Microcanonical Monte Carlo Simulation

Michael Creutz
Department of Physics, Brookhaven National Laboratory, Upton, New York 11973
(Received 24 February 1983)

A new algorithm for the simulation of statistical systems is presented. The procedure produces a random walk through configurations of a constant total energy. It is computationally simple and applicable to systems of both discrete and continuous variables.

PACS numbers: 05.40.+j, 02.50.+s, 05.20.-y, 11.15.Ha

As computers have improved in capability, the simulation of large statistical systems governed by known Hamiltonians has become an important tool of the theoretical physicist. Applications range from studies of phase transitions in condensed matter to calculation of hadronic properties via lattice gauge theory. Most of these simulations rely on adaptations of the algorithm of Metropolis *et al.*[1] This generates a sequence of configurations via a Markovian process such that ultimately the probability of encountering any given configuration C in the sequence is proportional to the Boltzmann weight

$$P(C) \propto \exp[-\beta S(C)], \tag{1}$$

where $S(C)$ is the energy for a statistical mechanics problem or the action for a quantum field theory simulation. Thus one obtains a sample of configurations which dominate the partition function sum or path integral

$$Z = \sum_C \exp[-\beta S(C)]. \tag{2}$$

An alternative simulation technique is the molecular-dynamics or microcanonical method. This begins with a set of equations for a dynamical evolution which conserves the total energy. Upon numerical integration the system will flow through phase space in a hopefully ergodic manner. (Indeed, nonergodic behavior would represent a fascinating exception to the generic case.) Such a program does not explicitly depend on an inverse temperature β, which is determined dynamically by measuring, say, the average kinetic energy, which by equipartition should be $\frac{1}{2}kT$ per degree of freedom. A microcanonical procedure effectively replaces the sum in Eq. (2) with

$$Z = \sum_C \sum_P \delta[S(C) + K(P) - E], \tag{3}$$

where E is an initially determined total energy, and $K(P)$ is the kinetic energy associated with the momenta P conjugate to the coordinates C. Standard arguments relate the microcanonical and canonical approaches, with β appearing as the position of a saddle point in an integral representation of the δ function in Eq. (3). Note that the conventional microcanonical simulations make no use of random numbers, which are effectively generated by the complexity of the system. This technique has recently been applied to lattice gauge theory.[2] Dropping the $K(P)$ term in Eq. (3) gives another microcanonical formulation which was discussed in the context of continuum field theory by Strominger.[3]

In this paper I present a new simulation technique which interpolates between the Metropolis *et al.* algorithm and a microcanonical formulation. I set up a random walk through configuration space while maintaining a constraint on the total energy. An extra degree of freedom, a "demon," travels around the system, transferring energy as it changes the dynamical variables. This new variable is analogous to the kinetic energy carried by the conjugate momenta in the microcanonical formulation, although the demon is not associated with any single degree of freedom in the original system. Calling the energy carried by the demon E_D, the algorithm simulates the microcanonical sum

$$Z = \sum_C \sum_{E_D} \delta[S(C) + E_D - E]. \tag{4}$$

To keep the demon from running off with all the energy, its energy must be restricted. The simplest constraint is that E_D be a positive number, but further limitations could be useful in certain cases.

For a specific example to illustrate the method, consider the Ising model in any number of dimensions. Thus, on any site i of a d-dimensional lattice is a spin variable s_i which takes values from the set $\{1, -1\}$, and the interaction of these spins is given by the Hamiltonian

$$S = \sum_{\{i,j\}} s_i s_j, \tag{5}$$

where $\{i,j\}$ denotes the set of all nearest-neighbor pairs of sites. The initial configuration of the

system can be selected by randomizing a portion of the lattice so as to obtain the desired total energy for the simulation. The program then releases the demon, which starts with, say, zero energy. As in conventional Monte Carlo simulation, its path through the lattice is arbitrary. As two possible options, it could sequentially sweep the sites, or it could randomly jump about. Upon reaching a site, the demon attempts to flip the corresponding spin. If this lowers the spin energy, it puts the energy thus obtained into its sack, that is it makes the flip and changes

$$E_D \to E_D' = E_D + S - S', \tag{6}$$

where S' is the new value of the spin Hamiltonian and S the old. If, on the other hand, the spin energy is increased, the demon can only make the flip if it has the energy. That is, if E_D' from Eq. (6) is positive, then the change is made, otherwise the demon moves on to the next spin leaving the lattice in its previous state.

This procedure amounts to an evolution through an ensemble of states of constant total energy for the combined system of the Ising lattice and the demon. To approach a uniform distribution of microstates, this algorithm satisfies a restricted form of detailed balance; if a step makes the change $\{E_D, s\} \to \{E_D', s'\}$, then at the same stage the algorithm could take $\{E_D', s'\}$ to $\{E_D, s\}$ with equal probability.

Standard statistical mechanics arguments show that the demon's energy will become exponentially distributed,

$$P(E_D) \propto \exp(-\beta E_D), \tag{7}$$

for energies small compared with the total system energy. This enables the determination of the inverse temperature from the average value of the demon energy:

$$\beta = \tfrac{1}{4} \ln(1 + 4/\langle E_D \rangle). \tag{8}$$

For a continuous system where the energy can take any positive value, this simplifies to

$$\beta = 1/\langle E_D \rangle. \tag{9}$$

On a large system the demon's energy represents only a small fraction of the total energy, and thus one effectively works with a fixed energy per spin.

Allowing the demon to visit a heat bath between steps converts this algorithm into the conventional Metropolis *et al.* procedure. That is, before visiting any site, the demon energy would be replaced by a new value randomly selected with the Boltzmann weight $\exp(-\beta E_D)$. Note that this means that a randomly hopping demon on a large lattice will locally reduce to the Metropolis *et al.* procedure because in its wanderings on distant parts of the lattice before visiting any given site, it will establish a Boltzmann distribution for its energy.

If instead of a single demon one releases a whole battalion, then they collectively can carry an appreciable amount of energy. When the number of demons becomes large compared with the number of lattice degrees of freedom, the algorithm again reduces to that of Metropolis *et al.* With a comparable number of demons and spins, the technique can smoothly continue between the microcanonical and canonical ensembles.

The generalization to a more general statistical system is immediate. For a lattice gauge theory with a continuous group, the demon would try to change a link variable by multiplying it by a matrix randomly selected from the same type of table one might use with the Metropolis *et al.* procedure. The tentative change would then be accepted or rejected depending on whether the demon has enough energy to make it. Just as with the conventional approach, it may be advantageous to invoke several trial changes before moving to the next variable.

Figure 1 shows the results of several simulations with SU(2) lattice gauge theory on a 6^4 site lattice. For normalization conventions, see Ref. 4. In Fig. 1(a), I show the evolution of the inverse temperature as calculated from Eq. (9), averaging over sequential sweeps of the demon over the lattice sites, each point representing one sweep. The initial condition was all sites ordered and the demon possessing a huge sack of energy. The amount in the sack gives a final average plaquette $P = \langle 1 - \frac{1}{2} \operatorname{Tr} U_P \rangle = 0.4166$. On the first sweep the demon essentially randomizes those links met until its sack is nearly depleted. This sets up an initial state much like the mixed-phase starts of Ref. 5. The linear drift in the first ten iterations represents the dissolution of this energy over the entire lattice. In Fig. 1(b) the same evolution is shown for a demon hopping randomly around the lattice. In this case the initial energy is randomly distributed and thus the early drift seen in Fig. 1(a) is considerably reduced. For comparison, Fig. 1(c) shows the evolution of the average plaquette or internal energy under a conventional Metropolis *et al.* procedure at $\beta = 2.25$, the value corresponding to the above runs. This was the same program used by Bhanot and Creutz.[6] All the runs in this figure

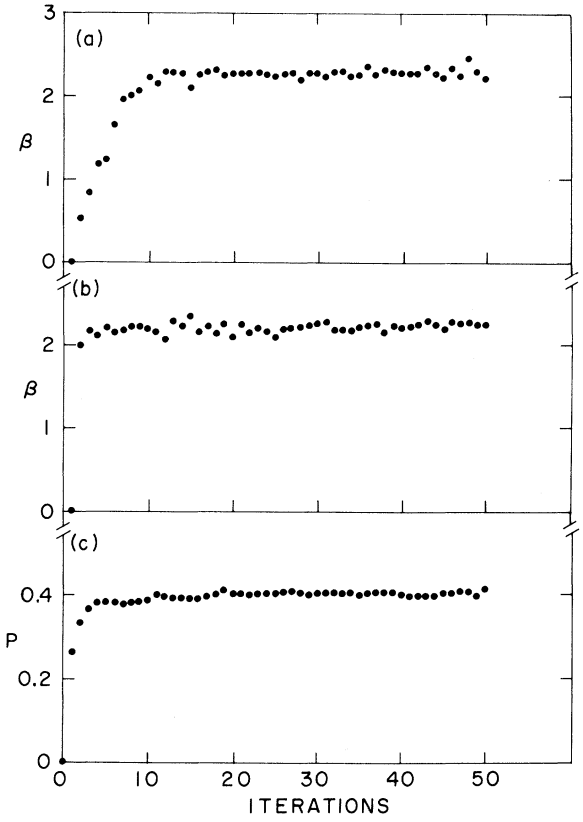

FIG. 1. Three simulations on SU(2) lattice gauge theory on a 6^4 lattice. In (a) the microcanonical demon moves sequentially through the lattice. In (b) it hops randomly from site to site. For comparison, (c) shows a conventional Monte Carlo simulation at $\beta = 2.25$.

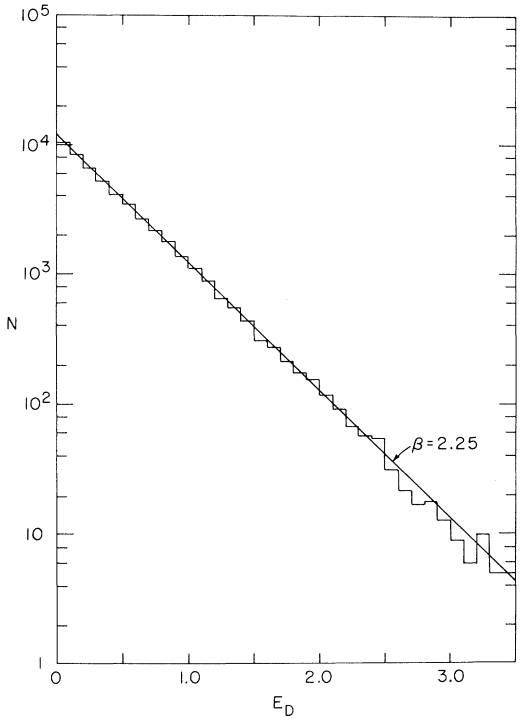

FIG. 2. The distribution of demon energies over the last 10 of the 50 iterations in Fig. 1(a). The ordinate counts the number of times out of 51 840 steps that the demon is in the corresponding energy bin of width 0.1. The straight line on this logarithmic plot is the Boltzmann distribution for $\beta = 2.25$.

used the same algorithm for trial changes and made ten hits on any link before proceeding. Figure 2 shows the distribution of demon energies over the last ten sweeps of Fig. 1(a). The straight line in this figure is the expected Boltzmann distribution for $\beta = 2.25$. Figure 3 shows the convergence of a simple correlation function, the 2×2 Wilson loop, for the three runs of Fig. 1.

This new algorithm performs comparably with the conventional Monte Carlo schemes. There may, however, be some advantages. First, the demon has no need for transcendental functions; its energy becomes automatically exponentially distributed. Unfortunately this is not much of a gain in practice because for discrete groups such functions can be put in tables, while for gauge theories so much time is spent multiplying neighbors that a few exponentials are inessential. A second advantage of the demon is that it is extremely lenient in its demands on the random number generator. A random walk in phase space will soon forget its beginning, even if it is not quite random. For the Ising example with a sequentially moving demon, no random numbers are needed at all; the lattice uses its complexity to generate its own. With a discrete group, a third advantage of this technique is that all arithmetic can be done with small integers. No floating point operations are required, and, with an upper limit on their energy, a platoon of demons could ride in one computer word via a multispin coding technique.[5,7] A final, but perhaps esoteric, advantage of the method is that it does not treat the Boltzmann weight as a probability; indeed, the demon does not even know what β is. As one of the problems with simulating fermionic systems is the lack of a probability interpretation for Grassman integrals, perhaps there is a hint here.

As with conventional Monte Carlo simulation, this algorithm has the advantage over conventional molecular-dynamics techniques of not requiring numerical integration of differential equations. Thus, at any step the change of a single variable

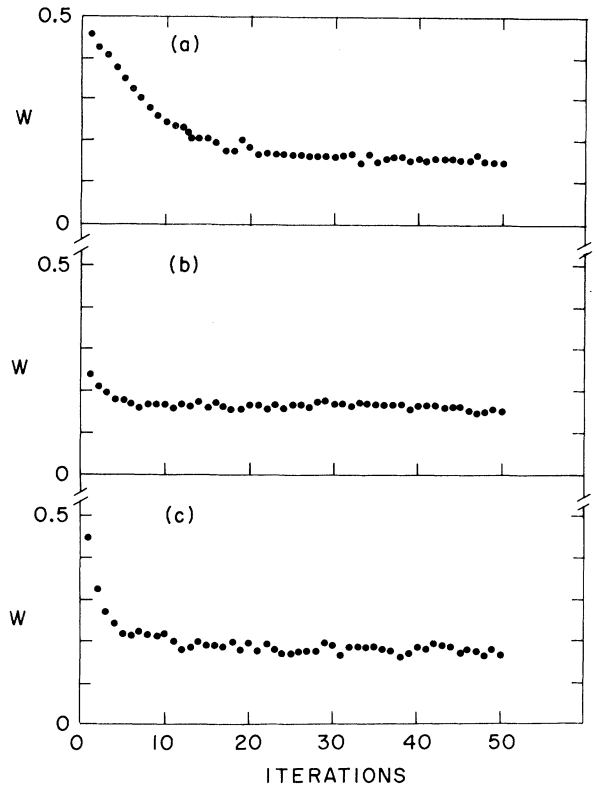

FIG. 3. The evolution of the 2×2 Wilson loop in the three runs from Fig. 1.

need not be infinitesimal. This advantage also applies over stochastic differential evolution.[8]

The approach may have a few disadvantages. First, finite-size effects differ from those in the canonical approach; on small systems trapping in metastable states is potentially more serious. Second, and related, one cannot directly use the fluctuations in the lattice energy to measure the specific heat. This problem can be circumvented by using the fluctuations in the Fourier transform of the energy density and extrapolating to zero Fourier component.

An interesting extension of the technique arises on considering more than one coupling parameter. For example, consider adding a magnetic field to the Ising model. Although the additional interaction could be added directly to the Hamiltonian, an amusing alternative would be to do a simulation at constant magnetization. In this case the demon would carry two sacks, one of energy and one of magnetization. A spin could be flipped only if the demon has enough of each. Such a program would have neither β nor the magnetic field as input parameters, but would determine them from the average contents of these sacks.

[1]N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller, J. Chem. Phys. 21, 1087 (1953).

[2]D. Callaway and A. Rahman, Phys. Rev. Lett. 49, 613 (1982).

[3]A. Strominger, to be published.

[4]M. Creutz, Phys. Rev. D 21, 2308 (1980).

[5]M. Creutz, L. Jacobs, and C. Rebbi, Phys. Rev. Lett. 42, 1390 (1980).

[6]G. Bhanot and M. Creutz, Phys. Rev. D 24, 3212 (1981).

[7]L. Jacobs and C. Rebbi, J. Comput. Phys. 41, 203 (1981).

[8]G. Parisi and Y.-S. Wu, Sci. Sin. 14, 483 (1981).