Path integral Monte Carlo using multigrid techniques ☆

Wolfhard Janke [a] and Tilman Sauer [b]

[a] *Höchstleistungsrechenzentrum, Forschungszentrum Jülich, Postfach 1913, W-5170 Jülich, Germany*
[b] *Institut für Theoretische Physik, Freie Universität Berlin, Arnimallee 14, W-1000 Berlin 33, Germany*

Received 28 August 1992

Monte Carlo simulations of euclidean path integrals based on local update algorithms are severely hampered by diverging autocorrelation times τ in the continuum limit. If ϵ denotes the discretization parameter we verify for simulations with the standard Metropolis algorithm the theoretically expected behaviour $\tau \propto (1/\epsilon)^2$. We then demonstrate numerically that this problem can be overcome by using mathematically very well-founded multigrid techniques.

1. Introduction

In the past few years considerable progress has been made in developing efficient update algorithms for Monte Carlo simulations in statistical mechanics and lattice field theory (for reviews, see ref. [1]). These improvements all aim at overcoming the well-known problem of "critical slowing down" that severely hampered simulations based on standard *local* update algorithms like, e.g., the Metropolis or the heat-bath update. For local algorithms the autocorrelation time τ_0 (measuring the number of sweeps through the lattice needed to get sufficiently uncorrelated configurations) grows with the correlation length ξ of the system as $\tau_0 \propto \xi^z$, with a dynamical critical exponent $z \approx 2$, as can be argued in a simple random walk picture. At a continuous phase transition ξ tends to infinity, and on lattices with linear size L the autocorrelation time behaves as $\tau_0 \propto L^z$, where it is usually assumed that the exponent is the same as before.

Most of the new algorithms (an exception is overrelaxation [2]) overcome this problem by proposing *global* update steps which allow to sample the phase space more rapidly. Particularly successful are cluster algorithms [3] and multigrid techniques [4–6]. While the algorithms of the first class make use of special properties of the model under consideration and are therefore, if applicable, usually more successful, multigrid techniques have the advantage of being very general.

In path integral Monte Carlo (PIMC) simulations [7–15] (for reviews, see ref. [16]) a problem very similar to critical slowing down occurs in the continuum limit and for its solution a number of modified algorithms have been proposed over the last years [10–13]. Only few investigations, however, are devoted to quantitative comparisons of the performance of these schemes [14,15] and, equally important, little is known about their mathematical foundations. This fact motivated us to test to which extent the very well founded new algorithms of statistical mechanics and field theory can be transferred to PIMC applications. Due to lack of space we shall concentrate in this Letter only on the multigrid approach. A full account of our investigations will be given elsewhere.

2. Path integral Monte Carlo

We want to simulate a quantum partition function $\mathcal{Z}$ at inverse temperature β expressed as a discretized path integral consisting of $N = 2^n$ "beads"

$$\mathcal{Z}^{(n)}(\beta) = \left(\prod_{j=1}^{N} \int \frac{\mathrm{d}x_j}{A} \right) \exp(-\mathcal{A}^{(n)}) , \qquad (1)$$

☆ Work supported in part by Deutsche Forschungsgemeinschaft under grant KI256.

with an action

$$\mathcal{A}^{(n)} = \epsilon \sum_{j=1}^{N} \left[\frac{1}{2} \left(\frac{x_j - x_{j-1}}{\epsilon} \right)^2 + V(x_j) \right], \qquad (2)$$

where V is a potential to be specified later on, $A = \sqrt{2\pi\epsilon}$, $\epsilon = \beta/N$, $x_0 = x_N$, and $\hbar = k_B = 1$.

The slowing down problem occurs in the continuum limit $\epsilon \to 0$ with β fixed, or, equivalently, $N \to \infty$. The reason for slowing down in PIMC is easily understood. The correlations $\langle x_i x_{i+j} \rangle$ only depend on β and on the gaps between the energy levels. Hence the correlation length ξ only depends on the physical parameters at hand, and consequently always diverges linearly with N if measured in units of the lattice spacing ϵ of the discretized path integral. Thus for local algorithms one expects that the autocorrelation time grows as N^z with $z \approx 2$. Note that in contrast to the infinite volume limit in statistical mechanics, in PIMC also the action changes its form in the limit $N \to \infty$ which is the distinguishing feature of PIMC. As a consequence slowing down occurs in the continuum limit for *any* fixed β and *any* set of potential parameters.

3. Multigrid techniques

The basic idea of the multigrid approach [17] is to perform nonlocal updates by working on a set of successively coarser lattices in order to take into account long wavelength fluctuations more systematically. Because of the idea to do updates on a sequence of coarsened grids the approach has been called *multigrid* Monte Carlo [4]. Equivalently, however, one may also look at the multigrid method from a *unigrid* point of view as a way to systematically update long wavelength components on the *original* grid. We will first describe the method from this *unigrid* point of view.

We define a sequence of update levels $n, \ldots, 0$ by the following prescription. On level n, the finest level, we would do Monte Carlo updates bead by bead as usual. On level $n-1$, however, we would move always two neighbouring beads at a time. We would thus consider moves $x_{2i-1} \to x_{2i-1} + \delta, x_{2i} \to x_{2i} + \delta$ with the same displacement δ and accept (or reject) the move only for both sites in conjunction. On level $n-2$ we would move always four beads at a time, and so on. Updates on level 0 clearly would just be collective moves of the whole chain.

Having defined these levels it is two main features that distinguish the multigrid approach. The first is a mere technical one, at least from the unigrid point of view which we emphasize here, and concerns the question of how the update on the different levels can efficiently be implemented. The second and more important aspect has to do with the sequence in which the various levels are updated.

Of course it would be possible to program the updates on the coarsened levels simply in the way we have just described them, and in fact we have done that to check the more involved multigrid case. The *multigrid* approach, however, allows one to implement these updates more efficiently. In order to do so we define a sequence of *coarsened grids* $\Xi^{(k)}$, $k = n-1, \ldots, 0$ of size 2^k. On these grids we define auxiliary variables $x_i^{(k)}$ which will turn out to be nothing else than the combined trial moves. We then prescribe an interpolation scheme taking the variables from level $k-1$ to k. Using the simplest case of piecewise constant interpolation we would have

$$\mathcal{P}\boldsymbol{x} = (x_1, x_1, x_2, x_2, \ldots, x_{2^{k-1}}, x_{2^{k-1}})^{\mathrm{T}}, \qquad (3)$$

where we have suppressed the superscripts $(k-1)$. Instead of performing the updates on the various levels by working on the original variables of the finest grid we can now implement the updates on these coarser grids with their auxiliary variables. In order to do so we define coarsened actions on these grids by the following simple prescription

$$\mathcal{A}^{(k-1)}(\boldsymbol{x}_l^{(k-1)}) = \mathcal{A}^{(k)}(\boldsymbol{x}_l^{(k)} + \mathcal{P}\boldsymbol{x}_l^{(k-1)}) . \qquad (4)$$

In essence this prescription defines an action on the coarse grid $\Xi^{(k-1)}$ by fixing the variables $x_i^{(k)}$ of the next finer grid $\Xi^{(k)}$ and calculating the effect of moves represented by the variables $x_i^{(k-1)}$ added onto $\Xi^{(k)}$ using the interpolation scheme (3). Note that by updating the variables of the coarse grids and interpolating them back onto the variables of the fine grid one performs exactly the above outlined combined move prescription. The efficiency of this implementation is enhanced by the fact that in many applications the action on level $k-1$ has the same functional form as the action on level k. In particular this is the case for the piecewise constant interpo-

lation scheme and for a polynomial potential which we shall consider here.

The second feature of the multigrid approach concerns the sequence in which the updates on the various levels are to be performed. In principle any sequence would be possible. Thus one could do, say, ten updates on level n followed by one update on level 0. Or one could do three updates on level n, followed by three updates on level $n-1$, followed by three updates on level $n-2$, and so on. Also one could define some random sequence of levels. Most of the multigrid applications, however, follow a scheme under the name of *V-cycle* or *W-cycle*, respectively. These schemes derive from the following recursive definition of the *multigrid algorithm*.

Suppose we are updating the variables of grid $\Xi^{(k)}$. The multigrid algorithm then consists of (a) n_1 sweeps using any valid local update algorithm. (b) If the coarsest level has not yet been reached one would fix the configuration for $\Xi^{(k)}$ and calculate the action for the next coarser grid $\Xi^{(k-1)}$. The variables $\Xi^{(k-1)}$ could then be updated with this conditional action again applying the *multigrid algorithm* repeatedly γ_k times starting from an initial configuration $x_i^{(k-1)} \equiv 0$. Otherwise one would simply go on with (d). (c) After performing the update procedure on $\Xi^{(k-1)}$ one would then interpolate the variables of $\Xi^{(k-1)}$ back onto the finer grid using the interpolation scheme defined above, i.e. $x_{\text{new}}^{(k)} = x_{\text{old}}^{(k)} + \mathcal{P}x^{(k-1)}$, and (d) perform another n_2 sweeps on the grid $\Xi^{(k)}$ with the local update algorithm. Note that at (b) the multigrid algorithm is defined recursively. The numbers γ_k in this algorithm control the sequence in which the various levels are visited. The above mentioned V- and W-cycles correspond to the special cases $\gamma_k \equiv 1$ or $\gamma_k \equiv 2$, respectively. These names are obvious from a graphical representation of the so defined sequence of level [#1].

At this point it should be pointed out that multigrid techniques have first been designed to solve partial differential equations [#2], and that there exists an extensive literature dealing with convergence properties of multigrid algorithms [17,18]. The fact that there is a close analogy between deterministic problems and Monte Carlo simulations allows one to transfer some of the rigorous convergence proofs for deterministic algorithms to statements about Monte Carlo autocorrelation times [4]. In particular it has been shown that for the W-cycle with piecewise constant interpolation when applied to quadratic potentials the autocorrelation times are independent of the number of beads N, and hence critical slowing down is completely eliminated. To our knowledge there are, however, no rigorous statements for non-quadratic potentials. In this note we have therefore investigated this case numerically.

4. Results

We simulated the path integral (1), (2) for the two characteristic potential shapes covering a wide range of physical phenomena, namely a convex potential (CP), relevant for studying fluctuations around a unique minimum, and a double-well potential (DW), exhibiting tunneling phenomena. More precisely, we studied the two potentials $V = 0.5x^2 + x^4$ (CP) and $V = -0.5x^2 + 0.04x^4$ (DW), but the qualitative behaviour of our results clearly does not depend on the specific form of these potentials.

We simulated the path integral for grids of size $2^3 = 8$ up to $2^9 = 512$ sites using either the standard local Metropolis algorithm [19] (M) or the multigrid algorithm with all $\gamma_k \equiv 1$ (V) or all $\gamma_k \equiv 2$ (W). In the multigrid case we always had $n_1 = 1$, and $n_2 = 0$, i.e. on each grid we performed one presweep and no postsweep. To take full advantage of the features of a vector computer we worked with a checkerboard-like implementation. The acceptance rates for the finest grid were adjusted to be roughly 0.5, we found that there was no need to adjust δ for the coarsened grids since the acceptance rates varied only weakly over the different levels. In all our simulations β was equal to 10.

An advantage of the simple one-dimensional one-particle system investigated here is that in the continuum limit all expectation values converge to values which can easily be obtained by other methods such as numerical solutions of the Schrödinger equation. To check the accuracy of our results we measured moments $\langle x_i^q \rangle$, $q = 1, \ldots, 4$, correlations

[#1] See, e.g., ref. [17], p. 33.

[#2] For a brief historical account, see ref. [17], ch. 2.6.5.

$\langle x_i x_{i+j} \rangle$, and the internal energy using the virial estimator $E = \langle \frac{1}{2} x_i V'(x_i) + V(x_i) \rangle$ [8,9], and convinced ourselves that the program reproduces the theoretically expected continuum values for large grids within the statistical errors. For the convex potential the continuum energy is 0.80377, in our simulations the measured energies are compatible with this value for $N=256$, the values being 0.8024(20) (M), 0.8011(13) (V), 0.8027(7) (W), and for $N=512$, the values being 0.8004(26) (M), 0.8042(18) (V), and 0.8032(7) (W). For the double-well potential we again find the measured energies compatible with the continuum energy which is -0.90397, the values being $-0.9032(19)$ (V), and $-0.9068(20)$ (W) for the $N=512$ grid. These values are obtained on the basis of N_m measurements, the values of N_m being given below.

Since the main focus of our investigation was to test the relative performance of the different algorithms we have taken care to measure precisely the autocorrelation times for the Metropolis and the multigrid algorithm. As mentioned above it is the autocorrelation time τ_0 which characterizes the (pseudo) dynamics of the slowest mode in the Monte Carlo process. Explicitly, the autocorrelation function $A(k)$ of an observable $\mathcal{O}$ is defined by

$$A(k) = \frac{\langle \mathcal{O}_i \mathcal{O}_{i+k} \rangle - \langle \mathcal{O}_i \rangle^2}{\langle \mathcal{O}_i^2 \rangle - \langle \mathcal{O}_i \rangle^2}, \tag{5}$$

where $\mathcal{O}_i$ stands short for the ith measurement of $\mathcal{O}$. The autocorrelation time τ_0 then follows from the asymptotic behaviour for large k, $A(k) \propto \exp(-k/\tau_0)$. In addition each observable $\mathcal{O}$ is associated with its own so-called integrated autocorrelation time τ, defined by the area under the autocorrelation function of this observable,

$$\tau \equiv \tfrac{1}{2} + \sum_{k=1}^{\infty} A(k) . \tag{6}$$

The integrated autocorrelation time usually behaves qualitatively as τ_0 and can be shown to enter in the estimate for the statistical error of mean values as $\Delta\hat{\mathcal{O}} = \sqrt{2\tau}\sqrt{\sigma^2/N_m}$, where σ^2 is the observable's variance and N_m the number of measurements used to compute the mean value $\hat{\mathcal{O}}$. The effective statistics is thus reduced to $N_{\text{eff}} = N_m/2\tau$. Or, in other words, to achieve a given error $\Delta\hat{\mathcal{O}}$, the run-time (i.e. the budget) has to be increased by a factor of 2τ.

In our simulation we performed N_e "empty" updates in between each measurement thereby adjusting the autocorrelation time scale so that $\hat{\tau} \equiv \tau/N_e$ never exceeded a value of 50 successive measurements. In addition we have always performed $N_e \times 5000$ updates without measuring in order to reach thermal equilibrium starting from an ordered configuration $x_i \equiv 0$. In our log–log plots the error bars for τ, calculated by the jackknife method [20], are smaller than the data symbols.

In order to get a solid basis for comparisons we first looked at the autocorrelation times of the standard Metropolis algorithm. Fig. 1 shows our results for the moments $\bar{x} \equiv \sum x_i/N$ and $\overline{x^2} \equiv \sum x_i^2/N$, both for the convex potential (filled symbols) and for the double-well potential (open symbols) on the basis of $N_m = 500000$ (CP) and $N_m = 100000$ (DW) measurements, respectively. The autocorrelation times for the odd moments $\tau_{\bar{x}}$ and for the even moments $\tau_{\overline{x^2}}$ only differ by a constant factor while the N dependence in both cases clearly shows the expected behaviour of $\tau \propto N^2$. While in the case of the double-well potential the autocorrelation time for $\overline{x^2}$ is not very much increased in comparison with the convex case, it is the average position of the path $\bar{x}$ which becomes drastically autocorrelated. Already for the $N=128$ grid we found an autocorrelation time of $2.36(47) \times 10^6$. Since the N^2 behaviour sets in only for sufficiently large grids, a fit $\tau = \alpha N^z$ expectedly underestimates z to be only 1.855(55). On the basis of this fit we estimate that for our largest grid with $N=512$ one would need at least $2\tau = 5.2 \times 10^7$ sweeps of the primitive Metropolis algorithm in order to obtain one statistically independent configuration. As to the higher moments we found that the autocorrelation times for $\overline{x^3}$ are in general only slightly smaller than those for $\bar{x}$, and the same is true for $\overline{x^4}$ and $\overline{x^2}$. Note that to obtain the internal energy using the virial estimator (for the quartic potentials considered here) only the even moments are needed. If, however, one would want to look at the level splitting which is essentially governed by the tunneling events one would also have to determine the odd moments accurately.

Let us now look at the results for the multigrid algorithm. Fig. 2 shows that for the convex potential the exponent z can considerably be reduced with the multigrid approach. Here again we had $N_m = 500000$.

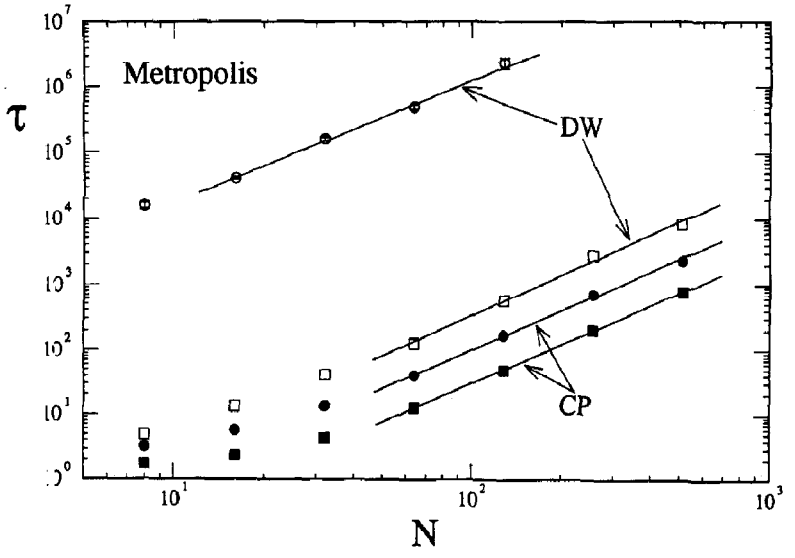

Fig. 1. Autocorrelation times for the convex (CP) ((●) $\bar{x}$; (■) $\overline{x^2}$) and the double-well (DW) ((○) $\bar{x}$; (□) $\overline{x^2}$) potential using the standard Metropolis algorithm. Solid lines are fits $\tau = \alpha N^z$ with an exponent z of 1.855(55) (○), 2.026(90) (□), 1.989(20) (●), and 1.990(13) (■).

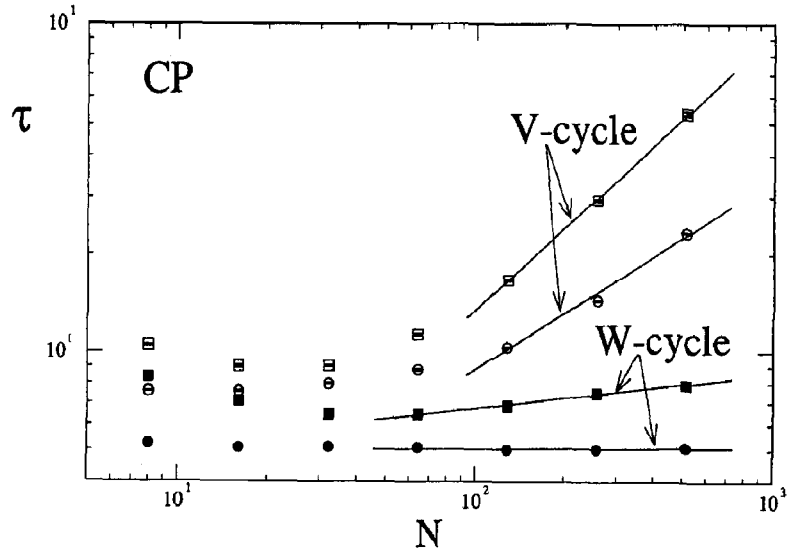

Fig. 2. Autocorrelation times for the convex potential using the V- ((○) $\bar{x}$; (□) $\overline{x^2}$) and W- ((●) $\bar{x}$; (■) $\overline{x^2}$) cycle multigrid algorithm. Solid lines are fits $\tau = \alpha N^z$ with an exponent z of 0.8356(92) (□), 0.5835 (48) (○), 0.1043(29) (■), and 0.0043(29) (●).

While for the W-cycle critical slowing down appears to be completely eliminated at least for $\tau_{\bar{x}}$, the critical exponent for the V-cycle seems to approach a value close to 1. Observe that with the multigrid algorithm the odd moments are more effectively updated than the even ones, thus $\bar{x}$ becomes even less autocorrelated than $\overline{x^2}$. As to the absolute values of τ, note the different scale of fig. 2 as compared to fig. 1.

Turning to the double-well potential, the next

question is whether topological features of the potential which would let the system reach different regions of the configuration space only by tunneling would destroy the efficiency of the multigrid algorithm. As fig. 3 shows, we find that the critical exponent z does not depend on the qualitative shape of the potential. Critical slowing down is completely eliminated for the W-cycle while for the V-cycle we obtain again values of z a little less than 1. In this case it is the prefactor α which separates the odd moments from the even ones. For the largest grid we found autocorrelation times $\tau_{\bar{x}} = 3.28(23) \times 10^4$ (V) and 269(17) (W). Thus the multigrid algorithm outperforms the primitive Metropolis algorithm by a factor of roughly 700 using the V-cycle and by a factor of almost 10^5 using the W-cycle.

Since no comparison of algorithms would be complete without a work estimate we finally remark that by counting the necessary operations one expects the work required for a full V-cycle to differ from the work for a Metropolis sweep by a constant factor, while the work required for a W-cycle compared with a Metropolis sweep should grow logarithmically with N [4]. Both these predictions could be qualitatively verified in our simulations. It is therefore ensured that the gain in efficiency by the reduction of z is not lost by an unlimited growth in computational effort for the multigrid cycles.

5. Concluding remarks

It has been shown that the multigrid approach to path integral Monte Carlo can completely eliminate the costly phenomenon of "critical slowing down" encountered in approaching the continuum limit of the discretized path integral. While in the past a number of modifications of the primitive Metropolis scheme has already been proposed to reduce slowing down by a similar strategy of incorporating nonlocal update moves [10–13] the multigrid approach differs by two important features. First, convergence properties of the multigrid idea have been extensively investigated in the mathematical literature in the application to deterministic problems, and recently also in applications to Monte Carlo simulations. It is therefore a well founded method and its convergence properties have in some cases been rigorously demonstrated. The second outstanding feature of the multigrid approach is its generality. While in this investigation the principal applicability of the multigrid method to path integral Monte Carlo has been demonstrated in simple test examples, it is con-

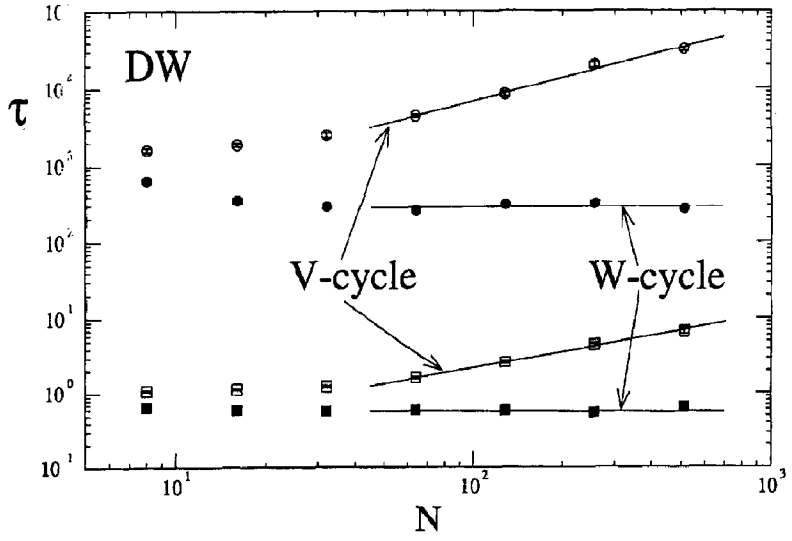

Fig. 3. Autocorrelation times for the double-well potential using the V- ((○) $\bar{x}$; (□) $\overline{x^2}$) and W- ((●) $\bar{x}$; (■) $\overline{x^2}$) cycle multigrid algorithm. Solid lines are fits $\tau = \alpha N^z$ with an exponent z of 0.975(37) (○), 0.013(43) (●), 0.715(27) (□), and −0.015(11) (■).

ceivable that it can also be used for more complicated and realistic path integral systems. An interesting application would be an investigation of quantum chains [21] or quantum crystals [22].

References

[1] R.H. Swendsen, J.-S. Wang and A.M. Ferrenberg, in: The Monte Carlo method in condensed matter physics, New Monte Carlo methods for improved efficiency of computer simulations in statistical mechanics, ed. K. Binder (Springer, Berlin, 1991);
A.D. Sokal, Monte Carlo methods in statistical mechanics: foundations and new algorithms, Cours de Troisième Cycle de la Physique en Suisse Romande, Lausanne (1989).

[2] M. Creutz, Phys. Rev. D 36 (1987) 515;
R. Gupta, J. DeLapp, G.G. Battrouni, G.C. Fox, C.F. Baillie and J. Apostolakis, Phys. Rev. Letters 61 (1988) 1996;
S.L. Adler, Phys. Rev. D 37 (1988) 458;
H. Neuberger, Phys. Lett. B 207 (1988) 461.

[3] R.H. Swendsen and J.-S. Wang, Phys. Rev. Letters 58 (1987) 86;
J.-S. Wang and R.H. Swendsen, Physica A 167 (1990) 565;
U. Wolff, Phys. Rev. Letters 62 (1989) 361; Nucl. Phys. B 334 (1990) 581.

[4] J. Goodman and A.D. Sokal, Phys. Rev. Letters 56 (1986) 1015; Phys. Rev. D 40 (1989) 2035.

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

[6] E. Loh Jr., Multigrid Monte Carlo methods, in: Computer simulation studies in condensed matter physics, ed. D.P. Landau et al. (Springer, Berlin, 1988).

[7] S.V. Lawande, C.A. Jensen and H.L. Sahlin, J. Comput. Phys. 3 (1969) 416;
J.A. Barker, J. Chem. Phys. 70 (1979) 2914;
E.V. Shuryak and O.V. Zhirov, Nucl. Phys. B 242 (1984) 393.

[8] M. Creutz and B. Freedman, Ann. Phys. 132 (1981) 427.

[9] M.F. Herman, E.J. Bruskin and B.J. Berne, J. Chem. Phys. 76 (1982) 5150.

[10] K.S. Schweizer, R.M. Stratt, D. Chandler and P.G. Wolynes, J. Chem. Phys. 75 (1981) 1347.

[11] E.L. Pollock and D.M. Ceperley, Phys. Rev. B 30 (1984) 2555.

[12] M. Sprik, M.L. Klein and D. Chandler, Phys. Rev. B 31 (1985) 4234.

[13] J.D. Doll, D.L. Freeman and T.L. Beck, Advan. Chem. Phys. 78 (1990) 61.

[14] A. Giansanti and G. Jacucci, J. Chem. Phys. 89 (1988) 7454.

[15] J.S. Cao and B.J. Berne, J. Chem. Phys. 91 (1989) 6359.

[16] B.J. Berne and D. Thirumalai, Ann. Rev. Phys. Chem. 37 (1986) 401;
D. Chandler, Introduction to modern statistical mechanics (Oxford Univ. Press, New York, 1987) ch. 6.

[17] W. Hackbusch, Multi-grid methods and applications (Springer, Berlin, 1985).

[18] S.F. McCormick, ed., Multigrid methods. Theory, applications, and supercomputing (Dekker, New York, 1988).

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

[20] R.G. Miller, Biometrica 61 (1974) 1;
B. Efron, The jackknife, the bootstrap and other resampling plans (SIAM, Philadelphia, PA, 1982).

[21] G.M. Wysin and A.R. Bishop, Phys. Rev. B 34 (1986) 3377;
A. Cuccoli, V. Tognetti, P. Verrucchi and R. Vaia, Florence preprint DFF 159/3/92.

[22] A. Cuccoli, A. Macchi, M. Neumann, V. Tognetti and R. Vaia, Phys. Rev. B 45 (1992) 2088.