View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by UNT Digital Library

LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

UCRL-CONF-226076

Monte Carlo Particle Transport Capability for Inertial Confinement Fusion Applications

Patrick S. Brantley, Linda M. Stuart

November 13, 2006

M&C + SNA 2007
Monterey, CA, United States
April 15, 2006 through April 19, 2006

Disclaimer

This document was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor the University of California nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or the University of California, and shall not be used for advertising or product endorsement purposes.

Joint International Topical Meeting on Mathematics & Computation and Supercomputing in Nuclear Applications (M&C + SNA 2007)
Monterey, California, April 15-19, 2007, on CD-ROM, American Nuclear Society, LaGrange Park, IL (2007)

MONTE CARLO PARTICLE TRANSPORT CAPABILITY FOR INERTIAL CONFINEMENT FUSION APPLICATIONS

Patrick S. Brantley and Linda M. Stuart
Lawrence Livermore National Laboratory
P.O. Box 808, L-023
Livermore, CA 94551 USA
brantley1@llnl.gov

ABSTRACT

A time-dependent massively-parallel Monte Carlo particle transport calculational module (ParticleMC) for inertial confinement fusion (ICF) applications is described. The ParticleMC package is designed with the long-term goal of transporting neutrons, charged particles, and gamma rays created during the simulation of ICF targets and surrounding materials, although currently the package treats neutrons and gamma rays. Neutrons created during thermonuclear burn provide a source of neutrons to the ParticleMC package. Other user-defined sources of particles are also available. The module is used within the context of a hydrodynamics client code, and the particle tracking is performed on the same computational mesh as used in the broader simulation. The module uses domain-decomposition and the MPI message passing interface to achieve parallel scaling for large numbers of computational cells. The Doppler effects of bulk hydrodynamic motion and the thermal effects due to the high temperatures encountered in ICF plasmas are directly included in the simulation. Numerical results for a three-dimensional benchmark test problem are presented in 3D XYZ geometry as a verification of the basic transport capability. In the full paper, additional numerical results including a prototype ICF simulation will be presented.

Key Words: Inertial confinement fusion, Monte Carlo particle transport.

1. INTRODUCTION

The Monte Carlo method has been used for decades to simulate particle transport. The accurate simulation of the transport of neutrons, charged particles, and gamma rays in inertial confinement fusion (ICF) targets and surrounding materials can aid a correct understanding of target performance as well as design of diagnostics. In this paper, we briefly describe a time-dependent massively-parallel Monte Carlo particle transport calculational module (ParticleMC) that has been developed for inertial confinement fusion (ICF) applications.

The remainder of this paper is organized as follows. In Section 2, we briefly summarize the numerical algorithms in the ParticleMC package. We present numerical results for a three-dimensional benchmark test problem in Section 3. We conclude with a brief discussion in Section 4.

2. Monte Carlo Algorithms

The ParticleMC package is written using object-oriented techniques in the standard C++ programming language. Extensive use of the concepts of inheritance and templating are utilized throughout the package. The ParticleMC package has, since inception, been designed for massively-parallel three-dimensional

simulations with large numbers of spatial zones in the computational mesh. As a result, the package can operate on computational meshes that are spatially domain-decomposed. Parallel communication is performed using the Message Passing Interface (MPI) [1]. Templating on the computational mesh type resulted in significant code reuse between 3D XYZ and 2D RZ geometry versions of the ParticleMC package.

The ParticleMC package tracks Monte Carlo particles (currently neutrons and gamma rays, and eventually light charged particles) through a computational mesh. Source particles can be generated either by a Monte Carlo simulation of thermonuclear processes [2] or by user-specified external sources. Each Monte Carlo particle maintains a three-dimensional Cartesian representation of its position, for both 3D XYZ meshes and 2D RZ meshes, as well as direction cosines with respect to the Cartesian coordinate system. Each Monte Carlo particle also maintains a type identifier (currently neutron or gamma ray), a spatial zone identifier corresponding to its position, its current kinetic energy, an energy group index, a weight value corresponding to the number of physical particles the Monte Carlo particle represents, and its current simulation time. Finally, each Monte Carlo particle carries an individual random number generator state to ensure that it encounters the same random number stream regardless of the number of processors on which the simulation is run. The ParticleMC package currently uses the RNG random number generator library [3] developed at LLNL, and the state size for this random number generator is small. Because floating point arithmetic is not commutative, the simulation results are not yet reproducible on different numbers of processors.

The Monte Carlo particles are tracked through the computational mesh in the laboratory frame. Since the package is designed for use in simulations with hydrodynamics, particles are transformed to the fluid frame to access nuclear cross sections and to perform collision kinematics. The ParticleMC package is typically used for time-dependent simulations, so Monte Carlo particles must be tracked from the beginning of the timestep to the end of the timestep (assuming they are not captured or escape the problem boundary within the timestep). Monte Carlo particles that reach the end of the timestep are stored in a census list of particles. At the beginning of the next timestep, the census list is processed through a census comb to select, via splitting and Russian Roulette, the desired number of Monte Carlo particles to be followed during the timestep. The Monte Carlo particle population can decrease due to capture or escape, can increase over simulation time if few particles are captured or escape during a timestep (e.g. for small timestep sizes or for highly-scattering media), and can significantly increase through the application of variance reduction techniques. Additional functionality maintains the desired apportioning of the Monte Carlo particles to be tracked between census particles and particles from external sources. At the beginning of a timestep, the position of a particle and the zone identifier it maintains may be inconsistent as a result of hydrodynamic motion or mesh relaxation. To make these two quantities consistent, we have implemented an idea suggested by Zimmerman [4] that tracks the particle from the center of the zone corresponding to its stored zone index to the position of the particle. This algorithm is efficient and naturally handles any communication required as a result of crossing a spatial domain boundary using the existing Monte Carlo particle tracking functionality.

Isotopic species sampling, nuclear reaction type sampling, collision kinematics calculations, and outgoing particle characteristics sampling are performed using the Monte Carlo All Particle Method (MCAPM) library [5]. This library utilizes a continuous value for the neutron kinetic energy and performs continuous collision kinematics but uses a multigroup representation of the neutron cross sections with typically 175 energy groups. Gamma ray cross sections are stored as pointwise data with typically 176 energy points.

We have implemented three neutron thermalization models to account for the effects of target thermal

motion encountered in the high-temperature plasmas typical of ICF calculations. We describe these models in order of increasing fidelity and computational expense. The first and simplest "$\frac{3}{2}kT$ floor" model uses room temperature microscopic neutron cross sections but enforces a minimum neutron energy of $\frac{3}{2}kT$, where kT is the zonal ion temperature, after a neutron interaction with the background matter: $E_n^{out} = max(E_n^{out}, \frac{3}{2}kT)$. Energy is conserved by subtracting the amount of energy required to increase the neutron to the zone thermal energy from the energy to be deposited from nuclear reactions to the background matter. The second neutron thermalization model was recently implemented in the MCAPM library [5] based on the the THERMAL subroutine of Cullen [6] and assumes that temperature-broadened microscopic neutron cross sections are available (the MCAPM library can provide these temperature-dependent cross sections). This model resamples elastic scattering collisions using thermally-broadened cross sections if the incident neutron energy satisfies $E_n^{in} < 10^4 \times (\frac{3}{2}kT)$. The third neutron thermalization model treats the interaction of a neutron with the hot plasma in two stages. The first stage involves determining the correct distance to collision which can only be accomplished using the thermally-broadened total macroscopic cross section [2, 7]. Once it is determined that a collision occurs, the second stage involves sampling the characteristics of the neutron reaction. The isotopic species undergoing the collision is sampled using the thermally-broadened cross sections. Given the sampled isotopic species, the thermal velocity of the nucleus must be sampled favoring energy and direction pairs with larger reaction rates (not simply an isotropic Maxwellian distribution) [2]. We accomplish this using the rejection technique recommended in Ref. [2]. After the target thermal velocity is sampled, the collision kinematics are performed using the MCAPM library collision functions in the frame of the thermal target.

For time-dependent, domain-decomposed simulations, statistical error estimates are difficult to obtain via techniques employed in typical Monte Carlo codes in which particle histories are tracked from beginning to end. As a result, we have chosen to use the technique known as *batching* [2] in which each Monte Carlo particle has a batch identifier. An estimate of the statistical error in a desired tally can then be obtained by computing the standard deviation in the mean of the tally as computed from the particles in each batch.

We have currently implemented variance reduction via importance regions coupled with splitting and Russian roulette [8]. The importance values can be specified on a zonal basis if desired. An important aspect of the implementation of this technique for a Monte Carlo method employed in time-dependent simulations is to correctly account for the importance values during the application of the census comb.

3. NUMERICAL RESULTS

In this section, we present numerical results for the semi-analytic three-dimensional searchlight benchmark problem proposed by Kornreich and Ganapol [9] as a verification of the basic Monte Carlo transport capability. This problem models a monoenergetic and monodirectional beam of neutrons incident on the surface of a semi-infinite isotropically-scattering half-space at the point (x, y, z) = (0, 0, 0) cm. The incident beam of neutrons is canted with respect to the free face of the half-space, resulting in a three-dimensional scalar flux distribution. The neutron transport is considered to be monoenergetic. The semi-analytic benchmark values for the scalar flux are tabulated every 0.5 mean free paths (mfp) into the half-space (along the z axis) in Ref. [9] for several values of the scattering ratio c. We have considered the case of the incident beam canted with a cosine of $\mu_0 = 0.9$ and a scattering ratio of c = 0.9. This problem is challenging for a Monte Carlo transport simulation, as the scalar flux decreases by over a factor of one hundred for the points of comparison. We solved this problem in a spatial domain of 60 $\times$ 60 $\times$ 30 mfp using an orthogonal computational mesh with 14,749 zones domain-decomposed into four domains. The spatial zones along the z axis (in which the zonal scalar flux values were computed using a pathlength

estimator [8]) were 0.1 × 0.1 × 0.1 mfp in size. The simulation used 20 million Monte Carlo neutrons and 32 particle batches for statistical error estimation. In Table I, we present the benchmark [9] and computed scalar flux values. We also show the percent relative standard deviation (PRSD) in the Monte Carlo scalar flux values as computed using the batching approach as well as the percent relative error (PRE) in the Monte Carlo values as compared to the benchmark values. Good agreement is obtained between the Monte Carlo results and the benchmark values: 40% and 90% of the points agree to within one and two standard deviations, respectively.

Table I. Benchmark and Computed Scalar Flux Values

	Scalar Flux			
z	Benchmark	Computed	PRSD	PRE
0.5	5.9281×10^{-1}	5.9967×10^{-1}	0.28	1.16
1.0	2.2346×10^{-1}	2.2479×10^{-1}	0.48	0.60
1.5	1.1243×10^{-1}	1.1201×10^{-1}	0.69	-0.37
2.0	6.3715×10^{-2}	6.3878×10^{-2}	0.91	0.26
2.5	3.8547×10^{-2}	3.7996×10^{-2}	1.18	-1.43
3.0	2.4311×10^{-2}	2.4205×10^{-2}	1.21	-0.43
3.5	1.5784×10^{-2}	1.5468×10^{-2}	1.58	-2.00
4.0	1.0470×10^{-2}	1.0652×10^{-2}	1.85	1.74
4.5	7.0614×10^{-3}	6.7934×10^{-3}	2.64	-3.80
5.0	4.8256×10^{-3}	4.6568×10^{-3}	3.13	-3.50

4. CONCLUSIONS

In this paper, we briefly described the capabilities of a time-dependent massively-parallel Monte Carlo particle transport capability designed for inertial confinement fusion applications. We presented numerical results for a three-dimensional benchmark transport problem as a verification of the basic Monte Carlo transport capability. In the full paper, additional numerical results including a prototype ICF simulation will be presented.

In future work, we plan to implement a Monte Carlo charged particle transport capability. In addition, we plan to pursue stategies to improve the computational efficiency of the package.

ACKNOWLEDGEMENTS

This work was performed under the auspices of the U.S. Department of Energy by the University of California Lawrence Livermore National Laboratory under contract No. W-7405-Eng-48. We would like to acknowledge the Monte Carlo particle tracking and parallel communication capabilities initially developed by Nick Gentile upon which the ParticleMC package is built. We also acknowledge many helpful conversations and much helpful guidance from George Zimmerman.

REFERENCES

[1] W. Gropp, E. Lusk, and A. Skjellum, *Using MPI: Portable Parallel Programming with the Message-Passing Interface*, The MIT Press, Cambridge, Massachusetts, USA (1999).

[2] G.B. Zimmerman, "Monte Carlo Methods in ICF," *Proceedings of Thirteenth International Conference on Laser Interactions and Related Plasma Phenomena*, Monterey, CA, April 13-18, 1997 (1997). See also Lawrence Livermore National Laboratory internal report UCRL-JC-126845 (1997).

[3] B.R. Beck and E.D. Brooks III, "The RNG Random Number Library," Lawrence Livermore National Laboratory internal report UCRL-MA-141673 (2000).

[4] G.B. Zimmerman, private communication (2002).

[5] P.S. Brantley, C.A. Hagmann, and J.A. Rathkopf, "MCAPM-C Generator and Collision Routine Documentation (Revision 1.2)," Lawrence Livermore National Laboratory internal report UCRL-MA-141957 (2003).

[6] D.E. Cullen, "THERMAL: A Routine Designed to Calculate Neutron Thermal Scattering," Lawrence Livermore National Laboratory internal report UCRL-ID-120560 (1995).

[7] E.H. Canfield, "Monte Carlo Methods for Neutrons in a Hot Gas," Lawrence Livermore National Laboratory internal report UCIR-694 (1973).

[8] E.E. Lewis and W.F. Miller, *Computational Methods of Neutron Transport*, American Nuclear Society, Lagrange Park, Illinois, USA, Ch. 1 (1993).

[9] D.E. Kornreich and B.D. Ganapol, "Numerical Evaluation of the Three-Dimensional Searchlight Problem in a Half-Space," *Nucl. Sci. Engr.*, **127**, pp. 317-337 (1997).