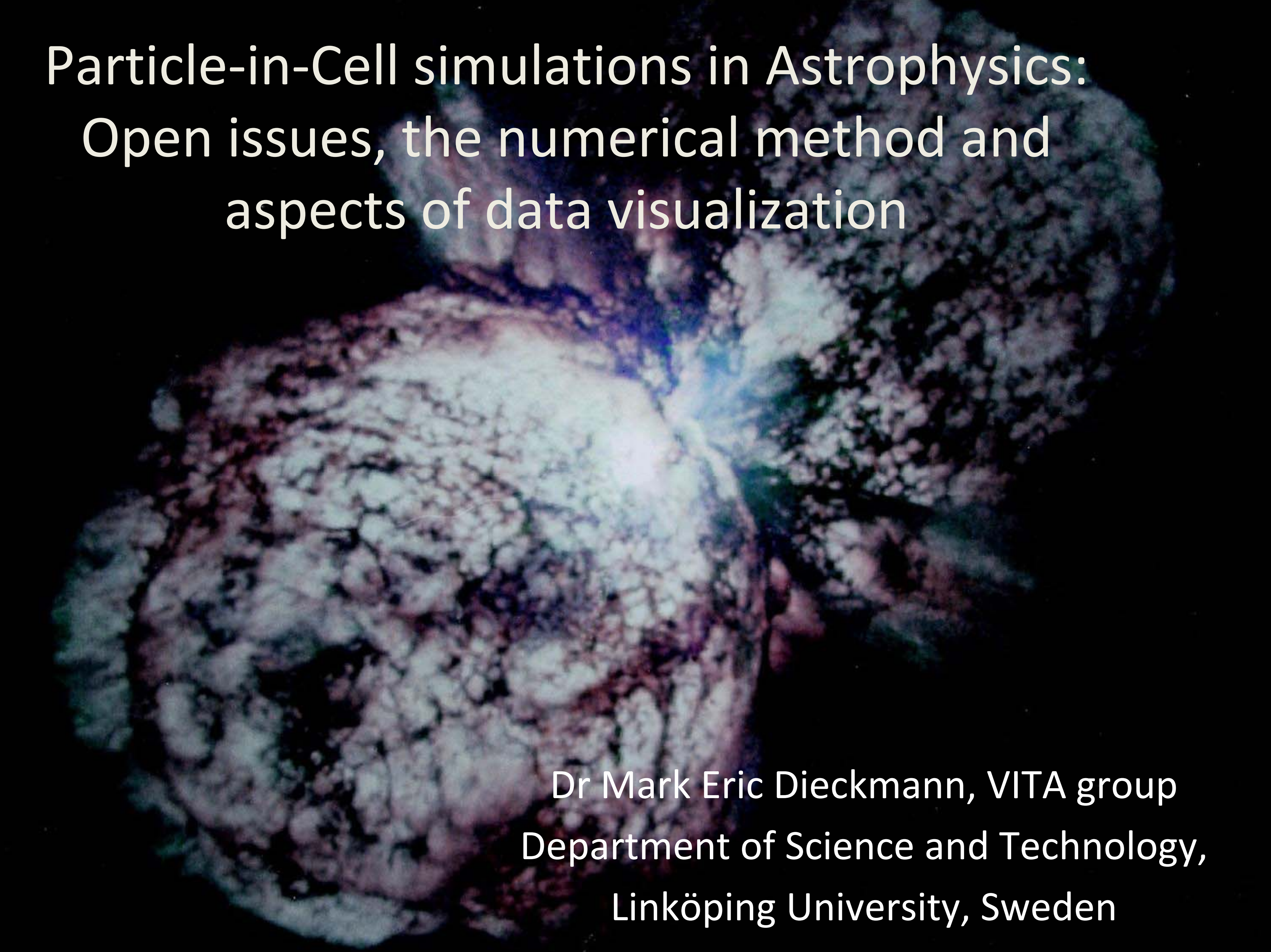
Particle-in-Cell simulations in Astrophysics: Open issues, the numerical method and aspects of data visualization



Dr Mark Eric Dieckmann, VITA group
Department of Science and Technology,
Linköping University, Sweden

Outline

- The Particle-In-Cell simulation method
- Aspects of data Visualization and simulation scales
- Presentation of ongoing research
- Conclusions

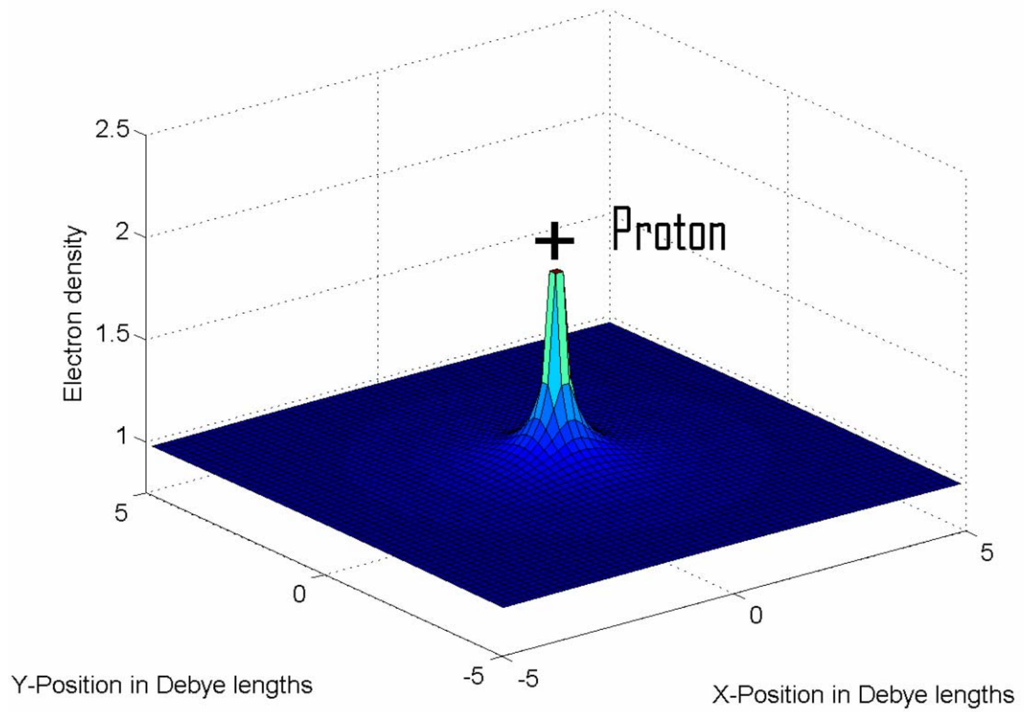
Phase Space Fluid

Fluid: No individual particles

A charged particle brought into a plasma causes a redistribution of the charge $\rho(x,t)$ (Fluid)

The particle is shielded over the distance v_t / ω_p (Thermal speed over plasma frequency): Debye length λ_D

No binary interactions are important for the plasma on scales $\lambda > \lambda_D$



- Plasma interaction: macroscopic electromagnetic fields.
- Phase space Fluid: Charge density $\rho(x,t) = q \int f(x,v,t)\, dv$

The Vlasov Equation

Consider a single species of nonrelativistic electrons.

Many electrons per Debye sphere $\Rightarrow f_e(\vec{x}, \vec{v}, t)$.

The moments are $\rho_e(\vec{x}, t) = -e \int f_e(\vec{x}, \vec{v}, t) d\vec{v}$ (Charge)

$\vec{J}_e = -e \int \vec{v} f_e(\vec{x}, \vec{v}, t) d\vec{v}$ (Current)

The (collision-less) electron plasma is then evolved in time through:

The Vlasov equation

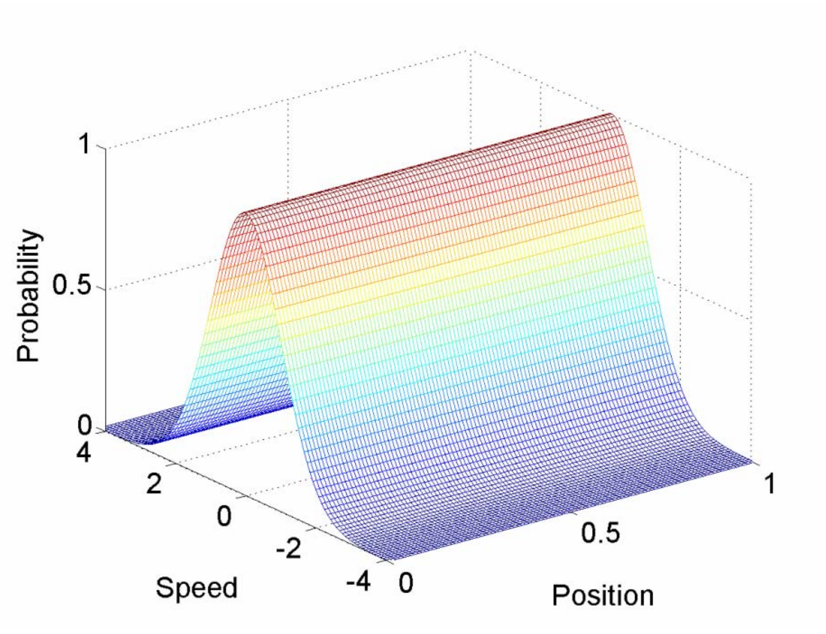
$$\frac{\partial f_e}{\partial t} + \vec{v}\frac{\partial f_e}{d\vec{x}} - \frac{e}{m_e}\left(\vec{E} + \vec{v} \times \vec{B}\right)\frac{\partial f_e}{d\vec{v}} = 0$$

The Maxwell's equations

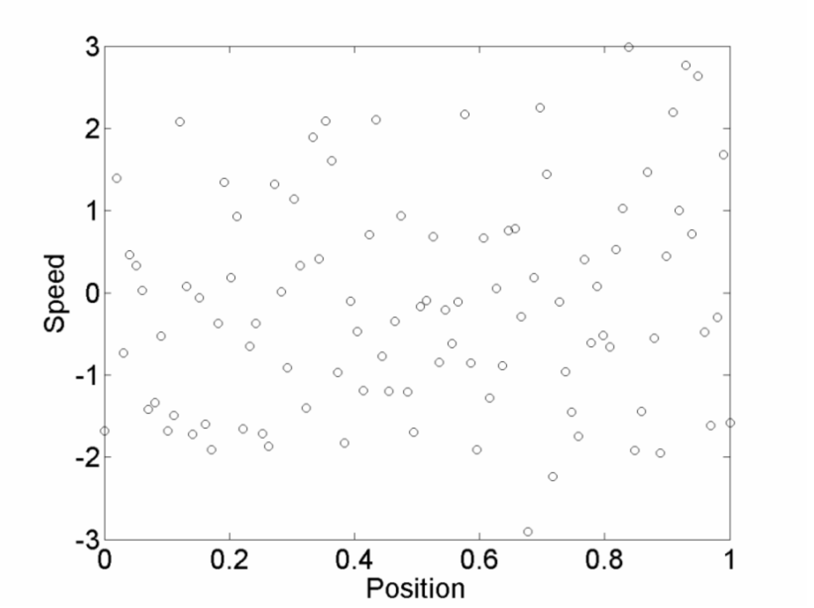
$$\nabla \times \vec{B} = \mu_0 \vec{J} + \mu_0 \epsilon_0 \frac{\partial \vec{E}}{\partial t}, \;\; \nabla \cdot \vec{B} = 0,$$

$$\nabla \times \vec{E} = -\frac{\partial \vec{B}}{\partial t}, \;\; \nabla \cdot \vec{E} = \rho_e / \epsilon_0$$

The Particle-In-Cell Method



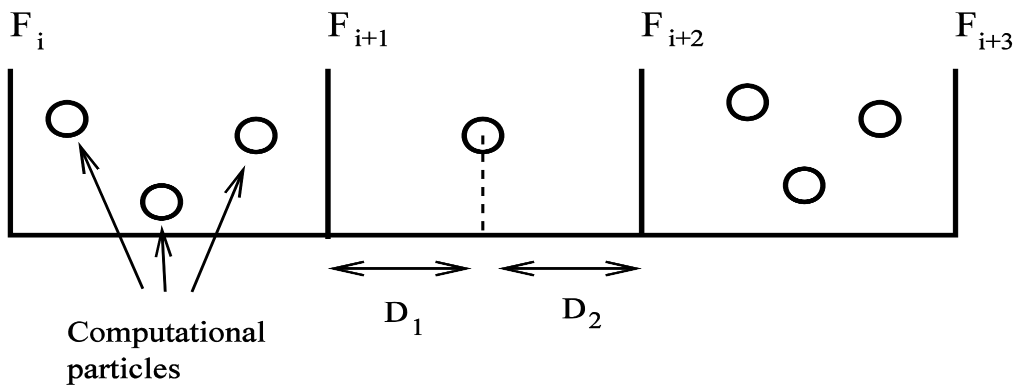
→



The PIC method approximates $f_e\,(x,v,t)$ by computational particles.

$$\frac{\partial f_e}{\partial t}+\vec{v}\frac{\partial f_e}{d\vec{x}}-\frac{e}{m_e}\left(\vec{E}+\vec{v}\times\vec{B}\right)\frac{\partial f_e}{d\vec{v}}=0\Rightarrow\frac{dv_j}{dt}=\frac{-e}{m_e}\left(\vec{E}[x_j]+\vec{v}_j\times\vec{B}[x_j]\right)$$

The Numerical Scheme



- Each computational electron j carries a micro-current $j_j \sim q_j\, v_j$
- Distribute micro-current onto neighboring grid nodes using D_1, D_2
- Sum over all computational electrons j : Get macroscopic current J at grid nodes F_i
- Update E and B with Maxwell's equations on grid nodes F_i
- Interpolate E and B from grid nodes F_i to computational electrons and update their velocity and position.

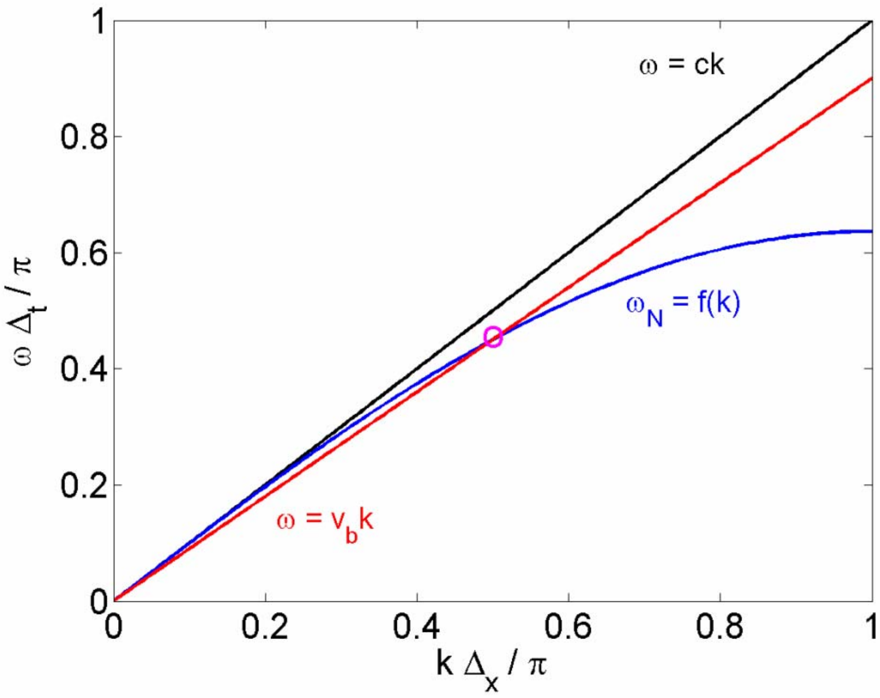
Consequence of finite grid (1)

Light waves at high wavenumbers *k* move through the plasma with speed of light *c*.

However: Result is obtained from the differential equations.

PIC codes solve Maxwell's equations on a grid → numerical effects close to scales comparable to the cell size Δx and to the time step Δt

For illustration: $\Delta x / \Delta t = c$



Black: True light mode.
Blue: Numerical light mode.
Red: Beam mode for $v_b = 0.9c$.

Purple circle: Plasma (finite grid) instability?

k > k(*purple circle*): Numerical Cherenkov?

Consequence of finite grid (2)

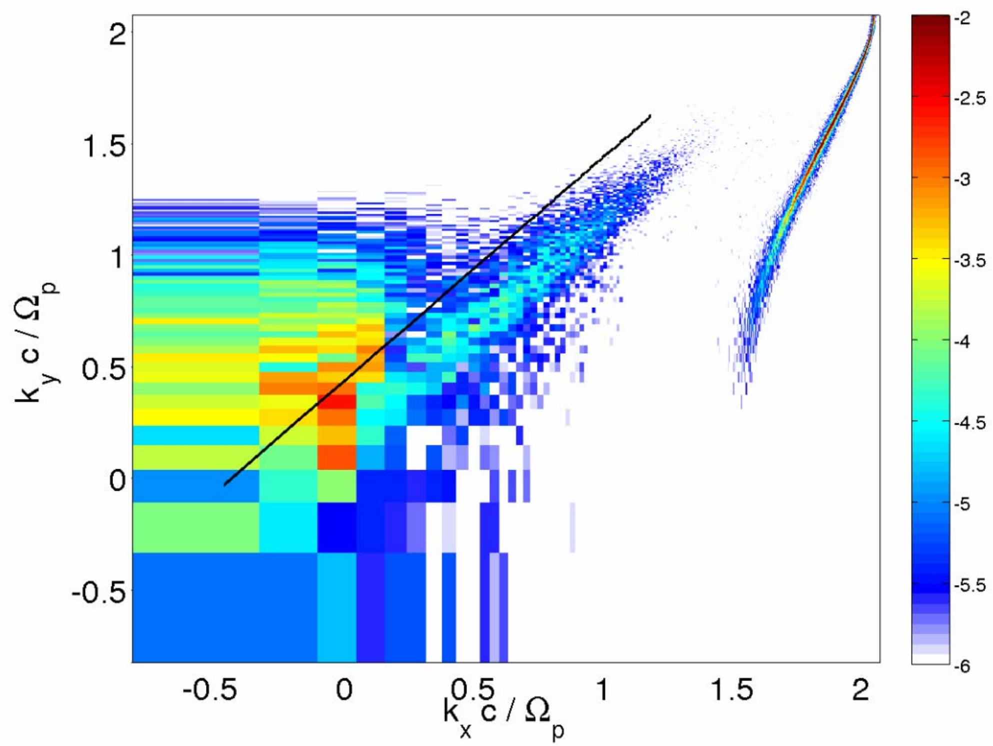
Simulation test: Beam instability in unmagnetized plasma

Bulk plasma: Protons and electrons, nonrelativistic temperature, no mean speed.

Beam plasma: Protons and electrons, nonrelativistic temperature, beam Lorentz factor 4, 0.1 times the bulk density, beam direction: *x*

Beam modes "superluminal" within a small angular interval around k_x-*axis* and at high wavenumbers.

We expect finite grid instability (sharp resonance) at high k_x and low k_y



(Note: k-axes are 10-logarithmic.)
Observation: "Unphysical wave modes" grow obliquely to beam velocity vector.

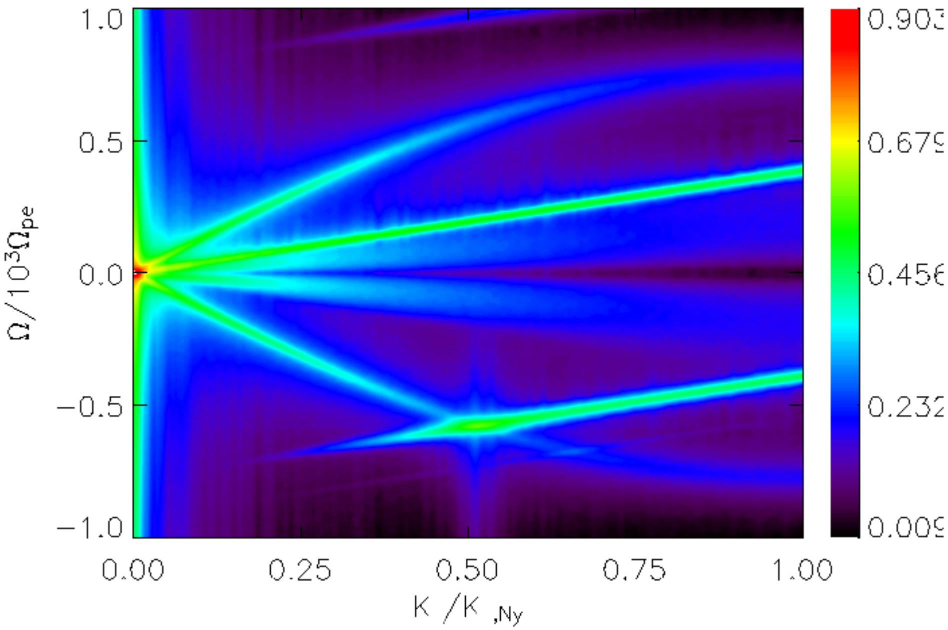
Consequence of finite grid (3)

- 1D PIC simulation along *x*, beam moves with the speed v_b obliquely to the box in *x,y* plane. Projected speed $v_{bx} < v_b$

- Oblique propagation: We excite the electromagnetic instability!

- Sample the $E_y(x, t)$ for some time.

- Get power spectrum $|E_y(k_x, \omega)|^2$



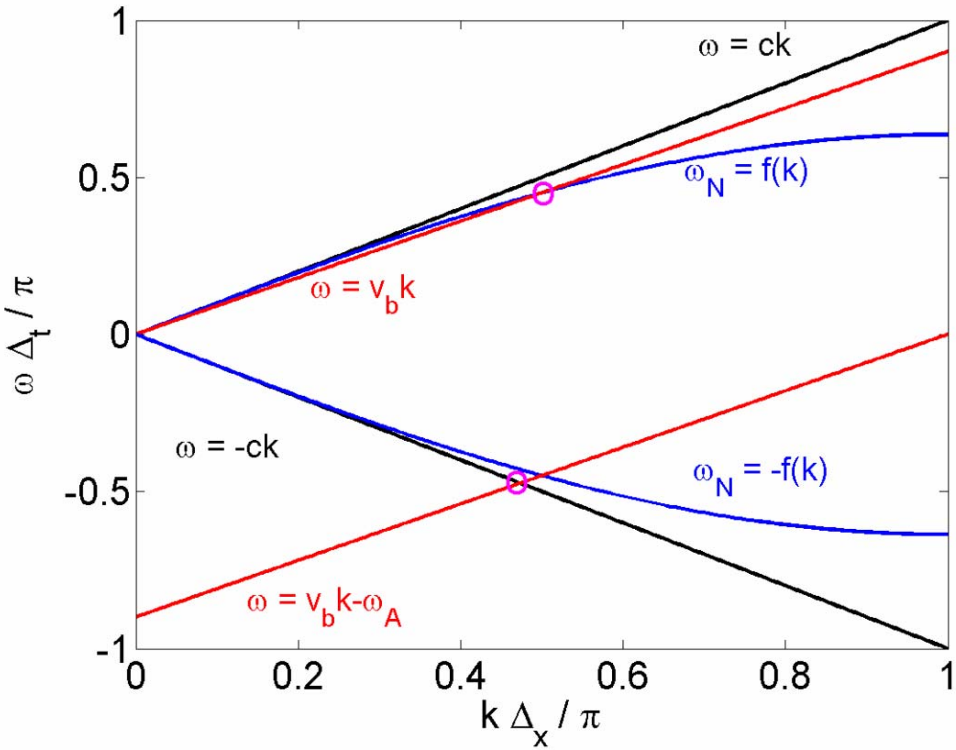
- We observe in the upper quadrant the beam dispersion relation $\omega \approx v_{bx} k$
- We observe in both quadrants the numerical light mode with decreasing phase speed modulus above $k \Delta x / \pi \approx 0.3$
- Two sidebands: The intersection of the lower with the O-mode is unstable.

Consequence of the finite grid (4)

A cold beam with speed v_b moves across a grid cell during the time $t_c = \Delta x / v_{bx}$.

The periodicity of a simulation grid is experienced by the beam with the frequency $\omega_A = 2\pi / t_c$

The beam motion across "periodic potentials" result in sidebands, separated by ω_A from the beam dispersion $\omega \approx v_{bx} k$



Simulation constraints (1)

- We need to resolve the Debye length v_e / ω_p (neglect ions for it)
- A temperature of 100 keV implies a thermal speed of $v_e \approx 0.4c$ and we need 2.5 cells per electron skin depth c / ω_p
- The proton skin depth (equal proton and electron density) is then resolved by 100 cells.
- Proton filament size is initially ≈ proton skin depth, but the filaments merge. Minimize periodicity effects → resolve 20 proton skin depths in each direction orthogonal to the flow direction, giving 2000 * 2000 cells.
- A full shock development may require 200 proton skin depths along the flow velocity vector: Grid size = 20000*2000*2000 ≈ 10^{11}cells

Simulation constraints (2)

- We need 10^{11} cells and 10^{13} computational particles.
- Each particle: 3 doubles for <u>*X*</u> and 3 for <u>*P*</u> = 48 bytes
- Physical memory: 10^2-10^3 Terabytes!
- Resolve well electron times: $\Delta t = 2\pi / \omega_p / 10$
- Resolve well shock times: $T_{sim} = 100 \ \ 1836^{1/2} \ \ 2\pi / \omega_p$
  → *40000 time steps*
- My code does on one CPU a time step with 4 millions of CPs per second.
- A time step on 1 CPU needs 10^6 - 10^7 s and the simulation 10^{11} s.
- Data write out: Sample electron and proton scales!
- → 10^3 samples of phase space, each with 10^3 Terabytes

Simulation constraints (3)

- *Let us assume the computer can do this:*
- A phase space fluid has in general 3 spatial and 3 velocity dimensions and the time dimension: 6D + Time
- We cannot visualize and understand this!
- *Consequence:* We must reduce dimensions or spatio-temporal scales (no heavy ions)
- Both can alter the physics.
- “The purpose of computing is *insight* not numbers“. (*Hamming*,1972)

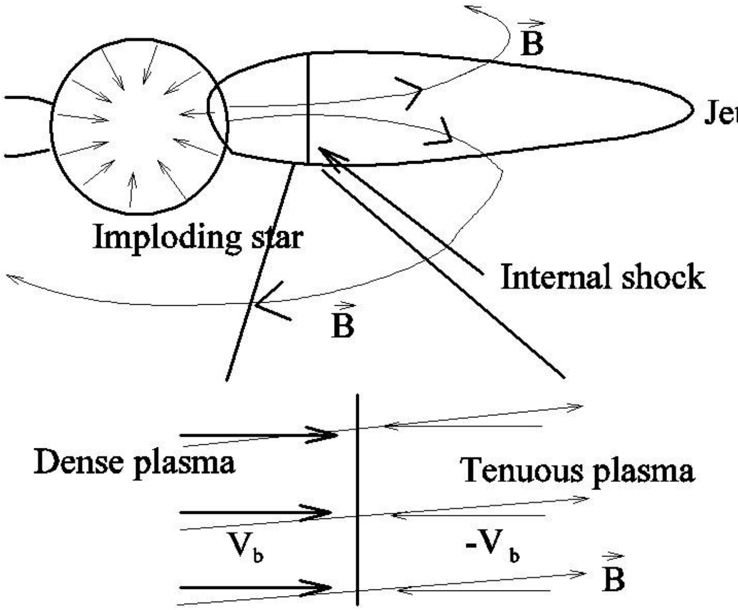
Gamma Ray Bursts

Radiation source: Accelerating charged particles

Particle acceleration: Macroscopic electromagnetic fields

Observed radiation requires: Extremely hot electrons and strong magnetic fields

Provided by: Fast shocks moving through the jet



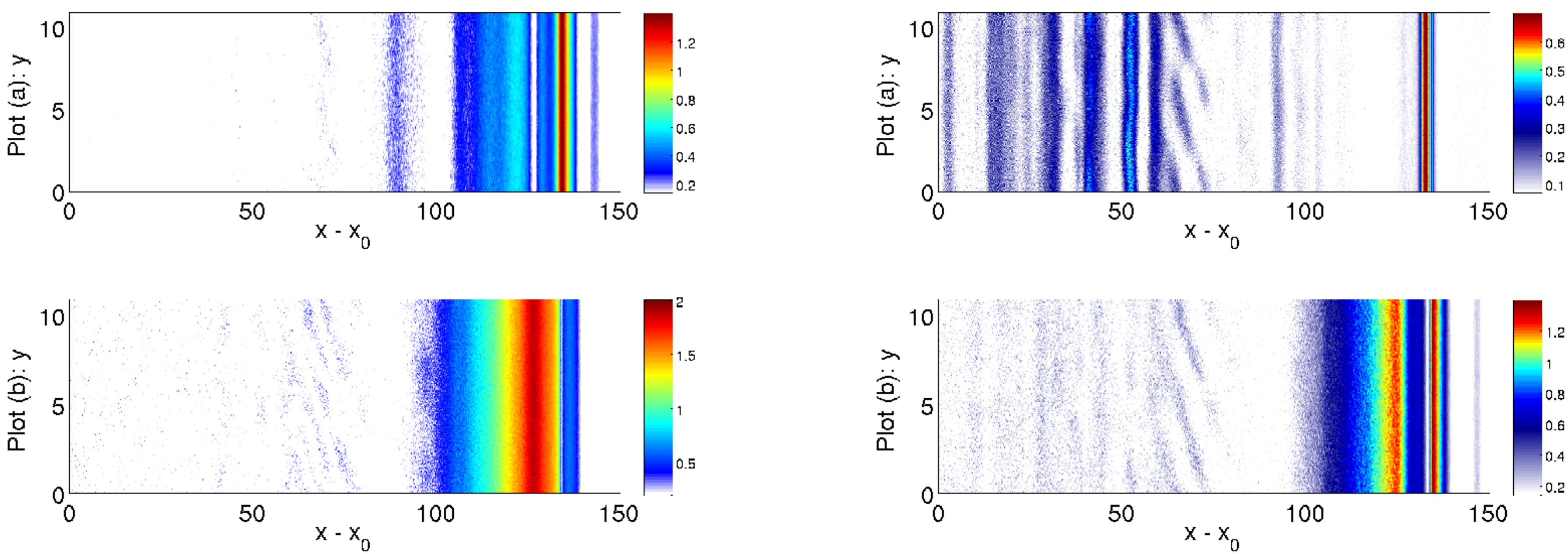
Filamentation instability does not like:

1. Asymmetric plasma clouds
2. Guiding magnetic fields
3. High plasma temperatures

Let's use all of it.

"Short" 2D simulations

•Let the plasma collide at $x=x_0$. The collision speed is 0.9c and points along x.
•The dense cloud moving to the right is 10 times denser than the thin cloud moving to the left. The clouds move at equal velocity moduli.
•We focus on the interval $x>x_0$ where the dense cloud moves into the thin cloud.
•Temperature = 100 keV, electron cyclotron frequency equals plasma frequency in dense cloud. Magnetic field tilted with 10 degrees relative to flow.
•$|B_y|$(a) and $|B_z|$ (b) in left column and E_x (a) and $|E_y + iE_z|$ (b) in right column. Simulation time $\omega_p\, t = 209$



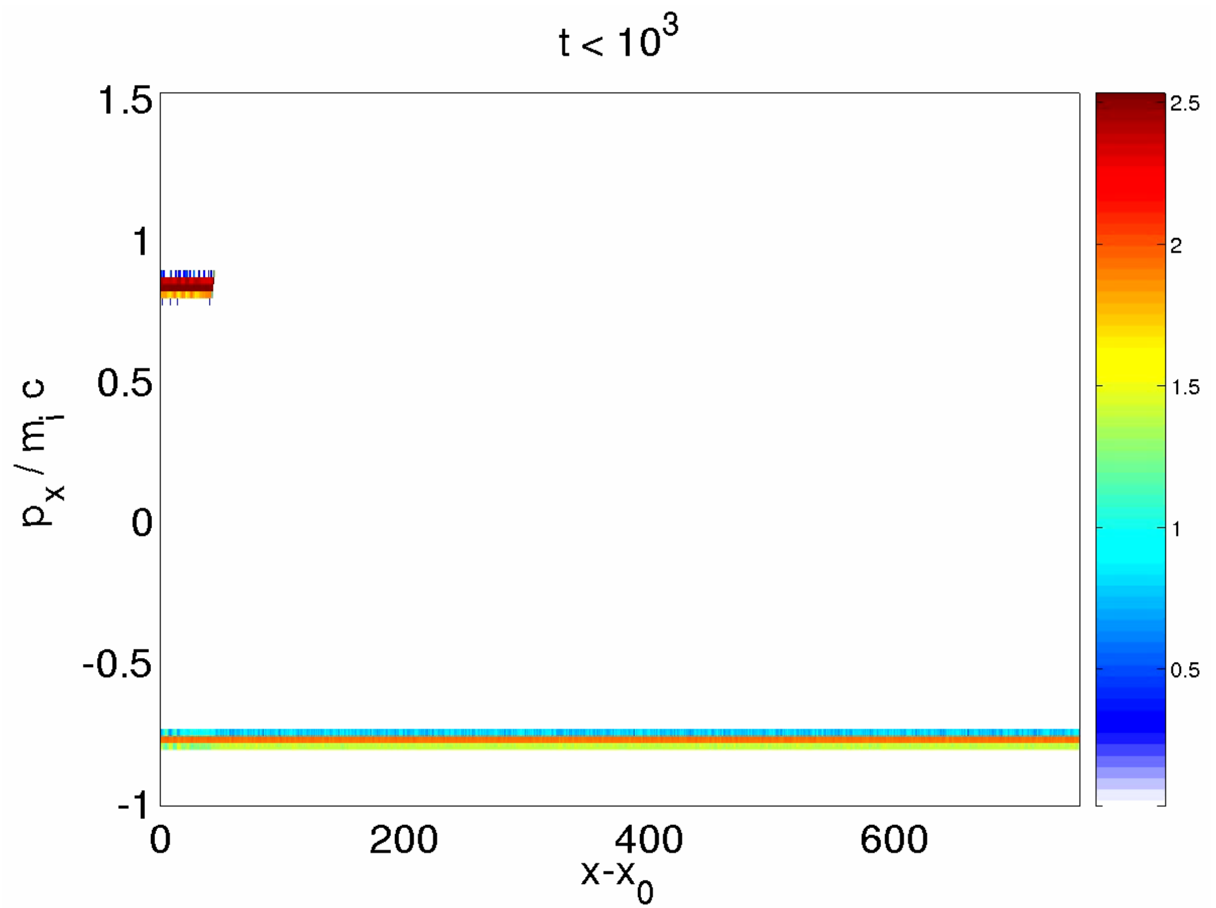
1D Simulation (1)

Here we animate in time the ion ($m_i / m_e = 400$) phase space distribution along the collision direction.

X-unit: Box-averaged electron skin depth.

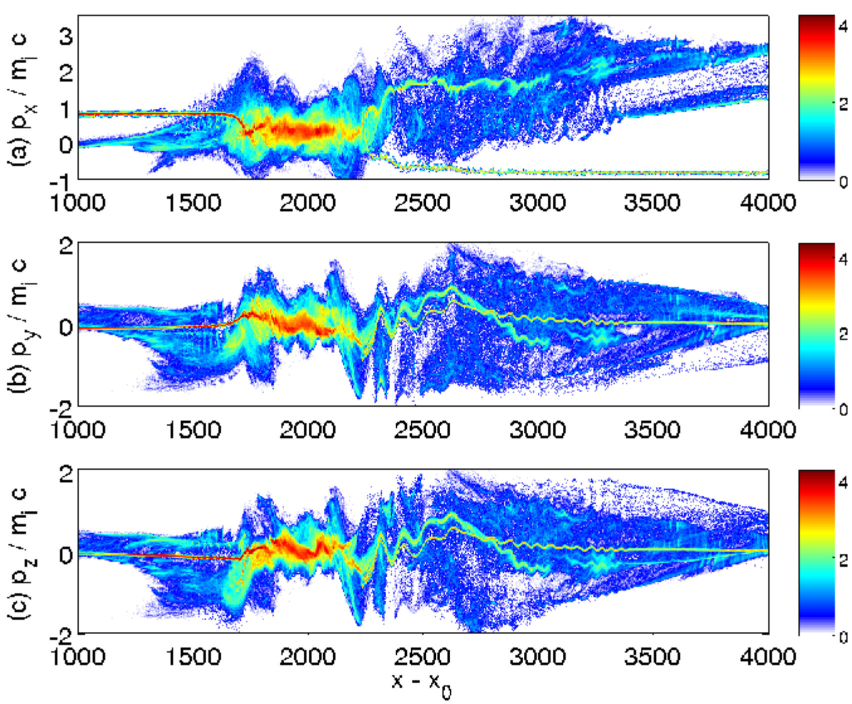
Time in inverse box-averaged electron plasma frequencies.

Colorbar: 10-logarithmic number of computational particles.

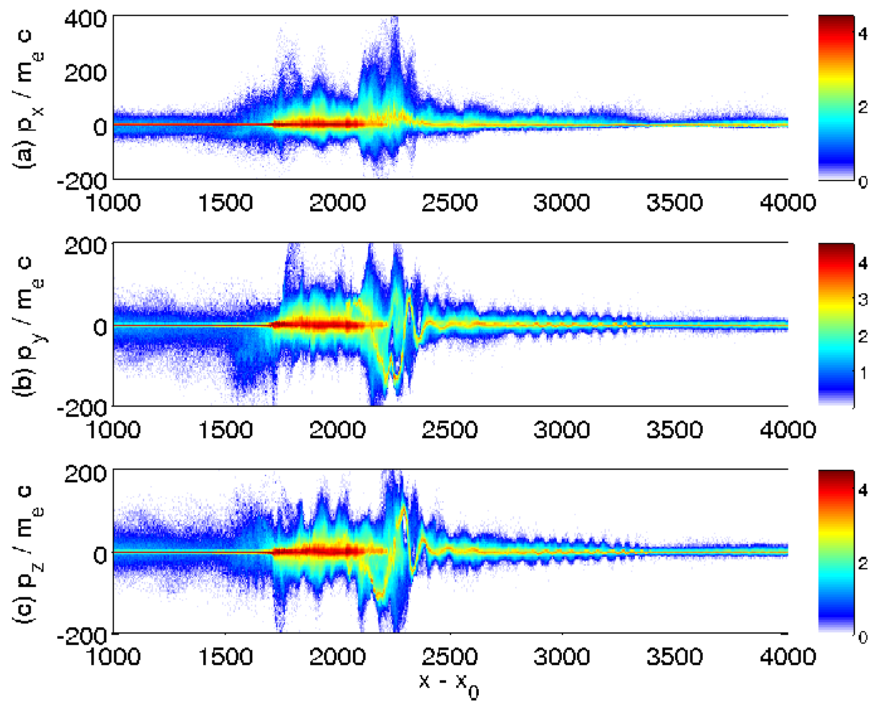


1D Simulation (2)

Left column: Projected phase space distributions of ions at simulation's end at *$t\ \omega_p \approx 5000$*



Left column: Projected phase space distributions of electrons at simulation's end at *$t\ \omega_p \approx 5000$*



Summary

- Brief outline of a phase space fluid and illustration of the Particle-In-Cell method.
- Aspects of the simulation: Need for reduced geometries
- Example simulation of an internal GRB shock.
- Future work: Vary parameters and assess their impact

Thank you for your attention!