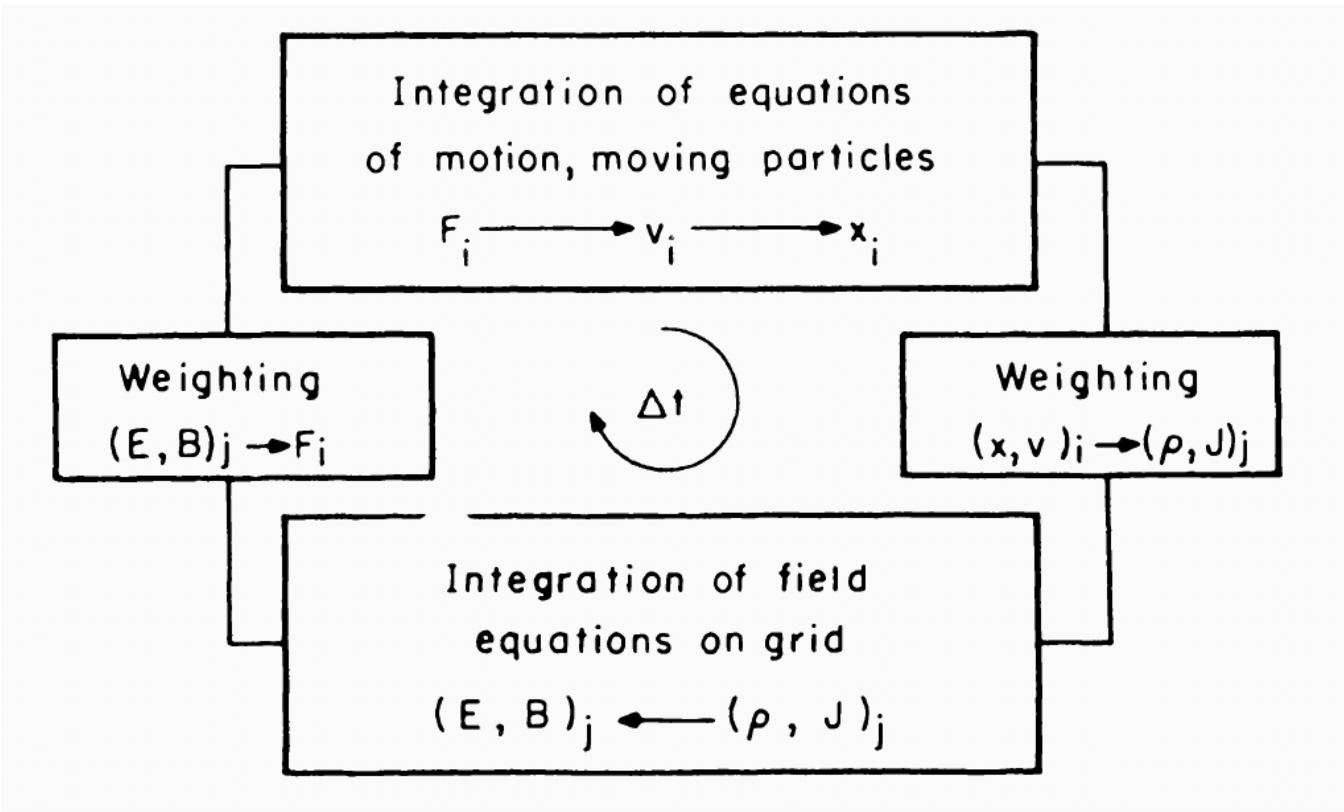
Notes on PIC Codes

PIC Computational Cycle

Given some initial distribution of particle quantities $(\mathbf{v}, \mathbf{x}, q)_i$, we calculate the bulk quantities on the spatial grid (ρ, J) which allow us to solve for the field quantities $(\mathbf{E}, \mathbf{B})$. Once we have these field quantities on the grid, we can define the force $\mathbf{F}_i$ on each particle. We apply the forces to the particles, calculate their new quantities from such an application over a predefined *time-grid*, and repeat the process until the final time step.



Integration of the Equations of Motion

- Use Leapfrog method to integrate the following equations of motion expressed as finite difference equations (recall limit definition of derivative):

$$m\frac{d\mathbf{v}}{dt} = \mathbf{F} \implies \mathbf{m}\frac{\mathbf{v}_{\text{new}} - \mathbf{v}_{\text{old}}}{\boldsymbol{\Delta}\mathbf{t}} = \mathbf{F}_{\text{old}}$$

$$\frac{d\mathbf{x}}{dt} = \mathbf{v} \implies \frac{\mathbf{x}_{\text{new}} - \mathbf{x}_{\text{old}}}{\boldsymbol{\Delta}\mathbf{t}} = \mathbf{v}_{\text{new}}$$

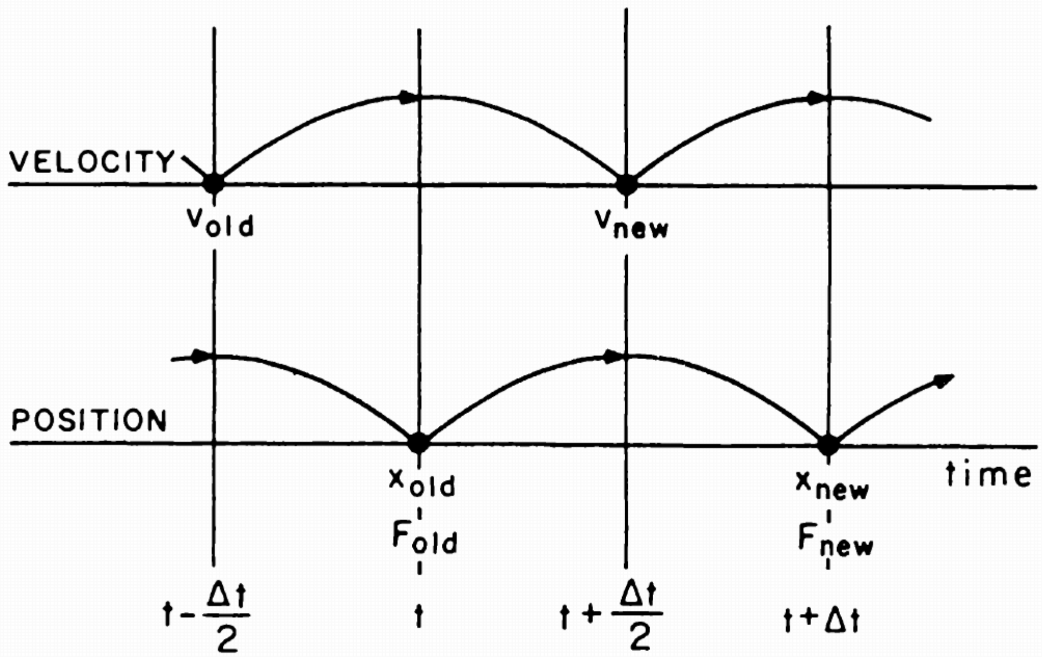


Figure 2-4a Sketch of leap-frog integration method showing time-centering of force **F** while advancing **v**, and of **v** while advancing **x**.

Integration of the Field Equations

Because the field quantities $(\mathbf{E}, \mathbf{B})$ are calculated only at the grid points themselves, we need only to solve for them after each update of the particle quantities. An update of the particle quantities gives us new values of $(x, q)_i$, which allows us to calculate the charge density at each grid point. This can be done in a variety of ways (more to explain later), but we attain ρ_j for each cell j and then do the following:

- Assuming the problem is electrostatic, we have that $\mathbf{B} = \mathbf{0}$. This allows us to write

  $$\mathbf{E} = -\nabla\phi$$

  for some potential ϕ, because $\nabla \times \mathbf{E} = \partial_\mathbf{t}\mathbf{B} = \mathbf{0}$.
- With the previous equation and Gauss' law $\nabla \cdot \mathbf{E} = \rho/\epsilon_\mathbf{0}$, we obtain *Poisson's equation*

  $$\nabla^2\phi = -\frac{\rho}{\epsilon_0}.$$

- Now, we use the Fast Fourier Transform (FFT) to solve these equations for $\mathbf{E}$ below. Note that the gradient is implemented again through a finite difference method.

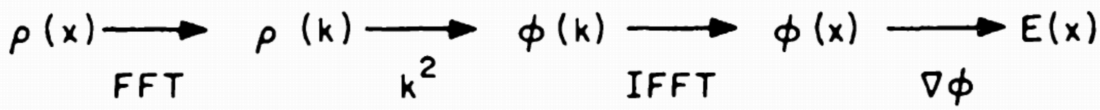


Figure 2-5b A possible sequence for solving Poisson's equation using the fast Fourier transform (FFT) and its inverse (IFFT).

Particle And Force Weighting; Connection Between Grid and Particle Quantities

Because particles can have positions and velocities in between grid points, we must have some way of converting for example the charge of particles *within* a cell to the charge density ρ_j at that cell. This is done through a weighting scheme.

- Zero order weighting assumes that the charge of the particle is distributed exactly at its location, resulting in a "rectangular shape" as seen below.

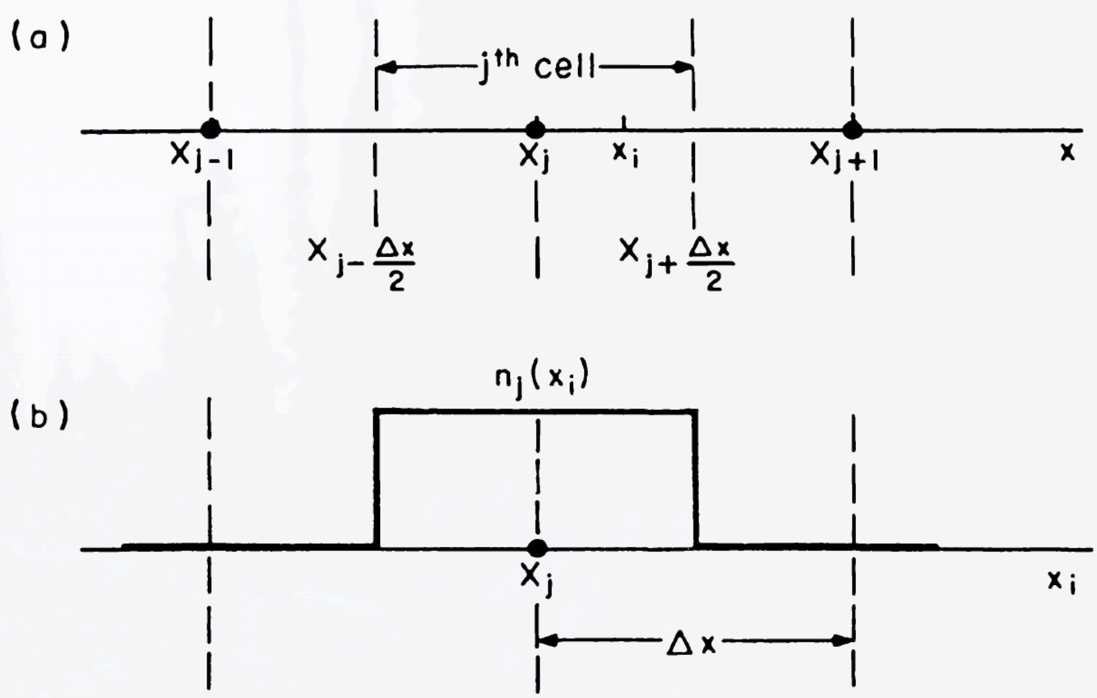


Figure 2-6a (a) Zero-order particle and field weighting, also called nearest-grid-point, or, NGP. Particles in the j^{th} cell, that is, with positions $x_i \epsilon X_j \pm \Delta x/2$, are assigned to X_j to obtain grid density $n(X_j)$. All of these particles are acted on by the field at X_j, $E(X_j)$. (b) The density $n_j(X_i)$ at point X_j due to a particle at x_i, as the particle moves through the cell centered on X_j. This density may be interpreted as the effective particle shape.

- A first order weighting assumes that the charge of a particle is distributed uniformly in a rectangular cloud with the particles position x_i as its center, and length Δx as seen below.

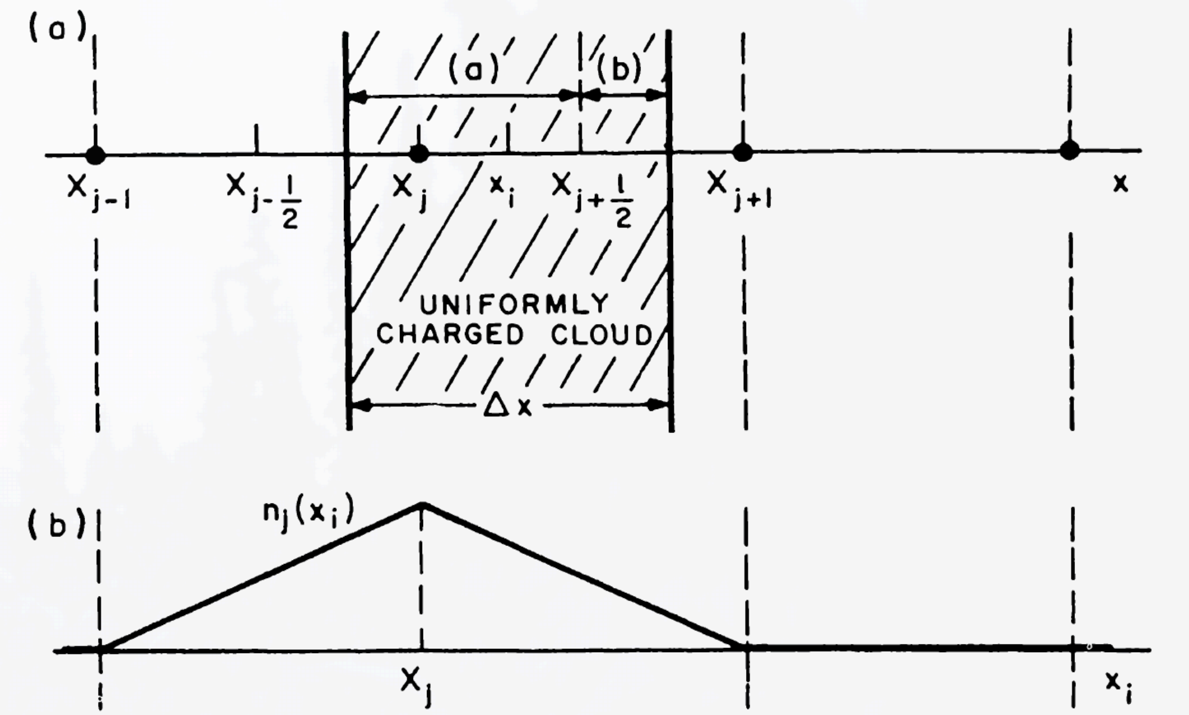


Figure 2-6b (a) First-order particle weighting, or cloud-in-cell model CIC. The nominal finite-size charged particle, or cloud, is one cell wide, with center at x_i. This weighting puts that part of the cloud which is in the j^{th} cell at X_j, fraction (a), and that part which is in the $(j+1)^{th}$ cell at X_{j+1}, fraction (b). This weighting is the same as applying NGP interpolation to each elemental part. (b) The grid density $n_j(x_i)$ at point x_i as the particle moves past X_j, again displaying the effective particle shape $S(x)$.

Normalization of Equations of Motion

We normalize our length parameter by the Debye wavelength

$$x' = \frac{x}{\lambda_D}$$

We normalize our time parameter by the plasma frequency

$$t' = t\omega_p$$

These transform the equations of motion in the following way. Notice how our normalized velocity is just the thermal velocity:

$$\frac{dx'}{dt'} = v' \implies \frac{1}{\omega_p \lambda_D}\frac{dx}{dt} = \frac{v}{\lambda_D\,\omega_p} \implies v' = \frac{v}{\lambda_D\,\omega_p} = v_{th}$$

$$m\frac{dv'}{dt'} = \frac{m}{\lambda_D\,\omega_p^2}\frac{dv}{dt} = \frac{qE}{v_{th}\,\omega_p}$$

We thus define the normalized electric field as follows, so the equation of motion becomes

$$E' = \frac{q}{v_{th}\,\omega_p} E \implies \frac{dv'}{dt'} = \frac{E'}{m}$$