SAND2007-6466C

Second-Order, Exact Charge Conservation for Electromagnetic Particle-in-Cell Simulation in Complex Geometry

Timothy D. Pointon

Sandia National Laboratories, Albuquerque, NM 87185

Introduction

Many authors have reported the benefits of using higher-order, exact charge conservation schemes for electromagnetic particle-in-cell (EM-PIC) simulations of laser-plasma interactions. With no complex boundary conditions or internal geometry to deal with, the implementation of higher-order schemes is relatively straightforward.[1] In contrast, most current EM-PIC simulations of high-power devices use the exact, first-order charge-conserving scheme.[2] This is usually adequate to self-consistently model the interaction of high-energy electrons with the electromagnetic field in vacuum. Provided that boundary conditions are handled well, simulations can accurately model the “ideal” behavior of a device, and agree well with experiment early in the power pulse. However, real devices often exhibit degraded performance later in time, attributed to the effects of cold, dense plasmas formed at electrode surfaces.[3] To have true predictive capability of real device behavior, these effects must be modeled. However, the Debye length of these electrode plasmas is of order microns or less. To simulate an entire device with such plasmas requires operating at $\Delta x/\lambda_{De} >> 1$. The first-order scheme simply cannot handle this problem adequately. To address this, we have developed a second-order, exact charge-conservation scheme that handles boundaries, implemented in the 3-D EM-PIC code QUICKSILVER.[4] The algorithm handles complex conductor geometries and standard external boundaries, on non-uniform grids, in either Cartesian, cylindrical or spherical coordinates.

Second-order algorithm in vacuum

We first describe the baseline particle weighting used for vacuum cells, starting with the 1-D case. Fig. 1 illustrates our nomenclature for the standard interleaved grid, with charge density defined at the “full-grid points”, at the center of “half-grid cells”, and current density at the normal faces of the half-grid cells. In Fig. 1, we have a single particle with charge q, moving from the initial location x^0 to the final location x^1, within a single half-grid cell. In everything that follows, if a particle crosses a half-grid cell boundary in the timestep, we break up the particle path into segments confined to a single half-grid cell, and use the algorithm for each segment. The current pulse, $J = q(x^1 - x^0)/(\Delta t \Delta V)$, is scattered to the two half-grid cell boundaries using linear weighting from the centroid of the particle path, $(x^0 + x^1)/2$. Defining the half-grid weight factors,

$$u^n_{x+} = \frac{x^n - X^h_{i-1}}{\Delta X^h_i}; u^n_{x-} = 1 - u^n_{x+}, \tag{1}$$

for time levels n = 0,1. the current density at the cell boundaries are

$$J_{x, i-1} = -\frac{1}{2}((u^1_{x-})^2 - (u^0_{x-})^2), \text{ and } J_{x, i} = \frac{1}{2}((u^1_{x+})^2 - (u^0_{x+})^2). \tag{2}$$

For notational simplicity, we omit factors of q, Δt and cell area in Eq. 2. To satisfy the discretized version of charge conservation on the grid, it is straightforward to show that the particle charge must be allocated to the three cells $i+\alpha$, $\alpha = [-1,0,1]$, using charge weights $w^n_{x\alpha}$:

$$w^n_{x-} = \frac{1}{2}(u^n_{x-})^2,\ w^n_{x+} = \frac{1}{2}(u^n_{x+})^2, \text{ and } w^n_{x0} = 1 - w^n_{x-} - w^n_{x+}. \qquad (3)$$

Having defined the charge weights, it is useful and instructive to rewrite Eq. 2 as

$$J_{x,\, i-1} = -(w^1_{x-} - w^0_{x-}), \text{ and } J_{x,\, i} = w^1_{x+} - w^0_{x+}. \qquad (4)$$

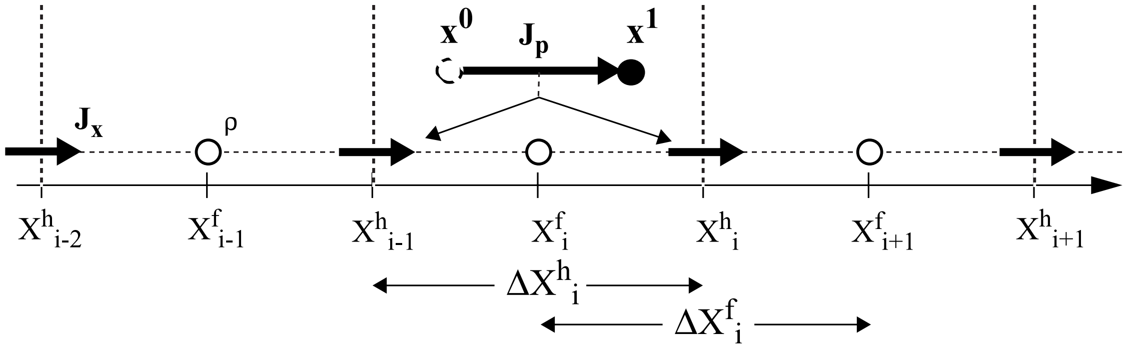

Figure 1. 1-D grid geometry and current density allocation for a single particle.

In three dimensions, each particle contributes charge to 27 cells,

$$\rho^n_{i+\alpha,\, j+\beta,\, k+\gamma} = w^n_{x\alpha} w^n_{y\beta} w^n_{z\gamma}. \qquad (5)$$

The transverse weighting of the current density depends on the path taken from the initial to the final location. Computing these weights is trivial for the six "rectangular" paths in which each coordinate is traversed with the others held fixed, *e.g.* first x^0 to x^1, then y^0 to y^1, and finally z^0 to z^1. However, the natural and physical choice is the straight line path between $\mathbf{x}^0$ and $\mathbf{x}^1$. To derive the transverse weights for this path, subdivide it into N segments, use *any* of the six *rectangular* paths for the N segments, and then let $N \to \infty$. The result is identical to simply averaging over the six full *rectangular* paths. For J_x, we have

$$J_{x,\, i-1,\, j+\beta,\, k+\gamma} = -(w^1_{x-} - w^0_{x-}) w^{Tx}_{\beta\gamma}, \text{ and } J_{x,\, i,\, j+\beta,\, k+\gamma} = (w^1_{x+} - w^0_{x+}) w^{Tx}_{\beta\gamma}, \qquad (6)$$

where

$$w^{Tx}_{\beta\gamma} = \frac{1}{3}(w^0_{y\beta} w^0_{z\gamma} + w^1_{y\beta} w^1_{z\gamma}) + \frac{1}{6}(w^0_{y\beta} w^1_{z\gamma} + w^1_{y\beta} w^0_{z\gamma}). \qquad (7)$$

J_y and J_z are similarly defined. An important property of defining J using Eqs. 6 and 7 is that charge is automatically conserved if the charge weights are modified from Eq. 3, provided that $w^n_{x0} = 1 - w^n_{x-} - w^n_{x+}$, with analogous constraints on the y and z weights.

Modification for bounded geometry

In QUICKSILVER, all surfaces on which particles are created or killed are coincident with faces of full-grid cells. With the first-order scheme, when a particle is at the surface of a kill cell, it contributes no charge to the interior vacuum grid points. Thus charge is conserved at interior points when the particle is removed from the system. This is not true with the second-order scheme; killing the particle leaves up to 1/8 of the particle charge at interior points. Similarly, if a particle is created at a conductor surface, charge is allocated to interior grid points, without charge-conserving modification to the electric field. The larger transverse stencil also leads to major complications for particle creation and particle/surface interaction models.

The approach we have taken to deal with boundaries is to modify the particle weighting near kill cells, smoothly transitioning from second to first order as particles move down to any kill surface. We define the half-grid cell's type, based on the kill status of the eight full-grid cells intersecting it. There are eight types: 0 = vacuum, 1-3 = flat surfaces, 4-6 = edges, and 7 = "other". Some examples are shown in Fig. 2 (slanted surfaces are implemented with a 3-D extension of Ref. 5). In a non-vacuum cell, we first compute the fraction of the second-order scheme to use, f_Q. For flat surfaces, f_Q is easily computed from the height above the surface. For edges, we compute f_Q by bilinear interpolation from values at the corners of the quadrant of the half-grid cell in which the particle is located. For conformal conductors, $f_Q = 1$ at corners in vacuum, zero otherwise. Slanted surfaces and slanted/conformal corners are treated as edges, using $f_Q = -1$ for the corner beneath the surface, as shown in Fig. 2d. For the *other* type, we must interpolate f_Q from values at the corners of an octant of the half-grid cell. The corner values of f_Q for edges and *other* types are all pre-computed at $t = 0$. Since they have at most three states, they are all packed into bit-fields of a single integer, along with the half-grid cell type.

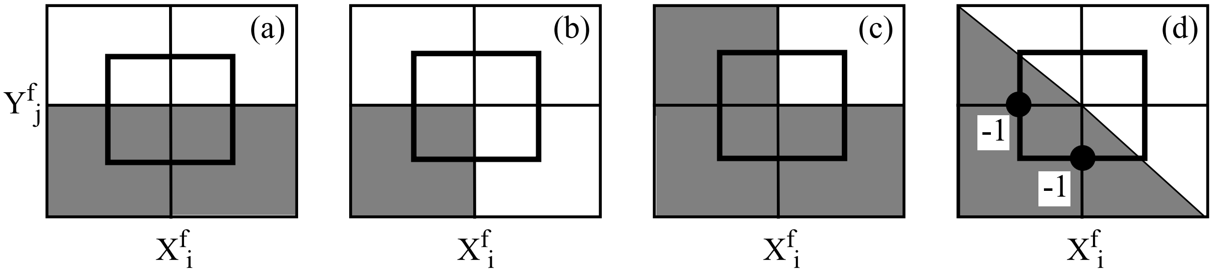

Figure 2. Examples of some half-grid cell types: (a) Flat surface, (b) and (c) are the two types of edge with conformal conductors, and (d) slanted surface.

The modified charge density weights for a non-vacuum cell are a linear combination of f_Q times the second-order weights using Eq. 5, and $(1-f_Q)$ times the standard first-order weights at the eight corners of the full-grid cell in which the particle is located. The current density is computed from the modified charge weights using Eqs. 6 and 7. This scheme results in allocation of ρ beneath the surface, but it is small enough that we do not bother to correct this; for a uniform grid, the worst case is 1/54 of a particle's charge beneath a flat surface. There are other details that the algorithm must handle to conserve charge. First, particle trajectories must be truncated at the surface of all conductors and all exterior boundaries. Second, mirror (PMC) boundaries require special handling. No transition to first-order weighting is needed, but **J** and ρ values accumulated behind the boundary must be mapped back to interior points.

Finally, we discuss the interpolation of electromagnetic fields to the particle. For the first-order scheme, it is customary to first interpolate **E** and **B** from their staggered locations on the Yee grid to the full-grid cell corners. The interpolation of all components of **E** and **B** to the particle can then done with a single set of weights. When $\Delta x/\lambda_{De} >> 1$, unstable electrostatic fluctuations cause unacceptable numerical heating. However, if **E** is interpolated directly from the Yee grid, using exactly the same weighting scheme with which **J** is laid down on the grid, energy conservation is excellent. This is the electromagnetic analog of well-known, energy-conserving electrostatic algorithms.[6] For the first-order scheme, this means using NGP weighting of **E** in the longitudinal direction, which may be excessively crude. However, for the second-order scheme, we have continuous, piecewise-linear interpolation of **E** in the longitudinal direction.

Results and Future Plans

Charge conservation has been verified in a wide variety of systems, some with very complex conductor geometry. The algorithm has also proved to be very useful for simulations involving plasma formation by ionization of a background gas: the second-order scheme successfully handles plasma densities at which the first-order scheme fails. The algorithm shows great promise for the dense electrode plasma problem. We have studied 1-D expansion of a collisionless plasma slab into vacuum, using cell size and timestep characteristic of our large production simulations, $\Delta x = 0.5$ mm and $\Delta t = 0.5$ ps. For plasma parameters $n = 10^{15}$ cm^{-3}, $T_e = 1$ eV, we have $\Delta x/\lambda_{De} = 2130$, $\omega_{pe}\Delta t = 0.89$. The plasma expansion is in good agreement with a multi-fluid simulation, and conserves total energy to within 5% over 200,000 timesteps. To actually use this capability, we are developing a new plasma emission model. We load electron/ion pairs near the surface to maintain a user-defined plasma density, 1 - 2 orders of magnitude higher than a standard space-charge-limited (SCL) emission algorithm [5] would generate. The code will then self-consistently extract the SCL current from this plasma. Finally, we note that particle collision models must also be developed to correctly treat plasma expansion across a magnetic field.

Acknowledgements

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin company, for the United States Department of Energy’s National Nuclear Security Administration under contract DE-AC04-94-AL85000.

References

1. T. Zh. Esirkepov, Comput. Phys. Comm. **135**, 144 (2001).

2. J. Villasenor, O. Buneman, Comput. Phys. Comm., **69**, 306 (1992).

3. M. E. Cuneo, IEEE Trans. Dielectr. Elect. Insul., **6**, 469 (1999).

4. J. P. Quintenz, D. B. Seidel, *et al.,* Laser Part. Beams 12, 283 (1994).

5. T. D. Pointon, J. Comput. Phys., **96**, 143 (1991).

6. C. K. Birdsall, A. B. Langdon, *Plasma Physics via Computer Simulation* (McGraw-Hill, New York, 1985), Ch. 10.