Description of Electrostatic Spectral Particle-in-Cell Code from the UPIC Framework

Viktor K. Decyk

Department of Physics and Astronomy
University of California, Los Angeles
Los Angeles, California 90095-1547

I. Introduction

This document presents the mathematical foundation of the periodic Particle-in-Cell electrostatic code in the UCLA Particle-in-Cell Framework. The electrostatic code uses only the Coulomb force of interaction between particles. This is the most fundamental of plasma models and is useful when inductive electric and magnetic fields are not important.

II. Electrostatic Plasma Model

The simplest model is the electrostatic model, where the force of interaction is determined by solving only the Poisson equation in Maxwell's equation. The main interaction loop is as follows:

1. Calculate charge density on a mesh from the particles:

$$\rho(\boldsymbol{x}) = \sum_i q_i S(\boldsymbol{x} - \boldsymbol{x}_i)$$

2. Solve Poisson's equation:

$$\nabla \bullet \boldsymbol{E} = 4\pi\rho$$

3. Advance particle co-ordinates using Newton's Law:

$$m_i \frac{d\boldsymbol{v}_i}{dt} = q_i \int \boldsymbol{E}(\boldsymbol{x}) S(\boldsymbol{x}_i - \boldsymbol{x})\, d\boldsymbol{x} \qquad \frac{d\boldsymbol{x}_i}{dt} = \boldsymbol{v}_i$$

The function S(**x**) is the particle shape function. For point particles, this would be a delta function, but in computer modeling extended shapes are commonly used. The codes described here are spectral and solve the electric field using Fourier transforms. For the electrostatic case and periodic boundary conditions, a procedure for a gridless system is as follows:

1. Fourier Transform the charge density:

$$\rho(\boldsymbol{k}, t) = \frac{1}{V} \int \sum_i q_i S(\boldsymbol{x} - \boldsymbol{x}_i(t)) e^{-i\boldsymbol{k}\bullet\boldsymbol{x}} d\boldsymbol{x} = \sum_i q_i S(\boldsymbol{k}) e^{-i\boldsymbol{k}\bullet\boldsymbol{x}_i(t)}$$

2. Solve Poisson's equation in Fourier space:

$$\boldsymbol{E}(\boldsymbol{k}) = \frac{-i\boldsymbol{k}}{k^2} 4\pi\rho(\boldsymbol{k})$$

Note that this equation implies that $\rho(\mathbf{k}=0) = 0$. This means that strictly periodic systems are charge neutral.

3. Fourier Transform the Smoothed Electric Field to real space:

$$\boldsymbol{E}_S(\boldsymbol{x}_i) = V \sum_{\boldsymbol{k}=-\infty}^{\infty} \boldsymbol{E}(\boldsymbol{k}) S(\boldsymbol{k}) e^{i\boldsymbol{k}\bullet\boldsymbol{x}i}$$

For delta function particles shapes, $S(\mathbf{k}) = 1/V$. When solving these equations on the computer, we generally use discrete space and time co-ordinates. In discretizing time, the explicit leap-frog integration scheme is commonly used, because it is second order accurate. In this scheme, the particle co-ordinates are known at staggered times.

The discrete equations of motion are as follows:

$$\boldsymbol{v}_i(t+\triangle t/2) = \boldsymbol{v}_i(t-\triangle t/2) + \frac{q_i}{m_i}\boldsymbol{E}_S(\boldsymbol{x}_i(t))\triangle t$$

$$\boldsymbol{x}_i(t+\triangle t) = \boldsymbol{x}_i(t) + \boldsymbol{v}_i(t+\triangle t/2)\triangle t$$

Although it is possible to use the gridless spectral field solver shown above in a very accurate particle code, this is quite slow. More commonly, charge density is accumulated on a grid from the particle co-ordinates according to some interpolation scheme. The fields are then calculated at the grid points, then interpolated to obtain the force at the particle's position. If we have (Nx,Ny,Nz) grid points for a system of size (Lx,Ly,Lz), then the grid spacings are:

$$\Delta_x = L_x/N_x \qquad \Delta_z = L_z/N_z \qquad \Delta_z = L_z/N_z$$

and the charge deposit is then defined to be:

$$\rho(\boldsymbol{r}) = \sum_i q_i \sum_{s'} W(\boldsymbol{r}-\boldsymbol{x}_i)\delta_{\boldsymbol{r},\boldsymbol{s}'}$$

where **r**, **s**′ are defined at integer values n, m, l, as follows:

$$\boldsymbol{r} = (n\Delta_x, m\Delta_y, l\Delta_z) \qquad\qquad \boldsymbol{s}' = (n', m', l')$$

and the vector delta function is the product of three Kronecker delta functions:

$$\delta_{\boldsymbol{r},\boldsymbol{s}'} = \delta_{n,n'}\delta_{m,m'}\delta_{l,l'}$$

This is analogous to the gridless case we had before:

$$\rho(\boldsymbol{x}) = \sum_i q_i S(\boldsymbol{x}-\boldsymbol{x}_i)$$

The important feature is that the interpolation function W be smooth and have limited support, that is, it is zero outside a small range. The interpolation function is usually the product of three interpolation functions in each co-ordinate. For example, the most common interpolation function is linear, given by:

$$W_x(x) = \begin{array}{ll} (\Delta_x + x)/\Delta_x^2, & -\Delta_x < x \leq 0 \\ (\Delta_x - x)/\Delta_x^2, & 0 \leq x < \Delta_x \end{array}$$

and similarly for the other co-ordinates. However, quadratic and cubic B-spline functions are sometimes used.

The Discrete Fourier Transform can now be used to obtain the Fourier transform of the density:

$$\rho(\boldsymbol{k}') = \frac{1}{N}\sum_{r}\rho(\boldsymbol{r})e^{-i\boldsymbol{k}'\cdot\boldsymbol{r}} = \frac{1}{N}\sum_{\boldsymbol{r}}\sum_{i} q_i W(\boldsymbol{r}-\boldsymbol{x}_i)e^{-i\boldsymbol{k}'\cdot\boldsymbol{r}}$$

where N = NxNyNz, and we define k′ as follows:

$$\boldsymbol{k}' = (\frac{2\pi n'}{L_x}, \frac{2\pi m'}{L_y}, \frac{2\pi l'}{L_z})$$

This is analogous to the gridless case we had before:

$$\rho(\boldsymbol{k}) = \frac{1}{V}\int\sum_{i} q_i S(\boldsymbol{x}-\boldsymbol{x}_i(t))e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}d\boldsymbol{x}$$

The major complication of using a grid is that non-physical grid forces can arise that were absent before. These arise from aliasing, which occurs when the continuous particle co-ordinates $\mathbf{x}_i$ have spatial variations less than the grid spacing. Such variations cannot be resolved, but get mapped onto longer wavelengths.

To see explicitly how this occurs, we can write the interpolation function as an infinite Fourier series, as follows:

$$W(\boldsymbol{x}) = \sum_{\boldsymbol{k}=-\infty}^{\infty} W(\boldsymbol{k})e^{i\boldsymbol{k}\cdot\boldsymbol{x}} \qquad\qquad W(\boldsymbol{k}) = \frac{1}{V}\int W(\boldsymbol{x})e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}d\boldsymbol{x}$$

this leads to:

$$\rho(\boldsymbol{k}') = \frac{1}{N}\sum_{i} q_i \sum_{\boldsymbol{k}=-\infty}^{\infty} W(\boldsymbol{k})e^{-i\boldsymbol{k}\cdot\boldsymbol{x}_i}\sum_{\boldsymbol{r}} e^{i(\boldsymbol{k}-\boldsymbol{k}')\cdot\boldsymbol{r}}$$

The last sum over $\mathbf{r}$ is a geometric series which can be summed explicitly to give:

$$\sum_{\boldsymbol{r}} e^{i(\boldsymbol{k}-\boldsymbol{k}')\cdot\boldsymbol{r}} = N\sum_{\boldsymbol{k}_N=-\infty}^{\infty}\delta_{\boldsymbol{k},\boldsymbol{k}'+\boldsymbol{k}_N}$$

where $\mathbf{k}_N$ represents wavelengths which cannot be resolved:

$$\boldsymbol{k}_N = (\frac{2\pi n''}{\Delta_x}, \frac{2\pi m''}{\Delta_y}, \frac{2\pi l''}{\Delta_z})$$

As a result, one can write the Fourier transform of the density as follows:

$$\rho(\boldsymbol{k}') = \sum_{i} q_i\left[W(\boldsymbol{k}') + \sum_{k_N\neq 0} W(\boldsymbol{k}'+\boldsymbol{k}_N)e^{-i\boldsymbol{k}_N\cdot\boldsymbol{x}_i}\right]e^{-i\boldsymbol{k}'\cdot\boldsymbol{x}_i}$$

Compare this with the gridless case we had before:

$$\rho(\boldsymbol{k}) = \sum_{i} q_i S(\boldsymbol{k})e^{-i\boldsymbol{k}\cdot\boldsymbol{x}_i}$$

One can see that W($\mathbf{k}'$) acts like a particle shape factor, similar to the function S($\mathbf{k}$) in the gridless case. The terms involving non-zero values of $\mathbf{k}_N$ are the non-physical aliased terms. The electric field is solved at the gridpoints as in the gridless case, except for the use of the Discrete Fourier Transform.

$$\boldsymbol{E}(\boldsymbol{k}') = \frac{-i\boldsymbol{k}'}{k'^2} 4\pi\rho(\boldsymbol{k}')$$

$$\boldsymbol{E}(\boldsymbol{r}) = \sum_{\boldsymbol{k}'} \boldsymbol{E}(\boldsymbol{k}') e^{i\boldsymbol{k}'\cdot\boldsymbol{r}}$$

Obtaining the electric field at the particle's location involves another interpolation.

$$\boldsymbol{E}_S(\boldsymbol{x}_i) = \sum_{\boldsymbol{r}} \boldsymbol{E}(r) W(\boldsymbol{x}_i - \boldsymbol{r}) \Delta$$

where

$$\Delta = \Delta_x \Delta_y \Delta_z$$

Proceeding as before, one can show that:

$$\boldsymbol{E}_S(\boldsymbol{x}_i) = V \sum_{\boldsymbol{k}'} \boldsymbol{E}(\boldsymbol{k}') \Big[W(\boldsymbol{k}') + \sum_{k_N \neq 0} W(\boldsymbol{k}' + \boldsymbol{k}_N) e^{i\boldsymbol{k}_N\cdot\boldsymbol{x}_i} \Big] e^{i\boldsymbol{k}'\cdot\boldsymbol{x}_i}$$

This is analogous to the gridless case:

$$\boldsymbol{E}_S(\boldsymbol{x}_i) = V \sum_{\boldsymbol{k}=-\infty}^{\infty} \boldsymbol{E}(\boldsymbol{k}) S(\boldsymbol{k}) e^{i\boldsymbol{k}\cdot\boldsymbol{x}i}$$

Note that with a grid, the force accelerating a particle no longer depends merely on the separation of particles, but also on the distance of each particle from the grid. In other words, the particles are not only interacting with each other, but also with a periodic structure formed by the grid itself. This non-conservative force usually leads to self-heating, sometimes to instability. The aliasing can be minimized in one of two ways. One is to use a higher order interpolation function whose Fourier series is a small as possible for $\mathbf{k} > \mathbf{k}_N$. This is more expensive in computer time. Alternatively, one can use an additional shape function S($\mathbf{x}$) in addition to the interpolation function. Both these methods effectively make the particles "fatter," and it is hard to maintain density variations that are smaller than the particle size. On can also regard them as filter functions.

The most common interpolation functions in use are the B-splines. They have a Fourier transforms for each component i given by:

$$W_n(k_i) = \frac{1}{L_i} \left[\frac{\sin(k_i \Delta /2)}{k_i \Delta /2} \right]^{n+1}$$

These functions have maxima near $\mathbf{k} = (p+1/2)\mathbf{k}_N$, where p is an integer > 1. The worst aliasing occurs for p = 1, which maps density variations at $\mathbf{k} = 3\mathbf{k}_N/2$ to $\mathbf{k} = \mathbf{k}_N/2$.

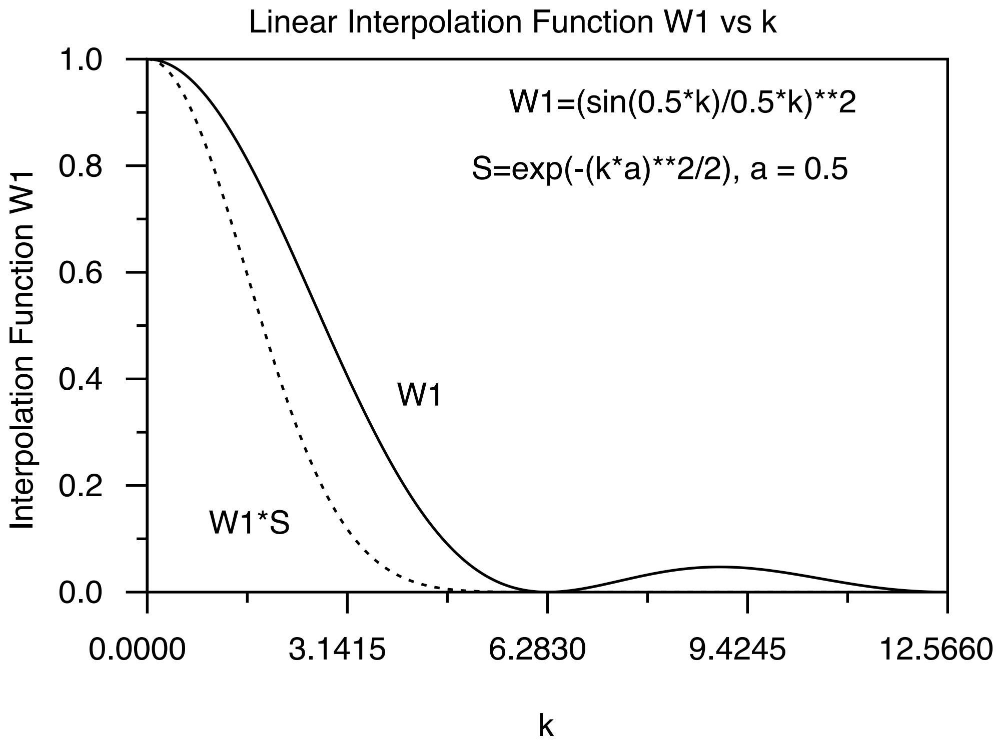

Figure 1. Fourier transform of first order interpolation function W1 with and without a gaussian smoothing function S. Modes with k > 2π mapped to modes with k < 2π.

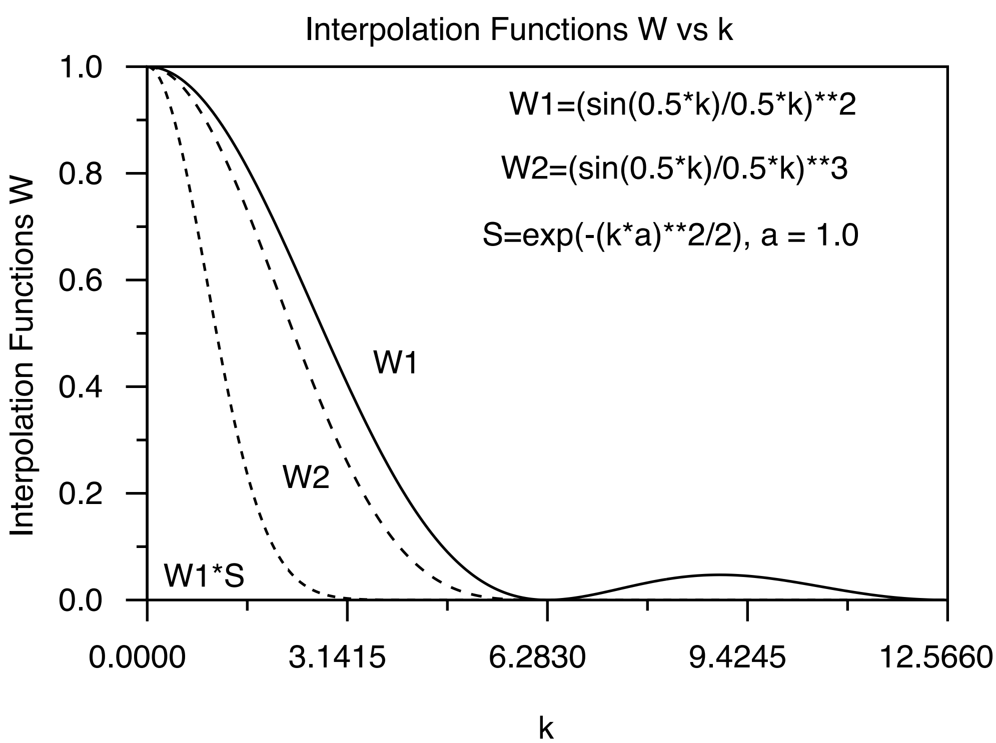

Figure 2. Fourier transforms of first and second order interpolation functions W1 and W2, and W1 with gaussian smoothing S.

A particle shape (or filter) function $S(\mathbf{k})$ which is small in the vicinity of $\mathbf{k}=\mathbf{k}_N/2$ will suppress the aliasing. Of course, one is also suppressing some physical modes, so this scheme is limiting the resolution of the model. The lesson here is that when using grids, one must suppress, one way or another, information which cannot be resolved. Higher order interpolations have better resolution, but are more costly. In Fourier space, a common filter function is:

$$S(k_i) = e^{-(k_i a_i)^2/2}/L_i$$

where $\mathbf{a}$ is the particle size, which corresponds to a gaussian particle in space.

If filtering is used to suppress aliasing, then the effective particle shape is given by:

$$S_{eff}(\boldsymbol{k}) = V \cdot \prod_i W(k_i) S(k_i)$$

In real space this corresponds to a convolution of the interpolation function with the filter function. For non-spectral codes, such filtering is typically done in real space. When filtering is used, spectral electrostatic codes can conserve energy to parts per million over thousands of time steps, and conserve momentum to round-off error.

Even when aliasing is suppressed, grid effects are still present, primarily due to the use of "fat" particles rather than point particles. The easiest way to understand this, is to note that in Fourier space one replaces $q_i \Rightarrow q_i S_{eff}(\mathbf{k})$, both in the charge deposit and in the force calculation. In plasma theory, charge enters only in the calculation of the plasma frequency, so that if aliasing is negligible:

$$\omega_{pe}^2 \Rightarrow \omega_{pe}^2 (V \cdot S_{eff}(\boldsymbol{k}))^2$$

Where the plasma frequency now depends on $\mathbf{k}$, and may not be isotropic. For linear interpolation, gaussian smoothing with $a/\Delta > 0.5$, and an isotropic grid, one has:

$$\omega_{pe}^2 \Rightarrow \omega_{pe}^2 e^{-k^2(a^2+\Delta^2/6)} \approx \omega_{pe}^2 \left[1 - k^2(a^2 + \Delta^2/6)\right]$$

Thus to see how the grid affects the plasma, one can replace the plasma frequency which appears in plasma theory with the above expression. Whether this k dependence of the plasma frequency is important or not depends on the plasma parameters and waves under study. For example, the dispersion relation for plasma waves:

$$\omega^2 = \omega_{pe}^2 + 3k^2 v_{the}^2$$

becomes:

$$\omega^2 = \omega_{pe}^2 + \left[3 - \frac{a^2 + \Delta^2/6}{\lambda_{DE}^2}\right] k^2 v_{the}^2$$

Whether this is important or not depends on the size of the grid relative to the Debye length.

III. Energy and Momentum Flux

For the electromagnetic model, the energy flux is well known to be given by the Poynting vector **S**:

$$\nabla \bullet \boldsymbol{S} + \frac{\partial}{\partial t}\left[\frac{\boldsymbol{E} \bullet \boldsymbol{E}}{8\pi} + \frac{\boldsymbol{B} \bullet \boldsymbol{B}}{8\pi}\right] = -\boldsymbol{j} \bullet \boldsymbol{E}$$

where

$$\boldsymbol{S} = \frac{c}{4\pi}\boldsymbol{E} \times \boldsymbol{B}$$

This equation describes the conservation of energy: the time rate of change of electromagnetic field energy plus the outflow of the energy is equal to the negative of the work done on the particles. This equation is not unique and other energy flux equations can also be derived: only differences in energy and flux are significant. It is less well known that analogous energy flux equations can be derived for the electrostatic and Darwin models.

For the electrostatic model, an energy flux equation is given by:

$$\nabla \bullet \boldsymbol{S} + \frac{\partial}{\partial t}\left[\frac{\boldsymbol{E}_L \bullet \boldsymbol{E}_L}{8\pi}\right] = -\boldsymbol{j} \bullet \boldsymbol{E}_L$$

where

$$\boldsymbol{S} = \left[\boldsymbol{j} - \frac{1}{4\pi}\nabla\frac{\partial \phi}{\partial t}\right]\phi$$

and

$$\boldsymbol{E}_L = -\nabla\phi$$

This equation can be easily shown by making use of the equation of continuity and the identity:

$$\nabla \bullet (f\boldsymbol{V}) = \boldsymbol{V} \bullet \nabla f + f(\nabla \bullet \boldsymbol{V})$$

An alternate form of this equation can be derived by using the result,

$$\nabla \bullet \left[\frac{\phi\nabla\phi}{8\pi}\right] = \frac{\boldsymbol{E}_L \bullet \boldsymbol{E}_L}{8\pi} - \frac{1}{2}\rho\phi$$

to obtain:

$$\nabla \bullet \boldsymbol{S}' + \frac{\partial}{\partial t}\left[\frac{1}{2}\rho\phi\right] = -\boldsymbol{j} \bullet \boldsymbol{E}_L$$

where the alternative energy flux vector is

$$\boldsymbol{S}' = \boldsymbol{j}\phi + \frac{1}{8\pi}\left[\frac{\partial \phi}{\partial t}\nabla\phi - \phi\nabla\frac{\partial \phi}{\partial t}\right]$$

The electrostatic energy in the form $\rho\phi/2$ is useful for isolated systems.

In addition to the energy flux, the momentum flux equation is also useful. For the electromagnetic case, the equation is well known:

$$\nabla \cdot \overleftrightarrow{\mathbf{T}} - \frac{1}{c^2}\frac{\partial \boldsymbol{S}}{\partial t} = \rho \boldsymbol{E} + \boldsymbol{j} \times \boldsymbol{B}/c$$

where

$$\overleftrightarrow{\mathbf{T}} = \frac{1}{4\pi}\left[\boldsymbol{EE} + \boldsymbol{BB} - \frac{1}{2}(\boldsymbol{E} \cdot \boldsymbol{E} + \boldsymbol{B} \cdot \boldsymbol{B})\,\overleftrightarrow{\mathbf{I}}\right]$$

is the Maxwell Stress Tensor. The quantity $\mathbf{S}/c^2$ is the momentum in the electromagnetic field.

In the electrostatic case, there is no momentum in the longitudinal field and the magnetic field vanishes, so the momentum flux equation reduces to:

$$\nabla \cdot \overleftrightarrow{\mathbf{T}} = \rho \boldsymbol{E}$$

where

$$\overleftrightarrow{\mathbf{T}} = \frac{1}{4\pi}\left[\boldsymbol{EE} - \frac{1}{2}(\boldsymbol{E} \cdot \boldsymbol{E})\,\overleftrightarrow{\mathbf{I}}\right]$$

These energy and momentum flux equations are not unique, and alternative forms are possible and useful.

IV. Units

These codes use dimensionless grid units, which means that distance is normalized to some distance δ. Generally, this distance δ is the smallest distance which needs to be resolved in the code, such as a Debye length. Time is normalized to some frequency ω_0. Generally this frequency is the highest frequency that needs to be resolved in the code, such as the plasma frequency. Charge is normalized to the absolute value of the charge of an electron e. Mass is normalized to the mass of an electron m_e. Other variables are normalized from some combination of these.

In summary, dimensionless position, time, velocity, charge, and mass are given by:

$$\widetilde{x} = x/\delta \qquad \widetilde{t} = \omega_0 t \qquad \widetilde{\boldsymbol{v}} = \boldsymbol{v}/\delta\omega_0 \qquad \widetilde{q} = q/e \qquad \widetilde{m_e} = m/m_e$$

Dimensionless charge and current densities are given by:

$$\widetilde{\rho} = \rho\delta^3/e \qquad \widetilde{j} = j\delta^3/e\delta\omega_0$$

Dimensionless electric field and potential are given by:

$$\widetilde{\boldsymbol{E}} = e\boldsymbol{E}/m_e\omega_0^2\delta \qquad \widetilde{\phi} = e\phi/m_e\omega_0^2\delta^2$$

Dimensionless energy is given by:

$$\widetilde{W} = W/m_e\omega_0^2\delta^2$$

The dimensionless particle equations of motion are:

$$\widetilde{m_i}\frac{d\widetilde{\boldsymbol{v}_i}}{d\widetilde{t}} = \widetilde{q_i}\left[\widetilde{\boldsymbol{E}} + \widetilde{\boldsymbol{v}_i} \times \widetilde{\boldsymbol{B}}\right] \qquad\qquad \frac{d\widetilde{\boldsymbol{x}_i}}{d\widetilde{t}} = \widetilde{\boldsymbol{v}_i}$$

The dimensionless Poisson equations is:

$$\widetilde{\nabla} \bullet \widetilde{\boldsymbol{E}} = A_f \widetilde{\rho}$$

where

$$A_f = \frac{4\pi e^2}{m_e\omega_0^2\delta^3}$$

defines the relation between the sources and the fields. Whatever time and space scales are chosen, these equations have the same form. Only the constant A_f changes.

In these codes, the normalization length is chosen to be the grid spacing,

$$\delta = L_x/N_x = L_y/N_y = L_z/N_z$$

and the normalization frequency to be the plasma frequency ω_{pe}. In that case, one can show that:

$$A_f = \frac{1}{n_o \delta^3} = \frac{N_x N_y N_z}{N_p}$$

where Np is the number of particles. The grid spacing is then related to some other dimensionless physical parameter, typically the Debye length. Thus:

$$\lambda_{De} / \delta = \frac{v_{the}}{\delta \omega_{pe}} = \tilde{v}_{the}$$

where the dimensionless thermal velocity is an input to the code. Note that if the grid space is equal to Debye length, then A_f is identical to the plasma parameter g which appears as an small expansion parameter in plasma theory.

Bibliography

[1] V. K. Decyk, “UPIC: A framework for massively parallel particle-in-cell codes,” Computer Phys. Comm. 177, 95 (2007).

[2] R. W. Hockney and J. W. Eastwood, Computer Simulation Using Particles [McGraw-Hill, New York, 1981].

[3] C. K. Birdsall and A. Bruce Langdon, Plasma Physics via Computer Simulation [McGraw-Hill, New York, 1985].

[4] V. K. Decyk, “Particle Simulation of RF Heating and Current Drive,” Computer Physics Reports 4, 245 (1986).

[5] V. K. Decyk, “Energy and Momentum Conservation Theorems for Electrostatic Simulations,” J. Computational Phys. 56, 461 (1984).

[6] Decyk, V.K. and Slottow, J.E., "Supercomputers in the Classroom," Computers in Physics, vol. 3, No. 2, 1989, p. 50.