C3P-945
SCCS-12
CRPC-TR90159
COLO-HEP-229

COMPARISON OF CLUSTER ALGORITHMS FOR TWO-DIMENSIONAL POTTS MODELS

Clive F. Baillie [1] and Paul D. Coddington [2]

Caltech Concurrent Computation Program,
California Institute of Technology,
Pasadena, CA 91125, USA

October, 1990

Abstract

We have measured the dynamical critical exponent z for the Swendsen-Wang and the Wolff cluster update algorithms, as well as a number of variants of these algorithms, for the $q = 2$ and $q = 3$ Potts models in two dimensions. We find that although the autocorrelation times differ considerably between algorithms, the critical exponents are the same. For $q = 2$, we find that although the data are better fitted by a logarithmic increase in the autocorrelation time with lattice size, they are also consistent with a power law with exponent $z \approx 0.25$, especially if there are non-negligible corrections to scaling.

[1] Current address: Physics Department, University of Colorado, Boulder CO 80309
[2] Current address: Physics Department, Syracuse University, Syracuse NY 13244

I. INTRODUCTION

Cluster update algorithms can greatly reduce critical slowing down in computer simulations of spin models. The original idea of Swendsen and Wang [1] (SW) was based on the work of Fortuin and Kasteleyn [2], which relates the Potts model [3] to a random bond percolation model [4]. The q-state Potts model consists of a lattice of spins σ_i which can take q different values, and its Hamiltonian is

$$H = K \sum_{<i,j>} \delta_{\sigma_i \sigma_j}, \tag{1}$$

where K is a dimensionless coupling constant. By introducing bonds between neighboring spins, with probability

$$p(\sigma_i, \sigma_j) = \delta_{\sigma_i \sigma_j}(1 - e^{-K}), \tag{2}$$

clusters of bonded spins are created. The SW algorithm consists of generating all such clusters, then choosing a random spin value for each cluster and assigning it to all the sites in that cluster. In the algorithm of Wolff [5], a site is chosen at random and a single cluster constructed around it. All the spins in this cluster are then flipped, i.e. changed to a single new Potts state (different from the current state of the cluster's spins) which is chosen at random from the $q-1$ possible values. Note that this is different from the SW update, which may leave the spins unchanged in some clusters. These algorithms, which update whole clusters of spins at a time, result in a dramatic decrease in critical slowing down over local algorithms which update only one spin at a time.

As shown by Wolff [6], the SW and Wolff algorithms are both special cases of the following general cluster update algorithm:

Step 1. The lattice is decomposed into N_c clusters c_i, using the bond probabilities of equation (2).
Step 2. A subset C of the c_i is chosen using some specified probability distribution, and all of the clusters of spins in C are flipped independently.

The SW algorithm corresponds to including each cluster in C with probability $(q-1)/q$, while the Wolff algorithm amounts to picking a single cluster with probability proportional to its size. Hence the only differences between the Wolff and SW algorithms are that the SW algorithm flips more clusters per iteration, while the Wolff algorithm flips relatively larger clusters. In order to investigate these differences further, we also studied some other methods of choosing the clusters to be flipped, including the following :

(1) flip all the clusters (hereafter denoted by AC),
(2) flip one of the clusters, chosen at random with equal probability for each cluster (denoted by RC), and

(3) flip the largest cluster (denoted by LC).

Note that flipping all of the clusters will not work in the case of the Ising model ($q = 2$), since it is equivalent to a global spin flip, and is thus highly non-ergodic! However for $q > 2$ it is a perfectly acceptable algorithm.

II. SIMULATIONS

We have made a high statistics study of the above cluster algorithms for the $q = 2$ and $q = 3$ Potts models in two dimensions. Preliminary results have been reported elsewhere [7]. We measured the autocorrelations in the energy $E = -\sum_{<i,j>} \delta_{\sigma_i \sigma_j}$ and the absolute magnetization $|M| = |\sum_j e^{2\pi i \sigma_j / q}|$. The normalized autocorrelation function for an operator A is

$$\rho_A(t) = \frac{< A(0)A(t) > - < A(0) >^2}{< A(0)A(0) > - < A(0) >^2}. \tag{3}$$

At large t this should decay as $e^{-t/\tau_{exp}}$, where τ_{exp} is the exponential autocorrelation time (independent of A unless A is orthogonal to the slowest mode). The quantity which is actually relevant to the error in A is the integrated autocorrelation time [8]

$$\tau_{int,A} = \frac{1}{2} + \sum_{t=1}^{\infty} \rho_A(t). \tag{4}$$

The integrated and exponential autocorrelation times have associated exponents $z_{int,A}$ and z_{exp} governing critical slowing down. These are given by $\tau(L) \sim L^z$ for both the exponential and integrated autocorrelations, where LxL is the size of the lattice and $\tau(L)$ is measured at the infinite volume critical point.

Autocorrelations are traditionally measured between each update of the entire lattice, so for the single cluster updates, where only a fraction of the lattice sites are updated at each iteration, the autocorrelation time needs to be scaled by a factor of the average cluster size $< |c| >$ divided by the lattice size, so that

$$\tau = \tau' < |c| > L^{-d}, \tag{5}$$

where τ' is the unscaled autocorrelation time and d is the lattice dimension (here $d = 2$). For the Wolff algorithm, the average cluster size scales like the susceptibility, hence $< |c|_W > \sim L^{\gamma/\nu}$. The average cluster size for SW is (apart from finite size effects) independent of L, thus $< |c|_{RC} > \sim L^0$. As in the percolation model [4], the maximum cluster size (normalized to the size of the lattice) scales like the magnetization, so that $< |c|_{LC} > \sim L^{d-\beta/\nu}$. From equation (5) and the definition of z we have that

$$\begin{aligned} z_W &= z'_W - (d - \gamma/\nu) \\ z_{RC} &= z'_{RC} - d \\ z_{LC} &= z'_{LC} - \beta/\nu \end{aligned} \tag{6}$$

where z' is the unscaled exponent, $\tau'(L) \sim L^{z'}$. In our calculations the scaled value of z is actually obtained by scaling the autocorrelations by the average cluster size, but we have checked that using equation (6) instead gives consistent answers for LC and Wolff. For RC, however, there are large finite size effects on the smaller lattices and the average cluster size does not scale well.

Note that the computational requirement for the Wolff algorithm is proportional to the average size of the clusters which are flipped [5], but for the other single cluster algorithms (RC and LC) it is necessary to form all the clusters in order to choose the one to flip. Hence the computational requirement for the RC and LC algorithms goes like L^d (the same as for SW), and in that sense the above scaling is unwarranted. However it may be possible to invent a clever algorithm which constructs single clusters with the required probability distribution, and in any case the scaling is necessary to allow a sensible comparison with the Wolff algorithm.

The autocorrelations were measured at the infinite volume critical point (which is known exactly for 2-d Potts models) over runs of typically 5×10^5 to 10^6 sweeps (10^6 to 10^7 sweeps for the RC algorithm), and 10 or 20 such runs were done for each of the different lattice sizes, from 8^2 up to 128^2. The results from the multiple runs were averaged to give the overall autocorrelation function $\rho_A(t)$. For the SW and Wolff algorithms, we also did runs on a 256^2 lattice. For SW we did a total of 7.5 million sweeps for $q = 2$ and 3.2 million for $q = 3$, while for Wolff we did 7.5 million sweeps for $q = 2$ and 5 million for $q = 3$. For the SW algorithm with $q = 2$ we also did a total of 5.6 million sweeps on a 512^2 lattice. In the SW case on these larger lattices we used a parallel cluster-finding algorithm which we have developed [9] .

The autocorrelation time τ_{exp} for an operator A was extracted from a fit to $\log \rho_A(t)$ in a region where it was linear, typically somewhere between τ_{exp} and $3\tau_{exp}$. $\tau_{int,A}$ was calculated by summing $\rho_A(t)$ for t less than some cutoff value, and then using values from the fit to approximate $\rho_A(t)$ for the remaining part of the sum, which is then just a geometric series and so can be summed analytically. The cutoff was taken to be the largest point used in the fit, which was usually between $2\tau_{exp}$ and $3\tau_{exp}$. This method is similar to that used by Wolff [10] . The above analysis was also done for each of the 10 or 20 multiple runs, which provided an estimate of the error in the autocorrelation times.

III. RESULTS

We found as expected that the energy and absolute magnetization have very similar values of τ_{exp}, however $|M|$ decorrelates faster than E, giving smaller values of both τ_{int} and z_{int}. Results for $\tau_{int,E}$ for the different algorithms are shown in Figs. 1 and 2 for $q = 2$ and $q = 3$ respectively, along with straight lines which represent chi-squared fits to a power law. The autocorrelation times as a function of L are given for the Wolff and SW algorithms in Tables 1 and 2, for $q = 2$ and $q = 3$ respectively. The values of z from fits to all the data are given in Table 3. We have not given any values for the RC algorithm, since the data are not good enough to reliably extract an exponent. The exponents for all the other algorithms are approximately the same, and in particular it appears that $z_{SW} = z_{Wolff}$ for $q = 2$ and $q = 3$ Potts models in two dimensions. The exponents for the LC algorithm with $q = 2$ are rather high, however this is due to the fact that the value of z tends to decrease slightly if the fits are done at larger L. The SW and Wolff algorithms also give $z \approx 0.30$ if the fits are done only up to $L = 128$.

Although the exponents are the same, the autocorrelation times differ substantially between algorithms. The Wolff algorithm has values which are less than those of SW by a factor of about 0.45 for $q = 2$ and 0.67 for $q = 3$. It is interesting to note that these ratios are practically the same as the probabilities for flipping a cluster in the SW algorithm, so that if the SW algorithm was scaled (like the single cluster algorithms) by the proportion of spins which are actually *changed* in the update, then the Wolff and SW algorithms would give nearly identical results. It therefore appears that (at least in two dimensions) the main advantage of a single cluster algorithm such as Wolff is that the cluster is always flipped. Notice however that flipping all the SW clusters (the AC algorithm) is actually worse than the standard SW algorithm, and in $d > 2$ this argument does not hold since the exponents (and not just the autocorrelation times) are different for the Wolff and SW algorithms [10, 11] .

Another proposed reason for the superiority of the Wolff algorithm over SW is that the average cluster size is larger, and this is supported by the fact that the LC algorithm, where the largest cluster is flipped, does slightly better than Wolff. It is therefore quite surprising to see that the RC algorithm, where the average cluster size is approximately the same as for SW, does even better than Wolff or LC. This suggests that (at least in two dimensions) it is the single cluster nature of these updates, rather than the size of the clusters, which is most important, and that successive clusters in a single cluster update are somehow less correlated than the clusters in a full lattice update such as the SW algorithm. The overlap between sites in successive clusters gives some indication of how successive clusters are correlated, and thus should affect the autocorrelations. The small RC clusters have a negligible average overlap, which is not the case for the Wolff clusters in two dimensions. This overlap between successive clusters in the Wolff algorithm was studied by Tamayo *et al.* [11] for the Ising model, and found to decrease markedly as d increases. It seems likely that it is a combination of the effect of larger clusters, and a small overlap between successive clusters, which results in the Wolff algorithm having smaller dynamical critical exponents than the SW algorithm for $d > 2$.

We have also done a preliminary investigation, with a single run of 10^5 to 10^6 sweeps for each lattice size, of another algorithm, for which all "large" clusters (those greater than a certain cutoff size) are flipped. In a study of the mean-field Ising model, this algorithm was found to give the same value of z as SW, but the values of τ were halved [12] . For the 2-d Potts model we chose the cutoff to be $L^{3/2}$, and found that the autocorrelation times for this algorithm are decreased only slightly, giving values approximately midway between those of SW and LC, with the same value of z.

A number of other studies have been made of the dynamical exponents of cluster algorithms for Potts models. In their original paper [1], Swendsen and Wang measured the autocorrelations on fairly small lattice sizes, and thus obtained a rather high value of 0.35(1) for z. Wolff [10] has compared his algorithm to SW for the 2-d and 3-d Ising models, and his results agree fairly well with our data. A study of the Wolff algorithm for the Ising model has also been carried out by Tamayo et al. [11], who obtain similar results. Their paper also contains some remarks on the RC algorithm, and suggests that $z_{RC} < z_W$ for the 2-d model. From our results it appears that this is certainly possible, however the trend at larger lattice sizes seems to imply that the exponent is probably the same as for the other algorithms. Much better data would be required to draw a firm conclusion. Our value for z_{SW} in the $q = 3$ case agrees with that of Swendsen and Wang [1], and the more recent result of Li and Sokal [13] . Kandel et al. [14] obtain a value of 0.4 by measuring the relaxation time from an ordered state to equilibrium, claiming to see a crossover to this smaller value of the exponent for $L > 32$, however we see no sign of such behavior.

An interesting recent development is the claim by Burkitt and Heermann [15] that the autocorrelations in the 2-d Ising model using the SW algorithm grow logarithmically rather than as a power law, so that $z_{SW} = 0$. It is very difficult to distinguish between a logarithm and a small power. Since power law or logarithmic behavior is only expected *asymptotically*, there is some uncertainty as to which data points should be included in the fit (the same is true in fitting the autocorrelation function to obtain τ), so the fits are somewhat subjective. If we look at the slopes of lines connecting successive points in Fig. 1 (i.e. the approximations to z using only two successive lattice sizes), we see a slight trend towards smaller values as L increases. The difficult problem is to determine whether this trend disappears at large L, resulting in a non-zero value of z, or continues, signifying logarithmic behavior. Note that the $q = 3$ data fit well to a power law even down to $L = 16$, and in this case a logarithmic increase in τ can be definitely excluded.

In Fig. 3 we plot the exponential and integrated autocorrelations in $|M|$, along with fits to a power law (solid lines) and a logarithm (dashed lines). As seen by Burkitt and Heermann, $\tau_{int,|M|}$ seems to be better fitted with a logarithm, especially for small values of L (although these are not really relevant, since we are looking for asymptotic behavior). It is therefore quite possible that $z_{int,|M|}$ is zero. The other exponents ($z_{int,E}$ and z_{exp} for E and $|M|$) are much harder to ascertain, since the autocorrelations do not give good fits to either a log or a power law unless smaller values of L are excluded. For $\tau_{int,E}$, this can be seen in Fig. 1, where the solid line is a power law fit, and the dashed line a fit to a logarithm. In the case of τ_{exp} for $|M|$, we found substantially greater values at large L than did Burkitt and Heermann, and consequently were unable to reproduce their good logarithmic fit down to small L. This is possibly due to large errors in their results at large L, although it is difficult to tell since no errors are given.

In Table 4 we give the values of chi-squared per degree of freedom for both fits, for the Wolff and SW algorithms. The fits include only data with $L \geq 64$ for SW, and $L \geq 50$ for Wolff, so that we are only fitting to five data points, and this may affect the reliability of the chi-squared values. The results favor logarithmic behavior, although we would not claim that the evidence is conclusive, especially since the differences are so small that corrections to scaling could be very important. Good data on much larger lattices will probably be necessary to conclusively differentiate between $z = 0$ and $z \approx 0.25$.

ACKNOWLEDGEMENTS

The simulations were done using a number of different parallel computers. The parallel algorithm was run on a 512 node NCUBE-1, a 192 node Symult S2010, and a 32 processor Meiko Computing Surface, while the sequential code was run on a 20 processor Sequent Symmetry, a 20 processor Encore Multimax, a 96 processor BBN GP1000 Butterfly, and a network of SUN and DEC workstations. We would like to thank the Caltech Concurrent Supercomputer Facility and the Advanced Computing Research Facility at Argonne National Laboratory for the use of these machines. This work was sponsored in part by Department of Energy grants DE-FG03-85ER25009 and DE-AC03-81ER40050. CFB is supported by Air Force Office of Scientific Research Grant AFOSR-89-0422 and by the Department of Energy under contract DE-AC02-86ER40253.

References

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

[2] P.W. Kasteleyn and C.M. Fortuin, J. Phys. Soc. Jpn. Suppl. **26s**, 11 (1969); C.M. Fortuin and P.W. Kasteleyn, Physica **57**, 536 (1972).

[3] R.B. Potts, Proc. Camb. Phil. Soc. **48**, 106 (1952); F.Y. Wu, Rev. Mod. Phys. **54**, 235 (1982).

[4] D. Stauffer, Phys. Rep. **54**, 1 (1978); J.W. Essam, Rep. Prog. Phys. **43**, 830 (1980).

[5] U. Wolff, Phys. Rev. Lett. **62**, 361 (1989).

[6] U. Wolff, Nucl. Phys. B **334**, 581 (1990).

[7] P. D. Coddington and C. F. Baillie, Proceedings of the International Workshop 'Lattice 89', ed. N. Cabibbo *et al.*, Nucl. Phys. B (Proc. Suppl.) **17**, 305 (1990).

[8] N. Madras and A.D. Sokal, J. Stat. Phys. **50**, 109 (1988).

[9] C.F. Baillie and P.D. Coddington, "Cluster Identification Algorithms for Spin Models — Sequential and Parallel", to be published in Concurrency: Practice and Experience.

[10] U. Wolff, Phys. Lett. **B228**, 379 (1989).

[11] P. Tamayo, R.C. Brower and W. Klein, J. Stat. Phys. **58**, 1083 (1990).

[12] T.S. Ray, P. Tamayo and W. Klein, Phys. Rev. A **39**, 5949 (1989).

[13] X.-J. Li and A.D. Sokal, Phys. Rev. Lett. **63**, 827 (1989).

[14] D. Kandel, E. Domany and A. Brandt, Phys. Rev. B **40**, 330 (1989).

[15] D.W. Heermann and A.N. Burkitt, Physica A **162**, 210 (1990).

	SW		WOLFF	
L	τ_{exp}	τ_{int}	τ_{exp}	τ_{int}
8	2.598 (10) 2.568 (13)	2.590 (5) 2.451 (6)	1.175 (29) 1.127 (29)	1.095 (1) 1.006 (1)
16	3.315 (13) 3.282 (17)	3.258 (5) 2.957 (5)	1.612 (22) 1.555 (22)	1.435 (4) 1.215 (3)
32	4.117 (18) 4.095 (19)	4.016 (5) 3.468 (8)	2.101 (11) 2.048 (14)	1.815 (3) 1.403 (3)
50	4.740 (10) 4.709 (25)	4.586 (5) 3.812 (6)	2.460 (21) 2.382 (21)	2.079 (5) 1.513 (3)
64	5.081 (21) 5.002 (29)	4.899 (10) 3.986 (7)	2.577 (31) 2.454 (31)	2.225 (6) 1.559 (5)
100	5.78 (4) 5.71 (5)	5.510 (17) 4.305 (12)	2.943 (25) 2.801 (23)	2.489 (7) 1.641 (3)
128	6.16 (4) 6.11 (5)	5.874 (16) 4.509 (13)	3.10 (4) 3.01 (4)	2.654 (12) 1.694 (5)
256	7.21 (7) 7.23 (6)	6.87 (3) 5.018 (13)	3.42 (6) 3.35 (5)	3.076 (24) 1.818 (9)
512	8.53 (18) 8.51 (9)	8.04 (10) 5.530 (20)	- -	- -

Table 1

Exponential and integrated autocorrelation times for the SW and Wolff algorithms as a function of the lattice size for $q = 2$. For each value of L, the first line displays the values for the energy, the second line the absolute magnetization.

	SW		WOLFF	
L	τ_{exp}	τ_{int}	τ_{exp}	τ_{int}
8	6.182 (18) 6.196 (22)	6.056 (9) 5.821 (9)	4.36 (3) 4.31 (3)	3.900 (8) 3.735 (10)
16	9.260 (8) 9.260 (8)	8.99 (3) 8.45 (3)	6.82 (6) 6.81 (6)	5.878 (15) 5.419 (14)
32	13.77 (13) 13.78 (13)	13.30 (6) 12.11 (5)	10.32 (12) 10.26 (12)	8.76 (4) 7.72 (3)
64	20.39 (22) 20.38 (20)	19.58 (12) 17.32 (10)	17.0 (7) 16.7 (5)	13.08 (16) 11.06 (11)
128	30.2 (4) 30.4 (3)	28.64 (12) 24.55 (12)	24.2 (1.0) 22.7 (6)	19.5 (3) 15.53 (14)
256	43.4 (2.0) 43.2 (10)	41.3 (1.5) 34.3 (1.0)	33 (3) 31.4 (1.3)	27.7 (8) 21.5 (3)

Table 2

Same as Table 1 but for $q = 3$.

	$Q = 2$		$Q = 3$	
	z_{exp}	z_{int}	z_{exp}	z_{int}
SW	0.26(1) 0.26(1)	0.25(1) 0.16(1)	0.56(2) 0.56(2)	0.55(1) 0.51(1)
Wolff	0.22(2) 0.22(2)	0.25(1) 0.12(1)	0.60(2) 0.57(2)	0.57(1) 0.51(1)
LC	0.32(2) 0.32(2)	0.29(1) 0.16(1)	0.59(2) 0.59(2)	0.57(1) 0.52(1)
AC	- -	- -	0.57(2) 0.57(2)	0.56(1) 0.53(1)

Table 3

Dynamical critical exponents for the different cluster algorithms. For each algorithm, the first line displays the values for the energy, the second line the absolute magnetization.

	SW		Wolff	
	power	log	power	log
$\tau_{exp,E}$	2.4	0.5	5.2	2.6
$\tau_{int,E}$	5.9	1.9	4.2	0.6
$\tau_{exp,\|M\|}$	2.2	0.7	3.8	2.7
$\tau_{int,\|M\|}$	6.2	0.5	1.1	0.4

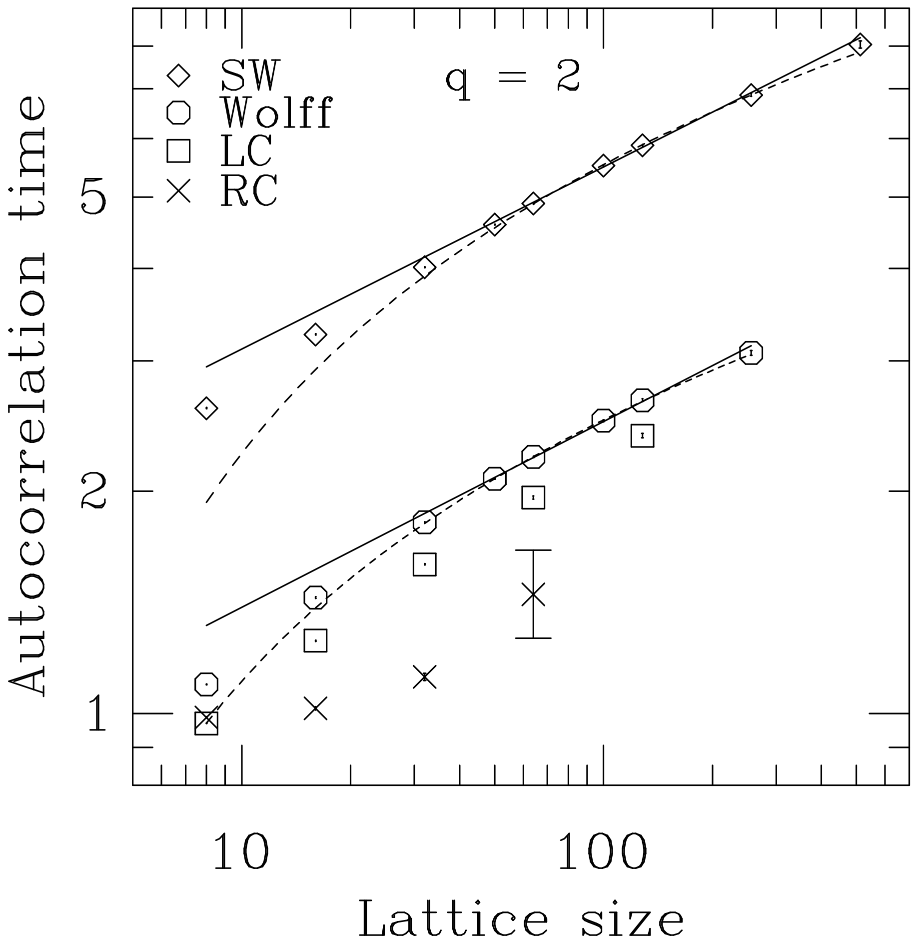
Table 4

Chi-squared per degree of freedom for fits to a power and a logarithm.

Figure 1. $\tau_{int,E}$ for the different algorithms for $q = 2$. The solid lines are fits to a power law, with exponents given in Table 1, while the dashed lines are fits to a logarithm.

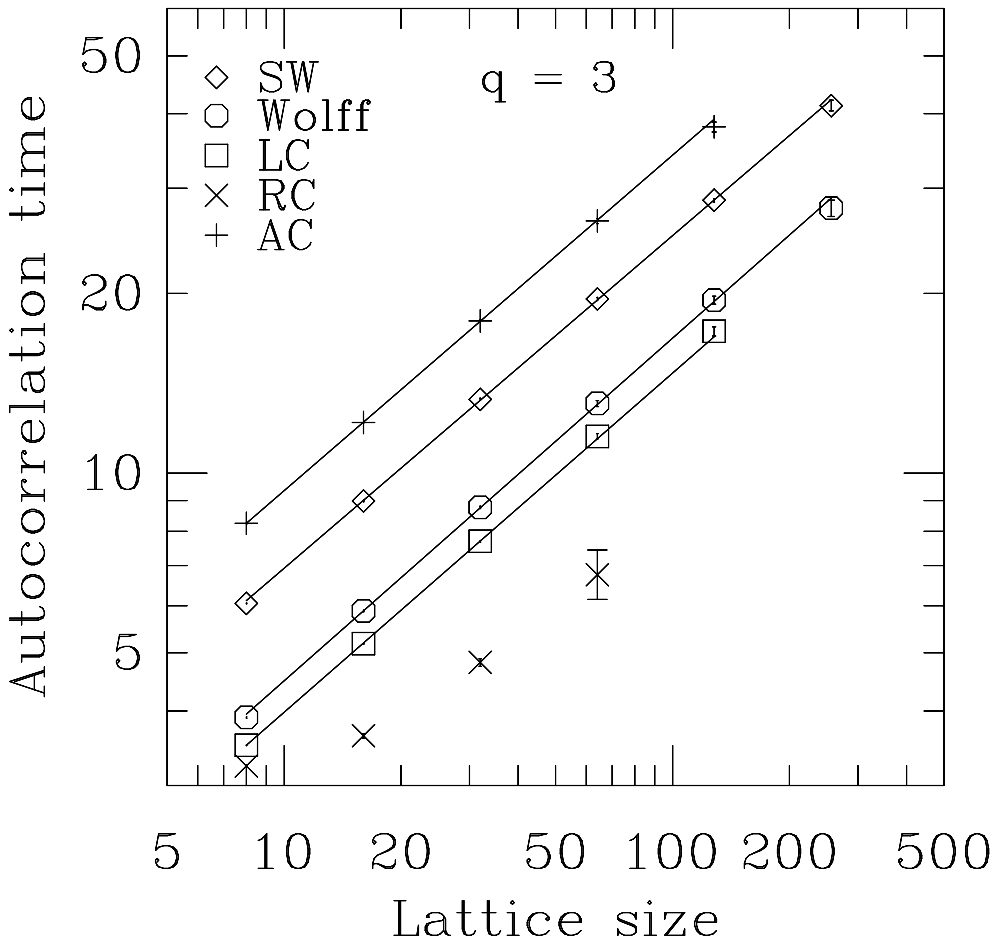
Figure 2. $\tau_{int,E}$ for the different algorithms for $q = 3$. The solid lines are fits to a power law.

Figure 3. Exponential and integrated autocorrelation times for $|M|$ for the Wolff and SW algorithms with $q = 2$. The solid lines are fits to a power law, while the dashed lines are fits to a logarithm.



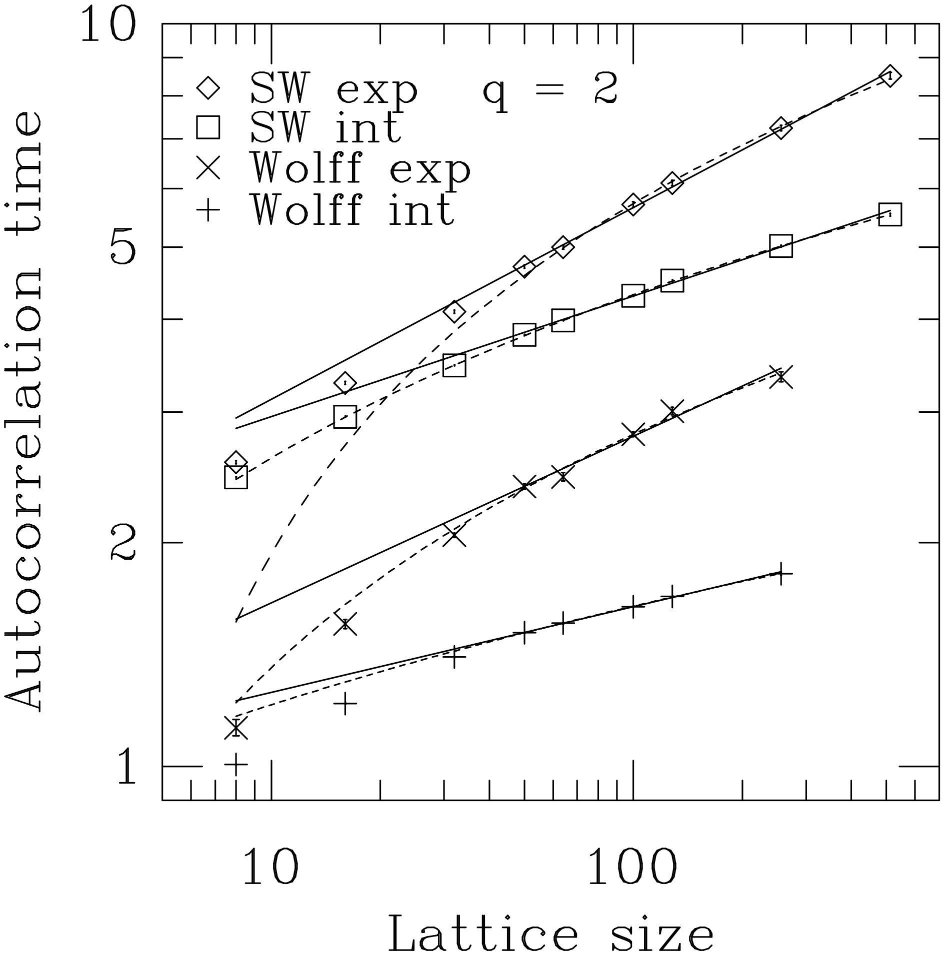
SW
Wolff
LC
RC
q = 2
Autocorrelation time
5
2
1
10
100
Lattice size

Fig. 1



q = 3
SW
Wolff
LC
RC
AC
Autocorrelation time
50
20
10
5
5
10
20
50
100
200
500
Lattice size

Fig. 2



10
5
2
1
10
100
Autocorrelation time
Lattice size
SW exp q = 2
SW int
Wolff exp
Wolff int

Fig. 3