www.ino.it

INO-CNR
ISTITUTO NAZIONALE DI OTTICA

Introduction to Particle-In-Cell (PIC) Simulations *(for absolute beginners)*

Andrea MACCHI *

Course on:
Astrophysical and Space Plasmas
L'Aquila (Italy),
September 3 – 7, 2012

INO Research Unit "Adriano Gozzini"
CNR, Pisa, Italy

*also at Dipartimento di Fisica "Enrico Fermi",
Largo Bruno Pontecorvo 3, 56127 Pisa, Italy
www.andreamacchi.eu

OUTLOOK

- Plasma simulations for beginners: a simple electrostatic model

- The PIC approach for the kinetic simulation of collisionless plasmas
  - basics: the Particle and the Cell
  - standard methods and algorithms
  - use on supercomputers and parallelization

- Adding extra physics: the example of Radiation Friction (from theory to numerical implementation)

- Snapshots from multi-dimensional PIC simulations

FOR ABSOLUTE BEGINNERS: SHEET MODEL

Dawson, “One-dimensional plasma model”, *Phys.Fluids* **5** (1962) 445

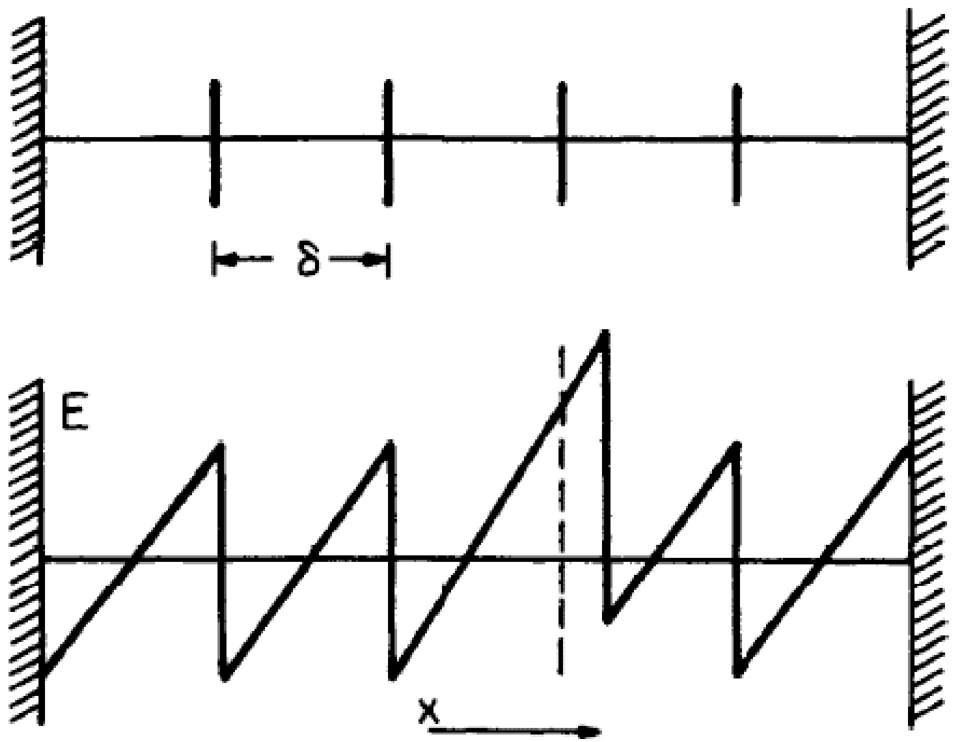


Figure 13-2a Original *Dawson* (1962) model, with thin electron sheets spaced $\delta = 1/n$ apart (in equilibrium) in a uniform positive ion background. The lower part shows $E(x)$ with one sheet displaced.

BASICS OF THE SHEET MODEL - I

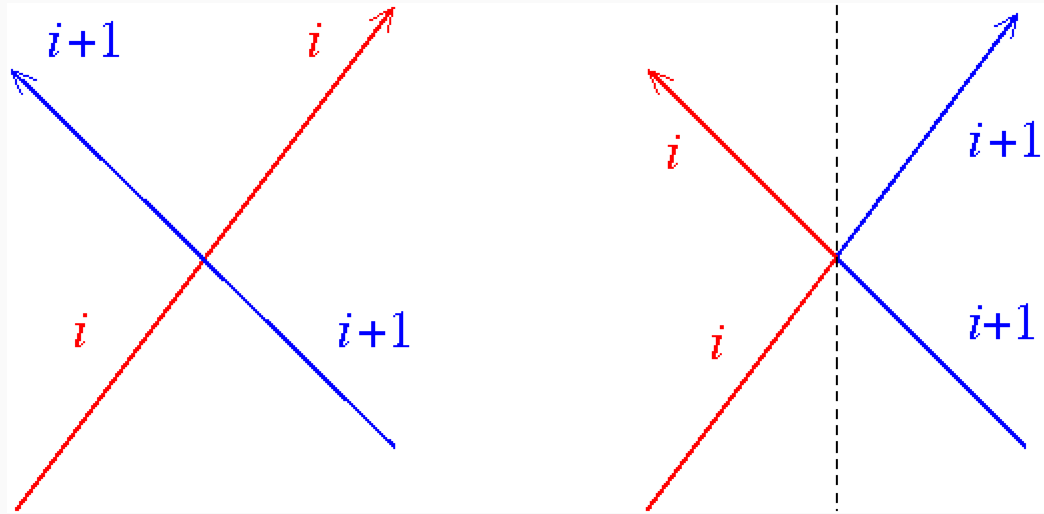
Motion of N electrons in 1D (so, “charge sheets”), fixed ions under the action of electrostatic field E_x (+ external force driver F_{ext})

$$
\begin{aligned}
X_i &= X_i(t)\,, \qquad i = 1, \ldots, N\,, \qquad X_i(0) = X_i^0 \\
\frac{d^2 X_i}{dt^2} &= -\frac{e}{m_e} E_x(X_i) + F_{\text{ext}} \\
E_x(X_i) &= \int_0^{X_i} 4\pi e(n_0 - n_e)dx = 4\pi e n_i X_i - 4\pi \sum_{j<i} \sigma_j \\
&= 4\pi e n_0 (X_i - X_i^0) \qquad (n_0 = Z n_{\text{i}} \text{ uniform, } \textbf{no crossings})
\end{aligned}
$$

If $F_{\text{ext}}=0$ and no crossing occurs between the sheets, the latter oscillate around $x=X_i^0$ at the plasma frequency $\omega_p = \sqrt{4\pi e^2 n_i / m_e}$ (for a homogeneous plasma)

BASICS OF THE SHEET MODEL - II

Crossing of neighboring sheets can be modeled as an “elastic collision” equivalent to a remapping of the sheet index: the field on a sheet due to other electron sheet is constant: $E_{el}[X_i^0(t)]=E_{el}(X_i^0)$



Using this trick + numerical integration of the equations of motion (with Runge-Kutta, Leapfrog, Verlet, ..., algorithms as preferred) yields an elementary plasma simulation code (can be generalized to inhomogenous plasma and/or external driver)

PLAYING WITH THE SHEET MODEL - I

INO-CNR Istituto Nazionale di Ottica

The sheet model can be used for a first numerical insight into kinetic dynamics (plasma oscillations, Debye shielding, Landau damping, wake excitation and collisionless stopping)

[see Birdsall & Langdon, "*Plasma Physics via Computer Simulation*" (IOP, 1991) Chap.13, p.277-292]

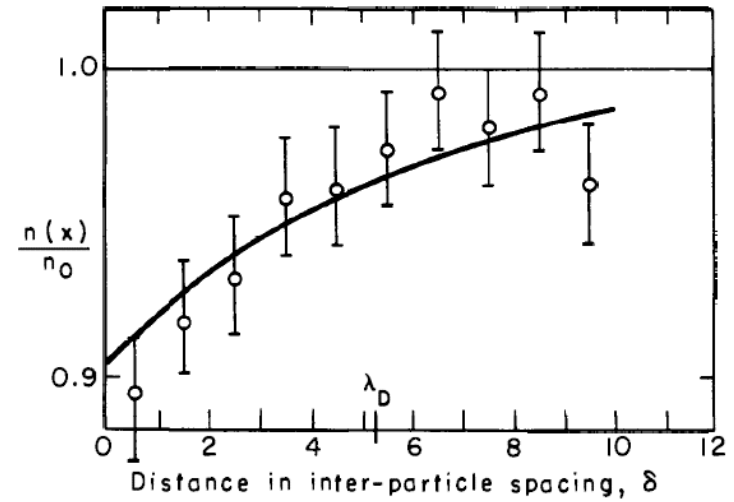
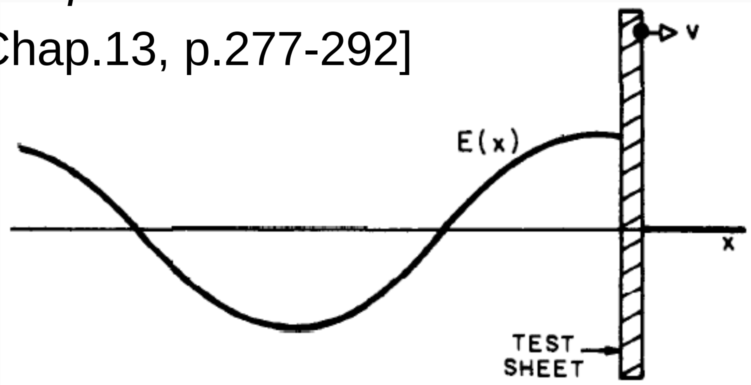
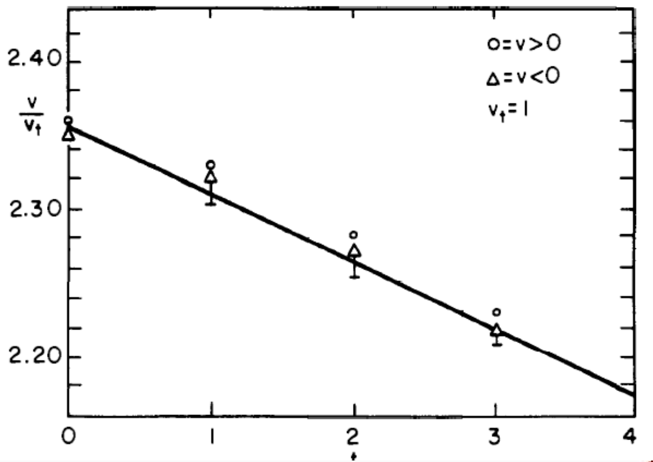


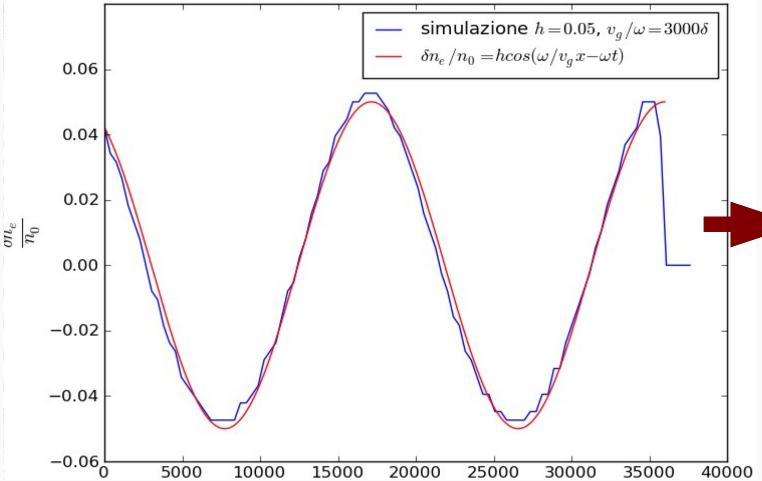
Figure 13-2c Average density of electrons around a test electron sheet at $x = 0$. The curve is the Debye shielding prediction. $n\lambda_D = 5.16$. *(From Dawson, 1962.)*



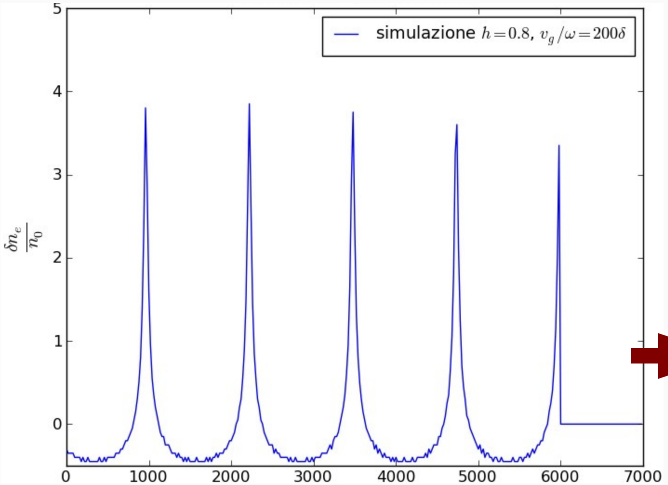


PLAYING WITH THE SHEET MODEL - II

Simulation of wake plasma wave generation by a fast projectile (EM pulse or energetic particle)
Relevant to wakefield acceleration of electrons



Upper plot: linear regime
(red: theory, blue: simulation)
Lower plot: nonlinear regime
(density spikes)



Courtesy of P. dell'Osso, seminar for the M.Sc. course, Pisa, 2012

PLAYING WITH THE SHEET MODEL - III

Application to stochastic heating

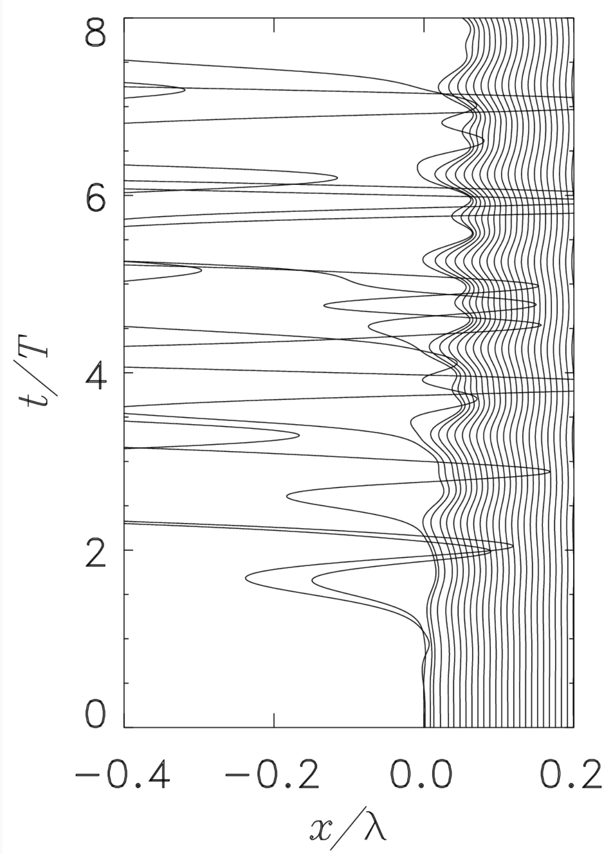
in plasma discharge sheaths

[Lieberman & Lichtenberg, *Principles of Plasma Discharges and Materials Processing* (Wiley, 2005)]

or at a steep laser-plasma interface

(leading to collisionless absorption)

[Mulser & Bauer, *High Power Laser-Matter Interaction* (Springer, 2010)]



Example figure shows

nonlinear electron oscillations driven

by an external laser field at a sharp plasma boundary

[Macchi, Borghesi, Passoni, Rev. Mod. Phys (2012) in press]

VLASOV-MAXWELL EQUATIONS

Continuity equation in 6D phase space **(r,p)** for each species a coupled via momenta of f_a to Maxwell's equations ("natural" units are used!)

$$\frac{df_a}{dt} = \frac{\partial f_a}{\partial t} + \frac{\partial}{\partial \mathbf{r}}(\dot{\mathbf{r}}_a f_a) + \frac{\partial}{\partial \mathbf{p}}(\dot{\mathbf{p}}_a f_a) = 0 \qquad f_a = f_a(\mathbf{r}, \mathbf{p}, t)$$

$$\dot{\mathbf{p}}_a = q_a(\mathbf{E} + \mathbf{v} \times \mathbf{B}) \qquad \dot{\mathbf{r}}_a = \mathbf{v} = \frac{\mathbf{p}}{\sqrt{\mathbf{p}^2 + m_a^2}}$$

$$\rho(\mathbf{r}, t) = \sum_{a=e,i} q_a \int d^3p f_a \qquad \mathbf{J}(\mathbf{r}, t) = \sum_{a=e,i} q_a \int d^3p \mathbf{v} f_a$$

$$\nabla \cdot \mathbf{E} = \rho \qquad \nabla \cdot \mathbf{B} = 0 \qquad \nabla \times \mathbf{E} = -\partial_t \mathbf{B} \qquad \nabla \times \mathbf{B} = \mathbf{J} + \partial_t \mathbf{E}$$

Basis for the kinetic description of a collisionless, relativistic, classical plasma with self-consistent mean EM fields

DULLY NUMERICAL IMPLEMENTATION

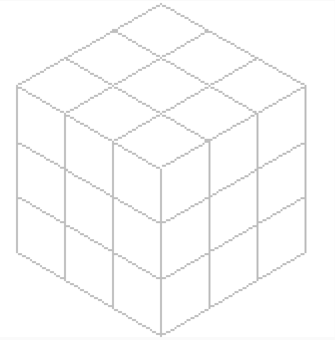
Assume an initial value problem (a study of plasma dynamics)

- Discretize phase space:

$x = i\Delta x$, $i = 0, 1, 2, \ldots$,

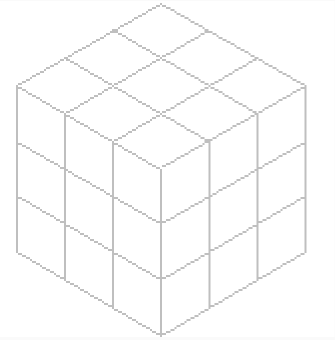
$p_x = j\Delta p$, $j = 0, 1, 2, \ldots$,

6D Numerical Grid
(not necessarily Cartesian!)



$x\ y\ z$

X



$p_x p_y p_z$

- Discretize time: $t = n\Delta t$, $n = 0, 1, 2, \ldots$,

- Find an algorithm (i.e. finite differences) to advance f with the desired accuracy (test conservation laws: mass, energy, ...)

- Write the code, debug, test, optimize, and run
(and check if the results converge with increasing resolution ...)

HOW LARGE MUST MY GRID BE?

In physical space the number of points on each axis is $N_q = L/\Delta x$

L = size of the system to be simulated

$\Delta x < d$ = smallest scale to be resolved: depending on the problem

$d = \lambda_D$, c/ω_p, λ (wavelength of an EM driver), ...

Rule of thumb $L \sim 10d$, $\Delta x \sim d/10$ ==> $N_q \sim 10^2$

In 3D we get $N_q^3 \sim 10^6$ gridpoints for the spatial sub-grid

If the grid is similar for momentum space $N_p \sim 10^2$

total $N \sim N_q^3 N_p^3 \sim 10^{12}$ gridpoints (for each species)

--> 8 TBytes allocated to represent f as a double precision number

SIZE MATTERS: HOW TO DEAL WITH IT?

“Plasma physics is just waiting for bigger computers” (Anonymous)

- Use ROADRUNNER if you can (needs efficient parallel programming)

- Use a different, memory-saving Approach: Particle-In-Cell (PIC) method – see next slides

- Restrict yourself to a “model problem”: lower dimensionality (1D, 2D), “feasible” parameters, ...

- Remember: **NO simulation can be really “realistic”** i.e. take actual space and time scales with appropriate resolution: Simulations are **models** for “real” physics

PARTICLE-IN-CELL (PIC) METHOD

Assume a discrete "particle" representation of f :

$$f(q,p,t) = \sum_{i=0}^{N_p-1} g[q - q_i(t)]\delta[p - p_i(t)]$$

By substituting into the kinetic Eq. for f we obtain the Equations of Motion for the N_p (vector) variables $p_i(t)$ and $q_i(t)$:

$$\dot{p}_i = \bar{F}_i \qquad \dot{q}_i = \frac{p_i}{m}$$

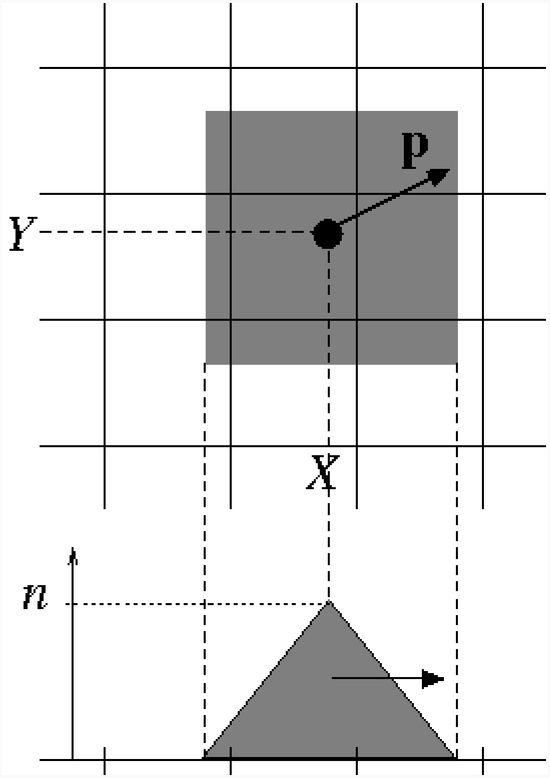
$$\bar{F}_i = \bar{F}_i(q_i, p_i, t) = \int g(q - q_i)F(q, p_i, t)dq$$

The phase space is represented as an ensemble of particles (delta-like in p and extended in q via the function $g(q)$):

PIC is a "Lagrangian" approach vs. "Eulerian" (also called "Vlasov")

THE PARTICLE AND THE CELL

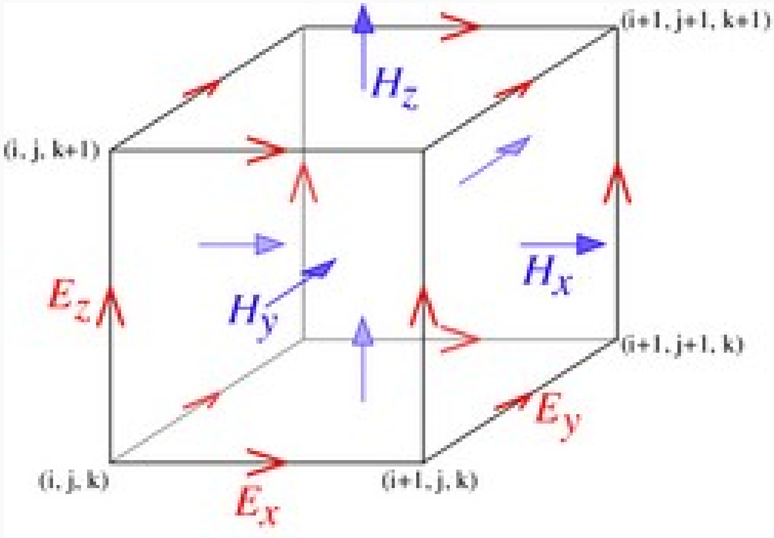
- The plasma is represented by a large (but limited) set of computational particles (simply named “electrons” and “ions”)
- The EM fields are allocated on a discrete grid, i.e. “in the cell”
- Each particle (usually extended in space) contributes to the charge and current densities in its parent cell (and its neighbors)
- The Lorentz force is evaluated as an average over the fields in the overlapping cells

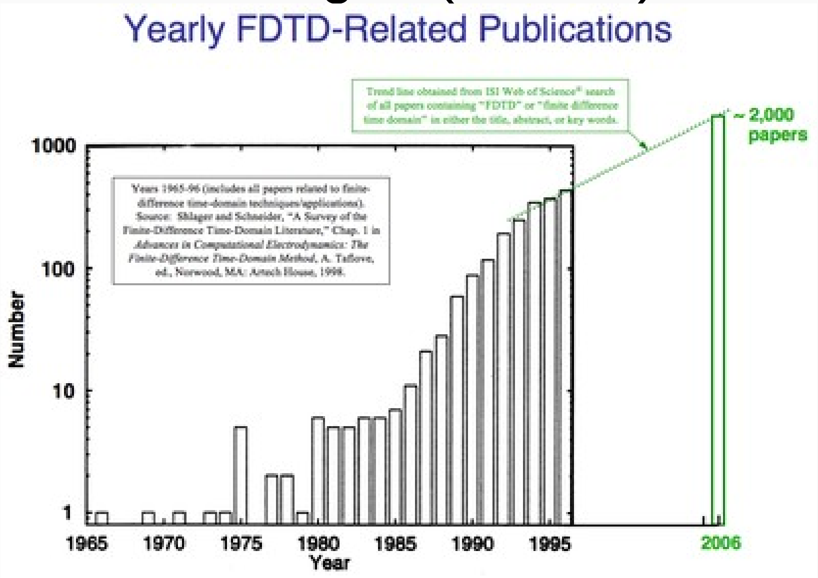


On a supercomputer up to some ~10^9 particles may be allocated (still typically orders of magnitude smaller than real numbers)

FDTD AND YEE LATTICE

- FDTD : Finite-Difference Time-Domain typical method to solve Maxwell equations in PIC codes
- Yee lattice: typical distribution of EM fields used in the cell of a Cartesian grid (2D, 3D)





Pictures taken from Wikipedia: http://en.wikipedia.org/wiki/Finite-difference_time-domain_method

INO-CNR
Istituto Nazionale di Ottica

ENFORCING THE CONTINUITY EQUATION

- Fields are advanced from $\partial_t \mathbf{B} = -\nabla \times \mathbf{E} \qquad \partial_t \mathbf{E} = \nabla \times \mathbf{B} - \mathbf{J}$
- the scheme preserves $\nabla \cdot \mathbf{B} = 0$
- possible inconsistency of the electrostatic field $(\nabla \cdot \mathbf{E} \neq \rho)$
- the equivalent continuity equation $\nabla \cdot \mathbf{J} = -\partial_t \rho$ is **not** exactly satisfied on the grid if densities are calculated "trivially" as

$$\begin{pmatrix} \rho \\ \mathbf{J} \end{pmatrix} = \sum_{i=0}^{N_p-1} Q_i \begin{pmatrix} 1 \\ \mathbf{v} \end{pmatrix} g[\mathbf{r} - \mathbf{r}_i(t)]$$

- **solution**: "smart" (non-trivial) reconstruction of $\mathbf{J}$ from the particle displacement to satisfy the continuity equation (or, improperly, "charge conservation")

[Eastwood, Comp.Phys.Comm. **64**, 252 (1991);
Villanesor & Buneman, *ibid.* **69**, 306 (1992); Esirkepov, *ibid.* **135**, 144 (2001)]

PARTICLE PUSHER EXAMPLE

- Advance positions:
Leapfrog algorithm

$$\mathbf{r}^{(n+1)} = \mathbf{r}^{(n)} + \mathbf{v}^{(n+1/2)}\Delta t + \mathcal{O}(\Delta t^2)$$

$$\left(\mathbf{v}^{(n+1/2)} = \frac{\mathbf{p}^{(n-1/2)}}{m_a \gamma^{(n-1/2)}}\right)$$

- Advance momenta:
Boris pusher
(1st half boost by $\mathbf{E}$
+ rotation by $\mathbf{B}$
+ 2nd half boost by $\mathbf{E}$)

$$\mathbf{p}^{(-)} = \mathbf{p}^{(n-1/2)} + \frac{q}{m}\mathbf{E}^{(n)}\frac{\Delta t}{2}$$

$$\mathbf{p}^{(+)} = \mathbf{p}^{(-)} + \mathbf{p}^{(-)} \times \mathbf{s} + (\mathbf{p}^{(-)} \times \mathbf{t}) \times \mathbf{s}$$

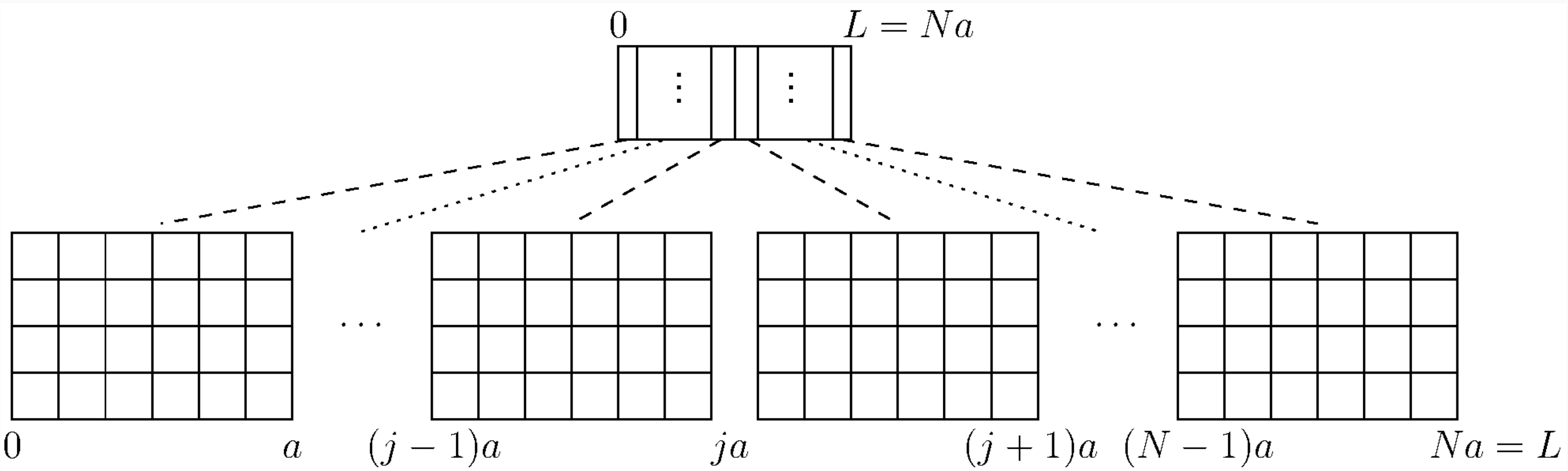
$$\gamma^{(n)} = \sqrt{1 + (\mathbf{p}^{(-)}/mc)^2} \qquad \mathbf{t} \equiv \frac{q\mathbf{B}^n}{m\gamma^n c} \qquad \mathbf{s} \equiv \frac{2\mathbf{t}}{1+t^2}$$

$$\mathbf{p}^{(n+1/2)} = \mathbf{p}^{(+)} + \frac{q}{m}\mathbf{E}^{(n)}\frac{\Delta t}{2}.$$

More accurate schemes (better than Δt^2 accuracy)
may be used at some computational cost

A PARALLELIZATION STRATEGY - I

- Divide the spatial domain (grid) along a direction of size L into N equal domains (subgrids) of size $a=L/N$
- Assign each domain to a processing node



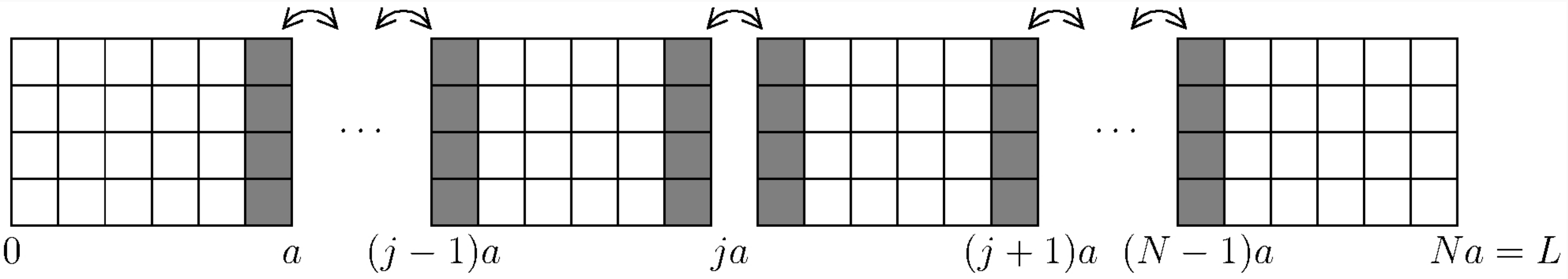
("Single Program Multiple Data" paradigm)

A PARALLELIZATION STRATEGY - II

If the algorithms are local*, only data from neighboring rows (layers) of cells need to be exchanged via message passing:

- communication effort is minimized

==> linear scaling of performance with number of nodes



* Since $|\Delta r| < c\Delta t$ for numerical stability reasons, particles move (and fields propagate) by at most one cell per timestep

A PARALLELIZATION STRATEGY - III

For most typical problems the memory allocated on each node is proportional to the number of particles located in the corresponding spatial sub-domain:

- for uniform domain decomposition, the plasma should be homogeneous along the direction chosen for the partition
- as particles move across domains the work load on each domain becomes different resulting in loss of performance: load balancing would need a dynamic partition reconfiguration

RADIATION FRICTION IN PIC

For point-like, highly relativistic electrons in strong fields the equation of motion must be modified in order to:

- take the back-reaction of the fields generated by the electron itself into account
- make the energy-momentum balance consistent with the emission of radiation due to the electron acceleration

Radiative losses and radiation friction may play a dominant role in relativistic current sheets whose dynamics and instabilities may explain ultra-high energy acceleration in astrophysics

See e.g.: Jaroschek and Hoshino, Phys. Rev. Lett. **103**, 075002 (2009)

INO-CNR Istituto Nazionale di Ottica

CHOICE OF THE RF FORCE

Classical force on the electron modified by Radiation Friction (RF) [Landau and Lifshitz (LL), *The Classical Theory of Fields* (ch.76)]

$$\mathbf{f} = -e\left(\mathbf{E} + \frac{\mathbf{v}}{c}\times\mathbf{B}\right) + \mathbf{f}_{\rm rad}$$

$$\mathbf{f}_{\rm rad} = \frac{2r_c^2}{3}\left\{-\gamma^2\left[\left(\mathbf{E}+\frac{\mathbf{v}}{c}\times\mathbf{B}\right)^2 - \left(\frac{\mathbf{v}}{c}\cdot\mathbf{E}\right)^2\right]\frac{\mathbf{v}}{c} + \right.$$
$$\left. + \left[\left(\mathbf{E}+\frac{\mathbf{v}}{c}\times\mathbf{B}\right)\times\mathbf{B} + \left(\frac{\mathbf{v}}{c}\cdot\mathbf{E}\right)\mathbf{E}\right] - \gamma\left(d_t\mathbf{E} + \frac{\mathbf{v}}{c}\times d_t\mathbf{B}\right)\right\}$$

Underlying approximations are almost always justified within the limits of validity of classical electrodynamics

Note: RF is being intensively revisited in the laser-plasma context because of forthcoming interaction regimes at ultra-high laser fields (offering the first possible direct tests of RF theories)

SOME REMARKS ON RF INCLUSION

- Unlike in standard Vlasov equation, the momentum divergence does **not** commute with the force:

$$\dot{\mathbf{p}}_e = -e(\mathbf{E}+\mathbf{v}\times\mathbf{B})+\mathbf{f}_{\mathrm{rad}} \quad \Longrightarrow \quad \frac{\partial}{\partial \mathbf{p}}(\dot{\mathbf{p}}_e f_e) \neq \dot{\mathbf{p}}_e \cdot \frac{\partial f_e}{\partial \mathbf{p}}$$

However the PIC method solves the more general continuity equation

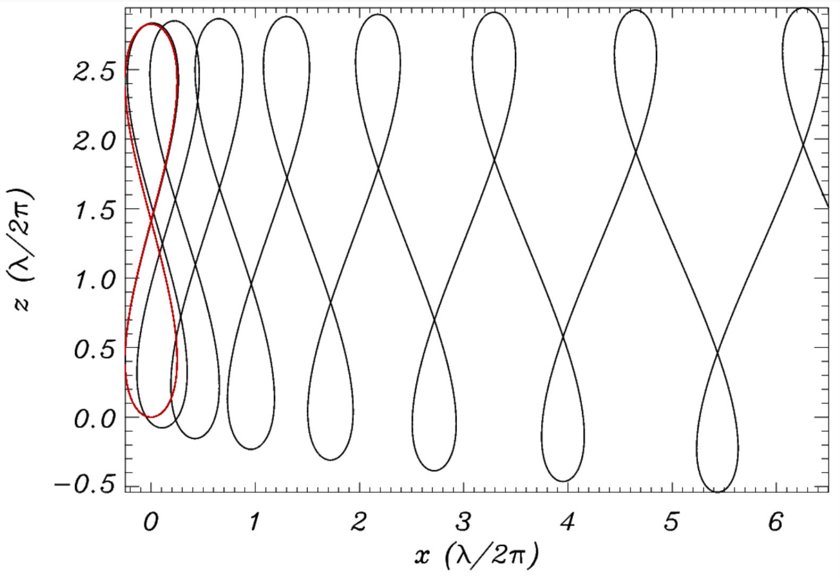
- The radiation emission is incoherent and dominant frequencies are high enough to be not resolved on the spatial grid:

$\mathbf{f}_{\mathrm{rad}}$ acts as a dissipative force (energy disappears from the system)

(in principle there might be a double counting of low-frequency contributions in $\mathbf{f}_{\mathrm{rad}}$ but the effect is negligibly small)

RF IMPLEMENTATION AND BENCHMARKS

- Benchmark of the numerical implementation
with exact solution for electron motion in plane wave including RF
[Di Piazza, Lett. Math. Phys. **83**, 305 (2008)]



Red: no RF, closed "Figure of 8"
Black: with RF, "drifting 8"

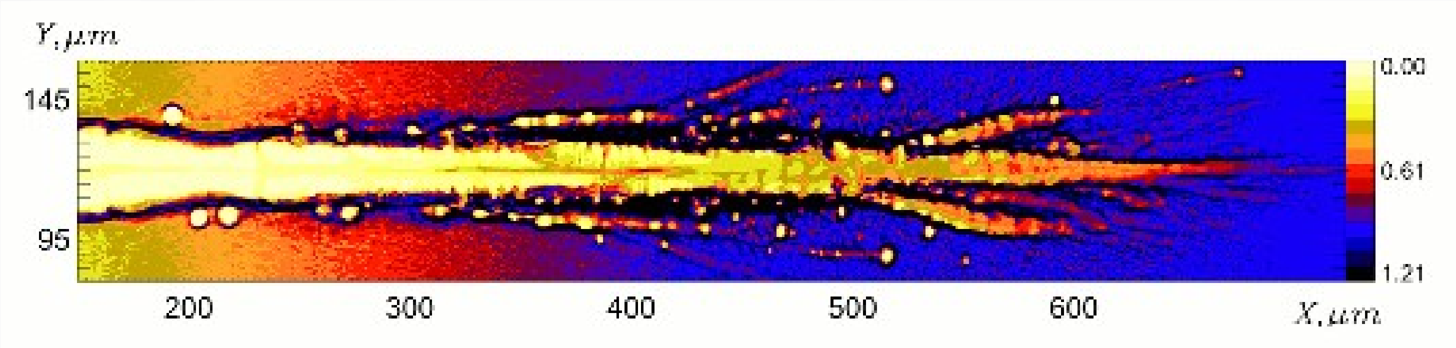
Good news: the non-time local term containing $d_t\mathbf{E}$ and $d_t\mathbf{B}$ in $\mathbf{f}_{\mathrm{rad}}$ is almost always negligible

- LL force can be inserted in a modular way in a standard PIC code
[Tamburini et al, New. J. Phys. **10**, 123005 (2010)]

PIC VS VLASOV: PROS AND CONS

EASY / HARD	FEASIBLE / UNKNOWN	PIC	VLASOV
DEVELOPMENT		easy, quite general, well documented	non trivial, specific
NOISE		significant	negligible (control of physical instabilities)
WORKLOAD		saving	very large
DENSITY RESOLUTION		problems with statistics & large gradients	excellent
MOMENTUM SPACE		unbounded	bounded
PARALLELIZATION		well suitable but non trivial load balancing	straightforward for local algorithms
FLEXIBILITY / ADD. PHYSICS		SWF models, collisions, ionization, ...	? unable to judge

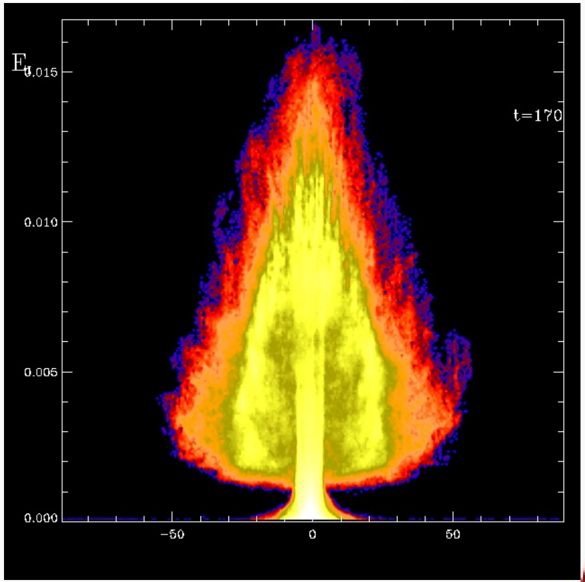
SOME LASER-PLASMA **PIC**-TURES - 2D



Top: self-channeling, breakup and soliton formation by an intense laser pulse

Right: momentum vs angle distribution of ions for radiation pressure acceleration of a dense plasma slab

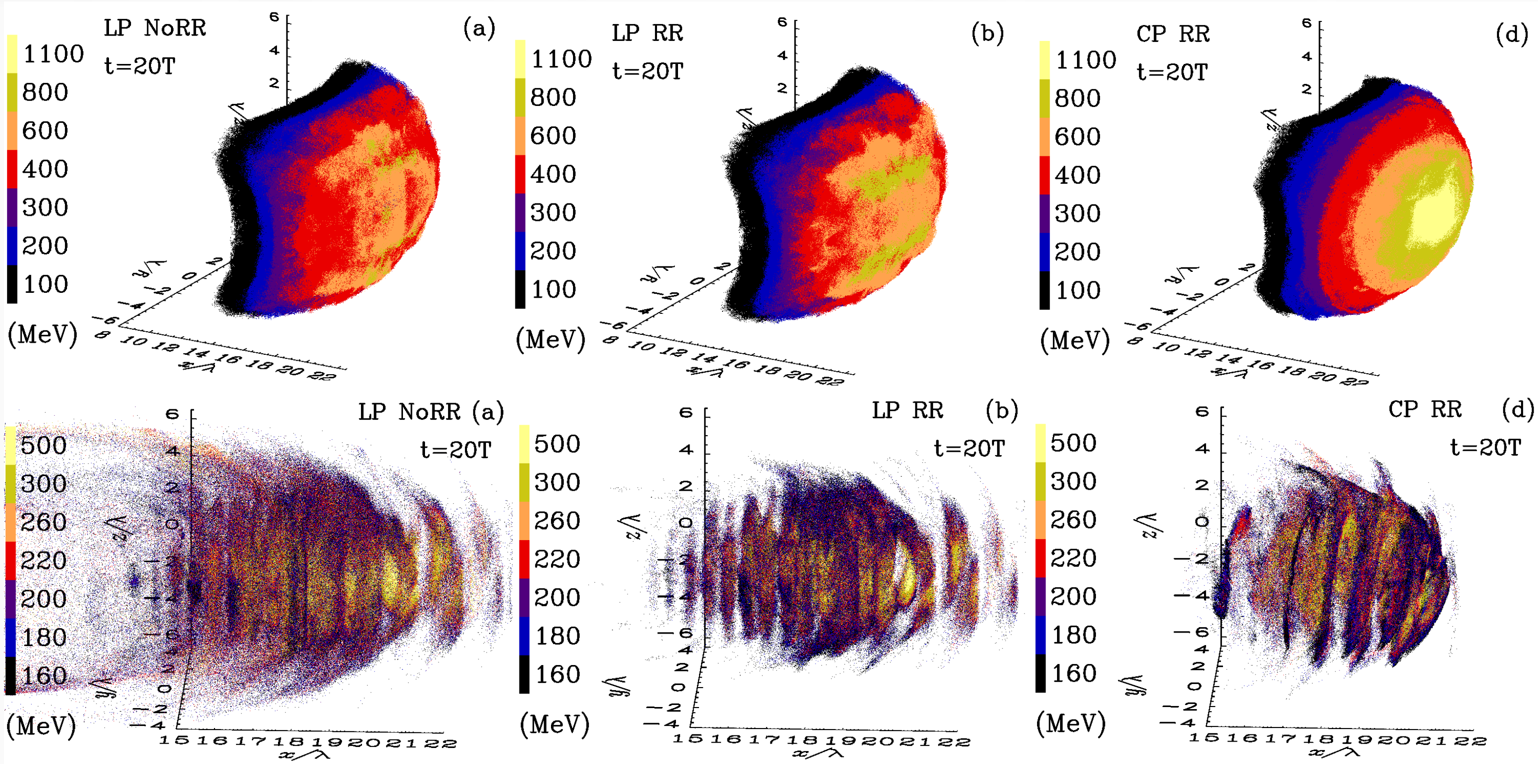
Simulations performed at CINECA, Italy



T.V.Liseykina and A.Macchi, IEEE Trans. Pl. Sc. **36**, 1136 (2008), Special Issue on "Images in Plasma Science"

INO-CNR
Istituto Nazionale di Ottica

SOME LASER-PLASMA **PIC**-TURES - 3D

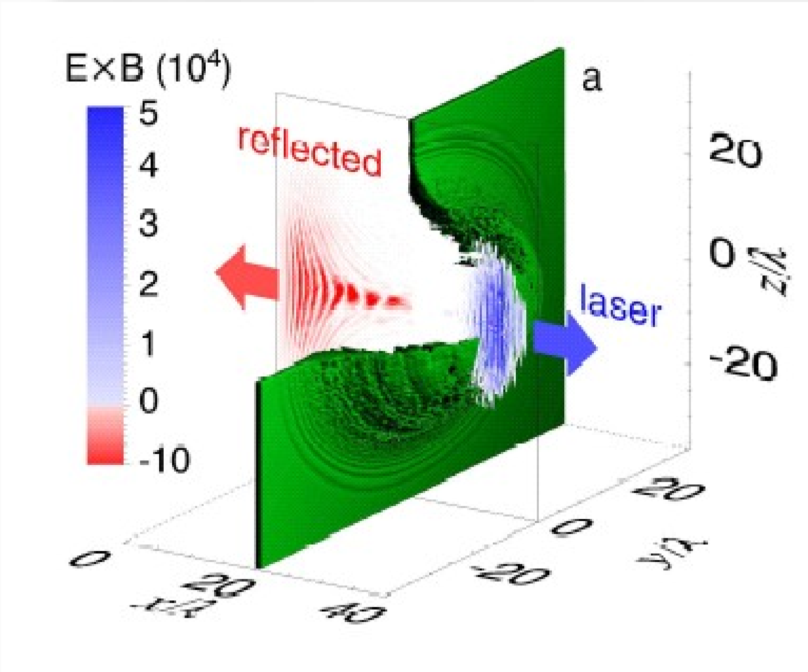


Radiation Pressure Acceleration of a thin plasma layer in 3D with radiation friction included

Tamburini, Lyseikina, Pegoraro, Macchi, Phys. Rev. E **85**, 16407 (2012)

INO-CNR
ISTITUTO NAZIONALE DI OTTICA

OTHER BEAUTIFUL 3D **PIC**-TURES ...

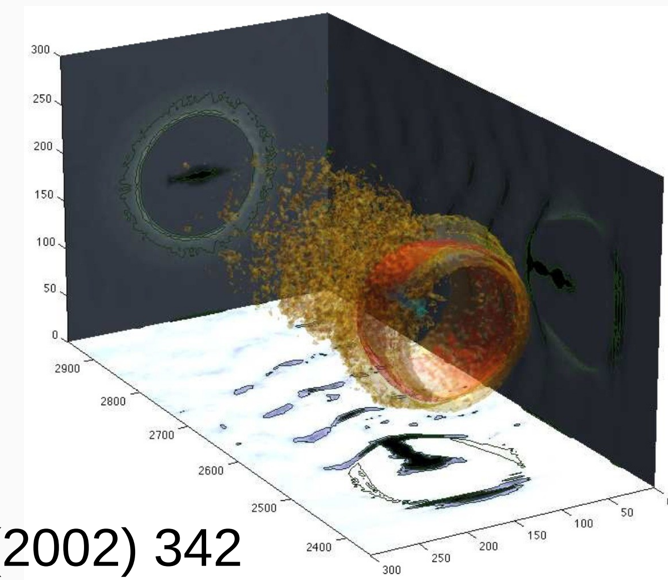


Left: radiation pressure acceleration

in the extreme intensity regime

T.Esirkepov et al,

Phys.Rev.Lett. **92** (2004) 175003



Right: plasma “bubble” formation

for laser-plasma electron acceleration

Simulation by OSIRIS code

L.Fonseca et, *Lect. Notes Comp. Sci.* **2331**(2002) 342

Smart data visualization is important (and a key to success...)

FINAL CONSIDERATIONS

- PIC is a well established approach to “ab initio” kinetic simulation of collisionless plasmas in a mean-field approach
- PIC may be adapted to suitable problem-oriented approximations (gyrokinetic, hybrid, ...) and may include additional physics (collisions, ionization, radiation friction, ...) and/or diagnostics
- Work is **always** in progress to fully exploit progress in computer hardware (new architectures, GPUs, ...), to better visualize data, ...

Advice to absolute beginners:

- don't always use codes as a **black box**
- be nit-pickers with testing, debugging, and comparing with known solutions (if there is any)
- train yourself on tool models (e.g. the sheet model)
- *have fun*!