

# COMPUTATIONAL CHALLENGES IN HIGH INTENSITY ION BEAM PHYSICS

ROBERT D. RYNE<sup>1</sup>, JI QIANG<sup>1</sup>, AND SALMAN HABIB<sup>2</sup>

<sup>1</sup>*Los Alamos Neutron Science Center, LANSCE-1*

<sup>2</sup>*Theoretical Division, T-8*

*Los Alamos National Laboratory, Los Alamos, NM 87545*

Experiments based on accelerator physics and technology have led to some of the most remarkable discoveries of the 20<sup>th</sup> century in fields such as high energy physics, nuclear physics, materials science, and biological science. Beyond facilitating fundamental discoveries, accelerators and the technology associated with them have also made substantial contributions to applied science and technology. Particle accelerators have had, and will continue to have, a profound impact on basic and applied scientific research. The design of the next generation of particle accelerators will require a new level of simulation capability as researchers push the frontiers of beam intensity, beam energy, and system complexity. For virtually all proposed major accelerator projects, modeling on parallel supercomputers is essential for design decisions aimed at evaluating feasibility, evaluating and reducing risk, reducing cost, and optimizing performance. As a case study we focus on modeling beam halo in high intensity ion rf accelerators. We describe how, through a DOE Grand Challenge in computational accelerator physics, simulation capability has increased 1000-fold in terms of problem size and speed of execution compared with what was possible a decade ago.

## 1 Introduction

A recent spate of articles in Science magazine [1,2,3] testifies to the importance of particle accelerator facilities in enabling scientific discovery. The headlines state,

*“Starting this fall, a machine called RHIC will collide gold nuclei with such force that they will melt into their primordial building blocks”*

*“Biologists and other researchers are lining up at synchrotrons to probe materials and molecules with hard x-rays”*

*“A new generation of accelerators capable of generating beams of exotic radioactive nuclei aims to simulate the element-building process in stars and shed light on nuclear structure”*

In another example, a synchrotron light source (the Advanced Light Source at Lawrence Berkeley National Laboratory) played a key role in one of Science magazine’s top 10 breakthroughs of the year by enabling the production of detailed images and information concerning the structure of the ribosome and its subunits [4,5].

Particle accelerators and their associated detectors are among the largest, most complex scientific instruments in the world. Experiments based on accelerator science and technology have made remarkable discoveries about the basic nature of matter including nuclear structure, quark dynamics, the nature of elementary particles, and unified field theories. Experiments with synchrotron light sources and spallation neutron sources have been crucial to advances in the biological and materials sciences. Beyond impacts to basic and applied science, accelerators-based systems have also been proposed that address environmental, energy-related, and national security issues through projects such as the Accelerator Transmutation of Waste (ATW), accelerator-driven fission energy production, the Accelerator production of Tritium (APT), and proton radiography for hydrodynamic imaging.

The successful development of large accelerator facilities involves enormous investments in the three paradigms of scientific research: theory, experiment, and simulation. Neglecting any of these can lead to an inability to meet performance requirements, cost overruns, and ultimately, project failure. Consider, for example, the case of the superconducting supercollider (SSC). In the early 1990's a lack of confidence in the design of the SSC, due, in part, to an inability to perform the required tracking calculations to predict the dynamic aperture, led to a decision to increase the beam pipe aperture by 1 cm. This resulted in an estimated \$1 billion cost increase due to its impact on the SSC's thousands of superconducting magnets. Less than ten year later, parallel tracking codes have now been developed that, if available at the time of the SSC, might have proven that the beam pipe aperture was large enough. At the very least, today's terascale computers, running codes targeted to parallel computing platforms, would have enabled the required tracking calculations, and the decision would have been based on well-trusted numerical predictions rather than uncertainty. Such tracking calculations involve simulating particle dynamics for hundreds of millions of revolutions in a circular collider, where each revolution itself involves several thousand beamline elements. Simulations of this type are now going on in earnest at CERN in support of the Large Hadron Collider project.

The design and construction of state-of-the-art particle accelerator facilities is an impressive accomplishment. But, as complex as they are, the next generation of accelerators will involve even greater complexity and will require unprecedented precision in accelerator design and beam control. Examples include the Next Linear Collider (NLC), in which beams will be manipulated on a scale ranging from millimeters in the main linac to tens of nanometers at the collision point; a muon collider, in which a 100 MW-class proton beam will be used to produce short-lived muons that must be cooled by six orders of magnitude and brought into collision in a matter of microseconds; and a 4<sup>th</sup> generation light source, requiring nanometer-scale smoothness in the beam pipe to successfully control an electron beam and produce intense, ultra-short pulses of x-rays for imaging ultra-fast phenomena. For all of these accelerator systems, terascale simulation will play a key role by facilitating

design decisions, increasing safety and reliability, optimizing performance, and ultimately, helping to ensure project completion within budget and on schedule.

While particle accelerators are evolving and becoming more complex, so too are the computational models and the computers themselves that are used to simulate them. Early cutting-edge simulations of accelerators were performed in the 1970's using CDC 7600 computers. Performance increased dramatically in the 1980's through the use of Cray computers with vector pipelines, but this required careful attention on the part of the scientific programmer. The introduction of massively parallel systems such as the Thinking Machines CM-5 in the early 1990's required computational scientists steeped in vector supercomputers to change their way of doing business. A shift to clusters of shared memory multi-processors in the mid-to-late 1990's is requiring a similar change. A glimpse of the future is provided by the recent announcement of IBM [6] that it is beginning a 5 year, \$100M research initiative to build "Blue Gene," a petaflops supercomputer for protein folding calculations, that will consist of 1 million processors. The processors and the new software that will be needed are still under design, but it is anticipated that the processors will have no cache memory, and that multi-threading techniques will be used to perform several calculations simultaneously on each node.

The complexity of future supercomputer systems, along with the continual change in supercomputing architectures, leads to the following conclusion: in order that large parallel applications have high performance and a long life-span, a multi-disciplinary approach is required that involves not only physical scientists, but also computer scientists, software engineers, applied mathematicians, and numerical analysts. This approach has been used successfully by several projects, including a DOE Grand Challenge in computational accelerator physics to be described below.

The remainder of this paper is organized as follows: In section 2 we introduce the beam halo issue for future high intensity accelerators. This leads to the need to simulate charged particle beams subject to both complex externally applied fields as well as strong self-fields (i.e. space-charge fields). In section 3, we describe the mathematical formulation and the computation methods used to model such systems. In section 4 we describe a new parallel beam dynamics code, called IMPACT, developed through the Accelerator Grand Challenge and the Los Alamos Accelerator Code Group. Lastly, in section 5 we provide a brief example of the use of high performance visualization to analyze the huge data sets produced in large-scale parallel simulations.

## **2 Modeling Beam Halo in High Intensity RF Accelerators**

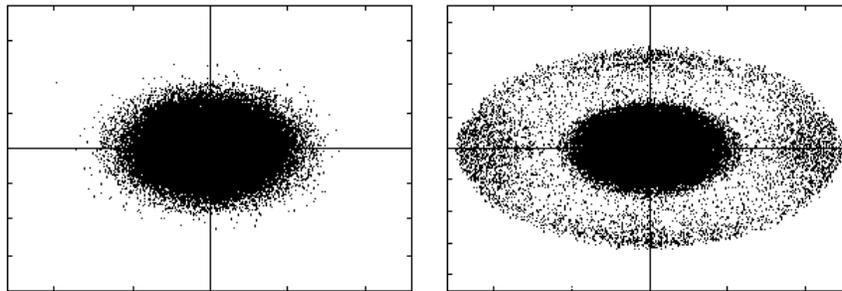
The study of halos in beams for new high intensity accelerator applications has received much attention in recent years [7-15]. Example projects for which beam halo is a key issue include the Spallation Neutron Source (SNS), the Accelerator Production of Tritium (APT), the Accelerator Transmutation of Waste (ATW), and

the driver for a muon-based neutrino source or collider. For all of these projects, controlling beam loss is essential to prevent excessive radioactivation that could hinder or prevent hands-on maintenance and adversely affect safety, availability, and reliability of the accelerator.

Figure 1 shows the simulated transverse (x-y) cross-section of an intense beam propagating in a quadrupole channel. On the left-hand side, the beam is rms-matched into the quadrupole channel, and almost no large amplitude particles are observed. On the right-hand side, the beam is rms-mismatched into the channel, and a large amplitude halo is present. Such halos lead to beam loss when particles intercept the beam pipe. Understanding the physics of beam halo formation and being able to predict beam halo with high accuracy represent a major challenge in the successful design and operation of future high intensity accelerators.

The method most widely used to simulate the dynamics of plasmas and beams is the Particle-In-Cell (PIC) approach [16,17]. While serial simulations on PC's or workstations are useful for rapid design and for predicting rms beam behavior, limitations on memory and performance make them inadequate for accurately modeling beam halo. The parallel PIC method, which was developed largely by the plasma physics community and, to a lesser extent, by the astrophysics community, has made it possible to perform large-scale PIC simulations on multi-processor platforms [18-25]. The parallel PIC approach provides a means to reduce fluctuations and increase spatial resolution by enabling the use of more macroparticles along with an increased grid density. It also dramatically reduces the computation time. The successful development of parallel PIC simulations for modeling high intensity beams is now well established [26-29]. Large-scale simulations with 100 million macroparticles are now done frequently, and simulations with up to 1 billion particles have been performed.

The next section provides an overview of single- and multi-particle simulation techniques, and the split-operator methods used to join them, in order to combine both capabilities in a single, parallel PIC model for simulating intense beams.



**Figure 1.** Cross-section of a beam properly matched into a quadrupole channel (left-hand side), and mismatched into the channel (right-hand side), resulting in a pronounced beam halo.

### 3 Mathematical Formulation and Computational Methods

#### 3.1 Single Particle Dynamics

As a prelude to discussing multiparticle simulation, we provide a brief overview of the methods used to treat single particle dynamics in accelerators. In the absence of radiation, the dynamics is governed by a Hamiltonian  $H(q,p,t)$ , where  $(q,p)$  denotes canonical coordinates and momenta, respectively, and where  $t$  denotes the independent variable. The Hamiltonian involves the electromagnetic potentials  $(A, \varphi)$  for the beamline element under consideration. For example, in the case of a magnetic quadrupole (with its axis oriented along the  $z$ -axis of a Cartesian coordinate system) one would specify the quadrupole gradient,  $g(z)$ , and some number of its derivatives ( $g', g''$ , etc.) with respect to  $z$ , depending on the order of the calculation. For an rf accelerating gap, this would involve specifying the electric field on-axis at fixed time,  $E(z)$ , and its derivatives. For a parallel faced bending magnet with midplane symmetry, one would specify the midplane magnetic field,  $B_0(x,z)=B(x,y=0,z)$ , and its derivatives with respect to  $x$  and  $z$ . In general, the high order treatment of beamline elements requires higher order derivatives of the functions used to specify the potentials.

The next step in the calculation of single particle dynamics is usually the specification of a Hamiltonian with a coordinate as the independent variable. The appropriate choice is the coordinate system that is the most natural to describe the element itself. For example, it is the axial coordinate (normally the  $z$ -axis) in rectilinear elements such as quadrupoles, rf gaps, high order multipoles (sextupoles, octupoles, etc.) and parallel faced bending magnets. In the case of a sector bend it is more natural to use polar coordinates and treat the angle  $\theta$  as the independent variable. It is possible to express the Hamiltonian with  $z$  or  $\theta$  as the independent variable so long as a particle does not “change direction” inside of an element. (As a side note, such an approach may therefore be inappropriate for modeling high power klystrons, since particles sometimes do change direction in the output cavity.) Expressed with a coordinate as the independent variable, the new Hamiltonian no longer contains the dependent variables  $(z, p_z)$ . Instead it involves  $(t, p_t)$ , which correspond to arrival time at the location  $z$  or  $\theta$  and the (negative) particle energy at that location, respectively.

Generally one is interested in motion near some reference trajectory. Thus, one expands the Hamiltonian in a power series around the reference trajectory and obtains a new Hamiltonian for the deviation variables. This new Hamiltonian may contain thousands of terms depending on the order of the calculation. For example, in six variables, a third order calculation involves 209 terms, and a fifth order calculation involves 923 terms. If one is interested in higher order nonlinear effects, or if one is interested in high order representations of maps in order to combine maps without significant loss in accuracy, the number of coefficients increases rapidly with order. For example, calculations of order 7, 9, and 11 involve 3002,

8007, and 18563 terms, respectively. It is worth noting that these denote the theoretical minimum number of terms needed to treat the dynamics at a specified order. Of course, if one includes other degrees of freedom (such as spin), the number of required parameters increases.

The field of magnetic optics, which addresses linear and nonlinear single-particle dynamics in accelerators, represents one of the great successes of modern accelerator theory. (See, for example, [30-36]). It is now possible to design beam optics systems to very high order, and in fact computations can be performed, in principal, to arbitrary order in deviations from the reference trajectory.

The fact that accelerators consist of localized beamline elements, and that, globally, there is a complicated variation in the potentials with respect to  $z$  and  $\theta$ , has led to the description of circular accelerators as “hopelessly complicated” dynamical systems [33]. Accelerator physicists utilize an approach in which individual elements are represented by *transfer maps*. These maps may be combined and manipulated for a variety of purposes. In circular machines, the maps can be combined to produce a one-turn map to be used for global analysis [34].

If we let the six-vector  $\zeta=(x,p_x,y,p_y,t,p_t)$  denote the canonical coordinates and momenta, then Hamilton’s equations can be written

$$\frac{d\zeta}{dt} = - :H : \zeta ,$$

where  $:H:$  is a Lie operator defined according to  $:H:\zeta = [H,\zeta]$ , and where  $[,]$  denotes the Poisson bracket. In the language of mappings, we say that there is a map,  $M$ , corresponding to the Hamiltonian,  $H$ , that maps initial phase space variables,  $\zeta^i$  into final phase space variables,  $\zeta^f$ , and we write

$$\zeta^f = M\zeta^i .$$

If  $H$  does not depend on the independent variable, then the map  $M$  is given by

$$M = e^{-t:H} .$$

From this simple argument, it is clear why Lie transformations (i.e. the exponentials of Lie operators) play such a central role in the mappings of accelerator theory and magnetic optics. Real systems, of course, have an extremely complicated dependence on the independent variable. Powerful techniques have been developed since the 1980’s, led by A. Dragt, to design, simulate, and analyze complex beamlines using Lie methods [30-36].

### 3.2 Multiparticle Dynamics

The previous section dealt with the dynamics of single particles subject to Hamilton's equations, and we introduced the concept of a map,  $M$ , associated with the dynamics. The corresponding equation governing the beam distribution function,  $f(\zeta, t)$ , follows from Liouville's theorem,

$$\frac{df}{dt} = \frac{\partial f}{\partial t} - [H, f] = 0,$$

from which we obtain the Liouville equation,

$$\frac{\partial f}{\partial t} = [H, f].$$

It is straightforward to show that the evolution of the distribution function can also be described in terms of  $M$ . Namely, a distribution function  $f(\zeta, t)$  whose initial value is  $f^0(\zeta) = f(\zeta, t=0)$ , evolves according to

$$f(\zeta, t) = f^0(M^{-1}\zeta).$$

Now suppose one is interested in modeling *intense* charged particle beams. In this case, one needs to determine the electromagnetic potentials associated with the beam's own self fields. If the fine-grained nature of the fields associated with discrete particles can be neglected, then the collective fields can be treated by a self-consistent average or mean field. Though particle motion in accelerators is usually relativistic, it is often the case that the motion of a bunch of particles relative to one another (i.e. the motion in the bunch frame) is nonrelativistic. In such cases, the effect of the electric and magnetic self-fields both follow from the scalar potential, which is simply related to the beam density,  $\rho$ , by Poisson's equation,

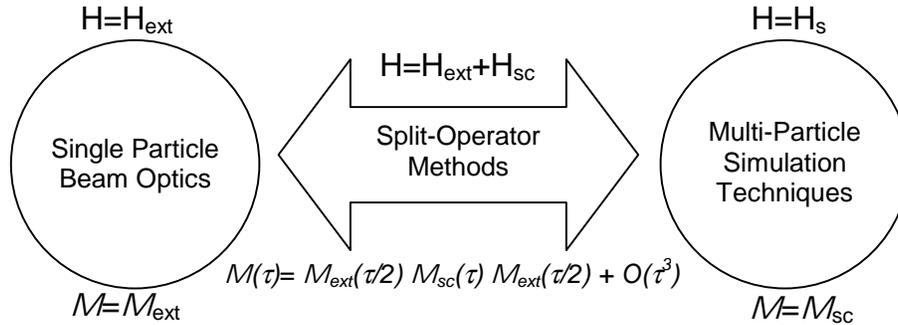
$$\nabla^2 \phi = -\rho / \epsilon_0.$$

Together, this equation and the equation governing the evolution of the distribution function constitute the Vlasov/Poisson equations. They describe the evolution of an intense charged particle beam in terms of a beam distribution function, under the assumptions that discreteness effects, collisional effects and radiation can be neglected.

### 3.3 Split-Operator Methods

We have now set the stage for answering the question, how can single-particle dynamics and multi-particle dynamics be treated within the same formalism?

In many cases, the Hamiltonian governing single particle dynamics in an intense beam can be written as  $H=H_{\text{ext}}+H_{\text{sc}}$ , where  $H_{\text{ext}}$  denotes the Hamiltonian in the presence of externally applied fields alone, and  $H_{\text{sc}}$  describes the added effect of the space-charge fields. Such a form is ideally suited to the application of split-operator methods (see figure 2). The left-hand side of the figure denotes the field of (single particle) magnetic optics, and the right hand side denotes multi-particle simulation.



**Figure 2.** Split-operator methods make it possible to combine methods from the field of *magnetic optics*, involving the dynamics of individual particles subject to complicated external forces, with methods of *multi-particle simulation*, involving the dynamics of strongly interacting particles.

Split-operator methods provide the means to draw on the success of both magnetic optics and multi-particle simulation techniques to simultaneously treat high-order optical effects and space-charge effects. Consider a Hamiltonian that can be written as a sum of two parts,  $H=H_1+H_2$ , where the dynamics for each part, separately, can be found exactly or to some desired accuracy. In other words, suppose we can compute the mapping  $M_1$  corresponding to  $H_1$  and the mapping  $M_2$  corresponding to  $H_2$ . Such is the case in simulations of intense beams, where the map corresponding to  $H_1=H_{\text{ext}}$  can be computed to virtually any order using the techniques of magnetic optics, and the map corresponding to  $H_2=H_{\text{sc}}$  can be found using particle simulation techniques. Given  $M_1$  and  $M_2$ , the following algorithm is accurate through second order in  $\tau$ ,

$$M(\tau) = M_1(\tau/2)M_2(\tau)M_1(\tau/2) + O(\tau^3),$$

where  $\tau$  denotes a step in the independent variable. A well-known fourth-order algorithm is due to Forest and Ruth [37], and an arbitrary order scheme was derived by Yoshida [38]. These techniques are examples of symplectic integration algorithms. The unexpected generality of Yoshida's result was pointed out by Forest [39], who used it to describe arbitrary-order implicit integrators that do not involve splitting the Hamiltonian into separate pieces.

As a side note, if we apply the above second-order integrator to the model Hamiltonian,  $H=p^2/2+\Phi(q)$ , with  $H_1= p^2/2$  and  $H_2=\Phi(q)$ , the resulting integration algorithm is equivalent to the widely used leap-frog algorithm. However, the split-operator approach provides a powerful framework capable of dealing with the far more complicated Hamiltonians often encountered in accelerator physics. Split-operator methods have been used in many contexts outside of accelerator physics, including simulating nonlinear coherent structures in field theories [40] and solving the Schrodinger equation [41,42].

### 3.4 Particle-Based Methods

In the particle simulation approach, the beam distribution function is represented indirectly by macroparticles that evolve according to the underlying (single particle) equations of motion. The Poisson equation is solved at every time-step and may or may not involve the use of a spatial grid, though grid-based schemes are the most popular in the beam dynamics community at the present time. The particle simulation approach has been the workhorse of the beam dynamics community for many years. Typically a beam is represented by a collection of particles initially generated using Monte Carlo techniques, which are propagated through the accelerator under study using time advance algorithms. This approach has two major advantages: (1) in a grid-based scheme, the memory requirement scales as the cube of the system size for three-dimensional problems, not to the sixth power as is the case for direct solvers (see below), and (2) the method does not break down even when subgrid scales are generated by the nonlinear evolution. Disadvantages of this approach include numerical collisionality and sampling errors associated with using a finite number of macroparticles much smaller than the physical number of particles in the beam bunch. When one is interested in high spatial accuracy, this necessitates a fine grid and a correspondingly large number of macroparticles. The systematics underlying this procedure is well-known [43] and for a typical simulation of an intense charged-particle beam requires roughly ten particles per grid cell. For example, a simulation with a  $512^3$  grid in (x,y,z)-space has 134 million grid points. A beam dynamics simulation would therefore require roughly 1 billion particles. Such a calculation is only possible on large memory parallel supercomputers.

In the parallel Particle-In-Cell (PIC) approach, there are three main bottlenecks to parallel performance: (1) the fact that particles, and the grid points that they contribute to, may reside on different processors, necessitating irregular inter-processor communication; (2) the fact that some processors may be depositing far

more or far fewer particles on the grid than other processors (load imbalance); and (3) the fact that, traditionally, a field-solve is performed at each step, involving global communication, e.g. a Fast Fourier Transform. The first issue can be dealt with by using an optimized *particle manager* which ensures that, at every time step, particles and their associated grid points are on the same processor. A particle manager can also be used to balance the load after a specified number of steps (or automatically when certain conditions are satisfied). The third problem may be dealt with on shared memory machines by reducing the number of processors involved in the field solution; another approach is to avoid a global field solve at every time step by using an algorithm in which the fields themselves are dynamical quantities.

At this point it is worth commenting on the difference between beam dynamics, plasma, and cosmology simulations. In the case of beams, the number of charged particles in a bunch ranges from roughly  $10^9$  -  $10^{14}$  for typical applications. As such, we are approaching a simulation regime with the present and near-term generation of parallel supercomputers where simulations can be run with close to or exactly the physical number of particles. This is a totally different situation compared to many plasma physics applications (typically  $10^{20}$  particles) and cosmology (typically  $10^{70}$  in dark matter simulation codes). This distinction is important because it allows for the almost complete elimination of numerical collisionality in beam physics codes and the reintroduction of the physical collisionality, if desired, using Langevin techniques (described later) without a crippling memory overhead.

### 3.5 *Direct Solvers*

An alternative approach is to use a direct method in which the beam distribution function is defined explicitly by its values on a numerical grid in phase space. The dimensionality of the grid is  $2N_{\text{dim}}$ , corresponding to the space spanned by  $(\mathbf{q}, \mathbf{p})$ , where  $N_{\text{dim}}$  denotes the dimensionality of the problem. For this reason, direct Vlasov solvers in 2D and higher can only be performed with reasonable accuracy on large memory parallel supercomputers. For example, a 2D problem performed on a  $256^4$  grid in 64-bit precision would require 32 Gbyte to store a single copy of the main array, 64 Gbyte if stored as a complex array as is common in spectral codes. In the 3D case, a  $64^6$  problem would require approximately 500 Gbyte to store the array as a real array, and 1 Tbyte for the complex case. For the first time, such simulations will be within reach on multi-terascale hardware. Advantages of direct solvers include (1) no particle sampling errors and direct access to the phase space distribution function, and (2) no load balance issues on parallel platforms. Disadvantages include (1) very high memory cost, and (2) need for sub-grid smoothing in phase space if such structures develop due to the dynamics. The appropriate applications for direct solvers are in studies of beam equilibria and their stability, in beam physics studies that require access to the distribution function, and when good statistical information is desired in the tails of the distribution.

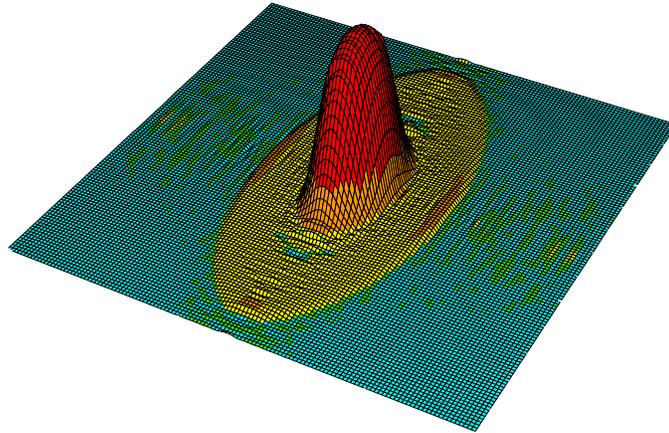
In analogy with their use in particle-based codes, split-operator methods can also be used in direct solvers. Consider the Vlasov equation written in its more traditional form (where  $H=p^2/2+\Phi$ , where  $\Phi$  is the potential):

$$\frac{\partial f}{\partial t} + (\mathbf{p} \cdot \partial_{\mathbf{q}})f - (\nabla\Phi \cdot \partial_{\mathbf{p}})f = 0.$$

Here, the potential  $\Phi$  is a sum of external and space charge contributions. To solve this equation, one approach is to use a spectral method combined with the above-mentioned split-operator techniques. For example, a 2<sup>nd</sup> algorithm is given by:

$$f(\mathbf{q}, \mathbf{p}, t) = \exp(-\frac{t}{2} \mathbf{p} \cdot \partial_{\mathbf{q}}) \exp(t \nabla\Phi \cdot \partial_{\mathbf{p}}) \exp(-\frac{t}{2} \mathbf{p} \cdot \partial_{\mathbf{q}}) f(\mathbf{q}, \mathbf{p}, 0) = 0.$$

To evaluate this using a spectral method, one first performs a Fourier transform of the distribution function,  $f(\mathbf{q}, \mathbf{p}, 0)$ , in position space, to which the right-most operator can be applied trivially; next one performs a backward transform in position and a forward transform in momentum, to which the middle operator can be applied; lastly one performs a backward transform in momentum and a forward transform in position, to which the third operator can be applied; a final backward transform in position returns  $f$  to the physical basis, and completes the step. In analogy to particle simulations, this can be generalized to arbitrary-order accuracy in time, if desired, using the algorithms of Forest and Ruth or Yoshida. Figure 3 shows the output from a parallel 2D spectral solver using a  $128^4$  grid (for a total of 268 million grid points).



**Figure 3.** Output from a parallel 2D direct solver showing the beam density in x-y space based on a  $128^4$  grid in phase space.

It is worth reiterating that the methods described have applicability beyond accelerator physics. Consider, for example, the numerical solution of the Schrodinger equation,

$$-\frac{\hbar^2}{2m}\nabla^2\psi + V\psi = i\hbar\frac{\partial\psi}{\partial t}.$$

In analogy with the treatment of the Vlasov equation, a second-order accurate integration algorithm for the Schrodinger equation is given by

$$\psi(x, \tau) = e^{-i\frac{\tau}{2}\frac{\hbar^2}{2m}\nabla^2} e^{iV\tau} e^{-i\frac{\tau}{2}\frac{\hbar^2}{2m}\nabla^2} \psi(x, 0).$$

As before, this can be evaluated using a spectral method by successively performing forward and backward Fourier transforms of  $\psi(x)$ . (In this case, the middle operator is simply a multiplication in real space, while the representation of the  $\nabla^2$  operation in the first and third operators is diagonal in Fourier space.) This was the algorithm used by Feit and Fleck to study molecular vibrations [41]. We now know that, by using the method of Yoshida, it is straightforward to extend this to arbitrary order accuracy in time if desired. Note that, for the Schrodinger equation, memory is not a major issue since the dimensionality of the grid does not grow as the square of the dimensionality of the problem, as was the case for the Vlasov equation. However, if one is interested in the quantum evolution of mixed states, then one would use a density matrix, in which case the memory would grow as the square of the system size. As a final comment, it is interesting to note that the Schrodinger equation can be solved using split-operator techniques without leaving coordinate space (i.e. without resorting to spectral methods). Consider the one-dimensional problem written as a classical field theory with Hamiltonian density [44] as shown:

$$H = -\frac{i\hbar}{2m}\frac{d\pi}{dx}\frac{d\psi}{dx} - \frac{i\hbar}{2}V\pi\psi.$$

In the above equation,  $\pi$  is canonically conjugate to  $\psi$  (and turns out to be proportional to the complex conjugate of  $\psi$ ). A simple discretization on a grid of size  $N$  leads to the following Hamiltonian,

$$H = -\frac{i\hbar}{2m}\sum_{j=1}^N \frac{(\pi_{j+1} - \pi_j)(\psi_{j+1} - \psi_j)}{\Delta^2} - \frac{i\hbar}{2}\sum_{j=1}^N V_j\pi_j\psi_j,$$

where  $\Delta$  is the grid spacing. This can be rewritten  $H=H_e+H_o$ , where the “even” and “odd” pieces involve only nearest-neighbor interactions, and where the mappings  $M_e$  and  $M_o$ , corresponding to  $H_e$  and  $H_o$ , can be found analytically. Thus, a split-operator integrator like that described above can be used to evolve the system forward in time. This method has been used by DeRaedt to study a variety of quantum systems [42]. Unlike the spectral approach, this method involves no global communication, only nearest-neighbor (or other nearby and regular) communication.

Lastly, it is worth pointing out a distinction between using Hamiltonian-based split-operator methods for finite- and infinite-dimensional field theories. In the finite-dimensional case, split-operator techniques are useful not only as integrators, but also for computing the associated symplectic mappings themselves. Map-based methods are widely used in magnetic optics not only for “pushing particles,” but also for optical design by performing analysis, fitting, and optimization of map coefficients. On the other hand, the dimensionality of truncated, discretized infinite-dimensional theories leads to very large system matrices. For example, the Schrodinger problem above has dimensionality  $2N$ , where  $N$  is the number of grid points. Hence, the transfer map is a  $2N \times 2N$  *dense* matrix. Clearly, high-resolution problems require very large memory. While such memory is not needed simply to evolve an initial distribution, it is needed to analyze the system matrix, which contains important information such as the energy eigenvalues of the system.

### 3.6 Stochastic Dynamics

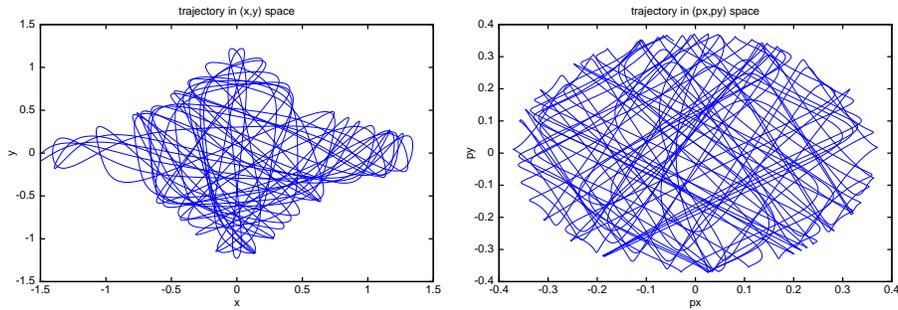
Stochastic contributions to Vlasov/Poisson evolution can occur due to particle collisions and noise in external fields. External noise is easy to model via standard stochastic (Langevin) techniques applied to the particle equations of motion. The treatment of particle collisions is more subtle and requires the distinction between weak and strong encounters. Weak or soft collisions refer to interactions in which  $\delta v \ll v$  where  $\delta v$  is the change of velocity during the collision. Strong or hard collisions have  $\delta v \sim v$ . Soft collisional effects may be included by generalizing the Vlasov/Poisson equations to a Fokker-Planck form (usually the assumption of locality, i.e. impact parameter small compared to system size, is also made). The corresponding change in the macroparticle dynamics is reflected in the appearance of noise and damping terms in the equations of motion as a result of a systematic expansion in inverse powers of the Coulomb logarithm [45]. Hard collisions can be included within the Boltzmann approximation. In particle codes, hard collisions may be treated using Direct Simulation Monte Carlo (DSMC) techniques.

Depending on the problem at hand, the Fokker-Planck equation to be solved may be more or less complicated. The simplest situations arise when the damping and diffusion coefficients are both constants (coupling to external environment) or

when simplifying assumptions such as an isotropic velocity distribution turn out to be valid. Simple Langevin models corresponding to some of these cases have already been added to our codes. Consider, for example, a system of interacting particles in a quartic potential governed by the following Hamiltonian which includes the self-potential  $\Phi$ :

$$H = \frac{1}{2}(p_x^2 + p_y^2) + \alpha x^4 + \beta y^4 + \frac{1}{2}x^2y^2 + q\Phi(x, y)$$

By including damping (via a time-dependent mass) and diffusion (via a stochastic force) it is possible to numerically drive this system to thermal equilibrium. Figure 4 shows the output from a 2D parallel Langevin code. The figure on the left shows the trajectory of a particle in coordinate space, and the figure on the right shows a trajectory in momentum space. For this simulation, we verified that equilibrium had been achieved by turning off the damping and diffusion part-way through the simulation and observing that there was no change in the rms emittances.



**Figure 4.** Output from a 2D, parallel Langevin code showing a particle trajectory in coordinate space and in momentum space.

Using the next generation of parallel supercomputers, it should be possible to solve, self-consistently, the full Fokker-Planck equation using techniques very similar to those used in parallel Vlasov/Poisson solvers. Consider, for example, the Landau form of the Fokker-Planck equation,

$$\frac{\partial}{\partial t} f + \sum_{i=1}^3 v_i \frac{\partial}{\partial x_i} f - \sum_{i=1}^3 \left( \frac{\partial}{\partial x_i} \varphi \right) \frac{\partial}{\partial v_i} f + \sum_{i=1}^3 \frac{\partial}{\partial v_i} F_i f - \sum_{i,j=1}^3 \frac{1}{2} \frac{\partial^2}{\partial v_i \partial v_j} D_{ij} f = 0$$

where the damping and diffusion terms can be cast in terms of Rosenbluth potentials

$$F_i \propto \frac{\partial}{\partial v_i} H, \quad D_{ij} \propto \frac{\partial^2}{\partial v_i \partial v_j} G,$$

which are in turn given by

$$H(x, v) = \int dv' \frac{f(x, v')}{|v - v'|}, \quad G(x, v) = \int dv' f(x, v') |v - v'|.$$

Such a situation is clearly nonlinear since, in analogy to the computation of the self-fields in the Vlasov/Poisson case, the calculation of F and D (or equivalently H and G) depends on the distribution function. Furthermore, it is clear from [46] that by introducing a grid in velocity space and depositing macroparticle velocities onto the grid, we may obtain the damping and diffusion coefficients using exact analogs of the techniques used to solve the Poisson equation in PIC codes. (It should be kept in mind that Fokker-Planck techniques can be correctly applied only when numerical collisionality is already negligible in the corresponding collisionless simulation.)

Even in the local approximation discussed here, the noise and damping terms in the Langevin equations that correspond to the above Fokker-Planck description are functions of both space and velocity. Thus, in principle, a six-dimensional space is required to hold the information relating to the values of the transport coefficients, with a potentially enormous memory cost. Fortunately, the corresponding phase space resolution needed for acceptable accuracy is far lower than that required for acceptable force accuracy in the parent PIC simulation (since the quantity being calculated is a modest correction treated as a stochastic force). For this reason, such simulations are now within reach. Nevertheless, the calculations will push the envelope of what is possible, even on multi-terascale platforms. For example, even a crude  $16^3$  grid in position space would involve the equivalent amount of computation of 4096 space-charge calculations. Though one may use symmetry or near-symmetry to reduce this to perhaps one-eighth this value, the prospect of performing 512 convolutions instead of one per step is clearly a task requiring very high performance. Also, the fact that the calculation of the damping and diffusion coefficients will take place on a small grid (e.g.  $16^3 \times 16^3$  in x-v space) means that 3D domain decomposition will likely be required, unlike the 1D or 2D decomposition that can be used for most Vlasov/Poisson problems. Lastly, it should be noted that the treatment of such a system using Langevin techniques would involve treating systems with multiplicative noise, since the stochastic force associated with F (appearing in the equation for  $d\mathbf{p}/dt$ ) is velocity dependent. Fortunately, we have already developed a 2nd order algorithm for treating systems with multiplicative noise [47].

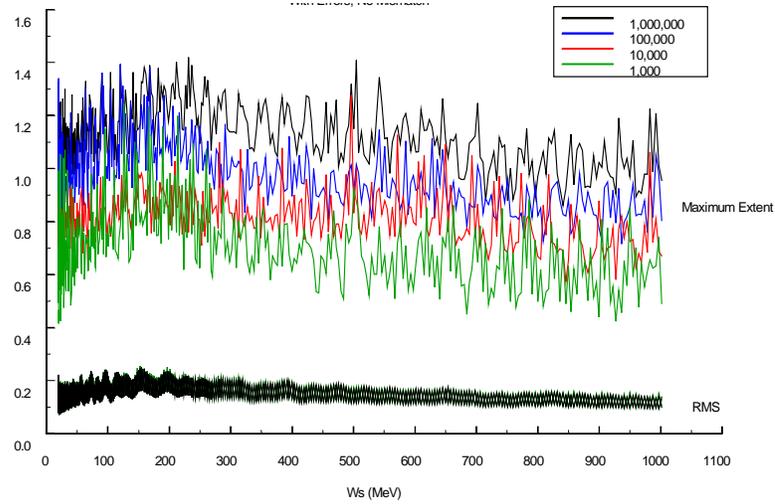
#### 4 The DOE Grand Challenge in Computational Accelerator Physics

In 1997 the U.S. DOE initiated a Grand Challenge in computational accelerator physics. The primary goal of this project is to develop a new generation of accelerator modeling tools, targeted to very large scale computing platforms, and apply them to accelerator projects of national importance. The multi-disciplinary Grand Challenge team consists of scientists from Los Alamos National Laboratory (LANL), Stanford Linear Accelerator Center, Stanford University, the University of California at Los Angeles, and two national computing centers, the Advanced Computing Laboratory (ACL) located at LANL and the National Energy Research Scientific Computing Center located at Lawrence Berkeley National Laboratory.

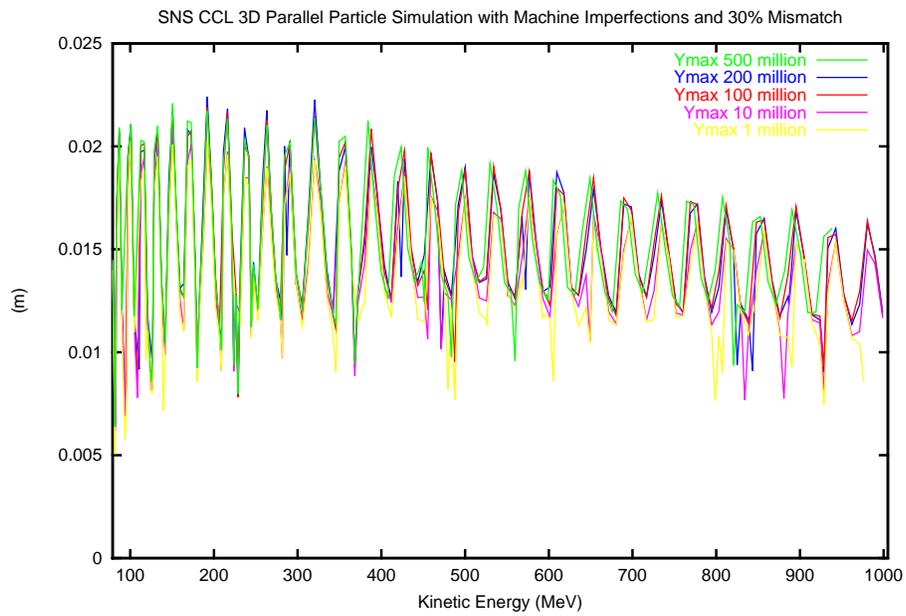
Beam dynamics is a major focus area of the grand challenge. Prior to the inception of the grand challenge, most beam dynamics simulations of rf accelerators were performed with 10,000 simulation particles, and occasionally with 100,000 particles. Larger simulations were rarely performed because they were at the limit of computing capability on single processors. New multiprocessor capabilities developed under the grand challenge have made 100 million particle simulations routine, and simulations with 1 billion particles have been performed.

A major accomplishment is the development of IMPACT (Integrated Map and Particle Accelerator Tracking code), a new parallel 3D beam dynamics code for modeling intense beams in rf linacs. IMPACT is based on the split-operator methods described in the preceding section. It has been implemented using three approaches: the POOMA (Parallel Object-Oriented Methods and Applications) approach, F90/MPI, and High Performance Fortran. IMPACT has been used to support the Spallation Neutron Source (SNS) project [48], the Accelerator Production of Tritium project, and in theoretical studies of beam halo phenomena. It runs on the SGI Origin 2000 at the ACL and on the SGI/Cray T3E and IBM SP at NERSC.

To illustrate how large-scale, parallel simulations have changed the way accelerator scientists do research, consider a simulation of the SNS linac. Figure 5 shows the predicted maximum beam size from a simulation using legacy code run on a PC using 1000 to 1 million macroparticles. Note that the maximum beam size does not converge as the number of macroparticles is increased. In contrast, figure 6 shows the analogous IMPACT simulations using 1 million to 500 million particles. For the 500 million particle case, the charge density was stored on a grid of size  $256^3$  (corresponding to a  $512^3$  computational grid, used along with a modified Green function, to rigorously treat open boundary conditions). It is seen that the maximum amplitude is approaching a limiting value above roughly 100 million particles. Note that a run using 100 million particles requires 5 hours on 256 processors of the ACL Nirvana system. Compared with a PC run of 1 million particles which required 2 days, the parallel run, which is 100 times bigger and 10 times faster than the PC run, represents an increase in simulation capability of a factor of 1000. Also, the PC run was two-dimensional, while the IMPACT run was three-dimensional.

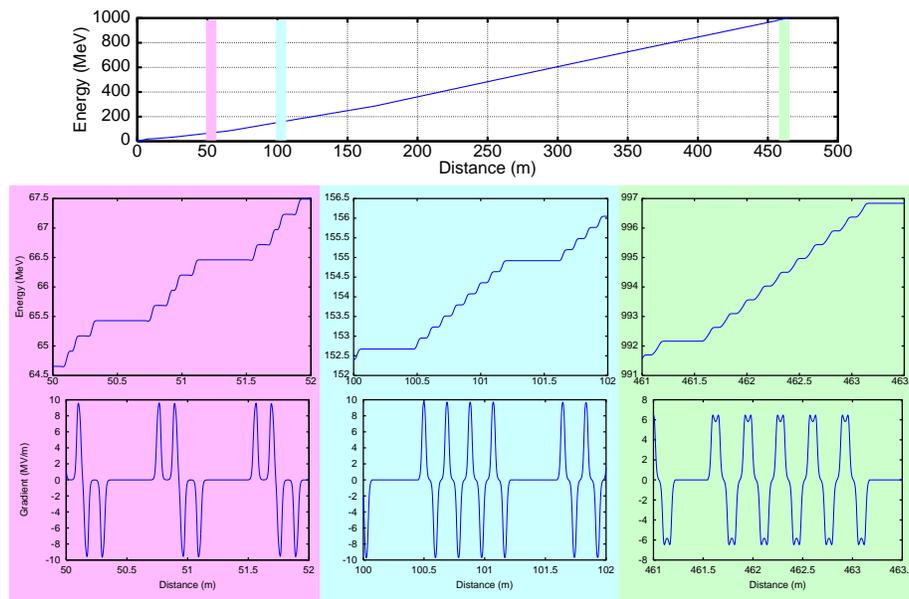


**Figure 5.** Maximum particle amplitude and rms beam size in the SNS linac based on PC simulations using 1000, 10000, 100000, and 1 million simulation particles.



**Figure 6.** Maximum particle amplitude in the SNS linac with varying number of macroparticles, simulated on the ACL Nirvana system using 32-1024 processors.

A unique feature of these simulations is the treatment of rf cavities. For these simulations, over 400 cavities were modeled using the code SUPERFISH. The upper part of figure 7 shows the energy gain of the synchronous particle as a function of distance. The three images beneath it show detail at 50 MeV, 100 MeV and 460 MeV. The three images show the rapid variation in the cavity fields as a function of distance. They also show that the energy gain within the cavities is modeled in very fine detail. This is done at almost no cost in computer time, because small integration steps are only used to compute the 12 coefficients in the gap transfer map. This illustrates an important feature of the split-operator approach: it allows one to separate the rapid variation in externally applied fields from the more slowly varying space-charge fields. As a result, one can take large steps between time-consuming space-charge kicks.



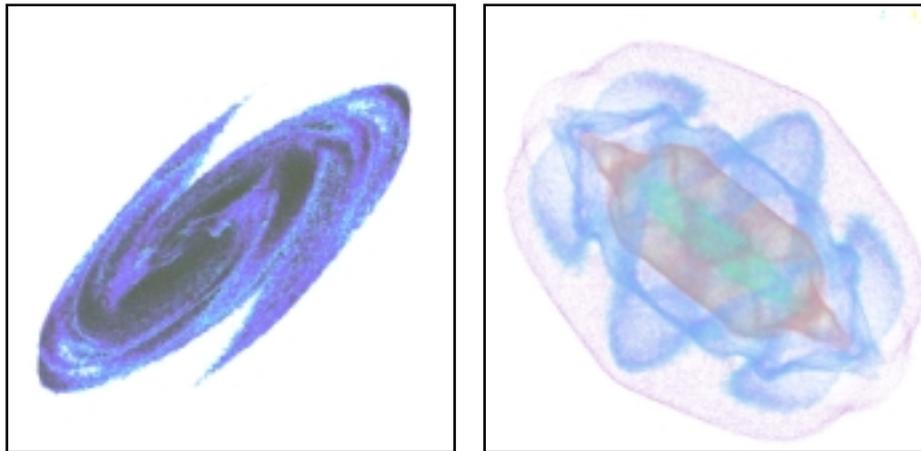
**Figure 7.** RF cavity fields and energy gain in the SNS linac modeling using IMPACT.

## 5 Visualization.

The huge amount of data produced by large-scale parallel simulations necessitates the use of high performance visualization resources to analyze the data. Consider, for example, a 500 million particle simulation. The particle array is of size  $6 \times 5M$  since there is a 6-vector  $(x, p_x, y, p_y, t, p_t)$  for each particle. Stored as 4 byte quantities,

the amount of data generated at each step is 12 Gbyte. A typical simulation involves several thousand steps, yielding a total output of several 10's of Tbytes. In order to extract the most useful information, one must also deal with the fact that the data are 6-dimensional. We typically visualize and analyze several 3D projections of our data, for example  $(x,y,z)$ ,  $(x,p_x,z)$ , and  $(x,t,p_t)$ . The first of these shows the beam density in real space; the second shows the transverse (horizontal) phase space as a function of longitudinal position within the beam bunch; and the third shows the longitudinal phase space as a function of transverse position within a bunch.

Working with the visualization team at the ACL, we are now using volume rendering techniques to look at our simulation results. In this approach, a 3D scalar field is treated as a semi-transparent medium. Color and transparency are controlled by scalar transfer functions, i.e. by "color" and "alpha" maps, respectively. The graphics pipes on the Nirvana system at the ACL each have 64 Mbyte of texture memory and can hold up to a  $512 \times 256 \times 256$  volume. A single pipe is able to render volumes of size  $256^3$  at interactive rates. However, we are now rendering  $512^3$  data sets by using multiple pipes, and subdividing the volume into sub-volumes each of which is handled by a separate pipe. Figure 8 shows images based on simulations of the SNS linac. The data being visualized is the 3D projection  $(x,p_x,z)$ . The two images illustrate the use of different color and alpha maps. The image on the left has a high degree of opacity and has the z-axis coming out of the paper, which provides for a clearer view of the  $(x-p_x)$  phase space, including its s-shaped tail produced by amplitude-dependent phase advance. The image on the right is from a different viewpoint and shows how, using transparency, we can look at what are effectively 4 separate isosurfaces of density.



**Figure 8.** Volume rendering of output from a simulation of the SNS linac, shown from different viewpoints and with varying degrees of transparency.

## 6 Acknowledgements

Partial funding for this work was provided by the U.S. Department of Energy, Office of Science (DOE/OS), Office of High Energy and Nuclear Physics, Division of High Energy Physics, through the Los Alamos Accelerator Code Group. Partial funding was also provided by the DOE/OS, Office of Advanced Scientific Computing Research, Division of Mathematical, Information, and Computational Sciences, through the Grand Challenge in computational accelerator physics. Computational resources were provided by the Advanced Computing Laboratory located at Los Alamos National Laboratory and the National Energy Research Scientific Computing Center located at Lawrence Berkeley National Laboratory.

## References

1. Science 285, no. 5431, p. 1194.
2. Science 285, no. 5432, p. 1342.
3. Science 286, no. 5437, p. 28.
4. Science 286, no. 5448, p. 2239.
5. Science 285, no. 5436, p. 2048.
6. Science 286, no. 5448, p. 2250.
7. R. L. Gluckstern, Phys. Rev. Lett. **73**, 1247 (1994).
8. J. Lagniel, Nucl. Inst. Meth. Res. A **345**, 405 (1994).
9. H. Okamoto and M. Ikegami, Phys. Rev. E **55**, 4694 (1997)
10. R. L. Gluckstern et al., Phys. Rev. E, **58**, 4 (1998).
11. T. Wangler et al., Phys. Rev. ST Accel. Beams 1, 084201 (1998).
12. A. V. Fedotov et al., Phys. Rev. ST Accel. Beams 2, 014201 (1999).
13. M. Ikegami, Phys. Rev. E **59**, 2330 (1999).
14. T. F. Wang, Phys. Rev. E **61**, 855 (2000).
15. J. Qiang, R. Ryne, and S. Habib, "Beam Halo Studies Using a 3-Dimensional Particle-Core Model," Proc. 1999 Particle Accelerator Conf., IEEE Press, Piscataway, N.J., Aug. 1999, pp.1845-1847. (submitted to Phys. Rev. ST Accel. Beams).
16. R. W. Hockney and J. W. Eastwood, Computer Simulation Using Particles, Adam Hilger, New York (1988).
17. C. K. Birdsall and A. B. Langdon, Plasma Physics via Computer Simulation, McGraw-Hill, New York (1985)
18. P. C. Liewer and V. K. Decyk, J. Comp. Phys. **85**, 302 (1989).
19. C. S. Lin et al., J. Parallel and Distributed Computing **8**, 196 (1990).
20. R. Ferrell and E. Bertschinger, Int. J. Mod. Phys. C **5**, 933 (1994).
21. J. Wang, P. Liewer, V. Decyk, Comp. Phys. Comm. **87**, 35 (1995).
22. V. K. Decyk, Comp. Phys. Comm. **87**, 87 (1995).
23. J. V. W. Reynders et al., Comp. Phys. Comm. **87**, 212 (1995).

24. P. M. Lyster et al., *Computers in Physics* **9**, 420 (1995).
25. H. X. Vu, *J. Comp. Phys.* **144**, 257 (1998).
26. A. Friedman, D. P. Grote, and I. Haber, *Phys. Fluids B* **4**, 2203 (1992).
27. R. Ryne and S. Habib, *Part. Accel.* **55**, 365 (1995).
28. J. Qiang, R. Ryne, S. Habib, V. Decyk, "An Object-Oriented Parallel Particle-In-Cell Code for Beam Dynamics Simulation in Linear Accelerators," Proceedings of the ACM/IEEE SC99 Conference, Portland, OR (1999). (submitted to *J. Comp. Phys.*).
29. J. Qiang, R. Ryne, and S. Habib, "Fortran Implementation of Object-Oriented Design in Parallel Beam Dynamics Simulations," to appear in *Computer Physics Communications*, 1999.
30. A. Dragt, "Lectures on Nonlinear Orbit Dynamics," in *Physics of High Energy Particle Accelerators*, AIP Conf. Proc. **87**, R. A. Carrigan et al., ed. (1982).
31. A. Dragt and E. Forest, *J. Math. Phys.* **24**, 2734 (1983)
32. A. Dragt et al., *Ann. Rev. Nucl. Part. Sci.* **38**, 455 (1988).
33. E. Forest, *J. Math. Phys.* **31** 5, 1133 (1990).
34. E. Forest, *Beam Dynamics: A New Attitude and Framework*, (in *The Physics and Technology of Particle and Photon Beams*, vol. 8), Harwood Academic Publishers (1998).
35. M. Berz, *Nucl. Inst. Meth. Phys. Res. A* **363**, 100 (1995).
36. J. Irwin, *Part. Accel.* **55**, 107 (1995).
37. E. Forest and R. Ruth, *Physica D* **43**, 105 (1990).
38. H. Yoshida, *Phys. Lett. A* **150**, 262 (1990).
39. E. Forest, J. Bengtsson, and M. Reusch, *Phys. Lett. A* **158**, 99 (1991).
40. S. Habib, in preparation.
41. M. D. Feit, J. A. Fleck, Jr., and A. Steiger, *J. Comp. Phys.* **47**, 412 (1982).
42. H. DeRaedt, *Comp. Phys. Reports* **7**, 1 (1987).
43. H. Okuda and C. K. Birdsall, *Phys. Fluids* **8**, 2123 (1970).
44. E. Merzbacher, *Quantum Mechanics*, Wiley, New York (1970).
45. H. Ichimaru, *Basic Principles of Plasma Physics*, Benjamin/Cummings (1973).
46. M. Rosenbluth, *Phys. Rev.* **107**, 1 (1957).
47. J. Qiang and S. Habib, "A Second-Order Stochastic Leap-Frog Algorithm for Multiplicative Noise Brownian Motion," Los Alamos Report LA-UR 99-5263, (submitted to *Physical Review E*).
48. Ji Qiang et al., "High Resolution Parallel Particle-In-Cell Simulations of Beam Dynamics in SNS Linac," Los Alamos Report LA-UR-99-6753 (submitted to *Nucl. Instr. and Meth. in Phys. Res. - Section A*).