The UCLA Particle-in-Cell and Kinetic Simulation Software Center (PICKSC)

PI: W.B. Mori; Co-PIs: V.K. Decyk, F.S. Tsung, R.E. Callisch
International Partner: R.A. Fonseca IST and ISCTE Portugal

Some activities are also supported by DOE and the UCLA Institute for Digital Research and Education.

PICKSC Center Mission and Activities

- The mission of the Particle-in-Cell and Kinetic Simulation Software Center (PICKSC) is to support an international community of PIC and plasma kinetic software developers, users, and educators, and to increase the use of PIC software for accelerating the rate of scientific discovery.
- It aims to make available and document illustrative software programs for different computing hardware, a flexible framework for rapid construction of parallelized PIC programs, and several distinct production programs.
- It will also include activities on developing and comparing different PIC algorithms and documenting best practices for developing and using PIC programs.
- It will also develop educational software for undergraduate and graduate courses in plasma physics and computer science.
- It will also sponsor an annual workshop to help build a community of developers and users. Visitors with specific needs are welcome.

Particle-in-Cell Codes

- PIC codes integrate the trajectories of many particles interacting self-consistently via electromagnetic fields. They model plasmas at the most fundamental, microscopic level of classical physics.
- PIC codes are used in almost all areas of plasma physics, such as fusion energy research, plasma accelerators, space physics, ion propulsion, plasma processing, and many other areas.
- PIC codes are the most complete, but most computationally expensive models. Used when more simple models fail, or to verify the realm of validity of more simple models.
- What distinguishes PIC codes from molecular dynamics codes is that a grid is used as a scaffolding to calculate fields rather than direct binary interactions — reduces calculation to order N rather than N^3.

Software Engineering of Complex Code Development is a Grand Challenge: Aim to develop plasmas education, educational codes, a framework for rapid code development, and production codes

2014 PICKSC Workshop: Software Interoperability within the PIC Community

Invited major developers within the explicit PIC community:
- This community shares ideas well, but not software
- There is no large community code
- For verification, reproduce results with independently developed codes

There was agreement on:
- Desire to easily reproduce results of a paper in hours, not months
- Desire for standard benchmark problems to verify or validate new codes
- Desire for standard inputs and outputs, to enable easy comparison
- Desire for code of ethics for sharing software
- Desire for standard unit tests to illustrate new algorithms

Barriers to code interoperability:
- Different units, languages
- Internals of codes not known or documented within community

2015 Workshop is being planned

PICKSC Codes

Open Source Skeleton PIC Codes for HPC (mini-apps)

- Open-source but fully functional multi-PIE PIC codes: illustrate different parallelization strategies
  - Electronic and electromagnetic, with Darwin under construction
  - Examples using MPI, OpenMP, SIMD, vectorization, CUDA, and hierarchy of levels
  - Examples with 2-3 levels of parallelism, from beginner level to HPC expert
  - Each code in Fortran and C, with interoperability between languages
  - 2.0-4.0 Teraflops of code

- Illustration of parallel PIC codes for beginners
- Example of parallelization techniques for an irregular problem
- Non-trivial benchmarks for computer science research
- Provide a simple test environment for prototyping new ideas and algorithms
- Ideas can be mined to incorporate to other codes
- Can be expanded to a production code by adding diagnostics, initial conditions, and other boundary conditions

Excellent Performance
- Electromagnetic code, 0 levels of parallelism — 110 nsec/particle/step on 1 CPU
- Electromagnetic code, 3 levels of parallelism — 13 ps/particle/step on 96 GPUs

UPIC-EMMA

UPIC-EMMA is the Larmor boosted frame with UPIC-EMMA, a spectral EPW-PIC code.

- High-fidelity simulation results compared with up to frame simulation, with 1000 time steps obtained using boosted frame techniques

Web Interface for PIC codes

Production PIC codes can have many hundreds or even thousands of input parameters. How can this be presented in a comprehensible way? It cannot?
- Can we make the same interface usable by many PIC codes?

Approach: Properties of inputs defined in a hierarchical description XML file
- How can this be grouped in a 3 level hierarchy
- Can limit display to only those inputs that differ from default values
- Help available for each input

Initial implementation uses standard Fortran90 namelists with Apache and PIP
- Design is generic
- Multiple namelists describe highest grouping: one XML file for each namelist
- 2 level groupings within each namelist are possible
- Namelists not needed for a problem can be omitted and not displayed

Future work: different codes use different units
- Translate units for individual codes from standard units presented to user
- Public names may differ from actual names used in code
- Present appropriate or invalid inputs

QuickPIC on GitHub

QuickPIC: a 3D parallel Open-Source PIC code, developed based on the framework UPIC-1.0

QuickPIC on CUDA

OSIRIS on GPU: Doing Production Science Runs

One, Two and Three Dimensions: All Features of Osiris Supported
- Dynamic GPU Load Balancing/Tuning
- Fully MPI Ready — Run on Thousands of GPU Nodes
- Tailored Support for Fermi and Kepler generations
- Ongoing rollout with new production runs

OSIRIS-CUDA is now being used with ~100 GPUs for non-linear plasma wave and laser-plasma instability studies
- Enabled a large number of parameter scans not previously feasible
- ~23 msec/particle/GPU

Production Codes Using Advanced Architectures

ONIRIS-CUDA is now being used with ~100 GPUs for non-linear plasma wave and laser-plasma instability studies
- Enabled a large number of parameter scans not previously feasible
- ~23 msec/particle/GPU

SRS in specified laser beams: Above, large EPW activity in an above-threshold laser specie has stimulated SRS in a below-threshold specie (the smaller amplitude packet of EPWs)

OSIRIS-CUDA is enabling 2D parameter scans to study inter-species SRS physics.