International Journal of Thermophysics, Vol. 28, No. 2, April 2007 (© 2007)
DOI: 10.1007/s10765-007-0193-z

Sequential Metropolis Algorithms for Fluid Simulations

Ruichao Ren,[1] C. J. O'Keeffe,[1] and G. Orkoulas[1,2]

In this work, the implementation of our recently proposed sequential Metropolis algorithm in the grand canonical ensemble, a case particularly relevant for continuum fluids, is considered. By performing Monte Carlo simulations for the two-dimensional lattice gas, it is shown that our algorithm converges faster than all known grand canonical algorithms that satisfy strict detailed balance. The main advantages of the new algorithm are its simplicity, generality, and the possibility of parallel implementation.

KEY WORDS: detailed balance; lattice gas; molecular simulation.

1. INTRODUCTION

Molecular simulation techniques have been extensively used as alternatives to analytical techniques in estimating fluid properties [1–3]. The pioneering work of Metropolis et al. [4] on the Monte Carlo method is based on the theory of Markov processes and the so-called principle of "detailed balance". Random updating of particles (or spins for magnetic systems) ensures that strict detailed balance (microscopic reversibility) is satisfied at any time. While it is known that strict detailed balance is a sufficient but not necessary condition for convergence to equilibrium [3, 5], no systematic attempts have been made to investigate alternative possibilities of improvement.

Recently, we proposed a Monte Carlo algorithm based on sequential updating moves with partial randomness [6]. The new algorithm breaks the constraint of strict reversibility, and it only satisfies the weaker "balance condition" [5, 6]. Analytical results based on transition

[1] Department of Chemical and Biomolecular Engineering, University of California, Los Angeles, California 90095, U.S.A.
[2] To whom correspondence should be addressed. E-mail: makis@seas.ucla.edu

matrix theory indicate that the new algorithm converges faster than the Metropolis algorithm with strict detailed balance. Monte Carlo simulations for the two-dimensional Ising model show that the new algorithm improves the sampling quality dramatically by reducing the autocorrelation of samples. The enhancement was attributed to the nature of the updating; for the ferromagnetic Ising model, the so-called "neighbor" effect was identified as the main contributing factor of acceleration: successful flip of a given spin triggers a much higher probability of flipping a neighboring spin in the same direction as the precursor. Sequential updating takes advantage of this so-called "neighbor" effect and results in a cascaded enhancement of sampling mobility which in turn improves the statistical efficiency of sampling.

In this work, we consider the algorithm of Ref. [6] for the well-known lattice gas in the grand canonical ensemble. This is done by exploring the equivalence of the Ising model to the lattice gas. Although several grand-canonical-type algorithms exist for continuum fluid models [7–10], there is no universally accepted grand canonical algorithm capable of achieving Ising-like precision for realistic molecular systems. As for the two-dimensional nearest-neighbor lattice gas on the square lattice, we find that the sequential algorithm is superior to the Metropolis-type of algorithm based on random updates as well as standard algorithms used for continuum fluid models [7–9]. The next attempt in the direction of sequential-type of algorithms is generalization for continuum fluid models.

2. MARKOV PROCESS

Consider a stochastic process that involves transitions between different states at discrete time intervals $0, \delta t, 2\delta t, \ldots$, where δt is a time interval associated with successive transitions. A Markov process is defined as the process for which the transition probability from state i to state j, P_{ij}, only depends on the two states and not on previous history. In matrix notation, $P = \{P_{ij}\}$ denotes the transition matrix (kernel) of the Markov process. Due to conservation of probability, each row of P sums to unity. Thus, in matrix notation, we may say

$$P \cdot \mathbf{1} = \mathbf{1} \tag{1}$$

where $\mathbf{1}$ is the column vector $\{1, 1, \ldots, 1\}$. If the initial probability distribution of states is $\boldsymbol{g}(0) = \{g_1(0), g_2(0), \ldots\}$, then after one transition

$$\boldsymbol{g}^T(1) = \boldsymbol{g}^T(0) \cdot P \tag{2}$$

where $\boldsymbol{g} = \{g_1, g_2, \ldots\}$ stands for a column vector and $\boldsymbol{g}^T$ (row vector) is the transpose of $\boldsymbol{g}$. The long-time limit of sequence Eq. (2) defines the stationary (equilibrium) distribution of states $\boldsymbol{\pi} = \{\pi_1, \pi_2, \ldots\}$,

$$\lim_{n\to\infty} \boldsymbol{g}^T(n) = \boldsymbol{g}^T(0) \cdot \lim_{n\to\infty} P^n = \boldsymbol{\pi}^T. \tag{3}$$

If the previous limit exists, further iterations leave the probability distribution invariant, i.e.,

$$\boldsymbol{\pi}^T \cdot P = \boldsymbol{\pi}^T. \tag{4}$$

In order for the limit Eq. (3) to exist, the transition kernel P must also have a well-defined limit. The limiting form may be found by decomposing P in terms of its eigenvalues and eigenvectors via standard similarity transforms. Assuming that the Jordan canonical form of P is diagonal, we may expand P in terms of matrices $M^{(j)}$ containing its left and right eigenvectors associated with eigenvalue λ_j. Thus we have

$$P = \lambda_1 M^{(1)} + \lambda_2 M^{(2)} + \cdots. \tag{5}$$

In view of Eqs. (1) and (4), one sees that $\boldsymbol{\pi}$ and $\mathbf{1}$ are the left and right eigenvectors associated with eigenvalue $\lambda_1 = 1$. In addition, $|\lambda_j| < 1$ for $j = 2, 3, \ldots$, otherwise, the matrix elements of P^n diverge as $n \to \infty$: see Eq. (5). If $\lambda_1 = 1$ and $|\lambda_j| < 1$ for $j \geq 2$, Eq. (5) yields

$$\lim_{n\to\infty} P^n = M^{(1)}, \tag{6}$$

where the matrix elements of $M^{(1)}$ are

$$M^{(1)}_{mn} = \pi_n. \tag{7}$$

In this case, the kernel P is called "primitive" [11]. The Perron-Frobenius theorem [11] implies that, for a primitive stochastic matrix, $\lambda_1 = 1$ and $|\lambda_j| < 1$ for $j \geq 2$. Thus, a primitive Markov chain always converges to its unique stationary distribution. Convergence can also be proved for the case for which the Jordan canonical form of P is not diagonal [5, 12].

Primitivity is often confused with irreducibility. A transition matrix is called "irreducible" if any state can be reached from any other state *within* a finite number of steps [12]. Irreducibility alone does not guarantee the convergence of a Markov chain in the case that every state leads back to itself after a fixed number of steps (periodic chain). Whereas direct proof of primitivity of a given transition matrix is difficult, it can be shown that an irreducible and aperiodic matrix is always primitive [6]. Irreducible and aperiodic Markov chains are also called *ergodic* [13], and they converge to

a unique stationary limit if and only if the transition matrix P satisfies Eq. (4), or equivalently,

$$\sum_i \pi_i P_{ij} = \pi_j \ , \ \forall j. \tag{8}$$

Equation (4) or (8) is commonly referred to as the *balance* condition [2, 5].

3. METROPOLIS METHOD

The algorithm of Metropolis et al. [4] can be easily explained in the canonical ensemble. The probability π_i of state i is

$$\pi_i = \frac{1}{Q} e^{-\beta E_i} \tag{9}$$

with $\beta = 1/(k_B T)$. k_B is Boltzmann's constant, T is the temperature, and E_i is the energy of state i. The partition function that appears in Eq. (9) as the normalization constant,

$$Q = \sum_j e^{-\beta E_j} \tag{10}$$

cannot be found in a closed, analytical form, expect in very few cases. Metropolis et al. [4] devised a Markov process such that, after long times, each state j appears with a probability proportional to its statistical-mechanical weight π_j. Given a state i, a new state j is generated by a small perturbation (i.e., local particle move, spin flip). The transition probability P_{ij} from i to j is chosen to satisfy the so-called detailed balance condition [2, 3];

$$\pi_i P_{ij} = \pi_j P_{ji}. \tag{11}$$

Detailed balance is a sufficient but not necessary condition [3, 5] for convergence to equilibrium;

$$\pi_i P_{ij} = \pi_j P_{ji} \Rightarrow \sum_i \pi_i P_{ij} = \pi_j. \tag{12}$$

The transition probability P_{ij} is defined through two additional probabilities: the *a priori* or proposal probability q_{ij}, and the acceptance probability α_{ij};

$$P_{ij} = q_{ij} \alpha_{ij}, \quad \forall i \neq j \quad , \tag{13}$$

$$P_{ii} = 1 - \sum_{i \neq j} P_{ij} \quad . \tag{14}$$

The acceptance probability α_{ij}, from i to j, is defined as

$$\alpha_{ij} = \min\left\{1, \frac{q_{ji}}{q_{ij}} \frac{\pi_j}{\pi_i}\right\} \quad . \tag{15}$$

The previous choice of α_{ij} automatically satisfies the detailed balance condition Eq. (11), for all i and j. A transition matrix that satisfies Eq. (11) is also called a "reversible kernel" [14]. According to Peskun [14], the Metropolis transition kernel is the optimal one within reversible kernels in terms of statistical quality. The Metropolis choice of α_{ij}, Eq. (15), avoids prior knowledge of the normalization Q, Eq. (4). Since each state j appears with frequency proportional to π_j, an estimate for the ensemble average $\langle X \rangle$ of a fluctuating quantity X is given by the simple arithmetic average

$$\overline{X} = \frac{1}{M} \sum_{m=1}^{M} X_m \tag{16}$$

over M successive configurations.

4. RANDOM AND SEQUENTIAL METROPOLIS ALGORITHMS

In the standard implementation of the Metropolis algorithm, a new state j is generated from a current one i, by a perturbation in a randomly selected component. By component, we mean a particle for the case of fluid systems and a spin for the case of magnetic systems. A perturbation comprises a local displacement for particles and a change in the orientation for spin systems. The attempted move is accepted according to the criterion given in Eq. (15). The random selection of the component ensures that detailed balance is strictly satisfied at all times. Alternatively, we may select components sequentially and accept with the same Metropolis criteria. For the case of sequential updates, strict detailed balance is not satisfied since each move cannot be reversed immediately.

In Ref. [6], we investigated transition kernels based on sequential updates. Each component n $(n = 1, 2, \ldots, k)$ of the system has a transition matrix itself, which is denoted as $S^{(n)}$. The transition kernel for a sweep (k attempted moves) is the product of all sequential updating matrices. If the updating sequence is the same, the transition kernel of the sweep is

$$W = \prod_{n=1}^{k} S^{(n)} = S^{(1)} \cdot S^{(2)} \cdots S^{(k)} \quad . \tag{17}$$

Since each single-component trial move is accepted with Metropolis criteria (15), detailed balance is satisfied upon updating a single component,

$$\pi_i S_{ij}^{(n)} = \pi_j S_{ji}^{(n)} \quad , \tag{18}$$

which, in view of Eq. (12), implies that the balance condition $\boldsymbol{\pi}^T \cdot S^{(n)} = \boldsymbol{\pi}^T$ is also satisfied on updating a single component. For a sweep, we can write

$$\boldsymbol{\pi}^T \cdot W = \boldsymbol{\pi}^T \prod_{n=1}^{k} S^{(n)} = \boldsymbol{\pi}^T \cdot S^{(2)} \cdots S^{(k)} = \cdots = \boldsymbol{\pi}^T \tag{19}$$

which implies that the balance condition for every sweep is satisfied. However, since immediate reversal of a move is not possible within a sweep, detailed balance is not satisfied for the transition kernel of each sweep, i.e.,

$$\pi_i W_{ij} \neq \pi_j W_{ji} \quad . \tag{20}$$

In Ref. [6], we showed that the transition kernel for random updating, P, can be expressed as an arithmetic mean of the sequential transition matrices $S^{(n)}$,

$$P = \frac{1}{k} \sum_{n=1}^{k} S^{(n)} \quad . \tag{21}$$

The transition kernel that corresponds to a sweep based on purely random updates is P^k. Using the previous definitions and the arithmetic-geometric mean properties, it can be seen that [6]

$$[P^k]_{ii} > W_{ii} \quad . \tag{22}$$

$[P^k]_{ii}$ are the diagonal elements of the matrix P^k, i.e., the remaining and revisiting probability of state i after a sweep. For sequential updates, W_{ii} is the remaining probability in state i after a sweep, since revisiting a configuration is not allowed within the same sweep. Equation (22) shows that the diagonal elements of the sequential sweep kernel are always smaller than those of the random sweep kernel. According to a theorem by Peskun [14], the quality of an estimate can be increased by decreasing the magnitude of the diagonal elements. A transition kernel with smaller diagonal elements is expected to be less "sticky" with respect to the current state. Thus, ergodic kernels with smaller diagonal elements have enhanced "mobility" over the sample space and the Markov chain may converge to equilibrium faster.

A Markov chain based on purely sequential updates may not always be ergodic due to periodic effects [5,6]. In Ref. [6], we proposed an algorithm based on sequential updates with partial randomness that annihilates oscillatory behavior. A randomly selected lattice site is bypassed at regular intervals (i.e., every few sweeps) of sequential updating and hence the name "random skipping" (RS) sequential Monte Carlo algorithm. The randomness introduced by the bypassed lattice site is tantamount to rendering the transition kernel irreducible and aperiodic. The efficiency of the proposed algorithm was investigated in Ref. [6] for the case of the Ising model on the symmetry (zero-field) axis. In this work, we compare the algorithm of Ref. [6] with standard grand canonical Monte Carlo algorithms for the fluid analogue of the Ising model, i.e., the lattice gas.

5. METROPOLIS ALGORITHMS FOR THE LATTICE GAS

The well-known Ising model comprises spins residing on the sites j of a d-dimensional lattice of coordination number q, linear dimension L, and total number of sites N. The spins can take two values $\sigma_j = \pm 1$, $j = 1, 2, \ldots, N$. For a ferromagnetic system, the interaction energy between a pair of nearest-neighbor spins, σ_i and σ_j, is $-J\sigma_i\sigma_j$ where $J > 0$. The interaction of a spin σ_i with an external magnetic field B is $-B\sigma_i$. The energy and magnetization of a given configuration $\{\sigma_j\}$ of the N spins are

$$E = -J \sum_{\langle ij \rangle} \sigma_i \sigma_j \quad , \quad M = \sum_{j=1}^{N} \sigma_j \quad . \tag{23}$$

The notation $\langle ij \rangle$ stands for nearest-neighbor pairs. The partition function of the Ising model is

$$Z = \sum_{\{\sigma_j\}} e^{-\beta E + hM} \quad , \tag{24}$$

with $h = \beta B$.

The lattice gas is defined by considering the lattice points to be cells of volume υ which may be occupied by at most one molecule [15,16]. The activity, z, of the molecule is defined as

$$z = \frac{1}{\Lambda^d} e^{\beta\mu} \tag{25}$$

where μ is the chemical potential and Λ is the thermal De Broglie wavelength. Each cell j is described by the occupation variable $n_j = 0, 1$ depending on whether the cell is empty or occupied by a molecule. The potential energy of interaction between a pair of nearest-neighbor cells i

and j with occupation numbers n_i and n_j, respectively, is $-\epsilon n_i n_j$ with $\epsilon > 0$. The potential energy and the number of molecules for a given set of the occupation variables $\{n_j\}$ of the cells are

$$U = -\epsilon \sum_{\langle ij \rangle} n_i n_j \quad , \quad n = \sum_{j=1}^{N} n_j \quad . \tag{26}$$

The grand partition function of the lattice gas is thus

$$\Xi = \sum_{\{n_j\}} (z\upsilon)^n e^{-\beta U} \quad . \tag{27}$$

Since $n_j = (\sigma_j + 1)/2$, the Ising model and the lattice gas are equivalent if [15,16]

$$\epsilon = 4J \quad , \quad \ln(z\upsilon) = 2(h - q\beta J) \quad . \tag{28}$$

In this paper, we investigate the equivalence of the Ising model to the lattice gas in order to implement grand canonical types of moves (particle insertions and removals) with sequential updating. Our objective is twofold. First, we wish to achieve an enhancement similar to that seen in Ref. [6] for the Ising model. Second, we wish to compare sequential grand canonical updating algorithms with the standard grand canonical algorithms [7–9] that have been widely used for continuum fluid models.

For the Ising model, random updating corresponds to changing the spin variable ($\sigma_j \to -\sigma_j$) of a randomly selected lattice site j. The proposed spin flip is accepted according to Eq. (15) and since $q_{ij} = q_{ji}$,

$$\alpha_{ij} = \min \left\{ 1, \exp \left[h\Delta M - \beta \Delta E \right] \right\} \quad . \tag{29}$$

ΔE and $\Delta M = \pm 2$ are the changes of energy and magnetization. For the lattice gas, random updating amounts to changing the occupation variable n_j ($0 \to 1$ or $1 \to 0$) of a randomly selected cell j. The acceptance probability is given by

$$\alpha_{ij} = \min \left\{ 1, (z\upsilon)^{\Delta n} \exp \left[-\beta \Delta U \right] \right\} \tag{30}$$

where ΔU and $\Delta n = \pm 1$ are the changes of energy and number of particles. For an empty cell, $\Delta n = +1$ and the update corresponds to particle insertion. For an occupied cell $\Delta n = -1$ which corresponds to particle removal. For RS sequential updates, the lattice points are selected sequentially and the proposed moves are accepted with the same acceptance criteria. A randomly selected site (cell) is always bypassed at regular intervals to ensure ergodicity.

The grand canonical algorithms of Norman–Filinov and Adams [7–9] have been extensively used for continuum fluids. During the course of the simulation, one decides on the type of grand canonical move, particle insertion or removal, with equal probability. If the proposed move corresponds to particle insertion, a new particle is placed at a uniform random position within the volume V. For particle removal, a randomly selected particle is removed. Both attempted moves are accepted with Metropolis acceptance criteria, Eq. (15). Note that for this case the *a priori* probabilities q_{ij} and q_{ji} are not equal. We have considered this algorithm—thereby called conventional—for the two-dimensional lattice gas. Our sole purpose is to obtain an idea as to the relative merits and the speed of convergence of each algorithm. In our implementation of the Norman and Filinov algorithm, we consider particle insertions and removals with equal probability, i.e., 1/2. For the insertion ($n \to n+1$) step we select a cell (any one) and we thus have

$$q_{ij} = \frac{1}{2}\frac{1}{N} \quad , \quad q_{ji} = \frac{1}{2}\frac{1}{n+1} \quad . \tag{31}$$

The acceptance probability is found from Eq. (15);

$$\alpha_{ij} = \min\left\{1, \frac{N}{n+1} z\upsilon e^{-\beta\Delta U}\right\} \quad . \tag{32}$$

For the removal ($n \to n-1$) step we select one of the n particles with equal probability. Thus, we have

$$q_{ij} = \frac{1}{2}\frac{1}{n} \quad , \quad q_{ji} = \frac{1}{2}\frac{1}{N} \tag{33}$$

and

$$\alpha_{ij} = \min\left\{1, \frac{n}{N}\frac{1}{z\upsilon} e^{-\beta\Delta U}\right\} \quad . \tag{34}$$

Since the volume of the system is $V = N\upsilon$, the acceptance criteria Eqs. (32) and (34) are identical to those of Refs. [1–3] and [7–9].

6. CONVERGENCE TESTS

We have considered the lattice gas on the square lattice on an isotherm that corresponds to a reduced temperature of $T^* = k_B T/\epsilon = 0.6$. We consider nearest-neighbor interactions only, thus $q = 4$. The behavior on the critical isochore is known analytically [17], due to the equivalence of this model to the Ising model: see Eq. (28). The critical temperature is [17]

$$\frac{k_B T_c}{\epsilon} = \frac{1}{2\ln(1+\sqrt{2})} \approx 0.567 \quad . \tag{35}$$

The relative deviation from the critical isotherm is $(T - T_c)/T_c \approx 0.06$. If $\phi = \langle n \rangle / N$ is the fraction of occupied cells (volume fraction), the critical value of ϕ is $\phi_c = 1/2$ due to particle-hole symmetry.

First, we obtain the eigenvalues of a 2×2 system for all three algorithms (random, RS sequential, and conventional grand canonical) under investigation. The approach to the stationary limit is determined by the magnitude of the modulus of the second largest eigenvalue, $|\lambda_2|$. For the case of the RS sequential algorithm, we consider a kernel A that corresponds to a random skip for every sweep. Matrix A can be expressed in terms of the singe component matrices, $S^{(n)}$ as

$$A = \frac{1}{k} \sum_{l=1}^{k} \prod_{n \neq l}^{k} S^{(n)} \quad . \tag{36}$$

We also consider kernel P that corresponds to purely random updates, see Eq. (21), and kernel F that corresponds to the "conventional" algorithm. Instead of the second largest eigenvalue λ_2, we consider a closely related variable τ_2,

$$\tau_2 = -\frac{1}{\delta t} \frac{1}{\ln |\lambda_2|} \quad , \tag{37}$$

which has the meaning of relaxation or correlation time [18]. The parameter δt sets the scale of time in terms of elementary steps. For the 2×2 system, one lattice sweep is defined as $k = 4$ elementary updates. Kernel A is associated with $k - 1 = 3$ elementary updates, whereas kernels P and F both correspond to 1 elementary move. Therefore, if $\delta t = 1$ for A, then $\delta t = k - 1 = 3$ for P and F for the comparison to be meaningful.

Our results for τ_2 in terms of volume fraction ϕ are shown in Fig. 1. It can be seen that the curve that corresponds to sequential updating always lies below that of the random updating and the conventional algorithm. Based on analysis of a 2×2 system, one can infer that sequential updating algorithms exhibit the fastest convergence to equilibrium. Similar behavior is observed for different isotherms. We also note that due to the nature of the updates, the curves that correspond to random and sequential updates are symmetric about $\phi = 1/2$. The conventional algorithm does not distinguish between empty and occupied cells and thus does not respect the particle-hole symmetry of the lattice gas. As $\phi \to 1/2$, the correlation time τ_2 increases for all algorithms. This is to be expected, since the $\phi = 1/2$ symmetry axis corresponds to maximum lattice gas compressibility. More importantly, as $T \to T_c$, the compressibility exhibits a power law type of divergence. In the vicinity of T_c, the time required to generate statistically independent samples increases as a power law of the system size L, i.e., $\tau \sim L^{\omega}$ where ω is the dynamic exponent [1].

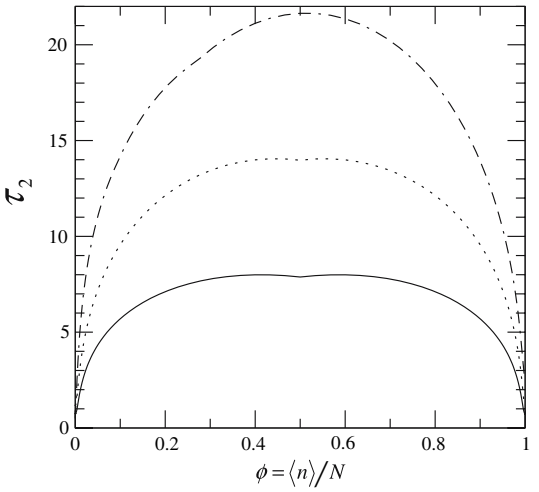

Fig. 1. Correlation time τ_2 as a function of volume fraction $\phi = \langle n \rangle / N$ for a 2×2 lattice gas model with $q = 4$ at $T^* = 0.6$. The dot-dashed line corresponds to the "conventional" grand canonical algorithm (see text), the dotted line corresponds to random updates, and the solid line corresponds to sequential updates with a random skip in every sweep.

We have also performed Monte Carlo simulations for a 100×100 lattice gas on the $T^* = k_B T/\epsilon = 0.6$ isotherm. In Fig. 2, we plot the volume fraction $\phi = \langle n \rangle / N$ and the energy density $u = \langle U \rangle / N$ vs. the reduced activity $z^* = z\upsilon$, for all three algorithms. For the case of sequential updates, we implemented a random skip in every other sweep. The efficiency of simulation algorithms can be inferred from measurements of equilibrium autocorrelation (or time-displaced) functions [1]. The autocorrelation function $\Phi_X(t)$ of a fluctuating variable X (such as energy, density or magnetization) is defined as

$$\Phi_X(t) = \frac{\langle X(0)X(t)\rangle - \langle X(0)\rangle^2}{\langle X(0)X(0)\rangle - \langle X(0)\rangle^2} \quad , \tag{38}$$

and it drops from $\Phi_X(0) = 1$ to zero as $t \to \infty$. The time scale, τ, characterizing the decay of $\Phi_X(t)$ at large times is the correlation or linear-relaxation time [1]. The behavior of the density autocorrelation function, $\Phi_n(t)$, is shown in Figs. 3 and 4. Figure 3 corresponds to a dilute system ($\phi \approx 0.1$) whereas Fig. 4 to an almost filled lattice ($\phi \approx 0.9$). It can be seen

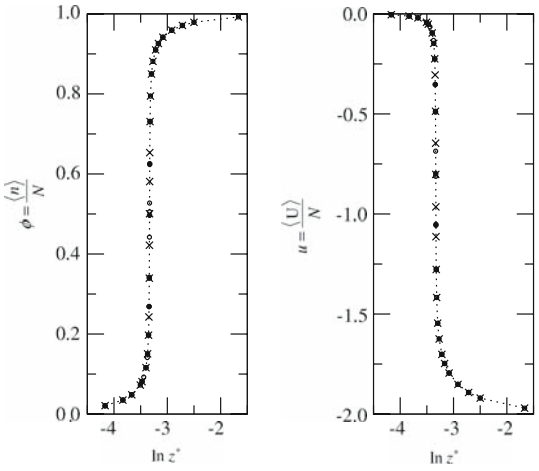

Fig. 2. Simulation results for the two-dimensional 100×100 lattice gas with $q = 4$ neighbors at $T^* = k_B T/\epsilon = 0.6$. Volume fraction $\phi = \langle n \rangle / N$ (left) and energy density $u = \langle U \rangle / N$ (right) vs. reduced activity $z^* = z\upsilon$. (+): sequential updates; (○): random updates; (●): conventional algorithm. Dotted lines are drawn for visual clarity.

that $\Phi_n(t)$ of the RS sequential algorithm exhibits the fastest decay, which implies that sequential updating yields samples of the highest quality. Similar behavior is also observed for other autocorrelation functions such as energy, $\Phi_U(t)$.

We have estimated the integrated relaxation time τ, defined as [1]

$$\tau = \int_0^\infty dt\, \Phi_X(t) \quad , \tag{39}$$

from the density autocorrelation function $\Phi_n(t)$ for all the simulated state points shown in Fig. 2. Our results for the relaxation time, τ, in lattice sweeps, vs. volume fraction, ϕ, on the isotherm $T^* = 0.6$ are shown in Fig. 5. Despite the fact that the time-scales are different, the similarity of Fig. 5 with Fig. 1 is evident. Once again, the sequential updating algorithm is seen to produce samples of the highest statistical quality. Our analysis of small and large systems confirms the enhancement obtained by using sequential updating algorithms. Thus, we conclude that the so-called "neighbor" effect seen in Ref. [6] for the Ising model also persists in the lattice gas. We anticipate that an analogous behavior similar to that of the lattice gas shown in Fig. 5 should also be seen for continuum fluids.

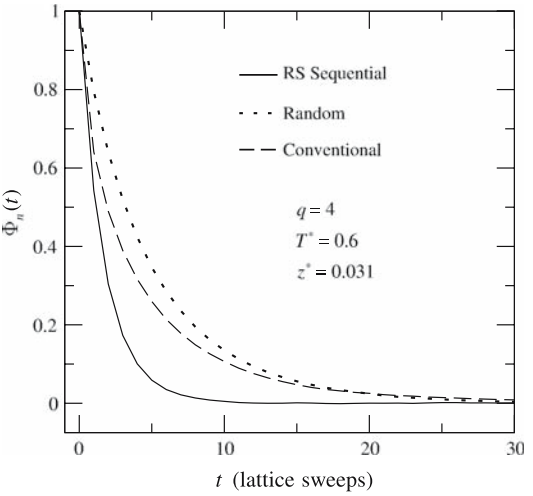

Fig. 3. Density autocorrelation function, $\Phi_n(t)$, for a 100×100 lattice gas model with $q = 4$ neighbors at $T^* = k_B T/\epsilon = 0.6$ and $z^* = z\upsilon = 0.031$. Time t is measured in lattice sweeps. For the case of the sequential algorithm, a random skip is implemented every other sweep.

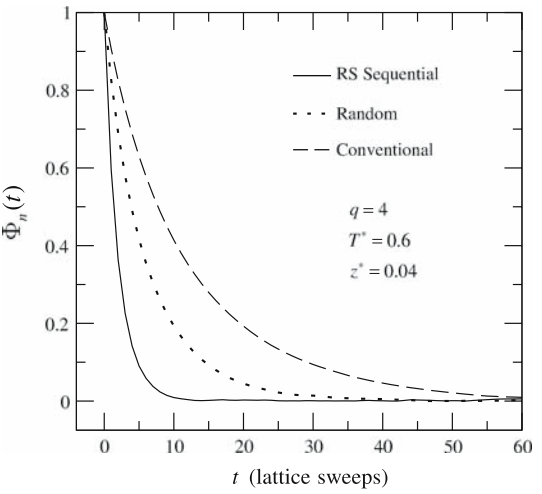

Fig. 4. Density autocorrelation function, $\Phi_n(t)$, for a 100×100 lattice gas model with $q = 4$ neighbors at $T^* = k_B T/\epsilon = 0.6$ and $z^* = z\upsilon = 0.04$. Time t is measured in lattice sweeps. For the case of the sequential algorithm, a random skip is implemented every other sweep.

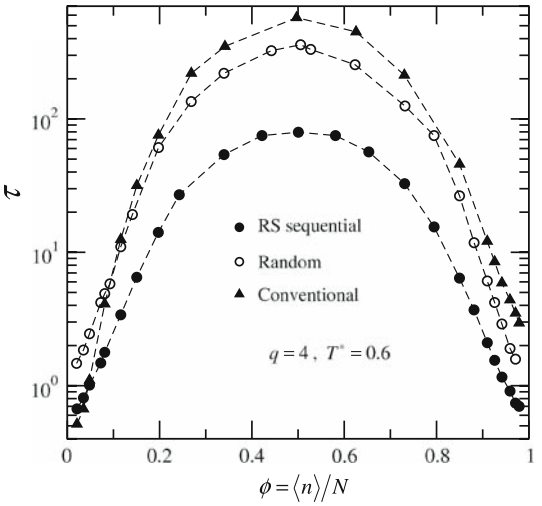

Fig. 5. Integrated autocorrelation time τ (lattice sweeps) as a function of volume fraction $\phi=\langle n\rangle/N$ for the two-dimensional 100×100 lattice gas with $q=4$ neighbors at $T^*=0.6$. (●): RS sequetial updating ; (○): random updating; (▲): conventional algorithm.

7. DISCUSSION AND CONCLUSIONS

Monte Carlo simulations suffer from convergence problems in many situations. The most notable example is the problem of critical slowing down [1] that occurs in the vicinity of the critical point, $T\approx T_c$. Monte Carlo algorithms will be more efficient if they have better mobility over the sample space. Although the Metropolis acceptance probability Eq. (15) is thought to be optimal, considerable freedom still exists in devising better transition kernels with the same acceptance criteria. In this direction, this work and the previous work of Ref. [6] shows that relaxation of the unnecessarily strong constraint of strict detailed balance results in enhanced statistical efficiency.

The improvement of sequential updating algorithms is due to an avalanche or cascade type of phenomenon caused by the so-called "neighbor effect." For the lattice gas, successful insertion or removal triggers a higher probability of insertion or removal in a neighboring cell as the previous cell. While the "neighbor effect" persists in sequential updating, it rarely occurs in random updating algorithms due to the random selection of the cell. The "neighbor effect" exists not only for lattice systems, but also for continuum fluid models such as Lennard-Jones. The next step in

the direction of sequential updating algorithms involves grand canonical simulations for continuum fluids.

The main advantages of algorithms based on sequential updates are their implicity, efficiency, and feasibility of parallel implementation. Metropolis himself [19] anticipated that Monte Carlo simulations can be implemented with massively parallel machines. However, strict detailed balance is a big obstacle in achieving this goal since it requires that every elementary update be reversed before all other moves are attempted. In parallel simulations based on domain decomposition, once a particle escapes out of a subdomain, processors need to exchange information immediately. However, frequent information exchange in parallel simulations reduces the efficiency dramatically. The efficiency may be even worse than serial implementation due to slow but frequent synchronization between processors. Due to the nature of the updates and the absence of strict detailed balance, sequential algorithms can be readily put in parallel format. The efficiency of parallel implementation of sequential algorithms is in addition to the improvement of serial implementation without loss of precision.

ACKNOWLEDGMENTS

The authors appreciate the encouragement and interest of Professor V.I. Manousiouthakis and Professor P.D. Christofides. The calculations were performed on Intel Xeon processors awarded to G.O. through Intel Higher Education Program Equipment Grants. Financial support from NSF, CBET-0652131, is gratefully acknowledged.

REFERENCES

1. D. P. Landau and K. Binder, *A Guide to Monte Carlo Simulations in Statistical Physics* (Cambridge, New York, 2000).
2. D. Frenkel and B. Smit, *Understanding Molecular Simulation*, 2nd Ed. (Academic, New York, 2002).
3. M. P. Allen and D. J. Tildesley, *Computer Simulation of Liquids* (Oxford, New York, 1987).
4. N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, *J. Chem. Phys.* **21**:1087 (1953).
5. V. I. Manousiouthakis and M. W. Deem, *J. Chem. Phys.* **110**:2753 (1999).
6. R. Ren and G. Orkoulas, *J. Chem. Phys.* **124**:064109 (2006).
7. G. E. Norman and V. S. Filinov, *High Temp.* (USSR) **7**:216 (1969).
8. D. J. Adams, *Mol. Phys.* **28**:1241 (1974).
9. D. J. Adams, *Mol. Phys.* **29**:307 (1975).
10. L. A. Rowley, D. Nicholson, and N. G. Parsonage, *J. Comp. Phys.* **17**:401 (1975).

11. R. B. Bapat and T. E. S. Raghavan, *Nonnegative Matrices and Applications* (Cambridge, New York, 1997).
12. J. S. Liu, *Monte Carlo Strategies in Scientific Computing* (Springer, New York, 2001).
13. E. Parzen, *Stochastic Processes* (Holden-Day, San Francisco, 1962).
14. P. H. Peskun, *Biometrica* **60**:607 (1973).
15. T. D. Lee and C. N. Yang, *Phys. Rev.* **87**:410 (1952).
16. K. Huang, *Statistical Mechanics* (Wiley, New York, 1963).
17. L. Onsager, *Phys. Rev.* **65**:117 (1944).
18. M. Nightingale and H. W. J. Blöte, *Phys. Rev. Lett.* **76**:4548 (1996).
19. N. Metropolis, *Los Alamos Science* (1987), No 16, pp. 125–130.