Argonne
NATIONAL LABORATORY

... for a brighter future

Developing Peta-scalable Algorithms for Beam Dynamic Simulations

Jin XU
Physics Division/
Math. & Computer Science Division

Argonne National Laboratory
9700 S. Cass Ave.
Argonne, IL 60439.

1st International Particle Accelerator Conference
Kyoto, Japan. May 25, 2010

U.S. Department of Energy

UChicago ▸ Argonne LLC

Office of Science
U.S. DEPARTMENT OF ENERGY

A U.S. Department of Energy laboratory managed by UChicago Argonne, LLC

Collaborators

- Peter Ostroumov (PHY, ANL)
- Brahim Mustapha (PHY, ANL)
- Jerry Nolen (PHY, ANL)
- Ivan Gonin (FNAL)
- Jean-Paul Carneiro (FNAL)
- Henry A. Belch Jr.(NE,ANL)
- Joel Furst (PHY, ANL)
- Paul Fischer (MCS, ANL)
- Misun Min (MCS, ANL)
- John Deford (AWR Corp., USA)
- Ben Held (AWR Corp., USA)
- Geoff Waldschmidt (APS, ANL)
- James J. Grudzinski (NE, ANL)
- Philip S. Strons (NE, ANL)

Contents

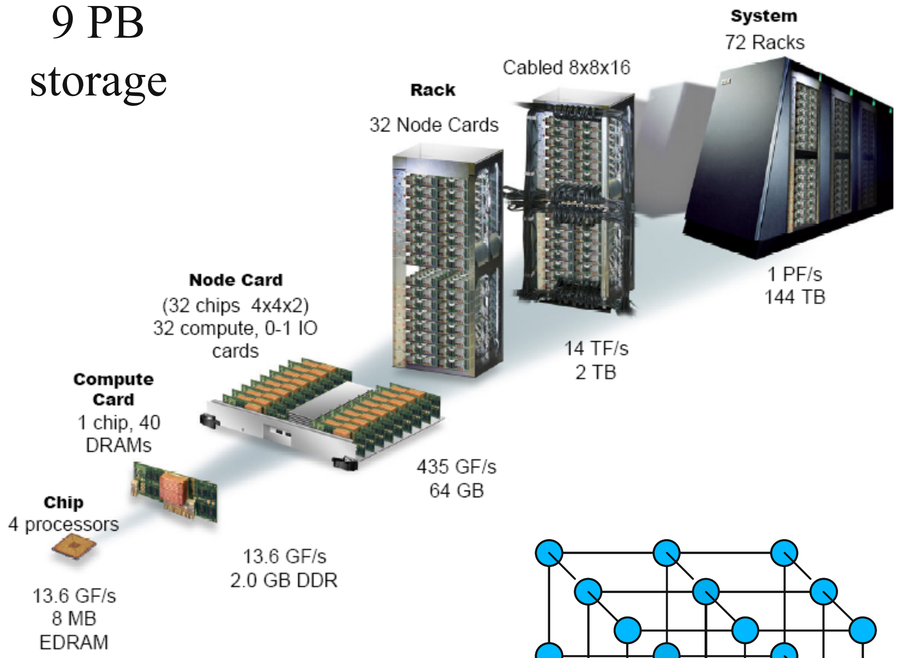
ANL Leadership Computing Facility (ALCF)

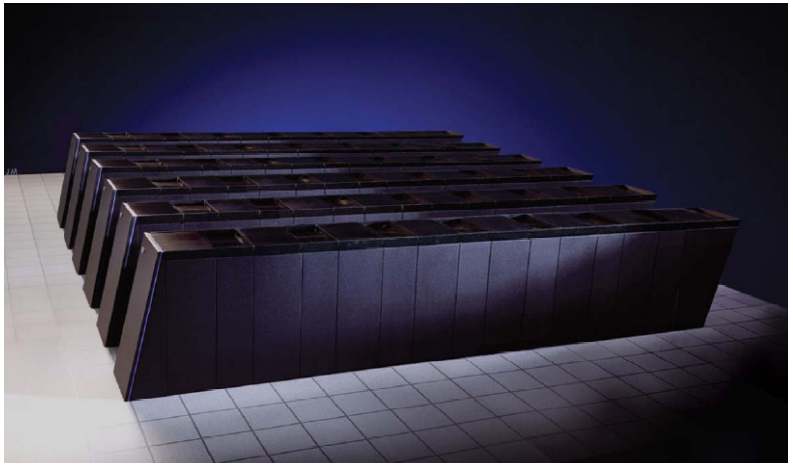
Current
557 TF/s
80 TB

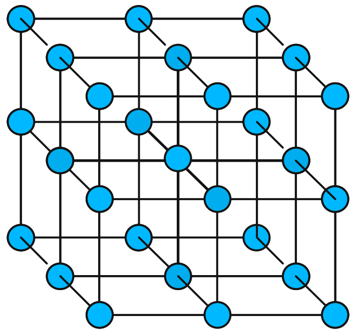
- **Intrepid (IBM Blue Gene/P)**
- 40 racks, 40,960 quad-core nodes (163,840 processors)
- 80 terabytes of memory.
- Peak performance is 556 teraflops.
- 512Mbs/processor

4096
CPU/rack

9 PB
storage







3D torus network
10 GB/s between any node

TOP500 list (11/2009)

No	Organization	Country	Cores	Speed (Tflops)
1	Oak Ridge National Laboratory	United States	224162	1759
2	DOE/NNSA/LANL	United States	122400	1042
3	National Institute for Computational Sciences/University of Tennessee	United States	98928	831.7
4	Forschungszentrum Juelich (FZJ)	Germany	294912	825.5
5	National SuperComputer Center in Tianjin/NUDT	China	71680	563.1
6	NASA/Ames Research Center/NAS	United States	56320	544.3
7	DOE/NNSA/LLNL	United States	212992	478.2
8	Argonne National Laboratory	United States	163840	458.611

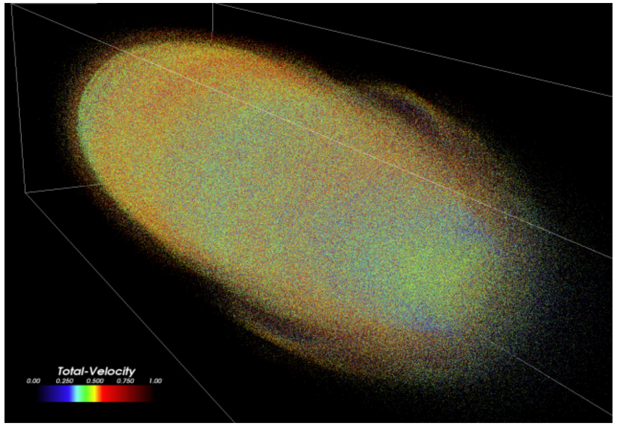
➢ Status

- Commercial software: Particle Studio, PARMILA, etc
- Academic software: IMPACT, ELEGANT, COSY,

TRACE3D, TRANSPORT, VORPAL, TRACK, ……

- Some can be run on tera-flops supercomputers
- Peta-scale computing has many challenges

100,000,000



➢ Challenges

- There are many challenges of BDS: scalability, visualization, I/O, etc.
- 10^9 particles can be simulated one-to-one and end-to-end in accelerators now.
- High intensive beam may contain $10^{12\sim18}$ particles.
- I/O challenges: 10^9 particles generate 180GB data.
- Transferring 180GB to hard drive takes half hours with 100MB/s network bandwidth!
- 10^{12} particles will generate 185 TB data! (SC08)

Status and Challenges

Numerical Methods and Peta-scalable Algorithms

- ✓ Numerical methods for plasma and charged beam
- ✓ Numerical methods for Poisson's equation
  - Finite Difference, Finite Element, Fourier Spectral Method, Fourier hp-Finite Element, hp-Finite Element, Multi-grid, Wavelet Method, etc
  - Scalability for peta-scale computing
- ✓ Particle-In-Cell method (PIC)
- ✓ Direct methods for solving the Vlasov equation (DVS)
  - Vlasov equations in 2D and 4D phase spaces
  - Structured vs. unstructured mesh
  - Semi-Lagrangian method
  - Discontinuous Galerkin method
- ✓ Domain decomposition in high dimension
- ✓ Comparison of PIC and DVS solvers

Beam Dynamics Simulation ∈ Plasma Simulation

Fluid model | Kinetic model | Microscopic

- o Numerical models for plasma simulation
  - Microscopic model (N-body problem in 6N dimension)
  - Kinetic model (Boltzmann or Vlasov + Maxwell Eqn. in 6D) → PIC (3D), DNS
  - Fluid model (MHD or two-fluid model in 3D)
- o Particle-In-Cell (PIC) method is a simplification in kinetic model to solve Vlasov Eqn. PDE→ODE. (Birdsall & Langdon, 1985)
- o Advantages: Easy to implement, fast speed, small grid
- o Disadvantages: Noise decreases as $1/\sqrt{N}$, hard to predict detailed beam structures, such as beam halo.

$$\frac{\partial f}{\partial t} + \frac{\vec{\boldsymbol{p}}}{\gamma m} \cdot \nabla_{\vec{x}} f + \vec{\boldsymbol{F}} \cdot \nabla_{\vec{p}} f = \left.\frac{\partial f}{\partial t}\right|_{collision}$$

$$\vec{\boldsymbol{F}} = q(\vec{\boldsymbol{E}} + \vec{v} \times \vec{\boldsymbol{B}})$$

PIC →

$$\frac{d\vec{\boldsymbol{x}}}{dt} = \frac{\vec{\boldsymbol{p}}}{\gamma m} = \vec{\boldsymbol{v}}$$

$$\frac{d\vec{\boldsymbol{v}}}{dt} = \vec{\boldsymbol{F}} = q \cdot (\vec{\boldsymbol{E}} + \vec{\boldsymbol{v}} \times \vec{\boldsymbol{B}})$$

Parallel Algorithm for the PIC code-PTRACK

Beam was accelerated through electromagnetic forces in LINAC.

Parallel Efficiency = Particle Tracking, 100% + Electromagnetic Fields, 0~100% (external EM and space charge)

$$\frac{d\vec{\boldsymbol{x}}}{dt} = \frac{\vec{\boldsymbol{p}}}{\gamma m} = \vec{\boldsymbol{v}}$$

$$\frac{d\vec{\boldsymbol{v}}}{dt} = \vec{\boldsymbol{F}} = q \cdot \vec{\boldsymbol{E}} = -\boldsymbol{q} \cdot \nabla \phi$$

Suppose external fields are known or develop our own EM solvers

Electromagnetic field simulation:

$$\nabla \cdot \vec{E} = \frac{\rho}{\varepsilon_0}$$

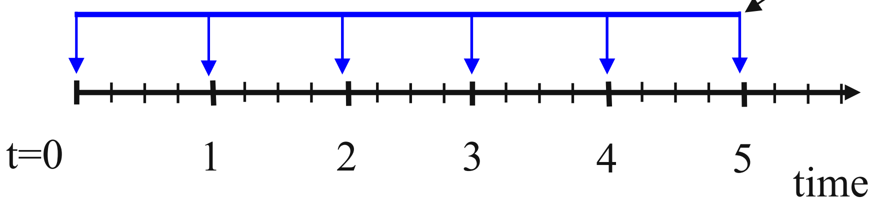
$$\nabla \times \vec{E} = -\frac{\partial \vec{B}}{\partial t}$$

$$\nabla \cdot \vec{B} = 0$$

$$\nabla \times \vec{B} = \mu_0 \vec{J} + \varepsilon_0 \mu_0 \frac{\partial \vec{E}}{\partial t}$$

Space charge kick in velocity space

$$\nabla^2 \phi = -\frac{\rho}{\varepsilon_0}$$



4th order Runge-Kutta Scheme

PTRACK: Z-Based Code and Current Capability

$$\frac{dx}{dz} = x'$$

$$\frac{dy}{dz} = y'$$

$$\frac{d\phi}{dz} = \frac{2\pi f_0 h}{\beta c}$$

$$\begin{pmatrix} x \\ x' \\ y \\ y' \\ \phi \\ \beta \end{pmatrix}$$

- Any type of RF resonator (3D fields)
- Static ion-optics devices (3D fields)
- Radio-Frequency Quadrupoles
- Solenoids with fringe fields
- Bending magnets with fringe fields
- Electrostatic and magnetic multipoles
- Multi-harmonic bunchers
- Axial-symmetric electrostatic lenses
- Entrance and exit of HV decks
- Accelerating tubes with DC voltage
- Stripping foils or film (for FRIB)
- Horizontal and vertical jaw slits

$$\frac{dx'}{dz} = \chi \frac{Q}{A} \frac{h}{\beta\gamma} \left[\frac{h}{\beta c}(E_x - x' E_z) + x' y' B_x - (1 + x'^2) B_y + y' B_z \right]$$

$$\frac{dy'}{dz} = \chi \frac{Q}{A} \frac{h}{\beta\gamma} \left[\frac{h}{\beta c}(E_y - y' E_z) + x' y' B_y + (1 + y'^2) B_x - x' B_z \right]$$

$$\frac{d\beta\gamma}{dz} = \chi \frac{Q}{A} \frac{h}{\beta c}(x' E_x + y' E_y + E_z)$$

Transverse planes:
(x, x'=dx/dz) horizontal
(y, y'=dy/dz) vertical

Longitudinal plane: $(\Delta\phi = 2\pi f \Delta t, \Delta W)$

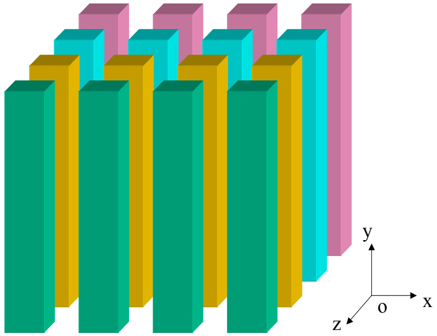
Relativistic Effect

Phase deviation Energy deviation

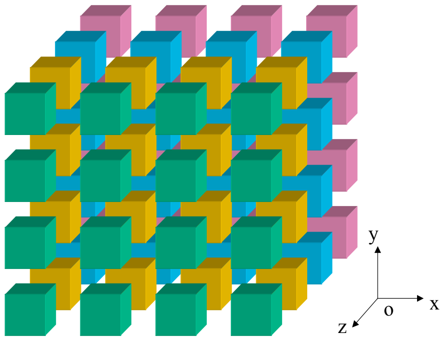
Parallel Models for Poisson's Equation

1D Model

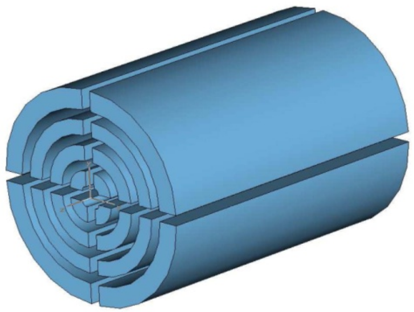
2D Model

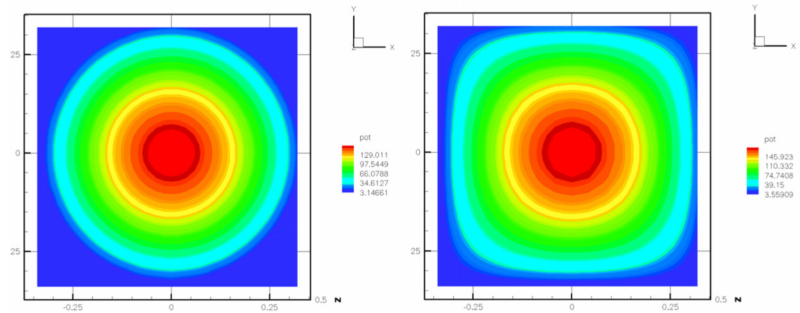


3D Model



Cylindrical

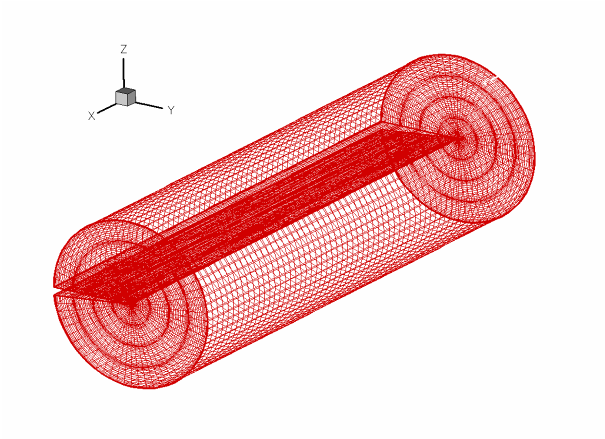




Space Charge Effect:

$$\Delta \phi = \frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} + \frac{\partial^2 \phi}{\partial z^2} = -\frac{\rho}{\varepsilon_0}$$

$$= \frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial \phi}{\partial r}\right) + \frac{1}{r^2}\frac{\partial^2 \phi}{\partial \theta^2} + \frac{\partial^2 \phi}{\partial z^2}$$



✓ **J. Xu, B. Mustapha, V. Aseev and P. Ostroumov, Phys. Rev. ST Accel. Beams 10, 014201 (2007)**

Multi-resolution Analysis

$$V_j = \{\phi_{j,k}, k \in Z\} \qquad \phi_{j,k}(x) = 2^{j/2}\phi(2^j x - k)$$

$$\cdots \subset V_{-2} \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \subset \cdots \subset L^2(R)$$

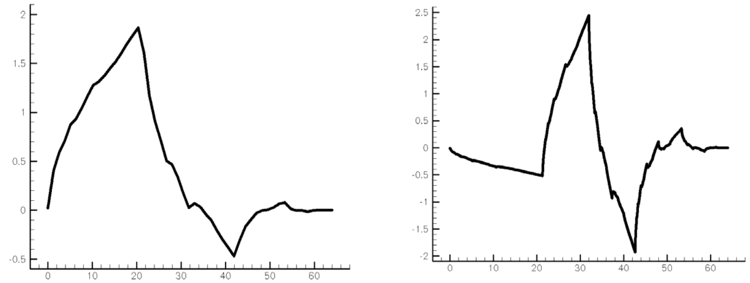
Since $V_j \supset V_{j+1}$ We can define $W_j \perp V_j$

and $W_j \supset V_{j+1}$ Such that $V_j \oplus W_j = V_{j+1}$

$$W_j = \{\psi_{j,k}, k \in Z\} \qquad \psi_{j,k}(x) = 2^{j/2}\psi(2^j x - k)$$

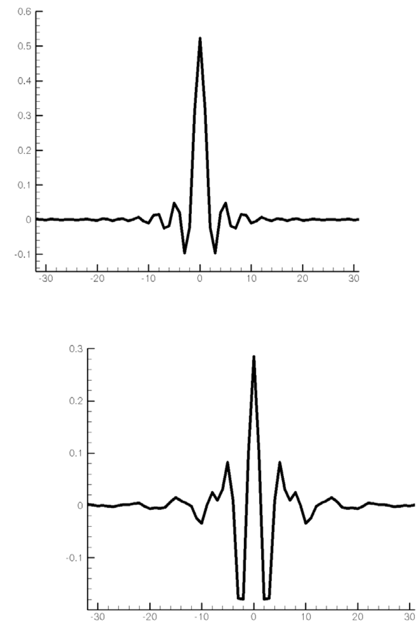
$$V_2 = W_1 \oplus V_1 = W_1 \oplus W_0 \oplus V_0 = W_1 \oplus W_0 \oplus W_{-1} \oplus V_{-1}$$

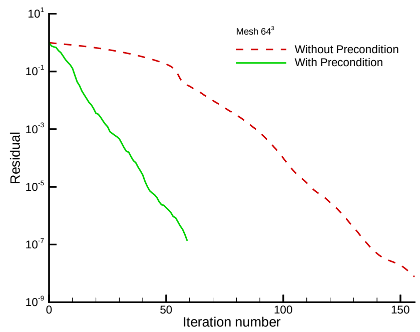
Wavelet Method



Daubechies D4

Meyer





$$\phi(x,y,z) = (x+\pi)(x-3\pi)(y+\pi)(y-3\pi)(z+\pi)(z-3\pi)$$

$$\Delta\phi(x,y,z) = -\rho(x,y,z)$$

$$\rho(x,y,z) = 2(x+\pi)(x-3\pi)(y+\pi)(y-3\pi) + 2(x+\pi)(x-3\pi)(z+\pi)(z-3\pi) + 2(y+\pi)(y-3\pi)(z+\pi)(z-3\pi)$$

Iterative Method: Preconditioning Conjugate Gradient (PCG)

PTRACK Benchmarks

150 GBs

Particle#	1M	10M	100M	865M
CPU#	1024= 16*16*4	4096= 16*16*16	4096= 16*16*16	32768= 32*32*32
Particle/CPU	1k	2.56k	25.6k	26.4k
SC Grid	$32 \times 32 \times 32$	$32 \times 32 \times 32$	$32 \times 32 \times 32$	$32 \times 32 \times 32$
I/O time(s)	14	110	36	135
Tot Time(h)	0.63	1.06	6.3	4.8

Weak scaling

CPU	Time/cell (s)	Particle #	Parallel Efficiency
256	384	55M	100%
512	384	110M	100%
1024	388.7	220M	98.8%
2048	400.6	440M	95.8%
4096	385	880M	99%

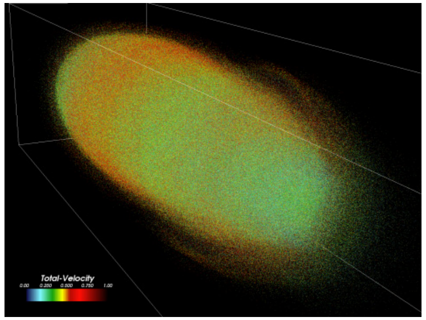
Strong scaling for 110 M particles

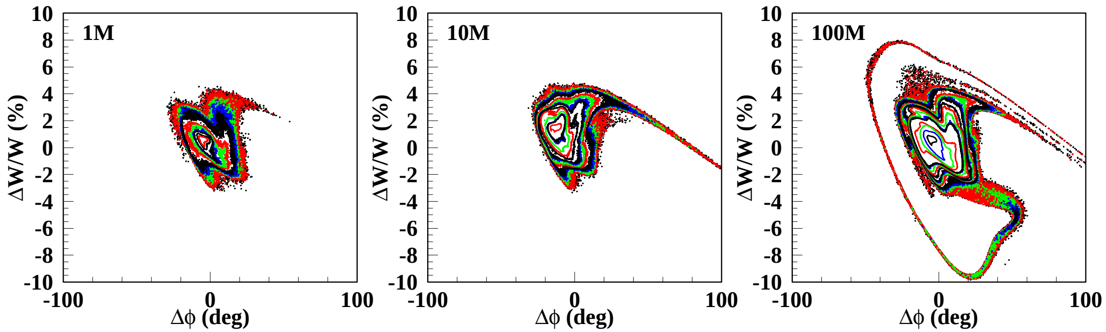
CPU	Time/cell (s)	Ideal Time (s)	Parallel Efficiency
512	384	384	100%
1024	225	192	85.3%
2048	107	96	89.7%
4096	63	48	76.1%

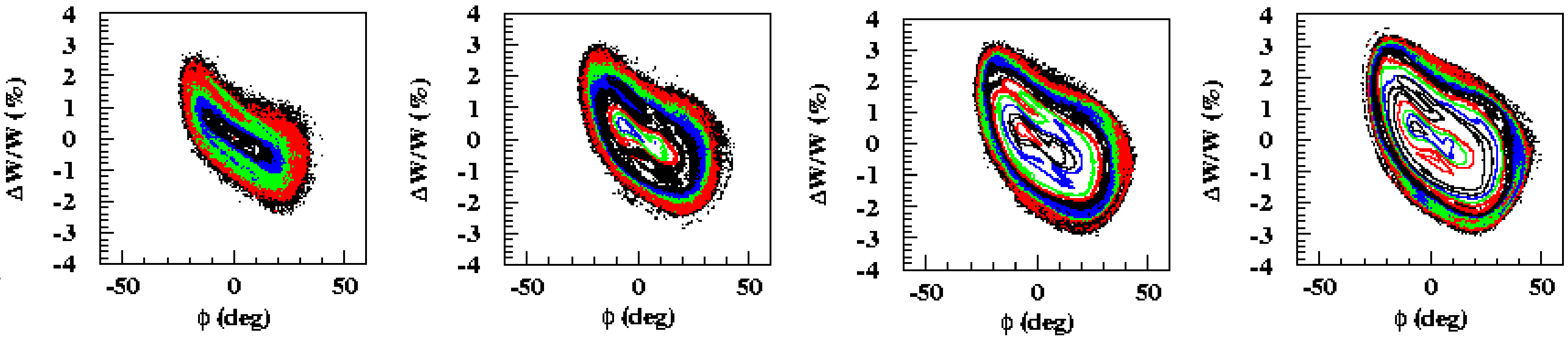
One to One Beam Dynamics Simulations

- Noise due to finite number of macro particles in PIC method.
- Suppose using 1M macro particles, each macro particle corresponding to 865 real particles. It is hard to simulate beam halo where particle number is less than 865.
- Simulated the actual number of particles in a 45 mA proton beam at 325 MHz accelerated in a RFQ from 50 keV to 2.5 MeV → 865 M particles on 32768 processors.
- Some beams have 10^{12} particles!
- PTRACK is ready, and has been used in large scale optimizations.

100M Particles



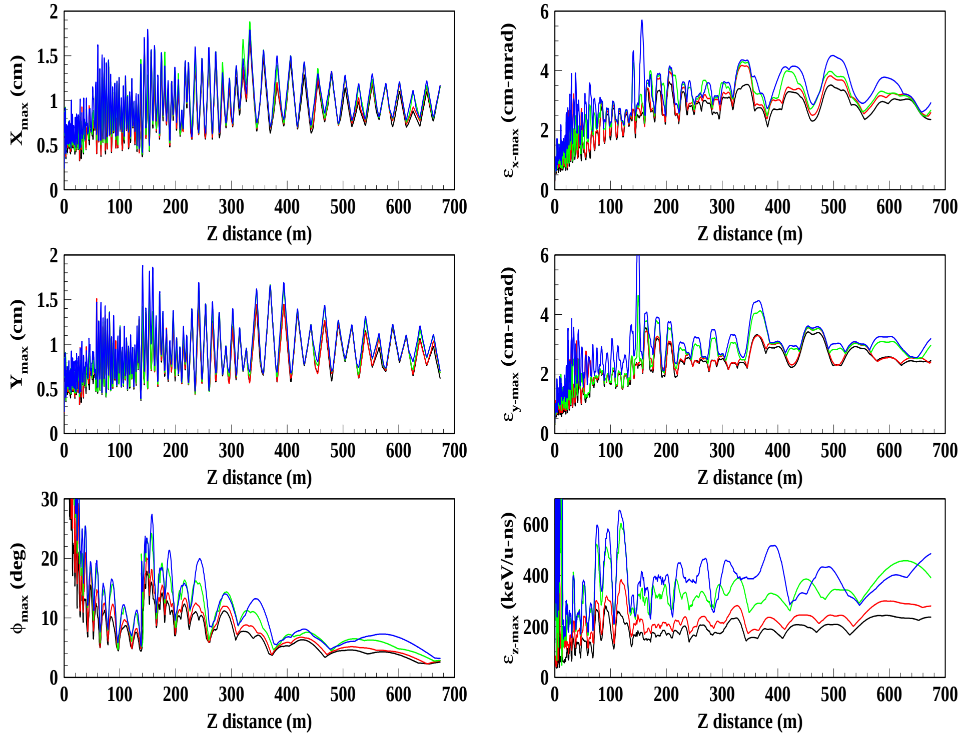




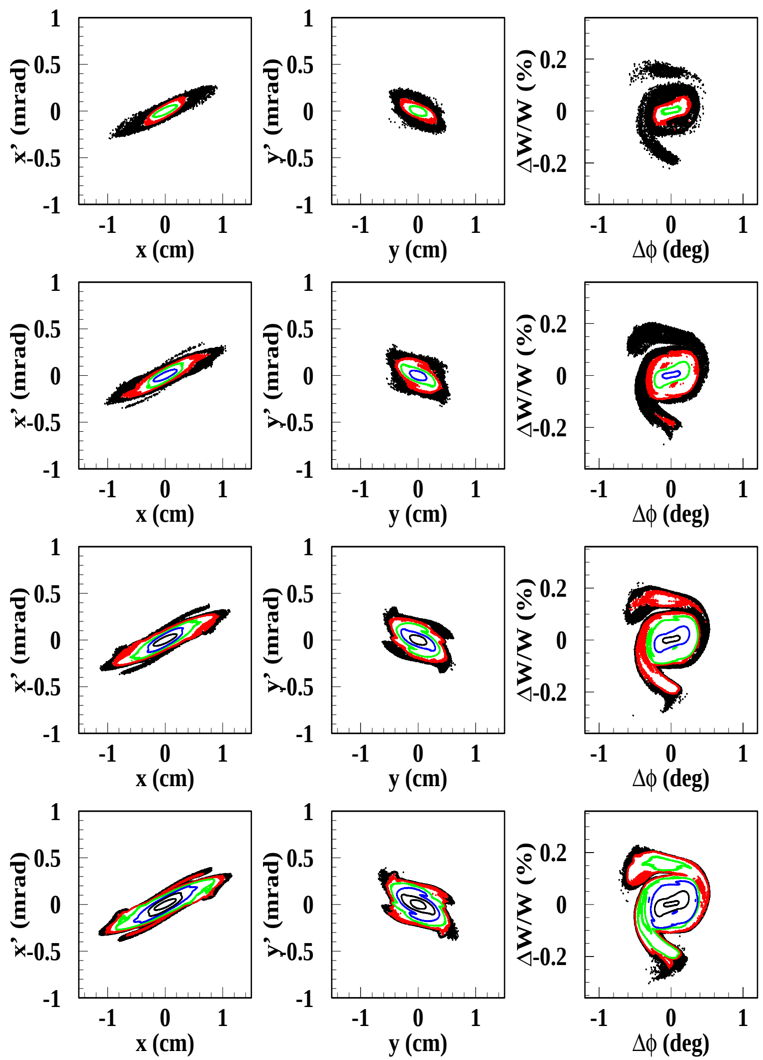
RFQ 1M 10M 100M 865M

Large Scale Optimization

Beam Dynamics with Increasing Statistics



Beam envelopes and total emittances along the HINS linac with increasing number of simulated particles: 1M, 10M, 100M, 865M



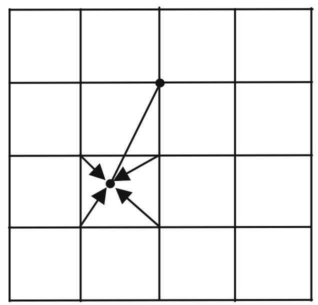
(1) 1M (2)10M (3)100M (4)865M

Comparison of DVS and PIC Methods

- **Advantages**:
  - Avoids random fluctuations caused by finite number of macro-particles
  - Can resolve fine structures in low density regions of phase space
  - Easy to simulate pure modes and detect instability

- **Disadvantages**:
  - ✓ Curse of dimensionality: N^{2d} grid points needed in d dimensions.

- Follow work from E. Sonnendrucker, F. Filbet, M. Gutnic, etc.
- High-order method + large scale computing
- Semi-Lagrangian method
- Discontinuous Galerkin method



Liouville Theorem:
Phase-space distribution function is constant on the trajectory of the system

Literature

References:

- E. Sonnendrucker, J. Roche, P. Bertrand and A. Ghizzo, The semi-Lagrangian method for the numerical resolution of Vlasov equations, J. Comput. Phys., 149, 201 (1998).
- F. Filbet, E. Sonnendr¨ucker, Modeling and Numerical Simulation of Space Charge Dominated Beams in the Paraxial Approximation Mathematical Models and Methods in Applied Sciences, Vol.16, No.5, 763-791 (2006).

- F. Filbet, E. Sonnendr¨ucker, Modeling and Numerical Simulation of Space Charge Dominated Beams in the Paraxial Approximation Mathematical Models and Methods in Applied Sciences, Vol.16, No.5, 763-791 (2006).

1D1V Vlasov Equation and Numerical Techniques

Consider ions and electrons with opposite charges of equal magnitude.

Mass ratio $M_r = m_i / m_e >> 1$

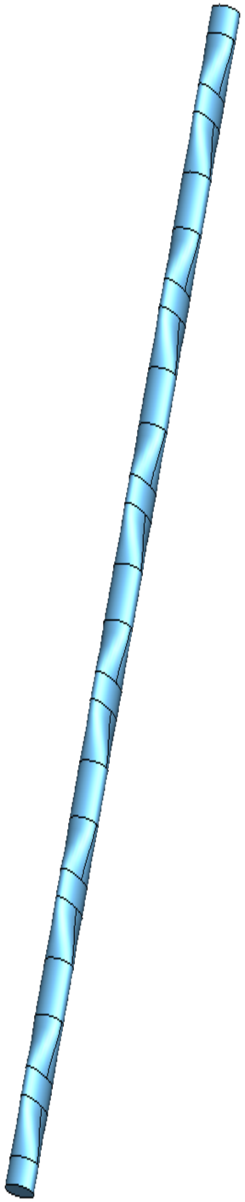
$$\frac{\partial f_e}{\partial t} + v\frac{\partial f_e}{\partial x} - E\frac{\partial f_e}{\partial v} = 0$$

$$\frac{\partial f_i}{\partial t} + v\frac{\partial f_i}{\partial x} + \frac{1}{M_r}E\frac{\partial f_i}{\partial v} = 0$$

$$\frac{\partial E}{\partial x} = \frac{q_e^2}{m_e \varepsilon_0}\int (f_i - f_e)dv = \int (f_i - f_e)dv$$

$$\frac{\partial f}{\partial t} + v\frac{\partial f}{\partial x} - E\frac{\partial f}{\partial v} = 0$$

$$\frac{\partial E}{\partial x} = 1 - \int f dv = -\frac{\partial^2 \phi}{\partial x^2}$$



Vlasov equation in 2D2V space: Paraxial Approximation

- Steady state beam: All partial time derivatives vanish.
- Beam propagating at constant velocity v_b along z axis.
- Paraxial model $$\frac{\partial f}{\partial z}+\frac{\vec{\boldsymbol{v}}}{v_b}\cdot\nabla_{\vec{x}}f+\frac{q}{\gamma_b m v_b}\left(-\frac{1}{\gamma_b^2}\nabla\phi^s+\vec{\boldsymbol{E}}^e+(\vec{\boldsymbol{v}},v_b)^T\times\vec{\mathrm{B}}^e\right)\cdot\nabla_{\vec{v}}f=0$$
- Self electric potential is solved from Poisson equation $$-\Delta_{\vec{x}}\phi^s=\frac{q}{\varepsilon_0}\int_{\Re^2}f(z,\vec{\boldsymbol{x}},\vec{\boldsymbol{v}})d\vec{\boldsymbol{v}}$$
- Self magnetic potentials:

$$\nabla\times\vec{E}=0,\quad \nabla\cdot\vec{E}=\frac{\rho}{\varepsilon_0},$$

$$\nabla\times\vec{B}=\mu_0\vec{J},\quad \nabla\cdot\vec{B}=0.$$

$$\vec{E}=-\nabla\phi^s,\quad \vec{B}=-\nabla^{\perp}\psi^s=-\left(\frac{\partial\psi^s}{\partial x},\frac{\partial\psi^s}{\partial y}\right),$$

$$-\Delta\phi^s=\frac{qn}{\varepsilon_0},\quad -\Delta\psi^s=\mu_0 v_b qn=\frac{qnv_b}{\varepsilon_0 c^2}.$$

$$\phi^s=\frac{v_b}{c^2}\psi^s$$

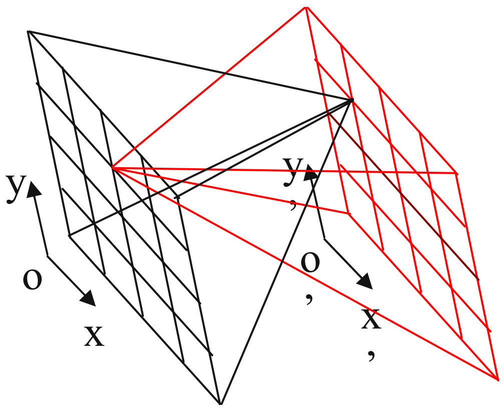
- External fields for $\vec{\boldsymbol{E}}$ and $\vec{\boldsymbol{B}}$
  - Uniform focusing Electric field:

$$\vec{\boldsymbol{E}}(\vec{\boldsymbol{x}})=-\frac{\gamma_b m}{q}\omega_0^2(x\vec{\boldsymbol{e}}_x+y\vec{\boldsymbol{e}}_y)$$

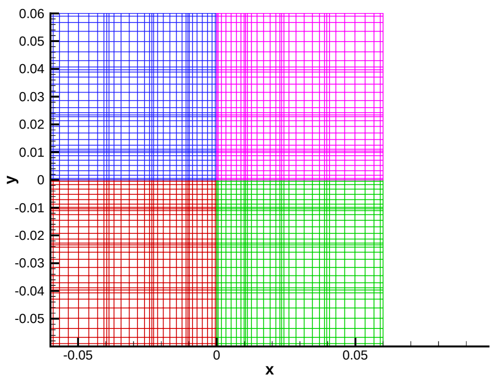
  - Periodic focusing magnetic field:

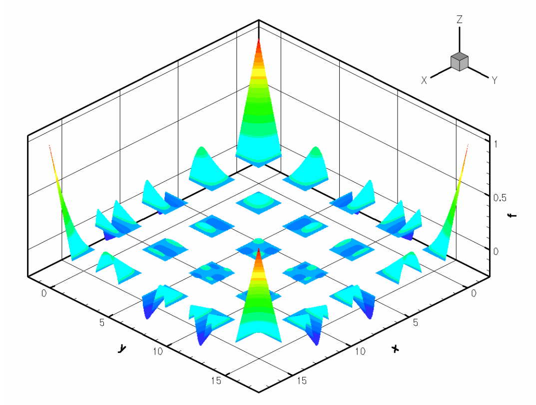
$$\vec{\boldsymbol{B}}(\vec{\boldsymbol{x}})=B(z)\vec{\boldsymbol{e}}_z-\frac{1}{2}B'(z)(x\vec{\boldsymbol{e}}_x+y\vec{\boldsymbol{e}}_y)$$

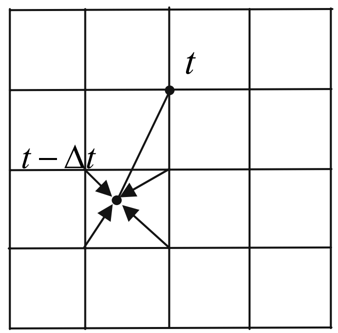
Semi-Lagrange Method to Solve Vlasov Equation



- ✓ Density function f conserved on the characteristics
- ✓ Find the origin of the characteristics ending at grid points
- ✓ Interpolate the density function at the origin of the characteristics from known grid points
- ✓ High order interpolation method is necessary
- ✓ Fast interpolation is also necessary







$$u(x,y,t)=\sum_{l}\sum_{k}u(l,k,t)P_{l.k}(x,y)$$

$$P_{l,k}(x,y)=P_l(x)*P_k(y)$$

$$P_l(x)=P_l(x)=\begin{cases}(\frac{1-x}{2}) & l=0\\ (\frac{1-x}{2})(\frac{1+x}{2})P_{l-1}^{1,1}(x) & 0<l<P_1\\ (\frac{1+x}{2}) & l=P_1\end{cases}$$

➢ Time splitting scheme (1976 Cheng and Knorr)

$$\frac{\partial f}{\partial t}+\frac{1}{2}\cdot\frac{\vec{\boldsymbol{p}}}{\gamma m}\cdot\nabla_{\vec{x}}f=0$$

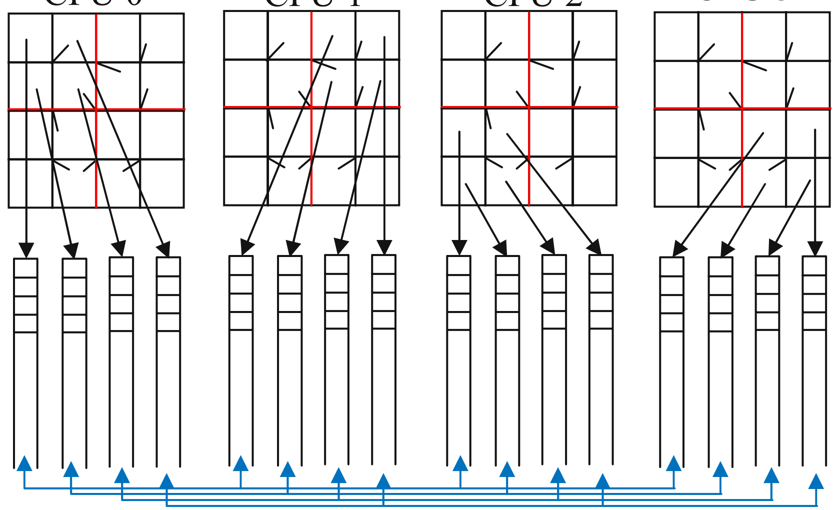
$$\frac{\partial f}{\partial t}+q(\vec{\boldsymbol{E}}+\vec{v}\times\vec{\boldsymbol{B}})\cdot\nabla_{\vec{p}}f=0$$

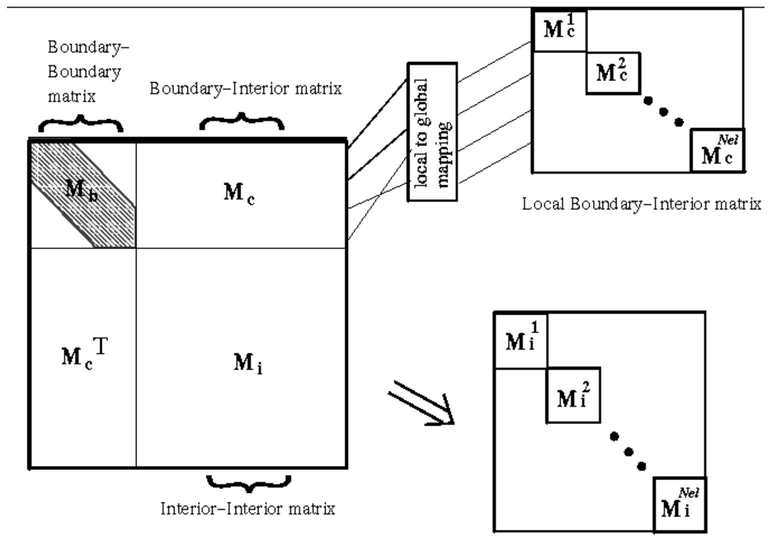
$$\frac{\partial f}{\partial t}+\frac{1}{2}\cdot\frac{\vec{\boldsymbol{p}}}{\gamma m}\cdot\nabla_{\vec{x}}f=0$$

Parallel Poisson and Vlasov Solver

- Continuous Galerkin method
- Iterative vs. Direct methods
- Domain decomposition for parallelization
- Iterative: Conjugate Gradient Method
- Direct: Fast speed, but need large memory if solve globally.

Load Balancing





$$\begin{pmatrix} A_{bb} & C_{bi} \\ C_{bi}^T & A_{ii} \end{pmatrix}\begin{pmatrix} u_b \\ u_i \end{pmatrix} = \begin{pmatrix} f_b \\ f_i \end{pmatrix}$$

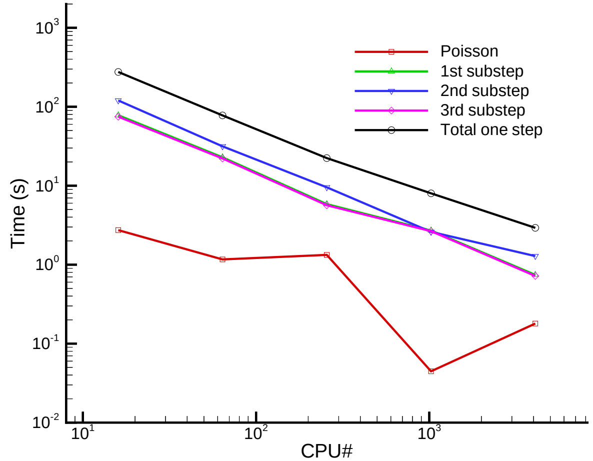
$$(A_{bb} - C_{bi}A_{ii}^{-1}C_{bi}^T)u_b = f_b - C_{bi}A_{ii}^{-1}f_i$$

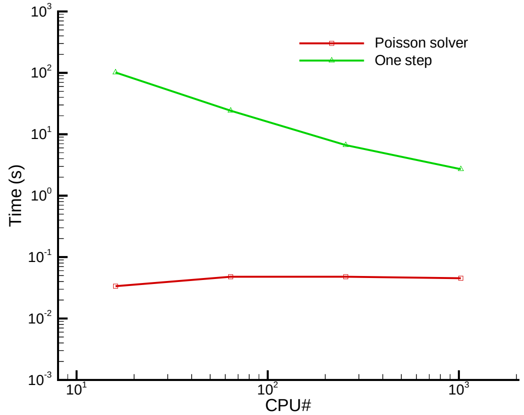
$$u_i = A_{ii}^{-1}(f_i - C_{bi}^T u_b)$$

Shur Complement

Benchmark on Poisson & Vlasov Solver

Total CPU	16	64	256	1024
CPU_x	2	4	4	8
CPU_y	2	4	4	8
CPU_{vx}	2	2	4	4
CPU_{vy}	2	2	4	4
Poisson solver	2.74	1.16	1.33	4.46e-02
1st substep	78.1	22.9	5.84	2.70
2nd substep	120.2	31.5	9.54	2.6
3rd substep	74.8	22.1	5.66	2.66
Total time	275.9	77.7	22.4	8.01
Speedup	1.0	3.55	12.32	34.4
Parallel Efficiency	1.0	0.89	0.77	0.54





(CPU_x, CPU_{vx})	Poisson time	Ratio	Total Time	Speedup	Parallel Efficiency
(16,1)	3.36e-02	0.033%	101.37	1.0	1.0
(16,4)	4.78e-02	0.20%	24.2	4.19	1.0
(16,16)	4.79e-02	0.72%	6.655	15.23	0.952
(16,64)	4.53e-02	1.68%	2.7	37.54	0.587

Benchmarks: Landau Damping

- Linear Landau Damping

$$f(0,x,v)=\frac{1}{\sqrt{2\pi}}\exp(-v^2/2)(1+\alpha\cos(kx))$$

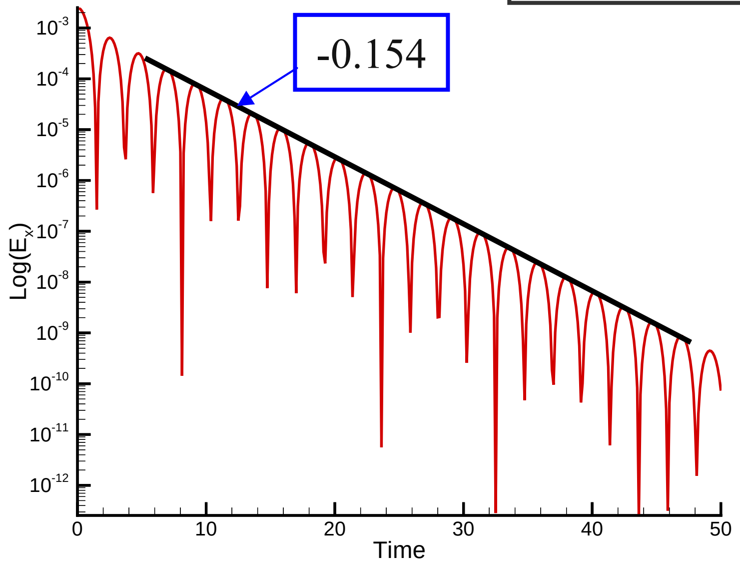
$$\forall(x,v)\in[0,L]\times R,\quad \alpha=0.01,\quad k=0.5$$

$$L=4\pi,\quad R=[-6,6],\quad \Delta t=1/8,$$

$$P=16,\quad E=64,\quad 1024\times1024$$

$$CPU=256,\quad T\sim 10\ \min s$$

Challenges



- Nonlinear (Strong) Landau Damping

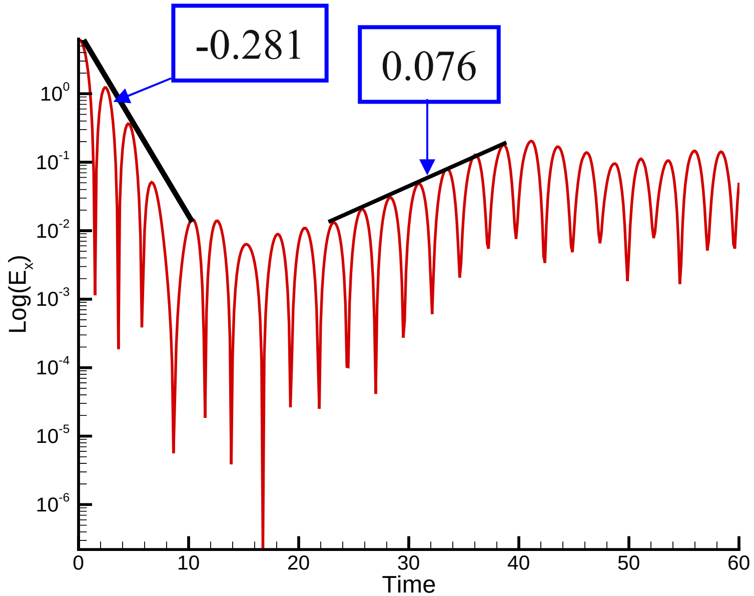
$$f(0,x,v)=\frac{1}{\sqrt{2\pi}}\exp(-v^2/2)(1+\alpha\cos(kx))$$

$$\forall(x,v)\in[0,L]\times R,\quad \alpha=0.5,\quad k=0.5$$

$$L=4\pi,\quad R=[-6,6],\quad \Delta t=1/8,$$

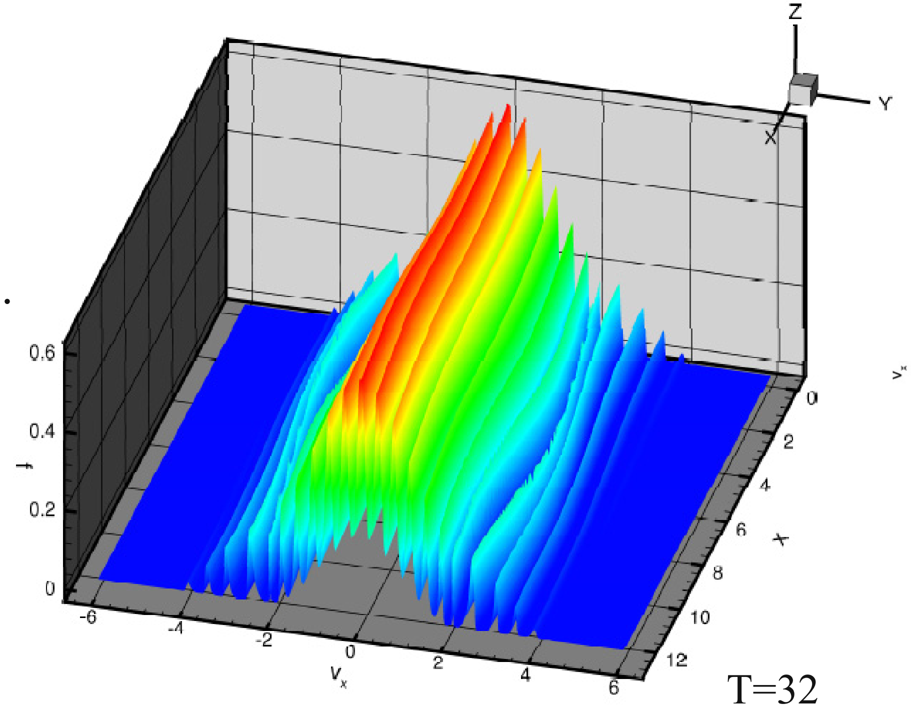
$$P=16,\quad E=64,\quad 1024\times1024$$

$$CPU=256,\quad T\sim 10\ \min s$$



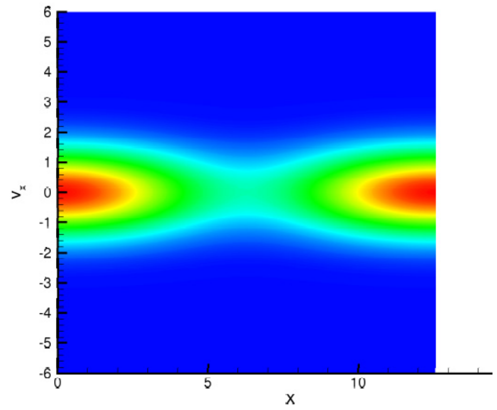
Simulations can help to understand physics, which is difficult in theoretical analysis.

Strong Landau Damping

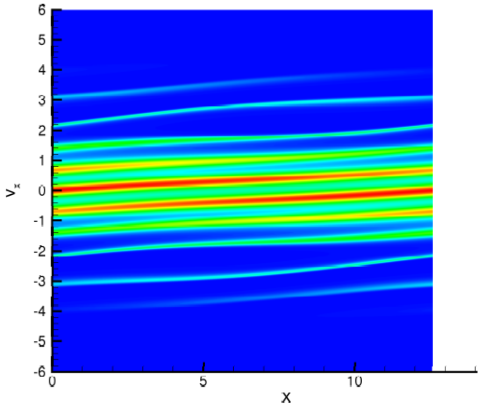


T=32

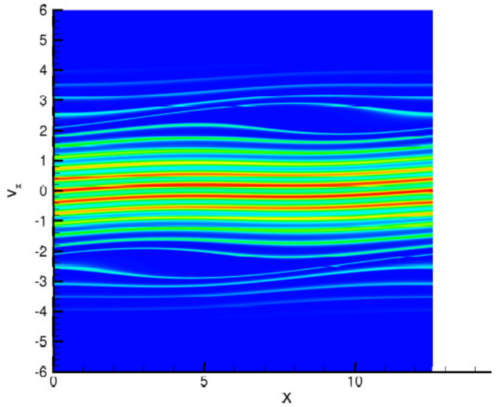
Filamentation



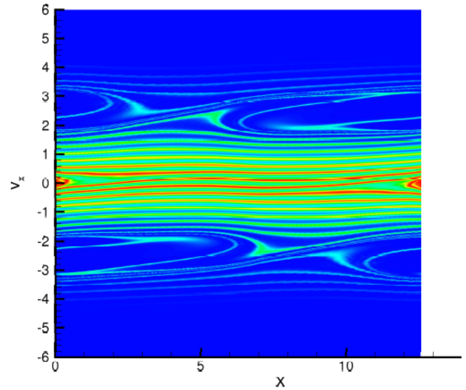
T=0



T=16



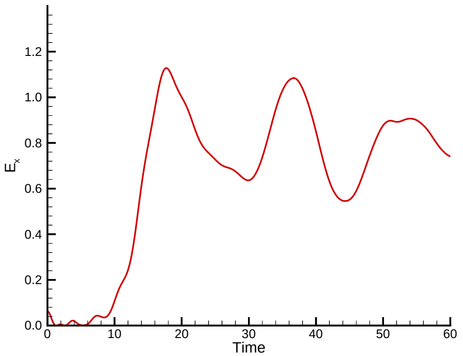
T=32

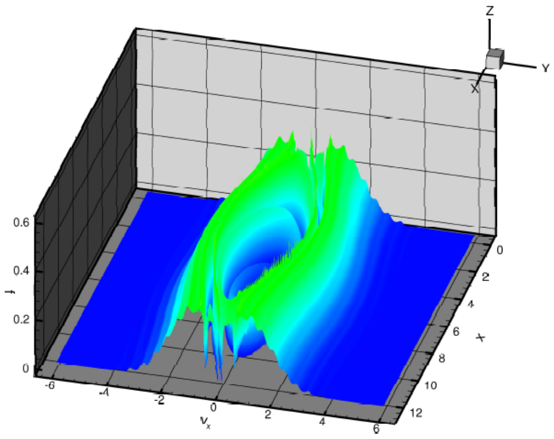


T=48

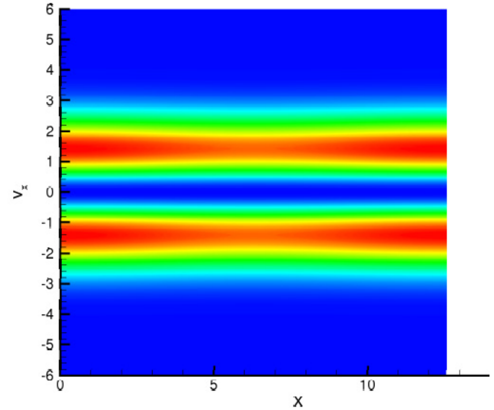
Two Stream Instability

$$f(0,x,v)=\frac{1}{\sqrt{2\pi}}v^2\exp(-v^2/2)(1+\alpha\cos(kx)),\quad \forall(x,v)\in[0,L]\times R.$$

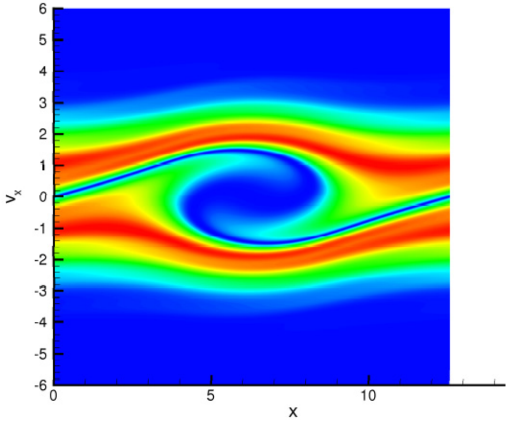




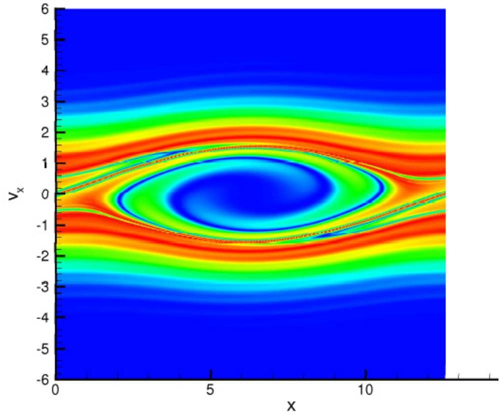
T=32



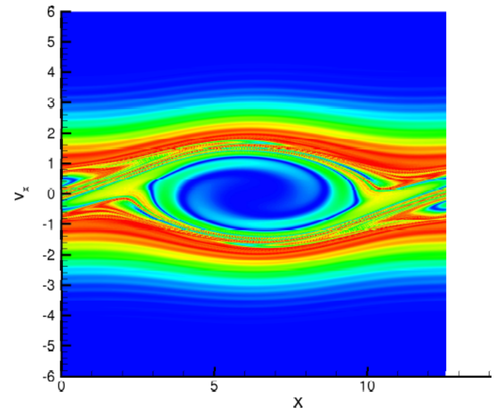
T=0



T=16



T=32



T=48

Other 1P1V Plasma Simulations

- Bump-on-Tail Instability

$$f(x,v) = f_{b.o.t}(v)(1+\alpha\cos(kx))$$

$$f_{b.o.t}(v) = \frac{n_p}{\sqrt{2\pi}}\exp(-\frac{1}{2}v^2) + \frac{n_b}{\sqrt{2\pi}}\exp(-\frac{1}{2}\frac{(v-v_b)^2}{v_t^2})$$

$$n_p = 0.9,\ n_b = 0.2,\ v_b = 4.5,\ v_t = 0.5,\ \alpha = 0.04,\ k = 0.3.$$

- Ion-Acoustic Turbulence

Ion distribution function: $f_i = \left(\frac{M_r}{2\pi}\right)^{1/2}\exp\left(-\frac{M_r}{2}v_i^2\right), \quad M_r = 1000.$

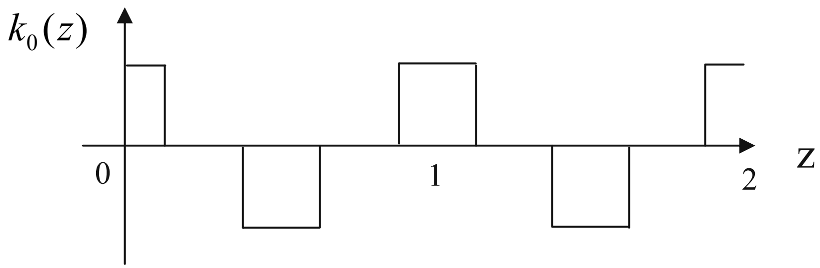
Electron distribution function: Drifting Maxwellian

$$f_e = (1+a(x))\left(\frac{1}{\sqrt{2\pi}}\right)\exp\left(-\frac{1}{2}(v_e - U_e)^2\right)$$

$$where, U_e = -2$$

$$a(x) = 0.01(\sin(x)+\sin(0.5x)+\sin(0.1x)+\sin(0.15x)+\sin(0.2x) + \cos(0.25x)+\cos(0.3x)+\cos(0.35x)).$$

Proton Beams



$> \text{Emittance} = 200\pi \text{ mm mrad}$

$> \text{Energy W} = 0.2 \text{ MeV}$

$> \text{Current I} = 0.1 \text{ A}$

$> \text{V}_\text{b} = 6.19 \times 10^6 \text{ m/s}$

$$> \vec{\text{E}}(\text{x},\text{y},\text{z}) = \begin{pmatrix} \text{k}_0(z)x \\ -k_0(z)y \end{pmatrix}$$

$$> k_0(z) = \begin{cases} V_b, & 0 < z < 1/8,\ 7/8 < z < 1 \\ 0, & 1/8 < z < 3/8,\ 5/8 < z < 7/8 \\ -V_b, & 3/8 < z < 5/8 \end{cases}$$

$$a = \frac{a_0}{2},\quad b = \frac{b_0}{2},\quad c = \frac{\varepsilon_x}{2a_0},\quad d = \frac{\varepsilon_y}{2b_0}$$

a_0, b_0 are from KV distribution

Gaussian Distribution

$$> \quad f_0(x,y,v_x,v_y) = \frac{n_0}{4\pi^2 abcd} \exp(-\frac{r^2}{2}),\quad r^2 = \left(\frac{x}{a}\right)^2 + \left(\frac{y}{b}\right)^2 + \left(\frac{v_x}{c}\right)^2 + \left(\frac{v_y}{d}\right)^2$$

Kapchinky-Vladimirsky Distribution

The KV distribution

$$f(z,x,y,v_x,v_y)=\frac{N_0}{\pi^2\varepsilon_x\varepsilon_y}\delta_0\left(\frac{x^2}{a^2}+\frac{y^2}{b^2}+\frac{(a\frac{v_x}{v_b}-a'x)^2}{\varepsilon_x^2}+\frac{(b\frac{v_y}{v_b}-b'y)^2}{\varepsilon_y^2}-1\right)$$

where a, b are the solutions of the so-called envelope equation

$$a''+\kappa_x(z)a-\frac{2K}{a+b}-\frac{\varepsilon_x^2}{a^3}=0,\quad b''+\kappa_y(z)b-\frac{2K}{a+b}-\frac{\varepsilon_y^2}{b^3}=0$$

Mathieu-Hill Differential Equation: Force field linear to (x, y)

$$X'=\frac{V_x}{v_b},\quad V_x'=-\bar{\kappa}_x(z)v_bX,\qquad \longrightarrow \qquad \frac{X^2}{a^2}+\frac{(a\frac{V_x}{v_b}-a'X)^2}{\varepsilon_x^2}=1$$

$$Y'=\frac{V_y}{v_b},\quad V_y'=-\bar{\kappa}_y(z)v_bY,\qquad where\ a=\sqrt{\varepsilon_x}w_x,\quad w_x''(z)+\bar{\kappa}_x(z)w_x(z)-\frac{1}{w_x^3(z)}=0$$

Use Trace2D to solve these envelope equations

Equivalent Beams

- Two beams of identical particles defined by their distribution function f are equivalent if they have same energy, same total number of particles and same 2nd order moments. (Sacherer and Lapostolle)

$$f(x,y,x',y') = N_0 f_0(\frac{x}{a},\frac{y}{b},\frac{x'}{c},\frac{y'}{d}) \qquad \chi_{rms}(f) = \sqrt{\frac{\int \chi(x,y,x',y')^2 f(x,y,x',y')dxdydx'dy'}{\int f(x,y,x',y')dxdydx'dy'}}$$

$$x_{rms}(f) = a \cdot x_{rms}(f_0),\; y_{rms}(f) = b \cdot y_{rms}(f_0),\; x'_{rms}(f) = c \cdot x'_{rms}(f_0),\; y'_{rms}(f) = d \cdot y'_{rms}(f_0)$$

For KV distribution
a_0, b_0 are from envelope equation

$$x_{rms}(f_{KV}) = \frac{a_0}{2}, \qquad y_{rms}(f_{KV}) = \frac{b_0}{2}$$

$$x'_{rms}(f_{KV}) = \frac{\varepsilon_x}{2a_0}, \qquad y'_{rms}(f_{KV}) = \frac{\varepsilon_y}{2b_0}$$

$$a = \frac{a_0}{2x_{rms}(f_0)},\; b = \frac{b_0}{2y_{rms}(f_0)},\; c = \frac{\varepsilon_x}{2a_0 x'_{rms}(f_0)},\; d = \frac{\varepsilon_y}{2b_0 y'_{rms}(f_0)}.$$

$$f_0(x,y,x',y') = \frac{1}{2\pi}\exp(-\frac{x'^2+y'^2}{2}), \qquad \left(\frac{x}{a}\right)^2 + \left(\frac{y}{b}\right)^2 < 1$$

$$x_{rms}(f_0) = y_{rms}(f_0) = \frac{1}{2},$$

$$x'_{rms}(f_0) = y'_{rms}(f_0) = 1,$$

$$a = a_0,\; b = b_0,\; c = \frac{\varepsilon_x}{2a_0},\; d = \frac{\varepsilon_y}{2b_0}.$$

Simulation Results

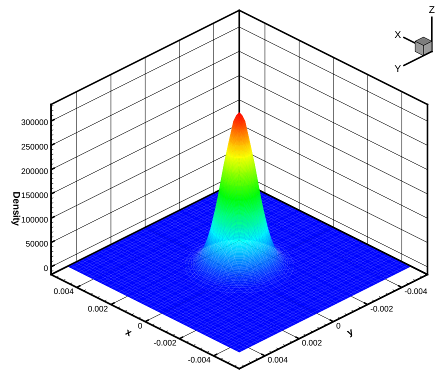
$$XX'_{rms} = \sqrt{\frac{\int X(x,y,x',y')X'(x,y,x',y')f(x,y,x',y')dxdydx'dy'}{\int f(x,y,x',y')dxdydx'dy'}}$$

$$YY'_{rms} = \sqrt{\frac{\int Y(x,y,x',y')Y'(x,y,x',y')f(x,y,x',y')dxdydx'dy'}{\int f(x,y,x',y')dxdydx'dy'}}$$

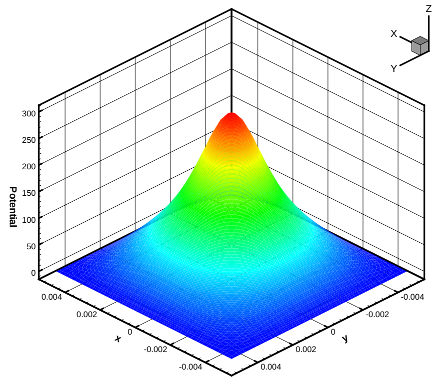
$$X_{rms} = \sqrt{\frac{\int X(x,y,x',y')^2 f(x,y,x',y')dxdydx'dy'}{\int f(x,y,x',y')dxdydx'dy'}}$$

$$Y_{rms} = \sqrt{\frac{\int Y(x,y,x',y')^2 f(x,y,x',y')dxdydx'dy'}{\int f(x,y,x',y')dxdydx'dy'}}$$

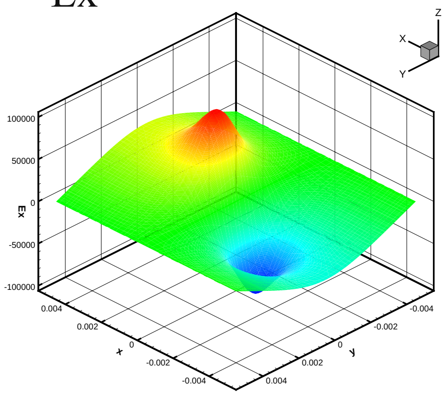
Density



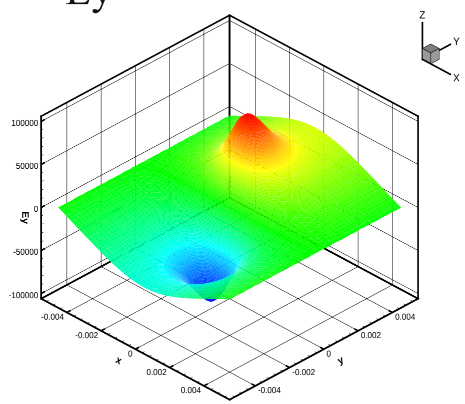
-Potential

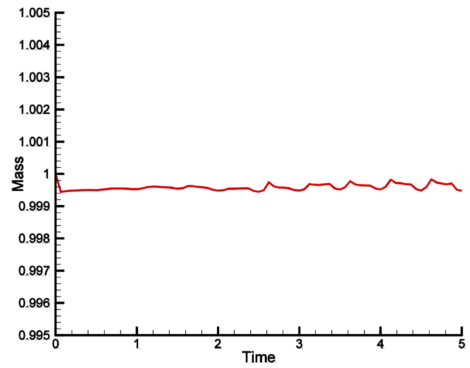


Ex

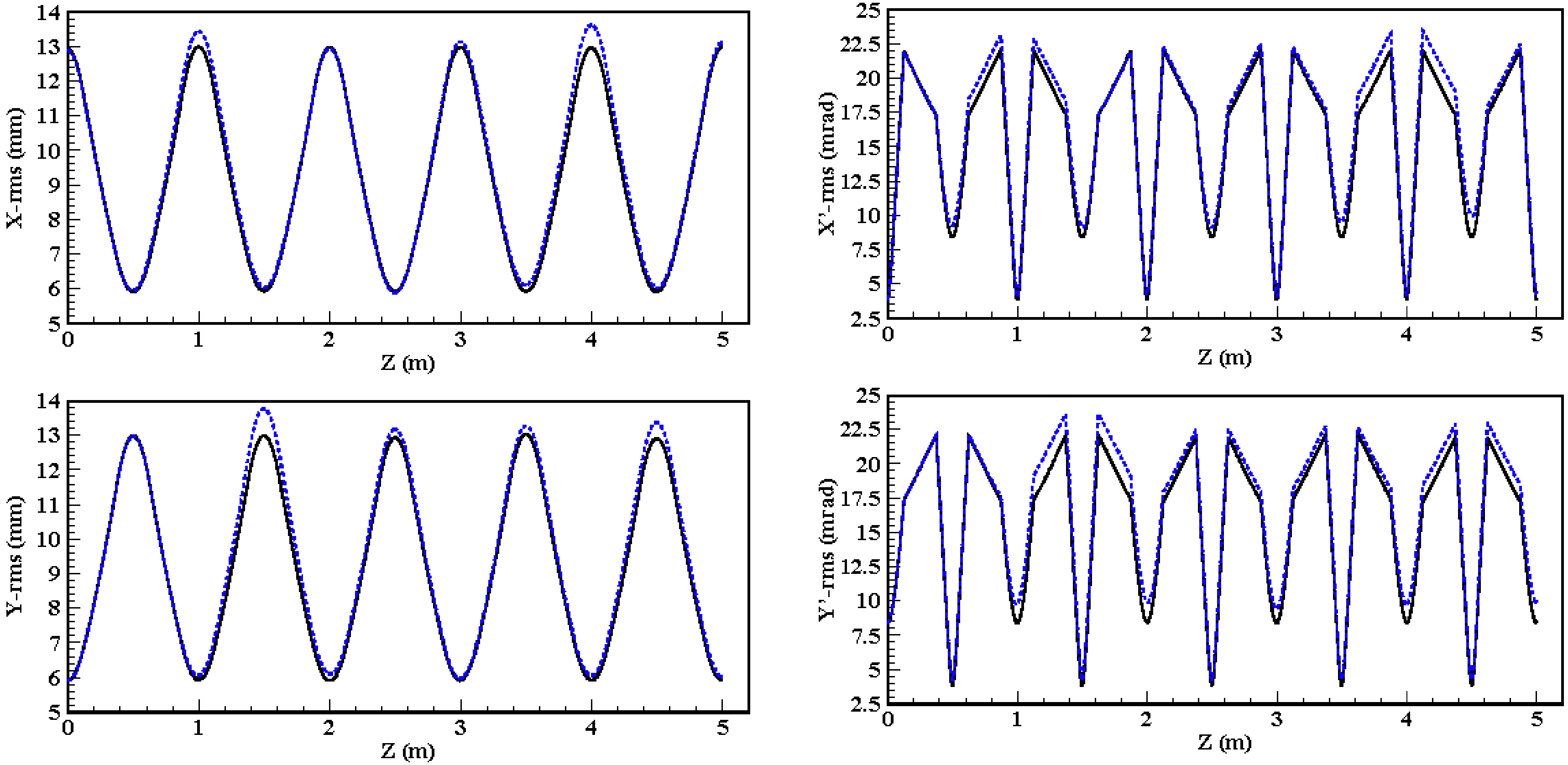


Ey

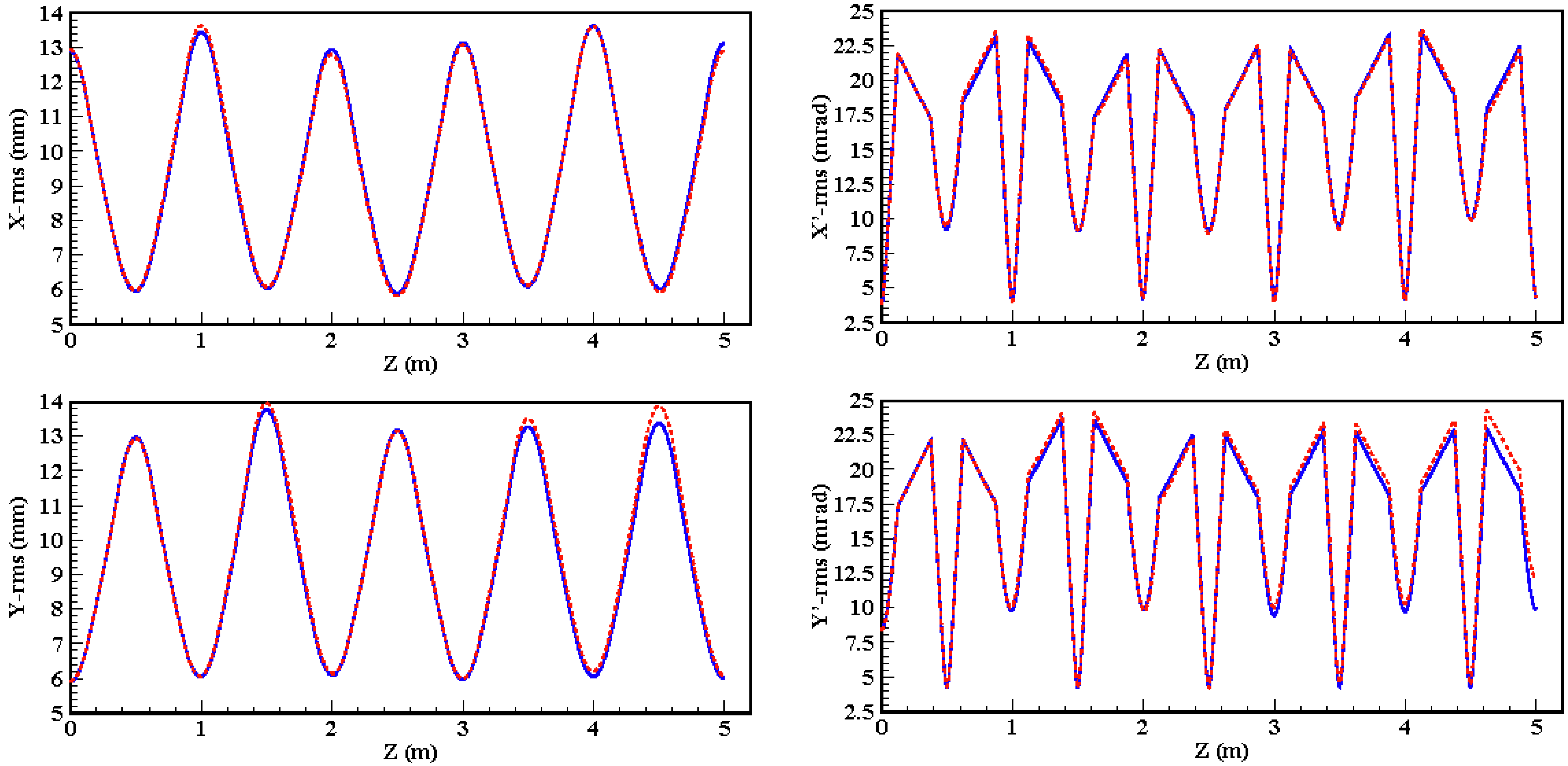




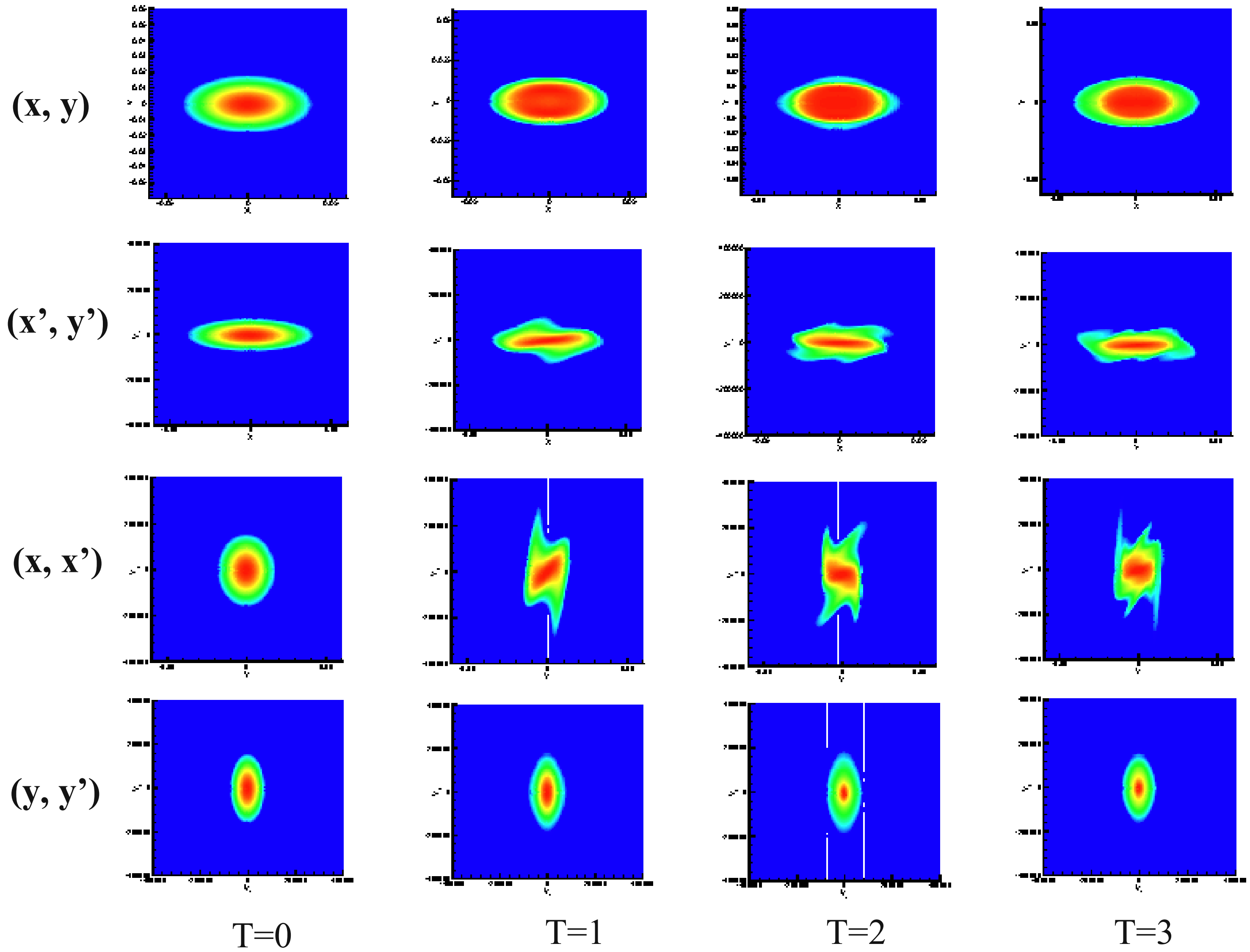
Comparison TRACE2D (solid) and TRACK (dotted) simulations using a KV beam:



Comparison of TRACK (solid) and Vlasov (dotted) simulations using a Gaussian beam



Simulation Results



Discontinuous Galerkin Method

- Linear hyperbolic equation

$$\frac{\partial u(\vec{x},t)}{\partial t} + \nabla \cdot \vec{f}(u(\vec{x},t),\vec{x},t) = 0,$$
$$u(\vec{x},t) = g(\vec{x},t), \quad \vec{x} \in \partial\Omega$$
$$u(\vec{x},0) = f(\vec{x}), \quad t = 0$$

- Continuous Galerkin

$$\int_{D^k} \left[\frac{\partial u_h^k}{\partial t} + \nabla \cdot \vec{f}_h^k \right] \varphi_i^k(\vec{x}) d\vec{x} = 0$$

- Advantages:
  Strong stability
  Local mass matrix,
  easy to invert ,
  easy parallelization

- Strong and weak forms of DG

Strong form:

$$\int_{D^k} \left[\frac{\partial u_h^k}{\partial t} + \nabla \cdot \vec{f}_h^k \right] \varphi_i^k(\vec{x}) d\vec{x} = \int_{\partial D^k} \vec{n} \cdot \left[\vec{f}_h^k - \vec{f}^* \right] \varphi_i^k(\vec{x}) d\vec{x}$$

Weak form:

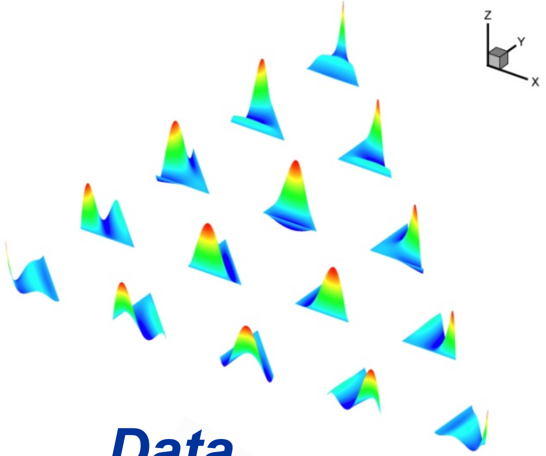
$$\int_{D^k} \left[\frac{\partial u_h^k}{\partial t} \varphi_i^k(\vec{x}) - \vec{f}_h^k \cdot \nabla \varphi_i^k(\vec{x}) \right] d\vec{x} = -\int_{\partial D^k} \vec{n} \cdot \vec{f}^* \varphi_i^k(\vec{x}) d\vec{x}$$

- Disadvantages: More memory

$$\frac{\partial \vec{B}}{\partial t} = -\nabla \times \vec{E}, \quad \nabla \cdot \vec{E} = \frac{\rho}{\varepsilon_0}$$
$$\varepsilon_0 \mu_0 \frac{\partial \vec{E}}{\partial t} = \mu_0 \vec{J} - \nabla \times \vec{B}, \quad \nabla \cdot \vec{B} = 0$$

2D2V Unstructured Mesh

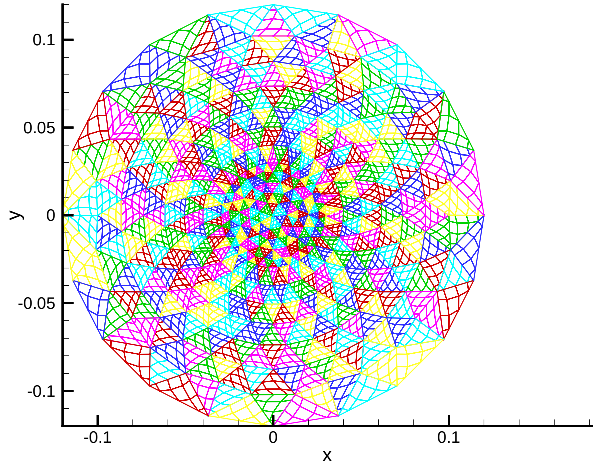
Nodal Based
hp FEM



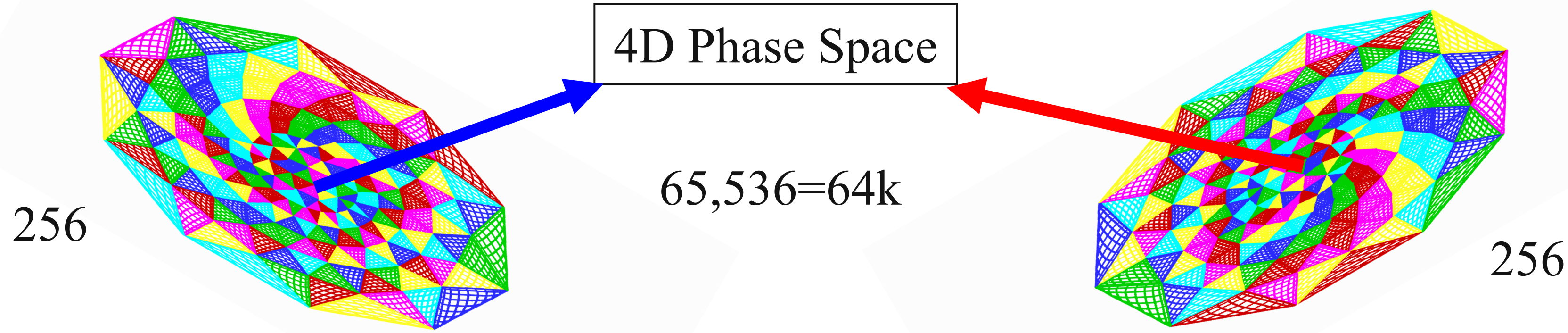
For E=1586, P=3 in 2D plane

Solving $2 \times 1586 \times 10 = 32720$
2D transport equations
Each of the equation has
15860 dof!

Total dof is 0.5 billion!



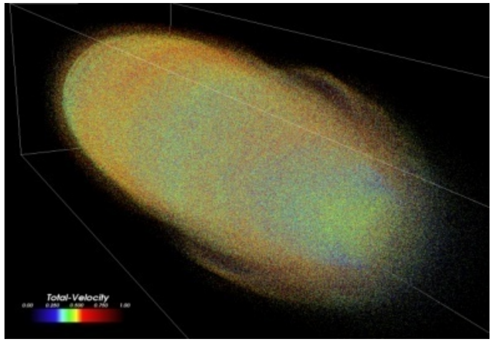
Data Structure

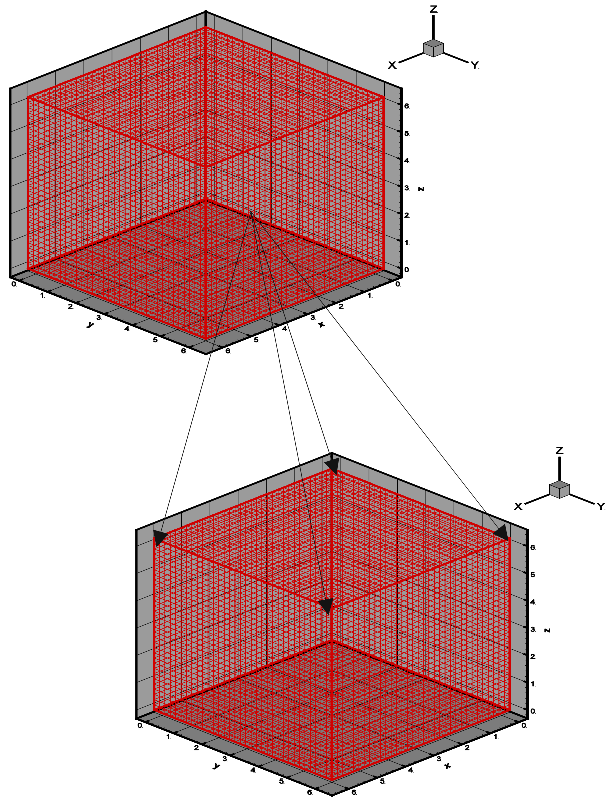


J. Xu, P.N. Ostroumov, J. Nolen and K.J Kim, "***Scalable Direct Vlasov Solver with Discontinuous Galerkin Method on Unstructured Mesh***", **Submitted to SIAM Journal on Scientific Computing.**

Equivalence between PIC and DVS

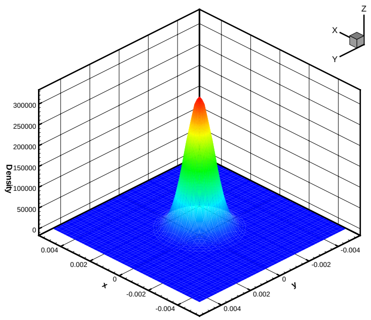
- Macro particles vs. Grid points
- Lagrange vs. Euler approach
- Uniform sampling vs. non-uniform sampling
- Both solving Poisson's equation on fixed grid
- Statistics over particles vs. grid points





Semi-Lagrange vs. DG method

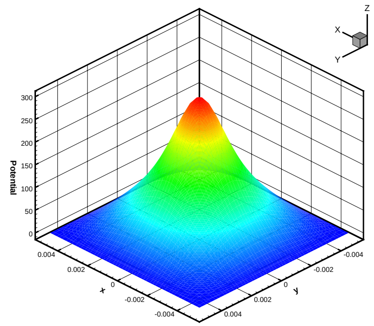
- Both have strong stability;
- DG has limitation on time step;
- Concentrated distribution function presents the challenges for DVS;



$$\frac{\partial}{\partial x}, \frac{\partial}{\partial y}$$

>>

Interpolation Error



Summary and Prospect

- Systematic software packages have been developed for Beam dynamic simulations (BDS) on peta-scale supercomputers;
- Scalable Poisson solvers have been developed and applied in BDS;
- Semi-Lagrange method has been used to solve Vlasov equation in both 1D1V and 2D2V on structured mesh;
- Discontinuous Galerkin method has been used to solve Vlasov equation in 2D2V on unstructured mesh;
- Semi-Lagrange method will be applied on unstructured mesh in the future;
- Numerical diffusion is the critical challenge for successful DVS;
- Developed DVS solvers can be applied to some plasma and 2D beams in accelerator devices;
- 3D3V simulations can use same algorithms as 2D2V, but need more scalable algorithms and faster supercomputers.