Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 225 (2007) 808–828
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

A new algorithm for charge deposition for multiple-grid method for PIC simulations in r–z cylindrical coordinates

Christophe Cornet *, Dixon T.K. Kwok

Applied and Plasma Physics Group, School of Physics, A28, University of Sydney, NSW 2006, Australia

Received 5 September 2005; received in revised form 7 December 2006; accepted 3 January 2007
Available online 16 January 2007

Abstract

A mesh of nodes used in particle-in-cell simulations may be refined in some regions to obtain better local spatial resolution without adding excessive computational cost. Each refinement can be seen as a finer grid in a multiple-grid system. The standard bilinear weighting method in two-dimensional (r–z) cylindrical coordinates using a multiple-grid system is detailed and an inadequacy in the method is presented. A new algorithm of particle to node weighting in a multiple-grid system is then described and compared to the standard bilinear weighting method in two-dimensional (r–z) cylindrical coordinates. This description pays particular attention to the nodes near the interface between grids of differing cell-size. The volume weighting method used in this study is geometrically similar to the standard bilinear PIC method in Cartesian coordinates, but weights in proportion to the volumes of revolution of the areas about the z-axis, rather than the areas themselves. As the volume weighting method has the axisymmetric geometry built in, it is a natural system to use in this particular case. A full particle-in-cell (PIC) simulation of metal plasma-immersion ion implantation and deposition (MePIIID) of a flat stage using three grids of cell size: 0.5×0.5, 1×1 and 2×2 mm; is performed and results are compared to those obtained from a single-grid simulation with the same parameters. The multiple grid method compares very well to the single-grid simulation.

Keywords: PIC; Multiple grid; Weighting; Volume weighting; Cylindrical

1. Introduction

Particle-in-cell (PIC) simulations have been extensively used in recent years for the modeling of plasmas [1–3]. It typically involves weighting particles to a uniform grid of nodes, calculating the electromagnetic fields on these nodes and interpolating the forces back to the particles [3]. When there is a need for higher spatial resolution in PIC simulations, it is common practice to reduce the cell size. Unfortunately this comes at a greatly increased cost to computational power.

In situations where detail is needed in one part of the simulation region and not another, one can use two or more grids of differing cell size. This has the advantage of using fewer nodes in the simulation, thereby

* Corresponding author. Tel.: +61 2 9351 3631; fax: +61 2 9351 7726.
E-mail address: cornet@physics.usyd.edu.au (C. Cornet).

0021-9991/$ - see front matter © 2007 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2007.01.004

reducing computation time, without sacrificing detail where it is needed. This method has been successfully implemented in various finite-difference simulations [4,5]. We aim to merge this multiple-grid method and the well-established PIC method. The calculation of the electric potential on a multiple-grid system has been investigated [6], and particle-mesh accumulation in each grid can be performed by the standard bilinear PIC volume-weighting method [3,7]. Adaptive mesh refinement to particle-in-cell simulations has been applied to plasmas and beams [8]. However, we show that particle-mesh accumulation becomes non-trivial for the interface nodes between two grids.

Our ultimate goal is to investigate metal plasma-immersion ion implantation and deposition on a sharp cone mounted on top of a disc-shaped target that is supported by a thin rod, similar to previous work [9,10]. In metal plasma immersion ion implantation, the nearly fully ionized metal plasma generated from a cathodic arc has a drifting velocity of 1.3×10^4 ms^{-1} [11] and an average charge state of 2.03 [9,12]. The electrons are in thermal equilibrium with a temperature of a few eV. Negative voltage pulses of a few to tens of kV are applied to the sample stage implanting the metal ions into the sample that sits on top of the stage. During the pulse-off periods, the ions are deposited on the sample surface and the whole process is referred to as Metal Plasma Immersion Ion Implantation and Deposition (MePIIID). The MePIIID on a disc-shaped sample stage was numerically simulated by the PIC method using a regular grid of 2 mm $\times$ 2 mm cells in two dimensions r–z cylindrical coordinates. Due to the axisymmetry of the problem, it is natural to use r–z cylindrical coordinates. The numerically estimated steady state ion sheath fitted well with the analytical Child–Langmuir law [2]. MePIIID of a cone placed on top of a disc-shaped sample stage was also numerically simulated by the PIC method with a regular grid of cell size 2 mm $\times$ 2 mm [1]. The 2 mm grid chosen was of sufficient resolution to simulate the motion of the ions in the plasma, but with this geometric resolution the cone tip was represented by a single point, leading to an anomalous electric field. To properly investigate the MePIIID of a cone we require the cone tip to be distributed over several grid points in a curved arc. Consequently, to correctly model this geometry we need a finer grid, which comes at a great computational cost. In order to resolve the tip region of the cone and maintain the 2 mm resolution for the rest of the simulated region, a three-level multiple-grid system of cell size 0.5, 1.0 and 2.0 was applied. We do not apply a fine grid of cell size 0.5 mm to the entire simulation region because of the high computational cost this would require, and the resolution obtained from the 2 mm grid had been shown to be sufficient to model plasma sheath phenomena [1].

In this paper, we focus on the establishment of a new algorithm for charge-deposition using multiple-grids in PIC simulations. In particular, we demonstrate the shortcomings of weighting a particle to the four nodes at the corners of the host cell. By addressing these shortcomings, we present another method and show that the results obtained from this second method are in agreement with previous work [1]. Systematic errors in cylindrical coordinates [13] further complicate calculations at each grid interface. A detailed study of these systematic errors has been performed by working from first principles [14]. We too use first principles to find a natural method of dealing with grid interfaces to obtain a method that can be used in one or two-dimensional PIC simulations in cylindrical coordinates.

2. One-dimensional PIC

In one-dimensional PIC simulations on a uniform grid in cylindrical coordinates, a systematic error has been described in Ref. [13]. This systematic error has been resolved in linear weighing, and can be extended to any weighting function [14]. In linear weighting, particles are weighted to the nodes in proportion to the distances between the particle and the nodes. In area-weighting [7], they are weighted in proportion to the areas swept out by revolving these distances about the origin. It is natural to use area-weighting as the axisymmetric geometry is implicit in the weighting function. To correct the systematic error, the effective area V_j of a node is calculated by weighting to the mesh using the same algorithm as the charge weighting [14],

$$V_j = a + b \tag{1a}$$

where a is an integral over the cell to the left of the node, and b the right.

$$a = \int_{r_{j-1}}^{r_j} 2\pi r \frac{r^2 - r_{j-1}^2}{r_j^2 - r_{j-1}^2}\,\mathrm{d}r = \frac{\pi}{2}(r_j^2 - r_{j-1}^2) \tag{1b}$$

$$b = \int_{r_j}^{r_{j+1}} 2\pi r \frac{r_{j+1}^2 - r^2}{r_{j+1}^2 - r_j^2}\,\mathrm{d}r = \frac{\pi}{2}(r_{j+1}^2 - r_j^2) \tag{1c}$$

The effective area is then

$$V_j = \frac{\pi}{2}(r_{j+1}^2 - r_{j-1}^2) \tag{2a}$$

The effective area of a node at the z-axis ($r = 0$) is

$$V_0 = \frac{\pi}{2} r_1^2 \tag{2b}$$

and a node at the edge of the outer boundary becomes

$$V_N = \frac{\pi}{2}(r_N^2 - r_{N-1}^2) \tag{2c}$$

The expressions derived here from area-weighting are simpler than those derived from linear weighting (Eqs. (19)–(21) in [14]). The above corrections to the effective area calculations were implemented and tested using a non-uniform grid in cylindrical coordinates. The results with a varying numbers of PIC particles in a uniform distribution, with an irregular grid are shown in Fig. 1. We see that using 100 particles in the simulation creates an error that peaks at around 4% in this particular case. More importantly, as the number of particles used increases, the maximum error drops quickly and the result converges towards the expected result for a uniform particle distribution. This convergence roughly scales as the square root of the average number of particles in each cell.

A particle receives an electric field contribution from the two nodes on either side of it. For instance, in Fig. 2, the particle p lies between the nodes j and $j+1$, therefore

$$E_p = E'_j + E'_{j+1} \tag{3a}$$

Each contribution is scaled by the same volume weighting shape function above. This is done to ensure the conservation of momentum and consequently avoid a self-force acting on the particles [3]. Therefore, for area-weighting

$$E'_j = E_j \frac{r_{j+1}^2 - r_p^2}{r_{j+1}^2 - r_j^2} \tag{3b}$$

$$E'_{j+1} = E_{j+1} \frac{r_p^2 - r_j^2}{r_{j+1}^2 - r_j^2} \tag{3c}$$

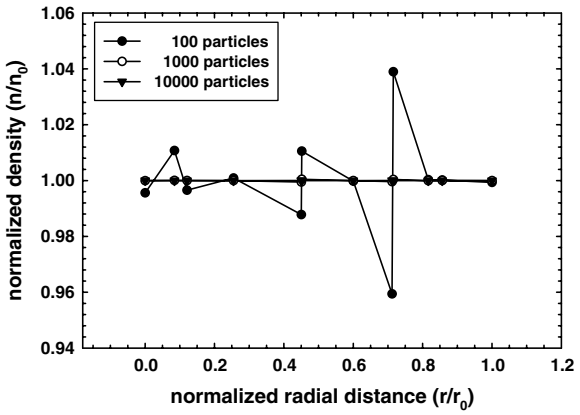

Fig. 1. Mesh densities computed using the method suggested in [14] using area-weighting on a non-uniform (random) mesh of 11 cells (12 nodes) in cylindrical coordinates. 100, 1000 and 10,000 particles were uniformly placed along the r-axis. As more particles are added to the simulation, the error decreases.

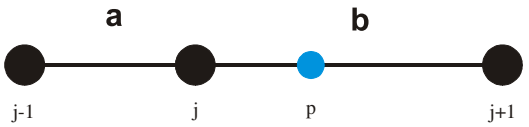

Fig. 2. A particle p positioned between the nodes j and $j+1$ weights its density to these two nodes; and the electric field calculated at each of these nodes contributes to the electric field of the particle.

The area-weighting method differs from linear interpolation in that the interpolated electric field is not a linear function of distance, as it should be in some cases, such as a uniform charge distribution. The deviation from linear interpolation is not dependent on cell size or number of particles, but is dependent on the number of cells from the central axis. The peak relative error of the area-weighting method in a linear field is

$$\Delta_{\max} = \frac{\frac{2n(n+1)}{2n+1} - \sqrt{n(n+1)}}{\sqrt{n(n+1)}} \tag{4}$$

where n is the number of whole cells away from the central axis. For the centermost cell ($n = 0$) where this equation breaks down, the relative error at a given position approaches unity as the position approaches the central axis. Nonetheless, for all other cells the relative error remains below 6%.

3. Volume-weighting in 2D cylindrical coordinates (r–z)

In two dimensions, the r–z plane is discretized by a uniform lattice of nodes. We use the same area-weighting method discussed in the previous section, but add the z-dimension thereby making it volume-weighting. In standard two-dimensional bilinear PIC weighting the particles in each cell are weighted to the four corner nodes of the cell [3]. The node (j,k) in Fig. 3, is on the corner of cells labeled A, B, C and D; and the charge density attributed to (j,k) is accumulated from each of these four cells

$$n_{j,k} = \frac{N_{\mathrm{A}} + N_{\mathrm{B}} + N_{\mathrm{C}} + N_{\mathrm{D}}}{a+b+c+d} \tag{5a}$$

The charge contributions N_{A} from the PIC particles $n_p(r,z)$ of an arbitrary continuum particle distribution in the A cell within the four cells are [14]

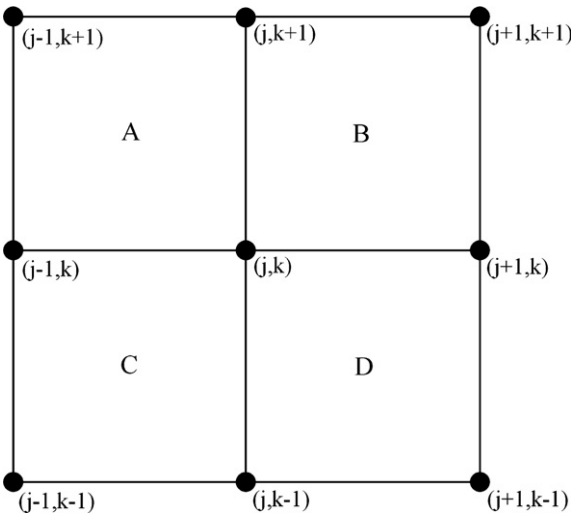

Fig. 3. In the standard method of bilinear weighting, each cell weights to its four corner nodes. Therefore the node (j,k) receives contributions from each of the A, B, C and D cells.

$$N_{\mathrm{A}} = \int_{z_k}^{z_{k+1}} \int_{r_{j-1}}^{r_j} n_p(r,z) \frac{\left(r^2 - r_{j-1}^2\right) \times (z_{k+1} - z)}{(r_j^2 - r_{j-1}^2) \times (z_{k+1} - z_k)} \,\mathrm{d}r\,\mathrm{d}z \tag{5b}$$

and likewise for the B, C and D cells. The contribution to the effective volume of the node can also be calculated [14]. The A cell has a contribution of

$$a = \int_{z_k}^{z_{k+1}} \int_{r_{j-1}}^{r_j} 2\pi r \frac{(r^2 - r_{j-1}^2) \times (z_{k+1} - z)}{(r_j^2 - r_{j-1}^2) \times (z_{k+1} - z_k)} \,\mathrm{d}r\,\mathrm{d}z = \frac{\pi}{4}(r_j^2 - r_{j-1}^2)(z_{k+1} - z_k) \tag{5c}$$

and similarly for the B, C and D cells. Thus for a uniform grid, the denominator in Eq. (5a) becomes

$$a + b + c + d = \frac{\pi}{4}(r_{j+1}^2 - r_{j-1}^2)(z_{k+1} - z_{k-1}) \tag{5d}$$

4. Multiple-grid and standard PIC weighting in 2D cylindrical coordinates (r–z)

In a multiple-grid system, a problem arises at the interface between two grids. In Fig. 4, a coarse grid is joined to a fine grid of half the cell-size. We see that there are two types of interface nodes, the node (j,k) is one type and the node $(j,k+1)$ is another. The node (j,k) is at the corner of four cells: A, B′, C and D′. Therefore by the standard bilinear weighting method, the PIC particles will only weight to the corners of the host cell, thus the particles in these four cells will contribute to the density of node (j,k). That is

$$n_{j,k} = \frac{N_{\mathrm{A}} + N'_{\mathrm{B}} + N_{\mathrm{C}} + N'_{\mathrm{D}}}{a + b' + c + d'} \tag{6a}$$

The contributions from the A and C cells are the same as the uniform grid case. However, the B′ and D′ cell contributions are modified to include the whole cell, which are twice as large in each dimension. The equations for the contributions from these cells are modified to

$$N'_{\mathrm{B}} = \int_{z_k}^{z_{k+2}} \int_{r_j}^{r_{j+2}} n_p(r,z) \frac{(r_{j+2}^2 - r^2) \times (z_{k+2} - z)}{(r_{j+2}^2 - r_j^2) \times (z_{k+2} - z_k)} \,\mathrm{d}r\,\mathrm{d}z \tag{6b}$$

$$N'_{\mathrm{D}} = \int_{z_{k-2}}^{z_k} \int_{r_j}^{r_{j+2}} n_p(r,z) \frac{(r_{j+2}^2 - r^2) \times (z - z_{k-2})}{(r_{j+2}^2 - r_j^2) \times (z_k - z_{k-2})} \,\mathrm{d}r\,\mathrm{d}z \tag{6c}$$

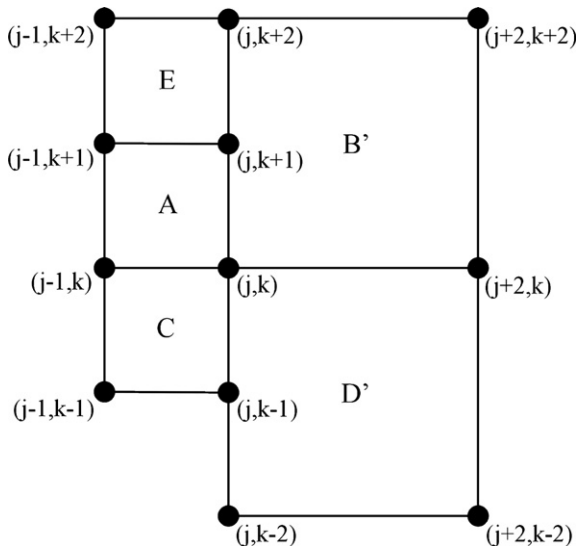

Fig. 4. At a multiple-grid interface, some nodes, e.g. (j,k), are at the corner of four cells, and other nodes, e.g. $(j,k+1)$, are at the corner of two cells. As cells weight to their four corners in the standard method, then the former receive contributions from four cells, and the latter receive contributions from only two cells.

and the effective volumes are

$$b' = \int_{z_k}^{z_{k+2}} \int_{r_j}^{r_{j+2}} 2\pi r \frac{(r_{j+2}^2 - r^2) \times (z_{k+2} - z)}{(r_{j+2}^2 - r_j^2) \times (z_{k+2} - z_k)} \, \mathrm{d}r \, \mathrm{d}z \tag{6d}$$

$$d' = \int_{z_{k-2}}^{z_k} \int_{r_j}^{r_{j+2}} 2\pi r \frac{(r_{j+2}^2 - r^2) \times (z - z_{k-2})}{(r_{j+2}^2 - r_j^2) \times (z_k - z_{k-2})} \, \mathrm{d}r \, \mathrm{d}z \tag{6e}$$

The node $(j, k+1)$ in Fig. 4 is another type of node, it is at the corner of only the A and E cells and not the B′ cell. By the standard bilinear weighting method, the particles in the B′ cell weight only to the corners of the B′ cell, therefore the node $(j, k+1)$ will not be weighted to by the particles within the B′ cell. Thus density at this node is accumulated from only the A and E cells, that is,

$$n_{j,k+1} = \frac{N_\mathrm{A} + N_\mathrm{E}}{a + e} \tag{7a}$$

$$N_\mathrm{E} = \int_{z_{k+1}}^{z_{k+2}} \int_{r_{j-1}}^{r_j} n_p(r,z) \frac{(r^2 - r_{j-1}^2) \times (z_{k+2} - z)}{(r_j^2 - r_{j-1}^2) \times (z_{k+2} - z_{k+1})} \, \mathrm{d}r \, \mathrm{d}z \tag{7b}$$

$$e = \int_{z_{k+1}}^{z_{k+2}} \int_{r_{j-1}}^{r_j} 2\pi r \frac{(r^2 - r_{j-1}^2) \times (z_{k+2} - z)}{(r_j^2 - r_{j-1}^2) \times (z_{k+2} - z_{k+1})} \, \mathrm{d}r \, \mathrm{d}z \tag{7c}$$

To test the validity of this method which corresponds to standard bilinear weighting, a cylindrical simulation chamber of radius 5 cm and height 10 cm was injected with a drifting uniform plasma from the top of the chamber, represented by a uniform particle distribution $n_p(r,z) = 2\pi r n_0$, [14] of density of $n_0 = 6 \times 10^{15}$ m^{-3}. A typical value for the drift velocity was used [9], $v_z = -1.3 \times 10^4$ ms^{-1}. Due to axisymmetry of the problem, the chamber was simulated in two-dimensional (r–z) cylindrical coordinates. The coarsest mesh had a grid spacing of 2 mm × 2 mm, this was refined to a medium-sized mesh of 1 mm × 1 mm at $r \leqslant 3$ cm, which had a further refined patch: $r \leqslant 1$ cm, 2 cm $\leqslant z \leqslant 6$ cm, where the grid spacing was 0.5 mm × 0.5 mm. As the purpose of this numerical test was to validate the described particle to grid weighting method, the electric field was not calculated nor weighted back to the particles. That is, the particles were simply drifting by their own momentum. The finest cells had 3 × 3 PIC particles uniformly spaced in them and the coarsest cells had 12 × 12 particles in them, the time step used was 0.92 ns. The particles were injected into the simulation region in the negative z-direction from the top, and were allowed to exit the simulation region from the right-hand side and the bottom. The results showed the expected analytic solution of a uniform plasma density throughout the entire chamber. However, in real plasma simulations using a drifting plasma obtained by a cathodic arc [9,12], the particles spread from the z-axis. Therefore it was important to have a test where the particles were injected with a slight radial velocity. The analytic result for the plasma density in this situation is

$$n(r,z) = \max\left\{0, n_0\left(1 + \frac{v_r}{v_z}\frac{Z_\mathrm{max} - z}{r}\right)\right\} \tag{8}$$

where n_0 is the density of the plasma being injected into the system, Z_max is the z-position of the top of the simulation region, and (v_r, v_z) is the r- and z-components of the initial ion velocity. The above mentioned standard method was tested with a radial velocity of $z_r = 65$ ms^{-1} and other parameters the same as in the uniform plasma simulation: $v_z = -1.3 \times 10^4$ ms^{-1}, $n_0 = 6 \times 10^{15}$ m^{-3} and $Z_\mathrm{max} = 10$ cm. Even in this simple expanding plasma simulation, a problem is evident near the grid interfaces (Fig. 5).

As shown in Fig. 5, at the boundary between the grids, the plasma density was not uniform. A plot of density along the vertical interfaces (Fig. 6) better shows the anomaly. The values alternate between two lines, which suggests that as the two types of interface nodes depicted in Fig. 4 had different weighting systems, then the calculations did not yield a similar result. The method clearly needed to be modified, and we aimed to find a method that deals with both types of interface nodes equally – thereby not displaying this even-odd pattern.

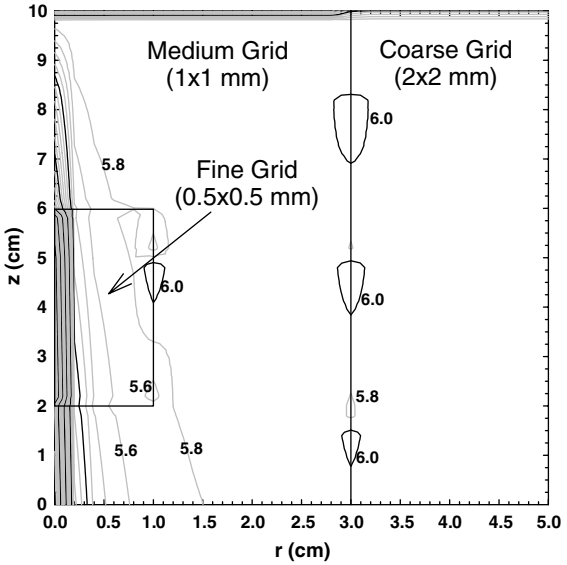

Fig. 5. A contour plot of the density of positive charges using the standard weighting method. Superimposed are the grid sizes used in this simulation. The numbers indicate multiples of 10^{15} m^{-3}.

5. Corrected multiple-grid weighting in 2D cylindrical coordinates (r–z)

As depicted in Fig. 7, we divided the large B′ cell in Fig. 4 into two half-cells: B″ and F. The node (j,k) in Fig. 7 is at the corner of two cells A and C, and two half-cells B″ and D″. Thus,

$$n_{j,k} = \frac{N_{\mathrm{A}} + N''_{\mathrm{B}} + N_{\mathrm{C}} + N''_{\mathrm{D}}}{a + b'' + c + d''} \tag{9a}$$

The particle contributions from the cells A and C are described in Eqs. 4(a–d) and the particle contributions from the two half-cells, B″ and D″ are

$$N''_{\mathrm{B}} = \int_{z_k}^{z_{k+1}} \int_{r_j}^{r_{j+2}} n_p(r,z) \frac{(r_{j+2}^2 - r^2) \times (z_{k+1} - z)}{(r_{j+2}^2 - r_j^2) \times (z_{k+1} - z_k)} \,\mathrm{d}r\,\mathrm{d}z \tag{9b}$$

$$N''_{\mathrm{D}} = \int_{z_{k-1}}^{z_k} \int_{r_j}^{r_{j+2}} n_p(r,z) \frac{(r_{j+2}^2 - r^2) \times (z - z_{k-1})}{(r_{j+2}^2 - r_j^2) \times (z_k - z_{k-1})} \,\mathrm{d}r\,\mathrm{d}z \tag{9c}$$

$$b'' = \int_{z_k}^{z_{k+1}} \int_{r_j}^{r_{j+2}} 2\pi r \frac{(r_{j+2}^2 - r^2) \times (z_{k+1} - z)}{(r_{j+2}^2 - r_j^2) \times (z_{k+1} - z_k)} \,\mathrm{d}r\,\mathrm{d}z \tag{9d}$$

$$d'' = \int_{z_{k-1}}^{z_k} \int_{r_j}^{r_{j+2}} 2\pi r \frac{(r_{j+2}^2 - r^2) \times (z - z_{k-1})}{(r_{j+2}^2 - r_j^2) \times (z_k - z_{k-1})} \,\mathrm{d}r\,\mathrm{d}z \tag{9e}$$

Just as the node (j,k) is at the corner of two cells and two half-cells, the node $(j,k+1)$ is also at the corner of two cells A and E, and two half-cells B″ and F

$$n_{j,k+1} = \frac{N_{\mathrm{A}} + N''_{\mathrm{B}} + N_{\mathrm{E}} + N_{\mathrm{F}}}{a + b'' + e + f} \tag{10a}$$

$$N_{\mathrm{F}} = \int_{z_{k+1}}^{z_{k+2}} \int_{r_j}^{r_{j+2}} n_p(r,z) \frac{(r_{j+2}^2 - r^2) \times (z_{k+2} - z)}{(r_{j+2}^2 - r_j^2) \times (z_{k+2} - z_{k+1})} \,\mathrm{d}r\,\mathrm{d}z \tag{10b}$$

$$f = \int_{z_{k+1}}^{z_{k+2}} \int_{r_j}^{r_{j+2}} 2\pi r \frac{(r_{j+2}^2 - r^2) \times (z_{k+2} - z)}{(r_{j+2}^2 - r_j^2) \times (z_{k+2} - z_{k+1})} \,\mathrm{d}r\,\mathrm{d}z \tag{10c}$$

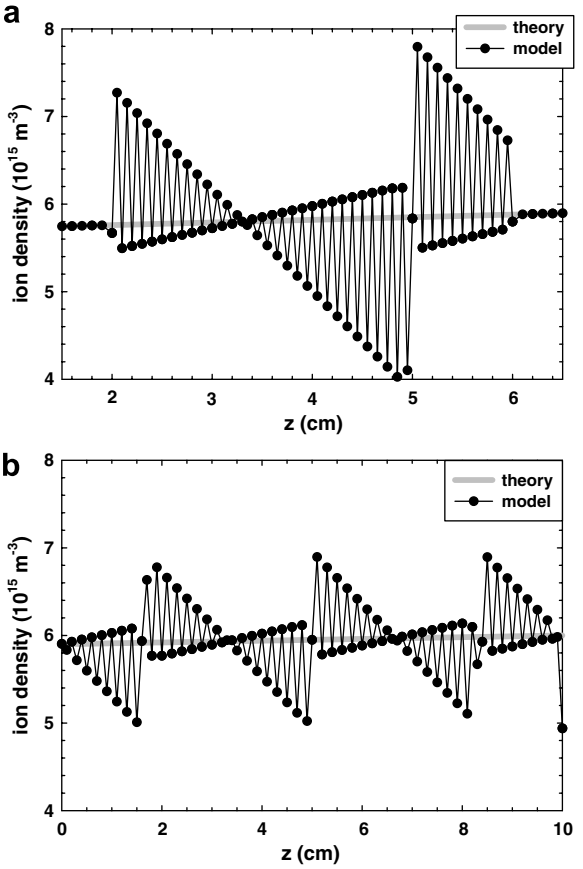

Fig. 6. A one-dimensional profile of the density as a function of z along the interface nodes at (a) $r = 1$ cm and (b) $r = 3$ cm. The thicker grey line is the theoretical value obtained by Eq. (7).

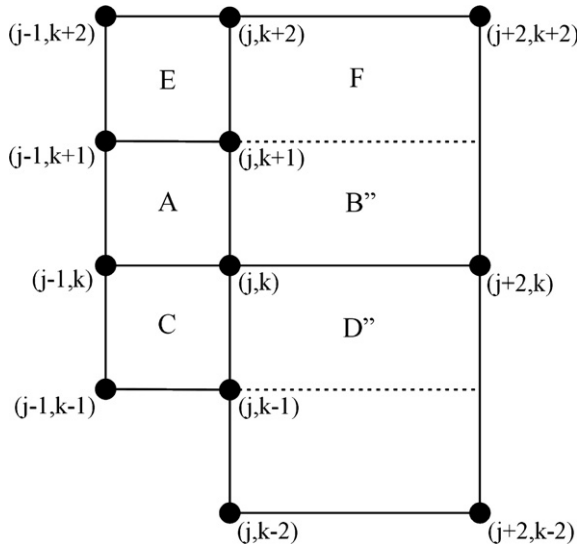

Fig. 7. The cells on the coarse side of the interface are divided into half-cells, B″, D″ and F.

A particle in the F half-cell, for instance, is weighted to the nodes on the left, $(j,k+2)$ and $(j,k+1)$, by the half-cell weighting method described above. The nodes to the right do not see the half-cells, they see only the whole B′ cell in Fig. 4. Consequently, the nodes that this particle is weighted to are the nodes that form the right-hand corners of the B′ cell, the nodes $(j+2,k+2)$ and $(j+2,k)$. Therefore, each particle is weighted to exactly four nodes.

This corrected method was tested in the simple expanding-plasma simulation. All plasma parameters were kept the same as the test described in the previous section, and the grids were arranged as per Fig. 5. The results in Fig. 8 show a remarkable improvement in the new particle to grid weighting method.

6. Electric field calculation and interpolation in multiple-grid 2D cylindrical coordinates (r–z)

The Poisson equation was solved within each grid using the standard five-point relaxation method with electrons following the Boltzmann distribution [2,15]. This involves obtaining the discrete version for the Laplace operator in axisymmetric cylindrical coordinates

$$\nabla^2\Phi = \frac{\partial^2\Phi}{\partial z^2} + \frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial\Phi}{\partial r}\right) = \frac{\partial^2\Phi}{\partial z^2} + \frac{\partial^2\Phi}{\partial r^2} + \frac{1}{r}\frac{\partial\Phi}{\partial r}$$
$$\rightarrow \frac{1}{\mathrm{d}z^2}\left(\Phi_{j,k+1} - 2\Phi_{j,k} + \Phi_{j,k-1}\right) + \frac{1}{\mathrm{d}r^2}\left(\Phi_{j+1,k} - 2\Phi_{j,k} + \Phi_{j-1,k}\right) + \frac{1}{r}\frac{1}{2\mathrm{d}r}\left(\Phi_{j+1,k} - \Phi_{j-1,k}\right) + \mathrm{O}(h^2) \quad (11)$$

This has a second-order truncation error. At the interfaces, the Poisson equation was also solved using a five-point method, but on an irregular grid. In this case, the discrete Laplace operator becomes

$$\nabla^2\Phi \rightarrow \frac{2}{\mathrm{d}z^+\,\mathrm{d}z^-}\left(\frac{\mathrm{d}z^-\Phi_{j,k+1} + \mathrm{d}z^+\Phi_{j,k+1}}{\mathrm{d}z^+ + \mathrm{d}z^-} - \Phi_{j,k}\right) + \frac{2}{\mathrm{d}r^+\,\mathrm{d}r^-}\left(\frac{\mathrm{d}r^-\Phi_{j,k+1} + \mathrm{d}r^+\Phi_{j,k+1}}{\mathrm{d}r^+ + \mathrm{d}r^-} - \Phi_{j,k}\right)$$
$$+ \frac{1}{r}\frac{1}{\mathrm{d}r^+ + \mathrm{d}r^-}\left(\Phi_{j+1,k} - \Phi_{j-1,k}\right) + \mathrm{O}(h) \quad (12)$$

As previously mentioned, there are two types of interface nodes. The stencil used for the two types of nodes is depicted in Fig. 9. For the nodes that do not have a neighbor in the coarse grid, a virtual node is created. The potential of the virtual node is linearly interpolated from the two nodes it neighbors. These five-point relaxation schemes are iterated over all nodes in the entire simulation region until convergence is reached.

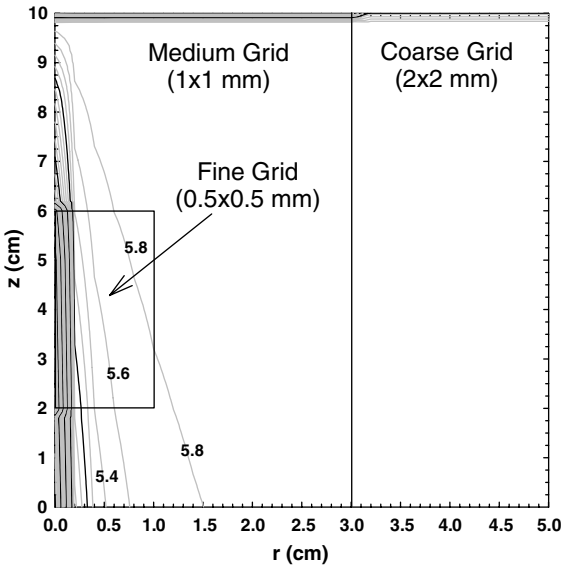

Fig. 8. A contour plot of the density of positive charges using the improved multiple-grid method. Superimposed are the grid sizes used in this simulation. The numbers indicate multiples of 10^{15} m^{-3}.

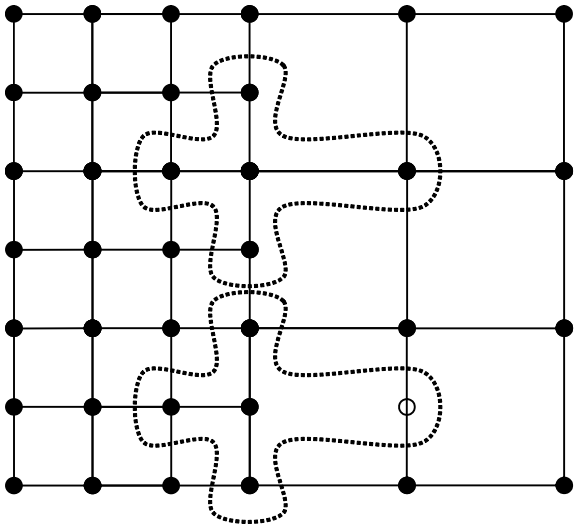

Fig. 9. The stencils used for solving the Poisson equation at the interface. The open circle is the virtual node.

The electric field is found by performing discrete partial differentiation in each direction. In a uniform grid, the electric field is interpolated back to the particle by the same weighting method used to weight the particles to the grid [3]. We correspondingly do the same with the corrected method. The electric field contribution from the nodes to the left is weighted by the half-cell weighting method, i.e. in proportion to the F cell in Fig. 7; and the electric field contribution from the nodes to the right is weighted by the whole-cell weighting method, i.e. in proportion to the B′ cell in Fig. 4.

As the particle is weighted to four nodes, the electric field on the particle is interpolated back from the same four nodes,

$$E_p = E'_{j,k+2} + E'_{j,k+1} + E'_{j+2,k+2} + E'_{j+2,k} \tag{13a}$$

where the contributions from the four nodes are

$$E'_{j,k+2} = E_{j,k+2}\frac{(r^2_{j+2} - r^2_p) \times (z_p - z_{k+1})}{(r^2_{j+2} - r^2_j) \times (z_{k+2} - z_{k+1})} \tag{13b}$$

$$E'_{j,k+1} = E_{j,k+1}\frac{(r^2_{j+2} - r^2_p) \times (z_{k+2} - z_p)}{(r^2_{j+2} - r^2_j) \times (z_{k+2} - z_{k+1})} \tag{13c}$$

$$E'_{j+2,k+2} = E_{j+2,k+2}\frac{(r^2_p - r^2_j) \times (z_p - z_k)}{(r^2_{j+2} - r^2_j) \times (z_{k+2} - z_k)} \tag{13d}$$

$$E'_{j+2,k} = E_{j+2,k}\frac{(r^2_{j+2} - r^2_p) \times (z_{k+2} - z_p)}{(r^2_{j+2} - r^2_j) \times (z_{k+2} - z_k)} \tag{13e}$$

7. Self-force analysis

The self-force of the new method presented here can be analyzed by deriving the total momentum of the demonstrating system [3]. The force on a PIC macro-particle can be written as

$$F_{r,z} = n_p(r,z)E_p \tag{14}$$

where $n_p(r,z)$ is the particle's charge and E_p is the electric field strength interpolating from the adjacent nodes. The total change of momentum can be calculated as

$$\frac{\mathrm{d}P}{\mathrm{d}t} = \int_V F_{r,z}\,\mathrm{d}V \tag{15}$$

where V is the volume of the simulation region. In our proposed scheme, the PIC particles on either side of the interface have different weighting schemes. The particles in the coarser cells weight in proportion to the whole cell, whereas the particles in the finer cell follow the half-cell method. These must be treated separately

$$\frac{\mathrm{d}P}{\mathrm{d}t} = \int_V^{\substack{\text{finer}\\ \text{cells}}} n_p(r,z)E_p\,\mathrm{d}V + \int_V^{\substack{\text{coarser}\\ \text{cells}}} n_p(r,z)E_p\,\mathrm{d}V \tag{16}$$

According to our scheme, the electric field at the nodes is interpolated back to the particles in two different ways

$$\frac{\mathrm{d}P}{\mathrm{d}t} = \int_V^{\substack{\text{finer}\\ \text{cells}}} n_p(r,z)\sum_{j,k}^{\text{adjacent}} E'_{j,k}\,\mathrm{d}V + \int_V^{\substack{\text{coarser}\\ \text{cells}}} n_p(r,z)\sum_{j,k}^{\text{adjacent}} E'_{j,k}\,\mathrm{d}V \tag{17}$$

Particles in the finer cells are treated similarly to the particles in a uniform grid. Thus, for example a particle in the B cell in Fig. 3, the interpolated electric fields can be written as

$$E'_{j,k} = E_{j,k}\frac{(r_{j+1}^2 - r_p^2)\times(z_{k+1}-z_p)}{(r_{j+1}^2 - r_j^2)\times(z_{k+1}-z_k)} \tag{18a}$$

$$E'_{j,k+1} = E_{j,k+1}\frac{(r_{j+1}^2 - r_p^2)\times(z_p-z_k)}{(r_{j+1}^2 - r_j^2)\times(z_{k+1}-z_k)} \tag{18b}$$

$$E'_{j+1,k} = E_{j+1,k}\frac{(r_p^2 - r_j^2)\times(z_{k+1}-z_p)}{(r_{j+1}^2 - r_j^2)\times(z_{k+1}-z_k)} \tag{18c}$$

$$E'_{j+1,k+1} = E_{j+1,k+1}\frac{(r_p^2 - r_j^2)\times(z_p-z_k)}{(r_{j+1}^2 - r_j^2)\times(z_{k+1}-z_k)} \tag{18d}$$

where $E_{j,k}, E_{j,k+1}, E_{j+1,k}, E_{j+1,k+1}$ are the electric fields of the four adjacent nodes of the cell. For a particle in a coarser cell, for example in the F half-cell of Fig. 7, the interpolated electric fields can be written as

$$E'_{j,k+2} = E_{j,k+2}\frac{(r_{j+2}^2 - r_p^2)\times(z_p-z_{k+1})}{(r_{j+2}^2 - r_j^2)\times(z_{k+2}-z_{k+1})} \tag{19a}$$

$$E'_{j,k+1} = E_{j,k+1}\frac{(r_{j+2}^2 - r_p^2)\times(z_{k+2}-z_p)}{(r_{j+2}^2 - r_j^2)\times(z_{k+2}-z_{k+1})} \tag{19b}$$

$$E'_{j+2,k+2} = E_{j+2,k+2}\frac{(r_p^2 - r_j^2)\times(z_p-z_k)}{(r_{j+2}^2 - r_j^2)\times(z_{k+2}-z_k)} \tag{19c}$$

$$E'_{j+2,k} = E_{j+2,k}\frac{(r_{j+2}^2 - r_p^2)\times(z_{k+2}-z_p)}{(r_{j+2}^2 - r_j^2)\times(z_{k+2}-z_k)} \tag{19d}$$

where $E_{j,k+2}$, $E_{j,k+1}$, $E_{j+2,k+2}$ and $E_{j+2,k}$ are the electric fields of the four adjacent nodes, according to the half cell weighting scheme, of the boundary cell as depicted in Fig. 7.

By changing the order of the sum and integral in (17), we have

$$\frac{\mathrm{d}P}{\mathrm{d}t} = \sum_{j,k}^{\substack{\text{regular}\\ \text{nodes}}} E_{j,k}(N_\mathrm{A}+N_\mathrm{B}+N_\mathrm{C}+N_\mathrm{D}) + \sum_{j,k}^{\substack{\text{interface}\\ \text{nodes}}} E_{j,k}(N_\mathrm{A}+N_{\mathrm{B}''}+N_\mathrm{C}+N_{\mathrm{D}''}) \tag{20}$$

For a regular node (j,k), as depicted in Fig. 3,

$$N_{\mathrm{A}} = \int_{z_k}^{z_{k+1}} \int_{r_{j-1}}^{r_j} n_p(r,z) \frac{(r^2 - r_{j-1}^2) \times (z_{k+1} - z)}{(r_j^2 - r_{j-1}^2) \times (z_{k+1} - z_k)} \,\mathrm{d}r\,\mathrm{d}z \tag{21a}$$

$$N_{\mathrm{B}} = \int_{z_k}^{z_{k+1}} \int_{r_j}^{r_{j+1}} n_p(r,z) \frac{(r_{j+1}^2 - r^2) \times (z_{k+1} - z)}{(r_{j+1}^2 - r_j^2) \times (z_{k+1} - z_k)} \,\mathrm{d}r\,\mathrm{d}z \tag{21b}$$

$$N_{\mathrm{C}} = \int_{z_{k-1}}^{z_k} \int_{r_{j-1}}^{r_j} n_p(r,z) \frac{(r^2 - r_{j-1}^2) \times (z - z_{k-1})}{(r_j^2 - r_{j-1}^2) \times (z_k - z_{k-1})} \,\mathrm{d}r\,\mathrm{d}z \tag{21c}$$

$$N_{\mathrm{D}} = \int_{z_{k-1}}^{z_k} \int_{r_j}^{r_{j+1}} n_p(r,z) \frac{(r_{j+1}^2 - r^2) \times (z - z_{k-1})}{(r_{j+1}^2 - r_j^2) \times (z_k - z_{k-1})} \,\mathrm{d}r\,\mathrm{d}z \tag{21d}$$

For an interface node (j,k), as depicted in Fig. 7,

$$N_{\mathrm{A}} = \int_{z_k}^{z_{k+1}} \int_{r_{j-1}}^{r_j} n_p(r,z) \frac{(r^2 - r_{j-1}^2) \times (z_{k+1} - z)}{(r_j^2 - r_{j-1}^2) \times (z_{k+1} - z_k)} \,\mathrm{d}r\,\mathrm{d}z \tag{22a}$$

$$N_{\mathrm{B}''} = \int_{z_k}^{z_{k+1}} \int_{r_j}^{r_{j+2}} n_p(r,z) \frac{(r_{j+2}^2 - r^2) \times (z_{k+1} - z)}{(r_{j+2}^2 - r_j^2) \times (z_{k+1} - z_k)} \,\mathrm{d}r\,\mathrm{d}z \tag{22b}$$

$$N_{\mathrm{C}} = \int_{z_{k-1}}^{z_k} \int_{r_{j-1}}^{r_j} n_p(r,z) \frac{(r^2 - r_{j-1}^2) \times (z - z_{k-1})}{(r_j^2 - r_{j-1}^2) \times (z_k - z_{k-1})} \,\mathrm{d}r\,\mathrm{d}z \tag{22c}$$

$$N_{\mathrm{D}''} = \int_{z_{k-1}}^{z_k} \int_{r_j}^{r_{j+2}} n_p(r,z) \frac{(r_{j+2}^2 - r^2) \times (z - z_{k-1})}{(r_{j+2}^2 - r_j^2) \times (z_k - z_{k-1})} \,\mathrm{d}r\,\mathrm{d}z \tag{22d}$$

Inserting the effective volume of individual nodes, we have

$$\frac{\mathrm{d}P}{\mathrm{d}t} = \sum_{j,k}^{\substack{\text{regular}\\ \text{nodes}}} V_{j,k} E_{j,k} n_{j,k} + \sum_{j,k}^{\substack{\text{interface}\\ \text{nodes}}} V_{j,k} E_{j,k} n_{j,k} \tag{23}$$

where $n_{j,k}$ is the charge density at the node (j,k). For a regular node as depicted in Fig. 3,

$$V_{j,k} = \frac{\pi}{4}(r_{j+1}^2 - r_{j-1}^2)(z_{k+1} - z_{k-1}) \tag{24}$$

For a boundary node as depicted in Fig. 7

$$V_{j,k} = \frac{\pi}{4}(r_{j+2}^2 - r_{j-1}^2)(z_{k+1} - z_{k-1}) \tag{25}$$

Therefore, the expression for the change of momentum (Eq. (20)) has no weighting function present. The self-force as described in Ref. [3] is eliminated as the question of momentum change is reduced to the properties of the calculation of the electric field from the charge density. In the following section, we discuss other sources of spurious self-forces, including the effect of the electric field calculation, by looking at a system with a single PIC particle.

8. Spurious self-forces in a single PIC particle system

A spurious self-force was observed in mesh refinement for particle-in-cell plasma simulations [16]. To investigate the presence of this spurious force, we have performed tests similar to Ref. [16], by putting a macroparticle into a rectangular grid (mother) with a fine grid in rectangular coordinates. The mother grid had a cell size of $\mathrm{d}x = \mathrm{d}y = 1$ mm. It consisted of 64 cells along the x-axis between $x = 0.0$ and $x = 0.064$ m, and 64 cells along the y-axis between $y = 0.0$ and $y = 0.064$ m. A fine grid was framed in the center of the mother gird with cell size of $\mathrm{d}x = \mathrm{d}y = 0.5$ mm. It consisted of 64 cells along the x-axis between $x = 0.016$ and $x = 0.048$ m, and 64 cells along the y-axis between $y = 0.016$ and $y = 0.048$ m. The macroparticle was initially placed at $x = 0.01775$ and $y = 0.032$ m, which is 4 fine cells away from the coarse-fine interface in the x-direction. An

argon macroparticle of atomic mass 40 amu and charge state of +1 was put into the test. The macroparticle carried a total charge of 2.5×10^9 e, and the time step used was 1 ns. The mother grid was surrounded by four grounded walls, i.e. the potential of the nodes at the edge of the simulation region was set to zero. At the interface between the coarse and fine grids ($x = 16$ mm) we use the three-point method of calculating the electric field

$$E_x(x = 16\text{ mm}) = -\frac{4\phi(x = 16.5\text{ mm}) - 3\phi(x = 16\text{ mm}) - \phi(x = 15\text{ mm})}{6h} + \mathrm{O}(h^2) \tag{26}$$

where $h = 0.5$ mm.

When the macroparticle reached the edge of the simulation region, it is reflected back into the simulation region. Two tests were also performed in a single grid system with cell size of $0.5 \times 0.5\text{ mm}^2$ and $1.0 \times 1.0\text{ mm}^2$. The finer single grid consisted of 128×128 cells and the coarse grid consisted of 64×64 cells.

The trajectories of the macroparticle are depicted in Fig. 10. Using the multiple-grid method and the three-point formula (Eq. (26)) for the electric field, the macroparticle experienced a spurious self-force and was pushed back into the fine region when it approached the interface. We can also calculate the electric field at the interface by looking only at the two points one coarse cell size away in both directions, i.e.

$$E_x(x = 16\text{ mm}) = -\frac{\phi(x = 17\text{ mm}) - \phi(x = 15\text{ mm})}{4h} + \mathrm{O}(h^2) \tag{27}$$

This also led to a spurious self-force that pushed the particle back into the fine grid, this has also been depicted in Fig. 10. However, when the electric field along the x-direction at the interface node ($x = 16$ mm, $y = 32$ mm) is changed to a first-order accurate two-point finite difference equation

$$E_x(x = 16\text{ mm}) = -\frac{\phi(x = 16.5\text{ mm}) - \phi(x = 15\text{ mm})}{3\,h} + \mathrm{O}(h) \tag{28}$$

the macroparticle did not bounce at the boundary as depicted in Fig. 10. Although Eq. (28) is only first-order accurate, the global effects of this are small and localized to the interface. When the particle is in the coarse grid, we use a second-order formula with truncation error proportional to the square of the coarser grid size, and only at the two cells adjacent to the interface do we use the first-order formula with truncation error proportional to the fine grid size. As interface is only a very small part of the simulation region, then particle

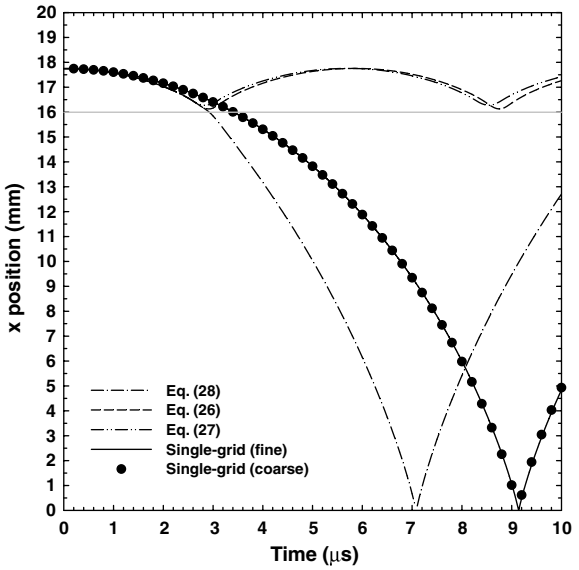

Fig. 10. The trajectory of a macroparticle as it falls towards the wall at $x = 0$ using various methods of calculating the electric field described in the text (Eqs. (26)–(28)). Also pictured are the reference cases, a single coarse (1 mm × 1 mm) grid and a single fine (0.5 mm × 0.5 mm) grid.

spends very little time in this region, so overall this single PIC particle system is roughly second-order accurate. Furthermore, in a single PIC particle system, the potential has a very large gradient, which amplifies anomalies obtained from electric field calculations. In Fig. 11 we have positioned a single PIC particle at x-positions between 14 and 17 mm, and calculated the electric field at the nodes using three different methods described in Eqs. (26)–(28). We see in Fig. 11 that using a second-order formula does not alleviate this problem. We will show in the following paragraph that using a higher-order equation to calculate the electric field does not mitigate the problem even when using a single grid.

To illustrate the effect of the choice of electric field formula, a single fine grid system of cell size 0.5×0.5 mm^2 was considered. When the macroparticle of total charge 2.5×10^9 e was placed at $x = 17.875$ mm and $y = 32.0$ mm, an ion density of 2.5×10^{15} m^{-3} was assigned to the node on its left ($x = 17.5$ mm, $y = 32.0$ mm) and an ion density of 7.5×10^{15} m^{-3} was assigned to the node on its right ($x = 18.0$ mm, $y = 32.0$ mm). The potential at the four nodes at $x = 17.0$ mm, 17.5 mm, 18.0 mm and 18.5 mm ($y = 32.0$ mm) were calculated by solving the Poisson equation, and are plotted in Fig. 12(a). The electric field, E_x, at the nodes $x = 17.5$ mm and $x = 18.0$ mm can be calculated by a two-point finite difference equation

$$E_x(x = 17.5\text{ mm}) = -\frac{\phi(x = 18.0\text{ mm}) - \phi(x = 17.0\text{ mm})}{2h} + \mathrm{O}(h^2) \tag{29}$$

$$E_x(x = 18\text{ mm}) = -\frac{\phi(x = 18.5\text{ mm}) - \phi(x = 17.5\text{ mm})}{2h} + \mathrm{O}(h^2) \tag{30}$$

An alternative is to use a four-point finite difference equation, see Appendix 2.

$$E_x(x = 17.5\text{ mm}) = -\frac{-\phi(x = 18.5\text{ mm}) + 6\phi(x = 18.0\text{ mm}) - 3\phi(x = 17.5\text{ mm}) - 2\phi(x = 17.0\text{ mm})}{6h} + \mathrm{O}(h^3) \tag{31}$$

$$E_x(x = 18\text{ mm}) = -\frac{2\phi(x = 18.5\text{ mm}) + 3\phi(x = 18.0\text{ mm}) - 6\phi(x = 17.5\text{ mm}) + \phi(x = 17.0\text{ mm})}{6h} + \mathrm{O}(h^3) \tag{32}$$

As seen in Fig. 12, the four-point method generated an electric field strength 4799 V/m less than the two-point method at the same two nodes. The macroparticle received a linearly interpolated electric field strength of -118.497 V/m from the two-point method but a considerably greater electric field of -4918.38 V/m from the four-point method. When the macroparticle was placed exactly between the two nodes ($x = 17.75$ mm), both the nodes receive an ion density of 5×10^{15} m^{-3} from the macroparticle, and both methods gave the same electric field strength at the nodes (Fig. 12(b)). The interpolated electric field strength at the particle was -120.27 V/m. When the macroparticle was placed further downwards at $x = 17.625$ mm, a large difference in electric field strength between the two methods was again observed (Fig. 12(c)). An ion density of

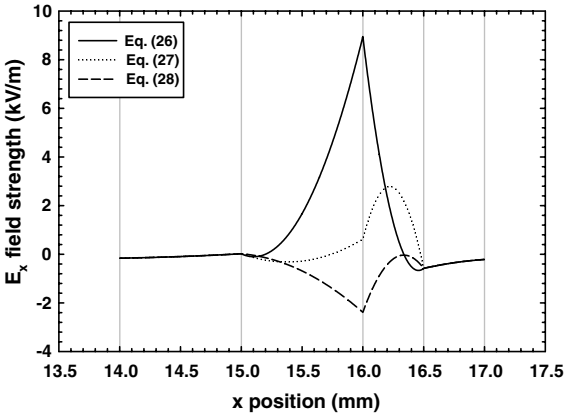

Fig. 11. The electric field gathered at a single PIC particle in a multiple-grid system as a function of its position. Three different methods of electric field calculation (Eqs. (26)–(28)) are compared. The grey vertical lines represent the positions of the nodes.

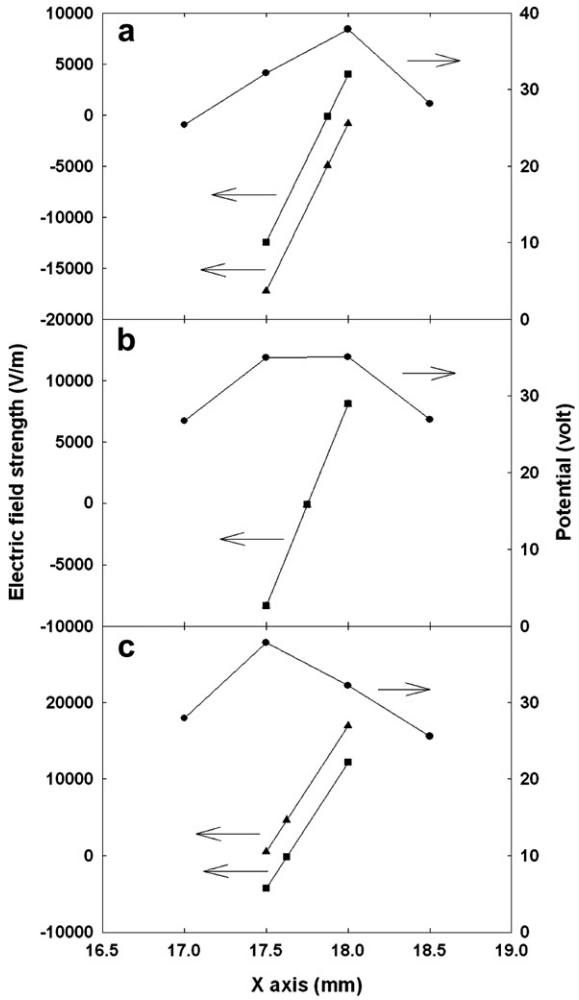

Fig. 12. The node potential (circle), electric field strengths as calculated by the two-point (square) and four-point methods (triangle) at different location (a) $x = 17.875$ mm, (b) $x = 17.75$ mm and (c) $x = 17.625$ mm. The electric field strength interpolated at macroparticle position by the two methods is also shown.

7.5×10^{15} m^{-3} was assigned to the left node ($x = 17.5$ mm, $y = 32$ mm) and an ion density of 2.5×10^{15} m^{-3} was assigned to the right node ($x = 18.0$ mm, $y = 32$ mm). The macroparticle received an interpolated electric field of -122.03 V/m from the two-point method but astonishingly received a positive electric field strength of 4677.88 V/m from the four-point method. Letting go of a macroparticle initially placed at $x = 17.875$ mm and $y = 32$ mm with $V_x = V_y = 0.0$, the electric field strength calculated by a two-point method will gradually accelerate the macroparticle towards the negative x-direction, as it should. However, the electric field strength calculated by a more accurate, in principle, four-point method may trap the macroparticle within the cell region between $x = 17.5$ mm and $x = 18.0$ mm.

In the previous section, the change of momentum has been shown to not be dependent on the weighting function, so this spurious force is due to the choice of electric field calculation. In a single PIC particle system, the potential has a large gradient on each side of the particle, which leads to a systematic anomaly in the calculated electric field. In a system where the plasma is represented by many wide-spread PIC particles, the charge is more evenly spread out by the particles so the calculated potential is much smoother, which dampens the anomaly obtained when using Eq. (26) at a coarse-fine interface. This might not readily apply to a quasi-neutral/neutral plasma where long range forces are screened and the importance of local fields, and thus any error done at this range, is larger.

9. MePIIID on a flat stage

To rigorously test the multiple-grid method proposed above, it was implemented in a study of metal plasma-immersion ion implantation and deposition (MePIIID) on a flat stage [9,10]. In a simulation chamber 10 cm in radius and 30 cm in height, we constructed a flat stage of radius 3.5 cm and thickness 1.2 cm, which was held in place by a rod of length 12.0 cm and radius 0.4 cm. A schematic diagram of the simulation is shown in Fig. 13. It has been shown that the density of a typical titanium plasma produced by cathodic arc with a background gas pressure of $\sim 10^{-6}$ Torr is 6×10^{15} m^{-3}, has a drift velocity of 1.3×10^4 ms^{-1}, and an average charge state of 2.03 [9,12]. These plasma characteristics were used in the simulations. The ions were assumed to be collisionless and electrons followed the Boltzmann relation:

$$n_e = n_0 \exp\left(\frac{e\phi}{kT_e}\right) \tag{33}$$

where the electron temperature was $T_e = 2.5$ eV. The Boltzmann relation is linearized and the Poisson equation is solved by relaxation [15]. The grid spacing was 2 mm × 2 mm and was refined to 1 mm × 1 mm in the region between the stage and the ceiling of the simulated region. A further refined patch, 0.5 mm × 0.5 mm, was inserted in a small region where r is less than 4 mm; and z lies between 14.5 cm and 22 cm. The electric field at the boundary was calculated by Eq. (26). The timestep was 0.85 ns. The PIC particles were uniformly spaced with $9 \times 9 = 81$ particles inserted into the finest cells and $36 \times 36 = 1296$ particles into the coarsest cells (see Fig. 14). The stage and rod were put at a bias potential of −8 kV and the simulation was run until a steady-state was reached. The results from the simulation (Fig. 15) were compared to a control simulation,

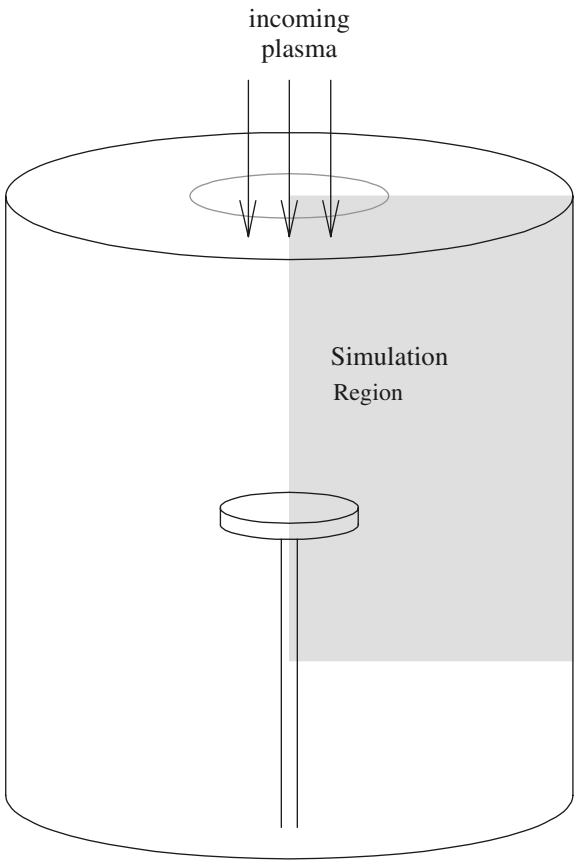

Fig. 13. A schematic diagram of the simulation used to test the new method. The plasma from a cathodic arc source drifts downwards from a duct. The stage and rod are set to a −8 kV bias, and the outer wall is earthed.

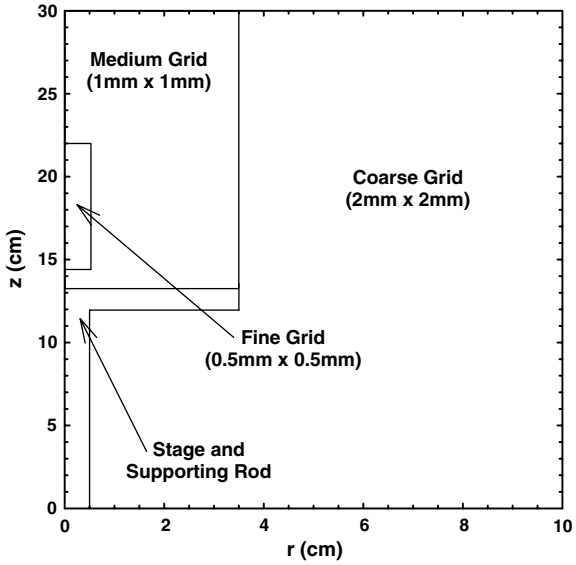

Fig. 14. A schematic diagram of the three grids used in the MePIIID simulation.

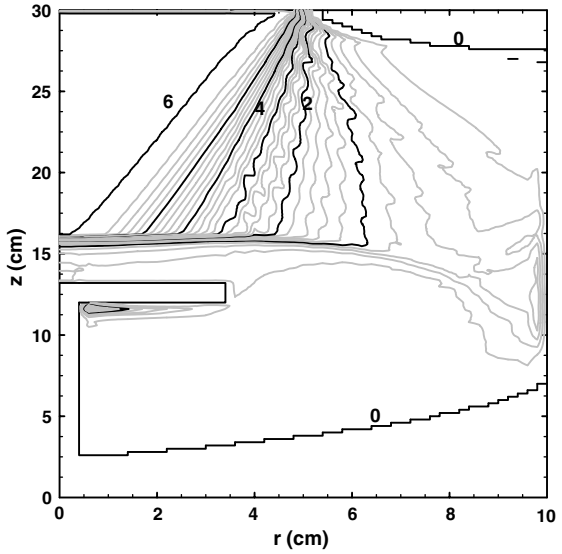

Fig. 15. The results obtained from a multiple-grid simulation. The numbers on the contour lines represent multiples of 10^{15} m^{-3}.

run with only the coarsest grid (Fig. 16). Both the sheath shape and position are similar in the multiple-grid and control runs. The 6×10^{15} m^{-3} and 0 m^{-3} contours did differ slightly, but these differences in density are less than 1%. The main feature is the equilibrium steady state ion sheath shape and position [9], which remains at $z = 16$ cm in both cases (Figs. 15 and 16).

As the Poisson equation solver along the interface nodes has a first-order truncation error, it is important to ensure that this loss in accuracy is kept small. The leading term in the truncation error for the Laplace operator nodes at the interface is

$$\frac{\mathrm{d}z^+ - \mathrm{d}z^-}{3}\frac{\partial^3 \Phi}{\partial z^3} + \frac{\mathrm{d}r^+ - \mathrm{d}r^-}{3}\frac{\partial^3 \Phi}{\partial r^3} - \frac{\mathrm{d}r^+ - \mathrm{d}r^-}{2}\frac{1}{r}\frac{\partial^2 \Phi}{\partial r^2} \tag{34}$$

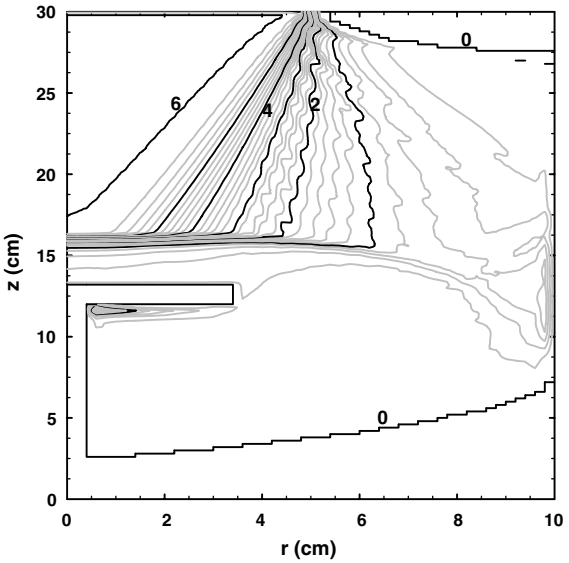

Fig. 16. The results obtained from a single grid (2 mm × 2 mm) simulation. The numbers on the contour lines represent multiples of 10^{15} m^{-3}.

This is first-order in the change of grid size, thus for a grid refinement factor of two, as was used in this simulation, this is first-order in the finer grid spacing. The adjoining nodes in the coarse grid are have a second-order truncation error, the leading term is

$$\frac{\mathrm{d}z^2}{12}\frac{\partial^4\Phi}{\partial z^4}+\frac{\mathrm{d}r^2}{12}\frac{\partial^4\Phi}{\partial r^4}+\frac{\mathrm{d}r^2}{6}\frac{1}{r}\frac{\partial^3\Phi}{\partial r^3} \tag{35}$$

however, this is second-order in the larger grid spacing. Upon substitution of numbers obtained from the simulation, we observed that the first-order truncation error term in the interface nodes (Eq. (34)) was comparable to the second-order truncation error term in the coarse nodes neighboring the interface (Eq. (35)).

10. Conclusion

An algorithm for weighting particles to nodes near interfaces between coarser and finer grids in a multiple-grid PIC system is described in two-dimensional (r–z) cylindrical coordinates. The method divides the cells on the coarser side of the interface into half-cells, which are used in the weighting of particles to nodes. The use of half-cells is more natural than weighting to the cell corners as it treats the nodes along the interface equally. The method assumes that the grid spacing of the coarser grid is double that of the finer grid, a gradual transition of cell size is required for a good model as the leading term of the truncation error at the interface scales as the change in grid size. The volume weighting method was used in this study as it is similar to the standard bilinear PIC method in Cartesian coordinates, but with areas rotated about the z-axis. As the volume weighting method has the axisymmetric geometry built in, it too is a natural system in this particular case. We have demonstrated a spurious self-force for the case when the potential gradient is large near the seam between the coarse and fine grids. We believe the origin of the self-force lies in the discontinuity in resolution introduced by the refinement, which affects both the method presented in Ref. [8] and the one presented in this paper. The algorithm was tested in MePIIID of a flat stage. The mesh consisted of three cell sizes of 0.5×0.5, 1×1 and 2×2 mm. The simulated equilibrium steady state ion sheath was equal to the result obtained by single grid of cell size 2×2 mm. We have used this method to simulate the problem of MePIIID on a cone and tip, which brought a measurable gain in accuracy by spatially resolving the sharp tip without significantly increasing the computational cost of the simulation. The simulation results have been published in IEEE Transactions on Plasma Science [17].

Acknowledgments

The authors would like to acknowledge the reviewers and editors of this paper for their detailed revisions and useful comments made during the revision process. Dixon T.K. Kwok is supported by the Denison Fellowship from the School of Physics, University of Sydney, Australia.

Appendix 1

For a horizontal interface with a coarse grid above and a fine grid below, as pictured in Fig. A.1. Eqs. (9) become

$$n_{j,k} = \frac{N_A + N''_B + N_C + N''_D}{a + b'' + c + d''} \tag{A1.1a}$$

$$N''_B = \int_{z_k}^{z_{k+2}} \int_{r_j}^{r_{j+1}} n_p(r,z) \frac{(r_{j+1}^2 - r^2) \times (z_{k+2} - z)}{(r_{j+1}^2 - r_j^2) \times (z_{k+2} - z_k)} \, dr\, dz \tag{A1.1b}$$

$$N''_D = \int_{z_k}^{z_{k+2}} \int_{r_{j-1}}^{r_j} n_p(r,z) \frac{(r^2 - r_{j-1}^2) \times (z_{k+2} - z)}{(r_j^2 - r_{j-1}^2) \times (z_{k+2} - z_k)} \, dr\, dz \tag{A1.1c}$$

$$b'' = \int_{z_k}^{z_{k+2}} \int_{r_j}^{r_{j+1}} 2\pi r \frac{(r_{j+1}^2 - r^2) \times (z_{k+2} - z)}{(r_{j+1}^2 - r_j^2) \times (z_{k+2} - z_k)} \, dr\, dz \tag{A1.1d}$$

$$d'' = \int_{z_k}^{z_{k+2}} \int_{r_{j-1}}^{r_j} 2\pi r \frac{(r^2 - r_{j-1}^2) \times (z_{k+2} - z)}{(r_j^2 - r_{j-1}^2) \times (z_{k+2} - z_k)} \, dr\, dz \tag{A1.1e}$$

Just as the node (j,k) is at the corner of two cells and two half-cells, the node $(j+1,k)$ is also at the corner of two cells A and E, and two half-cells B″ and F

$$n_{j,k+1} = \frac{N_A + N''_B + N_E + N_F}{a + b'' + e + f} \tag{A1.2a}$$

$$N_F = \int_{z_k}^{z_{k+2}} \int_{r_{j+1}}^{r_{j+2}} n_p(r,z) \frac{(r_{j+2}^2 - r^2) \times (z_{k+2} - z)}{(r_{j+2}^2 - r_{j+1}^2) \times (z_{k+2} - z_k)} \, dr\, dz \tag{A1.2b}$$

$$f = \int_{z_k}^{z_{k+2}} \int_{r_{j+1}}^{r_{j+2}} 2\pi r \frac{(r_{j+2}^2 - r^2) \times (z_{k+2} - z)}{(r_{j+2}^2 - r_{j+1}^2) \times (z_{k+2} - z_k)} \, dr\, dz \tag{A1.2c}$$

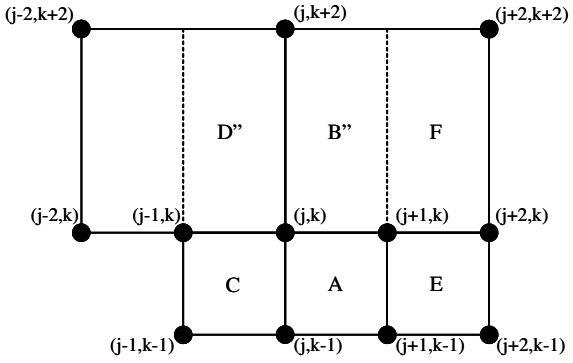

Fig. A.1. A horizontal interface is treated similarly to a vertical one.

Appendix 2

The Taylor series' for the three points $i-2h$, $i-h$ and $i+h$ are

$$\phi_{i-2h} = \phi_i - 2h\frac{\mathrm{d}\phi_i}{\mathrm{d}x} + 2h^2\frac{\mathrm{d}^2\phi_i}{\mathrm{d}x^2} - \frac{4}{3}h^3\frac{\mathrm{d}^3\phi_i}{\mathrm{d}x^3} + \mathrm{O}(h^4) \tag{A2.1}$$

$$\phi_{i-h} = \phi_i - h\frac{\mathrm{d}\phi_i}{\mathrm{d}x} + \frac{h^2}{2}\frac{\mathrm{d}^2\phi_i}{\mathrm{d}x^2} - \frac{h^3}{6}\frac{\mathrm{d}^3\phi_i}{\mathrm{d}x^3} + \mathrm{O}(h^4) \tag{A2.2}$$

$$\phi_{i+h} = \phi_i + h\frac{\mathrm{d}\phi_i}{\mathrm{d}x} + \frac{h^2}{2}\frac{\mathrm{d}^2\phi_i}{\mathrm{d}x^2} + \frac{h^3}{6}\frac{\mathrm{d}^3\phi_i}{\mathrm{d}x^3} + \mathrm{O}(h^4) \tag{A2.3}$$

Adding Eqs. (A2.2) and (A2.3) gives

$$\phi_{i+h} + \phi_{i-h} = 2\phi_i + h^2\frac{\mathrm{d}^2\phi_i}{\mathrm{d}x^2} + \mathrm{O}(h^4) \tag{A2.4}$$

Adding four times Eq. (A2.3) to half of Eq. (A2.1),

$$8\phi_{i+h} + \phi_{i-2h} = 9\phi_i + 6h\frac{\mathrm{d}\phi_i}{\mathrm{d}x} + 6h^2\frac{\mathrm{d}^2\phi_i}{\mathrm{d}x^2} + \mathrm{O}(h^4) \tag{A2.5}$$

Subtracting Eq. (A2.4) from Eq. (A2.5) six times,

$$\frac{2\phi_{i+h} + 3\phi_i - 6\phi_{i-h} + \phi_{i-2h}}{6h} = \frac{\mathrm{d}\phi_i}{\mathrm{d}x} + \mathrm{O}(h^3) \tag{A2.6}$$

For the nodes on the opposite side of the fine patch, we can simply replace h with $-h$,

$$\frac{-\phi_{i+2h} + 6\phi_{i+h} - 3\phi_i - 2\phi_{i-h}}{6h} = \frac{\mathrm{d}\phi_i}{\mathrm{d}x} + \mathrm{O}(h^3) \tag{A2.7}$$

References

[1] C. Cornet, D.T.K. Kwok, M.M.M. Bilek, D.R. McKenzie, Numerical simulation of metal plasma-immersion ion implantation and deposition on a cone, Journal of Applied Physics 96 (11) (2004) 6045–6052.
[2] M.A. Liberman, A.J. Lichtenberg, Principles of Plasma Discharges and Materials Processing, John Wiley & Sons, New York, 1994.
[3] C.K. Birdsall, A.B. Langdon, Plasma Physics via Computer Simulation, McGraw-Hill, New York, 1985 (Section 8-6).
[4] W. Skamarock, J. Oliger, R.L. Street, Adaptive grid refinement for numerical weather prediction, Journal of Computational Physics 80 (1989) 27–60.
[5] A.S. Almgren, J.B. Bell, P. Colella, L.H. Howell, M.L. Welcome, A conservative adaptive projection method for the variable density incompressible Navier–Stokes equations, Journal of Computational Physics 142 (1998) 1–46.
[6] P. McCorquodale, P. Colella, D.P. Grote, J.-L. Vay, A node-centered local refinement algorithm for Poisson's equation in complex geometries, Journal of Computational Physics 201 (2004) 34–60.
[7] W.M. Ruyten, Density-conserving shape factors for particle simulations in cylindrical and spherical coordinates, Journal of Computational Physics 105 (1992) 224–232.
[8] J.-L. Vay, P. Colella, J.W. Kwan, P. McCorquodale, D.B. Serafini, Application of adaptive mesh refinement to particle-in-cell simulations of plasmas and beams, Physics of Plasmas 11 (5) (2004) 2928–2934.
[9] D.T.K. Kwok, T.W.H. Oates, D.R. McKenzie, M.M.M. Bilek, Determination of the equilibrium ion sheath in the drifting plasma by numerical simulation, IEEE Transactions on Plasma Science 31 (5) (2003) 1044–1051.
[10] T.W.H. Oates, J. Piggot, D.R. McKenzie, M.M.M. Bilek, Electric probe measurements of high-voltage sheath collapse in cathodic arc plasmas due to surface charging of insulators, IEEE Transactions on Plasma Science 31 (3) (2003) 438–443.
[11] J. Kutzner, H.C. Miller, Integrated ion flux emitted from the cathode spot region of a diffuse vacuum arc, Journal of Physics D: Applied Physics 25 (1992) 686–693.
[12] E.M. Oks, A. Anders, I.G. Brown, M.R. Dickinson, R.A. MacGill, Ion charge state distributions in high current vacuum arc plasmas in a magnetic field, IEEE Transactions on Plasma Science 24 (3) (1996) 1174–1183.
[13] D.J. Larson, D.W. Hewett, A.B. Langdon, Correction factors for PIC accumulation on radial grids, Computer Physics Communications 90 (1995) 260–266.
[14] J.P. Verboncoeur, Symmetric spline weighting for charge and current density in particle simulation, Journal of Computational Physics 174 (2001) 421–427.
[15] G.A. Emmert, M.A. Henry, Numerical simulation of plasma sheath expansion, with applications to plasma-source ion implantation, Journal of Applied Physics 71 (1) (1992) 113–117.

[16] J.-L. Vay, P. Colella, P. McCorquodale, B.V. Straalen, A. Friedman, D.P. Grote, Mesh refinement for particle-in-cell plasma simulations: applications and benefits for heavy ion fusion, Laser and Particle Beams 20 (2002) 569–575.
[17] D.T.K. Kwok, C. Cornet, Numerical simulation of metal plasma immersion ion implantation (MePIIID) on a sharp cone and a fine tip by multiple grid particle-in-cell (pic) method, IEEE Transactions on Plasma Science 34 (2006) 2434–2442.