PHYSICAL REVIEW E

STATISTICAL PHYSICS, PLASMAS, FLUIDS, AND RELATED INTERDISCIPLINARY TOPICS

THIRD SERIES, VOLUME 53, NUMBER 5 PART A **MAY 1996**

RAPID COMMUNICATIONS

The Rapid Communications section is intended for the accelerated publication of important new results. Since manuscripts submitted to this section are given priority treatment both in the editorial office and in production, authors should explain in their submittal letter why the work justifies this special handling. A Rapid Communication should be no longer than 4 printed pages and must be accompanied by an abstract. Page proofs are sent to authors.

Fixed-cluster acceleration algorithm for spin systems

Tanniemola B. Liverpool[1] and Sharon C. Glotzer[2]

[1]*Institut für Theoretische Physik, Universität zu Köln, D-50937 Köln, Germany*

[2]*Materials Theory Center, National Institute of Standards and Technology, Gaithersburg, Maryland 20899*

(Received 6 December 1995)

A recently introduced cluster acceleration algorithm for spin models [J. Machta *et al.*, Phys. Rev. Lett. **75**, 2792 (1995)] based on invasion percolation is generalized to temperatures away from the critical point using concepts of random bond percolation. The generalized algorithm is equivalent to an ''optimized'' version of the Swendsen-Wang cluster algorithm [R. H. Swendsen and J. S. Wang, Phys. Rev. Lett. **58**, 86 (1987)], and we demonstrate its success and speed at all temperatures for the $d=2$ Ising model. We argue that the previously discussed connection to self-organized criticality may be viewed as one limit of the fixed-cluster algorithm. [S1063-651X(96)50305-3]

PACS number(s): 05.50.+q, 75.40.Mg, 75.10.Hk

Recently, a remarkably simple cluster algorithm for generating equilibrium spin configurations at critical points was introduced by Machta and co-workers [1]. This development in acceleration algorithms for unfrustrated spin systems is exciting because the method can be used to sample the critical region without knowing the critical temperature, and also because the method is significantly faster than the celebrated and closely related Swendsen-Wang (SW) [2] and Wolff [3] cluster algorithms. The recipe is simple: Start with *any* configuration of spins, and throw ''bonds'' with probability one between nearest-neighbor pairs of parallel (satisfied) spins anywhere on the lattice until a spanning cluster is formed. Then flip each cluster of bonded spins with probability 1/2, and repeat. After very few iterations, a spin configuration characteristic in every respect to one in equilibrium at the critical point is produced. This method—which may also be performed using invasion percolation to identify the clusters—was thus said to exhibit self-organized criticality.

In this paper, we generalize this ''invaded cluster'' (IC) [1] algorithm to generate equilibrium spin configurations at *any* temperature, not just the critical temperature T_c. This is achieved using a variation of random bond percolation, instead of invasion percolation [4], to identify the clusters, and flipping the clusters with probability 1/2 after a cluster of length ℓ is formed, rather than when a spanning cluster is formed [5]. We refer to the generalized algorithm as the fixed-cluster (FC) algorithm. For the purposes of illustration, we consider here the FC algorithm as applied to the two-dimensional Ising model in zero magnetic field.

Because the first paper on this new algorithm has just recently appeared [1], we begin by confirming that the prescription of Machta *et al.* indeed generates an equilibrated

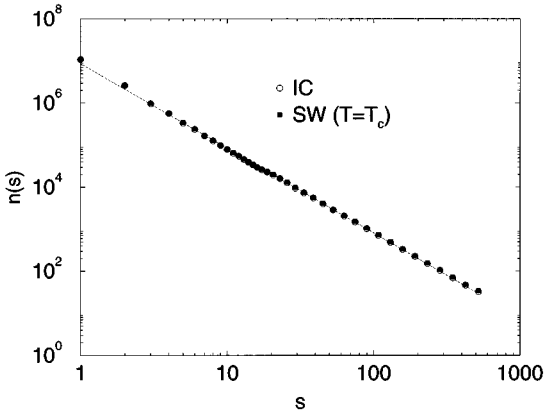

FIG. 1. Cluster size distribution calculated on lattices equilibrated with the invaded cluster (IC) and Swendson-Wang (SW) algorithms. The slope of the line is $\tau=2.04$.

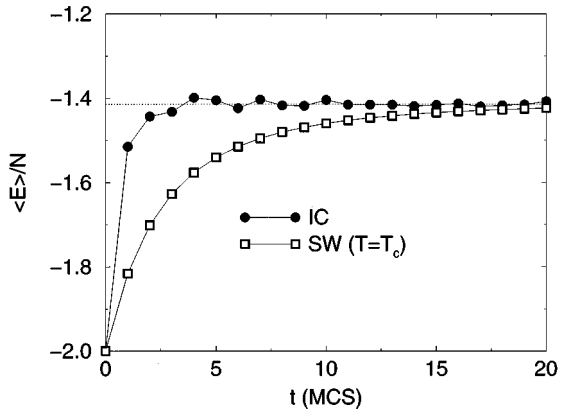

FIG. 2. Average energy per spin for IC and SW starting from all spins up for the first 20 iterations.

lattice at T_c, and is faster than SW. Figure 1 shows the number of clusters $n(s)$ at T_c containing s sites plotted double logarithmically vs s, and averaged over 1000 independent 500×500 square lattices simulated with the IC algorithm. To calculate the equilibrium cluster size distribution, clusters are identified as nearest-neighbor pairs of parallel spins connected [6,7] by bonds with probability $p_b=1-e^{-2J/k_BT}$, where J is the coupling strength. The slope of the line gives the cluster size distribution exponent $\tau=2.04$, in very good agreement with the accepted value $\tau=2.07$ when finite size effects are considered. Figure 1 also shows the cluster size distribution averaged over 1000 independent 500×500 lattices simulated with the SW algorithm at T_c; we see that the distributions obtained using the two algorithms are indistinguishable. Figure 2 shows the increase in energy E as the system equilibrates from an initial configuration of all spins up ($E=-2.0$) to the critical point ($E=-1.41$), calculated for both the IC and SW algorithms on 10 independent 4000×4000 lattices; we see that the SW algorithm requires over 10 times more iterations than the IC algorithm to reach the equilibrium energy. We also calculate the equilibrium energy and magnetization autocorrelation functions $\langle[E(t)-\bar{E}][E(0)-\bar{E}]\rangle/\,\mathrm{var}(E)$ and $\langle[|M(t)|-|\bar{M}|][|M(0)|-|\bar{M}|]\rangle/\,\mathrm{var}(M)$ for both algorithms at T_c, and find as Machta *et al.* did that the correlations decay to zero within two iterations for the IC algorithm, while the SW algorithm requires at least 20 iterations for the correlations to decay to zero on the same size lattice. Thus we confirm the success of the algorithm presented by Machta *et al.* in generating equilibrium spin configurations at the critical point in the $d=2$ Ising model.

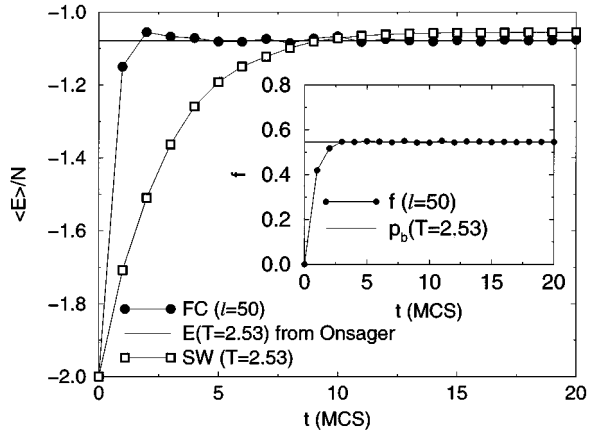

FIG. 3. Average energy per spin starting from all spins up for the first 20 iterations for FC at $\ell=50$ and SW at $T=2.53$. Inset: Fraction of satisfied bonds starting from all spins up for FC at $\ell=50$.

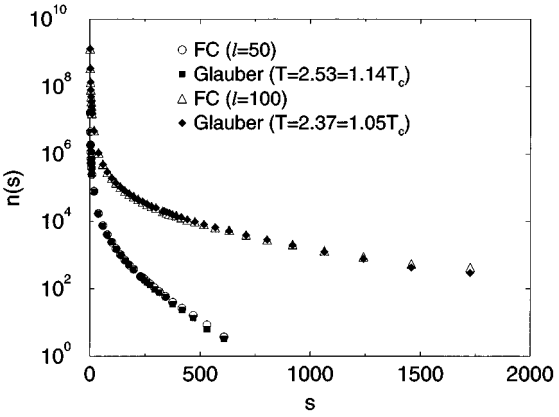

FIG. 4. Cluster size distribution $n(s)$ plotted against s for FC at $\ell=50$, for simple Glauber at $T=2.5331$ and for FC at $\ell=100$, Glauber at $T=2.3726$. Both sets of data are generated from 1000 independent 500×500 square lattices.

We generalize the IC algorithm to generate equilibrium spin configurations at $T\neq T_c$ by performing the following procedure: Starting with all spins up (although any initial configuration will work), throw bonds between nearest-neighbor parallel (satisfied) spins anywhere on the lattice with probability one. When any cluster of bonded spins reaches a prechosen length ℓ (where length is defined as the maximum linear extension), flip each cluster with probability 1/2, and repeat. Figure 3 shows the energy E and the ratio f of inserted bonds/satisfied pairs after each iteration, averaged over 100 500×500 lattices simulated with the FC algorithm with $\ell=50$. After three iterations, E and f reach their equilibrium values $E=-1.078$ and $f=0.546$.

To confirm that the configuration is indeed in equilibrium, and to determine the temperature of the lattice [8], we calculate T from the energy using Onsager's formula [9] for $E(T)$, and then independently equilibrate a new lattice using the Glauber spin-flip algorithm [10] at that temperature. The cluster size distribution $n(s)$ for both the FC-equilibrated lattices with $\ell=50$ and 100 and the Glauber-equilibrated lattices at the corresponding temperatures $T=2.5331$ and 2.3726 are shown in Fig. 4; we see that at both temperatures the cluster distributions generated by the two different algorithms are indistinguishable.

Figure 5 shows the mean cluster size $\langle s\rangle$ vs $(T-T_c)$ plotted double-logarithmically for the 500×500 lattices simulated with the FC algorithm for various ℓ. Here the mean cluster size, which scales in the same way as the susceptibility for $T\to T_c$, is defined as $\langle s\rangle=[\Sigma_s s^2 n(s)]/[\Sigma_s s n(s)]$. We find the exponent γ, which describes the divergence of the mean cluster size as the critical temperature is approached, to be $\gamma=1.73$, in excellent agreement with the exact theoretical value of $\gamma=1.75$.

To elucidate the reason why the FC algorithm works, we list in Table I the equilibrium values of f and E obtained for

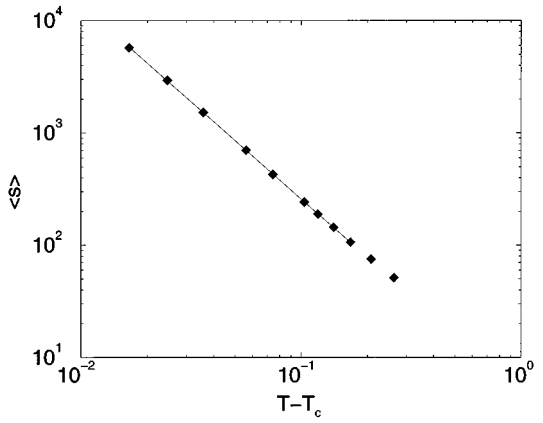

FIG. 5. Mean cluster size $\langle s \rangle$ plotted against $|T-T_c|$ for FC. The slope of the line is $\gamma = 1.73$.

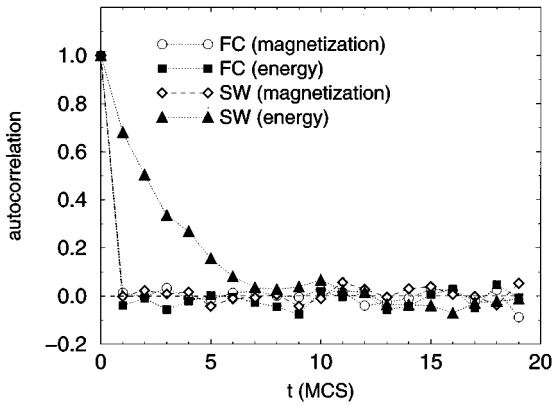

FIG. 6. Equilibrium energy and magnetization autocorrelation functions for FC at $\ell = 50$ and SW at $T = 2.5331$.

each value of ℓ, the corresponding temperature of the lattice calculated from the Onsager relation, and the corresponding bond probability $p_b = 1 - e^{-2J/kT}$ used in the SW algorithm. Recall that in the SW algorithm, bonds are thrown with probability p_b between nearest neighbor pairs of parallel (satisfied) spins, and clusters of spins connected by bonds are flipped with probability 1/2. We see from Table I that for each choice of fixed cluster length, $f = p_b$. This is why the FC algorithm works: once the system is in equilibrium, flipping the clusters after a cluster of length ℓ is grown is equivalent to flipping the clusters after a fraction f of bonds has been thrown, and this is equivalent to the SW prescription of flipping the clusters after throwing bonds between every satisfied pair of spins with probability $p_b = f$. That is, each iteration of the FC algorithm at fixed cluster length ℓ is essentially equivalent to one iteration of the SW algorithm with $p_b(T) = f(\ell)$ [11].

However, the FC algorithm guarantees that at each iteration the maximum cluster length present is *characteristic of the cluster distribution at the desired state point*. For example, for ℓ equal to the system size L the FC algorithm guarantees by definition that a spanning cluster is always present before the clusters are flipped. Since a spanning cluster is typical of the critical point, and since $f(\ell = L) = p_b(T_c)$, the algorithm is equivalent to the SW algorithm *with configurations not containing an incipient infinite cluster eliminated*. Thus, at any T (or equivalently for any ℓ), the FC algorithm is equivalent to an ''optimized'' SW algorithm—that is, SW with the maximum cluster length fixed to be representative of the desired cluster distribution. Consequently, since the SW algorithm generates configurations in the grand canonical ensemble, the FC algorithm generates only a subset of this ensemble.

This reasoning also explains the relative speed of the FC algorithm at all temperatures in generating successive decorrelated equilibrium configurations, as compared to SW (cf. Fig. 6). Note that the relative efficiency of the FC algorithm will be greatest close to the critical point, for large system sizes, and for increasing dimension, since there the effect of the guaranteed existence of the largest cluster will be most significant. Figure 7 shows the probability P_∞ of an infinite cluster vs the bond fraction f. As the lattice size approaches infinity, P_∞ approaches a step function at $f = p_c$. In the SW algorithm at T_c, bonds are thrown with probability p_c. Due to fluctuations, sometimes a more-than-spanning cluster will be present, sometimes no spanning cluster will be present, and sometimes an incipient infinite cluster will be present. With the FC algorithm at $\ell = L$, however—and consequently with the IC algorithm—the incipient infinite cluster is always present.

The FC algorithm is also faster than SW in *reaching* equilibrium at any temperature, starting from any initial starting configuration. The reason is that before the system is in equilibrium, f is not equal to the SW bond probability at the desired temperature, but rather it is equal to p_b at a *different*

TABLE I. Equilibrium values of f and E for different values of ℓ, for lattice size $L = 500$.

ℓ	$f(\ell)$	$E(\ell)$	T	$p_b(T)$
50	0.547	-1.0785	2.5331	0.546
60	0.555	-1.1267	2.4767	0.554
70	0.560	-1.1651	2.4366	0.560
80	0.565	-1.1935	2.4094	0.564
90	0.568	-1.2173	2.3883	0.567
100	0.570	-1.2362	2.3726	0.570
125	0.575	-1.2745	2.3433	0.575
150	0.577	-1.3010	2.3252	0.577
200	0.580	-1.3339	2.3050	0.580
500	0.586	-1.4054	2.2619	0.586

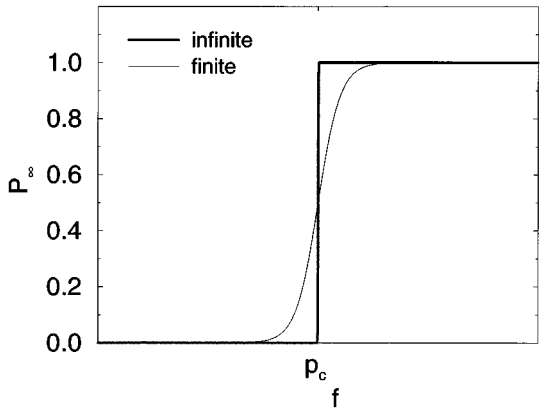

FIG. 7. Schematic plot of the probability P_∞ of finding a spanning cluster vs bond fraction f for a finite and an infinite lattice.

TABLE II. f, E, and M for the first 10 iterations of one 4000×4000 lattice with $\ell=L$ (corresponding to $T=T_c$), starting from all spins up. Also listed are the effective temperature after each iteration, and the ratio of T_{eff} to the target temperature.

t	$f(t)$	$E(t)$	$M(t)$	T_{eff}	T_{eff}/T_c
0	0.0000	-2.0000	1.0000	0.00	0.00
1	0.4995	-1.5148	-0.0717	2.89	1.27
2	0.5693	-1.4432	0.1027	2.37	1.04
3	0.5825	-1.4321	-0.0278	2.29	1.01
4	0.5752	-1.3986	-0.1261	2.34	1.03
5	0.5849	-1.4047	-0.1144	2.27	1.00
6	0.5932	-1.4235	0.0363	2.22	0.98
7	0.5782	-1.4031	-0.0108	2.32	1.02
8	0.5888	-1.4168	-0.0604	2.25	0.99
9	0.5835	-1.4178	-0.0502	2.28	1.00
10	0.5806	-1.4043	-0.0371	2.30	1.01

TABLE III. f, E, and M for the first 10 iterations of one 500×500 lattice with $\ell=50$ (corresponding to $T=2.5331$), starting from all spins up. Also listed are the effective temperature after each iteration, and the ratio of T_{eff} to the target temperature.

t	$f(t)$	$E(t)$	$M(t)$	T_{eff}	T_{eff}/T
0	0.0000	-2.0000	1.0000	0.000	0.000
1	0.4191	-1.1496	-0.1139	3.682	1.454
2	0.5187	-1.0545	0.0114	2.735	1.079
3	0.5470	-1.0673	-0.0023	2.526	0.997
4	0.5465	-1.0709	-0.0010	2.529	0.998
5	0.5497	-1.0806	0.0027	2.507	0.990
6	0.5473	-1.0808	0.0000	2.524	0.996
7	0.5443	-1.0735	-0.0014	2.545	1.005
8	0.5508	-1.0849	-0.0009	2.499	0.986
9	0.5428	-1.0723	0.0003	2.555	1.008
10	0.5435	-1.0664	0.0002	2.550	1.007

temperature. Table II shows f after each iteration of the FC algorithm for ℓ equal to the system size, starting from an initial configuration of all spins up. At the first iteration, $f(t=1)=0.499$ for this run. Had we used the SW algorithm, a fraction $f(t=1)=p_b(T_c)=0.586$ of bonds would have been thrown at the first iteration. Thus we can consider the first iteration of the FC algorithm as an iteration using SW with $p_b=0.499$, or equivalently with an effective temperature $T_{eff}=2.89$ as calculated from $p_b=1-e^{-2J/k_BT}$, which is *greater* than T_c. At the second iteration of the FC algorithm, $f(t=2)=0.569$, which corresponds to $T_{eff}=2.37$. Thus each successive iteration sets the temperature to be closer to but slightly higher than T_c, thereby heating the system faster. At each iteration, f increases, and at the sixth iteration for this run, f exceeds p_b, thus cooling the system to a temperature slightly below T_c. At the next iteration, a fraction $f<p_b(T_c)$ is thrown, ''warming'' the system back to T_c. This ''negative feedback'' mechanism described also by Machta *et al.—combined with fixing the maximum cluster length to be representative of the desired cluster distribution*—is responsible for the speed of the FC and IC algorithms. The same argument applies to finite ℓ, i.e., $T\neq T_c$, simulations (cf. Table III). Thus we see that the FC algorithm approaches equilibrium in a manner similar to that achieved through an optimized simulated annealing process, whereas SW approaches equilibrium in a manner similar to that achieved via quenching.

The FC algorithm uses a variation of random bond percolation to identify clusters of bonded spins. For finite lattices, a spanning cluster generated using either concepts of random bond percolation or invasion percolation will be statistically identical. However, random bond percolation is more naturally generalized to create maximum cluster sizes less than spanning, and thus to bond fractions away from p_c, and consequently the FC algorithm can be straightforwardly generalized to temperatures away from T_c. Thus while the IC algorithm was originally based on invasion percolation and introduced to sample the critical region, we have shown that the algorithm can be generalized to produce equilibrium spin configurations at *any* temperature—and thus sample regions of phase space away from the critical region—simply by choosing the maximum cluster length to be a finite value ℓ. Thus, tuning ℓ allows one to tune the final temperature of the lattice. From this point of view, the connection between the IC algorithm and self-organized criticality can be viewed as simply the $\ell\to\infty$ limit of the FC algorithm. From the point of view of invasion percolation, however, the association with self-organized criticality is a natural one.

S.C.G. acknowledges stimulating conversations with J. Machta and R.H. Swendsen. We thank J. Machta for sending us a copy of unpublished work, and A. Coniglio, S.A. Langer, P.H. Poole, and D. Stauffer for helpful comments on the manuscript. T.B.L. is supported by SFB 341.

[1] J. Machta, Y.S. Choi, A. Lucke, T. Schweizer, and L.V. Chayes, Phys. Rev. Lett. **75**, 2792 (1995).
[2] R.H. Swendsen and J.S. Wang, Phys. Rev. Lett. **58**, 86 (1987).
[3] U. Wolff, Phys. Rev. Lett. **62**, 361 (1989).
[4] D. Stauffer and A. Aharony, *Introduction to Percolation Theory*, 2nd ed. (Taylor and Francis, London, 1994).
[5] Equivalently, we may also use a criterion based on the typical cluster length (which scales as the correlation length), or some other aspect of the cluster distribution; this will be explored in a follow-up paper.
[6] A. Coniglio and W. Klein, J. Phys. A **13**, 2775 (1980).
[7] C.M. Fortuin and P.M. Kasteleyn, Physica (Utrecht) **57**, 536 (1972).
[8] The temperature may also be calculated by equating f a priori with the SW bond probability p_b. Using the Onsager formula simply gives an independent confirmation of T.
[9] B.M. McCoy and T.T. Wu, *The Two Dimensional Ising Model* (Harvard, Cambridge, 1973).
[10] *Applications of the Monte Carlo Method in Statistical Physics*, edited by K. Binder (Springer-Verlag, Berlin, 1987).
[11] Note that the choice of ℓ required to achieve a *particular* temperature will of course depend on the lattice size.