Monte Carlo simulation of 3D self-avoiding walks

Monte Carlo simulations of 3D self-avoiding walks

Peter Grassberger

Physics Department, University of Wuppertal, D-5600 Wuppertal 1, Federal Republic of Germany

Abstract. We present simulation results for 3D self-avoiding walks of moderate length ($N \leqslant 2000$), using a recently introduced recursive sampling method. We show that this method is indeed maximally efficient for determining the connectivity constant μ. Combining our data with previous results from exact enumerations and from simulations of very long walks using the pivot algorithm, we obtain very precise estimates of μ and of the critical exponents γ and ν. We also point out significant deviations from uniformity in a very popular random number generator which was considered safe until now.

Self-avoiding walks (SAWs) are one of the prime models in the theory of critical phenomena. This results not only from the fact that they are related to the $n \to 0$ limit of the O(n) model [1], but also from their great importance as a model for randomly coiled linear polymers.

While the asymptotic behaviour of 2D SAWs is basically understood, the critical exponents are still only approximately known in three dimensions. Besides field theoretic approximations (ϵ and loop expansions) it seems that Monte Carlo (MC) simulations and exact enumerations of short walks are the most efficient tools.

For estimating end-to-end distances and radii of gyration (and thus the critical exponent ν defined via $R_N \sim N^\nu$) the most efficient known MC method is the pivot algorithm [2]. It needs a CPU time roughly of O(N) to produce one statistically independent SAW of length N. It does this by making efficient global moves: each move consists of a random choice of a 'pivot' on the chain, and a subsequent rotation or reflection of one half of the chain around this pivot. This is to be compared with the best MC algorithm using local moves, due to Beretti and Sokal [3]. In this latter algorithm a grand canonical ensemble is generated at $\sim$ O(N^2) steps per statistically independent SAW of length N.

Recently, we introduced a new MC algorithm, called the 'recursive sampling' (RS) method in the following [4]. It is very similar to the incomplete enumeration method of Redner and Reynolds [5] (which is itself similar to the Beretti–Sokal algorithm). The basic difference from the Redner–Reynolds algorithm is that we do the incomplete enumeration by means of a recursive call to a subroutine. This subroutine has a lattice site as its argument, and essentially does nothing else but calling itself at the neighbouring lattice sites after having put a flag at its argument site (to indicate that the site has been visited) which is cleared when the subroutine is exited. A complete BASIC routine is given in the appendix of [4]. In contrast to complete enumeration, the subroutine is called for each neighbour of the present site only with a probability $P < 1$. The use of recursive subroutine calls makes the otherwise tedious book-keeping trivial. It also makes the algorithm very short, intuitive and flexible. The latter follows from the fact that the probability P does not have to be constant, but can depend, e.g., on the walk length N or on the present site i. It was essentially the

latter which was stressed and used in [4], where SAWs were studied in random media by means of a learning algorithm for optimizing the N-dependence of P_N.

The fact that it is very easy to obtain arbitrarily biased samples makes the RS method ideally suited for such problems as polymers near surfaces or edges, or polymers with self-attraction (θ polymers [1]). In these cases, the pivot algorithm should become inefficient due to the much lower acceptance rates for global moves, while the modifications in the RS method are trivial. Applications to these problems will be discussed elsewhere [6]. In the present paper I shall discuss ordinary SAWs in three dimensions, in spite of the fact that the method seems at first sight much less efficient than the pivot algorithm in this case.

The reasons why the RS method can also be useful for ordinary SAWs are twofold. First, due to the simple program and small overheads it is very fast for intermediate length SAWs (faster, it seems, than the known implementations of the pivot algorithm for $N \sim 10^2$), and results for such walks can help in estimating corrections to scaling. Second, we can show that the RS method indeed makes the best use of the random number generator (within a very large class of algorithms) for estimating the connectivity constant μ. The latter is defined via the number $C(N)$ of SAWs of length N as

$$C(N) \sim \mu^N N^{\gamma-1}. \tag{1}$$

Thus we can obtain the best estimate so far of μ, which is then combined with existing exact enumerations [7] to obtain a better estimate of γ.

Let us discuss the second aspect first. Assume that we use in an MC simulation a sequence of (pseudo-)random numbers $r_i \in [0, 1]$. Each r_i is compared with a fixed real number p, and some action is taken if and only if $r_i < p$. This is the typical way that random numbers are used in order to estimate a threshold such as the critical fugacity $p_c = 1/\mu$ of SAWs. We cannot prove that there is no other method which makes better use of the r_is (the above uses less than 1 bit, if $p \neq 1/2$), but we shall prove that the RS method is optimal within this class of *comparison-based* MC methods.

Assume that in running a comparison-based MC method one draws in total M random numbers. Let us call m the number of cases where $r_i < p$. Then we know that $\langle m \rangle = pM$, but m will fluctuate with a variance

$$(\Delta m)^2 = p(1 - p)M. \tag{2}$$

If we use this for estimating p_c from the number of actions taken (i.e. from m), then the error of p_c is at least

$$\Delta p_c \geqslant \sqrt{p_c(1 - p_c)/M}. \tag{3}$$

We will now show that this inequality can be saturated by a proper choice of the acceptance probabilities P_N. If P_N is the probability with which each possible 1-step extension of an $(N - 1)$-step SAW is accepted, then the average number of N-step SAWs is

$$n_N = C(N) \prod_{k=1}^{N} P_k \sim N^{\gamma-1} \prod_{k=1}^{N} \mu P_k. \tag{4}$$

The total CPU time for generating an ensemble of SAWs with n_0 starts will be

$$T = c \sum_N n_N \tag{5}$$

(with c a constant of order 1), and the total number of random number generator (RNG) calls will be

$$M = \sum_N n_N \mu. \tag{6}$$

The relative variance of n_k can be estimated recursively,

$$\left(\frac{\Delta n_k}{n_k}\right)^2 = \left(\frac{\Delta n_{k-1}}{n_{k-1}}\right)^2 + \frac{1-P_k}{n_k} = \sum_{j=1}^{k} \frac{1-P_j}{n_j}. \tag{7}$$

Assume now that we can only simulate SAWs of length $\leqslant N$ (due to storage limitations, for example), and we want to minimize Δn_N. Since the variance will decrease $\propto 1/T$ due to the central limit theorem, we have to minimize the product

$$T\left(\frac{\Delta n_k}{n_k}\right)^2 = c \sum_{j=1}^{N} n_j \sum_{k=1}^{N} \frac{1-P_k}{n_k}. \tag{8}$$

By the Schwarz inequality, the minimum is attained for $n_j \propto \sqrt{1-P_j}$. Together with (4) this gives a recursion relation for P_j which, for large N, gives

$$P_j = \left(\frac{j+1}{j}\right)^{\gamma-1} \mu^{-1} \qquad n_j = \text{constant}. \tag{9}$$

Inserting this into (8) and replacing T by M, we find (again for large N)

$$\left(\frac{\Delta n_N}{n_N}\right)^2 = \frac{(1-p_c)N^2}{p_c M}. \tag{10}$$

If we would know γ and all corrections to scaling exactly, then this error of n_N would imply an error of μ or of $p_c = 1/\mu$ which is given precisely by (3). Since the uncertainties of γ and the corrections to scaling do not influence the estimate of p_c for $N \to \infty$, we see that the optimal results are obtained by choosing N as large as possible [3].

In our simulations we computed errors of n_N from the sample-to-sample fluctuations and indeed found (3) to be saturated for the optimal choice of P_j given by (9). However, nearly as good results were obtained if we simply took $P_j = 1/\mu$.

The error in the end-to-end distance R_N cannot be estimated in the same rigorous way. But a simple heuristic argument due to Beretti and Sokal (as well as exact bounds given by them for the algorithm in [3]) also applies to the present algorithm; for the 'optimal' choice of P_j the algorithm essentially makes a random walk in the chain length k with reflecting boundaries at $k = 0$ and $k = N$. Thus we need $\sim N^2$ steps to reach $k = N$ when starting from $k = 0$, giving one statistically independent SAW of length N. The relative variance $(\Delta R_N/R_N)^2$ should be inversely proportional to the number of such SAWs, giving $\Delta R_N/R_N \sim N/\sqrt{M}$. During this simulation we can also measure R_k with $k < N$, and spend a fraction k/N of steps on its measurement. Thus the error $\Delta R_k/R_k$ from a simulation of SAWs with length up to N should be

$$\Delta R_k/R_k \sim \sqrt{kN/M} \qquad k = 1, 2, \ldots, N. \tag{11}$$

This argument is not quite correct since the walk in the chain length is not a simple uncorrelated random walk. We should thus expect that $\Delta R_k/R_k$ will increase with a slightly different power of k, while its dependence on N and M will be fixed by the central limit theorem. Numerically we found for the 'optimal' choice (9)

$$\Delta R_k/R_k \sim k^{0.55}\sqrt{N/M}. \tag{12}$$

If we use $P_N = 1/\mu = $ constant instead of (9), we find slightly worse behaviour, $\Delta R_k/R_k \propto k^{0.50}(N^\gamma/M)^{1/2}$, indicating that (9) is also the optimal choice for R_N.

Results for n_k are shown in figure 1. In this figure we have combined results from various runs with different $\{P_N\}$. Using the left hand part of (4) we first computed the corresponding $C(N)$, and then divided by suitable factors in order to enhance the significance of the plot. Indeed, the quantity plotted in figure 1 is

$$\frac{C(N)}{(N+0.446)^{\gamma-1}\mu^N} \tag{13}$$

for two different sets of (μ, γ) pairs. This should tend towards a constant for $N \to \infty$, if $\mu = 1/p_c$ and γ were chosen correctly. In addition to our MC data we also show exact enumeration results from [7] which agree perfectly with our results for small N.
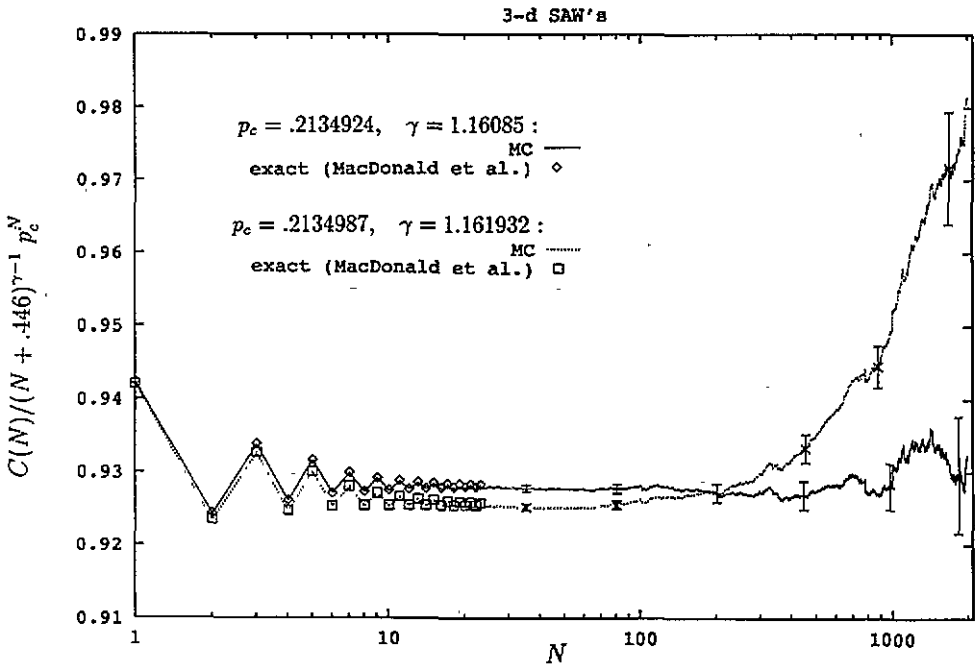

Figure 1. Semi-logarithmic plot of the quantity defined in (13) against $\log N$. For the correct values of p_c and γ the curve should be horizontal up to statistical fluctuations. The broken curve is for the best previous values of p_c and γ, while the full curve is for our proposed values. Typical error bars are only given in a few points in order not to overcrowd the picture.

The constant 0.446 added to N in the denominator of (13) takes into account the leading correction to scaling which was found to be compatible with it being analytic. This is in

agreement with [7], where it was also found that the non-analytic corrections known, e.g., from field theory [8] do not couple to $C(N)$.

One of the curves is obtained with the central values of [7], $p_c = 0.213\,498\,7$, $\gamma = 1.161\,932$. It seems definitely ruled out by our data, by roughly six standard deviations. The other (much flatter one) corresponds to the central values of our new estimate

$$p_c = 0.213\,492\,4 \pm 0.000\,001\,0 \qquad \gamma = 1.1608 \pm 0.0003. \qquad (14)$$

Notice that the estimate of p_c uses only the new MC data, but the estimate of γ needs both the MC data (to pin down p_c) and the enumeration data. Without the latter, the best estimate of γ would have been 1.160 ± 0.002. The observation that exact enumerations give poor results for p_c and can lead to very precise critical exponents when supplied with estimates of p_c from MC simulations was also made in [9] in another context. Note also that the error estimate is straightforward for p_c, while it is an 'educated guess' for λ, as is usual for critical exponents.

While we measured n_k from runs with N up to 2000 (using lattices of up to 4×10^7 sites with helical boundary conditions and multispin-coding [10]), we measured R_N only on walks of length up to $N = 1000$. This was done in order to reduce CPU time.

Our results for R_N are shown in figures 2 and 3. In figure 2 we plot the quantity $R_N/N^{0.59}$ against $\log N$. We show not only our own data, but also results from exact enumerations [11], from older MC simulations of Rapaport [12] and two sets of data obtained with the pivot algorithm [2, 13]. The simulations from [13] were indeed done for a slightly different geometry. Instead of allowing the SAWs to occupy the whole cubic lattice, a needle along one of the axes starting from the origin of the SAWs was excluded (for earlier simulations with this geometry, see [14]). In order to compare our results with these data, we have also performed some simulations with the same needle excluded. They are also shown in figure 2, along with our simulations for the ordinary geometry.

From figure 2 we can immediately make several comments. First, the data of [12] are definitely too low. Although no error estimates were given in [12], it was suggested there that the errors can be estimated from the deviations from a straight line in a log–log plot. As seen from figure 2, this is not true. Second, our simulations agree perfectly with the exact enumeration results and with the very high statistics results of [13]. They also agree with the results of [2], but the latter are considered as preliminary by the authors and should be superceded by more precise estimates.

Finally, we see very large corrections to scaling in the ordinary (no excluded needle) SAWs, but much smaller deviations in the excluded needle results. For the latter it is clear that the slope at large N is negative, i.e. $\nu < 0.59$. Both curves seem to converge for large N, which implies that there is substantially more curvature in the ordinary SAW data than in the excluded needle data at large N. This suggests that the leading correction to scaling exponent Δ defined by

$$R_N \sim N^\nu (1 + A/N^\Delta + \cdots) \qquad (15)$$

might be smaller for ordinary SAWs than for SAWs with an excluded needle.

The latter is also supported by studying effective exponents obtained from local slopes in a log–log plot,

$$\nu_{\text{eff}}(N) = \frac{\log(R_{aN}/R_{N/a})}{2 \log a}. \qquad (16)$$
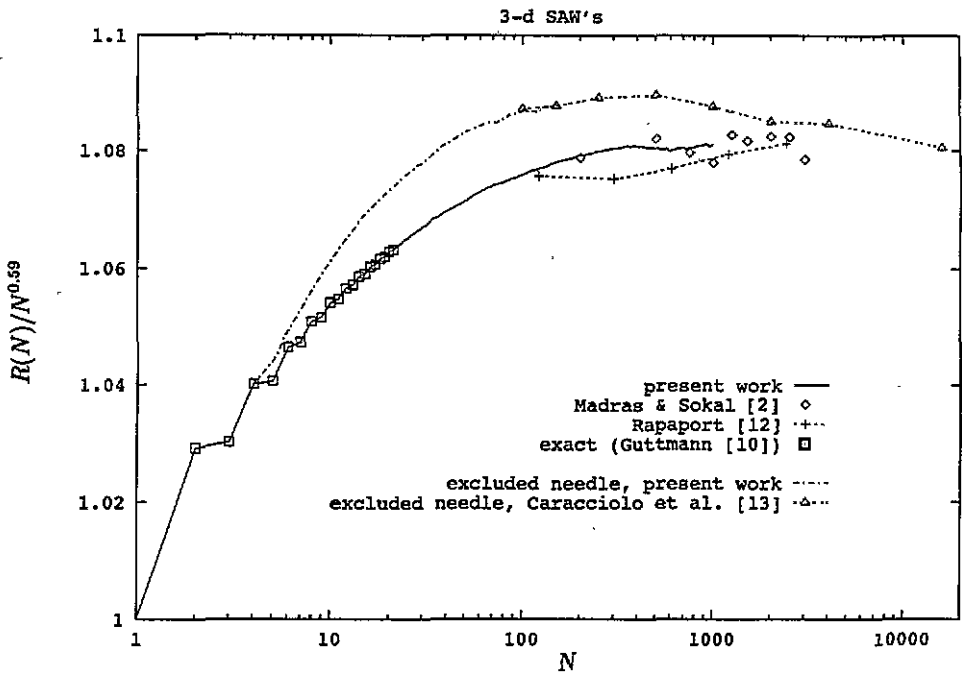
Figure 2. Semi-logarithmic plot of $R_N/N^{0.59}$ against $\log N$. The plot shows our MC data both for ordinary SAWs (full curve) and for an excluded needle (chain curve), together with older results: exact enumerations [11] ($\square$), MC data of [12] (+) and MC data obtained with the pivot algorithm. The latter are for ordinary SAWs from [2] ($\lozenge$) and for an excluded needle from [13] ($\triangle$). Relative errors of our MC data are $\pm0.06\%$ ($N = 1000$, ordinary SAWs) resp. $\pm0.05\%$ ($N = 150$, excluded needle).

For our data, we took $a = 2$, for the data from [3, 13] we used $a \approx 2$ when $a = 2$ was not possible. According to (15), $\nu_{\text{eff}}(N)$ should have a term $\propto 1/N^\Delta$ which should give a linear curve when plotted against $1/N^\Delta$. Such plots are presented in figure 3, with $\Delta = 0.22$ for ordinary SAWs and with $\Delta' = 0.45$ for the excluded needle. From the linearity of the curves we conclude that these are indeed the leading exponents, with errors $\approx \pm0.05$. The latter error estimate is just a very subjective guess, but it would seem very difficult to explain the data with the *same* Δ for both sets.

A leading correction exponent $\Delta = 0.22$ was predicted in [13] where essentially the excluded needle data were used. It seems ironic that we find that the amplitude of the leading term vanishes for just R_N for these data (it is non-zero in other observables such as z_N where z is the coordinate parallel to the needle), while we confirm it for ordinary SAWs. On the other hand, our value of Δ' agrees with field theory estimates [8]. We thus conjecture that the *leading* correction was missed in [8] and decouples for some unknown reason from R_N in the excluded needle geometry.

Using these corrections to scaling we can now estimate ν simply by extrapolating to $N \to \infty$ in figure 3. From the excluded needle data we get an estimate which is even somewhat lower than that of [13] (which is already lower than previous estimates), but compatible with it. For ordinary SAWs our extrapolation would be even smaller (by a considerable amount), but this extrapolation is much less certain because of the smaller Δ (see figure 3). Our final estimate
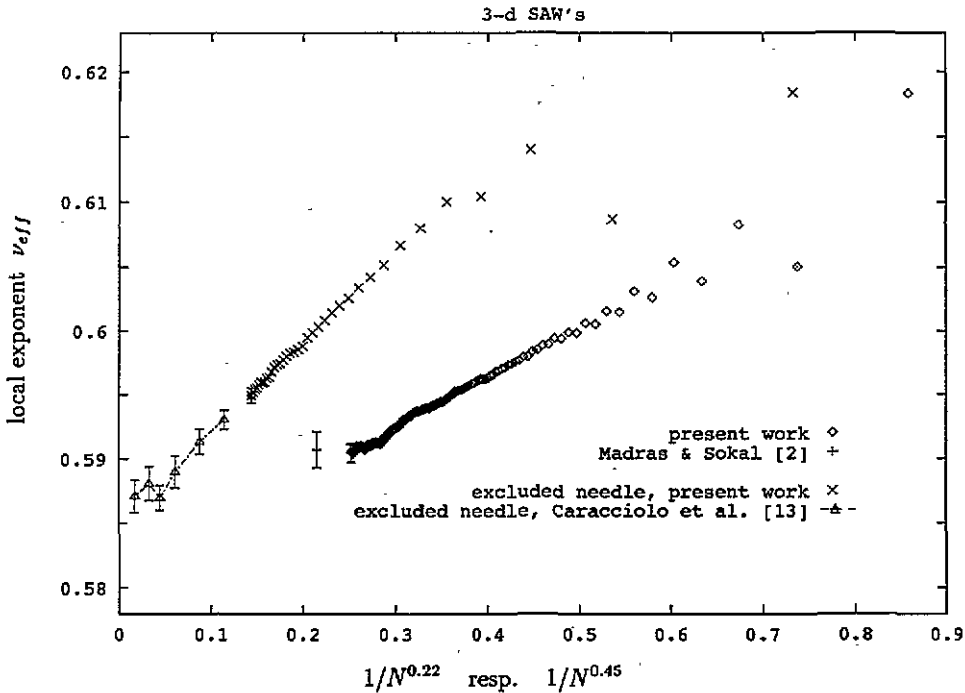
$$\nu = 0.585 \pm 0.0015 \tag{17}$$

Figure 3. Effective critical exponents defined in (16). On the horizontal axis $N^{-\Delta}$ with $\Delta = 0.22$ for ordinary SAWs ($\Diamond$, $+$), respectively $\Delta' = 0.45$ for an excluded needle ($\times$, $\triangle$) is shown. Error bars on our data are only given for the largest values of N, for smaller N they are typically smaller than the symbols.

is thus entirely based on the excluded needle data, and does not use the data from ordinary SAWs at all.

Finally let me make some comments on the (pseudo-) RNG used in the above simulations. This work was started with the Kirkpatrick–Stoll RNG [15] $x_n = x_{n-103}$ XOR x_{n-250}. This is a very popular RNG with very long periodicity (2^{250}) and no documented deviations from uniformity. While I had used it extensively for non-recursive algorithms with no apparent problems, problems (though statistically not quite significant) first appeared in a recursive (depth-first) study of 2D percolation backbones [16]. In the present simulations it gave *very* significant deviations which were already evident after $\approx 4 \times 10^9$ calls (corresponding to approximately 10 h CPU time on the low-cost work stations used for this work). This deviation was in the form of an excess in n_k at $k \approx 20$, and a dip at $k \approx N - 20$. It seems that the origin of these problems are triple correlations with characteristic times $\approx 10^2 - 10^3$. After completion of this work, I learned that similar problems with the above RNG were also recently observed in [17]. The results presented above were partly obtained with a more complicated shift register generator taken from [18], $x_n = x_{n-157}$ XOR x_{n-314} XOR x_{n-471} XOR x_{n-9689}, and partly by 'XOR-ing' the integer outputs from two simpler RNGs. Both methods seem to be safe for studies of the present size ($\approx 10^{12}$ RNG calls altogether).

In summary, we have used a novel recursive sampling method to obtain the most precise values to date for the connectivity constant of SAWs on a simple cubic lattice, and for the end-to-end distances at intermediate chain lengths. Combining this estimate of the connectivity constant with previous exact enumerations, we obtain what seems to be the most precise

estimate of γ. Combining our values of R_N with those for much longer chains obtained by means of the pivot algorithm, we obtained new and surprising values for the correction to scaling exponents Δ and Δ'. These, in turn, suggest that ν might be even lower than the value given in [13] which was already lower than previous estimates.

The efficiency of the recursive sampling method for estimating the connectivity constant results from a very efficient use of the random number generator. This also means, however, that the method is extremely sensitive to correlations in the random numbers.

References

[1] de Gennes P G 1979 *Scaling Concepts in Polymer Physics* (Cornell University: Cornell University Press)
[2] Madras N and Sokal A D 1988 *J. Stat. Phys.* **50** 109
[3] Beretti A and Sokal A D 1985 *J. Stat. Phys.* **40** 483
[4] Grassberger P 1993 *J. Phys. A: Math. Gen.* **26** 1023
[5] Redner S and Reynolds P J 1981 *J. Phys. A: Math. Gen.* **14** 2679
[6] Hegger R and Grassberger P in preparation
[7] MacDonald D, Hunter D L, Kelly K and Jan N 1992 *J. Phys. A: Math. Gen.* **25** 1429
[8] LeGuillou J C and Zinn-Justin J 1977 *Phys. Rev. Lett.* **39** 95
[9] Grassberger P 1989 *J. Phys. A: Math. Gen.* **22** 3673
[10] Kohring G A 1990 *Int. J. Mod. Phys.* C **1** 259
[11] Guttmann A J 1989 *J. Phys. A: Math. Gen.* **22** 2807
[12] Rapaport D C 1985 *J. Phys. A: Math. Gen.* **18** 113
[13] Caracciolo S, Ferraro G and Pelisetto A 1991 *J. Physique* **24** 3625
[14] Considine D and Redner S 1989 *J. Phys. A: Math. Gen.* **22** 1621
[15] Kirkpatrick S and Stoll E P 1981 *J. Comput. Phys.* **40** 517
[16] Grassberger P 1992 *J. Phys. A: Math. Gen.* **25** 5475
[17] Ferrenberg A M, Landau D P and Wong Y J 1992 *Phys. Rev. Lett.* **69** 3382
[18] Ziff R M and Stell G 1988 *Report #LaSC 88-4 University of Michigan* unpublished