LA-UR-13-21249
Approved for public release; distribution is unlimited.

Title:	Fully implicit particle-in-cell algorithms for kinetic simulation of plasmas
Author(s):	Chacon, Luis
Intended for:	Seminar at Plasma Science and Fusion Center, MIT on 3/1/13

Fully implicit particle-in-cell algorithms for kinetic simulation of plasmas

L. Chacón

Los Alamos National Laboratory
P.O. Box 1663, Los Alamos, NM 87544

Collaborators:
Guangye Chen (LANL),
D. C. Barnes (Coronado Consulting)
D. A. Knoll (LANL)

Plasma Science and Fusion Center (MIT) Seminar
March 1st, 2013
Cambridge, MA

Aspects of this work received support from ORNL LDRD, DOE-FES, and LANL LDRD

Los Alamos NATIONAL LABORATORY EST.1943

Luis Chacon, chacon@lanl.gov

Outline

- Particle methods for plasma simulation (PIC)
- State of the art algorithm: explicit approach
- Status of implicit PIC: problems and limitations
- Our approach: energy and charge-conserving implicit **electrostatic** PIC
  - Vlasov-Ampere vs. Vlasov-Poisson
  - Exact energy-conserving formulation
  - Exact charge-conserving mover
  - Momentum conservation error control: orbit adaptivity
- Generalization to mapped (body fitted) meshes
- Preconditioning: Moment-based acceleration
- Generalization to **electromagnetic** PIC: energy-conserving Darwin model
- Potential for heterogeneous computing: hybrid CPU-GPU implementation

Introduction

Luis Chacon, chacon@lanl.gov

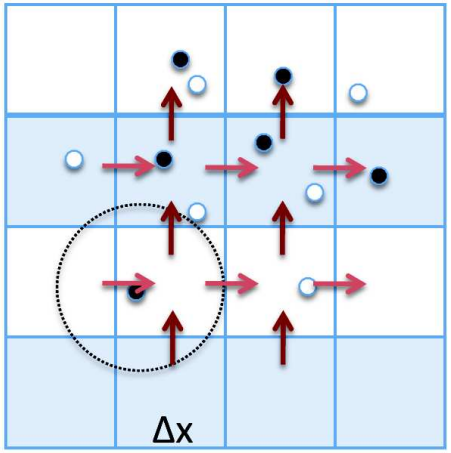
Particle-in-cell (PIC) methods for kinetic plasma simulation

$$\partial_t f + \mathbf{v}\cdot\nabla f + \frac{\mathbf{F}}{m}\cdot\nabla_v f = \left(\frac{\partial f}{\partial t}\right)_{col}$$

- Ignoring collisions$\Rightarrow$ Lagrangian solution by the **method of characteristics**:

$$f(\mathbf{x},\mathbf{v},t) = f_0\left(\mathbf{x} - \int_0^t dt\mathbf{v}, \mathbf{v} - \frac{1}{m}\int_0^t dt\mathbf{F}\right)\ ;\ \mathbf{x}(t=0)=\mathbf{x}_0\ ;\ \mathbf{v}(t=0)=\mathbf{v}_0$$

- PIC approach follows characteristics employing **macroparticles** (volumes in phase space)



$$f(\mathbf{x},\mathbf{v},t) = \sum_p \delta(\mathbf{x}-\mathbf{x}_p)\delta(\mathbf{v}-\mathbf{v}_p)$$

$$\begin{aligned} \dot{\mathbf{x}}_p &= \mathbf{v}_p \\ \dot{\mathbf{v}}_p &= \frac{q_p}{m_p}(\mathbf{E}+\mathbf{v}\times\mathbf{B}) \end{aligned}$$

$$\begin{aligned} \partial_t\mathbf{B} + \nabla\times\mathbf{E} &= 0 \\ -\mu_0\epsilon_0\partial_t\mathbf{E} + \nabla\times\mathbf{B} &= \mu_0\mathbf{j} \\ \nabla\cdot\mathbf{B} &= 0 \\ \nabla\cdot\mathbf{E} &= \frac{e(n_i-n_e)}{\epsilon_0} \end{aligned}$$

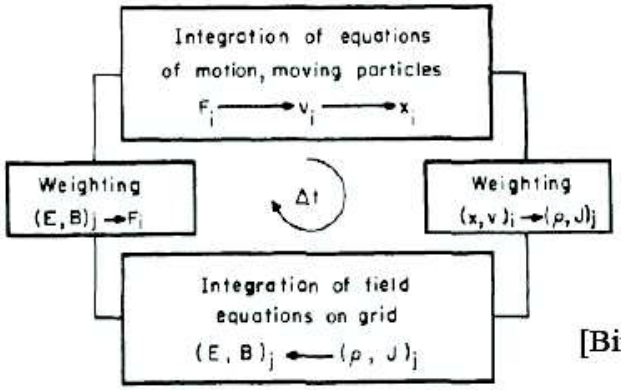
$$\delta(\mathbf{x}-\mathbf{x}_p) \longrightarrow S(\mathbf{x}-\mathbf{x}_p)\ ;\ E_p = \sum_i E_i S(x_i - x_p)\ ;\ j_i = \sum_p j_p S(x_i - x_p)$$

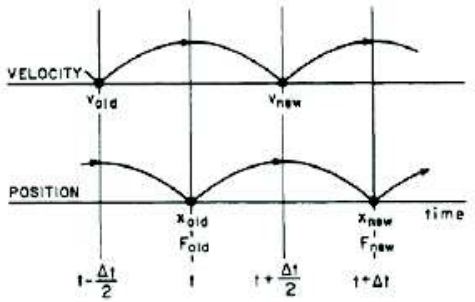
Los Alamos NATIONAL LABORATORY EST.1943

Luis Chacon, chacon@lanl.gov

State-of-the-art *classical* PIC algorithm is explicit

- Classical explicit PIC approach "leap-frogs" particle positions and velocities, solves for fields after position update:





[Birdsall and Langdon, Plasma physics via computer simulation]

- Severe performance limitations:
  - $\Delta x < \lambda_{Debye}$ (finite-grid instability: enforces a **minimum spatial resolution**)
  - $\omega_{pe}\Delta t < 1$ (CFL-type instability: enforces a **minimum temporal resolution**)
  - **Inefficient** for long-time, large-scale integrations
- In the presence of strong magnetic fields, gyro-averaging the Vlasov-Maxwell model can significantly ameliorate these limitations, but there are other issues (e.g. not asymptotic preserving, required order of expansion to capture some physical effects, treatment of nonlinear terms)

WE FOCUS ON ELECTROSTATIC PIC AS A PROOF OF PRINCIPLE

What about implicit PIC?

- Implicit PIC holds the promise of overcoming the difficulties and inefficiencies of explicit methods for long time-scale simulations
- Exploration of implicit PIC started in the 1980s
  - Moment method [Mason, 1981; Brackbill, 1982]
  - Direct method [Friedman, Langdon, Cohen, 1981]
- Early approaches used linearized, semi-implicit formulations:
  - Lack of nonlinear convergence
  - Inconsistencies between particles and moments
  - Inaccuracies! $\rightarrow$Plasma self-heating/cooling [Cohen, 1989]

Our goal is to explore the viability of a nonlinearly converged, fully implicit PIC algorithm

What is the nature of the resulting fully-coupled algebraic system?
Is it practical to invert?

Fully implicit electrostatic PIC

Luis Chacon, chacon@lanl.gov

Fully implicit PIC formulation

- A fully implicit formulation couples particles and fields non-trivially (integro-differential PDE):

$$\frac{f^{n+1}-f^n}{\Delta t}+\mathbf{v}\cdot\nabla\frac{f^{n+1}+f^n}{2} \quad - \quad \frac{q}{m}\nabla\frac{\Phi^{n+1}+\Phi^n}{2}\cdot\nabla_{\mathbf{v}}\frac{f^{n+1}+f^n}{2}=0$$

$$\nabla^2\Phi^{n+1} \quad = \quad \int d\mathbf{v}\, f^{n+1}(\mathbf{x},\mathbf{v},t)$$

- In PIC, f^{n+1} is sampled by a large collection of particles in phase space, $\{\mathbf{x},\mathbf{v}\}_p^{n+1}$.
  - There are N_p particles, each particle requiring $2\times d$ equations ($d\rightarrow$dimensions),
  - Field requires N_g equations, one per grid point.
- If implemented naively, an impractically large algebraic system of equations results:

$$\boxed{\mathbf{G}(\{\mathbf{x},\mathbf{v}\}_p^{n+1},\{\Phi^{n+1}\}_g)=0} \;\rightarrow\; \dim(\mathbf{G})=2dN_p+N_g\gg N_g$$

  - No current computing mainframe can afford the memory requirements
  - Algorithmic issues are showstoppers (e.g., how to precondition it?)
- An alternative strategy exists: nonlinear elimination (**particle enslavement**)

Particle enslavement (nonlinear elimination)

- Full residual $\mathbf{G}(\{x,v\}_p,\{\Phi\}_g)=0$ is impractical to implement
- Alternative: nonlinearly eliminate particle quantities **so that they are not dependent variables**:
  - Formally, particle equations of motion are functionals of the electrostatic potential:

$$x_p^{n+1}=x_p[\Phi^{n+1}]\ ;\ v_p^{n+1}=v_p[\Phi^{n+1}]$$

$$\mathbf{G}(\mathbf{x}_p{}^{n+1},\mathbf{v}_p{}^{n+1},\mathbf{\Phi}^{n+1})=\mathbf{G}(\mathbf{x}[\mathbf{\Phi}^{n+1}],\mathbf{v}[\mathbf{\Phi}^{n+1}],\mathbf{\Phi}^{n+1})=\tilde{\mathbf{G}}(\mathbf{\Phi}^{n+1})$$

Nonlinear residual can be *unambiguously* formulated in terms of electrostatic potential only!

- JFNK storage requirements are dramatically decreased, making it tractable:
  - Solver storage requirements $\propto N_g$, **comparable to a fluid simulation**
  - Particle quantities $\Rightarrow$ auxiliary variables: only a **single copy of particle population** needs to be maintained in memory throughout the nonlinear iteration

Jacobian-Free Newton-Krylov Methods

- After spatial and temporal discretization $\Rightarrow$ a large set of nonlinear equations: $\boxed{\vec{G}(\vec{x}^{n+1}) = \vec{0}}$
- Converging nonlinear couplings requires iteration: Newton-Raphson method:

$$\left.\frac{\partial \vec{G}}{\partial \vec{x}}\right|_k \delta\vec{x}_k = -\vec{G}(\vec{x}_k)$$

- Jacobian linear systems result, which require a linear solver $\Rightarrow$ Krylov subspace methods (GMRES)
  - Only require matrix-vector products to proceed.
  - Jacobian-vector product can be computed Jacobian-free (CRITICAL: no need to form Jacobian matrix):

$$\left(\frac{\partial \vec{G}}{\partial \vec{x}}\right)_k \vec{y} = J_k\vec{y} = \lim_{\epsilon\to 0}\frac{\vec{G}(\vec{x}_k + \epsilon\vec{y}) - \vec{G}(\vec{x}_k)}{\epsilon}$$

  - Krylov methods can be easily preconditioned: $P_k^{-1} \sim J_k^{-1}$

$$J_k P_k^{-1} P_k \delta\vec{x} = -\vec{G}_k$$

We will explore suitable preconditioning strategies later in this talk.

Field equation: Vlasov-Poisson vs. Vlasov-Ampere

- Nonlinear elimination procedure leads to $\mathbf{G}(\Phi) = 0$ (or $\mathbf{G}(E) = 0$)
- Two formulations are possible:

Vlasov-Poisson (VP)	**Vlasov-Ampère** (VA)
$\partial_t f + v\partial_x f + \frac{qE}{m}\partial_v f = 0$ $\partial_x E = \frac{\rho}{\epsilon_0}$ $E = -\partial_x \Phi$	$\partial_t f + v\partial_x f + \frac{qE}{m}\partial_v f = 0$ $\epsilon_0 \partial_t E + j = \langle j \rangle$
Two systems are equivalent in continuum, but not in the discrete.	
➤ Conventionally used in explicit PIC. ➤ Exact *local* charge conservation. ➤ Exact *global* momentum conservation. ➤ Unstable with orbit averaging in implicit context [Cohen and Freis, 1982].	➤ Exact *local* charge conservation. ➤ Exact *global* energy conservation. ➤ Suitable for orbit averaging. ➤ Can be extended to electromagnetic system.

- We will show, however, that an equivalent energy-conserving VP formulation exists.

Energy-conserving (EC) Vlasov-Ampère discretization

- Fully implicit Crank-Nicolson time discretization:

$$\varepsilon_0 \frac{E_i^{n+1} - E_i^n}{\Delta t} + \sum_p q_p v_p^{n+1/2} S(x_i - x_p^{n+1/2}) = 0$$

$$\frac{x_p^{n+1} - x_p^n}{\Delta t} = \frac{v_p^{n+1} + v_p^n}{2}$$

$$\frac{v_p^{n+1} - v_p^n}{\Delta t} = \frac{q_p}{m_p} \sum_i \frac{E_i^n + E_i^{n+1}}{2} S(x_i - x_p^{n+1/2})$$

In time:
centered, 2nd order;
implicit;
unconditionally stable;
non-dissipative.

- C-N enforces energy conservation to numerical round-off:

$$\sum_p \frac{m_p}{2}(v_p^{n+1} + v_p^n)(v_p^{n+1} - v_p^n) = -\sum_i \varepsilon_0 \frac{E_i^{n+1} - E_i^n}{\Delta t} \frac{E_i^{n+1} + E_i^n}{2} \Rightarrow \sum_p \frac{1}{2} m_p v_p^2 + \sum_i \frac{1}{2} \varepsilon_0 E_i^2 = \text{const}$$

- As a result, the formulation does not suffer from finite-grid instabilities (normal mode analysis)
  - ⇨ Unconstrained spatial resolution: $\Delta x \not< \lambda_D$!!
- Energy conservation is only realized when particles and fields are nonlinearly converged:
  - ⇨ Requires a tight nonlinear tolerance

Algorithmic implementation details

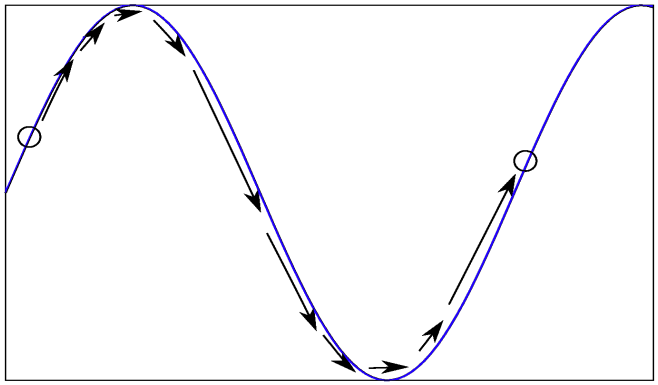
- The nonlinear residual formulation $\mathbf{G}(E^{n+1})$ based on Vlasov-Ampere formulation is as follows:
  1. Input E (given by JFNK iterative method)
  2. Move particles (i.e., find $x_p[E]$, $v_p[E]$ by solving equations of motion)
     (a) Requires inner (local) nonlinear iteration: Picard (not stiff)
     (b) Can be as complicated as we desire (substepping, adaptivity, etc)
  3. Compute moments (current)
  4. Form Vlasov-Ampere equation residual
  5. return
- Because particle move is performed within function evaluation, we have much freedom.
- Rest of the talk will describe improvements in particle mover to ensure **long-term accuracy**
  - ⇨ **Particle substepping and orbit averaging** (ensures orbit accuracy and preserves exact energy conservation)
  - ⇨ **Exact charge conservation strategy** (a new charge-conserving particle mover)
  - ⇨ **Orbit adaptivity** (to improve momentum conservation)

Particle orbit substepping

- In applications of interest, field time-scale (Δt) and orbit time-scale ($\Delta\tau$) can be well separated
  - Fields evolve *slowly* (dynamical time scale, Δt)
  - Particle orbits may still undergo *rapid change* ($\Delta\tau \ll \Delta t$)
- Particle orbits need to be resolved to **avoid large orbit integration errors**

Accurate orbit integration requires particle substepping!

- Field does not change appreciably: time-averaged value over long time scale is sufficient



$$\frac{x_p^{\nu+1} - x_p^{\nu}}{\Delta\tau} = v_p^{\nu+1/2}$$

$$\frac{v_p^{\nu+1} - v_p^{\nu}}{\Delta\tau} = \sum_i \underbrace{\frac{E_i^{n+1} + E_i^{n}}{2}}_{\text{slow}} S(x_i - x_p^{\nu+1/2})$$

Energy conservation and orbit averaging

- **Particle substepping breaks energy conservation**.
- Energy conservation theorem can be recovered by orbit averaging Ampère's law:

$$\epsilon_0 \partial_t E + j = \langle j \rangle \quad , \quad \frac{1}{\Delta t}\int_t^{t+\Delta t} d\tau [\cdots] \Rightarrow \epsilon_0 \frac{E^{n+1} - E^n}{\Delta t} + \bar{j} = \langle \bar{j} \rangle$$

- Orbit-averaged current is found as:

$$\bar{j} = \frac{1}{\Delta t}\int_t^{t+\Delta t} d\tau\, j \approx \frac{1}{\Delta t}\sum_p \sum_{\nu=1}^{N_\nu} q_p v_p S(x - x_p)\Delta\tau^\nu$$

- With these definitions, exact energy conservation is recovered:

$$\sum_p \sum_\nu \frac{m_p}{2}(v_p^{\nu+1} + v_p^\nu)(v_p^{\nu+1} - v_p^\nu) = -\sum_i \epsilon_0 \frac{E^{n+1} - E^n}{\Delta t}\frac{E_i^{n+1} + E_i^n}{2}$$

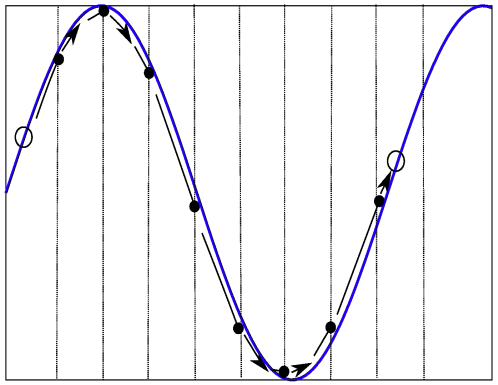
$$\Rightarrow \sum_p \frac{1}{2} m_p v_p^2 + \sum_i \frac{1}{2}\epsilon_0 E_i^2 = \text{const.}$$

Luis Chacon, chacon@lanl.gov

Exact charge conservation: charge-conserving particle mover

- Local charge conservation (enforced in the continuum by Gauss' law) is violated in discrete Vlasov-Ampère formulation.
- Local charge conservation is essential to ensure long-term accuracy of numerical algorithm
- Exact charge conservation requires a particle mover that satisfies a discrete charge continuity equation, $\partial_t \rho + \nabla \cdot \mathbf{j} = 0$ [Buneman 1968, Morse and Nielson, 1971]
  - ⇨ **Standard strategy** based on current redistribution when particle crosses boundary.
  - ⇨ In our context, current redistribution breaks energy conservation. **Need new strategy**.

Here, charge conservation is enforced by stopping particles at cell boundaries.



$$\left.\begin{aligned} \rho_{i+\frac{1}{2}} &= \sum_p q_p \frac{S_m(x - x_{i+\frac{1}{2}})}{\Delta x} \\ j_i &= \sum_p q_p v_p \frac{S_{m-1}(x - x_i)}{\Delta x} \\ S'_m(x) &= \frac{S_{m-1}(x+\frac{\Delta x}{2}) - S_{m-1}(x-\frac{\Delta x}{2})}{\Delta x} \end{aligned}\right\} \overset{(m=1,2)}{\Longrightarrow} \left[\partial_t \rho + \nabla \cdot \mathbf{j} = 0\right]_{i+\frac{1}{2}}^{n+\frac{1}{2}} = 0$$

Los Alamos NATIONAL LABORATORY EST. 1943

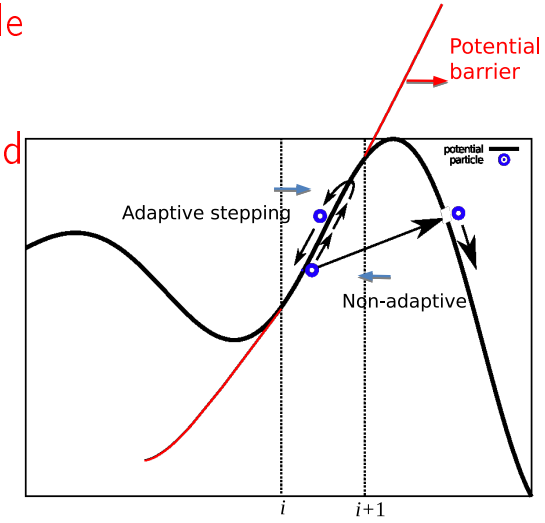
Luis Chacon, chacon@lanl.gov

Momentum conservation: adaptive orbit integrator

- EC/CC PIC algorithm does not enforce momentum conservation exactly.
  - **Controlling error** in momentum conservation is **crucial** for long-term accuracy
- Orbit integration errors can significantly affect momentum conservation: **particle tunneling**

- Adaptive orbit integration can be effective in suppressing particle tunneling and thus improve momentum conservation
- **Approach**: find $\Delta\tau$ to control local truncation error. Second order estimator gives:

$$\Delta\tau \leq \sqrt{12\epsilon_r \frac{m_p}{q_p} \left|\frac{dE}{dx}\right|_p^{-1}}$$

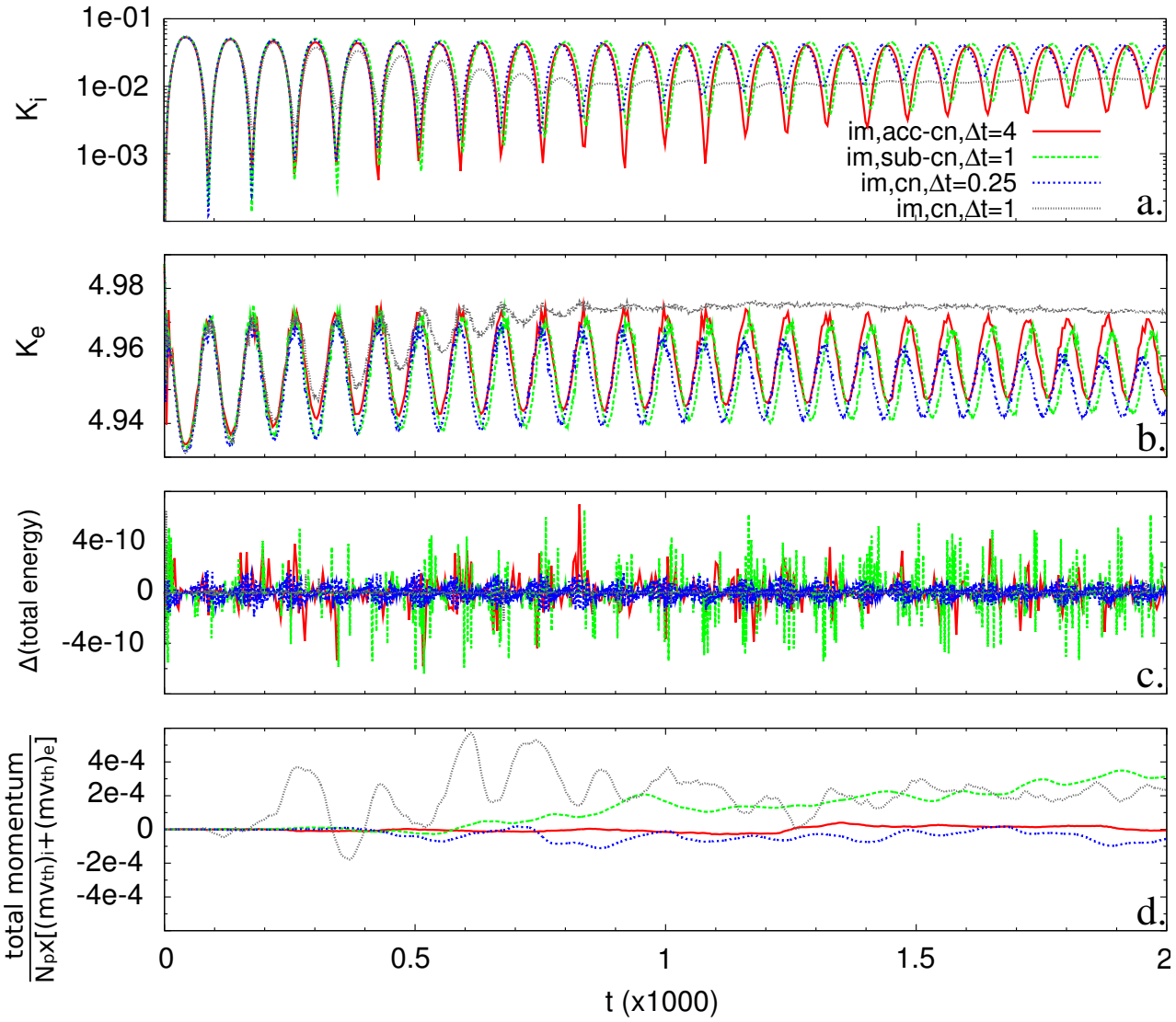


- Electric field gradient is estimated from cell-based gradient: $\frac{\partial E}{\partial x}\Big|_p \approx \frac{E_{i+1}-E_i}{\Delta x}$. **Provides potential barrier!**

- **Particle is stopped at cell boundaries** to ensure charge conservation.

Luis Chacon, chacon@lanl.gov

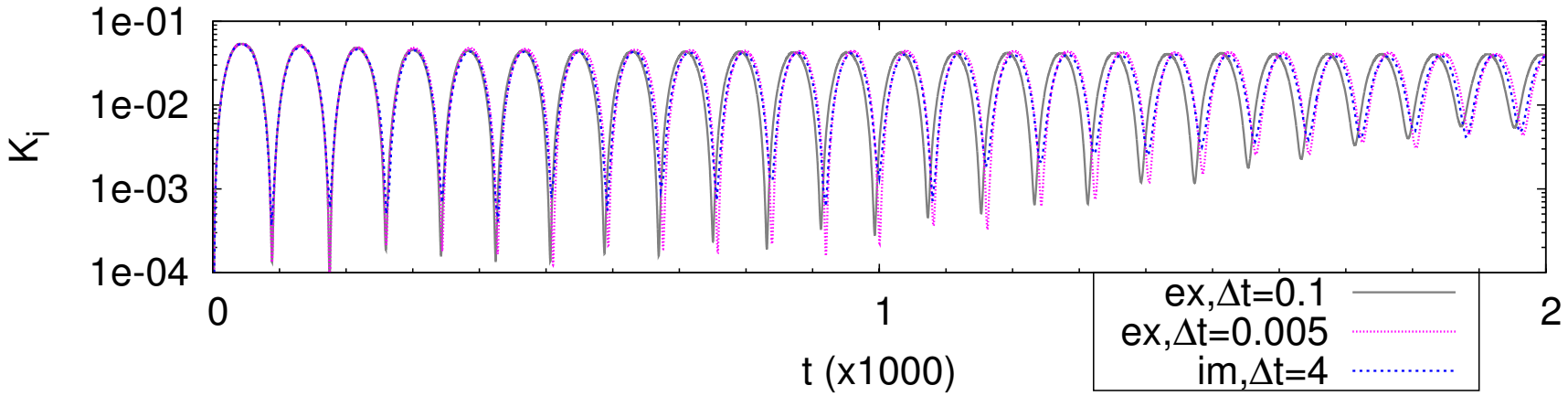
Ion acoustic wave (IAW): accuracy impact of different EC movers



Luis Chacon, chacon@lanl.gov

IAW: explicit vs. implicit (accuracy)

- Compare large-time-step implicit IAW vs explicit at CFL
- Found that explicit at CFL was not as accurate as implicit with $\Delta t \gg \Delta t_{CFL}$!!!

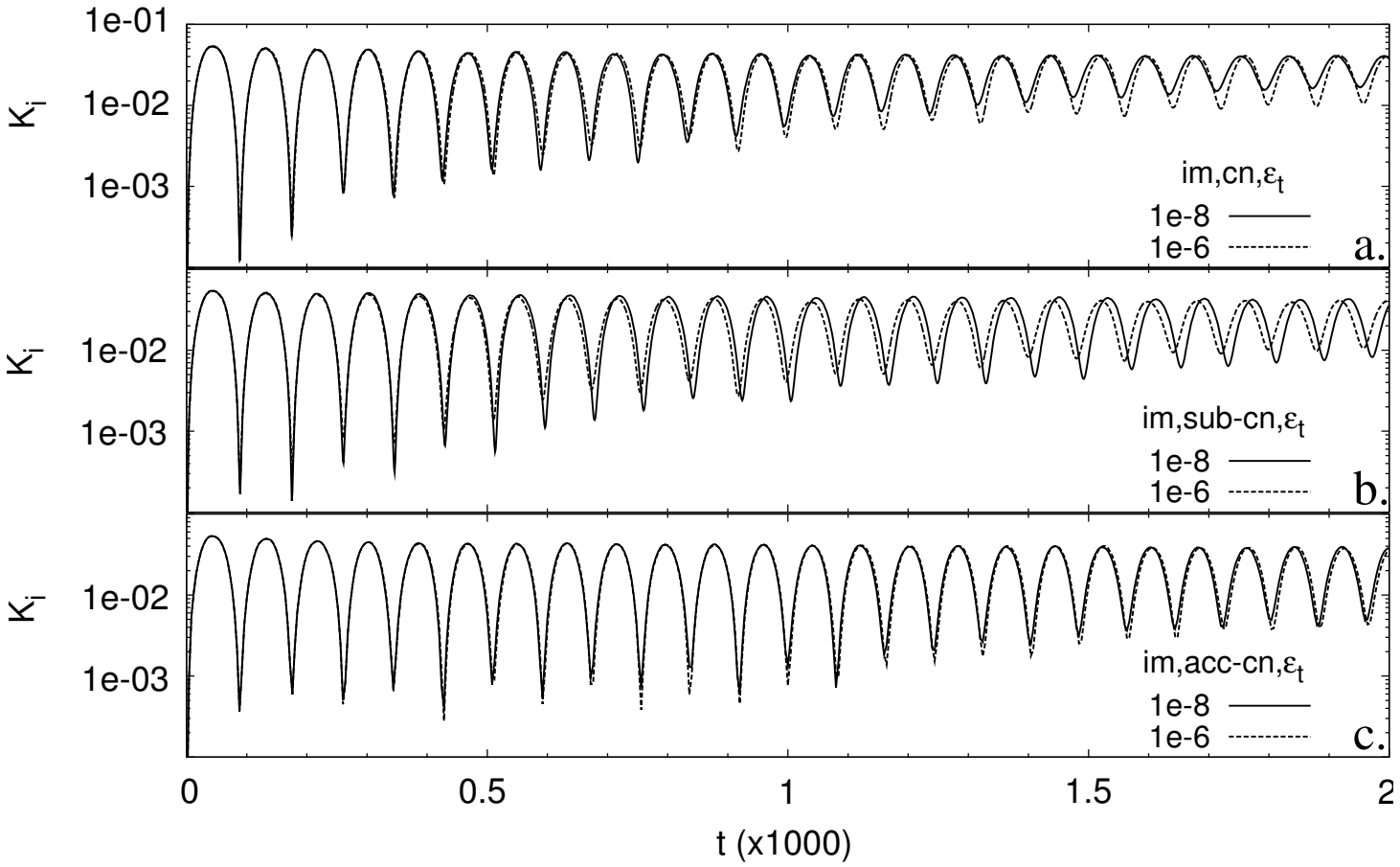


- CFL time-step is an "average" quantity (based on thermal velocity), and thus may still introduce inaccuracies in fast particles.

Luis Chacon, chacon@lanl.gov

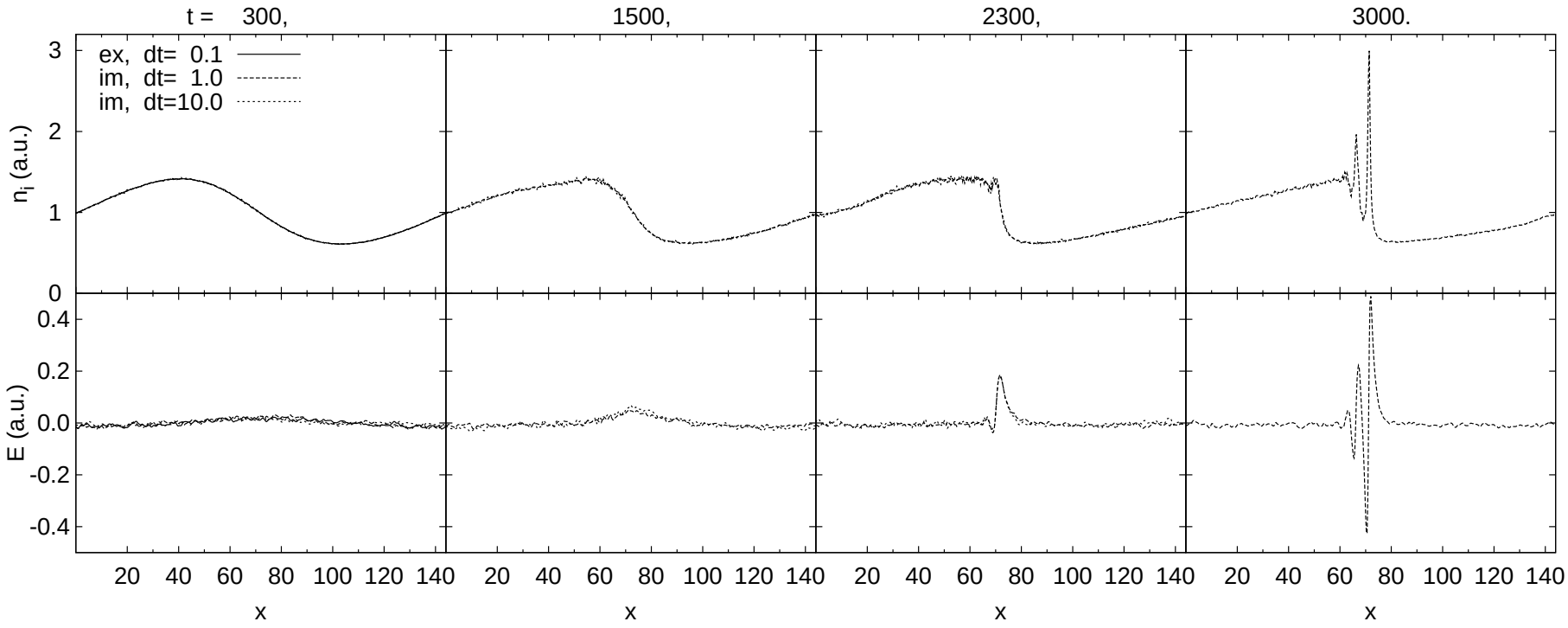
IAW: effect on nonlinear tolerance

- Exact energy conservation of implicit mover only holds for exact nonlinear solve
- It is of interest to understand robustness of mover when employing finite nonlinear tolerances



Adaptive-CC mover is the most robust!

Ion acoustic shock wave



- Propagating IAW with perturbation level $\epsilon = 0.4$, with 4000 particles/cell.
- Realistic mass ratio ($m_i/m_e = 2000$).
- Shock wave length scale$\sim$Debye length.

Los Alamos NATIONAL LABORATORY EST.1943

Luis Chacon, chacon@lanl.gov

CPU gain potential of implicit PIC vs. explicit PIC

- Back-of-the-envelope estimate of CPU gain:

$$CPU \sim \left(\frac{T}{\Delta t}\right)\left(\frac{L}{\Delta x}\right)^d n_p C^{solver} \quad ; \quad \frac{C^{imp}}{C^{ex}} \sim N_{FE}\frac{\Delta t_{imp}}{\Delta \tau_{imp}} \quad ; \quad \frac{CPU_{ex}}{CPU_{imp}} \sim \left(\frac{\Delta x_{imp}}{\Delta x_{ex}}\right)^d \frac{\Delta \tau_{imp}}{\Delta t_{ex}}\frac{1}{N_{FE}}$$

- Using reasonable estimates:

$$\begin{aligned} \Delta\tau_{imp} &\sim 0.1\frac{\Delta x_{imp}}{v_{th}} \\ \Delta t_{exp} &\sim 0.1/\omega_{pe} \\ k\Delta x_{imp} &\sim 0.2 \\ \Delta x_{ex} &\sim \lambda_D \end{aligned}$$

$$\frac{CPU_{ex}}{CPU_{imp}} \sim \frac{1}{(k\lambda_D)^{d+1}}\frac{1}{N_{FE}}$$

L	$k\lambda_D$	$\frac{N_x^{ex}}{N_x^{im}}$	$\frac{\Delta t_{im}}{\Delta t_{ex}}$	N_{FE}	$\frac{CPU_{ex}}{CPU_{im}}$
10	0.628	1	50	13.7	0.25
20	0.314	2	100	20	0.58
40	0.157	4	200	31.2	0.95
80	0.078	8	200	35.8	2.18
160	0.039	16	200	43.6	5.41
160	0.039	16	400	72.1	3.64
320	0.02	32	200	49.6	15.4
320	0.02	32	400	67.6	11.96

Luis Chacon, chacon@lanl.gov

Energy conserving implicit PIC on mapped meshes

Los Alamos NATIONAL LABORATORY EST.1943

Luis Chacon, chacon@lanl.gov

Generalization of implicit PIC algorithm to mapped meshes

- Implicit algorithm is most advantageous when resolution is coarse ($\Delta x \gg \lambda_D$).
- However, some problems develop thin layers nonlinearly (e.g. IASW) $\Rightarrow$spatial adaptivity.
- Here, we explore spatial adaptivity via a map $\mathbf{x}(\boldsymbol{\xi})$.
- Issues:
  - Presence of self-forces.
  - Particle deposition for charge conservation.
  - How should particles be pushed (logical space, physical space)?
- Properties of our implementation:
  - We recover energy and charge conservation theorems.
  - Key to the approach is a hybrid particle push [Swift, 1996; Wang, 1999]:
    - Position is updated in logical space (Cartesian-like)
    - Velocity is updated in physical space (no inertial forces due to geometry).

Formulation of equations in mapped geometry

$$\frac{\partial f_s}{\partial t} + \mathbf{v}\cdot\nabla f_s + \frac{q_s}{m_s}\mathbf{E}\cdot\nabla_v f_s = 0, \quad \longrightarrow \quad \frac{d\mathbf{x}_p}{dt} = \mathbf{v}_p \; ; \; \frac{d\mathbf{v}_p}{dt} = \frac{q}{m}\mathbf{E}_p.$$

$$\epsilon_0 \partial_t \mathbf{E} + \mathbf{j} = \nabla \times \mathbf{A}.$$

$\mathbf{x}(\xi)$ $\qquad$ $\boldsymbol{\alpha} = \nabla\xi$

1. Solve for contravariant field components.
2. Hybrid push: logical space physical velocity
3. Constant $\boldsymbol{\alpha}$ (contravariant base vector) per cell.
4. $J = |\partial \mathbf{x}/\partial \boldsymbol{\xi}|$ is the Jacobian.

$$\epsilon_0 \partial_t E^\alpha + j^\alpha = \frac{1}{J}\epsilon_{\alpha\beta\gamma}\partial_\beta A_\gamma.$$

$$\frac{d\xi_{\alpha,p}}{dt} = \mathbf{v}_p \cdot \boldsymbol{\alpha}_p,$$

$$\frac{d\mathbf{v}_p}{dt} = \frac{q}{m} E_{\alpha,p}\boldsymbol{\alpha}_p.$$

$E_\alpha = g_{\alpha\beta}E^\beta$

Luis Chacon, chacon@lanl.gov

Charge conservation theorem in mapped geometry

- Charge conservation equation in mapped geometry: $\partial_t(J\rho) + \partial_\alpha(J\, j^\alpha) = 0$
- Motivates following charge and current representations:

$$J\rho(\xi,t) = \frac{1}{\Delta\xi}\sum_p q_p S(\xi - \xi_p(t))$$

$$J\, j^\alpha(\xi,t) = \frac{1}{\Delta\xi}\sum_p q_p \mathbf{v}_p(t)\cdot\boldsymbol{\alpha}_p S(\xi - \xi_p(t))$$

- Shape functions interpolate charge and current, not their densities!
- Discrete charge conservation within a cell follows from:

$$\sum_p \frac{q_p}{\Delta\xi}\left[\frac{S_m(\xi_i - \xi_p^{n+1}) - S_m(\xi_i - \xi_p^n)}{\Delta t} + \frac{\xi_p^{n+1} - \xi_p^n}{\Delta t}\frac{S_{m-1}(\xi_{i+1/2} - \xi_p^{n+1/2}) - S_{m-1}(\xi_{i-1/2} - \xi_p^{n+1/2})}{\Delta\xi}\right] = 0,$$

  which is identical to Cartesian geometry form, and is an identity for $m = 1, 2$.
- Global charge conservation requires particles to land at cell boundaries, as in Cartesian case.

Discretization of PIC equations in mapped geometry

Equation of motion:

$$\frac{\xi_{\alpha,p}^{n+1} - \xi_{\alpha,p}^{n}}{\Delta t} = \mathbf{v}_p^{n+1/2} \cdot \boldsymbol{\alpha}_p^{n+1/2},$$

$$\frac{\mathbf{v}_p^{n+1} - \mathbf{v}_p^{n}}{\Delta t} = \frac{q}{m} E_{\alpha,p}^{n+1/2} \boldsymbol{\alpha}_p^{n+1/2}.$$

$$E_{\alpha,p}^{n+1/2} = \sum_i E_{\alpha,i+1/2}^{n+1/2} S_{m-1}(\xi_{i+1/2} - \xi_p^{n+1/2}).$$

$$(J\, j^{\alpha,n+1/2})_{i+1/2} = \frac{1}{\Delta\xi} \sum_p q_p \mathbf{v}_p^{n+1/2} \cdot \boldsymbol{\alpha}_p^{n+1/2} S_{m-1}(\xi_{i+1/2} - \xi_p^{n+1/2}).$$

Ampere's equation:

$$\epsilon_0 J_{i+1/2} \frac{E_{i+1/2}^{\alpha,n+1} - E_{i+1/2}^{\alpha,n}}{\Delta t} + (J\, j^{\alpha,n+1/2})_{i+1/2} = \langle j \rangle$$

Luis Chacon, chacon@lanl.gov

Energy conservation theorem in mapped geometry

- Start from equation of motion:

$$\frac{m_p}{2}\left[(v_p^{n+1})^2-(v_p^n)^2\right]=q_p E_{\alpha,p}^{n+1/2}\boldsymbol{\alpha}_p^{n+1/2}\cdot\mathbf{v}_p^{n+1/2}\Delta t.$$

- Summing over all particles:

$$K^{n+1}-K^n=\Delta t\sum_i \Delta\xi(J\,j^{\alpha,n+1/2}E_\alpha^{n+1/2})_{i+1/2}=-\epsilon_0\sum_i\Delta\xi J_{i+1/2}\left(\frac{(E_{i+1/2}^{n+1})^2}{2}-\frac{(E_{i+1/2}^{n})^2}{2}\right).$$

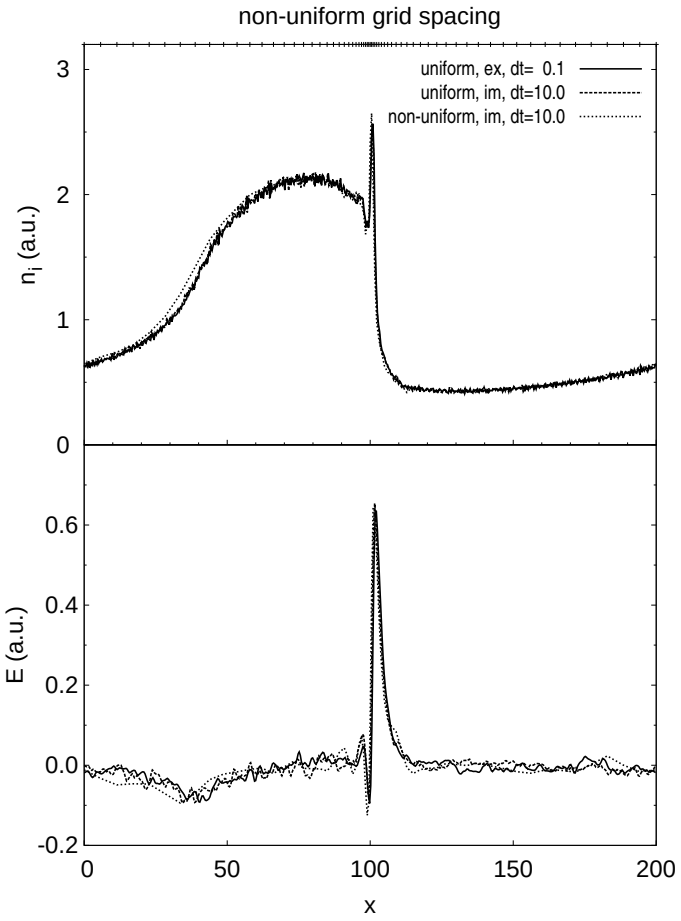
- As a result:

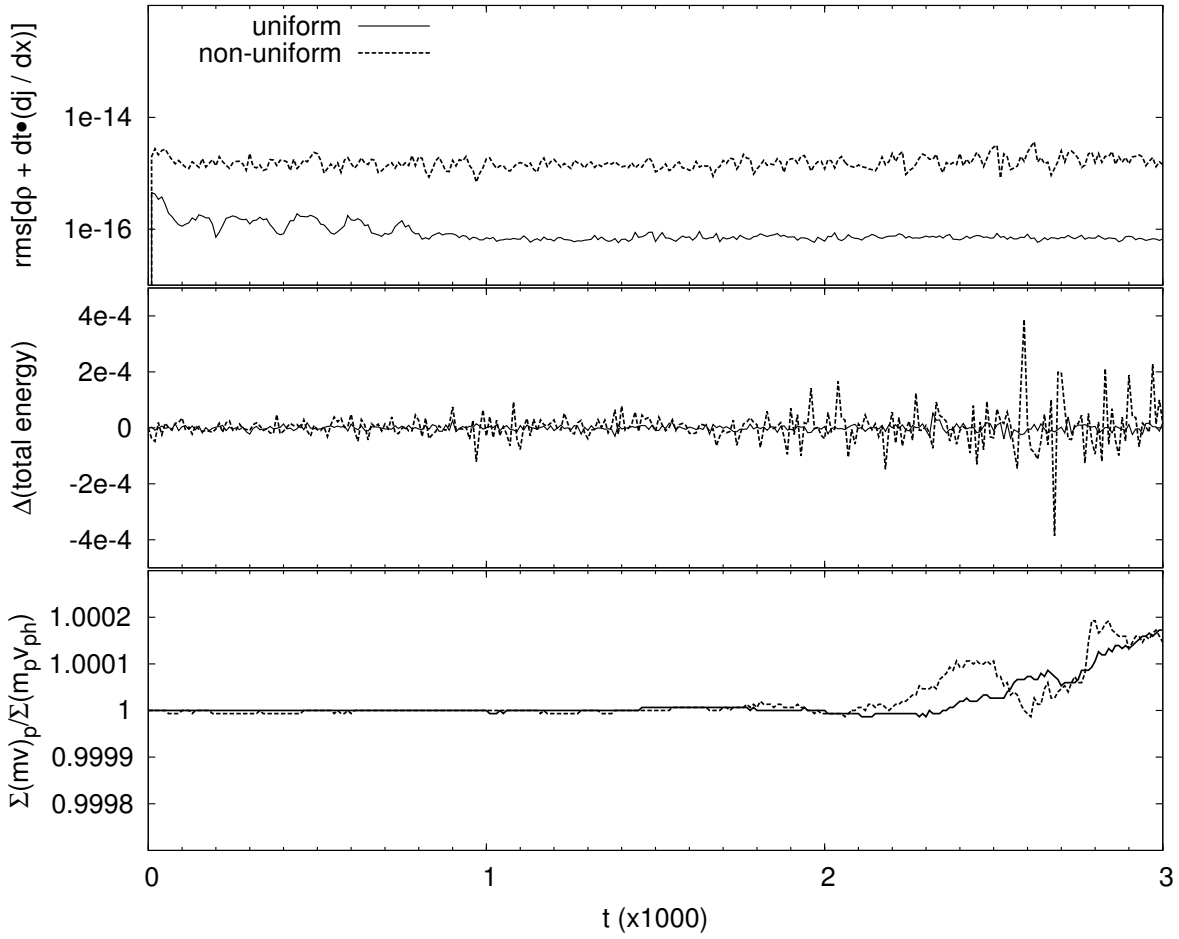
$$\left(\sum_p\frac{1}{2}m_p v_p^2+\frac{\varepsilon_0}{2}\sum_i\Delta\xi J_{i+1/2}E_{i+1/2}^2\right)\Bigg|_n^{n+1}=0.$$

Luis Chacon, chacon@lanl.gov

Ion acoustic shock wave test

non-uniform grid spacing





Los Alamos NATIONAL LABORATORY EST.1943

Luis Chacon, chacon@lanl.gov

Fluid preconditioning for fully implicit electrostatic PIC

Los Alamos NATIONAL LABORATORY EST.1943

Luis Chacon, chacon@lanl.gov

Moment-based acceleration of fully kinetic simulations

- Particle elimination formulates nonlinear residual in terms of fields/moments: $\mathbf{G}(\mathbf{E})$
- Preconditioner in JFNK needs to provide field/moment update: $\delta E \approx -P^{-1}\mathbf{G}$.
- Premise of acceleration: obtain δE from a fluid model, closed with current particle distribution

$$\begin{aligned}
\partial_t n_\alpha &= -\nabla \cdot \mathbf{\Gamma}_\alpha \\
m_\alpha \left[\partial_t \mathbf{\Gamma}_\alpha + \nabla \cdot \left(\frac{1}{n_\alpha}\mathbf{\Gamma}_\alpha \mathbf{\Gamma}_\alpha\right)\right] &= q_\alpha n_\alpha \mathbf{E} + \nabla \cdot \left(n_\alpha \left(\frac{\mathbf{\Pi}_\alpha}{n_\alpha}\right)_p\right) \\
\epsilon_0 \partial_t \mathbf{E} &= \sum_\alpha q_\alpha \mathbf{\Gamma}_\alpha
\end{aligned}$$

- Linearize:

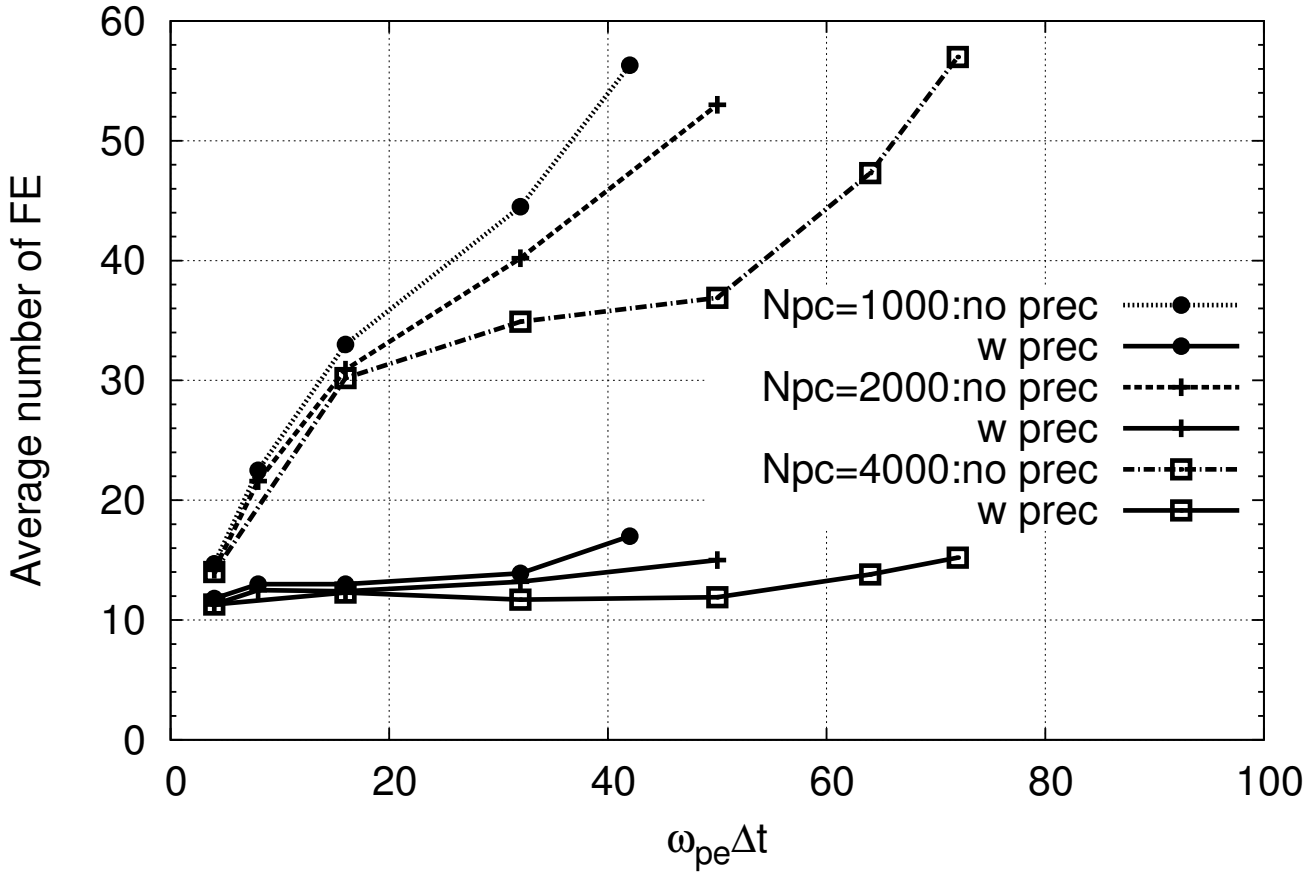
$$\begin{aligned}
\frac{\delta n_\alpha}{\Delta t} &= -\nabla \cdot \delta \mathbf{\Gamma}_\alpha \\
m_\alpha \left[\frac{\delta \mathbf{\Gamma}_\alpha}{\Delta t} + \nabla \cdot \delta\left(\frac{1}{n_\alpha}\mathbf{\Gamma}_\alpha \mathbf{\Gamma}_\alpha\right)\right] &\approx q_\alpha (\delta n_\alpha \mathbf{E} + n_\alpha \delta \mathbf{E}) + \nabla \cdot \left(\left(\frac{\mathbf{\Pi}_\alpha}{n_\alpha}\right)_p \delta n_\alpha\right) \\
\epsilon_0 \delta \mathbf{E} &= \Delta t \left[\sum_\alpha q_\alpha \delta \mathbf{\Gamma}_\alpha - \mathbf{G}(\mathbf{E})\right]
\end{aligned}$$

- δE can be obtained from $\mathbf{E}$, $\mathbf{G}(\mathbf{E})$, and particle closure $\left(\frac{\mathbf{\Pi}_\alpha}{n_\alpha}\right)_p$.

Los Alamos NATIONAL LABORATORY EST. 1943

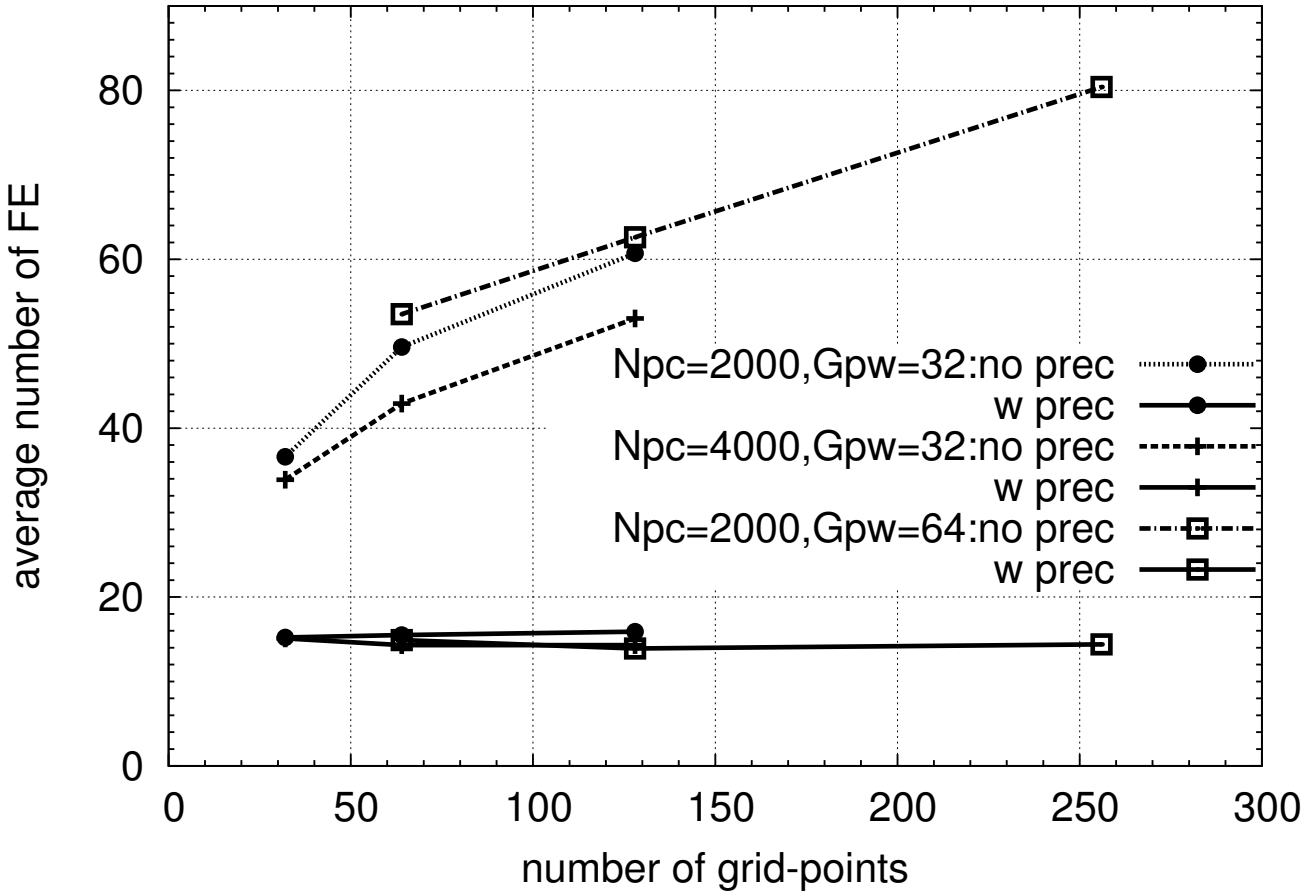
Luis Chacon, chacon@lanl.gov

Preconditioner performance with Δt



Luis Chacon, chacon@lanl.gov

Preconditioner performance with Δx

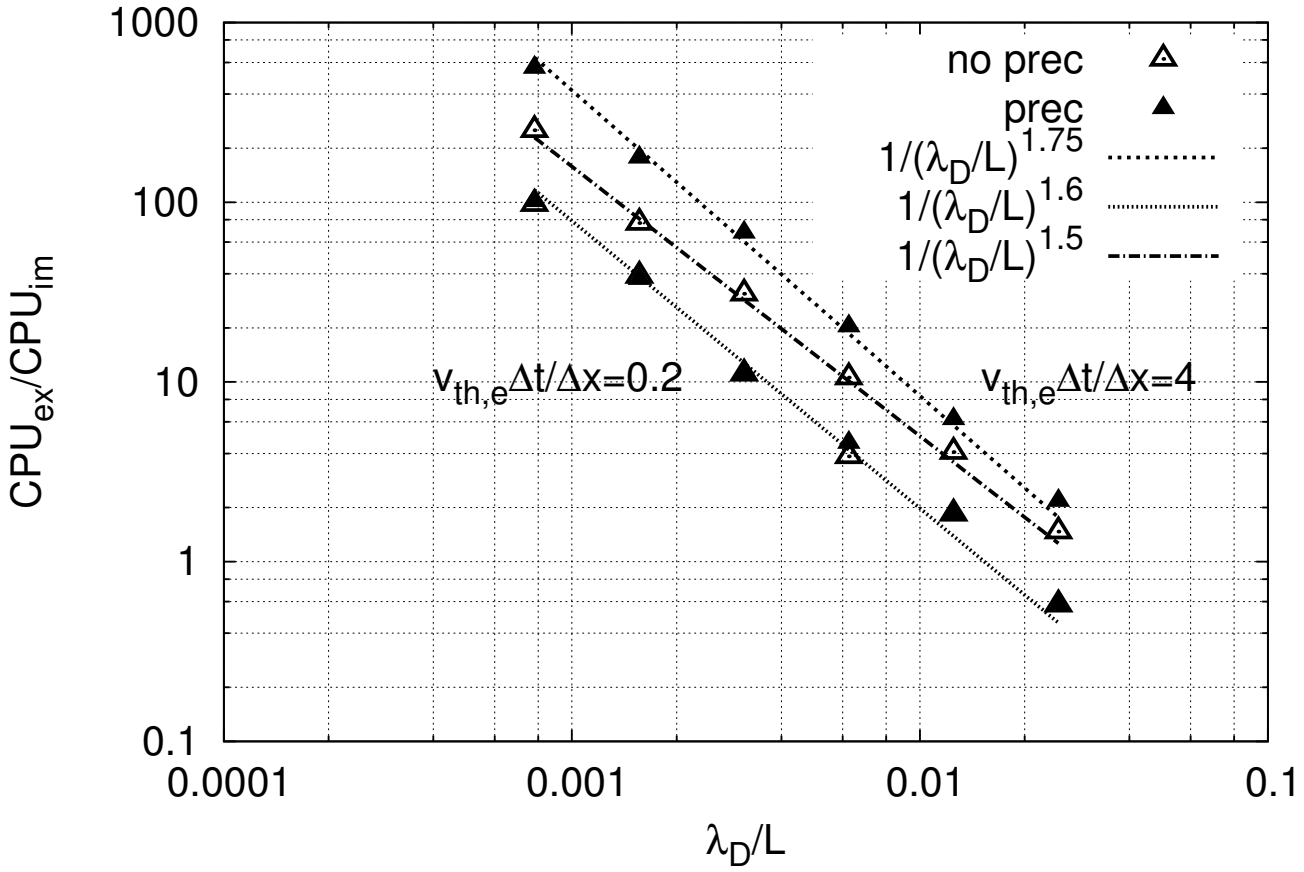


Los Alamos NATIONAL LABORATORY EST.1943

Luis Chacon, chacon@lanl.gov

Preconditioner performance: CPU scaling

$$\frac{CPU_{ex}}{CPU_{imp}} \sim \frac{1}{(k\lambda_D)^{d+1}} \frac{1}{N_{FE}}$$



Luis Chacon, chacon@lanl.gov

Generalization to electromagnetic PIC: Darwin (non-radiative) formulation

Los Alamos NATIONAL LABORATORY EST.1943

Luis Chacon, chacon@lanl.gov

Darwin approximation to Maxwell equations: motivation

- To analytically remove light-wave in non-relativistic plasma simulations *while preserving charge separation effects*
- If one keeps light wave with exact energy conservation in non-relativistic setting, one gets enhanced **numerical noise due to numerical Cherenkov radiation**

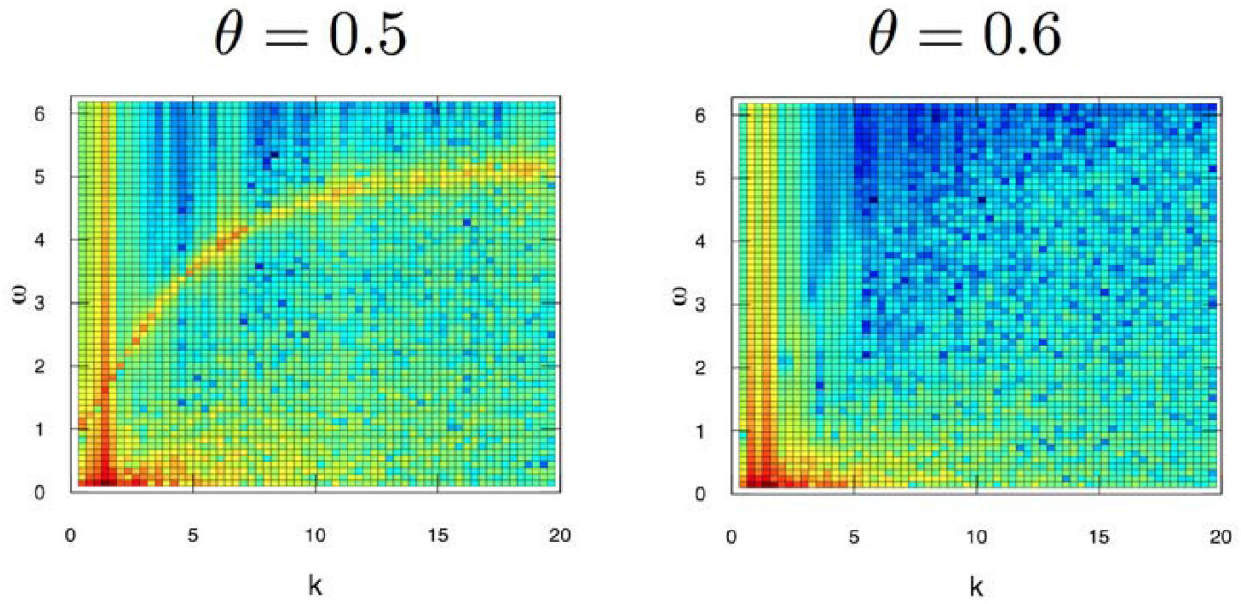


Figure 1: Fourier phase space for exactly energy conserving PIC (left) and dissipative PIC (right) [Markidis and Lapenta, JCP 2011].

Luis Chacon, chacon@lanl.gov

Darwin model (potential form)

- We consider potentials ϕ, $\mathbf{A}$ in the Coulomb gauge ($\nabla \cdot \mathbf{A} = 0$) such that:

$$\begin{aligned} \mathbf{B} &= \nabla \times \mathbf{A}. \\ \mathbf{E} &= -\nabla\phi - \partial_t \mathbf{A}. \end{aligned}$$

- Darwin model projects out the speed of light without enforcing quasineutrality (i.e., allowing for charge separation effects).

$$\begin{aligned} \nabla^2 \chi &= \nabla \cdot \mathbf{j}, \\ -\nabla^2 \mathbf{A} &= \mu_0 \left[\mathbf{j} - \nabla\chi\right], \\ \chi &= \epsilon_0 \partial_t \phi. \end{aligned}$$

- In 1D:

$$\begin{aligned} \epsilon_0 \partial_t E_x + j_x &= \langle j_x \rangle, \\ \frac{1}{\mu_0}\partial_x^2 A_y + j_y &= \langle j_y \rangle, \\ \frac{1}{\mu_0}\partial_x^2 A_z + j_z &= \langle j_z \rangle. \end{aligned} \qquad \begin{aligned} E_{y,i}^{n+1/2} &= -\frac{A_{y,i}^{n+1} - A_{y,i}^{n}}{\Delta t}, \\ E_{z,i}^{n+1/2} &= -\frac{A_{z,i}^{n+1} - A_{z,i}^{n}}{\Delta t}. \end{aligned}$$

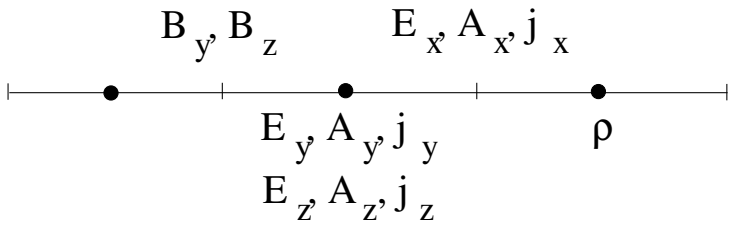
Energy conserving discrete 1D Darwin model

- Field equations:

$$\epsilon_0 \frac{E^{n+1}_{x,i+1/2} - E^{n}_{x,i+1/2}}{\Delta t} + \bar{j}^{n+1/2}_{x,i+1/2} = \langle j_x \rangle ,$$

$$\frac{1}{\mu_0}\, \partial_x^2 \frac{A^{n+1}_{y} + A^{n}_{y}}{2}\bigg|_i + \bar{j}^{n+1/2}_{y,i} = \langle j_y \rangle ,$$

$$\frac{1}{\mu_0}\, \partial_x^2 \frac{A^{n+1}_{z} + A^{n}_{z}}{2}\bigg|_i + \bar{j}^{n+1/2}_{z,i} = \langle j_z \rangle$$



- Current gather (with orbit averaging):

$$\bar{j}^{n+1/2}_{x,i+1/2} = \frac{1}{\Delta t \Delta x} \sum_p \sum_\nu q_p v^{\nu+1/2}_{p,x} S_m(x^{\nu+1/2}_p - x_{i+1/2}) \Delta\tau^\nu ,$$

$$\bar{j}^{n+1/2}_{y,i} = \frac{1}{\Delta t \Delta x} \sum_p \sum_\nu q_p v^{\nu+1/2}_{p,y} S_l(x^{\nu+1/2}_p - x_i) \Delta\tau^\nu ,$$

$$\bar{j}^{n+1/2}_{z,i} = \frac{1}{\Delta t \Delta x} \sum_p \sum_\nu q_p v^{\nu+1/2}_{p,z} S_l(x^{\nu+1/2}_p - x_i) \Delta\tau^\nu ,$$

Luis Chacon, chacon@lanl.gov

Implicit particle mover

- Subcycled particle equations of motion:

$$\begin{aligned}
\frac{x_p^{\nu+1} - x_p^{\nu}}{\Delta\tau^{\nu}} &= v_x^{\nu+1/2}, \\
\frac{\mathbf{v}_p^{\nu+1} - \mathbf{v}_p^{\nu}}{\Delta\tau^{\nu}} &= \frac{q_p}{m_p}\left(\mathbf{E}_p^{\nu+1/2}(x_p^{\nu+1/2}) + \mathbf{v}_p^{\nu+1/2} \times \mathbf{B}_p^{\nu+1/2}(x_p^{\nu+1/2})\right).
\end{aligned}$$

- This in an implicit nonlinear system. We invert it locally using Picard.
- Following Markidis and Lapenta [JCP 2011], we use an analytical inversion of the velocity equation

$$\begin{aligned}
\hat{\mathbf{v}}_p &= \mathbf{v}_p^{\nu} + \alpha \mathbf{E}_p^{\nu+1/2}\ ,\ \alpha = \frac{q_p \Delta\tau^{\nu}}{m_p 2} \\
\mathbf{v}_p^{\nu+1/2} &= \frac{\hat{\mathbf{v}}_p + \alpha\left[\hat{\mathbf{v}}_p \times \mathbf{B}_p^{\nu+1/2} + \alpha(\hat{\mathbf{v}}_p \cdot \mathbf{B}_p^{\nu+1/2})\mathbf{B}_p^{\nu+1/2}\right]}{1 + (\alpha B_p)^2}.
\end{aligned}$$

Final particle position and velocity are found from:

$$\begin{aligned}
x_p^{\nu+1} &= x_p^{\nu} + \Delta\tau^{\nu} v_{x,p}^{\nu+1/2}, \\
\mathbf{v}_p^{\nu+1} &= 2\mathbf{v}_p^{\nu+1/2} - \mathbf{v}_p^{\nu}.
\end{aligned}$$

Luis Chacon, chacon@lanl.gov

Field scatter to particles

- Electric field scatter (with orbit averaging):

$$
\begin{aligned}
E_{x,p}^{\nu+1/2} &= \sum_i \frac{E_{x,i+1/2}^{n+1} + E_{x,i+1/2}^{n}}{2} S_m(x_p^{\nu+1/2} - x_{i+1/2}), \\
E_{y,p}^{\nu+1/2} &= -\sum_i \frac{A_{y,i}^{n+1} - A_{y,i}^{n}}{\Delta t} S_l(x_p^{\nu+1/2} - x_i), \\
E_{z,p}^{\nu+1/2} &= -\sum_i \frac{A_{z,i}^{n+1} - A_{z,i}^{n}}{\Delta t} S_l(x_p^{\nu+1/2} - x_i).
\end{aligned}
$$

- Magnetic field scatter: conservation of canonical momenta in ignorable directions

$$\dot{p}_y = m_p \dot{v}_{p,y} + q_p \dot{A}_{p,y} = 0 \ , \ \dot{p}_z = m_p \dot{v}_{p,z} + q_p \dot{A}_{p,z} = 0$$

This can be enforced **exactly** along particle orbits, and yields:

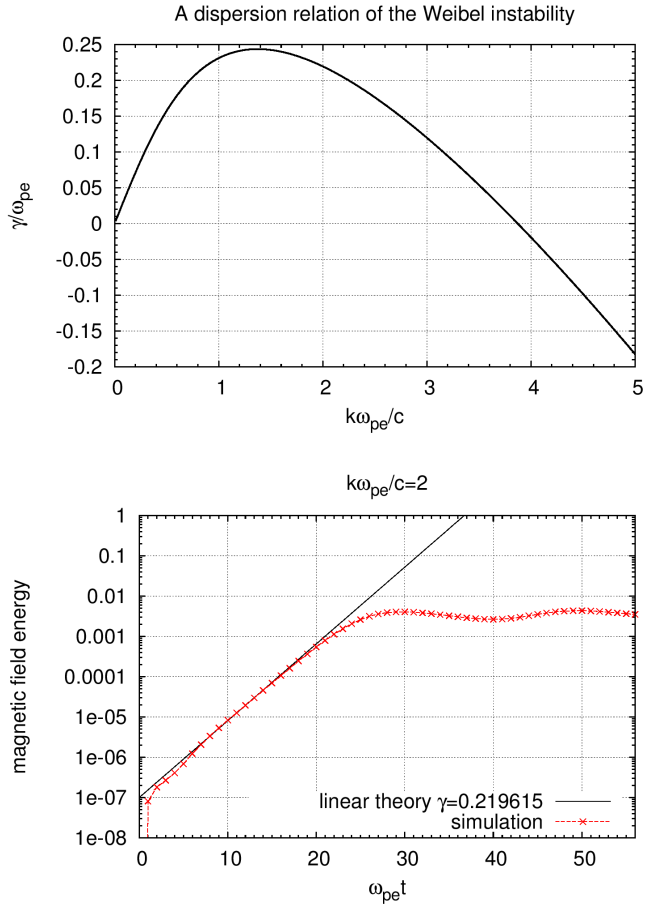
$$
\begin{aligned}
B_{y,p}^{\nu+1/2} &= -\sum_i \frac{A_{z,i+1}^{\nu+1/2} - A_{z,i}^{\nu+1/2}}{\Delta x} S_{l-1}(x_{i+1/2} - x_p^{\nu+1/2}) - \boxed{\sum_i \frac{\Delta A_{z,i-1}^{\nu} - 2\Delta A_{z,i}^{\nu} + \Delta A_{z,i+1}^{\nu}}{8} (x_p^{\nu+1} - x_p^{\nu})}, \\
B_{z,p}^{\nu+1/2} &= \sum_i \frac{A_{y,i+1}^{\nu+1/2} - A_{y,i}^{\nu+1/2}}{\Delta x} S_{l-1}(x_{i+1/2} - x_p^{\nu+1/2}) + \boxed{\sum_i \frac{\Delta A_{y,i-1}^{\nu} - 2\Delta A_{y,i}^{\nu} + \Delta A_{y,i+1}^{\nu}}{8} (x_p^{\nu+1} - x_p^{\nu})}.
\end{aligned}
$$

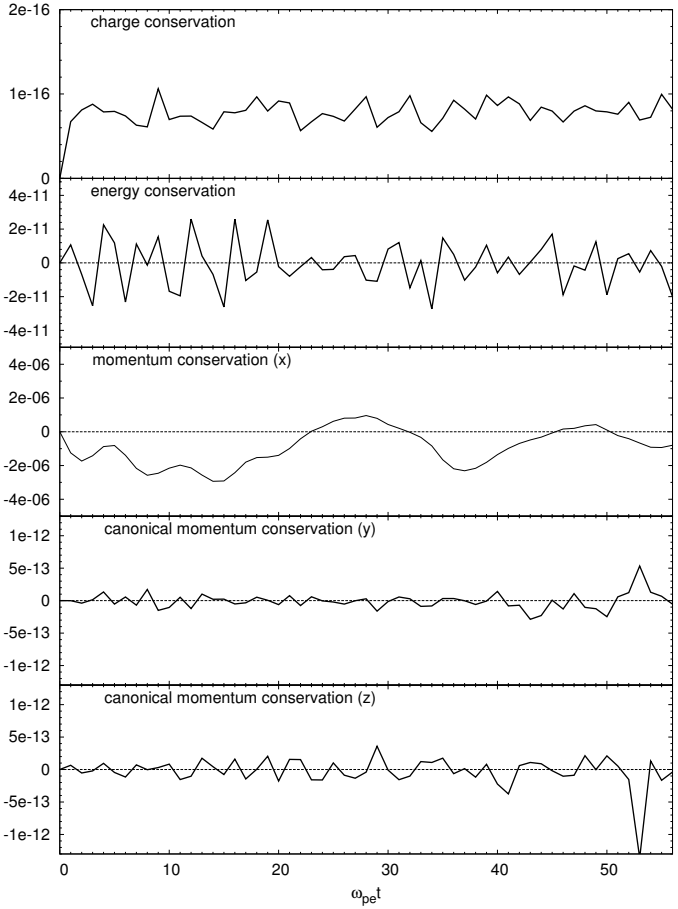
Luis Chacon, chacon@lanl.gov

Verification: Electron Weibel instability

- Isotropic ions, bi-Maxwellian electrons

$$m_i/m_e = 1836,\ T_{e\perp}/T_{e\parallel} = 16,\ N_{e,i}{=}128{,}000,\ L = 2\pi c/\omega_{pe},\ N_g{=}32.$$





Los Alamos NATIONAL LABORATORY EST. 1943

Luis Chacon, chacon@lanl.gov

Verification: Ion Weibel instability

- Isotropic electrons, bi-Maxwellian ions

$$m_i/m_e = 128,\ N_{e,i}{=}128{,}000,\ L = 0.88\pi c/\omega_{pi},\ N_g{=}32$$

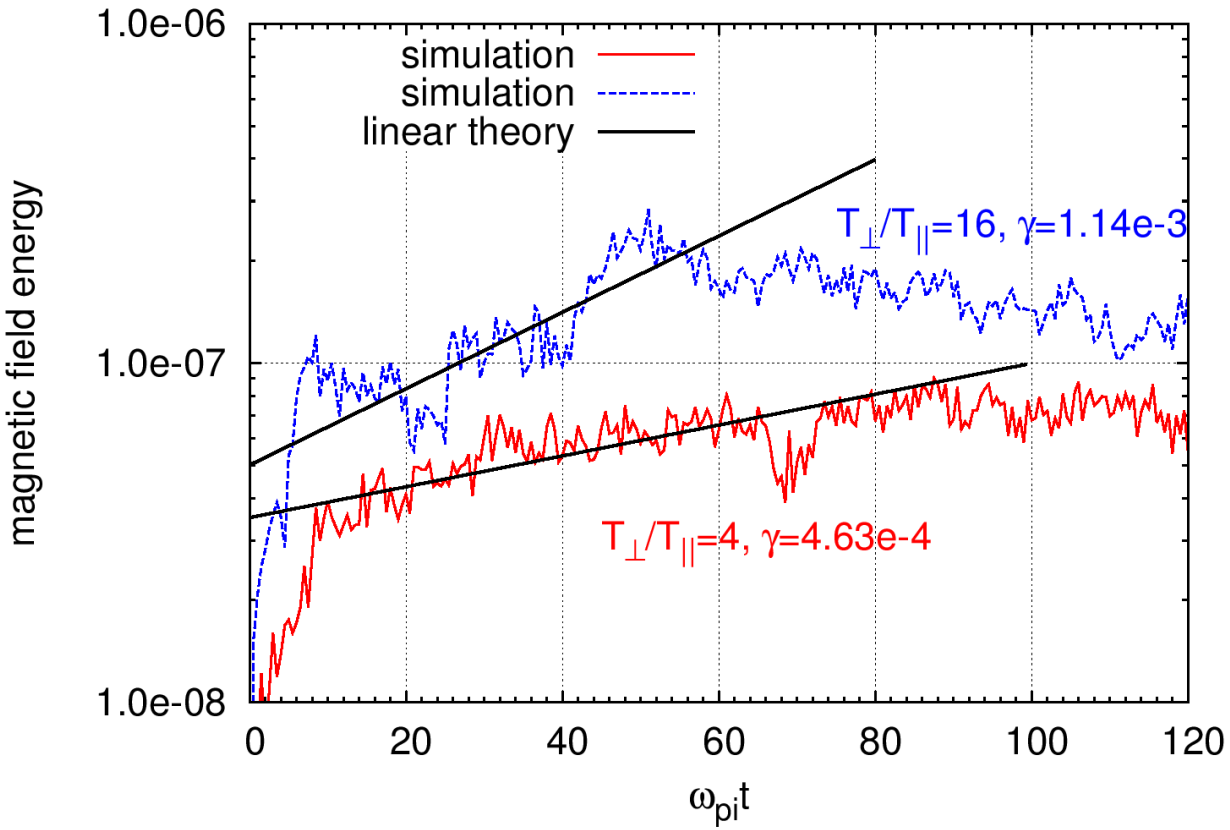


Figure 2:

Los Alamos NATIONAL LABORATORY EST.1943

Luis Chacon, chacon@lanl.gov

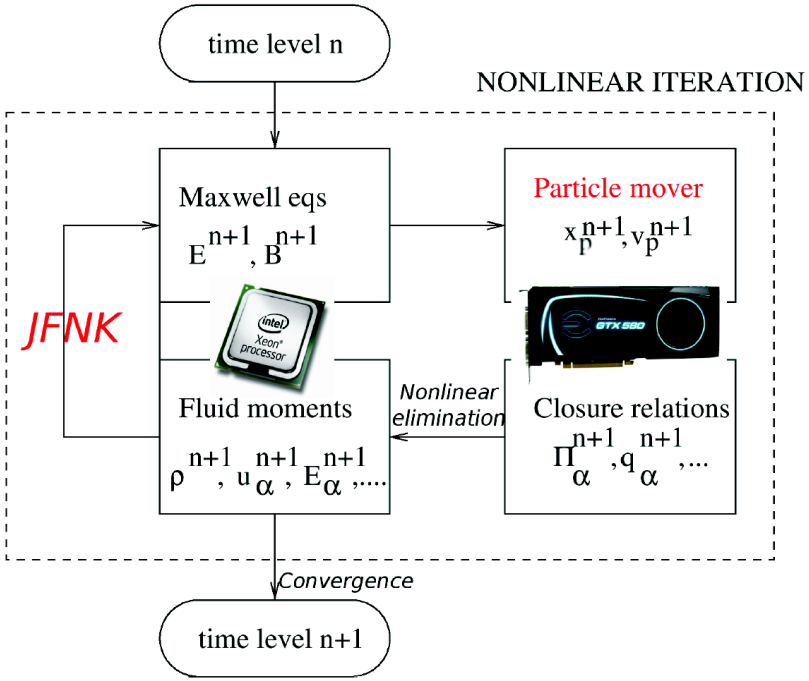
Hybrid CPU-GPU implementation (electrostatic PIC)

Los Alamos NATIONAL LABORATORY EST.1943

Luis Chacon, chacon@lanl.gov

Implementation of ACC particle mover on GPU architectures[1]

➤ Particle orbits are independent of each other ⇒ PIC algorithms are naturally data parallel.

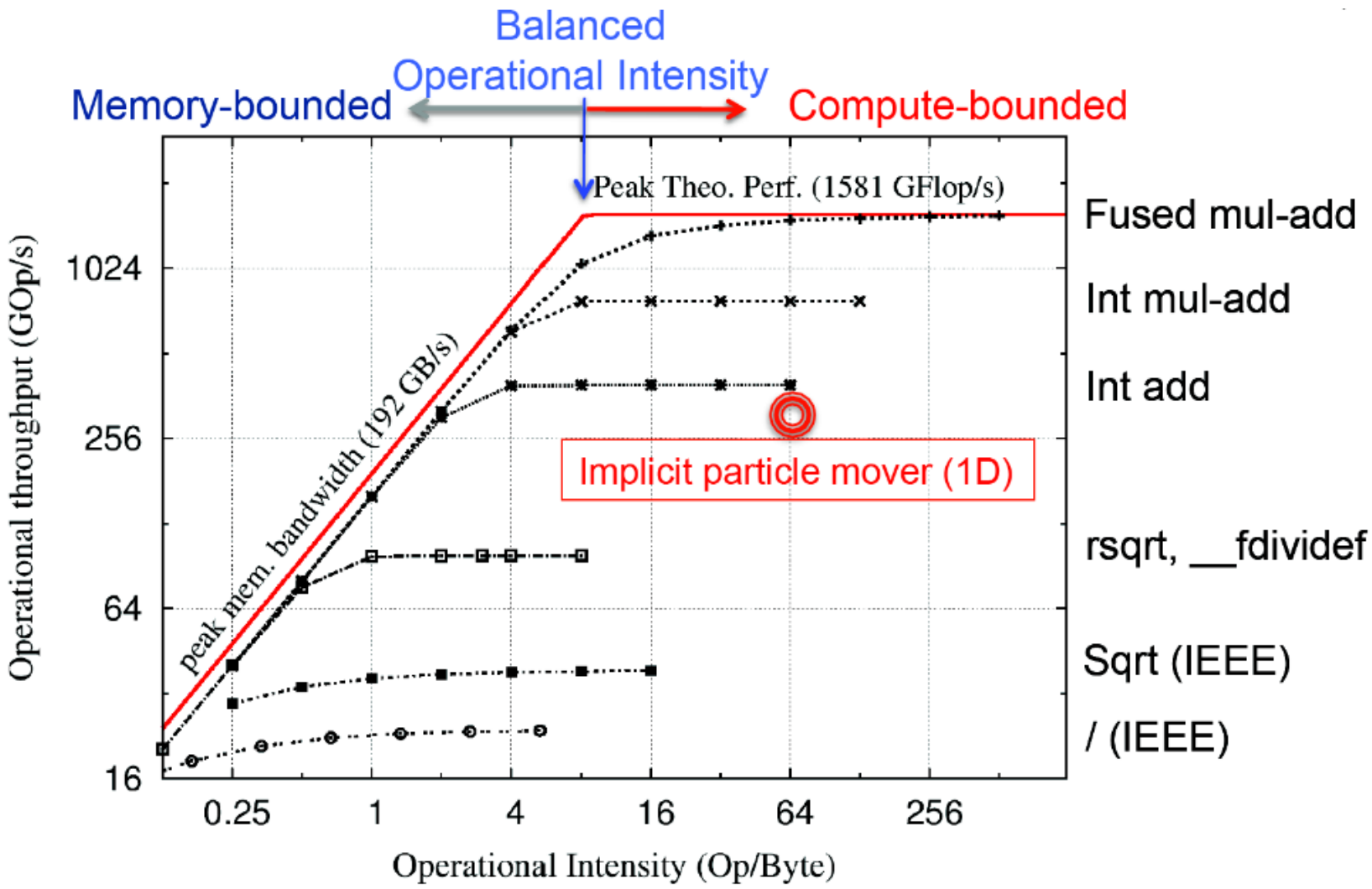


➤ Potential performance killers for our implicit PIC ACC particle mover:

- ⇨ Particle motion is **self-adaptive** (orbit accuracy) ⇒**workload imbalances**.
- ⇨ Particles **stop at cell boundaries** (charge conservation) ⇒**dynamic control flows**.

[1] Chen, Chacon, Barnes, JCP, 2012

Algorithm optimization on GPU: roofline model[2]

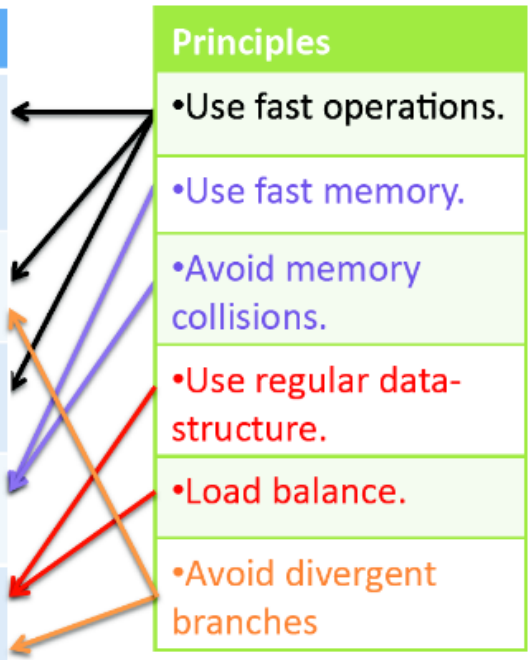


[2] S. Williams, A. Waterman, and D. Patterson, *Comm. ACM*, **52** (94) 2009

Optimization of ACC implicit particle mover

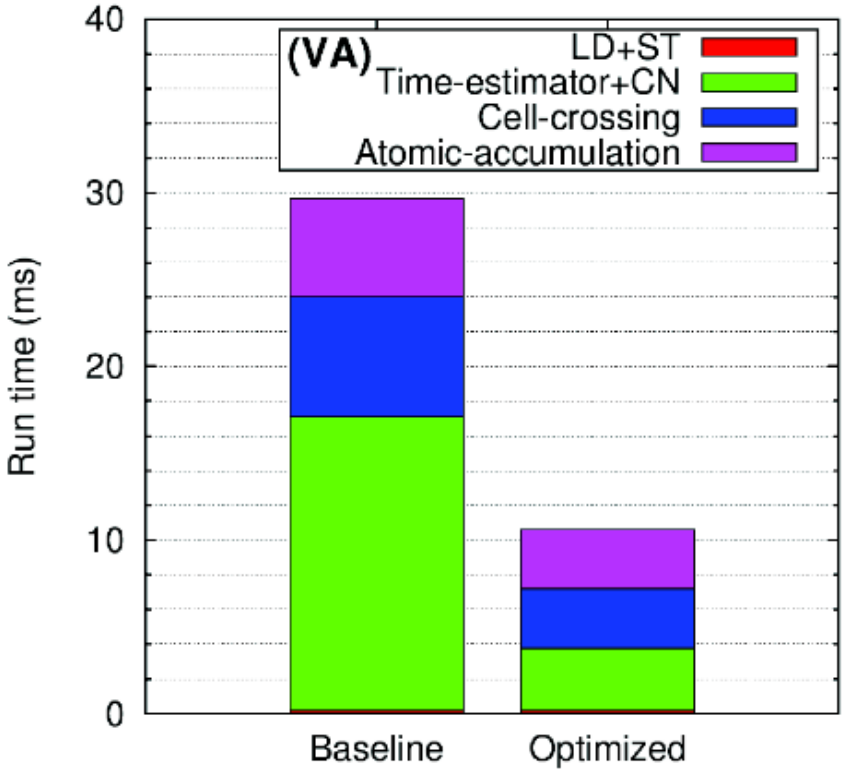
- Computationally intensive –> compute-bounded (vs. explicit schemes, typically memory-bounded)
- While loop introduces **control flow latencies and branch divergences**.
- Requires expensive operations **(sqrt, division)**, atomicAdd (for moment accumulation)

while(1) {	Original (baseline)	Optimized
Estimate sub-timestep	L2 norm, quadratic equation	L1 norm, split estimate w. abs and rel tol.
Crank-Nicolson update	Picard iteration	Direct solve using fast div + correction
Particle cell-crossing	Quadratic equation	Newton's method
Collect current(VA)	Shared→global	Register→ shared→global
If(dt_p==dt) break;		Particle sort; Warp vote.all
}; Collect charge(VP)		



Luis Chacon, chacon@lanl.gov

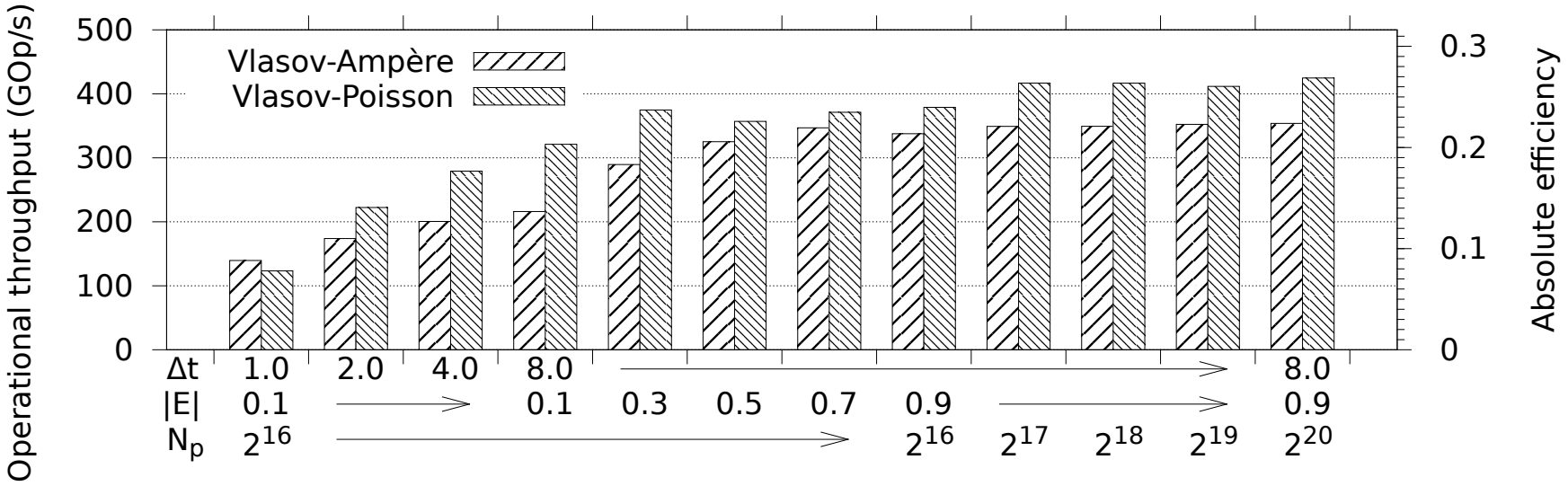
Performance results on GPU (single precision)



- Factor of 3 overall improvement after optimizations
  - ✓ Absolute efficiency 20-25%
    = real ops/Absolute theoretical peak (=1.6TGOps)
  - ✓ Intrinsic efficiency 50-70%
    = real ops/theoretical peak of the algorithm (~600GOps)
- Memory operations are negligible.
- Atomic accumulations are expensive in VA (negligible in VP).

Luis Chacon, chacon@lanl.gov

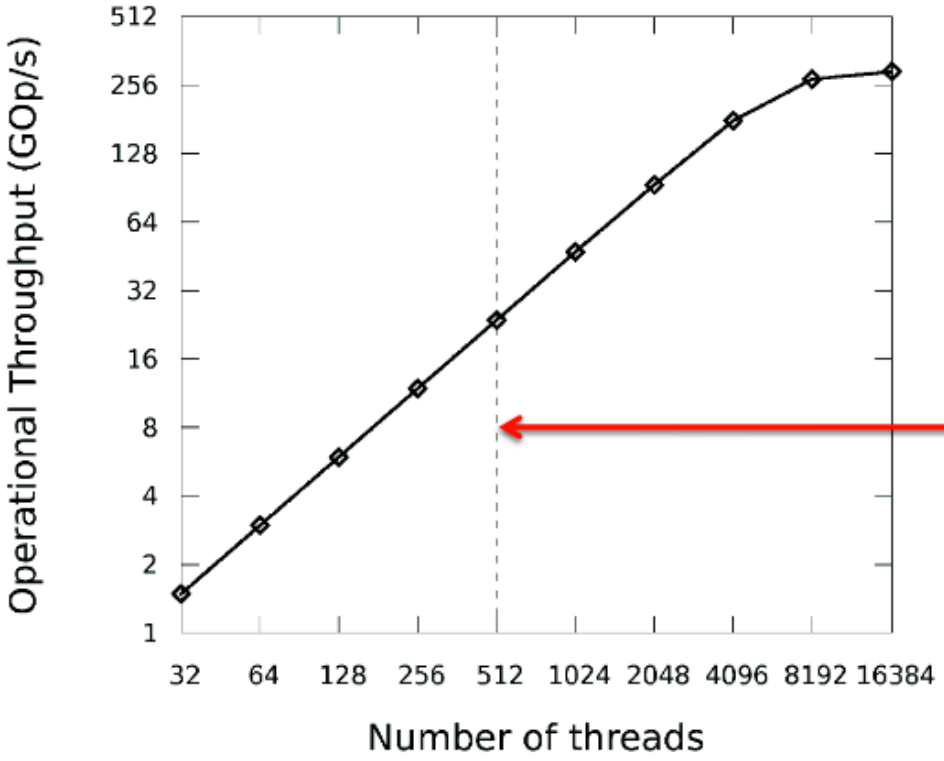
Sensitivity of GPU performance and efficiency



- All operations including floating, integer, and special functions are counted.
- Varied E, Δt, N_p to test performance sensitivity
  - ⇨ Performance is most sensitive to Δt: **more efficient for large Δt!**
- 300 to 400 GOps/s (20-30% efficiency of GPU peak) are obtained for large time steps, strong fields and many particles.

Luis Chacon, chacon@lanl.gov

GPU scaling with number of threads



Scale up to theoretical limit

Consistent with

Little's law:

Needed parallelism

= Latency × # CUDA cores

≈ 18 clock cycles

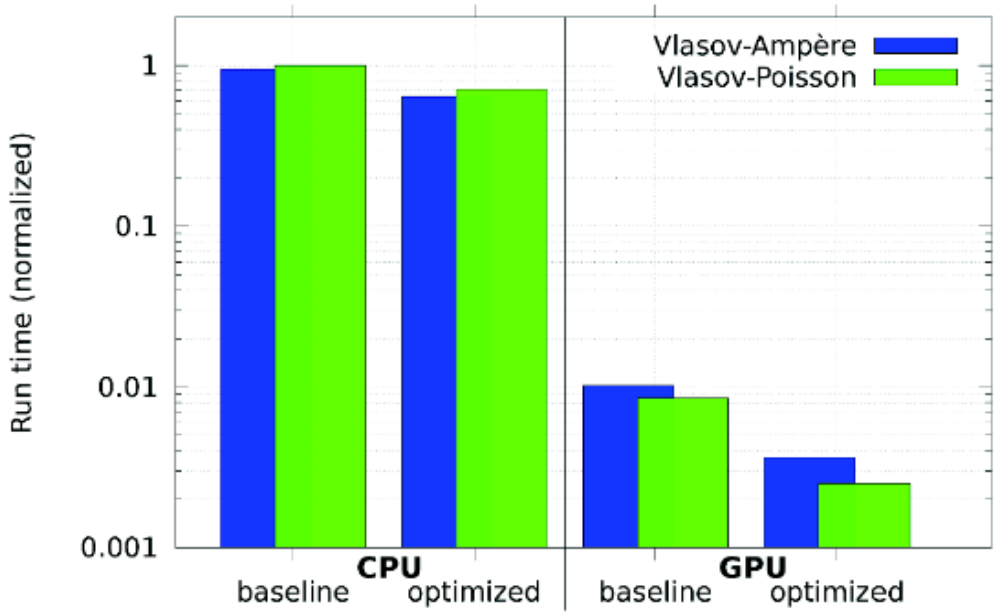
Instruction level latency (Fermi)

- Hardware limit is 512 threads (=32 cores/SMx16 SM/GPU) running concurrently;
- Large number of threads ($\gg$512) are useful to hide latencies.

CPU-GPU speedup

Intel Xeon
X5460@3.16GHz
Single-core
theoretical peak
performance (SP)
25.2 GFLOPS

CPU, Serial

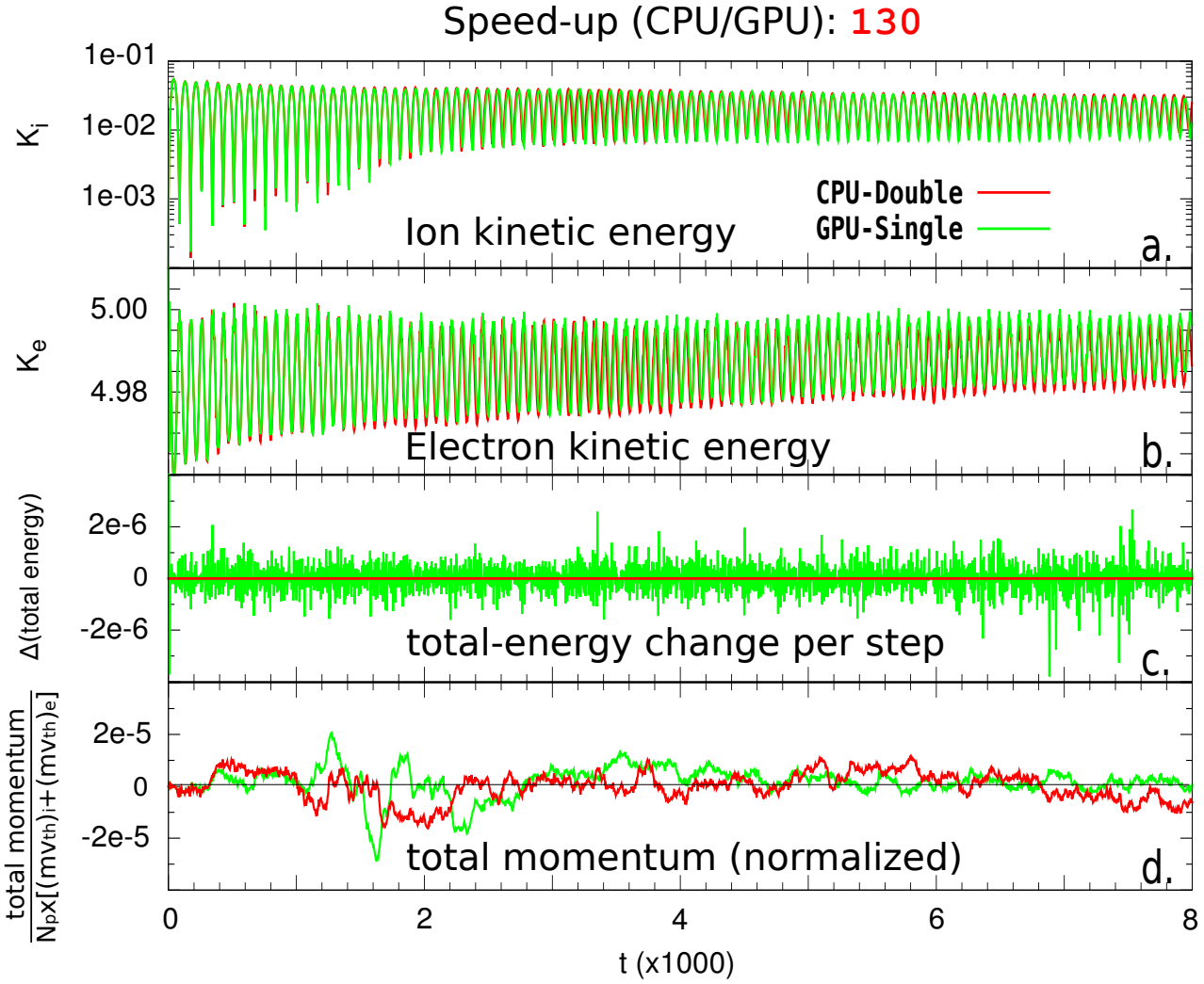


Nvidia Geforce
GTX 580@1.54GHz
many-core
theoretical peak
performance (SP)
1.58 TFLOPS

GPU, Parallel

- Straightforward GPU implementation accelerates ∼100 times;
- Optimizations have larger effects on GPU; not all optimizations introduced are effective on CPU.
- GPU-CPU speedup $\sim 200 - 300$, depending on algorithm (VA, VP)

Ion acoustic wave: accuracy and performance comparison



Los Alamos NATIONAL LABORATORY EST. 1943

Luis Chacon, chacon@lanl.gov

Summary and conclusions

- We have demonstrated, for the first time, a fully implicit, fully nonlinear electrostatic PIC formulation that features:
  - **Exact charge conservation** (via a novel particle mover strategy).
  - **Exact energy conservation** (no particle self-heating or self-cooling).
  - **Adaptive particle orbit integrator** to control errors in momentum conservation.
- The approach has been shown to be free of CFL and finite-grid numerical instabilities.
- As a result, the method is able to take time steps many times larger than explicit, and resolutions many times coarser.
- Central to our implementation is the concept of particle enslavement.
- We have generalized formulation to use spatial adaptivity via mapped coordinates.
- The method has much potential for efficiency gains vs. explicit in long-time-scale applications, with the CPU speedup scaling as $(k\lambda_D)^{-(d+1)}/N_{FE}$.
  - Minimize the number of nonlinear function evaluations N_{FE} for given Δt, $\Delta x \Rightarrow$ preconditioning!
  - We have formulated and implemented a very efficient moment-based preconditioner.
- We have ported the algorithm to GPU architectures
- We have generalized the algorithm to non-radiative electromagnetic regimes (Darwin model), where, **in addition to charge and energy, we also conserve canonical momenta**.