A RANDOMIZED APPROACH TO HYBRID MONTE CARLO SIMULATION

Brian Thorndyke, Paul Fishwick and Sanguthevar Rajasekaran
Department of Computer and Information Sciences and Engineering
University of Florida
301 CISE, P.O. Box 116120, Gainesville, FL 32611
e-mail: thorndyb@cis.ufl.edu

Keywords: *Molecular Dynamics, Hybrid Monte Carlo, Particle Simulation*

ABSTRACT

The two primary methods of simulating soft-sphere particle systems are Molecular Dynamics and Monte Carlo. Researchers have recently combined the two methods into a single Hybrid Monte Carlo algorithm, which combines the potential barrier-crossing ability of Monte Carlo with the large moves in phase space possible through Molecular Dynamics. We extend the algorithm by applying randomized selection to the Molecular Dynamics aspect of the hybrid model. By simulating a 2-D periodic box of Lennard-Jonesium, we show that our approach can reduce the computational cost of the hybrid model by up to 50%, without compromising the accuracy of the simulation.

1 INTRODUCTION

Soft-sphere particle simulations are inherently time-consuming. This is because every particle influences every other particle as the system moves through phase space, requiring quadratic time to naively update the system each time step. This rapidly becomes computationally prohibitive, even on modern-day workstations.

One way of reducing the time complexity is by approximating the interactions in some way. Unfortunately, the gain in speed usually comes at the expense of simulation accuracy, and one must be careful to ensure the validity of the simulation in such cases. A second approach is to sample the phase space more efficiently, perhaps maintaining the same computational complexity at each time step, but reducing the number of time steps required to measure the quantity of interest to sufficient precision.

Molecular Dynamics (MD) moves a system through phase space according to the Hamiltonian equations of motion, and in doing so maintains the correct dynamics of the system. However, local potential minima and harmonic modes can often trap the system, leading to poor statistical measurements. On the other hand, Monte Carlo (MC) samples directly from the phase space, based on random perturbations each time step. One advantage is that potential barriers can be crossed more easily than with MD. However, because MC is essentially a random walk through phase space, its expected distance after N steps is $\leq \sqrt{N}$ [Neal 1993]. As a result, it can take a long time for distant areas of phase space to be sampled.

Recent research has seen the development of hybrid models, in which the properties of MC are combined with those of MD. Guarnieri and Still [Guarnieri and Still 1994] developed an alternating MC / Stochastic Dynamics scheme, which was found to sample a system of n-pentane molecules significantly faster than MC or Stochastic Dynamics alone. Clamp *et al.* [1994] combined MD and MC into a Hybrid Monte Carlo (HMC) algorithm, and found that the HMC samples a system of Lennard-Jonesium an order of magnitude faster than MD alone. Our approach is to apply the concept of "randomized algorithms" to the MD part of the HMC, in an effort to further increase the efficiency of the algorithm, and to reduce the computational complexity.

We begin by briefly summarizing the HMC algorithm. We then show how randomization can be applied to the algorithm, and motivate its use. Next, we describe the details of the simulation, and explain how we determine its efficiency. Finally, we present the results of the simulation, and conclude the paper.

2 HYBRID MONTE CARLO

The idea of the HMC algorithm is to alternate between MD and MC steps throughout the simulation. At the beginning of each cycle, the particle velocities are redistributed according to a Gaussian distribution consistent with the system temperature. From that point, a series of MD steps are taken, leading to a new configuration. This configuration is then accepted or rejected according to standard MC criteria [Metropolis *et al.* 1953]. If rejected, the positions at the beginning of the MD run are restored. Either way, the cycle begins again, starting with the redistribution of particle velocities. Here is the pseudo-code:

```
while (simulation not finished) {
   redistribute velocities;
   calculate energy h0;
   for (length of md run) {
      for (each particle)
         calculate force from other
            particles;
      for (each particle)
         move according to calculated
            force;
   };
   calculate energy h1;
   accept with probability exp(-(h1-h0)/kT) {
      case 'accept':  keep new positions
      case 'reject':  restore old positions
                      before MD run
   };
}
```

Here, `h` is the total energy (potential + kinetic) of the system at a given moment, and `k` is the Boltzmann constant.

The velocity redistribution provides a means to escape local minima and to maintain the system at a constant temperature. The MD run allows the system to take relatively large steps in phase space, which would not be possible with MC alone. The accept/reject step ensures that the points in phase space are sampled correctly from the canonical distribution.

3 HMC WITH RANDOMIZED SELECTION

3.1 Randomized Algorithms

Over the last decade, there has been much literature concerning the use of randomized algorithms in other areas of Computer Science. These algorithms make decisions based on random number generation, and have proven to be substantially quicker than their sequential counterparts in areas such as sorting and classification. Sequential algorithms sometimes work poorly when data is distributed non-uniformly. An example would be a quicksort whose pivot is always taken to be the first item in the data subset. A presorted list takes $\bigcirc(N^2)$ time, while a randomly distributed list takes an average of $\bigcirc(N \log N)$ time. By randomly selecting the pivot for each data subset, every list, regardless of the input distribution, can be sorted in $\bigcirc(N \log N)$ time with high probability (where the high probability is over the space of outcomes of the random number generator) [Rajasekaran and Reif 1993].

We apply a similar idea to the particle selection process of MD. At each MD step, rather than determine the force on every particle, we calculate the force acting on only a subset of the particles, and then move these particles accordingly. If we randomly choose a new particle set at each step, then the general character of the MD moves should be maintained, regardless of the particle distribution.

3.2 Application to HMC

Clearly, such a randomized MD scheme may not conserve energy. This is acceptable, provided there's a mechanism in place for redistributing velocities appropriately, or in the case of the HMC, sampling correctly from phase space. Instead of taking a regular MD run using all the particles, we only move a randomly chosen subset of these particles at each step. The computational savings can be considerable, as the complexity is proportional to the size of the random subsets.

The only question which remains is whether or not the sampling efficiency can be maintained. To answer this question, we turn to simulation.

4 SIMULATION DETAILS

4.1 Setup

For purposes of comparison, we chose a system similar to that of Clamp *et al.* [1994]. Our system consisted of 100 Lennard-Jones particles, all interacting with the following potential:

$$V(r) = \begin{cases} -4\epsilon\left\{\left(\frac{\sigma}{r}\right)^6 - \left(\frac{\sigma}{r}\right)^{12}\right\}, & r \leq 2.5\sigma \\ 0, & r > 2.5\sigma \end{cases} \quad (1)$$

where σ and ϵ are constants of length and energy. Initially, the particles were arranged on a 10×10 2-D rectangular lattice, with periodic boundary conditions. In terms of reduced units of energy ($E^* \equiv E/\epsilon$), length ($r^* \equiv r/\sigma$) and time ($t^* \equiv \sqrt{\epsilon/(\sigma^2)}$) [Allen and Tildesley 1992], the simulations were run at a temperature of 2.0. The particle density was 0.83.

For each experiment, the system was then simulated with ordinary MD, using an ordinary velocity rescaling scheme, until equilibrium conditions were reached. At this point, the randomized HMC algorithm began. We used MD runs of 10 steps, and took system measurements at the end of each run. These measurements were based on the new positions and velocities if the move was accepted, and on the original positions and velocities if not. This process was repeated 20,000 times, for a total of 200,000 MD steps.

4.2 Determining Efficiency

For purposes of measurement, we considered the specific heat capacity, C_V, in the canonical ensemble,

$$C_V = \frac{<\delta E^2>}{k_B T^2},$$

where $<\delta E^2>$ is the variance of the total energy [Allen and Tildesley 1992]. Following Clamp *et al.* [1994], we calculated the specific heats for shorter runs, $C_{ave}(N)$, by averaging the values of the specific heats obtained from blocks of size N of the full run:

$$C_{ave}(N) = \frac{1}{m}\sum_{i=1}^{m} C_{block}(N(m-1), Nm),$$

where m is the integer part of $200,000/N$ and $C_{block}(N(m-1), Nm)$ is the specific heat obtained from the block of steps $N(m-1)$ to Nm.

Fig. 1 shows the typical behavior of $C_{ave}(N)$ as a function of block size. As the block size increases,

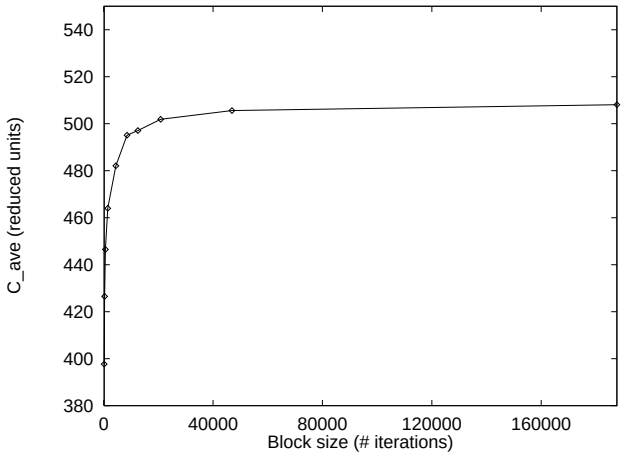

Figure 1: *The dependence of C_{ave} on the block size, using a random subset size of 10%, and $\Delta t^* = 0.008$.*

$C_{ave}(N)$ approaches a constant value. The more efficient the sampling of phase space, the smaller the block size required to reproduce the specific heat of the entire run, $C_V \equiv C_{ave}(200,000)$. We see that in Fig. 1, $C_{ave}(N) \approx C_V$ for $N \geq 22,000$.

4.3 Choosing the Timestep

It is important to choose the right timestep in the MD portion of randomized HMC. Because the MD paths are gauged by Metropolis selection, it is possible to use greater timesteps than would be stable for a pure MD simulation. If the timesteps are too small, then virtually all the MD paths are accepted, as the system moves very slowly through phase space. If the timesteps are too large, then the MD runs become unphysical, and the resulting trial states are virtually always rejected. For efficient sampling, a compromise must be found. Fig. 2 demonstrates how the timesize affects the acceptance rate.

5 RESULTS

The Lennard-Jonesium system was simulated for random subset sizes of 10%, 25%, 50%, 75% and 100%, where 100% is equivalent to no randomized selection whatsoever (i.e. the original HMC algorithm). In order to put each subset size on equal footing, the timestep leading to maximum efficiency in each case was determined, and used in the final simulation runs. Table 1 lists the timesteps used in each case, as well as the per-

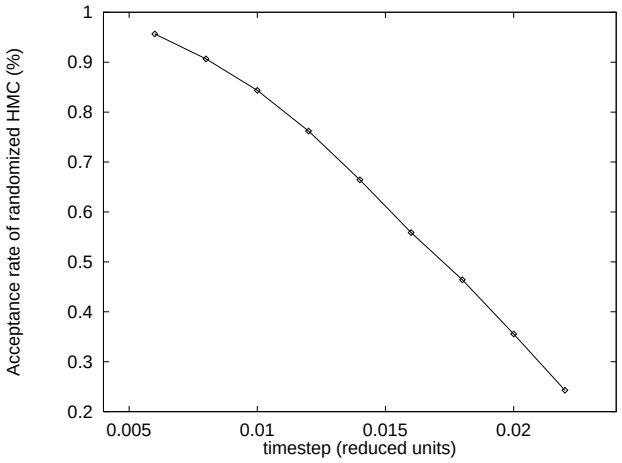

Figure 2: *The acceptance rate of randomized HMC with varying timesteps. The data was extracted from simulation runs using a random subset size of 10%.*

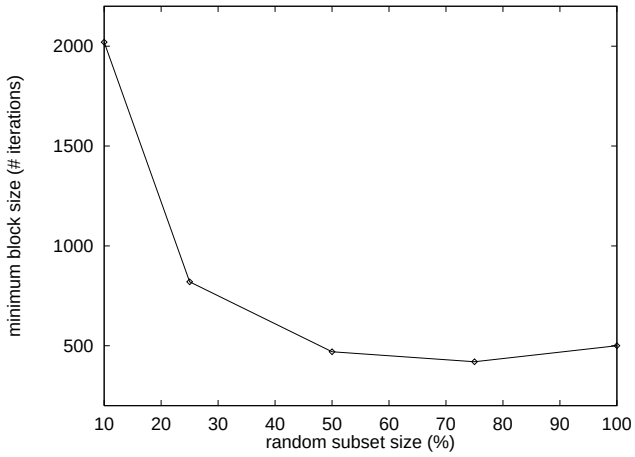

Figure 3: *The minimum block size required for convergence, as a function of the random subset size.*

subset size (%)	Δt^*	accept rate (%)	C_V^*
10	0.014	66.4	511.9
25	0.016	43.7	515.9
50	0.014	44.0	513.0
75	0.014	35.6	514.7
100	0.014	37.0	509.2

Table 1: *Timesteps, acceptance rates, and C_V^*.*

centage of MD runs which were accepted. Finally, the equilibrium values of C_V^* ($\equiv C_V/k_B$) are shown. Regardless of the random subset size, the heat capacities converged to within $\approx 1\%$.

In order to compare the efficiency of each simulation, we defined the "minimum block size" to be the smallest block size which still produced a specific heat capacity within 5% of the full run. Thus, the smaller the minimum block size, the better the sampling efficiency. The minimum block sizes are plotted in Fig. 3.

From Fig. 3, we see that the efficiency of the original HMC algorithm (random subset size = 100%) is maintained by our randomized version for a subset size as small as 50%. This translates to a computational savings of 50%, with no sacrifice in simulation accuracy. When the subset size is less than 50%, the equilibrium values of C_V are still consistent with that of 100% HMC, but the sampling of phase space becomes less efficient.

These results can be explained by considering the nature of the phase space exploration for each model. At one extreme, the original HMC (or 100%-subset randomized HMC) introduces randomness to the phase space exploration only by redistributing the velocities at the beginning of each MC cycle. From there, the MD moves through phase space in a perfectly deterministic fashion, leading to the next trial point. At the other extreme, the 10%-subset randomized HMC introduces randomness not only through the velocity redistribution, but also by grossly approximating a full (100%-subset) MD run. It appears that the optimum mixture of MC randomness and MD determinism isn't found at either extreme, but rather around the 50%-subset point, at least for the system studied here.

6 CONCLUSION

The HMC algorithm has been found by other researchers to be more efficient in exploring phase space than either MC or MD alone. The randomness of the MC element allows HMC to cross potential barriers and avoid harmonic modes, while the deterministic MD steps permit large movements through phase space.

We have extended the HMC simulation algorithm by introducing a new level of randomization to the MD moves. By randomly selecting a subset of the particles to be moved at each step, we increase the randomness of the trial states, while maintaining the character of the full MD moves.

The efficiency of the algorithm was measured by recording the specific heat capacity of the system through the length of the simulation, and determining

the minimum number of iterations required on average to approximate the specific heat capacity of the full run.

It was found that our randomized approach maintained equilibrium values of specific heat capacity to within 1% of the full HMC method. Furthermore, the efficiency of the randomized HMC was equal to that of HMC, for a random subset size as small as 50%. This represents a computational savings of 50%, with no sacrifice in simulation accuracy.

For 2-D Lennard-Jonesium, randomized selection has greatly improved the efficiency of the HMC algorithm. It remains to be seen if the randomized approach can be applied to other systems and algorithms, with similar success.

Acknowledgments

We would like to acknowledge the Engineering Research Center (ERC) for Particle Science and Technology at the University of Florida (National Science Foundation (NSF) Grant # EEC-94-02989 and the Industrial Partners of the ERC).

References

[Allen and Tildesley 1992] Allen, M.P. and D.J. Tildesley. 1992. *Computer Simulation of Liquids.* Oxford University Press, New York.

[Clamp *et al.* 1994] Clamp, M.E.; P.G. Baker; C.J. Stirling; and A. Brass. 1994. "Hybrid Monte Carlo: An efficient Algorithm for Condensed Matter Simulation." *Journal of Computational Chemistry 15*, no. 8 (Feb.): 838-846.

[Guarnieri and Still 1994] Guarnieri, Frank and W. Clark Still. 1994. "A Rapidly Convergent Simulation Method: Mixed Monte Carlo/Stochastic Dynamics." *Journal of Computational Chemistry 15*, no. 11 (Apr.): 1302-1310.

[Metropolis *et al.* 1953] Metropolis, Nicholas; Arianna W. Rosenbluth; Marshall N. Rosenbluth; Augusta H. Teller; and Edward Teller. 1953. "Equation of State Calculations by Fast Computing Machines." *Journal of Chemical Physics 21*, no. 6 (Jun.): 1087-1092.

[Neal 1993] Neal, Radford M. 1993. "Probabilistic Inference Using Markov Chain Monte Carlo Methods." Technical Report CRG-TR-93-1. Department of Computer Science, University of Toronto, Toronto, Ont. (Sep.).

[Rajasekaran and Reif 1993] Rajasekaran, S. and J.H. Reif. 1993. "Derivation of Randomized Algorithms for Sorting and Selection." In *Parallel Algorithms Derivation and Program Transformation,* Paige, Reif and Wachter, eds. Kluwer Academic Publishers, 187-205.