

A 3D model for ion beam formation and transport simulation

J. Qiang ^{*}, D. Todd, D. Leitner

Lawrence Berkeley National Laboratory Berkeley, CA 94720, USA

Received 2 February 2006; received in revised form 18 May 2006; accepted 20 May 2006

Available online 7 July 2006

Abstract

In this paper, we present a three-dimensional model for self consistently modeling ion beam formation from plasma ion sources and transporting in low energy beam transport systems. A multi-section overlapped computational domain has been used to break the original transport system into a number of weakly coupled subsystems. Within each subsystem, macro-particle tracking is used to obtain the charge density distribution in this subdomain. The three-dimensional Poisson equation is solved within the subdomain after each particle tracking to obtain the self-consistent space-charge forces and the particle tracking is repeated until the solution converges. Two new Poisson solvers based on a combination of the spectral method and the finite difference multigrid method have been developed to solve the Poisson equation in cylindrical coordinates for the straight beam transport section and in Frenet–Serret coordinates for the bending magnet section. This model can have important application in design and optimization of the low energy beam line optics of the proposed Rare Isotope Accelerator (RIA) front end.
© 2006 Elsevier B.V. All rights reserved.

PACS: 29.27.Bd; 29.27.Ac; 29.27.Eg

Keywords: Beam formation and transport; Poisson solver

1. Introduction

The RIA linac driver requires a great variety of high charge state, high intensity ion beams from the Electron Cyclotron Resonance (ECR) ion sources. This presents a strong challenge not only for the design of the ECR ion sources but also for the design of low energy beam transport (LEBT) systems. Computational tools help to explore a wide range of parameter space, to identify the particle loss conditions, and to optimize the system design and operation. A number of simulation tools have been developed in the past years to study the ion beam formation from ECR ion sources [1–4]. However, these tools used the successive over-relaxation (SOR) method, to calculate the space-charge forces (by solving the Poisson equation) during the ion beam formation. The convergence rate of the SOR method decreases dramatically as the mesh size gets finer and the number of grid points gets larger. Meanwhile, these tools do not effectively handle the space-charge effects inside a bending magnet. So far, they have not been used to self-consistently

simulate a three-dimensional ion beam transport inside a LEBT system. Previous designs of LEBT systems depended on a two-dimensional envelope model such as TRACE-2D [5–7], a time dependent bunched beam simulation with periodic boundary condition [8], or a simplified model with infinite beam pipe length [9]. A fully three-dimensional self-consistent simulation of multiple charge state ion beam transport in the LEBT system will help to optimize the design of the transport system and to minimize the particle losses in such a system.

2. Physical model and computational methods

The physical system in this model is a low energy ion beam transport system. A plot of a LEBT system together with the plasma ion source at the Lawrence Berkeley National Laboratory is given in Fig. 1 [10]. It consists of a high voltage extraction system for ion beam formation from the plasma ion source, a solenoid magnetic lens for transverse focusing and a double-focusing analyzing bend magnet for charge selection. Low energy ion beam transport inside the LEBT system is different from the beam transport inside a radio-frequency (RF) linac. Inside the RF linac the beam is longitudinally bunched

^{*} Corresponding author.

E-mail address: jqiang@lbl.gov (J. Qiang).

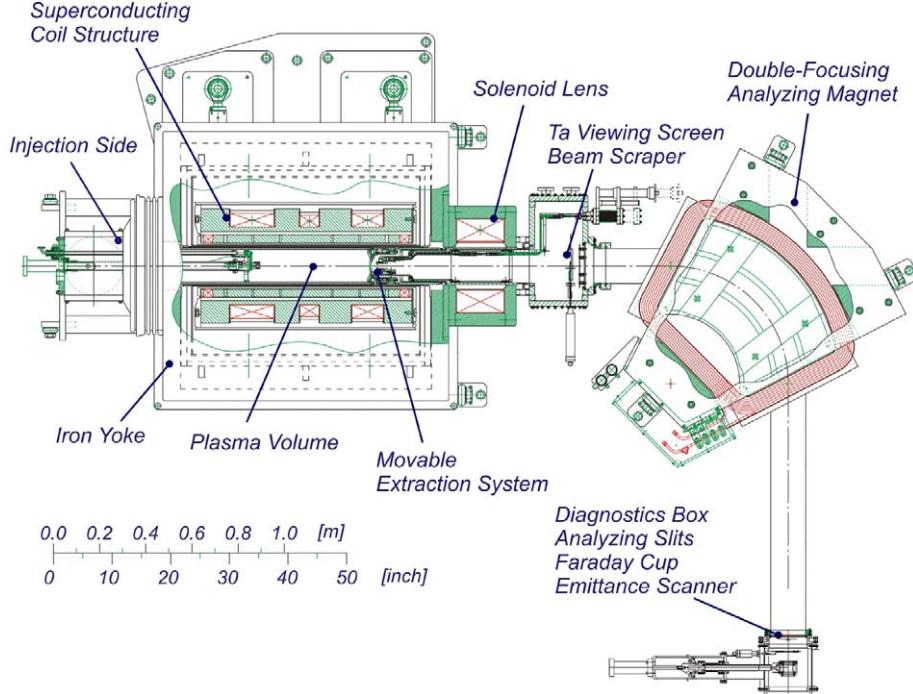


Fig. 1. A schematic plot of the low energy beam transport system together with the plasma ion source at Lawrence Berkeley National Laboratory.

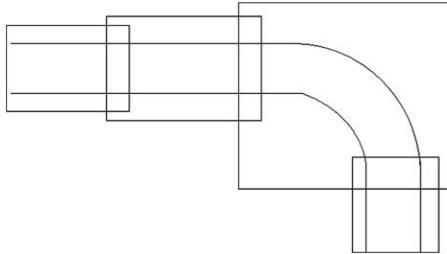


Fig. 2. A schematic plot of the overlapped multiple computational subdomains.

to a few millimeters by the time dependent RF fields. Inside the LEBT system, there is no longitudinal bunching, the particles extend longitudinally through the whole system to form a continuous steady state beam. The length of beam could be from a few metres to ten metres. To model the particle transport in such a system, we need to solve the time-independent Vlasov–Poisson equations including correct transverse boundary conditions. A brute force approach is to model the whole system as one computational domain directly. However, this is computationally impractical if a good numerical precision is required. For example, a very high numerical resolution is needed in order to accurately model the plasma surface at the exit of the ion source. This high accuracy may not be needed through the whole system. Furthermore, the particles at the beginning may not affect the particles near the end due to the large longitudinal-to-transverse aspect ratio and the shielding of transverse conducting wall. Hence, we can divide the whole beam into multiple overlapped segments. Fig. 2 shows a schematic plot of the overlapped multiple computational subdomains. Here, each subdomain overlaps with the neighboring subdomains. For each segment, we solve the time-independent

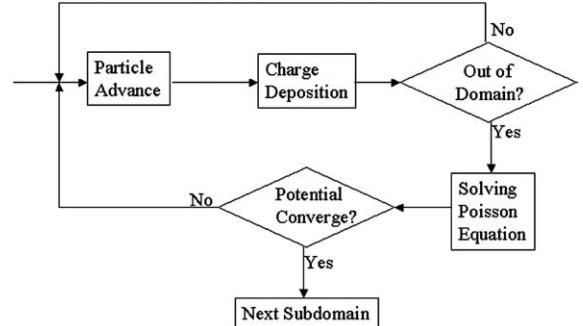


Fig. 3. A flow diagram of the iterative ray tracing procedure.

Vlasov–Poisson equations with the Dirichlet boundary conditions on the left end and the Neumann boundary conditions on the right end. The left end of the segment is chosen inside the domain of the preceding segment so that the potential and the density distribution obtained in the preceding segment are used as the left boundary conditions of this segment. Within each segment, an iterative particle-tracking method has been used to obtain the converged solution of the time-independent Vlasov–Poisson equations. The flow diagram of the iteration procedure is given in Fig. 3. The particles are advanced through the space by solving the Lorentz equations of motion for each particle subject to the external fields and the space-charge forces. These particles are deposited onto the computational grid to obtain the charge density distribution on the mesh. After all of the particles have passed through the local subdomain, the Poisson equation is solved to obtain the space-charge potential generated by the ion beam itself. The particle tracking is redone using the calculated new space-charge fields and the applied external fields of the beam transport system. This procedure is repeated until the

change of the space-charge potential between two iterations is below specified error tolerance level and the particle trajectories have converged.

In the process of particle tracking, the Lorentz equations of motion are solved in the Frenet–Serret coordinate system since it is a convenient coordinate system for specifying particle motion inside both the bending magnet and the straight section shown in Fig. 1. In this coordinate system, the equations of motion are written as [11,12]:

$$\frac{dp_x}{dt} = v_z \frac{h}{1 + hx} p_z + F_x, \quad (1)$$

$$\frac{dp_y}{dt} = F_y, \quad (2)$$

$$\frac{dp_z}{dt} = -v_z \frac{h}{1 + hx} p_x + F_z, \quad (3)$$

$$\frac{dx}{dt} = \frac{p_x}{m\gamma}, \quad (4)$$

$$\frac{dy}{dt} = \frac{p_y}{m\gamma}, \quad (5)$$

$$\frac{dz}{dt} = \frac{p_z}{m\gamma(1 + hx)}, \quad (6)$$

where x corresponds to the horizontal direction, y corresponds to the vertical direction, z corresponds to the longitudinal direction, $p_{x,y,z}$ is the x , y , and z component of mechanic momentum, $F_{x,y,z}$ is the corresponding force of each component, γ is the relativistic factor $\gamma = 1.0/\sqrt{1 - (\dot{\mathbf{r}}/c)^2}$, c is the speed of light in the vacuum, and h is the curvature of the bending magnet with a bend plane in $x-z$ (i.e. horizontal) plane. Inside the straight section, curvature $h=0$, the Frenet–Serret coordinates reduce into the standard Cartesian coordinates. The above non-linear equations of motion are solved using a modified leap-frog algorithm. The charge density on the grid is obtained from the summation of linear volume weighted deposition scheme in the cylindric coordinate system inside the straight sections and in the Frenet–Serret coordinate system inside the bending magnet section.

2.1. Solution of the 3D Poisson equation in cylindrical coordinates

A major part of the LEBT system, such as extraction region and solenoid focusing region, has a cylindrical geometry with azimuthal symmetry. For such a system, we can write the Poisson equation in cylindric coordinates as:

$$\frac{\partial^2 \phi}{\partial r^2} + \frac{1}{r} \frac{\partial \phi}{\partial r} + \frac{1}{r^2} \frac{\partial^2 \phi}{\partial \theta^2} + \frac{\partial^2 \phi}{\partial z^2} = -\rho/\epsilon_0. \quad (7)$$

Here, ϕ denotes the electrostatic potential generated by the beam itself, ρ the charge density function, r and z the radial and longitudinal distance. Since both the electric potential and the charge density are periodic function of θ , we can approximate the potential ϕ and source term ρ as:

$$\rho(r, \theta, z) = \sum_{m=-N_m/2}^{N_m/2-1} \rho^m(r, z) \exp(-im\theta), \quad (8)$$

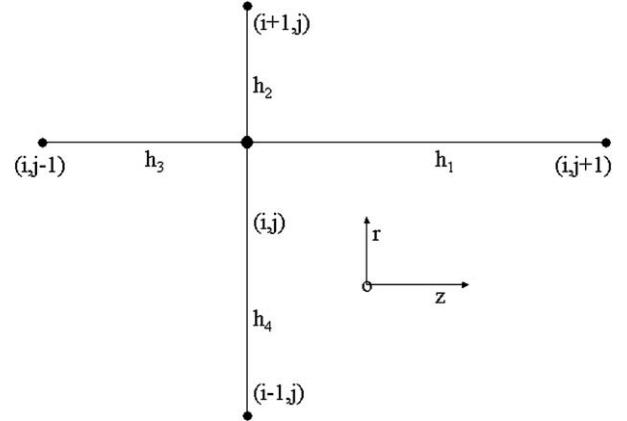


Fig. 4. A schematic plot of an irregular discretization point inside the computational domain.

$$\phi(r, \theta, z) = \sum_{m=-N_m/2}^{N_m/2-1} \phi^m(r, z) \exp(-im\theta). \quad (9)$$

Substituting Eqs. (8) and (9) into the Poisson equation (7), we obtain a group of decoupled two-dimensional partial differential equations in (r, z) as:

$$\frac{\partial^2 \phi^m}{\partial r^2} + \frac{1}{r} \frac{\partial \phi^m}{\partial r} - \frac{m^2}{r^2} \phi^m + \frac{\partial^2 \phi^m}{\partial z^2} = -\rho^m/\epsilon_0. \quad (10)$$

The boundary conditions at the radial edge are assumed as the Dirichlet boundary conditions with given potentials on the conducting wall or as the Neumann boundary conditions if there is no conducting wall. On the axis, the boundary conditions are

$$\frac{\partial \phi^m(0, \theta, z)}{\partial r} = 0 \quad \text{for } m = 0, \quad (11)$$

$$\phi^m(0, \theta, z) = 0 \quad \text{for } m > 0. \quad (12)$$

The boundary conditions at the left edge of the computational domain are assumed as the Dirichlet boundary conditions obtained from the preceding subdomain and as the Neumann boundary conditions at the right edge of the computational domain.

The above two-dimensional partial differential equations are solved using a finite difference multigrid method for each azimuthal mode m . The computational domain for the extraction region of the LEBT system contains electrodes with irregular shapes. For the grid points near the conducting surface, the mesh size could be different in r and z directions. An irregular discretization along r and z is needed. A schematic plot of an irregular discretization point is given in Fig. 4. For such a discretization scheme, the differential operator in Eq. (10) can be approximated as:

$$\frac{\partial \phi^m}{\partial r} \approx \frac{1}{h_2 h_4 (h_2 + h_4)} (\phi_{i+1,j}^m h_4^2 - \phi_{i-1,j}^m h_2^2 + \phi_{i,j}^m (h_2^2 - h_4^2)), \quad (13)$$

$$\frac{\partial^2 \phi^m}{\partial r^2} \approx \frac{2}{h_2 + h_4} \left(\frac{1}{h_2} (\phi_{i+1,j}^m - \phi_{i,j}^m) + \frac{1}{h_4} (\phi_{i-1,j}^m - \phi_{i,j}^m) \right), \quad (14)$$

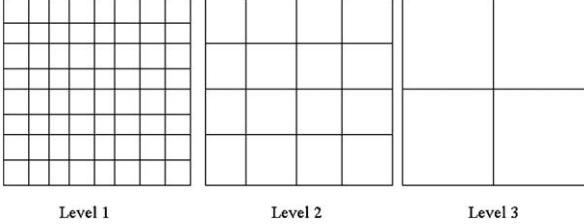


Fig. 5. Three-level discretization of a 2D computational domain.

$$\frac{\partial^2 \phi^m}{\partial z^2} \approx \frac{2}{h_1 + h_3} \left(\frac{1}{h_1} (\phi_{i,j+1}^m - \phi_{i,j}^m) + \frac{1}{h_3} (\phi_{i,j-1}^m - \phi_{i,j}^m) \right), \quad (15)$$

where h_1 and h_3 denote the uneven mesh size in the z direction, h_2 and h_4 denote the uneven mesh size in the r direction. Substituting these approximations into Eq. (10), we obtain a five point difference equation at grid point (i, j) as:

$$\begin{aligned} & \left(\frac{2r_{i,j}^2}{h_2 h_4} + \frac{2r_{i,j}^2}{h_1 h_3} + m^2 - \frac{r_{i,j}(h_2 - h_4)}{h_2 h_4} \right) \phi_{i,j}^m \\ &= r_{i,j}^2 \frac{\rho_{i,j}}{\epsilon_0} + \left(\frac{2r_{i,j}^2}{h_2(h_2 + h_4)} + \frac{r_{i,j} h_4}{h_2(h_2 + h_4)} \right) \phi_{i+1,j} \\ &+ \left(\frac{2r_{i,j}^2}{h_4(h_2 + h_4)} - \frac{r_{i,j} h_2}{h_4(h_2 + h_4)} \right) \phi_{i-1,j} \\ &+ \frac{2r_{i,j}^2}{h_1(h_1 + h_3)} \phi_{i,j+1} + \frac{2r_{i,j}^2}{h_3(h_1 + h_3)} \phi_{i,j-1}. \end{aligned} \quad (16)$$

This equation reduces to the standard second-order five point difference equation for the grid points away from the conducting surface where $h_1 = h_3 = h_z$ and $h_2 = h_4 = h_r$.

The linear algebraic equation (16) for each grid point inside the computational domain is solved using an iterative multigrid method [13–15]. The multigrid method is an iterative method based on the concept of smoothing out numerical iteration errors on multiple resolution scales. Instead of solving the original discrete Poisson equation on one fixed mesh size, the multigrid method solves the discrete Poisson problem on multiple levels of mesh size. Fig. 5 shows an example of three level discretization of a 2D computational domain. As the discretization level increases, the mesh size increases and the number of unknowns inside the computational domain decreases exponentially. At level three, only one unknown is left for the Dirichlet boundary conditions, which can be solved directly.

The discrete Poisson equation at one level of mesh size can be written in a general form of linear algebraic equation:

$$A^{(i)} X^{(i)} = B^{(i)}, \quad (17)$$

where $A^{(i)}$ is the discrete Poisson operator at level i , $X^{(i)}$ denotes the unknown solution vector at the finest level or the unknown correction vector at other levels, and $B^{(i)}$ is the source vector at the finest level or the residual vector at other levels. The multigrid algorithm using two grid levels consists of five basic operations: pre-smoothing, restriction, evaluation, prolongation, and post-smoothing. During the pre-smoothing stage, an

approximate solution of $X^{(i)}$, $\tilde{X}^{(i)}$, is obtained using a classical iterative method such as the Gauss–Seidel iteration:

$$\tilde{X}^{(i)} = (D^{(i)} + L^{(i)})^{-1} U^{(i)} \tilde{X}^{(i)} + (D^{(i)} + L^{(i)})^{-1} B^{(i)}, \quad (18)$$

where $D^{(i)}$, $L^{(i)}$ and $U^{(i)}$ correspond to the main diagonal part, below diagonal part and above diagonal part of the matrix $A^{(i)}$. The residual vector $r^{(i)}$ at this level is calculated from the approximate solution as

$$r^{(i)} = B^{(i)} - A^{(i)} \tilde{X}^{(i)}. \quad (19)$$

These residuals are interpolated from the fine grid i to the coarse grid $i + 1$ using a restriction operator \mathcal{R} to obtain the right-hand side of Eq. (17):

$$B^{(i+1)} = \mathcal{R} r^{(i)}. \quad (20)$$

Here, a linear restriction operator \mathcal{R} on a 2D grid is defined as:

$$\mathcal{R} = \begin{pmatrix} \frac{1}{16} & \frac{1}{8} & \frac{1}{16} \\ \frac{1}{8} & \frac{1}{4} & \frac{1}{8} \\ \frac{1}{16} & \frac{1}{8} & \frac{1}{16} \end{pmatrix} \quad (21)$$

which corresponds to a bilinear nine-point interpolation scheme. The evaluation operation on coarse grid $i + 1$ will solve the discrete Poisson equation for the correction vector through a direct or an iterative method. The obtained correction is re-interpolated back to the fine grid i from the coarse grid $i + 1$ using a prolongation operator. The improved solution on grid level i is given by

$$\tilde{X}_{\text{new}}^{(i)} = \tilde{X}^{(i)} + \mathcal{P} \tilde{X}^{(i+1)}, \quad (22)$$

where \mathcal{P} is the prolongation operator:

$$\mathcal{P} = \begin{pmatrix} \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ \frac{1}{2} & 1 & \frac{1}{2} \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \end{pmatrix} \quad (23)$$

which also corresponds to a bilinear interpolation scheme giving nonzero values at nine grid points. This new approximate solution is then used in the classical iterative method as a post-smoothing stage to further improve the accuracy of the solution. If the discrete equation on the coarse grid $i + 1$ can be solved using an evaluation operation, only two grid levels are used, and the algorithm is referred to as two-grid method. If the solution on the coarse grid $i + 1$ cannot be easily attained, the evaluation step can be replaced by more two-grid iterations. Depending on how many two-grid iterations are used when each time the number of grid levels is increased by one, the multigrid iteration can have a V cycle (one two-grid iteration is used) or a W cycle (two two-grid iterations are used) structure [15]. Fig. 6 gives a schematic plot of a V cycle and a W cycle structure with four grid levels. Here, S denotes a smoothing operation, E denotes an evaluation operation, each descending line \ denotes a restriction operation, and each ascending line / denotes a prolongation operation.

In the multigrid method, the iteration can start from finest grid level or start from the coarsest grid level. If a good initial guess of the solution is available, starting from the finest

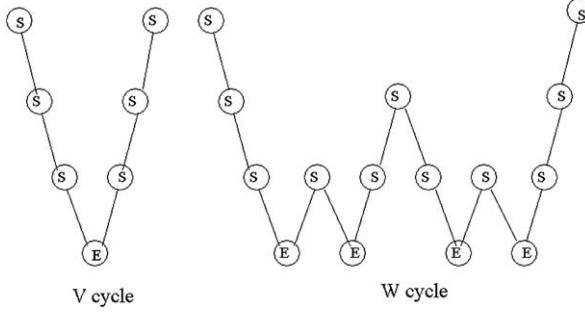


Fig. 6. A schematic plot of a V cycle and a W cycle iteration structure in the multigrid algorithm using four grid levels [15].

grid will be an appropriate method. Otherwise, starting from the coarsest grid will be more efficient since the solution on the coarsest grid can be obtained from the direct evaluation and interpolated to the next finer grid level. The interpolated solution on the finer grid level is used to start the smoothing operation at that level. After some V or W cycle of iterations, the solution at that level is further interpolated to next even finer grid level to start a new smoothing operation and iteration cycle. Such a process is repeated for a number of times until the finest grid level is reached. This type of multigrid iteration is called full multigrid method or nested iteration.

The multigrid method uses more grid levels than the conventional single grid iteration methods. This seems to be more computationally expensive than the single grid iteration. However, by changing the resolution of the discretization, i.e. scale of resolution, from one level to the next level, the low frequency errors in the numerical residues of the iteration can be removed by a coarse grid iteration, while the high frequency errors can be resolved on a fine grid iteration. Therefore, the multigrid method uses much less number of iterations on the finest grid level to obtain the converged solution than the single grid iteration method. Most computational work is done on coarse grids with much less number of operations on each grid level compared with the finest grid level iteration. It has been shown that the computational cost of this method scales linearly with the number of grid points [14]. Hence, the multigrid iteration provides a much faster convergence than the conventional iterative method such as SOR method. As a comparison, we have solved a 2D Laplace equation with Dirichlet boundary conditions using the multigrid method and the classic SOR ($\omega = 1.96$) method. Fig. 7 shows the computing time as a function of number of grid points using both methods. It is seen that the multigrid method scales linearly with the number of grid points while the SOR scales exponentially with the number of grid points. The computing time on the same number of grid points using the SOR is much higher than that using the multigrid method.

In this work, we have extended a 2D multigrid solver developed by Fortuna to include the Neumann boundary condition and the irregular geometry shape of the electrode in the extraction region [16]. We have used a point red and black Gaussian-Seidel iteration method as pre and post smoother on each level. For the first iteration during each particle tracking, we have

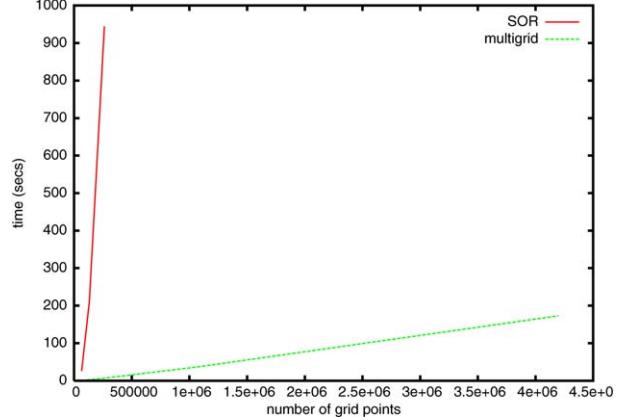


Fig. 7. Computing time as a function of grid point number for solving a 2D Laplace equation with Dirichlet boundary conditions using the SOR and the multigrid method.

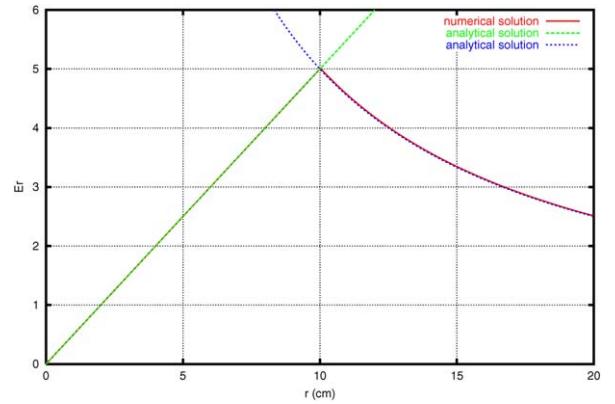


Fig. 8. Radial electric field from the numerical solution and from the analytical solution.

used the nested multigrid algorithm with a W cycle to solve the discrete Poisson equation (Eq. (16)) for each azimuthal mode since the initial guess of the solution on the finest grid level is not easily obtained. For the following iteration, we have started the iteration from the finest grid with the initial guess from the previous iteration. We have also used a bilinear interpolation for prolongation and restriction during the multigrid iteration. This solver is first tested with a simple uniform round beam inside a conducting pipe since the analytical solution can be obtained for this case. Here, we have used a Neumann boundary conditions at both ends. The numerical solution together with the analytical solution is shown in Fig. 8. The agreement between the numerical and the analytical solution is very good. As a second test, we calculated the potential inside the extraction region with a 20 kV plasma electrode, a -1 kV puller electrode, and a 0 kV ground electrode. The electric potential on the axis as a function of distance is shown in Fig. 9 together with a solution using the code WARP [17]. The agreement between the solutions obtained by the two different codes, which use different Poisson solvers, is quite good. The maximum relative difference is about 1%.

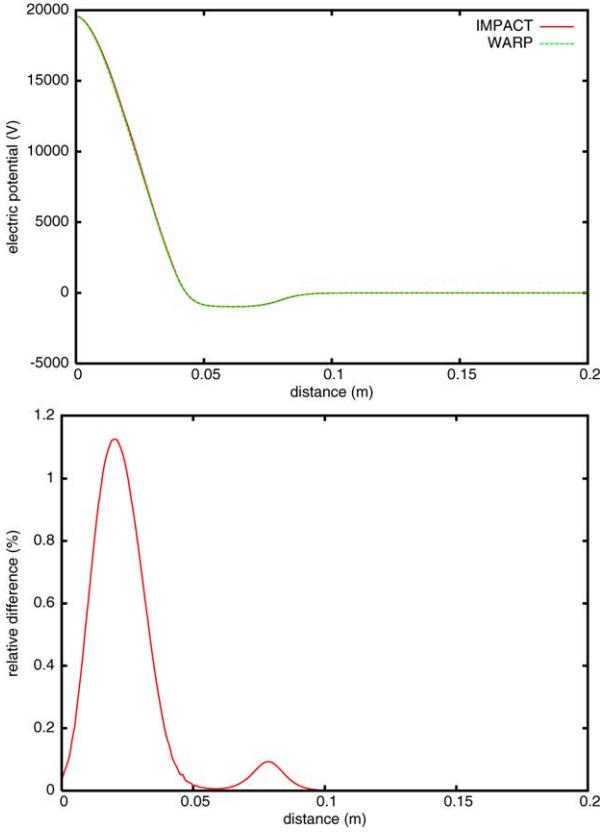


Fig. 9. Electric potential on the axis (top) and relative potential difference (bottom) from the new solver and from the WARP solver.

2.2. Solution of the 3D Poisson equation in Frenet–Serret coordinates

The preceding Poisson solver is applicable for a round system with azimuthally symmetric geometry. For some element in the LEBT system, such as the bending magnet, it has a rectangular cross-section with a horizontal bending angle. The bending magnet provides not only a transverse focusing of the ion beam but also a selection of different charge states for further transport. Inside this element, we use the Frenet–Serret coordinates since these coordinates reduce to the normal Cartesian coordinates when the curvature is zero. The 3D Poisson equation in this coordinate system can be written as:

$$\frac{1}{1+hx} \left(\frac{\partial}{\partial x} (1+hx) \frac{\partial \phi}{\partial x} \right) + \frac{\partial^2 \phi}{\partial y^2} + \frac{1}{1+hx} \left(\frac{\partial}{\partial z} \left(\frac{1}{1+hx} \frac{\partial \phi}{\partial z} \right) \right) = -\frac{\rho(x, y, z)}{\epsilon_0}, \quad (24)$$

where h is the curvature of the bending magnet, x corresponds to the horizontal direction, y corresponds to vertical direction and z corresponds to longitudinal direction. For perfect conducting plates, the electric potential will vanish on vertical walls. We can approximate the electric potential and the charge density distribution as:

$$\rho(x, y, z) = \sum_{m=1}^{N_m} \rho^m(x, z) \sin(m\pi y/a), \quad (25)$$

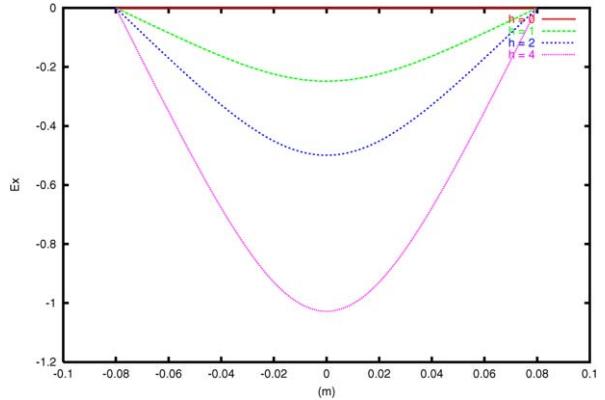


Fig. 10. Horizontal electric field along the vertical axis with a bended rectangular pipe with different curvatures.

$$\phi(x, y, z) = \sum_{m=1}^{N_m} \phi^m(x, z) \sin(m\pi y/a), \quad (26)$$

where a is the full vertical aperture width. Substituting these equations into the Poisson equation (24) and using the orthogonal condition of sine functions, we obtain a group of decoupled two-dimensional partial differential equations:

$$\frac{1}{1+hx} \left(\frac{\partial}{\partial x} (1+hx) \frac{\partial \phi^m}{\partial x} + \frac{\partial}{\partial z} \frac{1}{1+hx} \frac{\partial \phi^m}{\partial z} \right) - \frac{m^2 \pi^2}{a^2} \phi^m = -\frac{\rho^m(x, z)}{\epsilon_0}. \quad (27)$$

The resulting two-dimensional partial differential equations are solved using the same finite difference multigrid method described before.

As a test of this Poisson solver, we calculated the electric fields inside a bended conducting pipe with a longitudinal uniform and transverse Gaussian distribution beam and different bending curvatures. Fig. 10 shows the horizontal electric field as a function of y at $x = 0$ for different bending curvatures. We see that as the curvature approaches to zero, the horizontal field vanishes due to the symmetry of the beam density distribution.

3. Applications

The simulation tool developed here has been applied to the study of the ion beam formation and ion beam transport out of the superconducting ECR ion source VENUS at the Lawrence Berkeley National Laboratory [10]. It consists of an extraction region, a solenoid focusing lens, and a bending analyzing magnet for charge selection. The total length is about 3.5 meters. As a first application of the computational model developed here, we have done a simulation of a hydrogen ion beam formation from the ECR ion source. We have chosen a computational domain containing the extraction region and about 15 Debye lengths inside the plasma source from the plasma aperture so that the formation of ion beam is not sensitive to the location of the plasma boundary. The boundary condition at the left edge of the domain is set by the plasma potential ϕ_p which can be

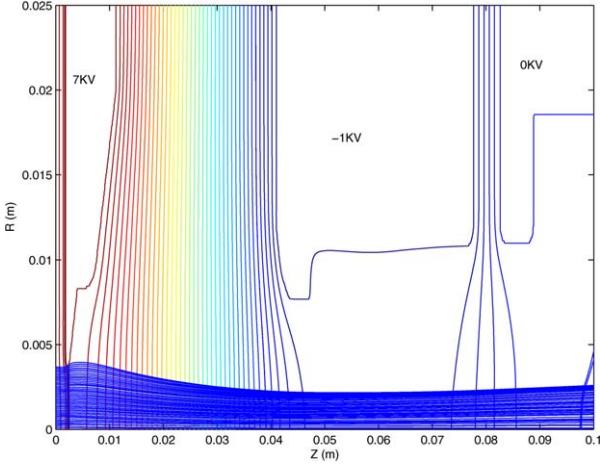


Fig. 11. Transverse rms beam size as a function of distance from the 3D simulation and 2D simulation for a beam with 1 mA current.

found from [18]:

$$\phi_p = \phi_w - kT_e \log \left(\sum_{i=1}^N \left(\frac{n_{i0}}{n_{e0}} \sqrt{\frac{\pi T_i m_e}{T_e M_i}} \right) \right), \quad (28)$$

where k is the Boltzmann constant, ϕ_w is the electric extraction potential on the wall, M_i is the mass of ion species i , m_e is the mass of electron, T_i is the ion temperature, n_{i0} is the ion density inside the plasma source, n_{e0} is the electron density inside the plasma source, and N is the total number of ion species. The boundary condition on the right edge of the domain is set as the Neumann boundary conditions. We have also assumed that the electrons in the plasma follow a Boltzmann density distribution:

$$n_e = n_{e0} \exp \left(-\frac{\phi_p - \phi}{kT_e} \right), \quad (29)$$

where ϕ is the space-charge electric potential from the solution of the Poisson equation including the contributions from both the electrons and the ions. During each numerical iteration, the electron density is calculated using the electric potential from last iteration following the Boltzmann ansatz equation (29). This electron density is added to the ion density from particle tracking to obtain the total charge density for the Poisson equation. As an example, Fig. 11 shows a hydrogen ion beam formation using an extraction voltage of 7 kV. The self-consistent plasma boundary can be seen in the figure. The plasma density and the extraction fields are nicely matched in this case producing an almost flat plasma boundary.

As another example of applications, we have simulated a 2 mA H^+ and H_2^+ ion beam transport through the LEBT system including the double focusing analyzing magnet. The bend magnet is treated fully three-dimensionally in this case. External fields from a OPERA-3d vector field calculation are used to describe magnetic fields of the bend [19]. We have used 20,480 macroparticles and five overlapped segments in the simulation. The transverse rms size for one charge state of the beam (H^+) as a function of distance is shown in Fig. 12. There is a double focusing in both horizontal and vertical directions of the

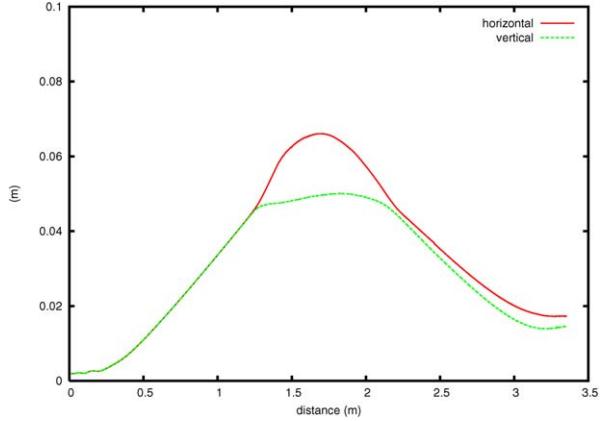


Fig. 12. Transverse rms beam size as a function of the VENUS LEBT system distance.

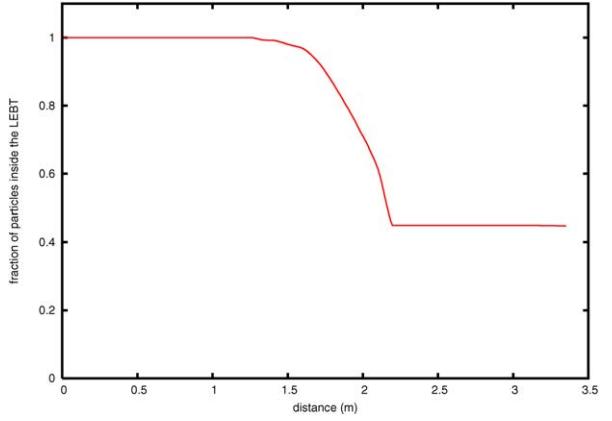


Fig. 13. Fraction of ion beam as a function of the VENUS LEBT system distance.

beam after the bending magnet. Fig. 13 shows the fraction of particles survives inside the LEBT system. With the chosen bending magnet strength set for the H^+ , all the H_2^+ particles are lost inside the magnet.

4. Conclusions

In this work, we have developed a three-dimensional model to self-consistently simulate the ion beam formation from plasma ion sources and the ion beam transport in low energy beam transport systems through a bending magnet. We have used an overlapped multi-section model to break the original large system into a number of small subsystems. We have developed two new three-dimensional Poisson solvers to calculate the space-charge forces generated by the beam inside the subsystem self-consistently. These new three-dimensional Poisson solvers based on a combination of spectral method and finite difference multigrid method are more computational efficient than the finite difference SOR method used in previous studies. We have also applied the model to studies of the hydrogen ion beam formation and transport from the ECR ion source at LBNL. The model developed here can have important application in design and optimization of the low energy beam line optics of the proposed Rare Isotope Accelerator front end.

Acknowledgements

We would like to thank Drs. A. Friedman, D. Grote, C.M. Lyneis, R.D. Ryne, and J. Vay for discussions and Dr. M.A. Furman for reading the manuscript. We also thank Dr. Fortuna for the open source 2D multigrid code, mgsor. This research used resources of the National Energy Research Scientific Computing Center, and the resources of the Center for Computational Sciences at Oak Ridge National Laboratory. This work was supported by the Director, Office of Energy Research, Office of High Energy and Nuclear Physics, Nuclear Physics Division of the U.S. DOE under Contract DE-AC02-05CH11231.

References

- [1] J.H. Whealton, R.W. McGaffey, P.S. Meszaros, *J. Comp. Phys.* 63 (1986) 20.
- [2] R. Becker, W.B. Herrmannsfeldt, *Rev. Sci. Instrum.* 63 (1993) 2756.
- [3] J.E. Boers, *Proc. Particle Accelerator Conf.* (1993) 327.
- [4] P. Spadte, *Rev. Sci. Instrum.* 65 (1994) 1419.
- [5] K.R. Crandall, D.P. Rusthoi, LA-UR-97-886, Los Alamos National Laboratory Report, May 1997.
- [6] D. Wutte, M.A. Leitner, C.M. Lyneis, C.E. Taylor, Z.Q. Xie, in: *Proc. Eighth International Conference on Heavy-Ion Accelerator Technology*, October 5–9 1998, Argonne, IL.
- [7] S. Gammino, G. Ciavola, L. Celona, L. Ando, S. Passarello, X.Zh. Zhang, P. Spaedtke, M. Winkler, *Rev. Sci. Instrum.* 75 (2004) 1637.
- [8] Q. Zhao, et al., in: *Proc. 16th International Workshop on ECR Ion Sources*, September 26–30 2004, Berkeley, CA, in: *AIP Conf. Proc.*, vol. 749, 2004, p. 242.
- [9] T. Thuillier, et al., in: *Proc. 16th International Workshop on ECR Ion Sources*, September 26–30, 2004, Berkeley, CA, in: *AIP Conf. Proc.*, vol. 749, 2004, p. 41.
- [10] D. Leitner, C.M. Lyneis, S.R. Abbott, D. Collins, R.D. Dwinell, M.L. Galloway, M. Leitner, D.S. Todd, *Nucl. Instrum. Methods Phys. Res. Sect. B* 235 (2005) 486.
- [11] R. Li, Ya.S. Derbenev, *JLAB-TN-2-054*, 2002.
- [12] A.C. Kabel, in: *Proc. 2003 Particle Accelerator Conference*, Portland, OR, 2003, p. 2252.
- [13] W. Hackbusch, *Multi-Grid Methods and Applications*, Springer-Verlag, New York, 1985.
- [14] P. Wesseling, *An Introduction to Multigrid Methods*, John Wiley & Sons, Chichester, 1992.
- [15] W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, *Numerical Recipes in FORTRAN: The Art of Scientific Computing*, second ed., Cambridge University Press, Cambridge, 1992.
- [16] A.O. Fortuna, <http://www.mgnet.org/mgnet-codes-fortuna.html>, 1998.
- [17] A. Friedman, D.P. Grote, I. Haber, *Phys. Fluids B* 4 (1992) 2203.
- [18] R. Becker, *Rev. Sci. Instrum.* 63 (4) (1992) 2756.
- [19] Opera-3D version 10.0, Vector Fields Limited, 24 Bankside, Kidlington, Oxford OX5 1JE, England.