An Object-Oriented Parallel Particle-In-Cell Code for Beam Dynamics Simulation in Linear Accelerators

Ji Qiang, Robert Ryne and Salman Habib

Los Alamos National Laboratory

Viktor Decyk

University of California at Los Angeles

Work performed under the auspices of the
DOE Grand Challenge in Computational Accelerator Physics
using resources at the
Advanced Computing Laboratory and the
National Energy Research Scientific Computing Center

Outline

- Introduction
- Physical Model and Numerical Method
- Object-Oriented Parallel Implementation
- Performance Test
- Applications and Visualization
- Conclusions

Importance of Particle Accelerators

- **Basic Physics**
  - elementary particles
  - nuclear structure
- **Biological and Material Sciences**
  - ribosome complex
  - high temperature superconductivity
- **Medicine Application**
  - irradiation therapy
- **Solving Problems of National Importance**
  - nuclear waste transmutation
  - accelerator-driven energy production

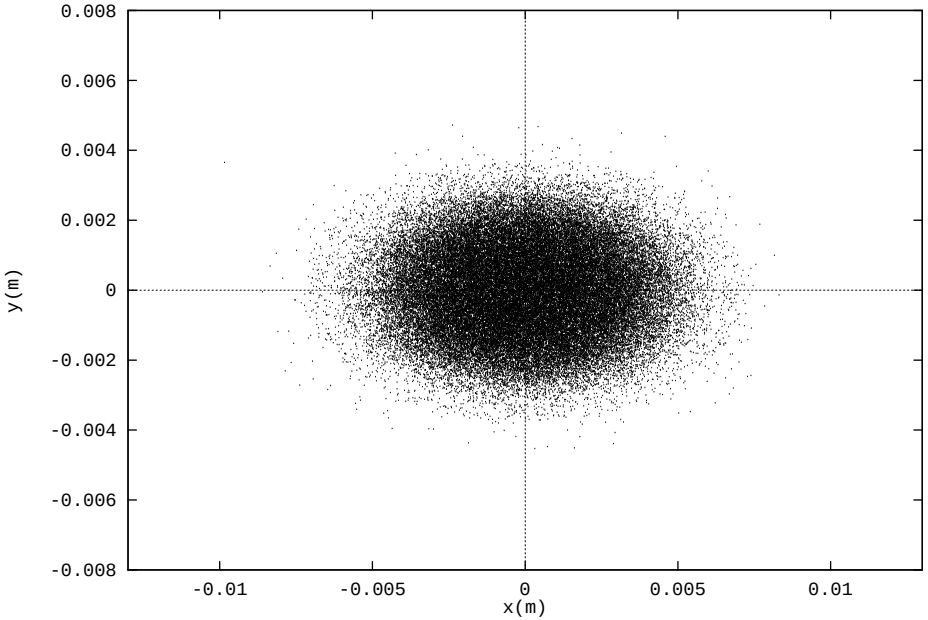
Accelerators: Expensive and Complex

- Billion dollar-class facilities under consideration:
  - Spallation Neutron Source
  - Next Linear Collider
  - 4th Generation Light Source:
  - Muon Accelerator/Neutrino Factory
- Complex Systems:
  - highly nonlinear dynamics
  - large range of spatial scales: cm - to - nanometers
  - multi-particle interactions

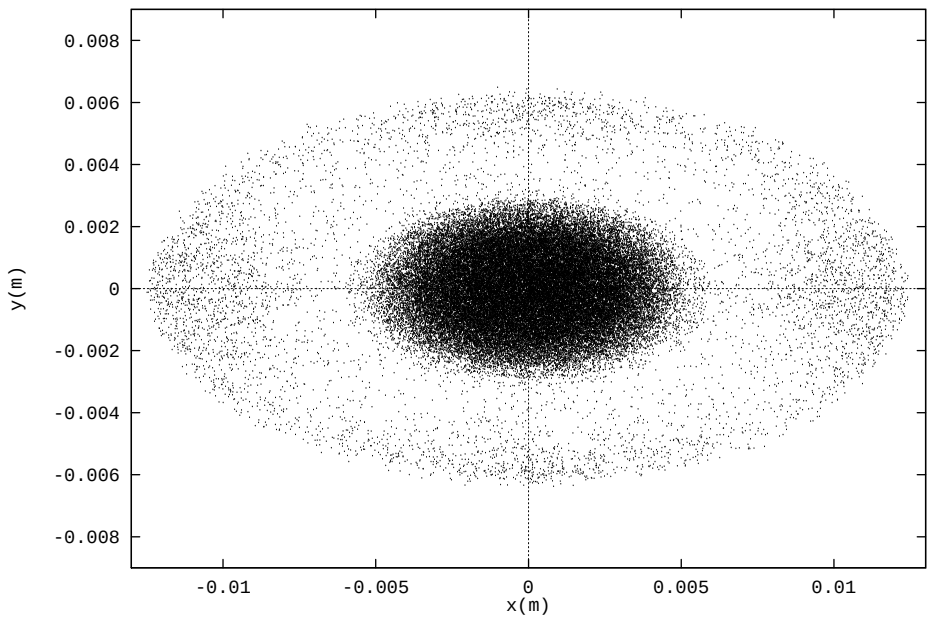
Motivation for Grand Challenge in Computational Accelerator Physics

- Modeling on parallel computers essential for design decisions
  - evaluate/reduce risk
  - reduce cost
  - optimize performance
- Example problems:
  - predicting beam halo for Spallation Neutron Source
  - designing ***large*** electromagnetic structures for Next Linear Collider

Halo Formation from Mismatched Beam



Matched beam
(correct initial condition)



Mismatched beam
(incorrect initial condition)

Large Scale Beam Dynamics Simulations are Needed to Accurately Model Beam Halo

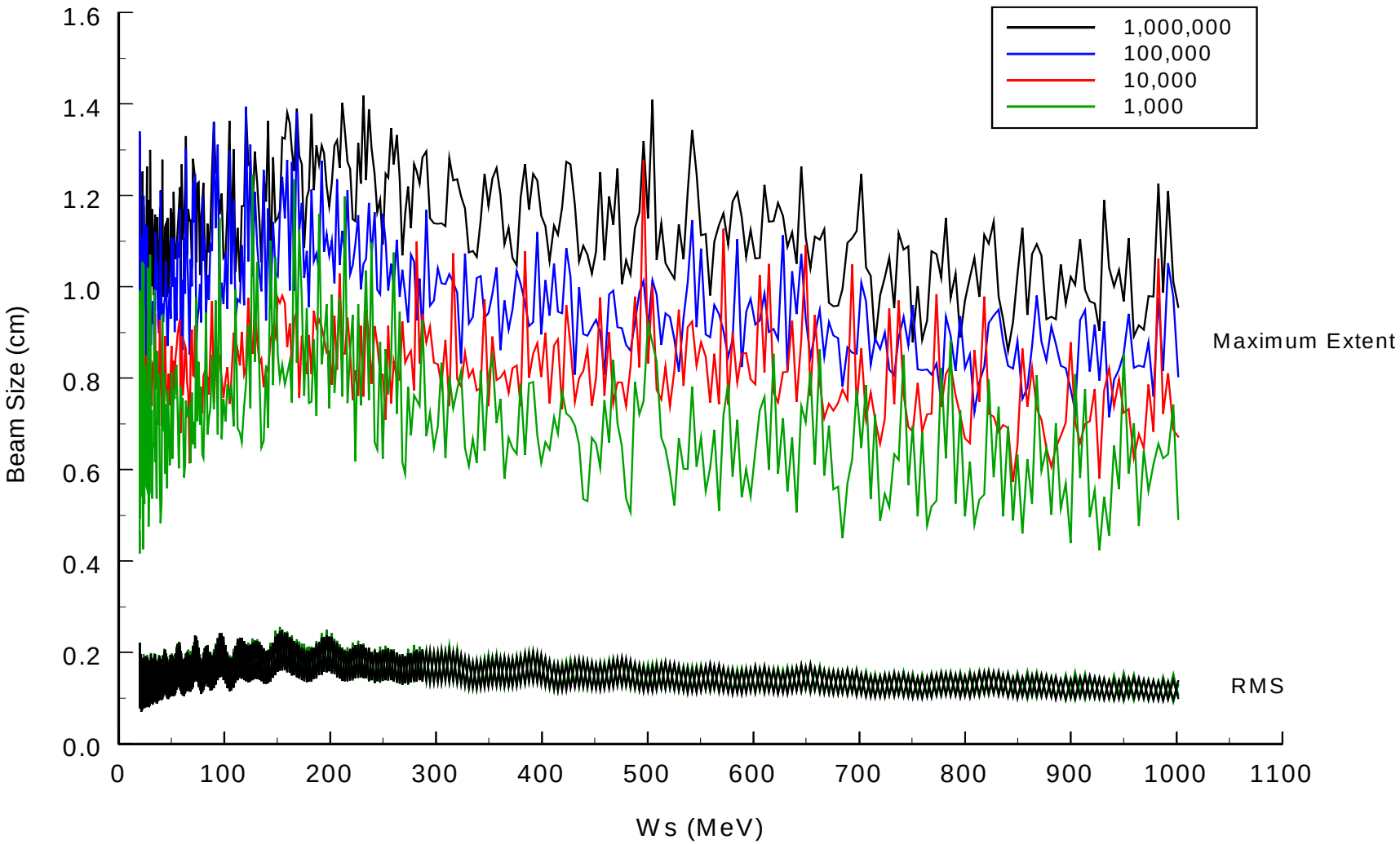
- Future high average power accelerators will have to operate with very low losses
- Excess losses
  - cause radioactivation of accelerator components
  - reduce reliability and availability

Allowed loss is ~ 1 - 10 particles out of 1,000,000,000

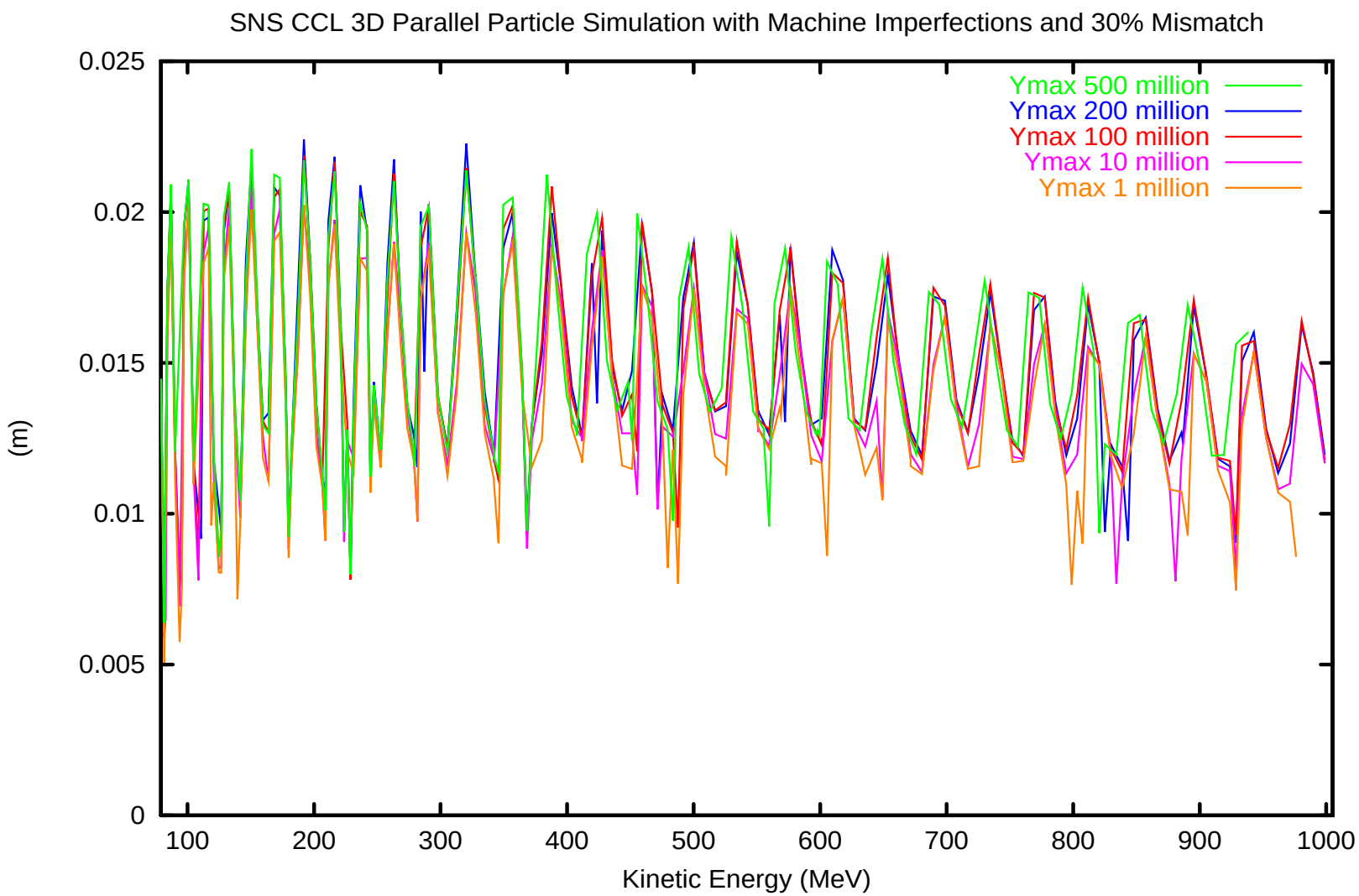
- Major loss mechanism: large amplitude "halo" particles striking the beam pipe

High-Resolution modeling using 0.1-1 billion particles is necessary to make quantitative predictions of beam halo

PC Simulation of CCDTL/CCL Beam Profile with Machine Imperfections, No Mismatch



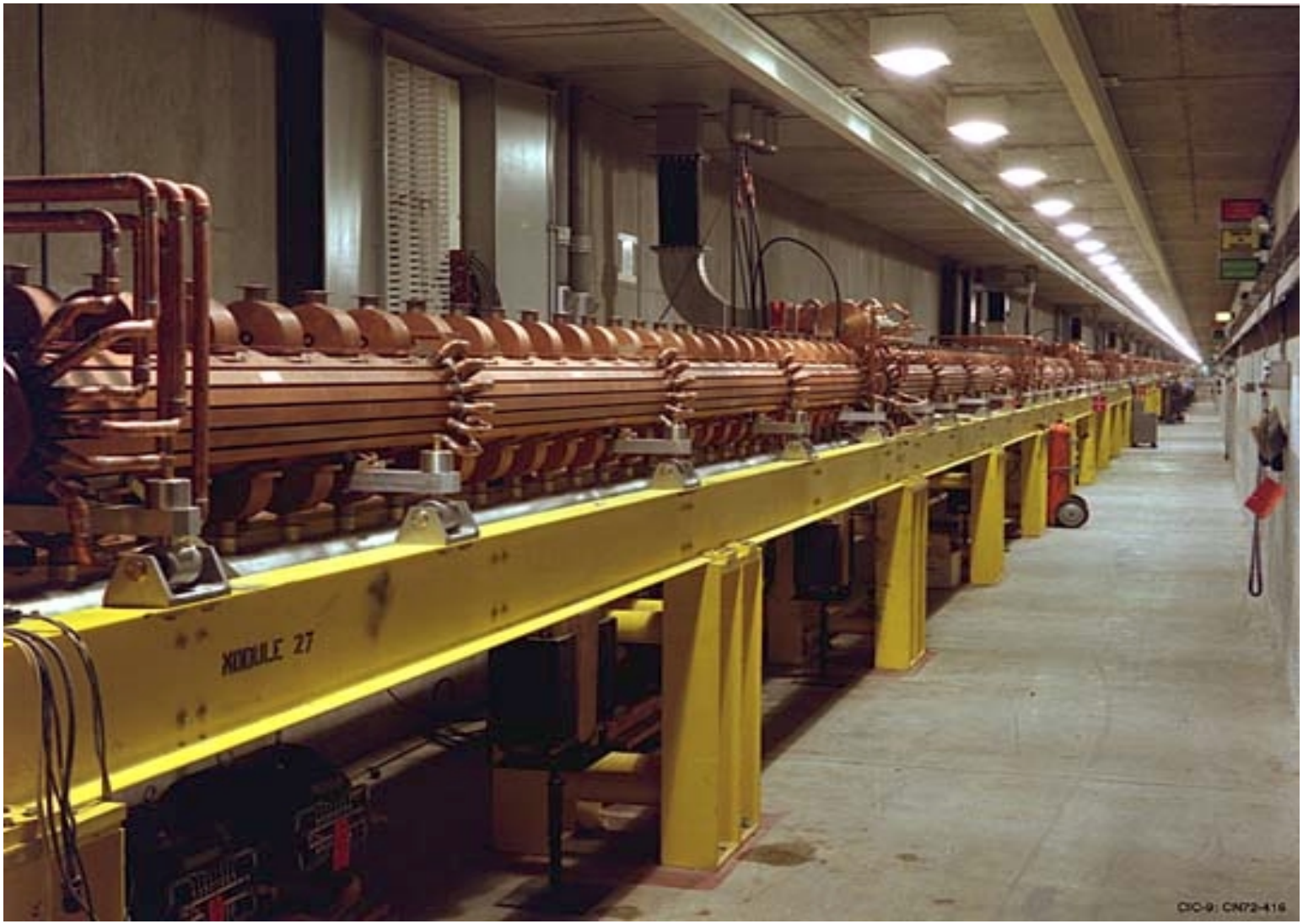
Maximum Particle Amplitude in the SNS Linac with Varying # of Simulation Particles



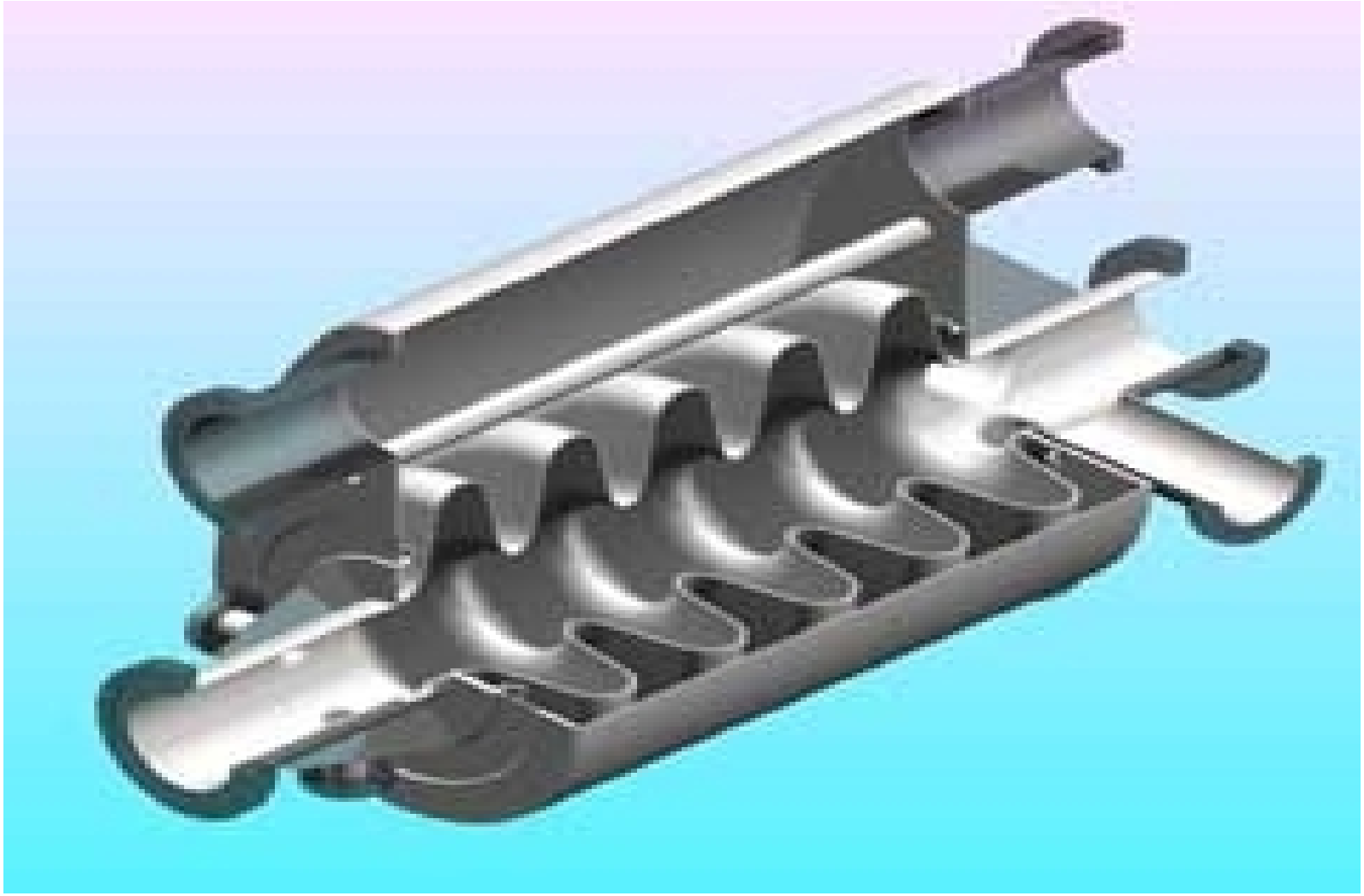
Physical Model and Numerical Method

- Main components of linear accelerator:
  - Drift space
  - Magnetic quadrupoles for transverse focusing
  - RF structures for acceleration and longitudinal focusing
- Charged particles subject to:
  - External fields
  - Self-fields due to Coulomb interaction
- Evolution of charged particle distribution is governed by Poisson-Vlasov equations.

The Linear Accelerator at Los Alamos Neutron Science Center



A Superconducting RF Cavity Structure



Particle-In-Cell Simulation with Split-Operator Method

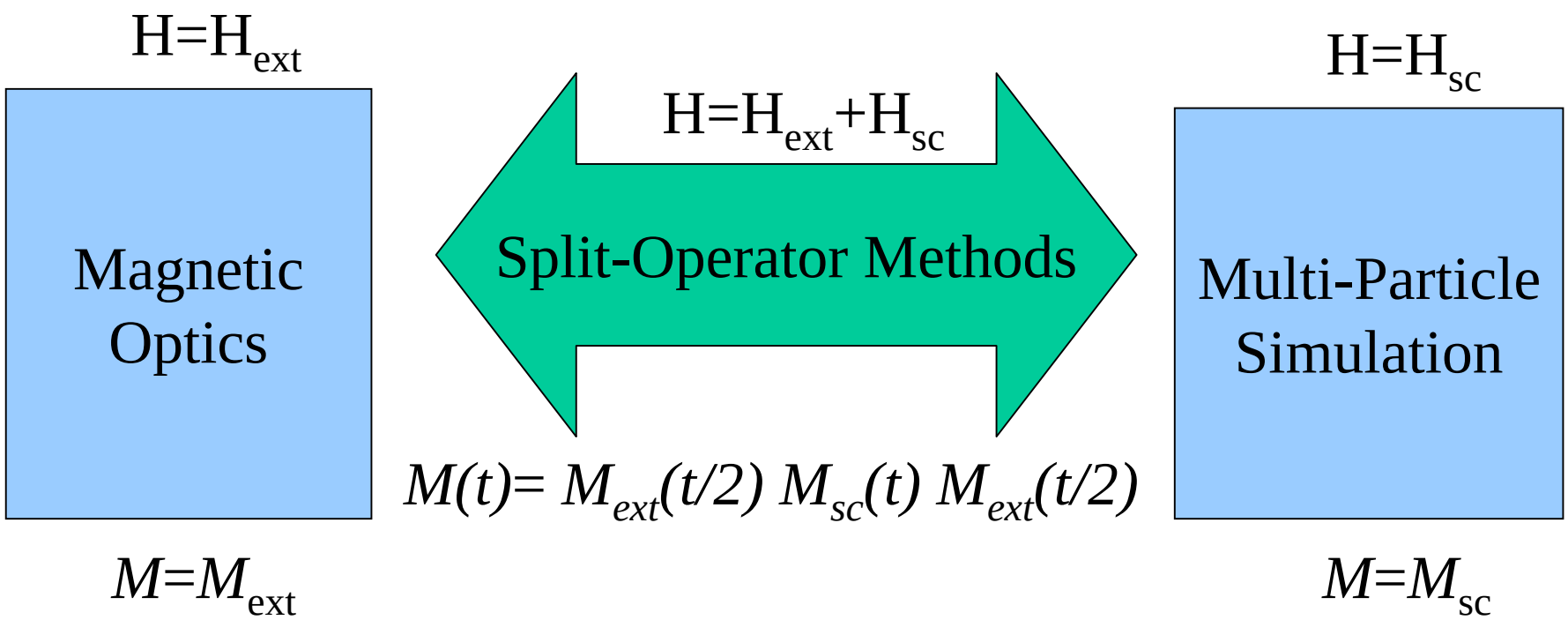
- Particle-in-cell approach:
  - Charge deposition on a grid
  - Field solution via FFT-based convolution with open boundary conditions
  - Field interpolation from grid to particles
- Split-operator method with $H = H_{external} + H_{space\ charge}$
- Philosophy:

  Do not take tiny steps to push ~100 million particles.

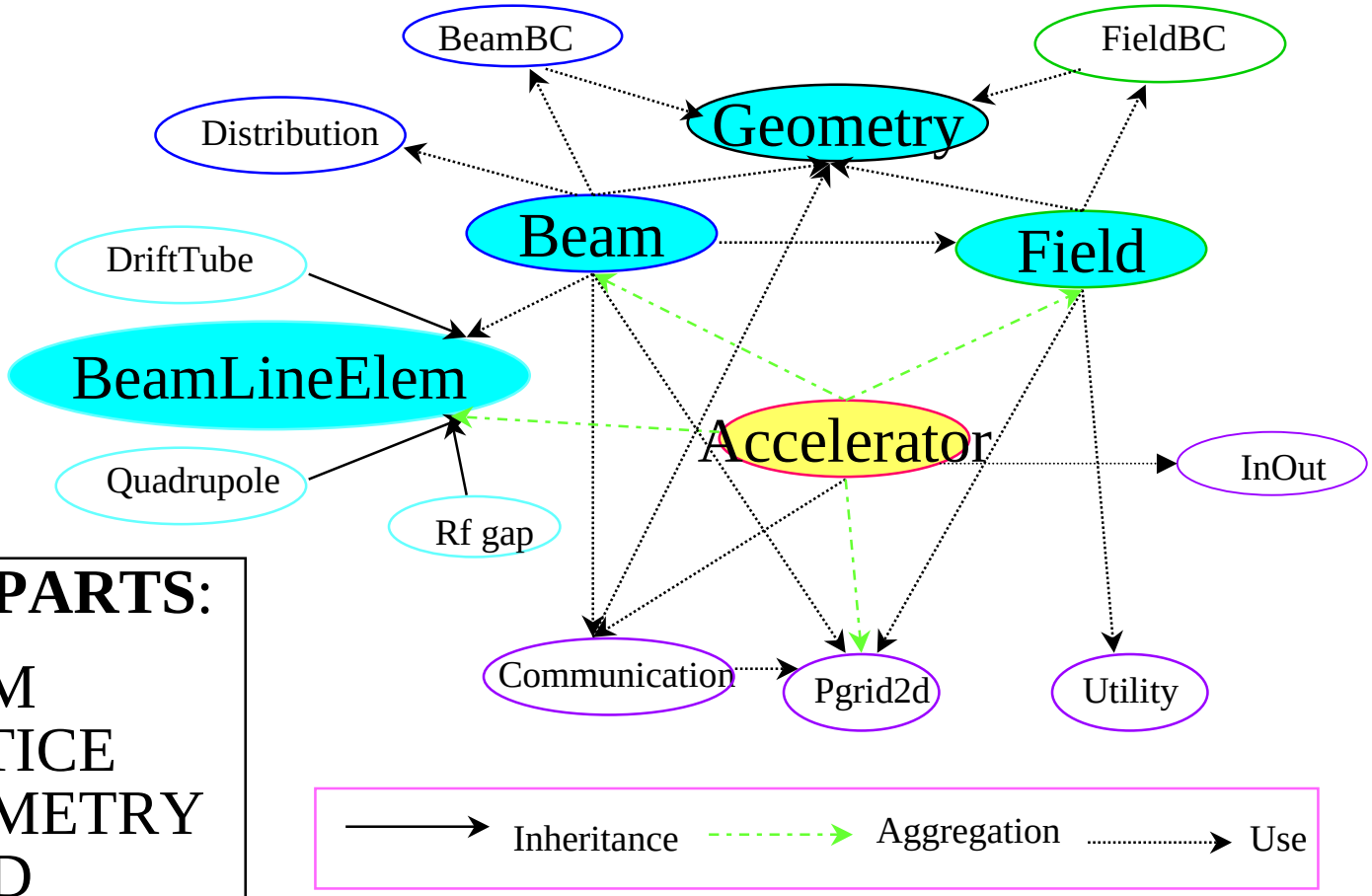
  Do take tiny steps to compute maps, then push w/ maps

Split-Operator Approach

How to turn ***any*** Magnetic Optics code into a tracking code with space charge



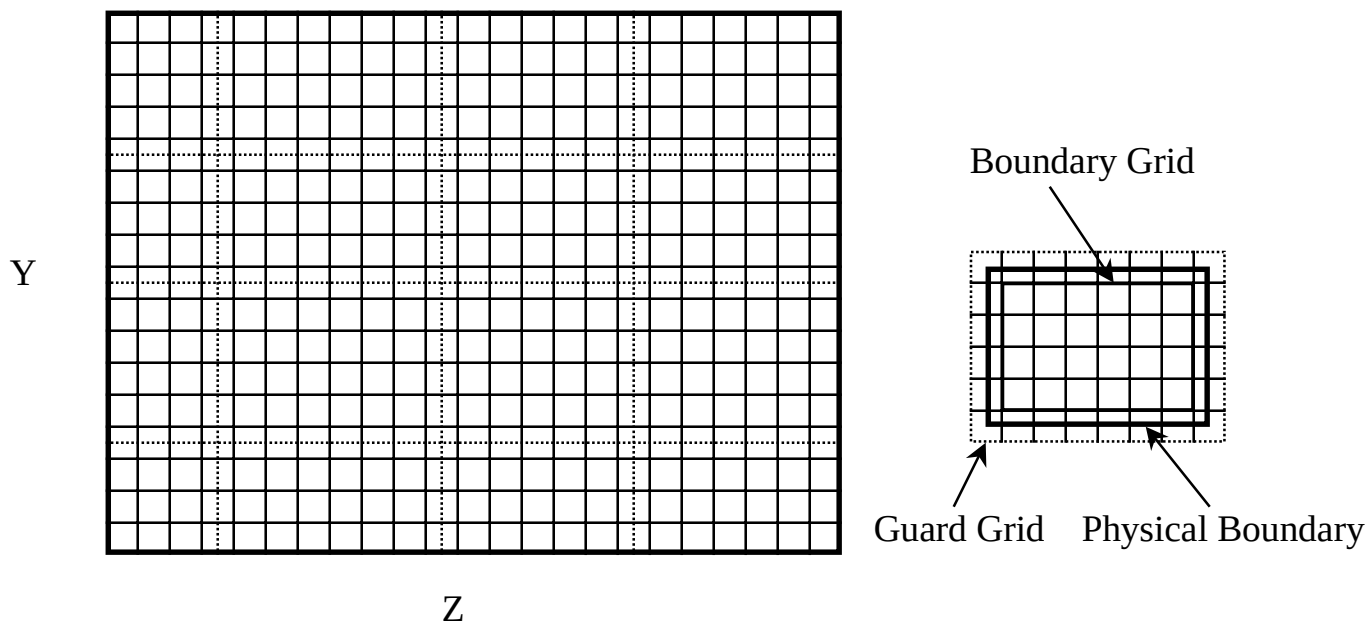
Class Diagram of Accelerator Beam Dynamics System



FIVE PARTS:

BEAM
LATTICE
GEOMETRY
FIELD
AUXILIARY

Two-Dimensional Domain Decomposition



Explicit Communication

- Particle manager:
  - sends the particles to the neighboring processor
- Field manager:
  - exchanges the information between the boundary grids and guard grids

Dynamic Load Balance

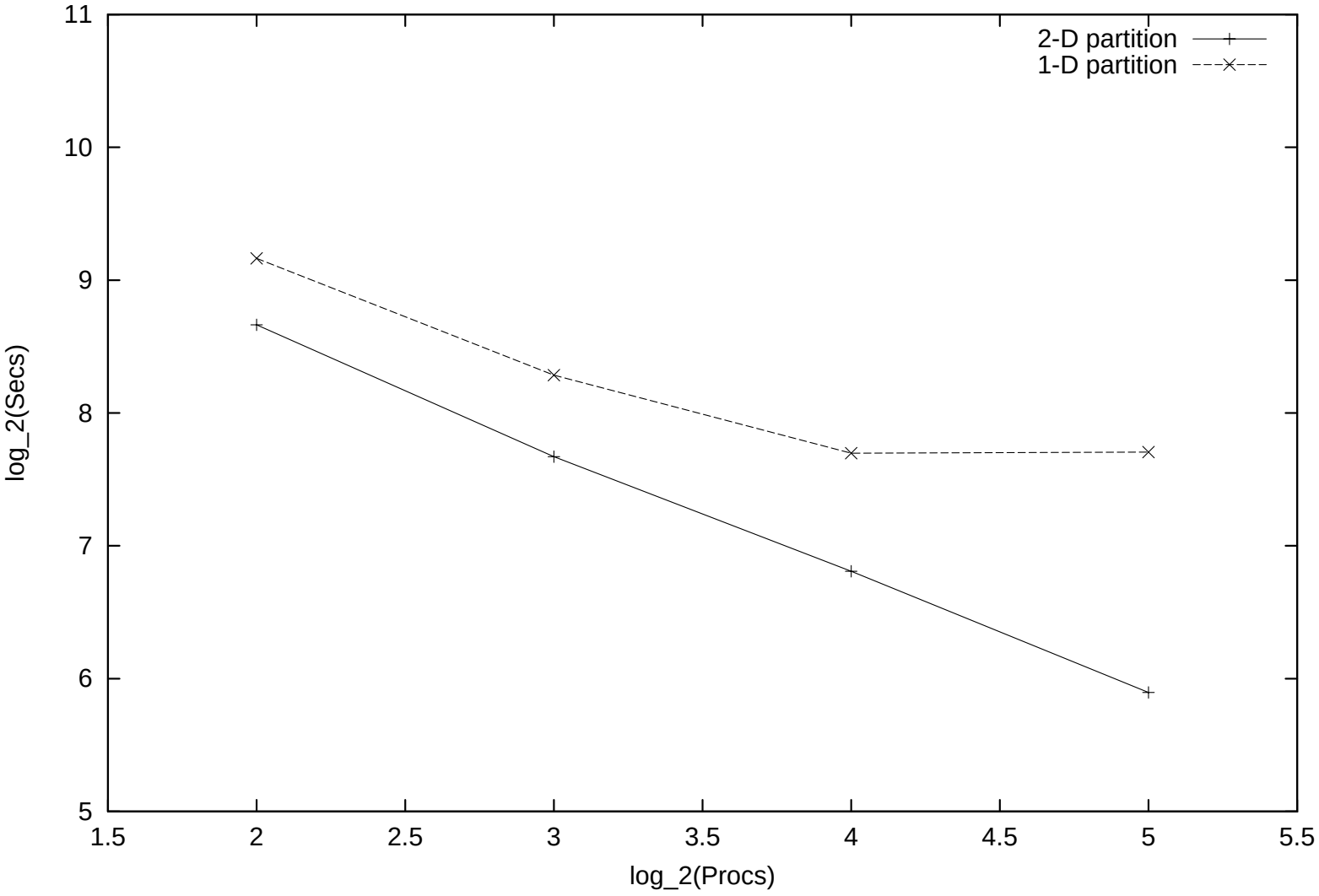
- Define a 2-D density distribution function
- Assume that it is separable in y and z
- Load balance in the y direction
- Load balance in the z direction

IMPACT: Integrated Map and Particle Accelerator Tracking Code

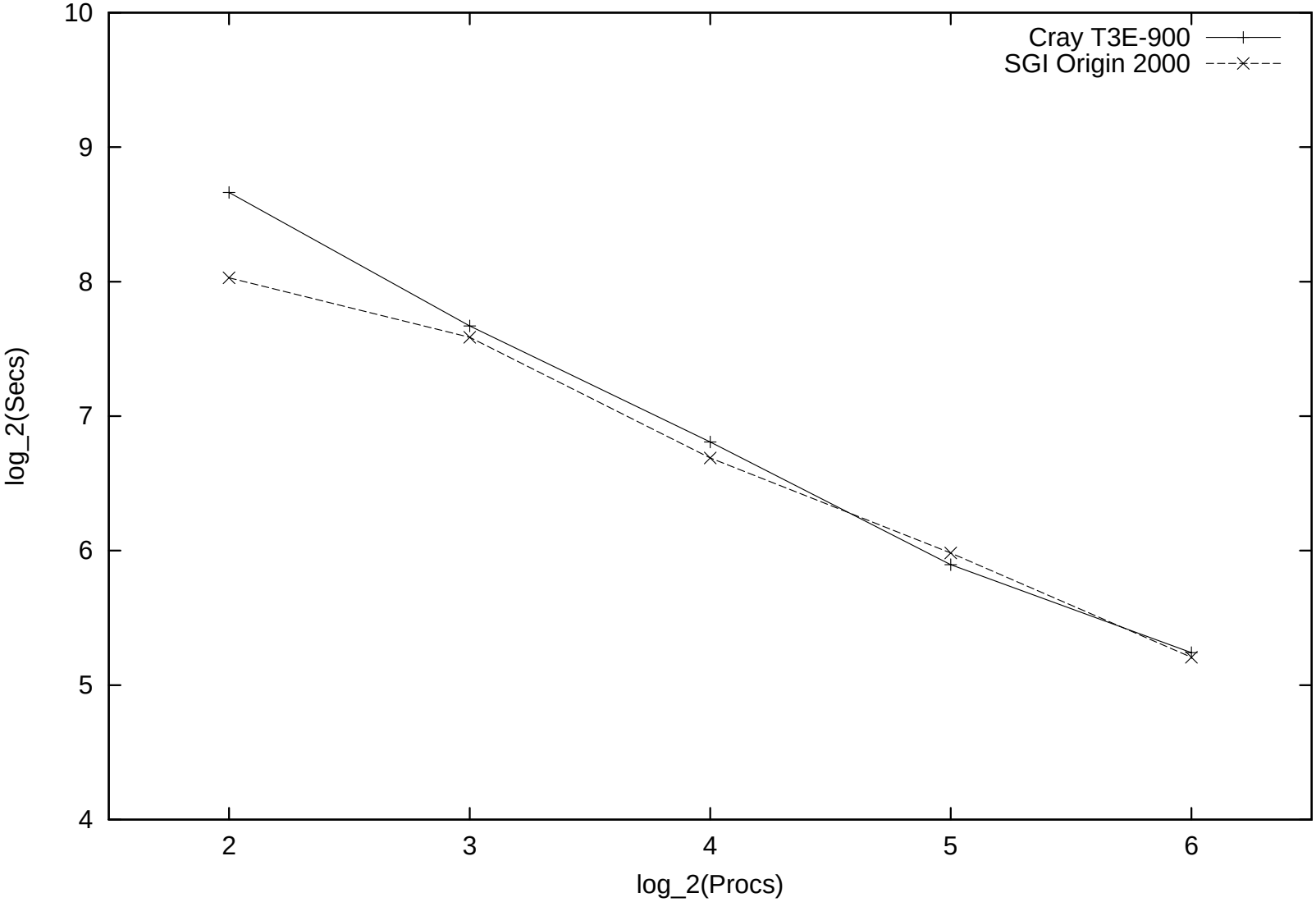
Main Features:

- Use of split-operator method
- Fully three-dimensional space charge calculation
- Relativistic effects included
- Dynamic load balance
- Choice of particle managers
- Parallel implementation using F90/MPI, POOMA, HPF

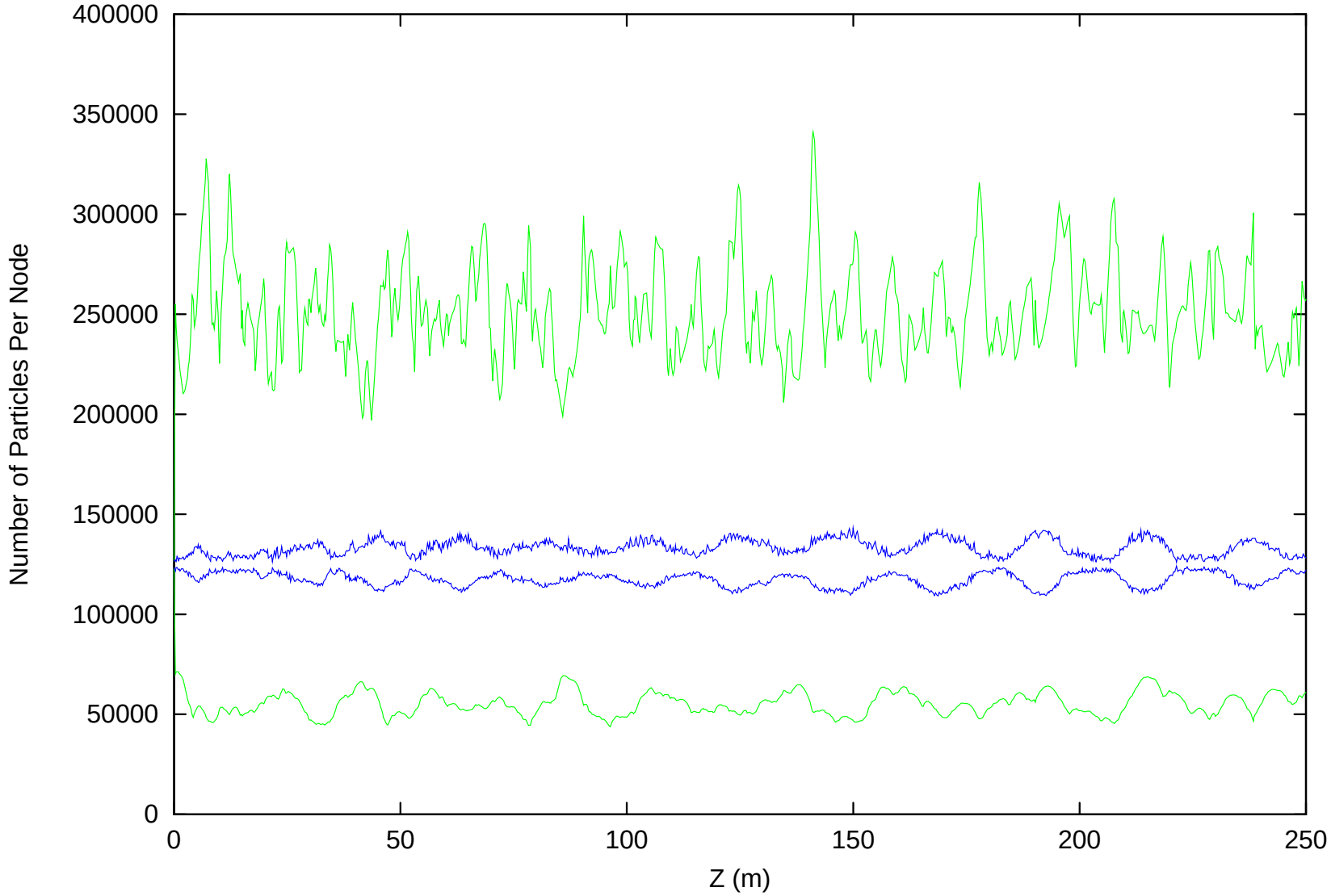
IMPACT Scaling for Different Domain Decomposition



IMPACT Scaling with # of Processors

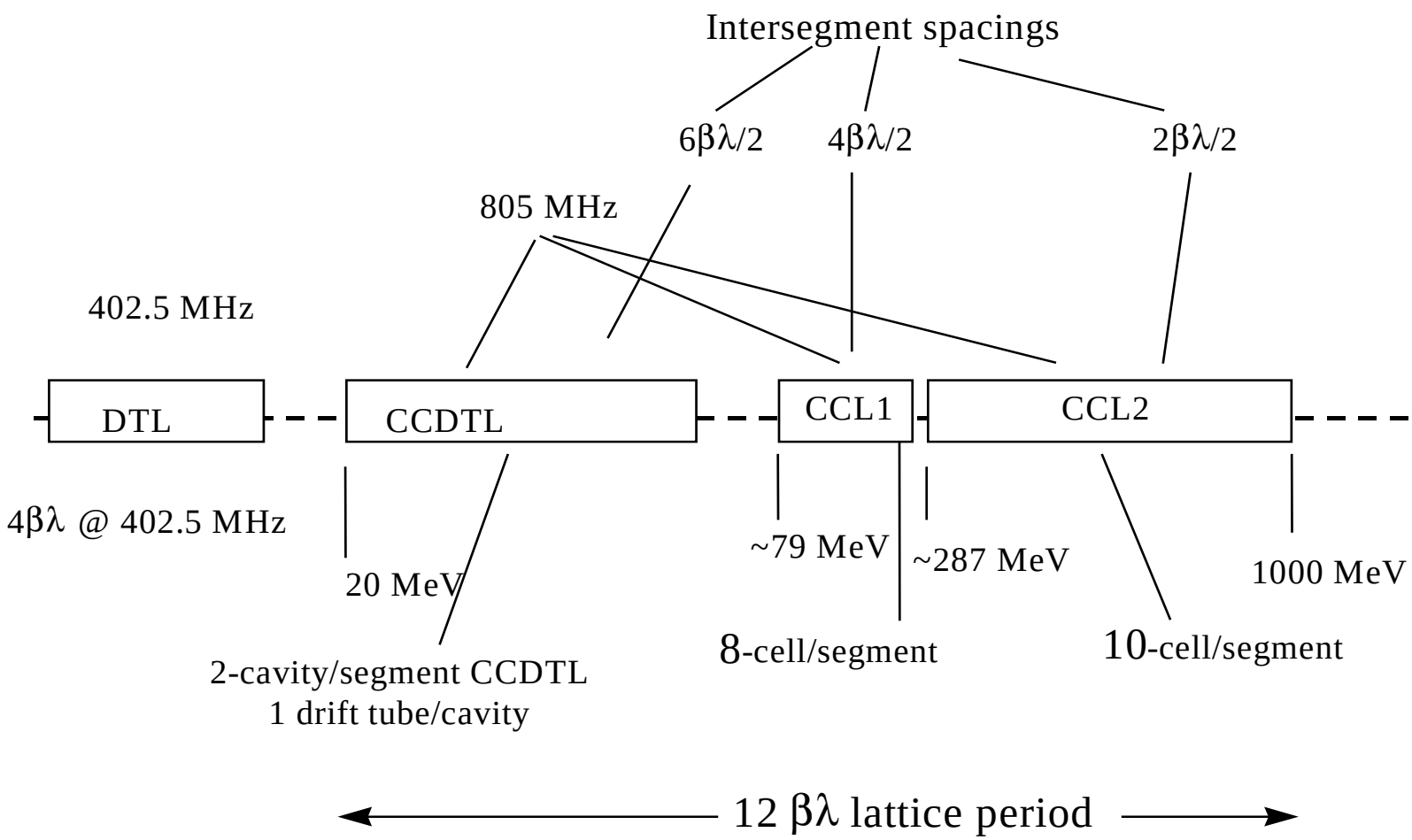


IMPACT: Success of Dynamic Load Balancing

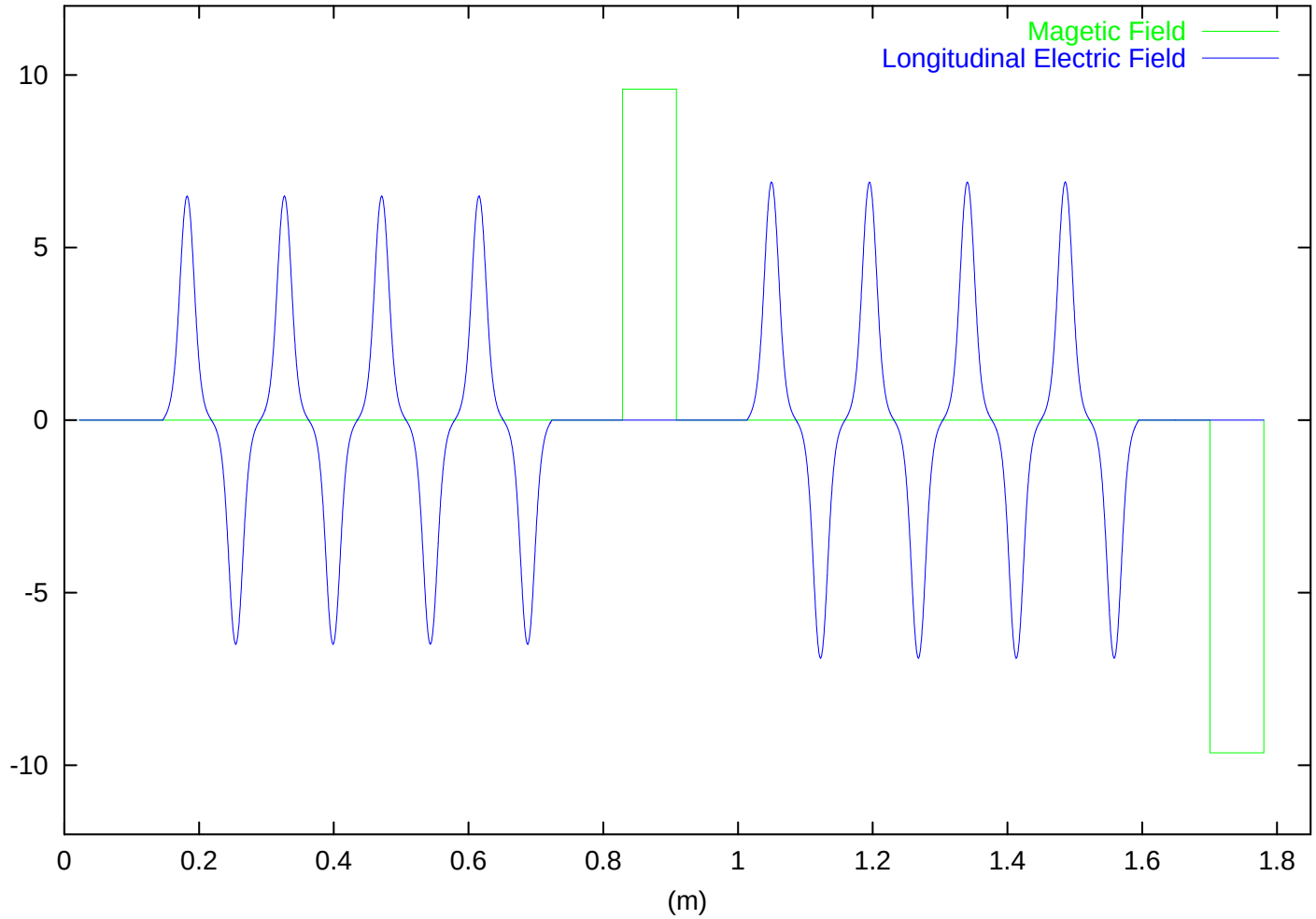


Spallation Neutron Source

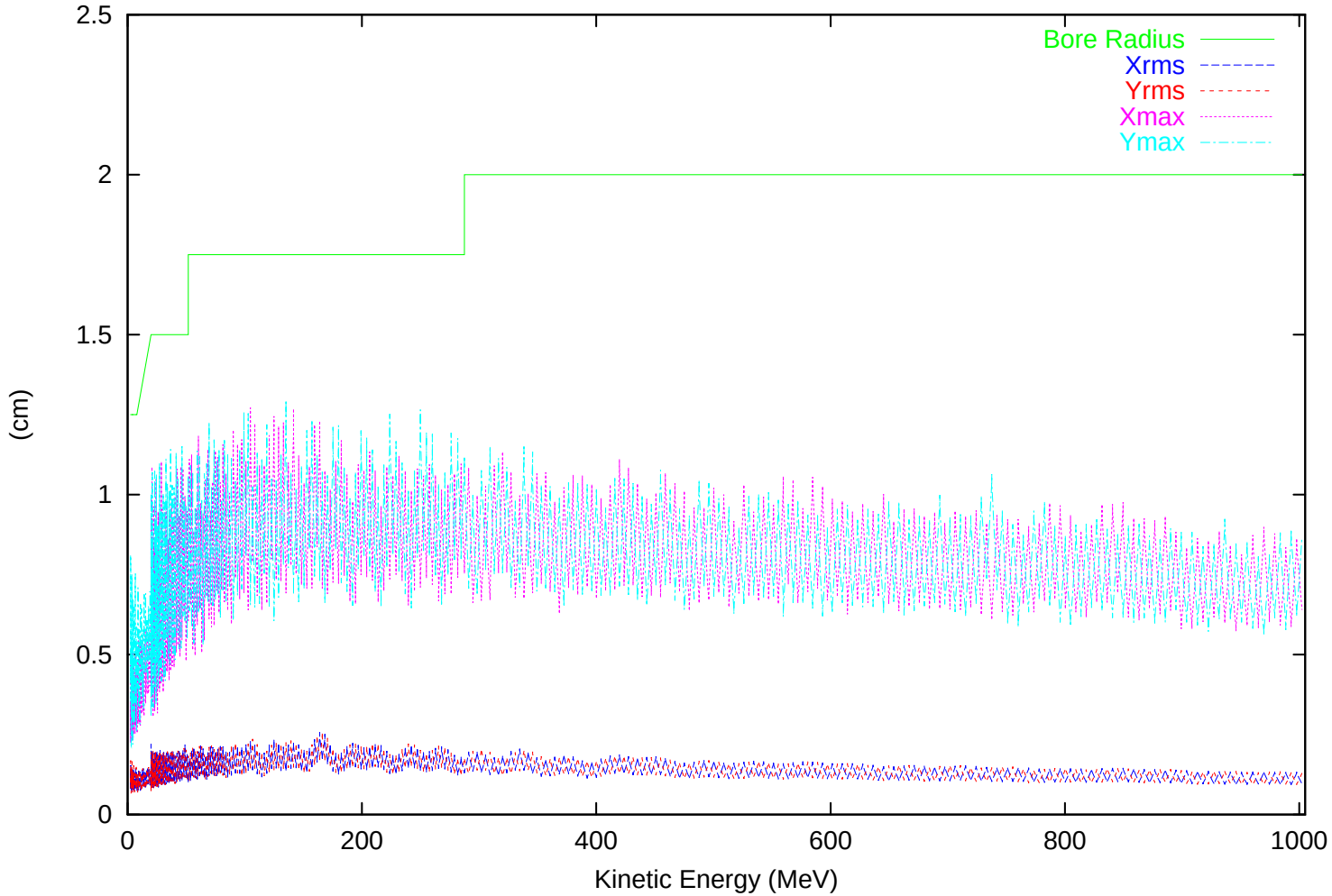
SNS Linac Configuration



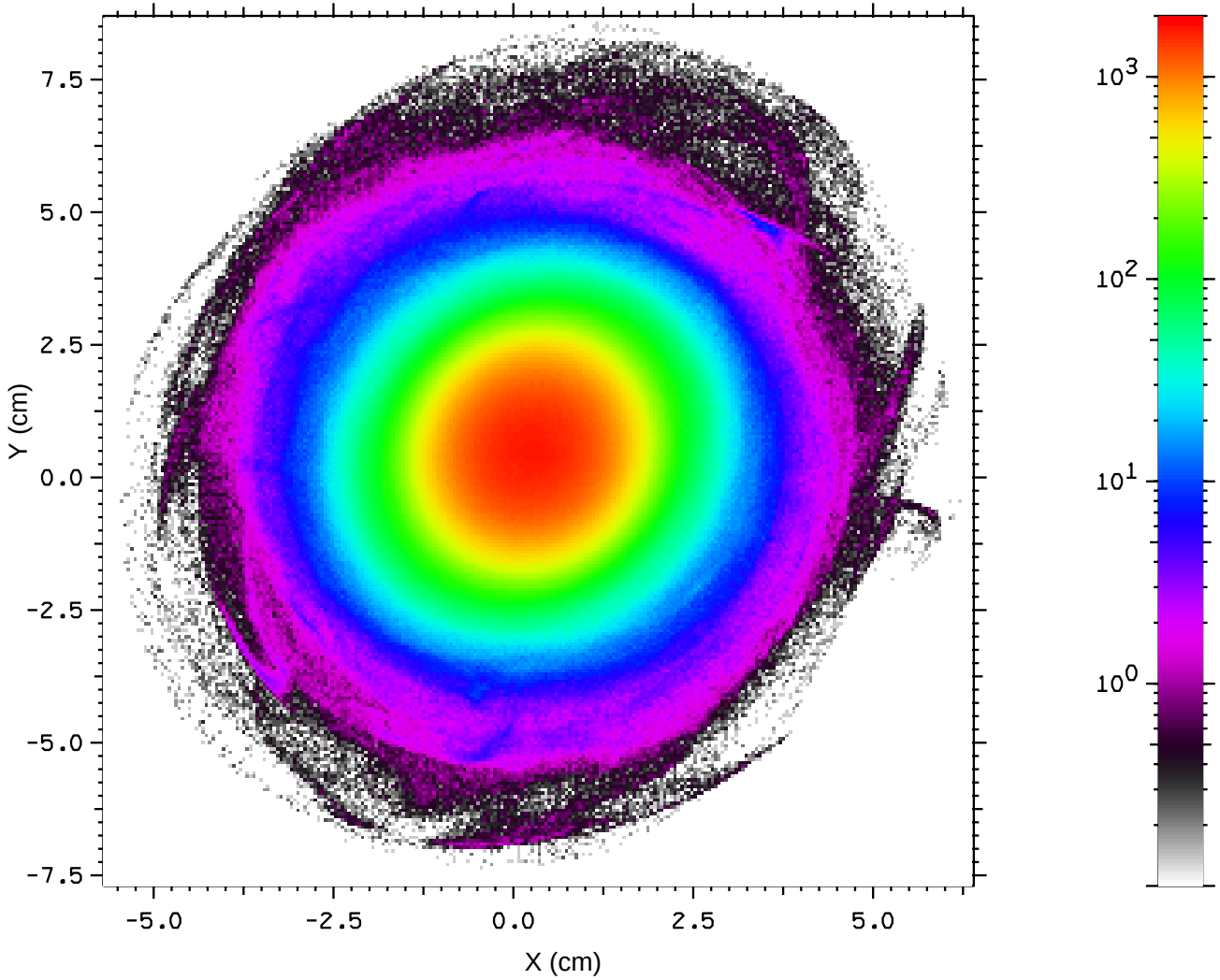
Field Distribution in One Lattice Period



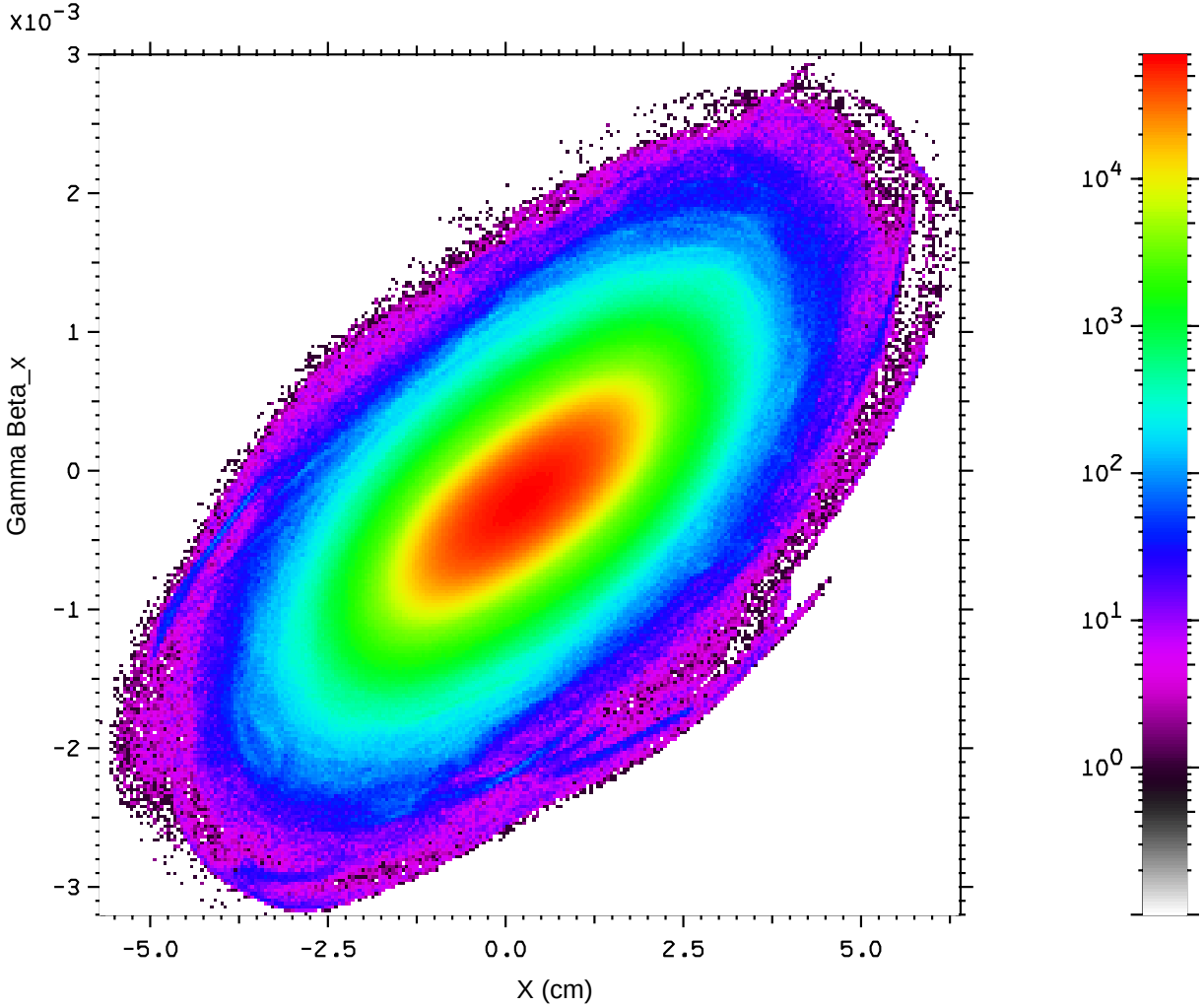
Transverse Beam Size as a Function of Energy Using 100 Million Particle Simulation



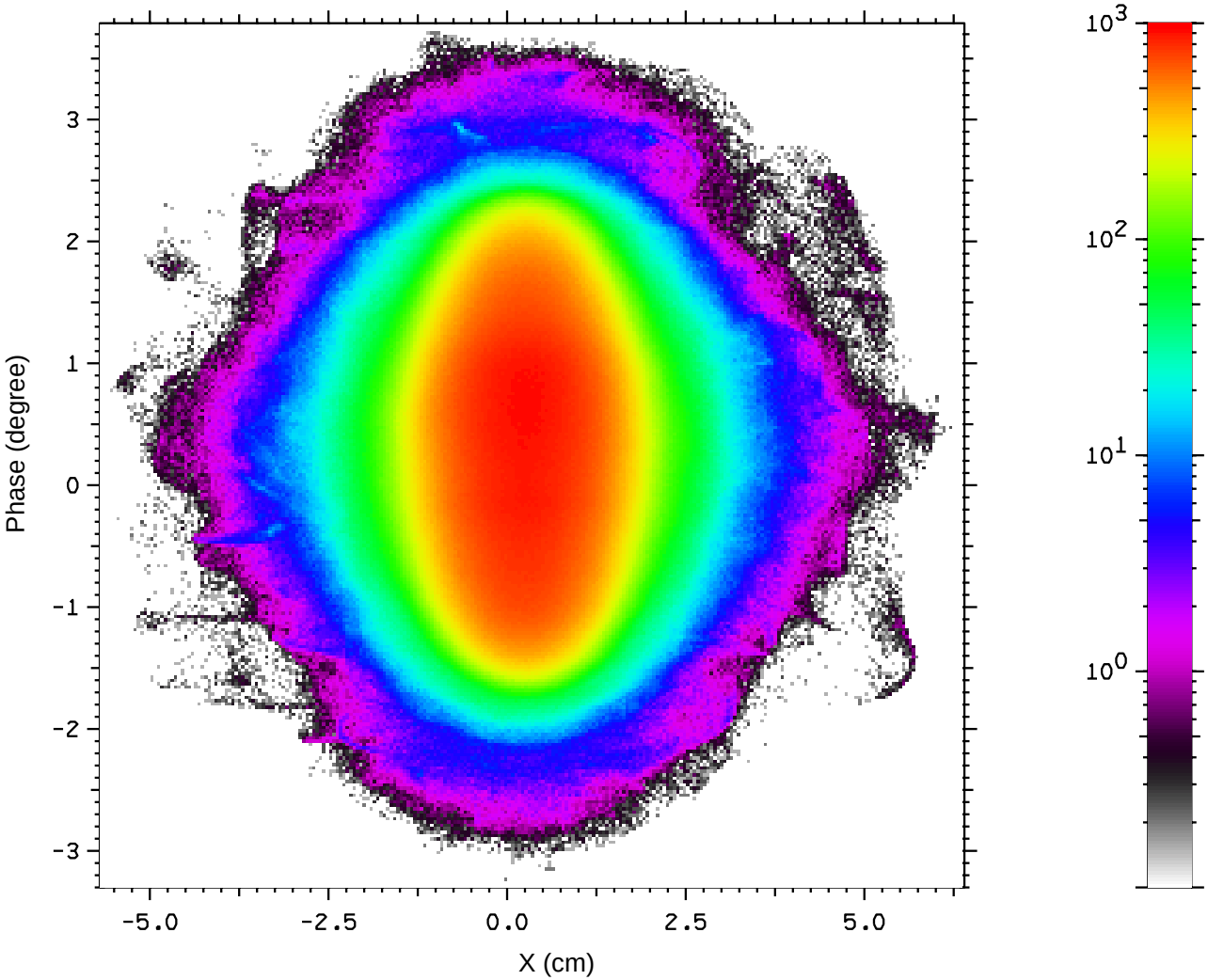
X-Y Plot at 1 GeV



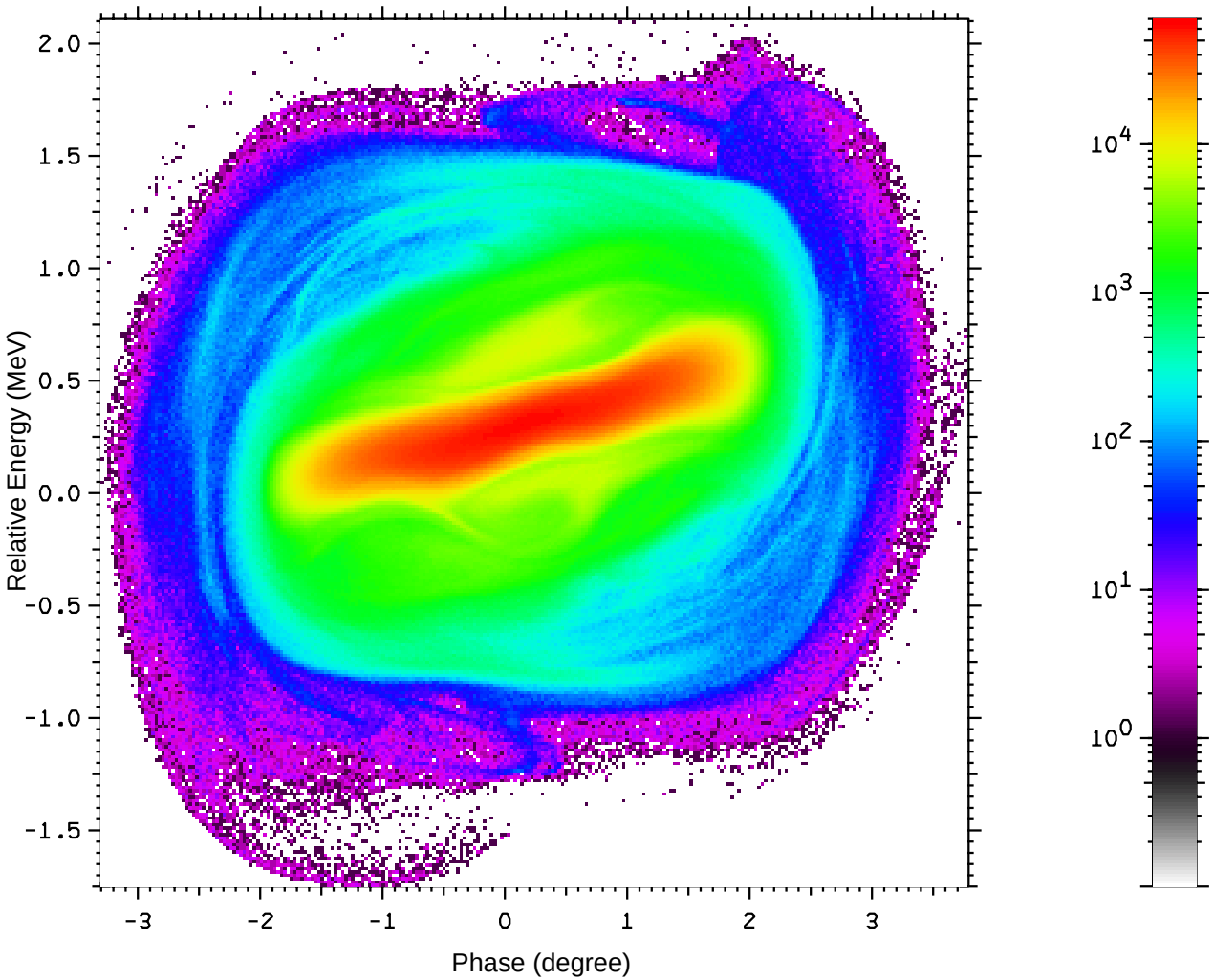
X-Px Plot at 1 GeV



X-Phase Plot at 1 GeV



Phase-Energy Plot at 1 GeV



LARGE SCALE VISUALIZATION

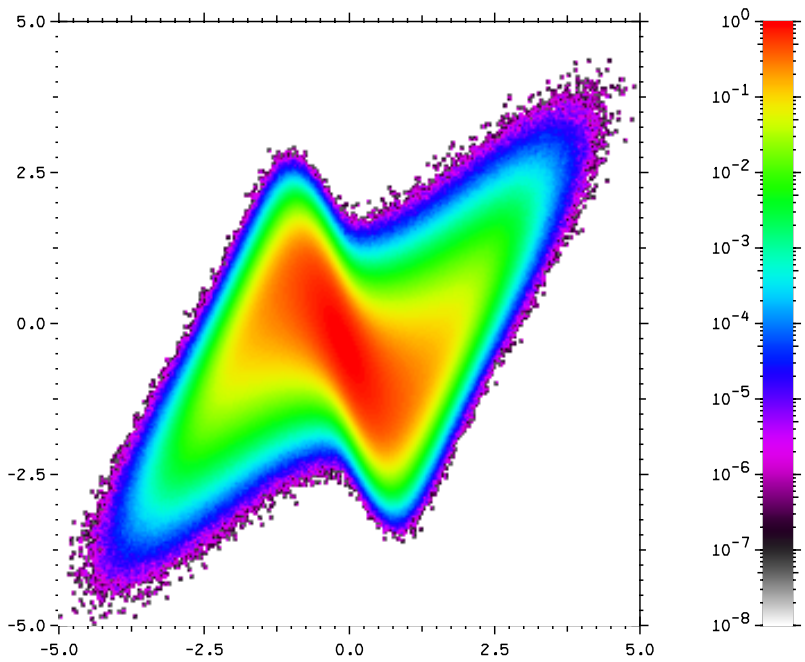
Why do we need large-scale visualization?

- **Huge** data sets
  - e.g. 0.1 - 1 billion particles, 5 - 50 GByte/timestep, TBytes per simulation
- Visualization is needed to:
  - provide physical picture of charge distribution
  - show time-evolution of the beam in phase space
  - explore very low density regions (beam halo)

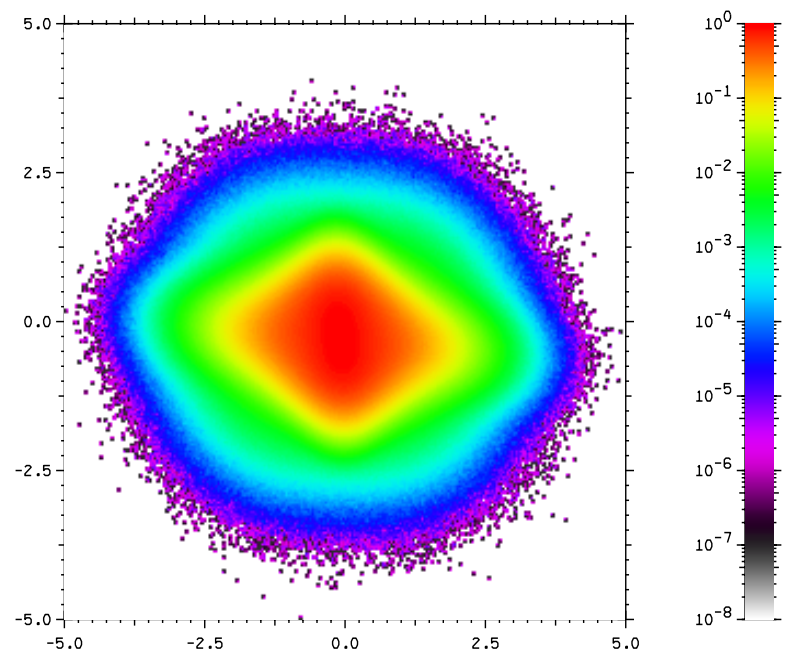
High performance computing and visualization are needed to *reduce, render* & *explore* the data, and *provide guidance* & *validation* in the design process

Plots of “Integrated Density” from a 500 Million Particle SNS Simulation

X-P_X Plot

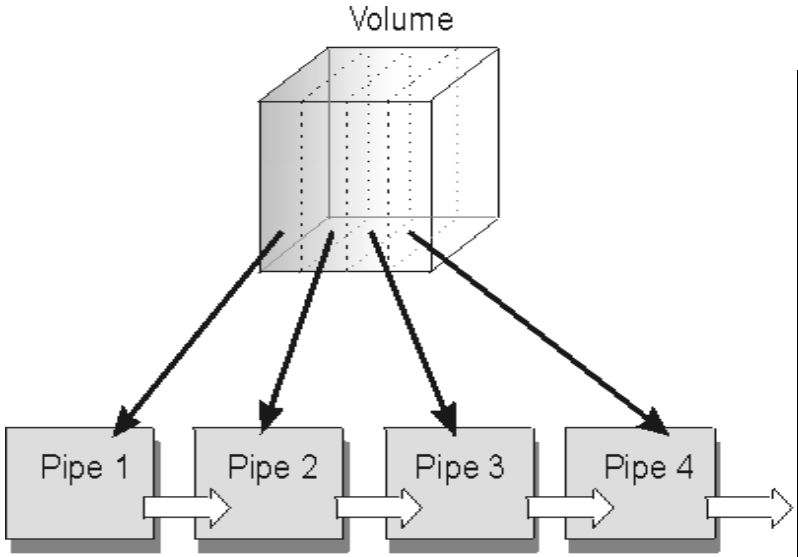


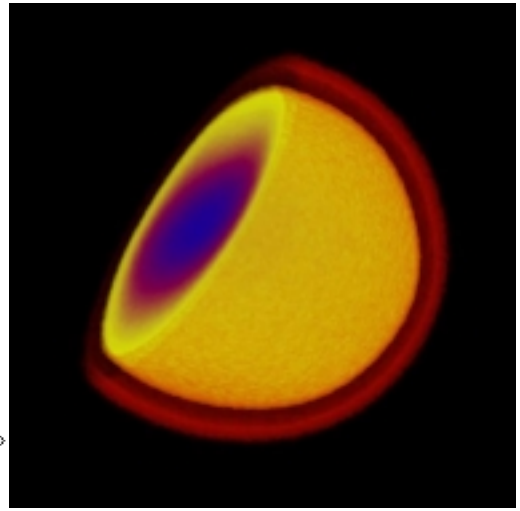
X-Y Plot



High Performance Volume Rendering

- By equally subdividing a volume and assigning each subset to a pipe we can:
  - Support larger data sets -- texture memory scales with each pipe.
  - Maintain rendering performance -- fill rate scales with each pipe.
  - Disadvantage -- pipeline architecture causes latency to scale with each pipe.

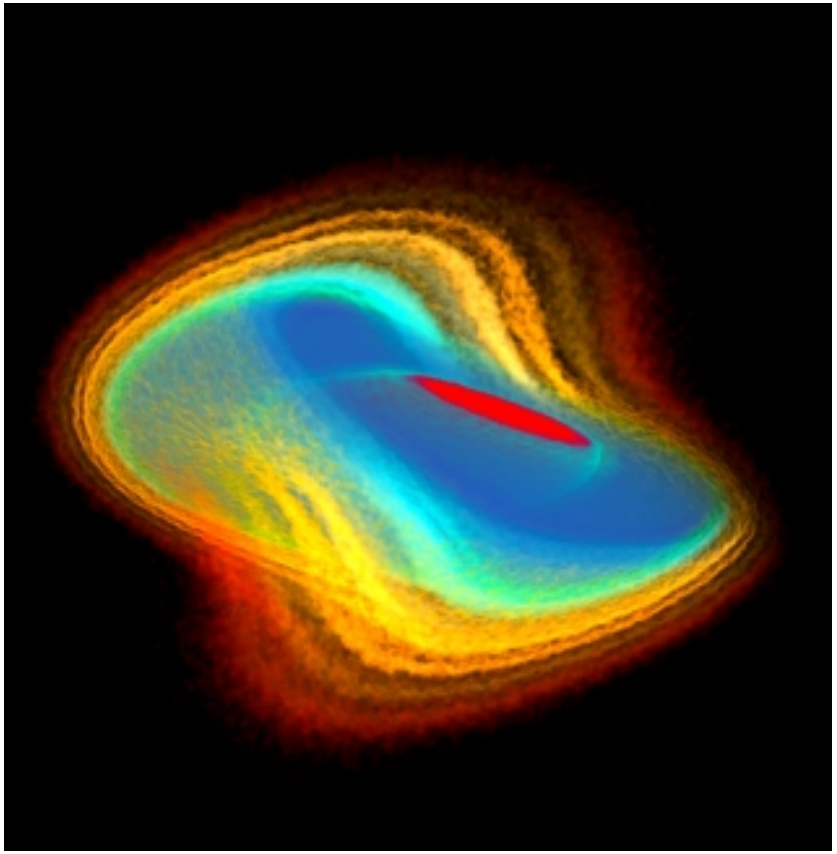




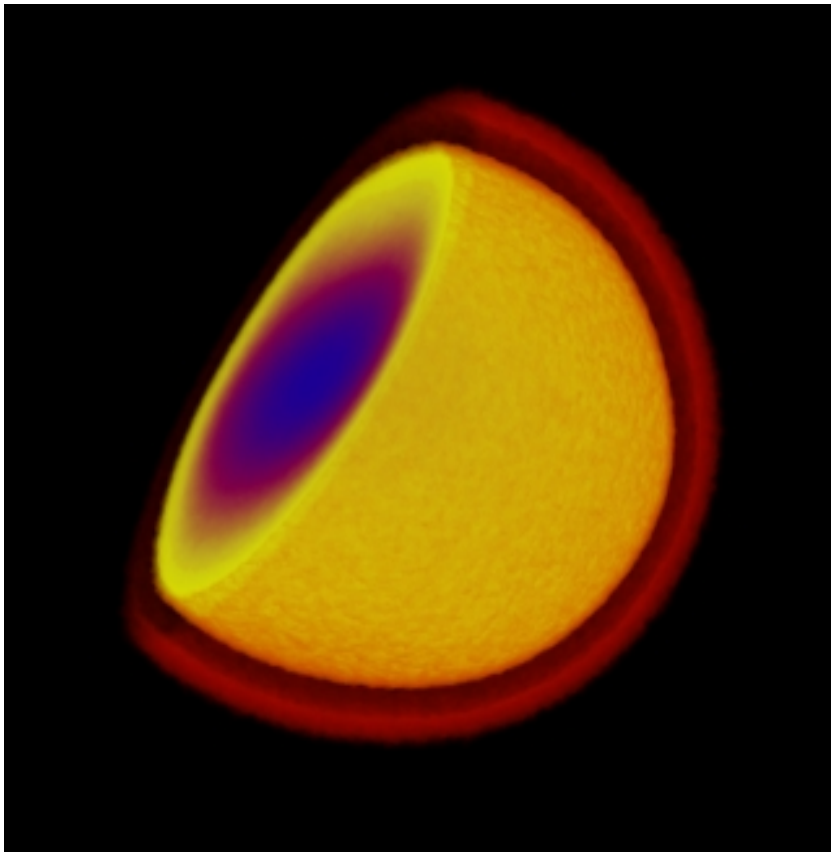
Final Image

Early work done in collaboration with SGI.

Volume Rendering of APT Simulation Results



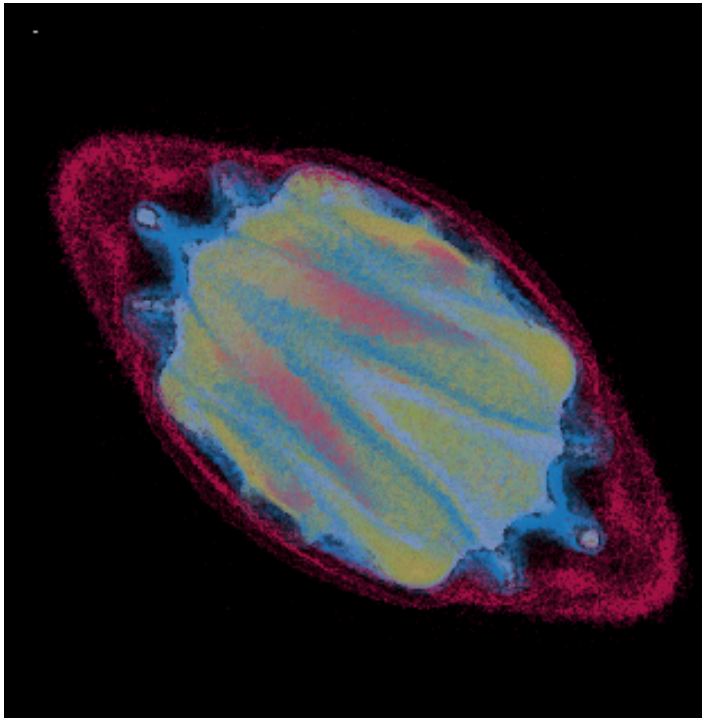
x-p_x-z space



x-y-z space

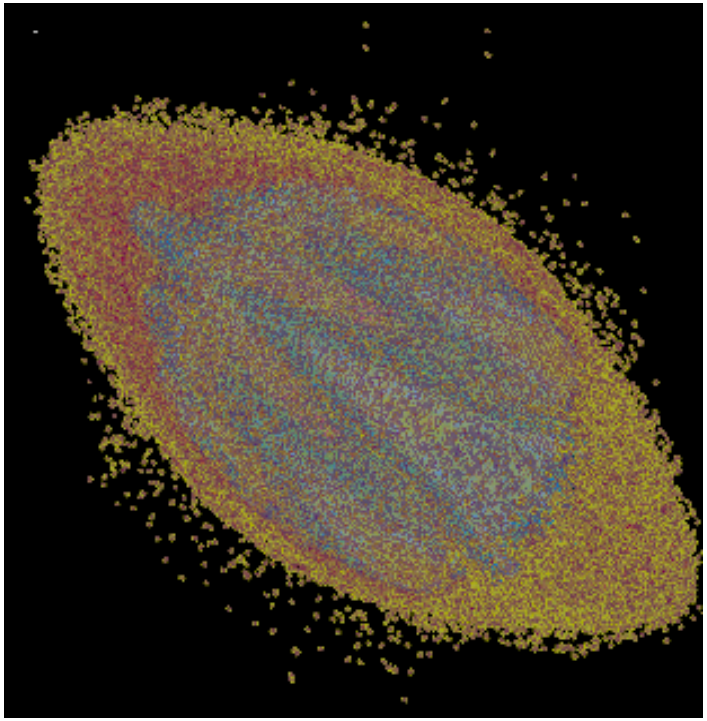
Volume Rendering of SNS Simulation Results

Transparency to see inside volume



x-p_x-z space

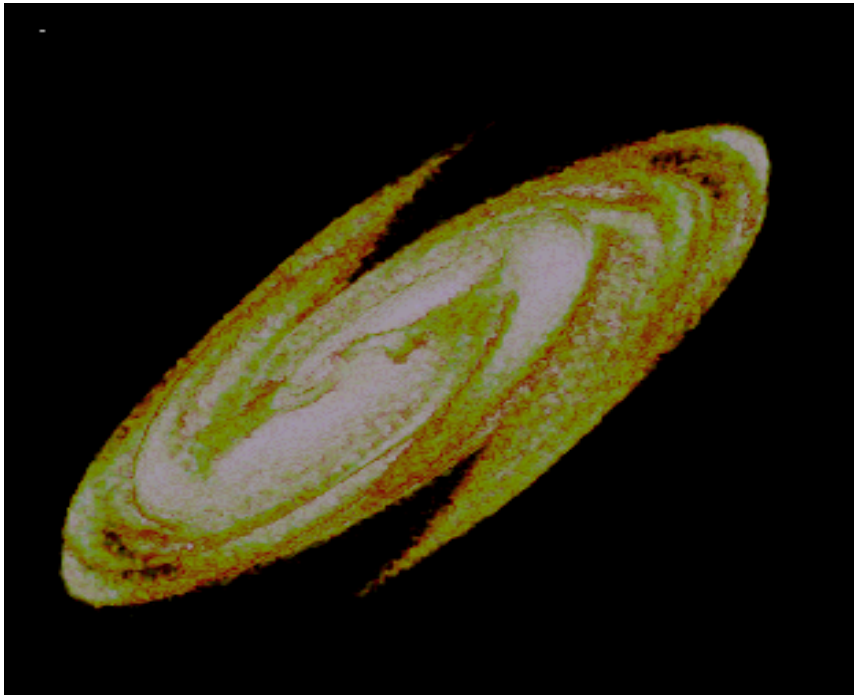
Opacity to accentuate halo



x-p_x-z space

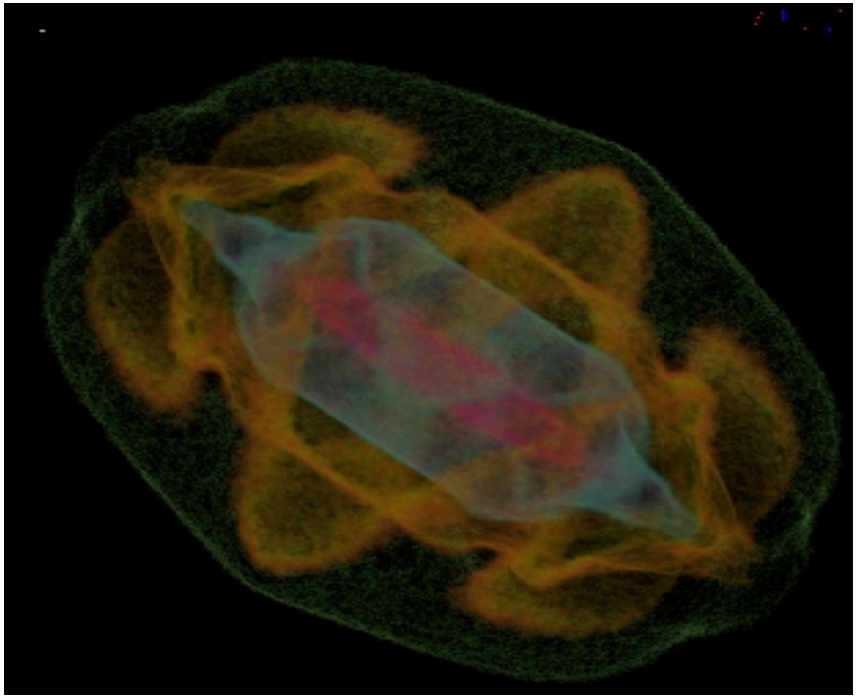
Volume Rendering of SNS Simulation Results

Oriented to exhibit x-p_x space



x-p_x-z space

Alpha & color maps set to show 4 isosurfaces



x-p_x-z space

Summary of IMPACT Capability

- Largest, most detailed linac simulations ever performed
  - Physical model used 400 distinct accelerating structures
  - Computational model used 800M particles on a 512x512x512 grid
- 3 order-of-magnitude increase in simulation capability
  - 100x larger simulations performed in 1/10 the time

Conclusions

- Object-oriented parallel particle-in-cell code:
  - improves the model accuracy
  - increases numerical resolution
  - enables exploration of new physical regions
  - drastically reduces computing time
- The code also has better maintainability, reusability and extensibility.