Monte Carlo Simulations: Hidden Errors from "Good" Random Number Generators

Alan M. Ferrenberg and D. P. Landau
Center for Simulational Physics, The University of Georgia, Athens, Georgia 30602

Y. Joanna Wong
IBM Corporation, Supercomputing Systems, Kingston, New York 12401
(Received 29 July 1992)

The Wolff algorithm is now accepted as the best cluster-flipping Monte Carlo algorithm for beating "critical slowing down." We show how this method can yield *incorrect* answers due to subtle correlations in "high quality" random number generators.

PACS numbers: 75.40.Mg, 05.70.Jk, 64.60.Fr

The explosive growth in the use of Monte Carlo simulations in diverse areas of physics has prompted extensive investigation of new methods and of the reliability of both old and new techniques. Monte Carlo simulations are subject to both statistical and systematic errors from multiple sources, some of which are well understood [1]. It has long been known that poor quality random number generation can lead to systematic errors in Monte Carlo simulation [2,3]; in fact, early problems with congruential generators led to the development of improved methods for producing pseudorandom numbers. One of these, the (Tausworthe) shift-register method, has been well tested and used extensively for nearly two decades [4]. Even better generators, from the same generalized "family" of algorithms, have been discovered and their properties have been carefully examined using a battery of mathematical tests [5]. Although the "quality" of a sequence of random numbers is notoriously difficult to assess, all indications have been that any residual errors from random number generation were now smaller than statistical errors in Monte Carlo studies. One result of this situation is that there have been a series of investigations of critical phenomena by Monte Carlo and Monte Carlo renormalization-group methods which yielded results with heretofore unprecedented precision [6].

Although the Metropolis single-spin-flip Monte Carlo method has been widely used for a broad range of simulations [7], the limiting feature of these studies has been the presence of long time correlations (critical slowing down) which develop as the critical temperature is approached. However, new cluster-flipping algorithms have been shown to dramatically reduce critical slowing down [8]. The best of these methods, the Wolff algorithm [9], generates large clusters on a lattice by connecting bonds from the starting point to nearest neighbors with the same spin with a probability

$$p = 1 - \exp(-2J/k_B T),$$

where J is the energy of a bond and T is the temperature. This procedure is repeated for neighbors of the sites connected to the starting point, etc. Simulations have been performed on a number of Potts models, including the Ising model, to study the time correlations, but so far there has been no careful study of the accuracy of the thermodynamic properties which are extracted from the configurations generated by this process.

Using an IBM RISC/6000 workstation, we have carried out extensive simulations on $L \times L$ Ising square lattices with periodic boundary conditions (for which exact results are known [10]) using several different combinations of algorithms and random number generators. We have implemented the Wolff algorithm [9] using a recursive cluster growth routine written in C. Additional simulations were performed using the Swendsen-Wang [11] algorithm as well as a standard single-spin-flip Metropolis algorithm with sequential updating. Several different random number generators were used in these programs:

(i) A 32-bit linear congruential algorithm (CONG)

$$X_n = (16807 X_{n-1}) \bmod (2^{31} - 1).$$

(ii) Two different shift register algorithms (R250 and R1279)

$$X_n = X_{n-103} .\mathrm{XOR}. X_{n-250},$$

$$X_n = X_{n-1063} .\mathrm{XOR}. X_{n-1279},$$

where .XOR. is the bitwise exclusive OR operator.

(iii) A subtract with carry generator [12] (SWC)

$$X_n = X_{n-22} - X_{n-43} - C,$$

$$\text{if } X_n \geq 0,\ C = 0,$$

$$\text{if } X_n < 0,\ X_n = X_n + (2^{32} - 5),\ C = 1.$$

(iv) A combined subtract with carry Weyl generator [12] (SWCW)

$$Z_n = X_{n-22} - Z_{n-43} - C,$$

$$\text{if } Z_n \geq 0,\ C = 0,$$

$$\text{if } Z_n < 0,\ Z_n = Z_n + (2^{32} - 5),\ C = 1;$$

$$Y_n = (Y_{n-1} - 362436069) \bmod 2^{32},$$

$$X_n = (Z_n - Y_n) \bmod 2^{32}.$$

In the above equations, X_n represents the random number used in the simulation. It is known that the performance of a random number generator can be adversely affected by improper initialization of its lookup table [4]. To avoid such problems, we initialized the tables for the R250, R1279, SWC, and SWCW generators by using the CONG generator to randomly set each bit of every integer in the table.

The simulations were performed on a cluster of IBM RISC/6000 model 350 workstations. Most of the simulations were performed exactly at $T=T_c$, although additional simulations were also done at $T=0.7T_c$ and $T=1.5T_c$. Between 5 and 10 runs of 10^7 updates were performed [13].

Surprisingly, we find that the use of the "high quality" generators [14] together with the Wolff algorithm produces *systematically incorrect* results. Simulations on a 16×16 Ising square lattice using R250 produce energies which are systematically too *low* and specific heats which are too *high* (see Table I). Each of the ten runs was made at the infinite lattice critical temperature and calculated averages over 10^6 MCS (Monte Carlo steps); the deviation from the exact value of the energy was over 40σ. Runs made using the SWC generator gave better results, but even these data showed noticeable systematic errors which had the opposite sign from those produced using R250. In contrast, data obtained using the simple 32-bit congruential generator CONG produced answers which were correct to within the error bars. Even use of the mixed generator SWCW did not yield results which were free of bias, although the systematic errors were much smaller (2σ for the energy and 4σ for the specific heat). Use of another shift-register random number generator, R1279, resulted in data which were in substantially better agreement with exact values than were the R250 values. These data may be contrasted to those which were obtained using the identical random number generators in conjunction with the single-spin-flip Metropolis method and the multicluster-flipping approach of Swendsen and Wang [11]. For all combinations of simulation methods and random number generators, the energy and specific heat values (shown in Table II) are correct to within a few σ of the respective simulations; except for the CONG generator with Metropolis and R250 with Swendsen and Wang, the answers agree to within 1σ.

Data obtained for using the Wolff algorithm at $T=0.7T_c$ and $T=1.5T_c$ showed some rather smaller deviations which were to a great extent within the errors; the specific heat exhibited the worst defects.

A possible explanation for these systematic effects lies

TABLE I. Values of the internal energy (top) and specific heat (bottom) for ten independent runs with $L=16$ at K_c obtained using the Wolff algorithm. The last number in each column, labeled "dev," gives the difference between the simulation value and the exact value, measured in terms of the standard deviation σ of the simulation.

	CONG	R250	R1279	SWC	SWCW
	1.453089	1.455096	1.453237	1.452321	1.453058
	1.453107	1.454697	1.452947	1.452321	1.453132
	1.452866	1.455126	1.453036	1.452097	1.453330
	1.453056	1.455011	1.452910	1.452544	1.453219
	1.453035	1.454866	1.453040	1.452366	1.452828
	1.453198	1.455054	1.453065	1.452388	1.453273
	1.453032	1.454989	1.453129	1.452444	1.453128
	1.453169	1.454988	1.453091	1.452321	1.453083
	1.452970	1.455178	1.453146	1.452306	1.453216
	1.453033	1.455162	1.452961	1.452093	1.453266
$-<E>$	1.453055	1.455017	1.453056	1.452320	1.453153
error	0.000030	0.000046	0.000032	0.000044	0.000046
dev.	-0.31σ	42.09σ	-0.27σ	-16.95σ	1.94σ
	1.499210	1.447436	1.497665	1.515966	1.497988
	1.498099	1.451072	1.498049	1.515966	1.497813
	1.498866	1.446619	1.497026	1.514664	1.496413
	1.499150	1.447657	1.498608	1.512534	1.497631
	1.499907	1.450726	1.499018	1.513009	1.499337
	1.498127	1.447349	1.497292	1.513267	1.496294
	1.498484	1.448782	1.498314	1.512298	1.496332
	1.498532	1.449522	1.498801	1.513575	1.497203
	1.499409	1.449012	1.496602	1.516258	1.498850
	1.498814	1.448098	1.497887	1.514838	1.496123
$<C>$	1.498860	1.448627	1.497926	1.514237	1.497398
error	0.000182	0.000467	0.000250	0.000473	0.000356
dev.	0.82σ	-107.16σ	-3.14σ	32.81σ	-3.68σ

TABLE II. Values of the internal energy (top) and specific heat (bottom) for $L=16$ at K_c obtained using different random number generators together with Metropolis and Swendsen-Wang algorithms. The values labeled "dev" show the difference between the simulation results and the exact values in terms of standard deviations σ of the simulations.

	METROP CONG	SW CONG	METROP R250	SW R250	METROP SWC	SW SWC
$-\langle E\rangle$	1.452783	1.453019	1.453150	1.452988	1.453051	1.45323
error	0.000021	0.000053	0.000053	0.000056	0.000080	0.00004
dev.	-13.25σ	-0.86σ	1.62σ	-1.36σ	-0.17σ	4.16σ
$\langle C\rangle$	1.497925	1.498816	1.498742	1.496603	1.498794	1.49986
error	0.000179	0.000338	0.000511	0.000326	0.000430	0.00043
dev.	-4.40σ	0.31σ	0.06σ	-6.47σ	0.19σ	2.65σ

in subtle correlations in the random number sequence which affect the Wolff algorithm in a special way. Sequences of random numbers may appear in which the high order bits are zero. In such cases these bits may remain zero in newly generated random numbers. Since the Wolff algorithm compares random numbers with only a single bond probability, this effect may lead to a very small bias in the size of the cluster generated. As a comparison, in Table I we also show data which were obtained at T_c using R250 and SWC generators in which every fifth random number was used. With this simple modification the generators produced correct results. As a further check, we performed an "inefficient" Wolff simulation by growing all the clusters according to the Swendsen-Wang procedure but then choosing only one site at random and flipping the cluster to which it belongs. This modified algorithm did not exhibit the dramatic errors of the efficient implementation, presumably because the string of random numbers used to grow the cluster was not generated sequentially. It has also been found [15] that when the Wolff algorithm is used together with Wolff's embedding scheme [9] to study the classical spin Heisenberg model, no systematic difference can be found when compared with Swendsen-Wang or Metropolis simulations. We believe this results because each embedding generates a random bond Ising model and the resultant variation in bond probabilities is enough to destroy the effect of correlations in the random numbers.

The problems which we have encountered with the Wolff method are, in principle, a concern with other algorithms. Although Metropolis simulations are not as sensitive to these correlations, as resolution improves some very small bias may appear. Hidden errors obviously pose a subtle, potential danger for many geometric simulations such as percolation or random walks of various kinds which generate geometric structures using similar "growth algorithms" as the Wolff method.

In conclusion, extensive Monte Carlo simulations on an Ising model for which the exact answers are known have shown that ostensibly high quality random number generators may lead to subtle, but *dramatic,* systematic errors for some algorithms, but not others. Since there is no reason to believe that the model which we have investigated has any special idiosyncracies, these results offer another stern warning about the need to very carefully test the implementation of new algorithms. In particular, this means that a specific algorithm must be tested together with the random number generator being used *regardless* of the tests which the generator has passed.

This work is supported in part by NSF Grant No. DMR-9100692.

[1] A. M. Ferrenberg, D. P. Landau, and K. Binder, J. Stat. Phys. **63**, 867 (1991).
[2] G. Marsaglia, Proc. Natl. Acad. Sci. **61**, 25 (1968).
[3] M. N. Barber, R. B. Pearson, D. Toussaint, and J. L. Richardson, Phys. Rev. B **32**, 1720 (1985).
[4] S. Kirkpatrick and E. Stoll, J. Comput. Phys. **40**, 517 (1981).
[5] G. Marsaglia, in *Computer Science and Statistics: The Interface*, edited by L. Billard (Elsevier, Amsterdam, 1985).
[6] See A. M. Ferrenberg and D. P. Landau, Phys. Rev. B **44**, 5081 (1991).
[7] See, e.g., D. P. Landau, in *Applications of the Monte Carlo Method*, edited by K. Binder (Springer-Verlag, Heidelberg, 1987).
[8] R. H. Swendsen, J.-S. Wang, and A. M. Ferrenberg, in *The Monte Carlo Method in Condensed Matter Physics*, edited by K. Binder (Springer-Verlag, Heidelberg, 1992).
[9] U. Wolff, Phys. Rev. Lett. **62**, 361 (1989).
[10] A. E. Ferdinand and M. E. Fisher, Phys. Rev. **185**, 832 (1969).
[11] R. H. Swendsen and J.-S. Wang, Phys. Rev. Lett. **58**, 86 (1987).
[12] G. Marsaglia, B. Narasimhan, and A. Zaman, Comput. Phys. Commun. **60**, 345 (1990).
[13] Note that for the Swendsen-Wang and Metropolis algorithms, one update means one complete update of the lattice (MCS); in the Wolff algorithm, one update is less than one MCS and depends on the temperature. For simulations at T_c, a Wolff update is approximately 0.55 MCS.
[14] By this we refer to the judgement of generators as measured by their ability to pass a battery of statistical tests.
[15] K. Chen, A. M. Ferrenberg, and D. P. Landau (to be published).