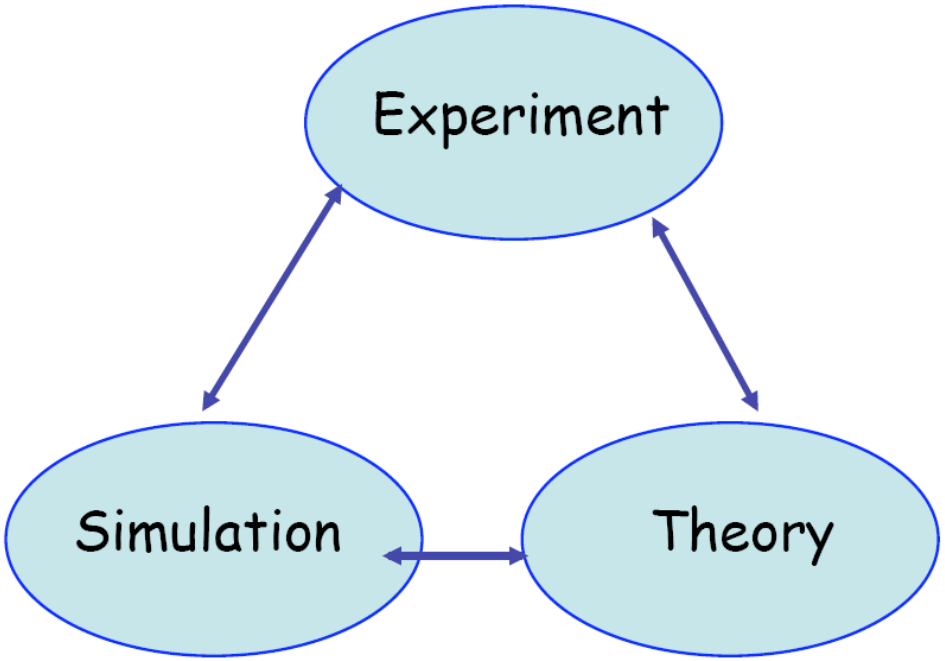
Simulation Is an Important Tool in Scientific Research

- Computer simulations are carried out to
  - Understand the consequences of fundamental physical laws
  - Help interpret experiments
    - Can provide detailed information which is difficult to measure
  - Design and predict new experiments



Numerical simulation is not theory!

Approximate theoretical models important, even if analytical solution does not exist

What is numerical simulation?

- The numerical simulation is: writing a program and using the computer to perform a numerical experiment, which can show you the time evolution of the nonlinear phenomena.
- Linear problems: analytical solutions are available for most linear problems, so numerical simulation is not necessary. Linear theory provides theoretical basis for validating the simulation results.
- Unfortunately, most plasma-based systems are highly nonlinear. So the numerical simulation could provide a more solid theoretical basis for studying a highly nonlinear and highly relativistic system.

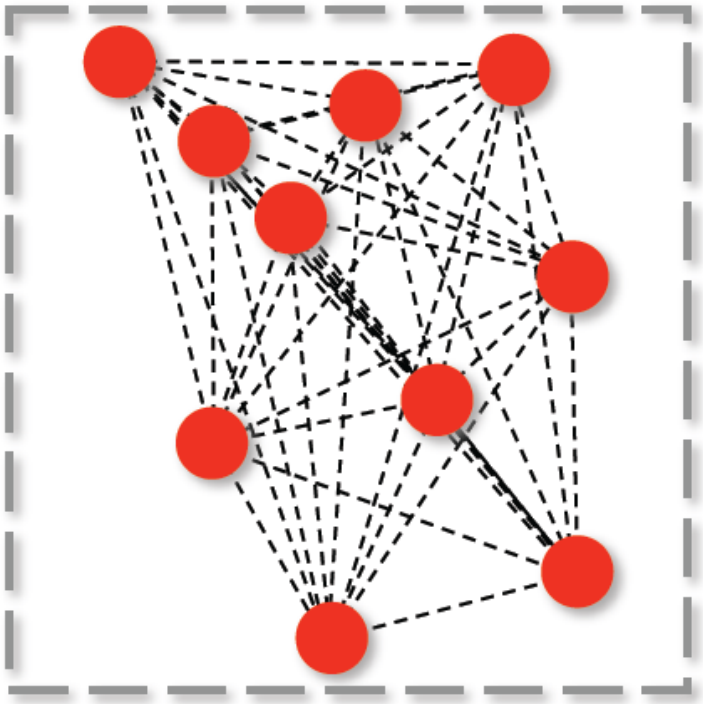
Numerical simulation methods

- **Plasma dynamics simulations**
  - Kinetic methods [$\Rightarrow$ distribution function $f(\boldsymbol{r},\boldsymbol{p},t)$]
    - Particle methods
      - Test particle methods
      - Particle-particle methods (small particle number, inefficient)
      - Particle-mesh methods – PIC
      - Mesh-free methods – Tree codes
      - Molecular dynamics methods
    - Solution of equation for distribution function
      - Vlasov equation
      - Fokker-Planck equation (Vlasov-Boltzmann equation)
  - Fluid (hydrodynamic) methods [$\rho(\boldsymbol{r},t)$, $\boldsymbol{u}(\boldsymbol{r},t)$, $T(\boldsymbol{r},t)$]
    - Two-fluid hydrodynamics (rare)
    - One-fluid hydrodynamics (often 2 temperatures, radiation, …)
    - MHD
  - Hybrid methods - part of plasma (e.g. electrons or fast electrons) treated e.g. in kinetic way and part (e.g. ions or ions and thermal electrons) e.g. via fluid methods

Particle simulation

If interactions particle-particle are included for N particles $\Rightarrow$ N^2 forces

When two particles get very near to each other – large variation of velocities in small Δt $\Rightarrow$ short timestep



Such approach possible for $N \leq 1000$

For illustration 10^8 particles, 10 Tflop/s,

1 timestep – 3 hours

10^4 timesteps – 3 years

Larger systems

- ***Tree codes*** – interaction particle-cluster
- **Particle-In-Cell** - interaction particle-grid

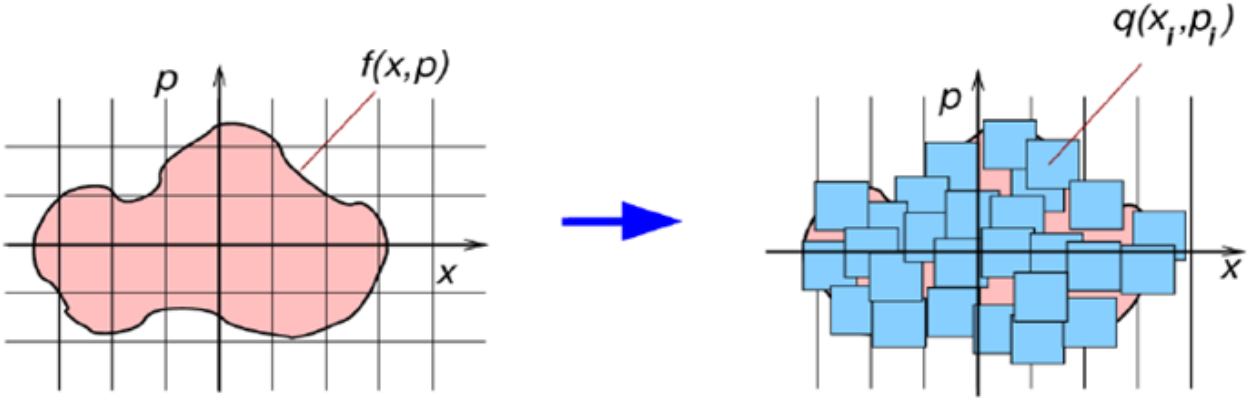
PIC is very widely applied method

Aim is not accurate description of binary interactions, but oppositely, aim is to obtain equivalent of Vlasov equation. Thus, macroscopic electromagnetic field and motion of particles in it.

PIC - 1 timestep ~ 0.3 ms, 10^4 steps ~ 3 s

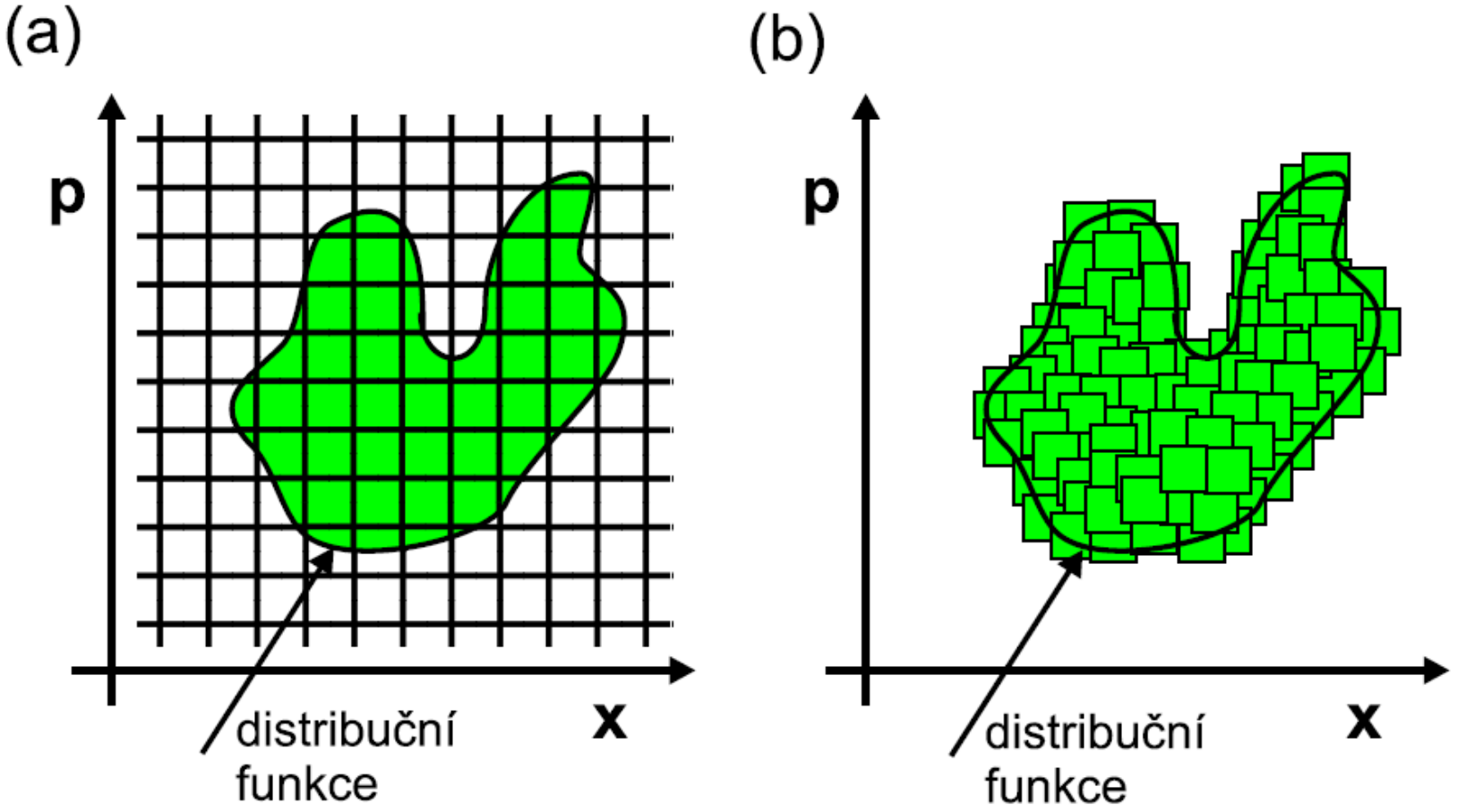
Particle-mesh technique

- Numerical mesh added to more effectively compute the forces acting on model particles.
- Force evaluation based on continuum representation of electromagnetic fields calculated from charge and current densities (**Particle-in-Cell - PIC**)
- The number of floating point operations typically scales as $\alpha N + \beta N_g \ln(N_g) + \gamma N_g$ (N, N_g - number of macroparticles and grid points).



- Kinetic equation solved via sampling
- Macroparticles are clouds of particles (particle dimensions equal to grid cells dimensions, typically comparable to Debye length λ_D)
- Macroparticle has δ function distribution in velocity
- Using mesh practically filters out collisions (strong near binary correlations), thus collisionless Vlasov kinetic equation is solved by PIC codes

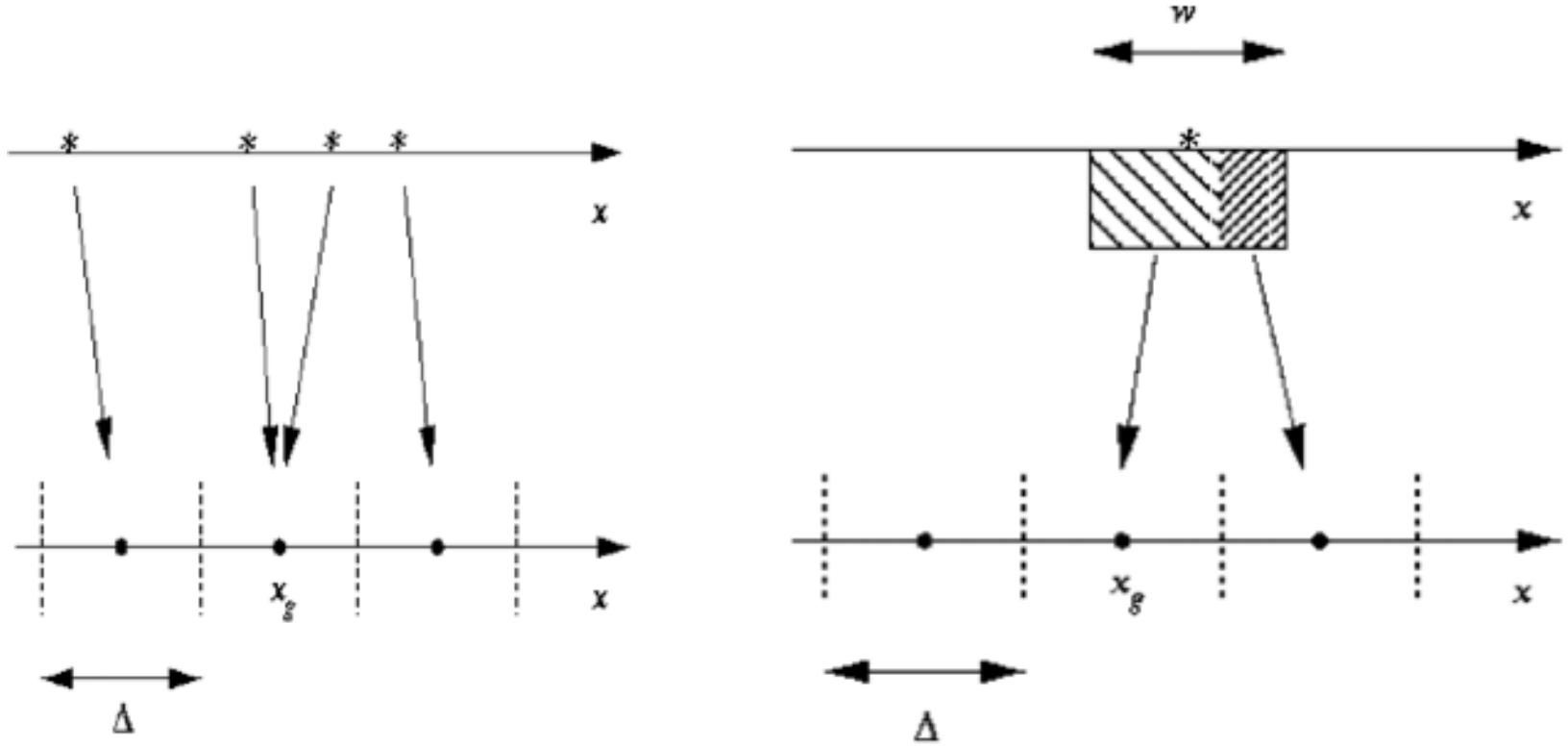
Numerical solving of Vlasov equation – much slower, usually done in less dimensions (often only in 1D), noise-free, favorable for distribution tails
PIC method is in fact solving of Vlasov equation via sampling



PIC solves Vlasov equation only in places, where particles are present;
it is parallelized easily

Particle assignment and force interpolation

Nearest point assignment (1.order) Linear interpol. (cloud in cell 2. order)



Maxwell's equations

N cells (length 1) – area length $L=N$, particle number M, length $\delta=1$,

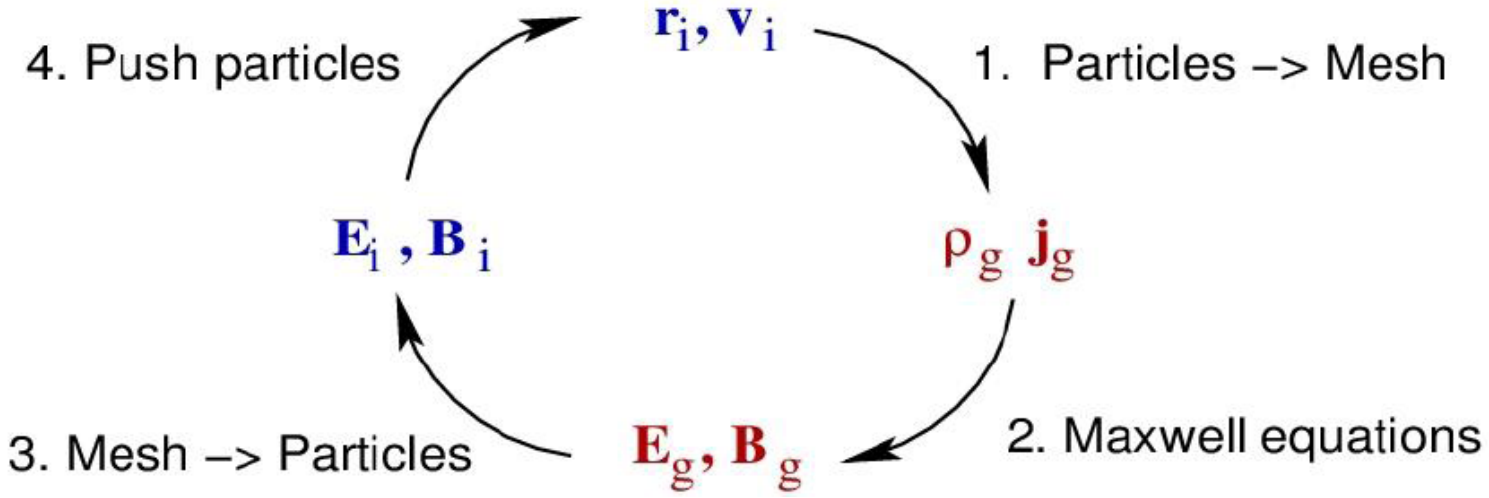
particle number per cell $n_{av}=M/L$

Cloud-in-cell – part of charge in $x < (x_i+x_{i+1})/2$ assigned to point i

$$\Delta\rho_i = q(1-\Delta x) \qquad \Delta\rho_{i+1} = q\Delta x$$

Particle-in-Cell code

- The basic cycle of a PIC code

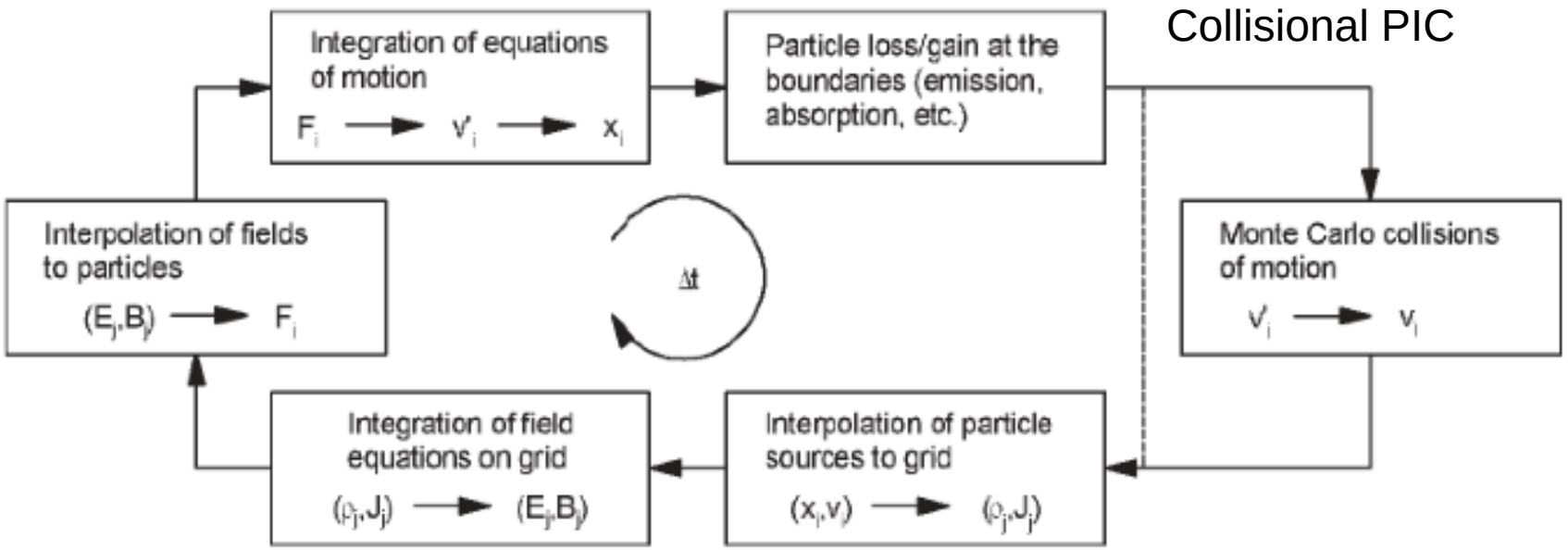


- PIC codes are suitable for uncorrelated (collisionless) or weakly correlated (weakly collisional systems)
- Such systems are dominated by collective modes due to long range Coulomb interaction, the range of wavelengths of these collective modes is bounded at the lower end by the Debye length λ_D due to efficient Landau damping of shorter modes, grid has thus little impact on collective modes
- Most of widely use explicit temporal differencing, the time step is shorter or comparable with inverse plasma frequency ω_p^{-1}.
- Implicit differencing enables longer time step but coding is more difficult.

Categories of PIC models

- **Field components:** Electrostatic, Magnetostatic, Electromagnetic
- **Geometry:** 1/2/3D 1/2/3V, boost frame
- **Equation of motion:** Relativistic or Non-relativistic
- **Boundary conditions:** Absorbing, reflecting or periodic for both particles and fields
- **Binary collisions** may be added (e.g. via Monte Carlo algorithm)

Flow scheme for a typical PIC-MCC code



Electromagnetic code – Maxwell's equations – charges and currents

In seminars - electrostatic code ES1 $\rightarrow$ Poisson equation

$$\frac{\partial E}{\partial x} = \frac{\rho}{\varepsilon_0} \qquad E(i+1) = E(i) + \frac{\rho(i) + \rho(i+1)}{2\varepsilon_0}$$

kód ES1-Fourier transform $ikE_k = \rho_k / \varepsilon_0$

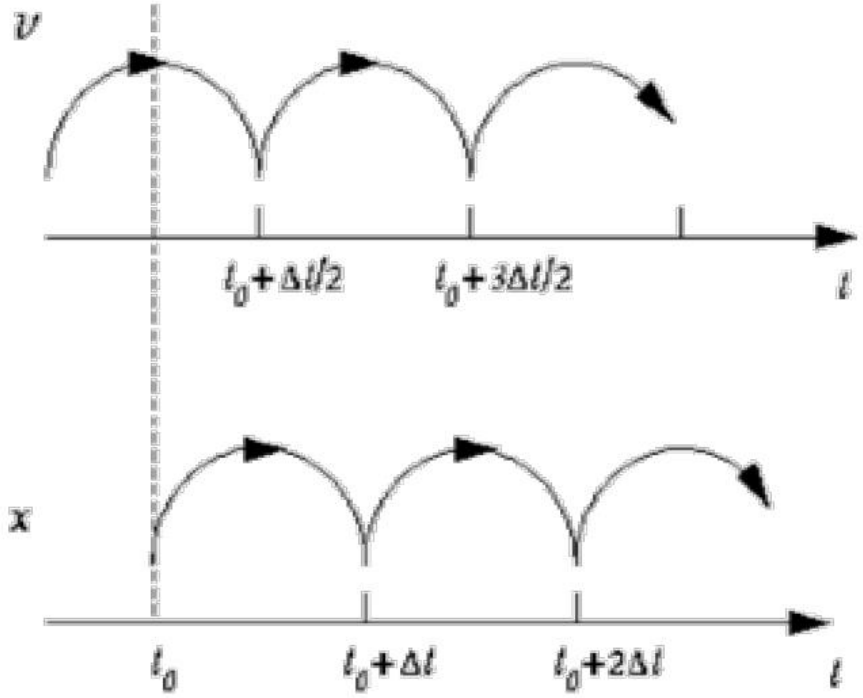
Force acting in $x_i+\Delta x$ $F = qE_i\left(1-\Delta x\right) + qE_{i+1}\Delta x$

Leap-frog method often used for particle motion

Equation of motion

$$\mathrm{v}^{n+1/2} = \mathrm{v}^{n-1/2} + F^n \Delta t / m$$

$$x^{n+1} = x^n + \mathrm{v}^{n+1/2}\Delta t$$



Normalization (multiplying by constants takes time)

$$t' = \omega_{pe} t \qquad x' = x / \delta$$

$$\mathrm{v}' = \frac{\mathrm{v}}{\omega_{\mathrm{pe}} \delta} \qquad E' = \frac{\varepsilon_0 E}{e N_{av}} = \frac{\varepsilon_0 E}{e n_{av} \delta}$$

and equations transform into form

$$\frac{\mathrm{d}\, x'}{\mathrm{d}\, t'} = \mathrm{v}' \qquad \frac{\mathrm{d}\, \mathrm{v}'}{\mathrm{d}\, t'} = -E' \qquad \frac{\mathrm{d}\, E'}{\mathrm{d}\, x'} = 1 - \frac{N}{N_{av}}$$

simplest case

motion of electrons solved, ions as homogenous neutralizing background

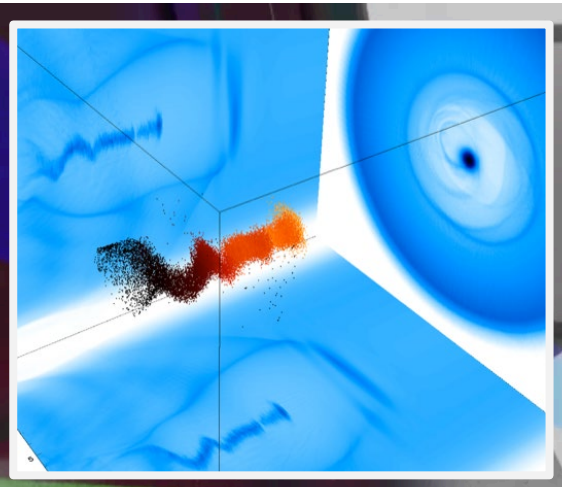
Check – total energy conservation

$$\epsilon = \sum_{j=1}^{M} \frac{m}{2} \mathrm{v}_{\mathrm{j}}^2 + \sum_{i=1}^{N} \frac{\varepsilon_0 E_i^2}{2} \delta = \frac{m \omega_{pe}^2 \delta^2}{2} \left[\sum_{j=1}^{M} \mathrm{v}'^2_{\mathrm{j}} + N_{av} \sum_{i=1}^{N} E'^2_{\mathrm{i}} \right]$$

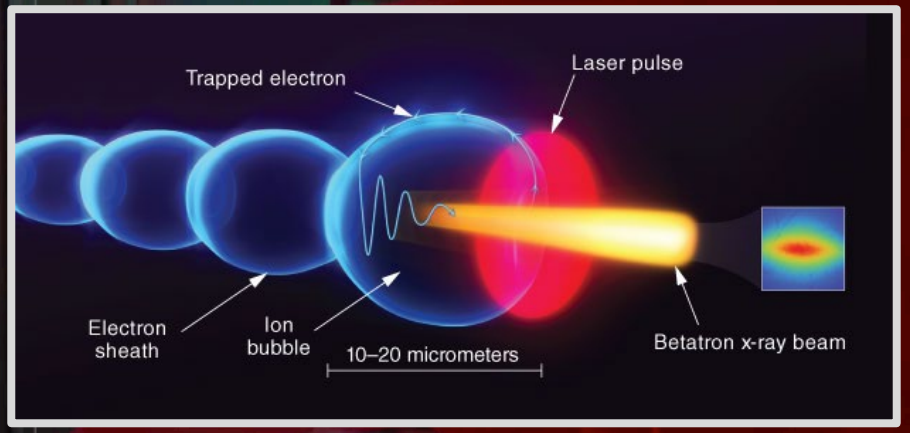
ES1 – 1D electrostatic non-relativistic, today for education,

Moving ions, possible $B_0 \perp x$, periodic boundaries *E(N+1)=E(1)*

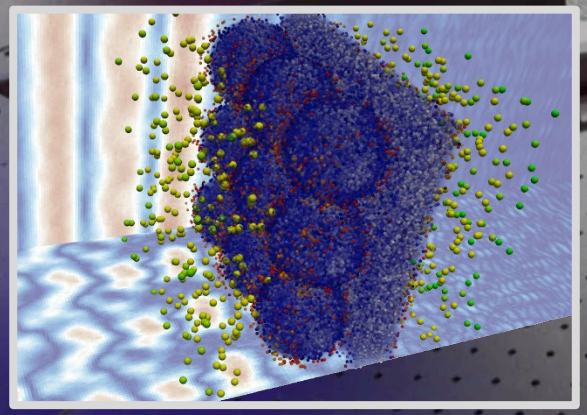
PIC simulations examples



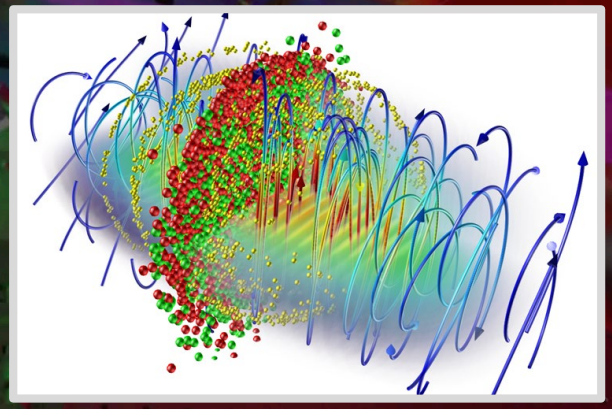
electron beam acceleration



generation of energetic radiation



ion beam acceleration



electron-positron pair production