with space charge neutralization (7 x 10^{-5} Torr (9.3 x 10^{-3} Pa) hydrogen).

The second example is a high-perveance extraction system used on a high-current dc source being developed at Chalk River[14]. Current density is 400 mA/cm^2. Figure 2 shows the beam shapes without and with neutralization. The electrodes were originally designed (using AXCEL) to give a minimum divergence beam. With neutralization, the beam is larger and more divergent because of space-charge over-compensation in the original design. The calculation with neutralization corresponds closely to the behaviour of the real extraction column.

Future Development

BEAM is proving to be a very useful and easy to use code, but some enhancements are still desirable. BEAM does not presently treat magnetic fields. Incorporation of this feature is difficult with the axial expansion used. The code will be modified so that the mesh density in different regions can be user-defined. This not only eases incorporation of magnetic fields, but also allows the user more control over the problem definition. Some problems, for example a xenon injector for heavy-ion fusion designed at Chalk River[15], use Einzel lenses. The ability to define a finer mesh in the lens region would prove very useful. Use of a smaller number of different size mesh squares would require less storage than does the exponential axial expansion. Another possible change is the remodeling of the sheath suggested by more recent work[16].

Acknowledgement

The authors would like to thank W.L. Michel for his assistance in revising the graphics for BEAM.

References

1. E.R. Harrison, J. App. Phys. 29, 909 (1958).
2. J.R. Coupland et al., CLM-P312, Culham Lab. (1972).
3. R.K. Wakerling and A.C. Helmholtz, TID S217, 227 (1947).
4. E. Thompson, Particle Accelerators 2, 69 (1972).
5. W.B. Herrmannsfeldt, SLAC-226, Stanford Linear Accelerator Laboratory (1979).
6. D. Dirmikis, PhD Thesis, Sheffield Univ. (1975).
7. P.T. Kirstein and J.S. Hornsby, IEEE Trans. Electron Devices ED-11, 196 (1964).
8. D.G. Bates, CLM-R53, Culham Lab. (1966).
9. J.C. Whitson et al., J. Comp. Phys. 28, 408 (1978).
10. J.C. Whitson et al., ORNL/TM-6512, Oak Ridge Nat. Lab. (1978).
11. J.E. Boers, SAND-79-1027, Sandia Nat. Lab. (1980).
12. A.J.T. Holmes, Phys. Rev. A 19, 389 (1979).
13. J.D. Hepburn et al., IEEE Trans. Nucl. Sci., NS-22, 1809 (1975).
14. M.R. Shubaly, IEEE Trans. Nucl. Sci., NS-26, 3065 (1979).
15. M.R. Shubaly and R.W. Hamm, IEEE Trans. Nucl. Sci., NS-28, April 1981 (to be published).
16. G.A. Emmert et al., Phys. Fluids 23, 803 (1980).