Consideration of Asynchronous Algorithms for Particle-Grid Simulations

Alice Koniges,[1] Jean-Luc Vay,[1] Alex Friedman,[2]
Hartmut Kaiser,[3] and Thomas Sterling[4]

[1]Lawrence Berkeley National Laboratory, [2]Lawrence Livermore National Laboratory, [3]Louisiana State University, and [4]Indiana University

ExaMath 2013

DOE Workshop on Applied Mathematics Research for Exascale Computing, August 21-22, 2013, Washington DC

National Energy Research Scientific Computing Center

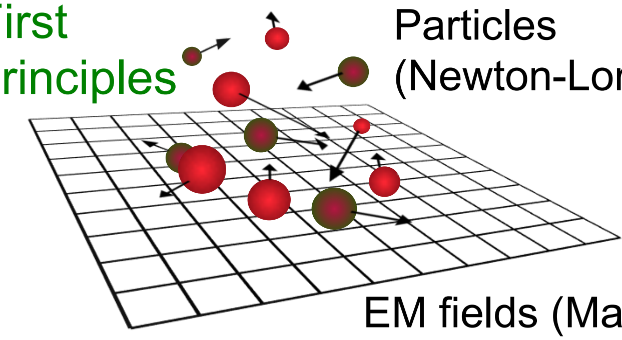
U.S. DEPARTMENT OF ENERGY | Office of Science

Particle-Grid algorithms combined with developing programming languages offer real advantages

- **A broad family of computations using discrete-particle methods already perform at extremely high scalability**
  - **Exascale will be constrained by lock-step nature**
  - **Consider new and rethought algorithms that break away from traditional lock-step programming**
    - **Compute-send;compute-send=>limited overlap**
  - **HPX runtime system implementation exposes intrinsic parallelism and latency hiding**
  - **Use a message-driven work-queue based approach to finer grain parallelism based on lightweight constraint-based synchronization**

A combination of new OS+runtime+languages with proven event-driven models can surpass performance of traditional time-step models

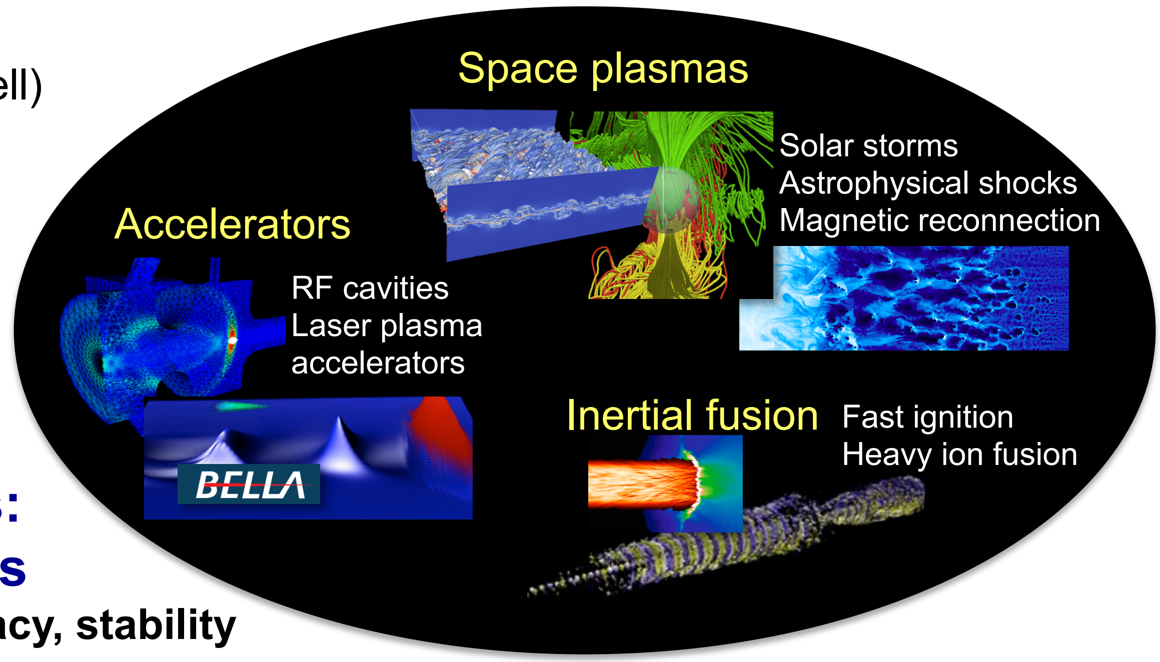
Particle-In-Cell (PIC): Method of choice for space & laboratory plasmas



– **simple, robust, scales well to 100,000s CPUs,**
– **EM-PIC applications burn millions of CPU-hours at NERSC & elsewhere.**

No approximation beyond discretization, interpolation & sampling

– **all 3D non-linear effects included**



Traditional math challenges: finite-difference field solvers

Time step, cell aspect ratio, accuracy, stability

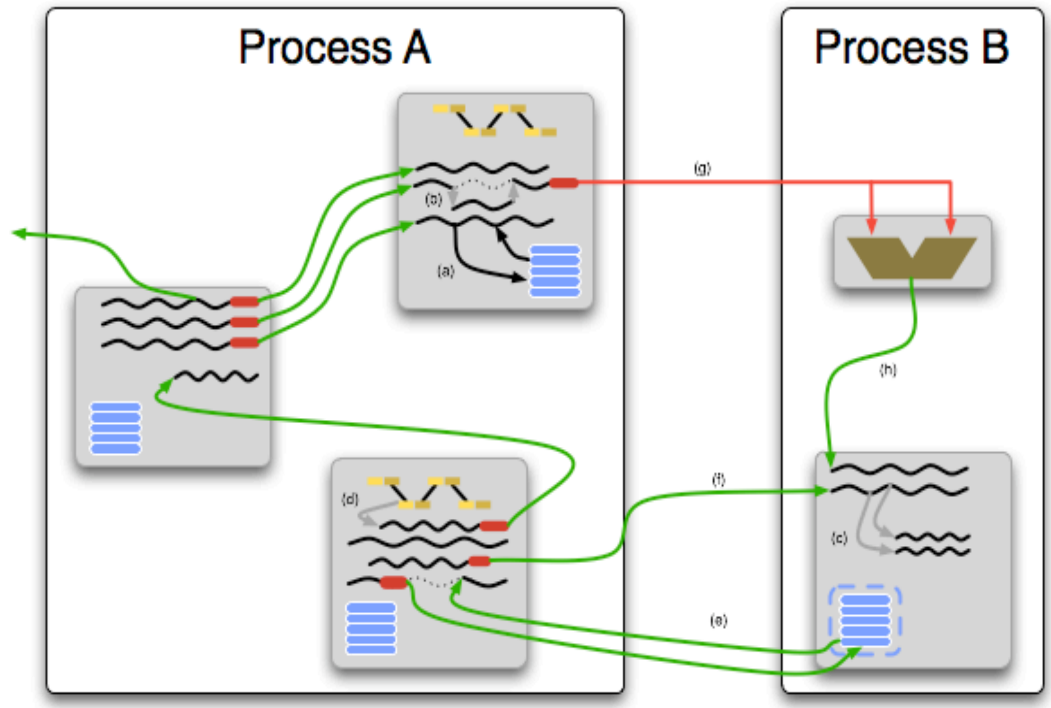
Spectral solver **offers extreme accuracy and stability, with no constraint on time step or cell shape**
- New version allows for non-global solves

PIC codes are mainstays of great petascale methods, but they must be improved for exascale

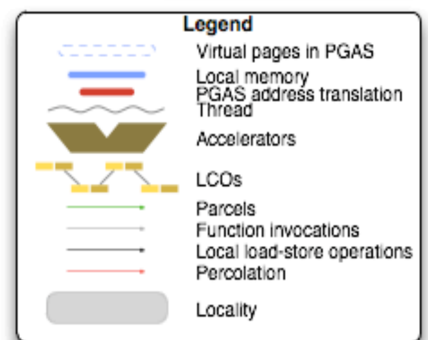
- "particle-in-cell" because plasma macro-quantities (number density, current density, etc. in cells) are computed from simulation particles
- Particles can live anywhere on the domain, but field and macro-quantities are calculated only on the mesh points
- Traditional time-driven approach follows a cycle ⇐ **Examine for Exascale**
  - Integration of the equations of motion
  - Interpolation of charge and current source terms to the field mesh
  - Computation of the fields on mesh points (field solve)
  - Interpolation of the fields from the mesh to the particle locations
- PIC codes differ from Molecular Dynamics in use of fields on a grid rather than direct binary interactions, goes from N^2 to N
- PIC codes are radically different from standard PDE solver codes and show real promise for the exascale
- PIC codes for EM don't have lock-step Poisson Solver requirements

New algorithms should work in concert with new exascale operating systems: ParalleX Execution Model

- **Lightweight multi-threading**
  - Divides work into smaller tasks
  - Increases concurrency
- **Message-driven computation**
  - Move work to data
  - Keeps work local, stops blocking
- **Constraint-based synchronization**
  - Declarative criteria for work
  - Event driven
  - Eliminates global barriers
- **Data-directed execution**
  - Merger of flow control and data structure
- **Shared name space**
  - Global address space
  - Simplifies random gathers



(a) Local data access
(b) Local thread invocation (co-routine)
(c) Local thread invocation (concurrent threads)
(d) LCO spawning a thread
(e) Remote atomic memory operation through parcels
(f) Remote thread invocation through parcels
(g) Percolation
(h) Thread creation as result of continuation action



Proof of concept: Replace a standard time-driven simulation with an event-driven simulation

For an example of PIC event-driven sim ideas and results:
H. Karimabadi, J. Driscoll, Y.A. Omelchenko, N. Omidi
Journal of Computational Physics 205 (2005) 755–775

inject new particles; then reschedule next injection

Event Queue

push the particle with the earliest exit time

calculate fields in affected shells

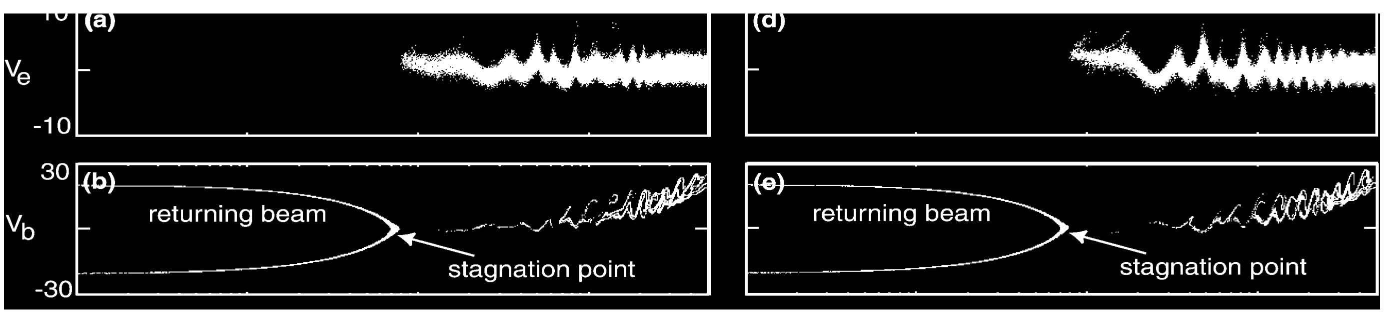
call wake-ups, if necessary

reschedule particle "push", using next exit time

Why Now?

New enabling Exascale Technology is HPX/ParalleX as part of XPRESS

Paper shows good agreement between time- and event-driven sims



New concept* enables scaling of spectral solvers
-- based on finite propagation time of electromagnetic waves

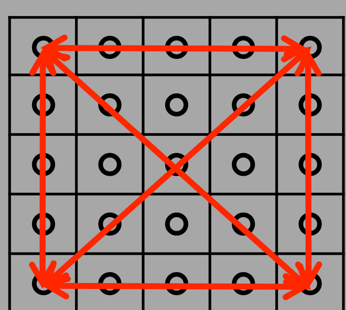
Finite speed of light ensures that

- **Errors due to truncation in guard cells stay local during a time advance**

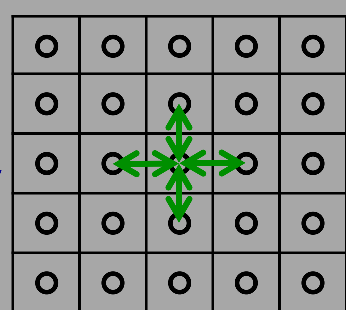
Enabling the use of local Fourier Transform

Replacing

global "costly" | **local "cheap"**



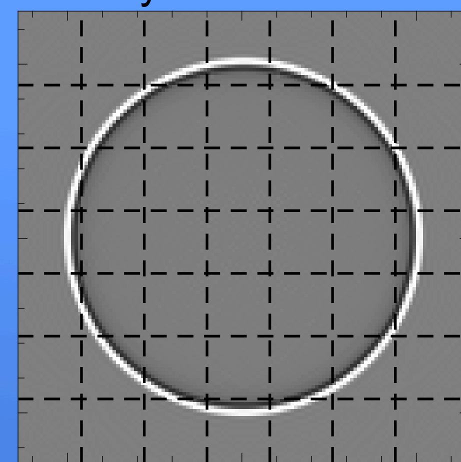
by



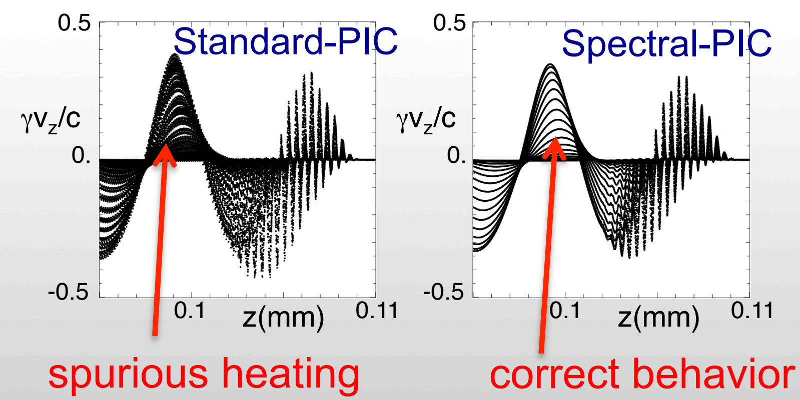
communications

Hard to scale | **Easy to scale**

Successfully tested on 7x7 domain



Prototype remedies key issue in laser plasma accelerators



*J.-L. Vay, I. Haber, B. Godfrey, J. Comput. Phys., in press

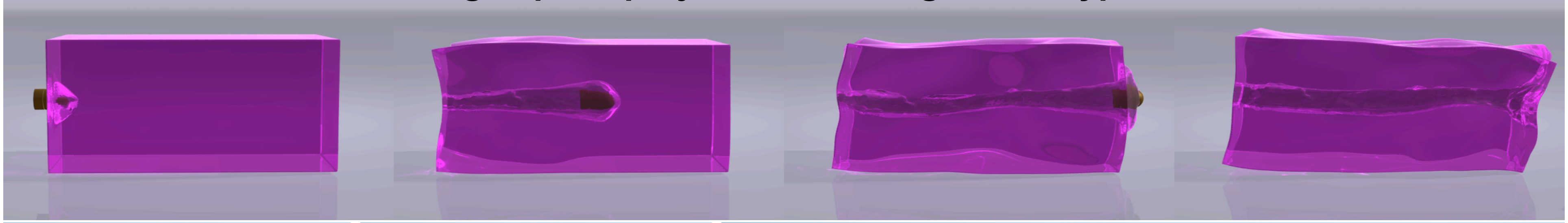
The proposed event-driven methodology has applications to a variety of areas beyond vanilla PIC

Material point methods (MPM), which use particle based advection on a background grid to discretize stress derivatives, are very relevant

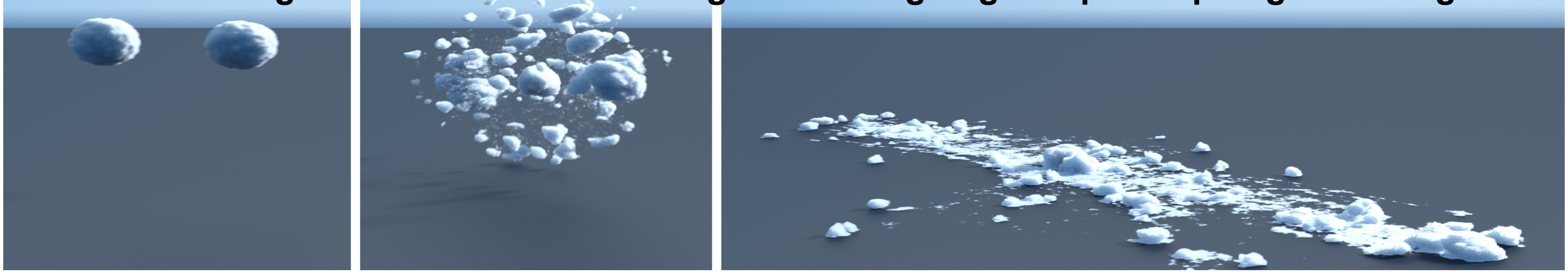
These methods are particularly important for simulating multiple phases in the presence of extreme deformation and topological change

(MPM Simulations below from J. Teran, Applied Math Dept., UCLA)

Simulation of a high speed projective colliding with a hyperelastic solid



Simulation of granular materials colliding and undergoing complex topological change



Summary: Particle-grid algorithms combined with new language/OS provide viable exascale path

- **Mathematical Challenges include:**
  - **Optimized implementations & analyses using new ideas and frameworks**
  - **Solvers for EM equations that allow for localized updates & higher accuracy, but remain compatible with the event-driven methodology**
  - **Convergence studies of PIC and other methods**
    - **e.g., PIC convergence to a Klimontovich model***
  - **Error analysis of time-driven simulations compared with event-driven**
- **Methods developed via event-based formulation are extensible to other applications including material point methods (see, e.g., Teran & Koniges; Zhang & Knoll, this workshop)**
- **Critical that new algorithms be developed in concert with dynamic adaptive runtime systems and advanced execution models appropriate for Exascale, e.g., **XPRESS**

*Su, Decyk, and Mori, in press. 2013 **https://www.xstackwiki.com/index.php/XPRESS