VORPAL for Petascale Optimization and Design (VPOD)

Mark L. Green, Ph.D.
Research Scientist
Tech-X Corporation
Boulder, Colorado

Project Overview

Project: VORPAL

Science goals: prediction of beams produced by colliding pulse injection in advanced accelerators

Participants: led by John R. Cary, Ph.D.

History: NSF funded grant “Application of Modern Computing Methods of Plasma Simulation” 2001; primarily DOE and commercial revenue funded at present

Sponsor: Tech-X Corporation

Goals: To develop a flexible framework for plasma modeling on structured grids

Science Lesson

What does the application do, and how?

VORPAL offers a unique combination of physical models to cover the entire range of plasma simulation problems.

Laser wakefield accelerators, plasma thrusters, high-power microwave guides, electromagnetic cavities and photonic devices, and plasma processing chambers are only a few of the many applications.

VORPAL algorithms include FDTD, Implicit EM, Poisson, relativistic and nonrelativistic PIC, pressureless and Euler fluids, ...

VORPAL development based in OO in C++, with well defined interfaces allowing plug and play with multiple algorithms and models

Parallel Programming Model

- MPI, OpenMP, Hybrid, Pthreads, etc.
  - MPI (OpenMP in future, considering hybrid)
- Languages, runtime libraries, build requirements, other infrastructure (python, sockets, java tools, etc)
  - TxBase, TxPhysics, Trilinos, HDF5, Python, C++
- What platforms does the application currently run on?
  - Cray, BG/L, Linux, Mac OS X, and AIX
- Current status & future plans for the programming model
  - Operator splitting between EM and particles; implicit Future: Iterative solution, dynamic load balancing, in memory checkpointing

Computational Methods

What algorithms and math libraries do you use? (PDE, FFT, etc)

Multigrid (AMG) from Trilinos

Current status and future plans for your computation

PETSc, HYPRE, Weighted Essentially Non-Oscillatory (WENO) for EM, Higher-order EM for interior and boundaries.

I/O Patterns and Strategy

- Input I/O and output I/O patterns (one file per MPI process?, pNetCDF? HDF5?, etc)
  - one file per entity (particle species or a field) for all processes (HDF5 single files)
- Approximate sizes of inputs and outputs (before, during, and after computation)
  - petascale simulation ~10MB, 1-200TB, ~300TB
- Checkpoint / Restart capabilities: what does it look like?
  - Checkpoint/restart: Full dump of state using HDF5
- Current status and future plans for I/O
  - use HDF5 compression, HDF5 custom compression

Visualization and Analysis

How do you explore the data generated?

- VisIt, VorpalView, IDL, User Preference

Do you have a visualization workflow?

- User defined - e.g., consists of a background process that waits for each dump and then puts out a specified set of plots

Current status and future plans for your viz and analysis

- VisIt -- specialized VORPAL plugins for VisIt for remote,large data sets.

Performance

What tools do you use now to explore performance

Tau, PAPI, Jumpshot, Totalview, Buffy, Integrated Performance Monitoring (IPM)

What do you believe is your current bottleneck to better performance?

hierarchical memory management, MPI

What do you believe is your current bottleneck to better scaling?

EM and PIC load imbalance

What features would you like to see in perf tools

ease of instrumentation

Current status and future plans for improving performance

Tools

How do you debug your code?

Totalview, Jumpshot, IPM, Valgrind, TxTest Regression Testing, gdb

What other tools do you use?

N/A

Current status and future plans for improved tool integration and support

no plan

Status and Scalability

How does your application scale now?

- 4K cores

Where do you want to be in a year?

- 10K cores

What are your top 5 pains? Load imbalance, MPI, hierarchical topology mapping, fault tolerance

What did you change to achieve current scalability? Haven't changed

Current status and future plans for improving scaling. listed above and to collect more sends and receives into groups to reduce effects of latency.

Roadmap

- Where will your science take you over the next 2 years?
  - into engineering design and optimization, shape optimization
- What improvements will you need to make (algorithms, I/O, etc)?
  - add new algorithms, locality optimizations, integration of post-processing and compression for I/O optimization, in memory checkpointing
- What are your plans?
  - parallel process task optimization