Particle-in-cell simulations of mirror mode structures in the magnetosphere

K. Germaschewski, N. Ahmadi, and J. Raeder

Space Science Center / Dept. of Physics
University of New Hampshire

July 2, 2013

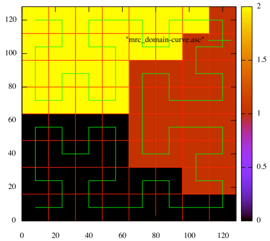
UNIVERSITY *of* NEW HAMPSHIRE

CICART

Outline

Plasma Simulation Code (PSC)

- 1D, 2D, 3D configuration space
- relativistic, electromagnetic
- boost frame, moving window, PMLs, collisions, ionization...
- modular architecture: switching from legacy Fortran particle pusher to GPU pusher can be done on the command line.
- support for modern hardware (GPUs, Intel MIC)



Color indicates the MPI process responsible for local domain.

Kinetic Kelvin-Helmholtz Instability

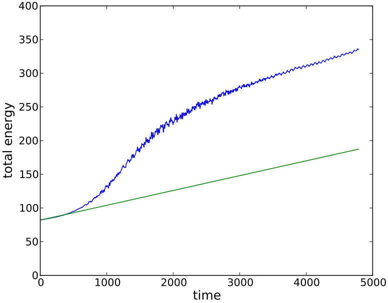
(Loading khc025_2.mp4)

15360 × 7680 cells, 100 particles per cell
performed on 900 GPUs (M2090, TitanDev) in $\approx$ 24 h wallclock

Particle-in-cell: Numerical Heating

Numerical Heating

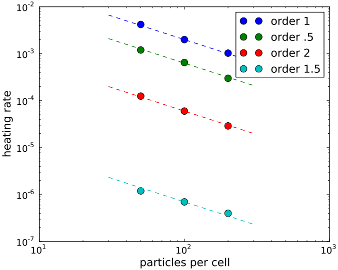
- **Finite Grid Instability.** Aliasing of unresolved grid modes gives rise to a numerical instability if the Debye length is not resolved.
- **Stochastic heating.** Particle noise leads to errors in the electromagnetic fields that heat the plasma linearly ($\propto 1/N$).



Numerical Heating: dependence on particle shape

Remedies: Use more particles, or use higher order particles.

Heating rate



Performance

(16-core AMD Opteron / Nvidia K20X)

pusher	performance
order 2/1.5	23 M/sec
order 1	59 M/sec
order 1 (single)	78 M/sec
order 1 (SSE2)	94 M/sec
order 1 (CUDA)	824 M/sec

PSC on GPUs

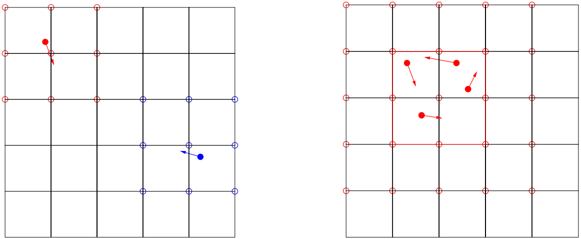
Multi-level decomposition of the problem, expose parallelism

- At the top-level, decompose spatial domain into *patches*. Each MPI process gets assigned one or more patches. Patches communicate via ghost cells / particle exchange.
- (Hybrid level can be introduced here: Each MPI process will distribute patches onto a set of cores or GPUs using OpenMP / threads)
- GPU: Each patch gets further divided into *blocks* (a.k.a. supercells) of multiple cells. These blocks are handled (in parallel) by threadblocks.
- Particles in a block are processed in parallel by threads in the threadblock (GPU) / by SIMD instructions (CPU/MIC).

PSC on GPUs

Particle-in-cell algorithm

```
for timestep $n = 0,1,2,...$:
    for each particle $m$:
        advance momentum: $\vec{p}_m^n \rightarrow \vec{p}_m^{n+1}$
            (using interpolated $\vec{E}^{n+1/2}, \vec{B}^{n+1/2}$)
        advance position: $\vec{x}_m^{n+1/2} \rightarrow \vec{x}_m^{n+3/2}$
        deposit current density contribution $\vec{j}_m^{n+1}$ onto mesh.
    advance fields: $\vec{E}^{n+1/2}, \vec{B}^{n+1/2} \rightarrow \vec{E}^{n+3/2}, \vec{B}^{n+3/2}$ using $\vec{j}^{n+1}$.
```

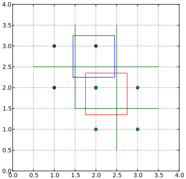


PSC on GPUs

Current deposition – 1st order Villasenor-Buneman

PSC used a 2nd order particle shape function and a corresponding charge-conserving current deposition scheme [Esirkepov 2001]. It calculates 40 contributions to the current density.

Using 1st order particle shape functions and the charge-conserving scheme proposed by [Villasenor, Buneman 1992], at most 10 points contribute.



Kernel	Performance [particles/sec]
1st V-B, sorted by cell	250×10^6
1st V-B, randomized by block	500×10^6

PSC on GPUs – TitanDev/BlueWaters Performance

16-core AMD 6274 CPU, Nvidia Tesla M2090 / Tesla K20X

Kernel	Performance [particles/sec]
2D push & V-B current, CPU (AMD)	130×10^6
2D push & V-B current, GPU (M2090)	565×10^6
2D push & V-B current, GPU (K20X)	710×10^6

For best performance, need to use GPU and CPU simultaneously. Patch-based load balancing enables us to do that: On each node, we have 1 MPI-process that has $\approx$ 45 patches that are processed on the GPU, and 15 MPI-processes that have 1 patch each that are processed on the remaining CPU cores.

PSC on GPUs – Sorting

Particles need to be kept sorted in order to make use of shared memory for caching fields and updating currents.

GPUs do best when“streaming” through particles, complicated data structures are often detrimental to highly threaded parallel performance.

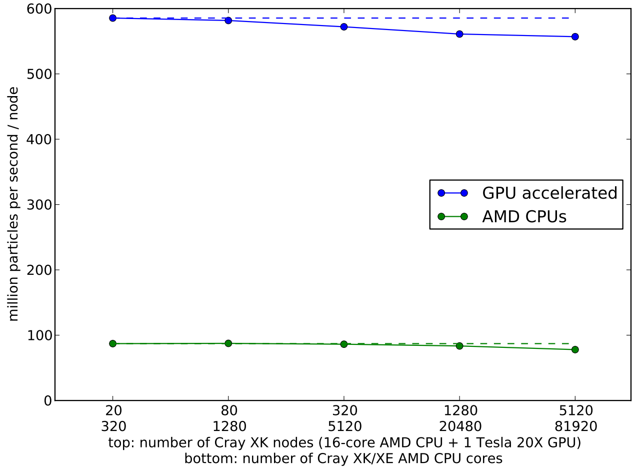
Our approach: Keep particle data in a streaming-friendly one-large-array layout (or SOA), sorted by block (a.k.a. “supercell”, a block of, e.g. 4×4 cells.)

Rearranging particles

1. Find block indices
2. Sort (block index, index) pairs
3. Use sort result to move each particle into its new position.

PSC on GPUs – Parallel Performance

Weak scaling study on Craxy XK7 “Titan” at ORNL.



Temperatur anisotropy instabilities

Ion temperature $T_\perp > T_\parallel$ anistropy in a plasma with background magnetic field can drive kinetic instabilities:

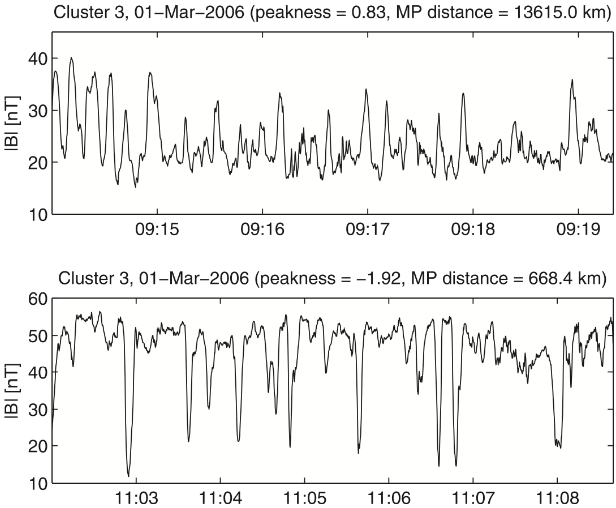
- ion cyclotron instability (propagating)
- mirror instability (standing)

Mirror mode instability condition

$$\frac{T_\perp}{T_\parallel} > 1 + 1/\beta_\perp$$

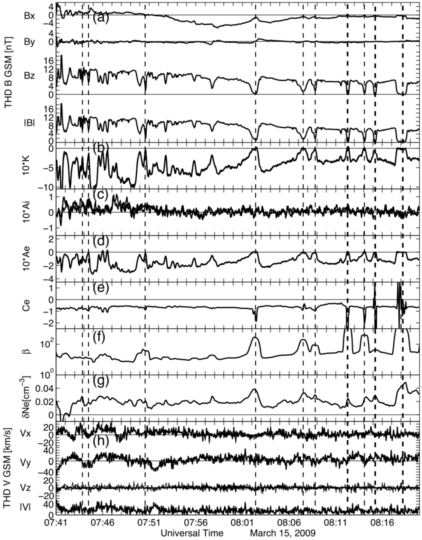
Mirror modes are observed in the solar wind, magnetosheath, and magnetosphere (near tail).

Mirror mode structures in the magnetosheath



Soucek et al., 2008

Mirror mode structures in the near tail



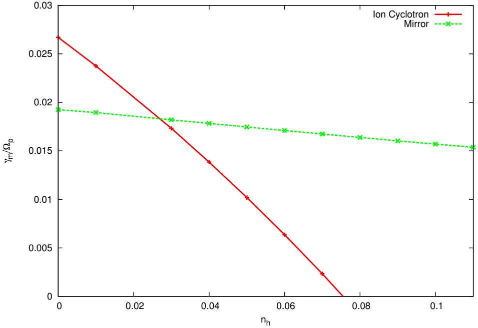
Ge et al., 2011

Mirror mode questions

- Why is the mirror mode observed, rather than ion cyclotron waves? (depends on β, Helium stabilizes IC, IC propagate away, 2-d vs 3-d)
- How does the mirror mode evolve nonlinearly? What determines peaks vs dips? (mirror mode stable vs unstable regions)
- What role do electrons play? (isotropic in mirror structures)
- What is the spatial extent of mirror mode structures? (observations: smaller than ion gyroradius, tens of electron radii)

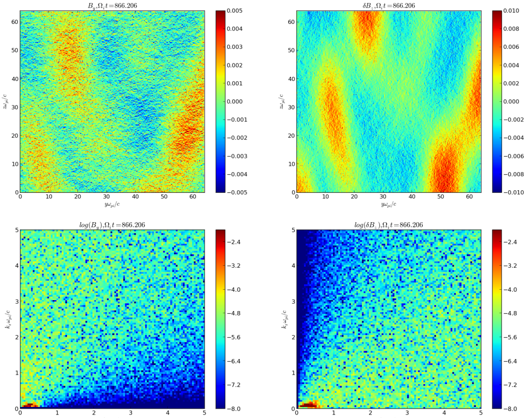
Linear growth rates in the presence of Helium

Maximum growth rates for $\beta_i = 4., \beta_e = 0.4,$
$\beta_h = 4., T_{\perp i}/T_{\parallel i} = 1.5,\ T_{\perp e}/T_{\parallel e} = 1,\ T_{\perp h}/T_{\parallel h} = 1.5,$
$m_h/m_i = 4.,$

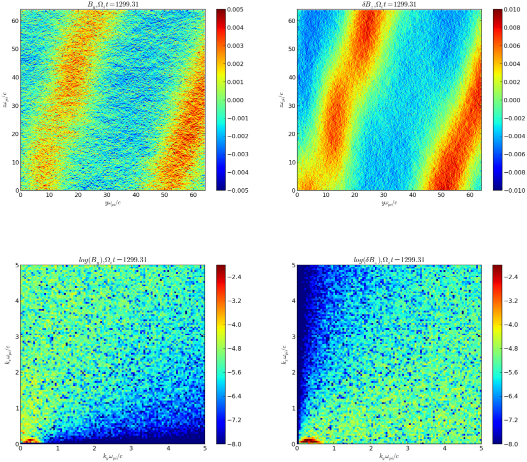


2-d PIC simulation

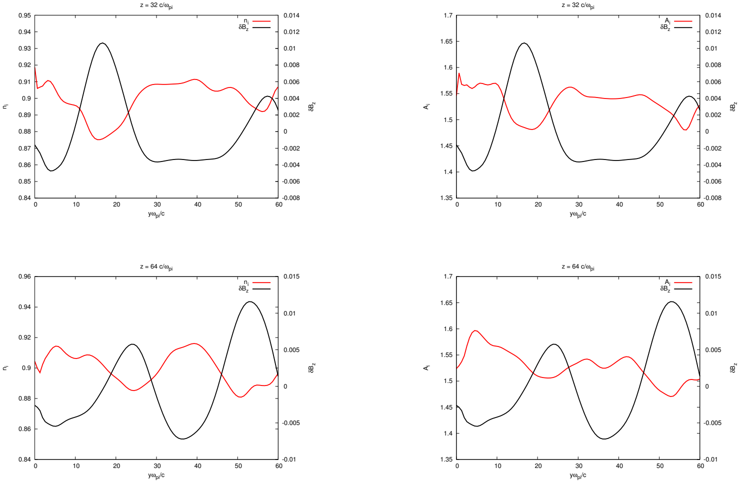
Bi-Maxwellian ions and helium are uniformly distributed in the simulation space with $T_{\perp i}/T_{\parallel i} = 1.5$. A constant background magnetic field $B_0 = 0.1$ is assumed in the z direction. Other parameters are: $\beta_i = 4.$, $\beta_e = 2.$, $\beta_h = 4.$, $T_{\perp e}/T_{\parallel e} = 1$, $T_{\perp h}/T_{\parallel h} = 1.5$, $m_h/m_i = 4.$, $m_i/m_e = 25$, $n_h = 0.1$



2-d PIC simulation

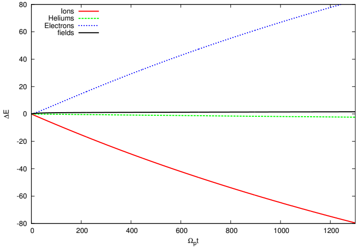


Anti-correlation of n_i and δB_z

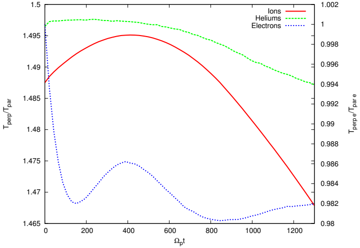


Energy Balance, Evolution of Anisotropy

Energy Balance



Anisotropies



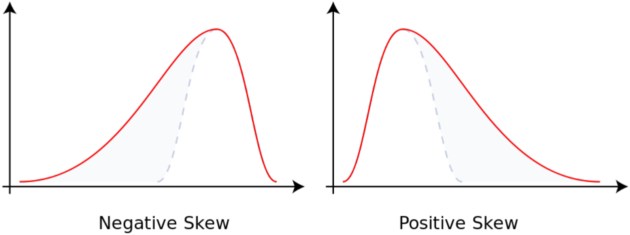
$T_{\perp i} = 0.03,\ T_{\parallel i} = 0.02,\ T_{\perp h} = 0.03,\ T_{\parallel h} = 0.02,\ T_{\perp e} = T_{\parallel e} = 0.01,$
$m_i/m_e = 25$

Skewness

The skewness is a statistical value to measure an asymmetry of a distribution of samples. We use the skewness (*Soucek et al., 2008*) to identify the magnetic structures as magnetic peaks (positive) or dips (negative).

For a sample of *n* values the sample skewness is

$$S = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^3}{(\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2)^{3/2}}$$

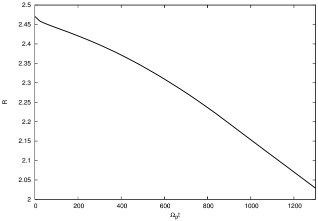


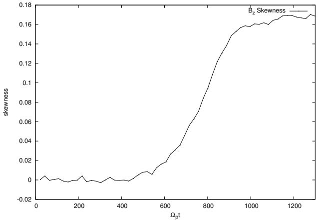
where $\bar{x}$ is sample mean.

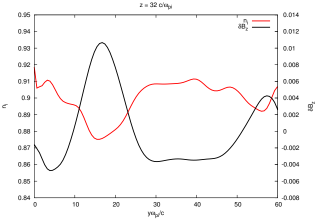
Figure: http://en.wikipedia.org/wiki/Skewness

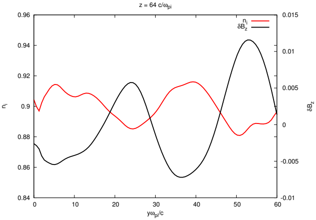
Peaks or Dips

In mirror unstable region ($R = \beta_\perp (T_\perp / T_\parallel - 1) > 0$), we see peaks.









Summary / Outlook

- GPU can accelerate PIC calculations substantially ($> 4\times$ on Titan)
- Particle shape order is very important for numerical heating, more work is needed to support 2nd order Esirkepov charge deposition on GPUs.
- Mirror instability has been benchmarked with nonlinear theory, nonlinear results show peaks as expected, but there are still many open questions.
  - Can we find dips in mirror stable regime? ($R < 0$)
  - What is the energy exchange process between ion, electron thermal energy?
  - What role does the simulation mass ratio play?
  - Analyze the evolution of electron anisotropy.