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

Fundamental to efficient computation at the exascale will be methods for computationally modeling complex physical systems on platforms with likely billion-way hardware and heterogeneous components. A broad family of computations that employ discrete-particle methods already enjoys good parallel performance; however, many applications are constrained by the lock-step nature (e.g., BSP-Bulk Synchronous Parallel) of traditional algorithms, and/or by inadequate accuracy (due to, e.g., numerical dispersion and anisotropy) that arises when traditional methods are applied to computationally-large domains. Scientific simulations can be formulated so as to offer accurate, stable, and efficient asynchronous parallel execution, and thus come closer to realizing the full potential of emerging exascale architectures.

Traditional simulation methods will be constrained at exascale by fundamental limitations arising from 1) **Starvation**, because parts of the computer do not get enough work; 2) **Latency**, including memory latency on local nodes; 3) **Overhead,** to manage parallelism; and 4) **Waiting,** for contention resolution (memory wall). The most common uses of MPI message passing contribute to these problems by employing a traditional lock-step programming style based on Compute-Send-Compute-Send, with limited chance for overlap. However, this is not at all required. For example, ParalleX (an asynchronous execution model) aims to replace, for some classes of applications, the Communicating Sequential Processes (CSP)/MPI model with a more efficient and scalable one, in support of innovative parallel algorithms. The HPX (High Performance ParalleX) [1,2] runtime system implementation addresses performance issues by exposing intrinsic parallelism, employing intrinsic system-wide latency hiding and dynamic adaptive resource management, and introducing a global name space. In a related effort, the XPRESS project (as part of the XSTACK project suite) is using an active global address space (AGAS), a message driven, work-queue based approach to finer grain parallelism based on lightweight, constraint based synchronization, and adaptive locality control at runtime, to allow the implementation of more flexible models.

New algorithms can be developed that will benefit from the new programming paradigm offered by use of the HPX runtime and similar new execution models that support AGAS, finer grain parallelism, and asynchronous execution. Progress can be made by relaxing (ultimately bypassing) lock-step time-advancement for classes of problems in which causal relationships are such that an asynchronous advance is both natural and efficient. A major focus is on Vlasov-Maxwell particle-in-cell (PIC) codes [3,4]; however the methods will apply to a variety of problems amenable to asynchronous (event driven or otherwise) simulation, as well as to discrete-particle,

Contact: Alice Koniges, Lawrence Berkeley National Laboratory, aekoniges@lbl.gov

grid-based hyperbolic, and coupled particle-grid problems. Some methodological and programming elements needed for PIC can be extrapolated from recent N-body particle simulations that demonstrate the efficacy of HPX parallel programming [5].

The predominant paradigm for self-consistent modeling of particle beams, plasmas, and important classes of gravitating systems is the PIC formulation. The use of a computational grid removes the N-squared scaling while simultaneously affording an effective interaction kernel that smoothes short-range interactions, thus reducing spurious granularity effects. These methods provide a natural mapping of physics interactions onto a computational resource and are used with great success in a wide range of fields. The standard formulation uses finite differences in both space and time of Maxwell's (or other relevant) field equations, and for certain classes of problems scales very well to 100K processors or more. PIC modeling and related methods have been extended to space-weather, biological systems, and general fluid and solid mechanics [see e.g., 6]. However, use of lock-step time advancement leads to challenging load balancing in many cases of interest, and more importantly can be inefficient when a domain-global small step size is used; by breaking the lock-step constraint, small steps can be used only as needed in isolated subdomains. Optimal adaptations require new algorithms, not just programming models. Updated classic PIC algorithms using programming models that overlap computation and communication and/or utilize one-sided communication demonstrate improvement at very large concurrency [7,8], but these are still limited by the confines of the algorithm and do not take advantage of the inherently asynchronous physics.

Exploitation of the finite speed of information propagation in physical systems enables preservation of causality and enabling highly parallel computations, while replacing all-to-all communications by local data exchanges. This readily applies to finite-difference time-domain calculations of hyperbolic and parabolic systems, using domain decomposition, but in fact is more general. Recently, a method for performing a domain-decomposed pseudo-spectral solve in the field calculation step has been shown to eliminate the need for all-to-all communication (e.g., global FFT) at each time step [4]. This method promises the greatest benefit to computationally-large problems that require the high accuracy and minimal dispersion and anisotropy afforded by the pseudo-spectral solve. Experience has already been attained in fielding PIC calculations with adaptive time-stepping [9-12], but these methods need to be merged with computation techniques suited to the emerging computer architectures and future dynamic runtime system software to deliver superior efficiency and dramatic improvements in scalability.

It will be important to support a "futurization" approach, wherein calculations proceed more or less independently, driven primarily by causality and accuracy considerations; in the PIC case, the field that provides the particle push should be updated only as needed. Such futurization requires that, within the programming model, various threads can perform alternative work while waiting for a result "from the future," thus enabling the computation to proceed at its own pace everywhere in the computational domain.

[1] H. Kaiser, M. Brodowicz and T. Sterling, “ParalleX: An Advanced Parallel Execution Model for Scaling-Impaired Applications,” International Conference on Parallel Processing Workshops, Los Alamos, California, 394-401 (2009)

[2] HPX Github repository, https://github.com/STEllAR-GROUP/hpx/

[3] C. Birdsall, A. Langdon, Plasma physics via computer simulation, Adam-Hilger, 1991.

[4] J.-L. Vay, I. Haber, B. B. Godfrey, "A domain decomposition method for pseudo-spectral electromagnetic simulations of plasmas," J. Comp. Phys. **243**, 260 (2013)

[5] C. Dekate, M. Anderson, M. Brodowicz, H. Kaiser, B. Adelstein-Lelbach and T. Sterling, “Improving the Scalability of Parallel N-body Applications with an Event-driven Constraint-based Execution Model,” International Journal of High Performance Computing Applications **26**, 319-332 (2012)

[6] J. Dallon, “Numerical Aspects of Discrete and Continuum Hybrid Models in Cell Biology” Applied Numerical Mathematics **32**, 137–159 (2000)

[7] A. Koniges, R. Preissl, J. Kim, D. Eder, A. Fisher, N. Masters, V. Mlaker, S. Ethier, W. Wang, M. Head-Gordon, and N. Wichmann, “Application acceleration on current and future Cray platforms,” Proc. Cray User Group Meeting, Edinburgh, Scotland (2010)

[8] R. Preissl, N. Wichmann, B. Long, J. Shalf, S. Ethier, and A. Koniges, “Multithreaded global address space communication techniques for gyrokinetic fusion applications on ultra-scale platforms”, Proceedings of SC11, 78 (2011)

[9] A. Friedman, S. Parker, S. Ray, and C. Birdsall, “Multiscale particle-in-cell plasma simulation,” J. Comp. Phys. **96**, 54 (1991)

[10] S. E. Parker, A. Friedman, S. L. Ray, and C. K. Birdsall, “Bounded Multi-Scale Plasma Simulation: Application to Sheath Problems,” J. Comp. Phys. **107**, 388 (1993)

[11] H. Karimabadi, J. Driscoll, Y. A. Omelchenko, and N. Omidi, “A new asynchronous methodology for modeling of physical systems: breaking the curse of Courant condition,” J. Comp. Phys. **205**, 755 (2005)

[12] Y. A. Omelchenko and H. Karimabadi, “Event-driven, hybrid particle-in-cell simulation: A new paradigm for multi-scale plasma modeling,” J. Comp. Phys. **216**, 153 (2006)