PHYSICAL REVIEW

LETTERS

VOLUME 65 20 AUGUST 1990 NUMBER 8

Cluster Dynamics for Fully Frustrated Systems

Daniel Kandel, Radel Ben-Av, and Eytan Domany

Electronics and Physics Departments, Weizmann Institute of Science, Rehovot 76100, Israel

(Received 8 June 1990)

We present a novel cluster algorithm for Monte Carlo simulations of the fully frustrated Ising model on the square lattice. The new method does not suffer from problems of metastability, and is extremely efficient even at $T=0$. Our algorithm is a special case of a more general Monte Carlo simulation scheme. The general scheme unifies many cluster algorithms that were developed recently in order to accelerate Monte Carlo simulations.

PACS numbers: 05.50.+q, 75.40.Mg

The work of Swendsen and Wang[1] (SW) on acceleration of simulations of ferromagnetic Potts models opened a new field of interest in computational physics. The improved efficiency of their cluster algorithm gave hope that similar methods may be used to accelerate simulations of other systems, for which standard techniques are very inefficient. Indeed, generalizations appeared[2-8] soon after the work of SW was published.

Clearly, one needs different cluster algorithms to accelerate simulations of different models and physical systems. A most important unsolved problem is that of simulating models with competing interactions and frustration. All known algorithms[1-8] become inefficient when competing interactions are introduced. They fail to identify the "correct" clusters, and in most cases almost all of the lattice ends up in the same cluster, leading to a trivial move. Thus, it is still very difficult, if not impossible, to perform simulations of spin glasses at low temperatures. Many optimization problems (e.g., the wiring problem in computer design,[9] the problem of finding the location of atoms in the unit cell of a crystal from x-ray scattering information,[10] etc.) fall into this class of models with frustration. Simulated annealing,[9] which is the most efficient method for such problems, also suffers from severe slowing down. The "replica" Monte Carlo algorithm[11] of SW improves simulations of the two-dimensional Ising spin glass, but is not as effective in other cases. For example, it is much less efficient than the algorithm we present here for the fully frustrated Ising model on the square lattice.

This Letter makes a first step towards solving some of the problems in the simulation of frustrated systems. We propose a novel cluster algorithm for simulating the fully frustrated Ising model on the square lattice. We show that it is extremely efficient even at $T=0$, and does not have metastable states; hence we move between ground states of the model without simulated annealing.

The paper is organized as follows: First, we describe the algorithm in detail, and compare its performance at $T=0$ to that of Metropolis *et al.*[12] We find that while typical time scales of the Metropolis algorithm diverge very strongly as a function of system size, our algorithm does not suffer from significant slowing down. Consequently, we can easily measure the dependence of the magnetic susceptibility on system size, and confirm that at $T=0$ the system behaves as a ferromagnet at criticality. As we increase L, the linear size of the system, the susceptibility χ diverges as $\chi \sim L^{2-\eta}$, where $\eta = \frac{1}{2}$, in agreement with exact results.[13] We also give an intuitive explanation for the extraordinary performance of the algorithm. Last, we demonstrate that our new method is a special case of a more general cluster Monte Carlo technique. The general scheme includes as special cases other previously developed cluster algorithms.[1-3,5,6] We show that this general method satisfies the detailed balance condition, and conclude that our algorithm is a legitimate Monte Carlo procedure.

Description of the model and the algorithm.—We

consider the fully frustrated Ising model on the square lattice with nearest-neighbor interactions and periodic boundary conditions in both directions. All horizontal bonds are ferromagnetic, while vertical bonds are ferromagnetic on even columns and antiferromagnetic on odd columns. All bonds have the same absolute strength J. The algorithm works equally well for any other arrangement of bonds such that all elementary plaquettes are frustrated (i.e., at most three of the bonds can be satisfied simultaneously). The ground state of this model is highly degenerate and its entropy at $T=0$ is finite.[14]

Single-spin-flip Metropolis simulations of this system get stuck in metastable regions of configuration space and cannot sample all ground states. Even cluster algorithms such as the SW,[1] multigrid,[4] or Wolff[15] methods for Potts models do not improve the situation. We claim that the failure of these techniques is due to the fact that they treat single bonds independently. While in systems without frustration the sign of a single bond determines whether the two spins that it connects tend to be aligned or antialigned, this is not so when competing interactions are present. For example, in the model we consider here, one has to take into account at least four bonds that form an elementary plaquette in order to see that the relative orientation of two neighboring spins is not well defined in the ground state. Thus, we propose to take into account correlations within elementary plaquettes rather than treating single bonds independently.

We now describe how our algorithm generates a transition from any given spin configurations u to u'. First, we partition the plaquettes into a checkerboard pattern, and randomly choose either the shaded or unshaded set (Fig. 1). Say we choose the shaded ones: Each bond belongs to one and only one shaded plaquette. The energy

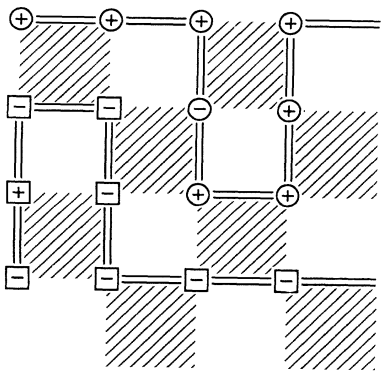

FIG. 1. A possible outcome of the clustering procedure. Here, we started from one of the ground states of a 4×4 lattice (with periodic boundary conditions), and applied the freeze-delete rules at $T=0$. Plaquettes that belong to the checkerboard partition that was chosen are shaded. Frozen bonds are denoted by double lines and deleted bonds are not marked. The system is divided into two clusters. Spins that belong to the first (second) cluster are marked by circles (squares).

E_p of a plaquette p is either $E_p=-2J$ (when three of the four bonds are satisfied and one is not), or $E_p=2J$ (when three of the bonds are unsatisfied). Next, we make an independent decision on each of the shaded plaquettes. The decisions depend on the spin configuration u through the plaquettes' energies. If $E_p=2J$, all four bonds of the plaquette p are deleted, i.e., replaced by bonds of strength $\tilde{J}=0$. If $E_p=-2J$, we decide on one of three alternatives in a probabilistic manner. We delete all four bonds with probability $P_1=e^{-4\beta J}$. With probability $P_2=(1-e^{-2\beta J})^2$ we delete the unsatisfied bond and the satisfied bond parallel to it, while the other two bonds are frozen, i.e., replaced by bonds of strength $|\tilde{J}|=\infty$ (keeping the sign of the bonds fixed). Last, with probability $1-P_1-P_2$ we randomly choose one of the two satisfied bonds which are perpendicular to the unsatisfied bond. This bond and the unsatisfied bond are deleted while the other two are frozen. A possible outcome of our procedure is shown in Fig. 1. In this example we started from a ground state of the system, and applied the rules described above at $T=0$. Frozen bonds are marked by double lines, while bonds that were deleted are not marked.

As a result of this procedure the spins are divided into clusters; two spins belong to the same cluster if there is a line of frozen bonds connecting them. In the example of Fig. 1 the lattice is divided into two clusters. Spins that belong to the first (second) cluster are marked by circles (squares). Our last step to generate a new spin configuration u' is to flip each cluster with probability $\frac{1}{2}$, as done in the SW method.[1] This completes one clustering move of the algorithm, and in order to continue the simulation we restore the original Hamiltonian and repeat the steps described above, starting from the new spin configuration we have generated.

As described here, our procedure is not ergodic at $T=0$. In order to ensure ergodicity we supplement it by single-spin-flip Metropolis sweeps. The number of Metropolis sweeps after each clustering move is a tunable parameter of the algorithm. We have not yet tested the significance of this parameter, but we believe it can only change typical time scales by a factor, which does not depend on lattice size or temperature. In the simulations that we describe below, each cycle consisted of one clustering move followed by a single Metropolis sweep.

Results and comparisons.—In order to test the efficiency of the new cluster algorithm we performed simulations of systems of linear sizes $L=4$, 8, 16, 32, 64, and 128 at low temperatures. Here, we compare results of Metropolis simulations with those obtained from simulations with our cluster algorithm. Time is measured in units of sweeps through the lattice. Since each sweep of the cluster algorithm consists of one freeze-delete cycle followed by a single Metropolis sweep, it is clear that a sweep of the cluster algorithm is more expensive than a Metropolis sweep in terms of computer time. However, this difference is not significant since

Metropolis sweeps are faster by a factor of less than 2.

As a first test we measured the magnetization per spin as a function of time for a 32×32 system at various temperatures ($\beta=0.5$, 1.0, 2.0, 3.0), and compared between Metropolis and cluster runs of 1000 sweeps each. From the time scales associated with fluctuations of the magnetization we could see that the Metropolis simulations suffer from severe slowing down. In fact, for $\beta>1.0$ the system was trapped in metastable regions of configuration space and did not equilibrate at all. The cluster simulations, on the other hand, were not affected by the decrease in temperature. The results indicated that even at $\beta=3.0$ typical time scales of the new algorithm are much smaller than 10 sweeps. Thus, our method is unique in the sense that it does not suffer from the problem of metastable states.

Now we turn to more accurate measurements of autocorrelation times. We simulated lattices of different sizes at very low temperatures ($\beta=10$) and measured the time-delayed autocorrelation function[16] of the square of the magnetization:

$$C(t)=\frac{\langle M^2(t_0)M^2(t_0+t)\rangle-\langle M^2\rangle^2}{\langle M^4\rangle-\langle M^2\rangle^2}, \qquad (1)$$

where M is the magnetization of the system. From the exponential decay of $C(t)$ at long times we can deduce the exponential autocorrelation time τ. For each system size we performed 25000 sweeps with each algorithm and discarded the first 5000. From the remaining 20000 we calculated the autocorrelation function $C(t)$, and the results for $L=64$ are plotted in Fig. 2. Autocorrelation times of the Metropolis algorithm are so large that for lattices with $L\geq32$ we see only transients; our Metropolis simulations are not extensive enough to measure the asymptotic decay of $C(t)$ at long times. The cluster algorithm, on the other hand, is extremely efficient. Its autocorrelation times barely depend on system size, and are very small (of the order of 2 or 3 sweeps). Within our precision we can hardly see any slowing down as we increase the size of the lattice. Results of more extensive simulations, with more precise statements concerning autocorrelation times and slowing down in both algorithms, will be published elsewhere.[17]

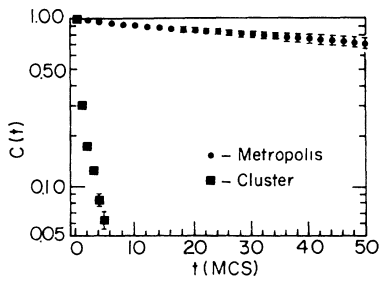

FIG. 2. Log-log plots of time-delayed autocorrelation functions of the square of the magnetization (see text for definition) as a function of time, measured in Monte Carlo sweeps (MCS). Metropolis and cluster simulations of a 64×64 lattice are compared.

Because of the extraordinary efficiency of our method we can easily perform measurements of the magnetic susceptibility per spin $\chi=\langle M^2\rangle/L^2$. At $T=0$ we find[17] $\chi\sim L^{2-\eta}$, with $\eta=0.507\pm0.009$. Hence our model has a (ferromagnetic) critical point at $T=0$. By measuring the fourth-order cumulant of the magnetization we also verified that hyperscaling is obeyed. These results are consistent with exact results of Forgacs.[13]

Why does it work?—Simulation methods, such as single-spin-flip Metropolis and straightforward application of standard cluster algorithms,[1,15] all fail to equilibrate the fully frustrated Ising model at low temperatures. The Metropolis algorithm fails because many spins are frozen in the typical ground state (i.e., cannot be flipped). Moreover, to ensure that the system will not get trapped in metastable regions of configuration space, one has to allow for large-scale moves. Standard cluster algorithms, on the other hand, freeze nearly the entire lattice into one cluster, and generate only trivial large-scale moves. Therefore, it is clear that in order to devise an efficient simulation method for this model we must look for an algorithm that does not freeze the whole lattice into a single cluster, even when the system is in a ground state at $T=0$. Indeed we can show that our procedure does not suffer from this deficiency. In particular, we proved that when the system is in a ground state at $T=0$, our algorithm divides the lattice into several clusters. Moreover, there are at least two large clusters, each consisting of at least L spins, where L is the linear size of the system. We give a detailed proof of these statements in Ref. 17.

The reason that our method succeeds where other techniques fail can be traced to the fact that standard cluster algorithms "work" on *bonds*. In all ground states of our model every plaquette contains three satisfied bonds while the fourth bond is unsatisfied. When standard clustering methods are applied, even though the unsatisfied bond is always deleted, the two spins that it connects do end up in the same cluster at $T=0$. This is due to the fact that the three satisfied bonds are frozen (at $T=0$, a satisfied bond is always frozen). Thus, unsatisfied bonds are effectively frozen, and a single cluster is generated. We realized that in order to overcome this problem we must break plaquettes, and hence cannot use single bonds as the basic interaction (to be frozen or deleted). Note that breaking plaquettes is a necessary but not sufficient condition to ensure that the lattice is divided into several large clusters. Plaquettes may become frozen indirectly, in the same way that unsatisfied bonds are effectively frozen by standard cluster algorithms. We examined a number of freezing schemes that do break plaquettes, but only for the scheme described earlier were we able to prove that the lattice is indeed divided into several large clusters. We conclude that our

algorithm, when acting on any ground state, performs nontrivial large-scale moves. As demonstrated above, our procedure, when supplemented by single-spin-flip steps, allows the system to move freely between all ground states, which is an essential property for accelerating equilibration.

We turn now to show that our algorithm is a legitimate Monte Carlo procedure, i.e., satisfies the detailed balance condition. To that end we describe below a general legitimate simulation procedure, and then show that our algorithm for the fully frustrated Ising model is a special case of this general scheme.

A general cluster Monte Carlo scheme.—The question of generalizations of cluster algorithms is of importance for the future development of efficient simulation techniques. We present a general scheme that satisfies the detailed balance condition, and may therefore serve as a useful tool to devise new algorithms.

Consider a model Hamiltonian $\mathcal{H}$ that can be written as a sum of the form $\mathcal{H}=\sum_l V_l$. For example, one can write the nearest-neighbor Ising Hamiltonian on the square lattice as a sum over bonds or as a sum over elementary plaquettes. The first step of our general Monte Carlo scheme is to make independent decisions on each of the terms V_l of the Hamiltonian. The decision may depend on the configuration of the system u. As a result of this decision the term $V_l(u)$ is replaced by $\tilde{V}_l^i(u)$ with probability $P_l^i(u)$, where $i=1,\ldots,n$. The n possibilities are generalizations of the freeze-delete operations which are used in many cluster algorithms and their probabilities are normalized such that $\sum_i P_l^i(u)=1$ for any term l and configuration u. Next, we perform a simulation of the model using any procedure that satisfies the detailed balance condition with respect to the new Hamiltonian $\tilde{\mathcal{H}}=\sum_l \tilde{V}_l^i$, and generate a new configuration of the system u'. These two steps define a cycle of the procedure, and after completing the cycle we restore the original Hamiltonian and repeat the procedure starting from the new configuration u'. It is easy to show (and will be shown explicitly elsewhere[17]) that if we choose $\tilde{V}_l^i(u)$ and $P_l^i(u)$ such that

$$\beta\tilde{V}_l^i(u)=\beta V_l(u)-\ln[P_l^i(u)]\,, \tag{2}$$

our procedure satisfies the detailed balance condition with respect to the Hamiltonian $\mathcal{H}$.

As an example for the general procedure, consider the algorithm for the fully frustrated Ising model that was described above. We divided the Hamiltonian into elementary plaquettes. Next, the energy of each plaquette was changed by a freeze-delete decision. We either deleted two of the four bonds and froze the other two (six possibilities) or deleted all four bonds (one possibility), and therefore $n=7$. It is straightforward to see (and will be shown in detail in Ref. 17) that the new bonds $\tilde{J}$ constitute a Hamiltonian that satisfies Eq. (2), and therefore our algorithm is a legitimate Monte Carlo procedure.

Other previously developed cluster Monte Carlo algorithms are also special cases[17] of the general scheme described here. This is obviously the case for the SW algorithm[1] and for the ϕ^4 and O(n) algorithms.[5,6] The Niedermayer[2] and Edwards and Sokal[3] generalizations can be described in terms of our general scheme as well.

To summarize, we have devised an extremely efficient cluster Monte Carlo algorithm for the fully frustrated Ising model on the square lattice. In order to develop the new method, we introduced some new ideas that may be useful for acceleration of simulations of frustrated systems in general. We have also presented a general Monte Carlo scheme that satisfies the detailed balance condition. It includes most previously developed cluster algorithms as special cases, and may serve as a general framework for developing new cluster acceleration techniques.

This work was supported in part by the U.S.-Israel Binational Science Foundation (BSF).

[1]R. H. Swendsen and J. S. Wang, Phys. Rev. Lett. **58**, 86 (1987).

[2]F. Niedermayer, Phys. Rev. Lett. **61**, 2026 (1988).

[3]R. G. Edwards and A. D. Sokal, Phys. Rev. D **38**, 2009 (1988).

[4]D. Kandel, E. Domany, D. Ron, A. Brandt, and E. Loh, Phys. Rev. Lett. **60**, 1591 (1988); D. Kandel, E. Domany, and A. Brandt, Phys. Rev. B **40**, 330 (1989).

[5]R. C. Brower and P. Tamayo, Phys. Rev. Lett. **62**, 1087 (1989).

[6]U. Wolff, Phys. Rev. Lett. **62**, 361 (1989); Nucl. Phys. **B322**, 759 (1989).

[7]J. S. Wang, R. H. Swendsen, and R. Kotecky, Phys. Rev. Lett. **63**, 5303 (1989).

[8]R. Ben-Av, D. Kandel, E. Katzenelson, P. G. Lauers, and S. Solomon, J. Stat. Phys. **58**, 125 (1990).

[9]S. Kirkpatrick, C. D. Gellat, and M. P. Vecchi, Science **220**, 671 (1983).

[10]R. Bruinsma, Phys. Rev. Lett. **61**, 1966 (1988).

[11]R. H. Swendsen and J. S. Wang, Phys. Rev. Lett. **57**, 1607 (1986); J. S. Wang and R. H. Swendsen, Phys. Rev. B **38**, 4840 (1988).

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

[13]G. Forgacs, Phys. Rev. B **22**, 4473 (1980).

[14]See, for example, J. Villain, J. Phys. C **10**, 1717 (1977).

[15]U. Wolff, Phys. Lett. B **228**, 379 (1989).

[16]For standard methods of statistical time-series analysis see, for example, T. W. Anderson, *The Statistical Analysis of Time Series* (Wiley, New York, 1971); M. B. Priestley, *Spectral Analysis and Time Series* (Academic, London, 1981).

[17]D. Kandel, R. Ben-Av, and E. Domany (unpublished).