Introduction to Computational Physics

Lecture N-Body Interactions

Andreas Adelmann

Paul Scherrer Institut, Villigen
E-mail: andreas.adelmann@psi.ch

https://moodle-app2.let.ethz.ch/course/view.php?id=15323

12.1 Modelling of charged particles I

Now we know how to model fields and assumed that *somehow* the values at the nodes of the grid exists.

- Many different applications in areas such as: Astrophysics, Particle Accelerator Science, Fusion and Fluid Dynamics - just to name a few, involve non linear interaction of charged particles.
- What are the adequate models ?
  - Microscopic model: System of N particles, whose state at any given time is determined by position and velocity
    - 6N-dimensional space.
    - Exact knowledge of this system impossible, because N very large.
    - Approximate models needed.

Approximate models

- Two levels of approximation
  - Kinetic model
    Statistical approach is used. Consider probability of particle to have a given position and velocity → very large system, simplifications are needed.

12.1 Modelling of charged particles II

  - ⋆ Mean-field interactions → Vlasov
  - ⋆ Binary collisions → Boltzmann

- Fluid model (we will not discuss)
  Obtained by taking a few moments (generally 2 or 3) of the kinetic equation and adding a closure relation.

12.2 A generic N-Body Framework I

We want to compute trajectories of N particles (small and large masses, charged particles).

For a particle i at position $\mathbf{r}_i$, Newton's law of motion is

$$m_i \frac{d\mathbf{v}_i}{dt} = \mathbf{F}_i, \qquad \mathbf{v}_i = \frac{d\mathbf{r}_i}{dt},$$

where the force F_i is due to the remaining $N-1$ particles, plus maybe external forces,

$$\mathbf{F}_i = \sum_{j \neq i} \mathbf{F}_{ij} + \mathbf{F}_{\mathrm{ext}}.$$

Ref: R.W.Hockney and J.W. Eastwood: Computer Simulation using particles. IOP Publishing 1988.

$\mathbf{F}_{ij}$ is the force that particle j exerts on particle i. With mass particles this is gravitation

$$\mathbf{F}_{ij} = -G m_i m_j \frac{(\mathbf{r}_j - \mathbf{r}_i)}{|\mathbf{r}_j - \mathbf{r}_i|^3} = -\nabla_i \phi_{ij}$$

12.2 A generic N-Body Framework II

with the (pairwise) gravitational potentials

$$\phi_{ij} = G\frac{m_i m_j}{r_{ij}}, \qquad r_{ij} = |\mathbf{r}_j - \mathbf{r}_i|.$$

Potential	Form
Gravitation	$G\frac{m_i m_j}{r}$
Coulomb	$\frac{q_i q_j}{4\pi\varepsilon_0}\frac{1}{r}$
van der Waals	$\frac{a}{r^{12}} - \frac{b}{r^6}$

- the particle-particle interactions with N particle requires the computation of $N(N-1)/2 = \mathcal{O}(N^2)$ pairwise distances in each time step.
- this restricts the number of simulation particles
- simulations with billions of particles not practicable

12.2 A generic N-Body Framework III

We proceed as follows. Notice that the force and potential energy of particle i are given by

$$\mathbf{F}_i = \mathbf{F}(\mathbf{r})|_{\mathbf{r}=\mathbf{r}_i} \qquad \phi_i = \phi(\mathbf{r})|_{\mathbf{r}=\mathbf{r}_i}, \qquad \mathbf{F}(\mathbf{r}) = -\nabla\phi(\mathbf{r}).$$

The potential field ϕ is related to the particle mass distribution by the field equation, for gravitational problems (and similarly for electrostatic problems) this is the Poisson equation

$$-\Delta\phi = \rho,$$

where ρ is the mass/charge density.

Solve the Poisson equation on a **mesh** that contains all particles.

Two approximations:

1. the masses of the individual particles are lumped in the grid points of the mesh.

12.2 A generic N-Body Framework IV

2. the values of ϕ at the location of individual mass points have to be interpolated from the values of ϕ at the grid points.

We require for consistency

- these operations should be $\mathcal{O}(h^2)$ accurate
- the interpolation is done in the (bi-, tri-)linear fashion

12.3 Particle Methods I

- Particle-In-Cell (PIC) method :
  - Idea: Follow particle trajectories, use grid field solve.
  - Literature:
    - Physics [Birdsall-Langdon 1991, Hockney-Eastwood 1988]
    - mathematical analysis [Neunzert-Wick 1979, Cottet-Raviart 1984, Victory-Allen 1991]
- P3M:
  - split forces into short and long range
  - includes collisions
  - [Hockney-Eastwood 1988]
- Langevin
  - discretization of collisional operator using Rosenbluth potentials
- Smooth Particle Hydrodynamics (SPH) methods:
  - Idea: compute interaction between finite sized macro-particles.
  - meshfree Lagrangian method where the co-ordinates move with the macro-particles
  - Literature: [Bateson-Hewett 1998]

12.3 Particle Methods II

- Tree Methods
    - mesh free

12.4 Discretization of the Vlasov equation by a particle method I

- Particle approximation of the Vlasov equation. Distribution function is approximated by

$$f_h(x, v, t) = \sum_k W_k \delta(x - x_k(t))\delta(v - v_k(t)).$$

- Deterministic, pseudo-random or Monte-Carlo approximation of f_0.
- Once particles have been initialized, they are advanced using deterministic equations of motion

Remark

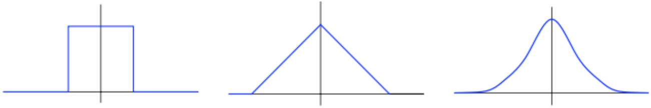
The particle motion is independent of the mesh

Coupling particles with fields through shape functions

- Particle method defines point particles (Dirac masses).

12.4 Discretization of the Vlasov equation by a particle method II

- Regularization by convolution with a finite width smoothing kernel generally called weighting function W.
    - Splines of different orders on are generally used on structured grids

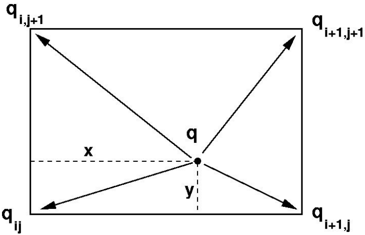


NGP CIC order 2

-
    - P^1 Finite Element shape functions on unstructured grids. Generally fields obtained by Poisson or Maxwell field solver first interpolated at vertices of mesh
    - Truncated Gaussians have also been used by some authors [Jacobs-Hesthaven 2006].

12.5 CIC in 2 D I

Charge assignment (lumping) can (again) be done in various ways.
Assigning a charge to the nearest grid point is in general too crude.
In general fractions of a charge in a cell are assigned to each four (in 2D) vertices of the cell.



$$
\begin{aligned}
q_{ij} &= q(1-\frac{x}{h_x})(1-\frac{y}{h_y}) \\
q_{i+1,j} &= q\frac{x}{h_x}(1-\frac{y}{h_y}) \\
q_{i,j+1} &= q(1-\frac{x}{h_x})\frac{y}{h_y} \\
q_{i+1,j+1} &= q\frac{x}{h_x}\frac{y}{h_y}
\end{aligned}
$$

Clearly, $q_{ij} + q_{i+1,j} + q_{i,j+1} + q_{i+1,j+1} = q$.

12.6 Coupling with the Field solver: The Particle-In-Cell (PIC) Method I

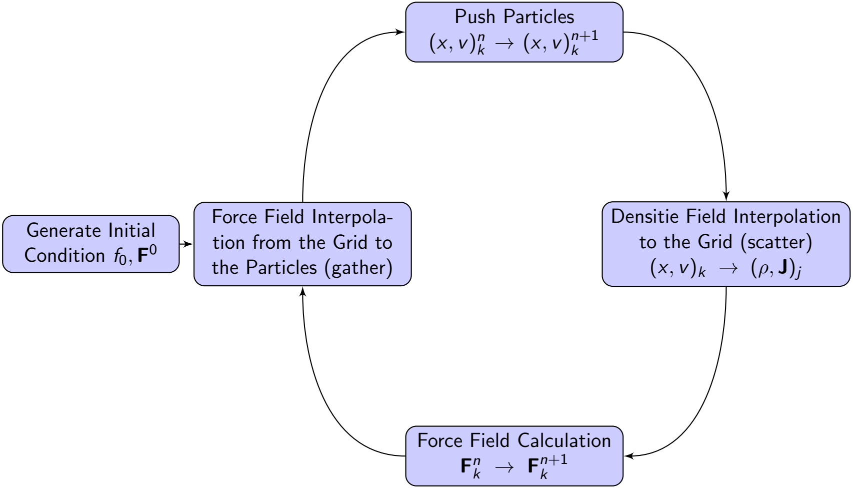
- Particle data scattered to surrounding grid points to compute charge and current densities using weight function:

$$\rho_i = q \sum_k W(x_k - x_i), \quad \mathbf{J}_i = q \sum_k W(x_k - x_i)\mathbf{v}_k.$$

- Field solve performed on grid: $\mathbf{E}(x_i)$, $\mathbf{B}(x_i)$ are computed using some grid based field solver.
- Fields are computed on particles using interpolation e.g. $\mathbf{E}(x_k) = \sum W(x_k - x_i)\mathbf{E}(x_i)$.

12.6 Coupling with the Field solver: The Particle-In-Cell (PIC) Method II

Typical PIC Loop



Generate Initial Condition $f_0, \mathbf{F}^0$

Force Field Interpolation from the Grid to the Particles (gather)

Push Particles $(x, v)_k^n \rightarrow (x, v)_k^{n+1}$

Densitie Field Interpolation to the Grid (scatter) $(x, v)_k \rightarrow (\rho, \mathbf{J})_j$

Force Field Calculation $\mathbf{F}_k^n \rightarrow \mathbf{F}_k^{n+1}$

Remarks I

- External forces (as magnetic field) act on all particles in a simple way. No particle–particle interaction. Trivial to parallelize. $\mathcal{O}(n)$ complexity.
- Near-field forces (as forces between bound atoms). Subdivision of physical space such that each subdomain contains about equally many particles. Communication restricted to nearest neighbor subdomains. $\mathcal{O}(n)$ complexity.
  Possible difficulties with load-balancing:
  - particles not equally distributed
  - particles moving from one subdomain to another

Remarks II

- Far-field forces (gravitation, Coulomb forces)
  Each particle exerts force on each other,

  $$\mathbf{F}_i^{\text{far-field}} = \sum_{j \neq i} \mathbf{F}_{ij} \qquad (*)$$

  Some far-field forces can be treated like near-field forces if they decay very rapidly with their distance $r = |\mathbf{x}_i - \mathbf{x}_j|$, like van der Waals forces ($\sim 1/r^6$).

  Formula $(*)$ indicates $\mathcal{O}(n^2)$ complexity.

  Now we want to find out how we can compute these sums with less work:

  - $\mathcal{O}(n)$
  - $\mathcal{O}(n \log n)$,

  and how to parallelize the field computation.

Remarks III

Methods: trade accuracy for speed.

- Particle-mesh methods
  Use rectangular grid. Move particles to grid points. Compute force potential via Poisson equation using fast solvers as FFT or multigrid.
- Method of Barnes-Hut and Fast multipole method (FMM)
  Are based on the idea that the influence of a cluster of stars (like a galaxies) on remote object is essentially given by the overall mass of the cluster of stars localized at its center of mass.

12.7 FFT Based Particle-Mesh Methods I

Observation, the scalar potential can be obtained by convoluting a density field ρ with the appropriate Green's function G.
In electrostatics we have a charge distribution from which the the electric field is expressed as the negative gradient of the electric potential and with Gauss's law we obtain:

$$\mathbf{E} = -\nabla\phi \tag{1}$$

$$\nabla \cdot \mathbf{E} = -\frac{\rho}{\varepsilon_0} \tag{2}$$

Now we combine this to:

$$\nabla^2\phi = -\frac{\rho}{\varepsilon_0} \tag{3}$$

and obtain Poisson's equation.

We can now identify $G(\mathbf{x}, \mathbf{x}')$ as Green's function which gives the response of the system to the point charge q and deduce further more that the Green's function of this operator of the form: $-\frac{\varepsilon_0}{q}\nabla^2$ and hence we can write for $\phi(\mathbf{x})$:

$$\phi(\mathbf{x}) = \int_{x'} G(\mathbf{x}, \mathbf{x}')\rho(\mathbf{x}')d\mathbf{x}'. \tag{4}$$

This concludes the physical modelling of the problem (not considering in detail boundary conditions). The next steps will be:

- $\mathcal{O}(n^2) \rightarrow \mathcal{O}(n \log n) \rightarrow$ convolution in Fourier space
- discretisation → CIC
- (parallelisaiton)

12.8 The Convolution Theorem I

The famous convolution theorem tells us that convolutions can be carried out (very efficiently) via point-wise multiplication in the Fourier domain.

Theorem

The convolution $x \otimes y$ can be computed by taking the inverse Fourier transform of $\widehat{x} \circ \widehat{y}$, where $\widehat{x}$ and $\widehat{y}$ are the Fourier transforms of x and y, respectively, and $\circ$ denotes the component-wise product also known as Hadamard product.

For our discussion we have a fast way to evaluate:

$$\phi(\mathbf{x}) = G(\mathbf{x}, \mathbf{x}') \otimes \rho(\mathbf{x}')t.$$

```
1: procedure 3DFieldCalculation(In: $\rho(\mathbf{x})$, $G(\mathbf{x})$, Out: $\mathbf{E}(\mathbf{x})$)
2: Create 3D rectangular grid which contains all particles
3: Interpolate the charge $q$ of each macro-particle to nearby mesh points to obtain $\rho(i,j,k)$
4: FFT $\rho(i,j,k)$ and $G(i,j,k)$ to obtain $\widehat{\rho}(i,j,k)$ and $\widehat{G}(i,j,k)$
5: Determine $\widehat{\phi}(i,j,k) = \widehat{\rho}(i,j,k) \cdot \widehat{G}(i,j,k)$
6: Use FFT$^{-1}$ of $\widehat{\phi}(i,j,k)$ to obtain $\phi(i,j,k)$
7: Compute $\mathbf{E}(i,j,k) = -\nabla\phi(i,j,k)$
8: Interpolate $\mathbf{E}(\mathbf{x})$ at the particle positions from $\mathbf{E}(i,j,k)$
9: end procedure
```

Question: which of the steps are parallel ?

Disorder induced Heating I

What is missing ?

Disorder induced Heating I

The Vlasov equation that describe self-consistent evolution of distribution function $f(\mathbf{x}, \mathbf{v}, t) \in \mathbb{R}^{3\times 3\times 1}$

$$\frac{\partial f_s}{\partial t} + \nabla_{\mathbf{x}} \cdot (\mathbf{v} f_s) + \nabla_{\mathbf{v}} \cdot (\mathbf{F}_s f_s) = 0$$

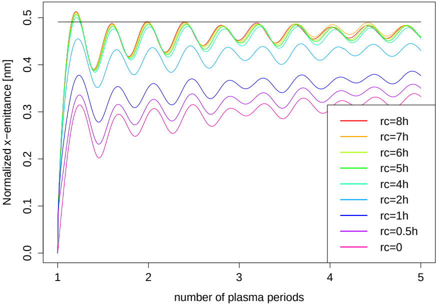
where $\mathbf{F}_s$ comes from the gradient of a scalar field.

Disorder induced Heating I

The Vlasov-Boltzmann equation that describe self-consistent evolution of distribution function $f(\mathbf{x}, \mathbf{v}, t) \in \mathbb{R}^{3\times3\times1}$

$$\frac{\partial f_s}{\partial t} + \nabla_{\mathbf{x}} \cdot (\mathbf{v} f_s) + \nabla_{\mathbf{v}} \cdot (\mathbf{F}_s f_s) = \left(\frac{\partial f_s}{\partial t}\right)_c$$

where $\mathbf{F}_s$ comes from the gradient of a scalar field.



Disorder induced Heating I

B, Ulmer MSc http://amas.web.psi.ch/people/aadelmann/ETH-Accel-Lecture-1/projectscompleted/cse/thesisBUlmer.pdf

The heating associated with the relaxation of a random, near-zero temperature distribution of charges is well known to the ultracold neutral plasma (UNP) community. In such systems, a cold gas is laser ionized, and after a time on the order of the $\tau = 2\pi\omega_p^{-1}$. In the traditional plasma physics terminology, this effect is referred to as disorder induced heating (DIH).

- the simulation parameters are chosen to correspond to a realistic beam close to the electron gun
- a spherical coasting beam of radius $R = 17.74\,\mu\text{m}$ carrying 25 fC) ($N_e = 156055$)
- simulation duration 5 plasma periods
- constant linear focusing, periodic BC

Disorder induced Heating II

B, Ulmer MSc http://amas.web.psi.ch/people/aadelmann/ETH-Accel-Lecture-1/projectscompleted/cse/thesisBUlmer.pdf

$$\omega_p = \sqrt{\frac{Ne^2}{m_0\varepsilon_0}} \approx 1.45 \times 10^{11}\frac{1}{s} \ . \tag{5}$$

Hence, one plasma period has a duration of

$$\tau = \frac{2\pi}{\omega_p} \approx 43\,\mathrm{fs} \tag{6}$$

and 5 plasma periods correspond to a total simulation time of

$$T_{\mathrm{end}} = 215\,\mathrm{fs} \ . \tag{7}$$

- Since the particle collisions can lead to very large momenta between two colliding particles a relatively small time step of $\Delta t = T_{\mathrm{end}}/1000 = 2.15\,\mathrm{fs}$ has been chosen.

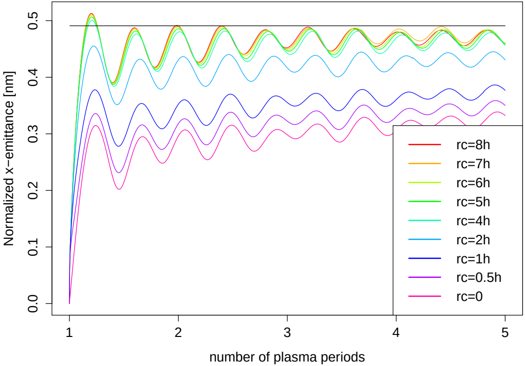
Disorder induced Heating III

B, Ulmer MSc http://amas.web.psi.ch/people/aadelmann/ETH-Accel-Lecture-1/projectscompleted/cse/thesisBUlmer.pdf

- The $\mathrm{P}^3\mathrm{M}$ implementation uses an interaction splitting based on Gaussian screening.
- The computational mesh width is chosen to be $h = 0.39\,\mu\mathrm{m}$. In order to study the influence of the particle-particle part of the solver corresponding to the Coulomb collisions between particles, the cutoff r_c has been varied from $r_c = 0\,\mu\mathrm{m}$ to $r_c = 3.125\,\mu\mathrm{m}$. The splitting parameter α was chosen to be $\alpha = 2/r_{cut}$.

Disorder induced Heating IV

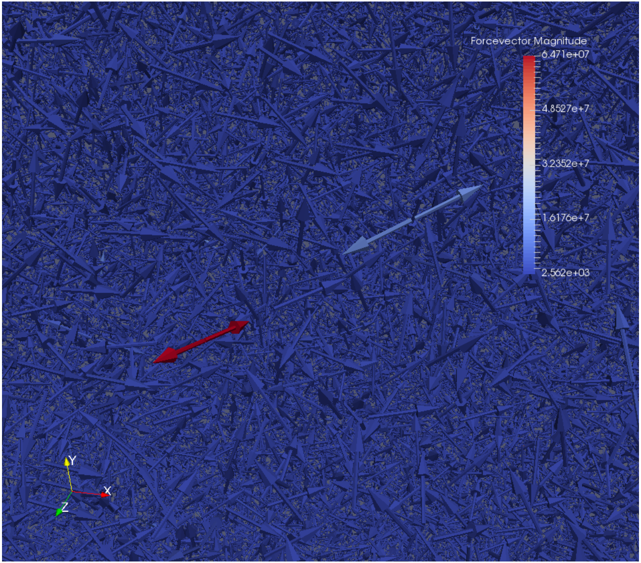
B, Ulmer MSc http://amas.web.psi.ch/people/aadelmann/ETH-Accel-Lecture-1/projectscompleted/cse/thesisBUlmer.pdf



Emittance growth for the disorder induced heating process for various values of the cutoff radius r_c. Analytic solution $\varepsilon_x = 0.491\,\text{nm}$ for the thermal equilibrium is represented by the horizontal black line.

Disorder induced Heating V

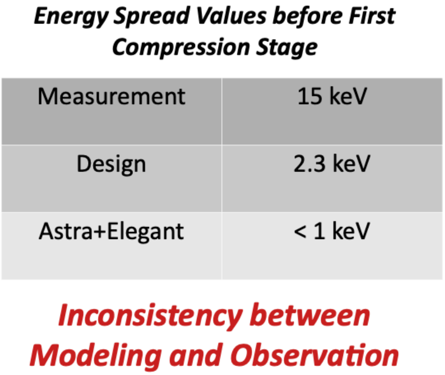
B, Ulmer MSc http://amas.web.psi.ch/people/aadelmann/ETH-Accel-Lecture-1/projectscompleted/cse/thesisBUlmer.pdf



When are collisions important? I

https://www.youtube.com/watch?v=Q0cDmBUIdrA&list=PL0b-ynhMVTd6gkXLhDv1BYZY-8qiPeG9M

- sometimes when they are not expected
- example SwissFEL



Energy Spread Values before First Compression Stage

Measurement	15 keV
Design	2.3 keV
Astra+Elegant	< 1 keV

Inconsistency between Modeling and Observation

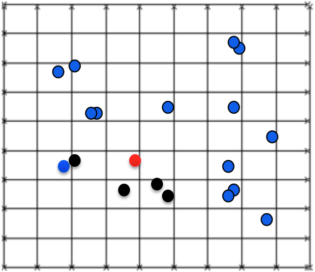
12.9 Collisions via the Particle Particle Particle Method I

Splitting into short-range and long-range

To compute the force on a particle (red), represent as a combination of

- Contributions from nearby particles (black), using (local) N-Body calculations.
- Contributions from far-away particles, using PIC (blue)

How do you do this without double-counting, or having many PIC solves ? What is the error of the resulting method for an arbitrary distribution of a finite number of particles?



P3M I

Hockney and Eastwood

$\mathrm{P}^3\mathrm{M}$ = **P**article-**P**article + **P**article-**M**esh

- high resolution from PP part
- good performance from PM part
- adjustable influence of Coulomb collisions
- PP and PM limit correspond to widely used methods
- possibility of start to end beam simulations with adjustable Coulomb interaction

P3M II

Hockney and Eastwood

The electrostatic potential $\Phi(\vec{r})$ of a system of interacting point charges $q_i(\vec{r})$ with charge distribution $\rho(\vec{r})$ is described by the Poisson Equation.

$$\vec{\nabla}^2\Phi(\vec{r}) = -\rho(\vec{r})$$

With the appropriate Green's function

$$G(\vec{r},\vec{r}') = \frac{1}{|\vec{r}-\vec{r}'|}$$

interpreted as the potential that arises due to a point charge at $\vec{r}'$, the solution for an arbitrary charge distribution is given by the convolution

$$\vec{\Phi}(\vec{r}) = \int G(\vec{r},\vec{r}')\rho(\vec{r}')d^3\vec{r}'$$

P3M III

Hockney and Eastwood

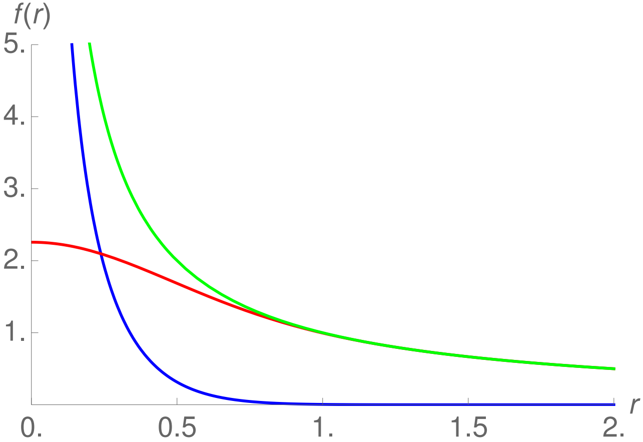
The main concept behind the P^3M algorithm is a splitting of the interaction function $G(\vec{r})$ into a **short-range** contribution $G_{pp}(\vec{r})$ and a **long-range** contribution $G_{pm}(\vec{r})$. This splitting can be done using a Gaussian screening charge distribution

$$G(r) = \frac{1}{r} = \underbrace{\frac{1 - erf(\alpha r)}{r}}_{G_{PP}} + \underbrace{\frac{erf(\alpha r)}{r}}_{G_{PM}}$$

The long-range interactions are computed on a grid using a particle-mesh solver and the short range interactions are evaluated using an N-body solver with appropriate cutoff.

P3M IV

Hockney and Eastwood



Greens function splitting for $r_c = 1$ and $\alpha = 2$: $G(r)$ in green, $G_{PM}(r)$ in red and $G_{PP}(r)$ in blue

- C. K. Birdsall, A. B. Langdon, *Plasma physics via computer simulation*, Institute of Physics, Bristol (1991) p. 359.
- RW Hockney and JW Eastwood. *Computer Simulation Using Particles*. Adam Hilger, Philadelphia, 1988.
- William B. Bateson, Dennis W. Hewett, *Grid and particle hydrodynamics: beyond hydrodynamics via fluid element particle-in-cell.* J. Comput. Phys. 144 (1998), no. 2, 358–378.
- G.-H. Cottet, P.-A. Raviart, *Particle methods for the one-dimensional Vlasov-Poisson equations.* SIAM J. Numer. Anal. 21 (1984), no. 1, 52–76.
- H. D. Victory, Jr., Edward J. Allen, *The convergence theory of particle-in-cell methods for multidimensional Vlasov-Poisson systems.* SIAM J. Numer. Anal. 28 (1991), no. 5, 1207–1241.
- H. Neunzert, J. Wick, *The convergence of simulation methods in plasma physics.* Mathematical methods of plasmaphysics

(Oberwolfach, 1979), 271–286, Methoden Verfahren Math. Phys., 20, Lang, Frankfurt, 1980.

Jacobs, G. B.; Hesthaven, J. S. High-order nodal discontinuous Galerkin particle-in-cell method on unstructured grids. J. Comput. Phys. 214 (2006), no. 1, 96–121.