Single-cluster dynamics for the random-cluster model

Youjin Deng,[1] Xiaofeng Qian,[2] and Henk W. J. Blöte[2,3]

[1]*Hefei National Laboratory for Physical Sciences at Microscale and Department of Modern Physics,*
University of Science and Technology of China, Hefei 230027, China
[2]*Lorentz Institute, Leiden University, P.O. Box 9506, 2300 RA Leiden, The Netherlands*
[3]*Faculty of Applied Sciences, Delft University of Technology, P.O. Box 5046, 2600 GA Delft, The Netherlands*

We formulate a single-cluster Monte Carlo algorithm for the simulation of the random-cluster model. This algorithm is a generalization of the Wolff single-cluster method for the q-state Potts model to noninteger values $q > 1$. Its results for static quantities are in a satisfactory agreement with those of the existing Swendsen-Wang-Chayes-Machta (SWCM) algorithm, which involves a full-cluster decomposition of random-cluster configurations. We explore the critical dynamics of this algorithm for several two-dimensional Potts and random-cluster models. For integer q, the single-cluster algorithm can be reduced to the Wolff algorithm, for which case we find that the autocorrelation functions decay almost purely exponentially, with dynamic exponents $z_{exp} = 0.07$ (1), 0.521 (7), and 1.007 (9) for $q = 2$, 3, and 4, respectively. For noninteger q, the dynamical behavior of the single-cluster algorithm appears to be very dissimilar to that of the SWCM algorithm. For large critical systems, the autocorrelation function displays a range of power-law behavior as a function of time. The dynamic exponents are relatively large. We provide an explanation for this peculiar dynamic behavior.

I. INTRODUCTION

The Kasteleyn-Fortuin mapping [1] of the q-state Potts model [2] onto the random-cluster model provides a way to define the Swendsen-Wang [3] and related cluster Monte Carlo algorithms [4,5] for the Potts model. These algorithms can apply nonlocal changes to the configuration. For systems with long-range correlations, these nonlocal methods appear to be very efficient in comparison with the standard Metropolis Monte Carlo method [6], which applies only local updates.

The number q of Potts states appears as a continuous variable in the random-cluster model; the latter model can thus be seen as a generalization of the Potts model to noninteger values of q. There exist several ways to simulate noninteger-q random-cluster models. First, Sweeny applied local updates of the bond variables [7]. While the Sweeny algorithm, like cluster algorithms, suppresses most of the critical slowing down, a bond update requires a nonlocal task which increases the computer time requirements. Another algorithm was formulated by Hu [8]. It generates percolation configurations and applies a statistical reweighting in order to obtain the correct averages for the random-cluster model. A cluster algorithm of the Swendsen-Wang type was formulated by Chayes and Machta [9] for the $q \geq 1$ random-cluster model. While all these algorithms, if using a random generator of a sufficient quality, lead to results that are only subject to statistical errors, it was found that the Swendsen-Wang-Chayes-Machta (SWCM) cluster algorithm (where applicable, i.e., $q > 1$) is much more efficient than the reweighting method [10]. It was also found [10] to be more efficient than the Sweeny method, although the latter result depends strongly on the sophistication of the bond update method. Detailed studies of the dynamic critical behavior of the SWCM and the Sweeny algorithms can also be found in Ref. [11].

In this work, we present a single-cluster algorithm for the random-cluster model with real $q > 1$. This algorithm thus has elements in common with the Wolff [4] as well as with the SWCM cluster algorithm [9]. Since, for integer q, the Wolff method is about as efficient as the Swendsen-Wang algorithm, one might expect that the same holds for our single-cluster algorithm, in comparison with the SWCM algorithm. A test of this expectation is also included in the present work.

Section II provides an explanation of the theoretical aspects of the algorithm. We include a simple example of such an algorithm, and prove that the condition of detailed balance is satisfied. Furthermore, we describe other variants of the algorithm that are applicable to models with $q > 2$, and describe how the algorithm can reduce to the Wolff algorithm for integer q. In Sec. III we test the validity of the algorithm, and determine its dynamic exponent for two-dimensional random-cluster models on the square lattice, using several values of q. The generic dynamical behavior appears to be very different from that of the Wolff algorithm. We conclude with a discussion of our findings in Sec. IV, which also includes an explanation of the mechanism responsible for the unusual dynamical behavior.

II. ALGORITHM

A. Kasteleyn-Fortuin mapping

We recall the mapping of the Potts model and the random-cluster model on a model with bond as well as site variables, as described in Refs. [9,10]. For a review of the Potts model, see, e.g., Ref. [12]. The Potts partition sum is

$$Z_\sigma \equiv \left[\prod_{i=1}^{N} \sum_{\sigma_i=1}^{q} \right] \prod_{\langle ij \rangle} \exp(K\delta_{\sigma_i \sigma_j}), \qquad (1)$$

where the summations are on all site variables σ_i, where i labels the lattice sites. The second product sign indicated by

$\langle ij \rangle$ is on all nearest-neighbor pairs (i,j). The coupling K is reduced, i.e., it includes a factor $1/k_B T$. We consider the case of ferromagnetic couplings $K \geq 0$. The Kasteleyn-Fortuin mapping of Eq. (1) on the random-cluster model [1] introduces bond variables $b_{ij} = 0$ or 1 between all neighbor pairs (i,j), after which the site variables $\sigma_i = 1, 2, \ldots, q$ can be summed out so that only the bond variables remain as the degrees of freedom of the random-cluster model. Bonds $b_{ij} = 1$ (0) are considered to be present (absent).

The random-cluster partition sum thus assumes the form

$$Z_\sigma = Z_b \equiv \left[\prod_{\langle ij \rangle} \sum_{b_{ij}=0}^{1} \right] q^{n_c} u^{n_b} = \sum_{\{b\}} \prod_{k=1}^{n_c} q u^{n_b^{(k)}}, \quad (2)$$

where $u \equiv e^K - 1$ is the temperaturelike parameter and $n_b \equiv \Sigma b_{ij}$ denotes the number of present bonds. The number of clusters is denoted as n_c. The sum on $\{b\}$ is shorthand for the sum on all configurations of bond variables, and $n_b^{(k)}$ is the number of nonzero bonds in the kth cluster.

For $q > 1$ one can divide the cluster weight q in two positive contributions 1 and $q-1$. The first contribution can be associated with one of the original Potts states. To this purpose we introduce "color" variables $\tilde{t}_k = 0$ or 1 for each cluster $k = 1, 2, \cdots, n_c$,

$$Z_b = \sum_{\{b\}} \prod_{k=1}^{n_c} \sum_{\tilde{t}_k=0}^{1} u^{n_b^{(k)} \tilde{t}_k} [(q-1) u^{n_b^{(k)}}]^{1-\tilde{t}_k}. \quad (3)$$

Clusters of colors 0 and 1 have weight $q-1$ and 1, respectively. The sum on the cluster colors is replaced by a sum over N site-color variables $t_i = 0$ or 1, together with a factor $\delta_{t_i t_j}^{b_{ij}}$ (with the convention $0^0 = 1$) for each bond variable, so that each cluster contains only sites of one color,

$$Z_b = Z_{tb} \equiv \sum_{\{t\}} \sum_{\{b\}} \prod_{\langle ij \rangle} (u \delta_{t_i t_j})^{b_{ij}} \prod_{k=1}^{n_c} (q-1)^{1-t_{s(k)}}, \quad (4)$$

where the color of the kth cluster is denoted $t_{s(k)}$, where $s(k)$ is a site in that cluster. In a site configuration $\{t\}$ we identify three types of bonds (ij),

$$\text{type } 0 : t_i = t_j = 0,$$

$$\text{type } 1 : t_i = t_j = 1,$$

$$\text{type } 2 : t_i + t_j = 1.$$

Summations and products involving only one of these types of bond are specified by appending corresponding superscripts to the pertinent summation and product signs:

$$Z_{tb} = \sum_{\{t\}} \left[\sum_{\{b\}}{}^{(0)} \prod_{\langle ij \rangle}{}^{(0)} u^{b_{ij}} \right] \left[\prod_{k=1}^{n_c^{(0)}} (q-1) \right] \left[\sum_{\{b\}}{}^{(1)} \prod_{\langle ij \rangle}{}^{(1)} u^{b_{ij}} \right]$$
$$\times \left[\sum_{\{b\}}{}^{(2)} \prod_{\langle ij \rangle}{}^{(2)} (1 - b_{ij}) \right], \quad (5)$$

where the clusters of color 0 are labeled $1, 2, \ldots, n_c^{(0)}$. The type 1 and 2 sums can now be executed. After rewriting the type-0 sum one obtains the partition sum expressed in site variables and type 0 bond variables,

$$Z_{tb} = Z_{tb1} \equiv \sum_{\{t\}} \sum_{\{b\}}{}^{(0)} \left[\prod_{k=1}^{n_c^{(0)}} (q-1) u^{n_b^{(k)}} \right] \left[\prod_{\langle ij \rangle}{}^{(1)} (1 + u) \right]. \quad (6)$$

Equation (6) specifies the probability distribution of a system of site variables $t_i = 0$, 1 and bond variables b_{ij} between nearest-neighbor sites of type 0. Each term in the second sum in Eq. (6) specifies a cluster decomposition $\mathcal{D}(\{b\})$ of the sublattice formed by sites k with $t_k = 0$. Different sets of bond variables $\{b\}$ may still correspond with the same cluster decomposition. Thus, if we replace the sum on $\{b\}$ by a sum on all cluster decompositions of the color-0 regions, we have to insert a summation on all $\{b\}$ that are consistent with $\mathcal{D}$,

$$Z_{tb} = Z_{t\mathcal{D}} \equiv \sum_{\{t\}} \left[\prod_{\langle ij \rangle}{}^{(1)} (1 + u) \right] \sum_{\{\mathcal{D}\}}{}^{(0)} \sum_{\{b\} | \mathcal{D}} \left[\prod_{k=1}^{n_c^{(0)}} (q-1) u^{n_b^{(k)}} \right]. \quad (7)$$

B. Simplest form of the algorithm

Equation (7) can serve as the basis on which a single-cluster Monte Carlo algorithm can be constructed. This algorithm is applied as follows to a mixed configuration specified by the site variables t_i and a cluster decomposition $\mathcal{D}$ of the color-0 sites. An initial configuration can, for instance, be obtained from a random-cluster configuration and assigning color 1 to each cluster with probability $1/q$. Then, a single-cluster step is executed as follows:

(1) Choose a random site i. The action taken by the algorithm depends on the color variable t_i.

(2) (a) If $t_i = 1$, do with probability $p_1 = (q-1)/q$ the following: form a random cluster around site i with bond probability $p = u/(u+1)$ between sites of color 1. The sites j in the newly formed cluster are assigned color 0 (i.e., $t_j = 0$) and the number $n_c^{(0)}$ of clusters of color 0 is thus increased by 1. (b) If $t_i = 0$, do with probability $p_2 = 1/q$ the following: assign color 1 to all sites of the cluster containing site i, and thus decrease the number of clusters of color 0 by 1.

C. Proof of detailed balance

The proof of detailed balance can be formulated as follows. Consider two mixed configurations S_1 and S_2, which differ only in a region $\mathcal{C}$ whose sites j have color $t_j = 1$ in S_1, and whose sites belong to a single cluster in S_2, and thus have color $t_j = 0$. According to the rules given in the preceding subsection, the transition probability to move from S_1 to S_2 is

$$T(2,1) = \frac{(q-1)N_c}{qN} \sum_{\{b\} | \mathcal{C}} \left(\frac{u}{u+1} \right)^{n_b} \left(\frac{1}{u+1} \right)^{n_p + n_{nn} - n_b}, \quad (8)$$

where N_c is the number of sites in region $\mathcal{C}$; N is the total number of sites in the system; $\{b\}$ stands for the n_{nn} bond variables on the edges between nearest-neighbor sites in $\mathcal{C}$; the combination on $\{b\} | \mathcal{C}$ indicates the sum on all configurations $\{b\}$ that connect all sites in $\mathcal{C}$ into a single cluster; n_b

denotes the number of nonzero bond variables in $\{b\}$; and n_p is the number of bond variables connecting sites inside $\mathcal{C}$ with those outside $\mathcal{C}$ of color 1 (i.e., the number of bonds along the boundary of $\mathcal{C}$ that is broken when the color of $\mathcal{C}$ is changed). The prefactor $(q-1)N_c/qN$ describes the probability that the cluster formation starts within $\mathcal{C}$. Each of the n_b "present" bonds contributes a factor $u/(u+1)$, and each of the $n_{nn} - n_b$ "absent" bonds a factor $1/(u+1)$. Also each "broken" bond along the perimeter of $\mathcal{C}$ contributes a factor $1/(u+1)$.

The rules given in the preceding subsection also specify the probability of the inverse move, namely, from S_2 to S_1, as

$$T(1,2) = \frac{N_c}{qN}. \tag{9}$$

The condition of detailed balance requires that the transition probabilities $T(2,1)$ and $T(1,2)$ are related to the equilibrium probabilities $P(1)$ and $P(2)$ of configurations 1 and 2, respectively,

$$T(2,1)/T(1,2) = P(2)/P(1). \tag{10}$$

Since the probabilities $P(1)$ and $P(2)$ are proportional to the configuration weights specified by Eq. (7), we may write

$$P(2)/P(1) = W(2)/W(1), \tag{11}$$

where the weights associated with region $\mathcal{C}$ in Eq. (7) are

$$W(1) = (1+u)^{n_p+n_{nn}} \tag{12}$$

and

$$W(2) = (q-1)\sum_{\{b\}|\mathcal{C}} u^{n_b}. \tag{13}$$

From Eqs. (8) and (9), and from Eqs. (12) and (13), we conclude that

$$T(2,1)/T(1,2) = (q-1)(1+u)^{-n_p-n_{nn}}\sum_{\{b\}|\mathcal{C}} u^{n_b} = W(2)/W(1), \tag{14}$$

which shows that the condition of detailed balance, Eq. (10), indeed is satisfied.

D. Other versions

The probabilities p_1 and p_2 in Sec. II B can be chosen differently, depending on the value of q. For $1 < q < 2$ we may take $p_1 = q-1$ and $p_2 = 1$. For $q > 2$, this is not possible but other possibilities arise. One can generalize the algorithm by allowing more than two values of the color variables t_i. The most obvious way is to allow $n \equiv [q]$ (the integer part of q) values with weight one, and one special value with weight $q-n$. Sites of the latter color are divided in clusters (just as before); sites of the n remaining colors are not. A cluster step starting from a randomly chosen site can then be specified as follows: if that site belongs to a cluster (thus, of the special color 0), then the cluster is erased and its sites are given a random color $1, 2, \ldots, n$ with probability $1/n$ each. If the cluster step starts from a randomly chosen site of color

$1, 2, \ldots, n$, then a single cluster is formed. Its sites receive one of the $n-1$ other weight-1 colors with probability $(2n-q)/[n(n-1)]$ each, and the cluster receives the special color with probability $(q-n)/n$. This choice of probabilities satisfies detailed balance and maximizes the probability of a cluster flip. For integer q, with the choice $n=q$, it leads to the Wolff algorithm for the Potts model with transition probabilities $1/(q-1)$ of each color to the $q-1$ other colors [5].

E. Test of the algorithm

We tested the single-cluster algorithm for the cases of the $q=2$, 3, and 4 Potts model on the square lattice, by comparing its numerical results to those of the Wolff algorithm. We set $n=q-1$ (see Sec. II D) and the weight of the color-0 clusters is thus $q-n=1$. Simulations were performed on $L \times L$ lattices with periodic boundary conditions. After each single-cluster step, we sampled various quantities, including the densities ρ_i of Potts variables in states $i=1,2,\ldots,q$, and the single-cluster size S. The single-cluster size is counted as the total number of lattice sites in the cluster as constructed by the algorithm. If the number q of Potts states is an integer, the squared Potts magnetization density m^2 can be expressed in the densities ρ_i as

$$m^2 = \frac{1}{q-1}\sum_i\sum_{j<i}(\rho_i-\rho_j)^2 = \frac{q}{q-1}\sum_i(\rho_i-1/q)^2. \tag{15}$$

The sum on the right-hand side of this equation contains q terms whose expectation value is equal, due to the Potts symmetry. Thus, for the expectation value $\langle m^2\rangle$ of m^2 we may write

$$\langle m^2\rangle = \frac{q^2}{n(q-1)}\sum_{i=1}^n\langle(\rho_i-1/q)^2\rangle, \tag{16}$$

with $1 \le n \le q$. Thus it is sufficient to sample $(\rho_i-1/q)^2$ in order to obtain $\langle m^2\rangle$. While q is taken to be an integer in this subsection, Eq. (16) still applies for general $q > 1$. If q is not an integer, n will usually be chosen as $n=[q]$, where $[q]$ denotes the integer part of q. Although, in the case $n < q$, Eq. (16) still leads to the same expectation values as those obtained by averaging on the basis of a full-cluster decomposition, the autocorrelations of m^2 may depend on the sampling method and thus be different in both cases.

As should be expected, for Potts models with integer values of q, the Wolff and the present algorithm did indeed yield mutually consistent results. This is demonstrated by the data for $\langle m^2\rangle$ and $\langle S\rangle$ in Table I obtained by the two algorithms for the case $q=2$, $n=1$. Furthermore, since the probability to hit a cluster is equal to its relative size, it follows that the two expectation values $\langle m^2\rangle$ and $\langle S\rangle$ are equal. Our simulation results were also in a good agreement with this relation, as illustrated by the data shown in Table I for the critical Ising model.

Since both simulations involve the same number of samples, the statistical uncertainties, shown between brackets in Table I, reflect the relative efficiency of the Wolff and the single-cluster algorithm. For size $L=8$, the Wolff method is about ten times as efficient as the present algorithm, while

TABLE I. Simulation results for the average squared magnetization $\langle m^2 \rangle$ and the single-cluster size S for the critical $q=2$ random-cluster model, as obtained by the Wolff (W) and the present single-cluster algorithm (S) with $n=q-1$ as defined in the text. The parameter L specifies the linear system size. The number of samples per system size is 4×10^6 for each simulation, and the number of clusters formed between subsequent samples is two for $L \leq 24$ and 3 for $L=32$. The numbers between brackets show the statistical error margins in the last two decimal places.

L	8	12	16	20	24	32
m^2 (W)	0.64693(18)	0.58581(18)	0.54537(17)	0.51584(16)	0.49305(17)	0.45874(14)
m^2 (S)	0.6478(6)	0.5861(8)	0.5442(9)	0.5164(10)	0.4932(13)	0.4610(12)
S (W)	0.64666(18)	0.58581(18)	0.54544(17)	0.51594(16)	0.49311(17)	0.45878(14)
S (S)	0.6470(6)	0.5860(8)	0.5441(9)	0.5165(10)	0.4932(13)	0.4610(12)

this difference increases to a factor of about 100 for $L=32$. It thus appears that the two algorithms have different dynamic exponents.

III. DYNAMIC EXPONENTS

A. Autocorrelation functions and autocorrelation times

Consider an observable $\mathcal{O}$, whose evolution in time t' is described by the time series $\mathcal{O}(t')$, where each unit of t' corresponds to one step of the single-cluster algorithm. The autocovariance function of $\mathcal{O}$ is defined to be

$$C_{\mathcal{O}}(t') \equiv \langle \mathcal{O}(0)\mathcal{O}(t') \rangle - \langle \mathcal{O} \rangle^2, \qquad (17)$$

and its autocorrelation function is

$$A_{\mathcal{O}}(t') \equiv \frac{C_{\mathcal{O}}(t')}{C_{\mathcal{O}}(0)}. \qquad (18)$$

We then normalize time t' as $t=t'S/L^2$ so that the time unit of t is the average number of cluster steps in which each lattice site is visited once. From $A_{\mathcal{O}}(t)$ we then define the integrated autocorrelation time as

$$\tau_{\text{int},\mathcal{O}} \equiv \frac{1}{2} + \sum_{t=1}^{\infty} A_{\mathcal{O}}(t), \qquad (19)$$

and the exponential autocorrelation time as

$$\tau_{\text{exp},\mathcal{O}} \equiv \lim_{t \to \infty} \frac{-t}{\log A_{\mathcal{O}}(t)}. \qquad (20)$$

Finally, the exponential autocorrelation time of the system is defined as

$$\tau_{\text{exp}} = \sup_{\mathcal{O}} \tau_{\text{exp},\mathcal{O}}, \qquad (21)$$

where the supremum is taken over all observables $\mathcal{O}$. This autocorrelation time measures the decay rate of the slowest mode of the system. All observables that are not orthogonal to this slowest mode satisfy $\tau_{\text{exp},\mathcal{O}} = \tau_{\text{exp}}$.

B. Integer q

For integer $q=2,3,4$, we may set $n=q$, in which case the color-0 clusters have zero weight and are thus absent, so that

the single-cluster algorithm reduces to the version of the Wolff algorithm [4] adapted to the Potts model. Such Wolff simulations were performed at criticality. The system sizes were chosen as powers of 2 in the range $4 \leq L \leq 4096$ for $q=2$, $4 \leq L \leq 2048$ for $q=3$, and $4 \leq L \leq 1024$ for $q=4$. Samples were taken at intervals of one single-cluster step. The number of samples taken for each system size is shown in Table II.

After a fast initial decay, the autocorrelation functions for S and m^2 decay approximately exponentially, but with an amplitude proportional to a power of the linear size L. Except for the initial decay, the behavior can be described as

$$A_{\mathcal{O}}(t) \propto L^{-s_{\mathcal{O}}} e^{-t/\tau_{\text{exp}}(L)}, \qquad (22)$$

with $\mathcal{O}=S$ or m^2, which implies that $z_{\text{int},\mathcal{O}}=z_{\text{exp}}-s_{\mathcal{O}}$. Accordingly, a data collapse is obtained by plotting the quantity $L^{s_s}A_S$ versus t/τ_{exp}. This is shown in Fig. 1, with the exponent of L fixed as $s_s=0.37$.

Correlations between subsequent Wolff steps are thought to arise from overlap between the two pertinent clusters. The average Wolff cluster size, relative with respect to the size L^d of the system at criticality, scales with L as $S \propto L^{2y_h-2d}$, where y_h is the magnetic exponent and $d=2$ is the dimensionality of the lattice. The probability that two subsequent clusters overlap may thus be crudely estimated as L^{4d-4y_h}. The histogram of the cluster-size distribution is however very wide with a large-size cutoff that scales as L^{y_h}. Since large clusters contribute more to the autocorrelation function than small ones, one may expect that the correlations at short times scale with L instead as L^{-s_s} with $s_s < 4d-4y_h$. The results for $q=2$, 3, and 4 are shown in Table III. It seems that for $q=4$ the Wolff algorithm is slightly less efficient than the Swendsen-Wang method.

TABLE II. Lengths of the Wolff-type simulations in Sec. III B for $L \geq 16$ and $q=2$, 3, and 4, in units of 10^7 samples.

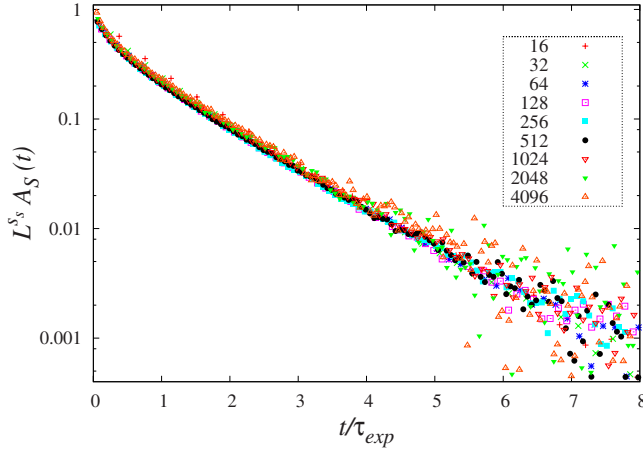| L	16	32	64	128	256	512	1024	2048	4096
$q=2$	12	12	12	12	16	16	32	8	8
$q=3$	4	4	4	4	8	8	12	12	
$q=4$	8	12	20	32	48	72	64		

FIG. 1. (Color online) Data collapse of the autocorrelation function of the single-cluster size, shown as $L^{s_s}A_S$ on a logarithmic scale, versus $t/\tau_{\exp}$, with $s_s=0.37$. These results apply to $q=2$ Wolff dynamics.

During the simulations, also the energy density E, which is defined as the nearest-neighbor correlation function, was sampled. The corresponding autocorrelation function $A_E(t)$ is shown in Fig. 2 for $q=2$. This figure indicates that A_E decays approximately exponentially as a function of t, with an amplitude that has little or no dependence on the system size. It thus follows that $z_{\exp,E} \approx z_{\text{int},E}$. The autocorrelation times τ_{int} and $\tau_{\exp}$ were obtained by integration and least-squares fits respectively. The autocorrelation times for $L \geq 16$ were fitted by

$$\tau_{\text{int},E}(L) = a + bL^{z_{\text{int},E}}, \tag{23}$$

and similarly for $\tau_{\exp}$. The fit yields $z_{\exp} \approx z_{\text{int},E}=0.07$ (1). This nonzero dynamic exponent is in agreement with the upward curvature of the data for $\tau_{\exp}$ versus L on a logarithmic scale, shown in Fig. 3. However, we cannot exclude the possibility that the dynamic exponent is zero because the data for $L \geq 16$ can also be described by $\tau_{\text{int},E}(L)=\tau_0 +\ln L(a_0+a_1/L+a_2/L^2)$, which has only one more parameter

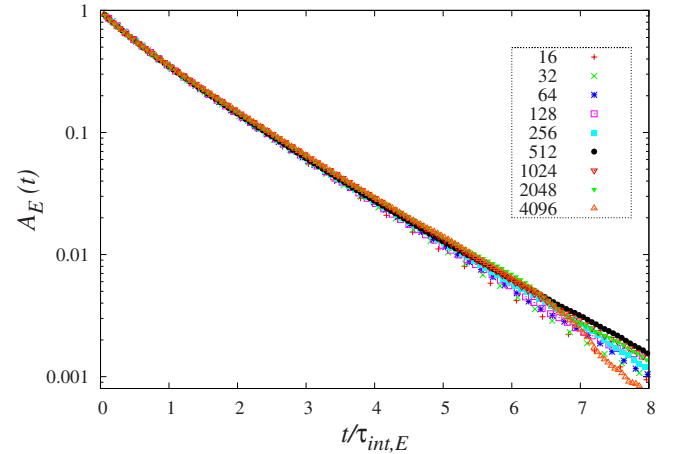FIG. 2. (Color online) Data collapse of the autocorrelation function A_E vs $t/\tau_{\text{int},E}$ for $q=2$ Wolff dynamics. The system sizes L are shown in the figure. The statistical uncertainties become appreciable at large times.

than Eq. (23), with $\tau_0=-1.02$ (7), $a_0=0.76$ (2), $a_1=3.4$ (6), and $a_2=-10$ (5); this is illustrated in Fig. 3. Such behavior would mean that the Li-Sokal bound [13] is sharp for the Wolff dynamics of the two-dimensional Ising model.

C. Noninteger q

We performed simulations of critical random-cluster systems with sizes $4 \leq L \leq 256$ for $q=1.25$, 1.50, 1.75, 2.25, 2.50, and 2.75, with $n=[q]$. Samples were taken after each single-cluster step, with a total number of samples of 6×10^7 for each L,q. The squared magnetization was obtained using Eq. (16).

The autocorrelation functions $A_{m^2}(t)$ and $A_S(t)$ were found to display a fast decay at short times $t \sim O(1)$, then decay algebraically, and ultimately exponentially with t. Such a range of algebraic behavior, which extends to $t \gg 1$ for at large L, is absent for Wolff dynamics. In the case of $A_{m^2}(t)$, the fast initial decay at small t appears to be hardly size

TABLE III. Single-cluster dynamics for several values of q. The exponents r_s, and r_m are those in Eq. (24) for S, and m^2, respectively; the same labeling applies to s_s and s_m. The values of s_m are not significantly different from zero for noninteger values of q. For the purpose of comparison, the last column shows results [11] for $z_{\exp}$ applying to SWCM cluster dynamics. Furthermore, some data are included for integer values $q=2$, 3, and 4, with the choice $n=q$, so that these results apply to the Wolff algorithm.

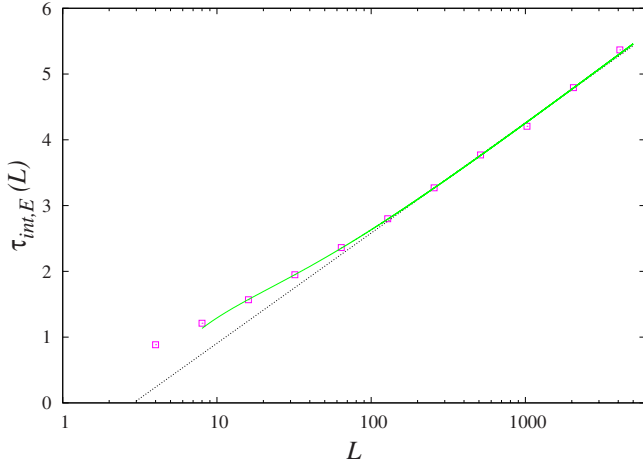| q	s_s	r_s	s_m	r_m	$z_{\exp}$	$z_{\exp}$ (SW)
1.25	0.25(2)	0.25(2)	0.00(2)	0.25(1)	2.0(2)	0.00
1.50	0.19(2)	0.19(2)	0.00(2)	0.19(1)	2.0(2)	0.00
1.75	0.14(2)	0.14(2)	0.00(2)	0.14(1)	2.0(2)	0.06(1)
2.25	0.26(2)	0.15(2)	0.00(2)	0.14(1)	2.0(2)	0.24(1)
2.50	0.22(2)	0.10(2)	0.00(2)	0.12(1)	2.0(2)	0.31(1)
2.75	0.17(2)	0.06(2)	0.00(2)	0.10(1)	2.0(2)	0.42(2)
2.00	0.37(2)		0.14(2)		0.07(1)	0.14(1)
3.00	0.34(2)		0.05(2)		0.521(7)	0.49(1)
4.00	0.25(2)		0.00(2)		1.007(9)	0.93(2)

FIG. 3. (Color online) Semilogarithmic plot of the integrated autocorrelation function $\tau_{int,E}(L)$ versus L for $q=2$ Wolff dynamics. The error bars are of the same size as the data points. The solid (green) curved line is obtained from the logarithmic fit. The difference with the power-law fit would not be visible on this scale. The straight dashed line represents pure logarithmic behavior $\tau \propto \ln L$, and serves only for the purpose of illustration.

dependent, as can be seen in Fig. 4. In contrast, for $A_S(t)$, the amplitude of the algebraic decay is found to be size dependent.

These dynamic phenomena are very different from those for integer q, where the autocorrelation functions for both quantities decay almost as a pure exponential law. It seems that the behavior of $A_{\mathcal{O}}(t)$ can be described by

$$A_{\mathcal{O}}(t,L) = L^{-s_{\mathcal{O}}} t^{-r_{\mathcal{O}}} f(t/\tau_{\exp}(L)) \quad \text{for } t \gg 1, \qquad (24)$$

where f is a universal function. For large t, it behaves as

$$f(t/\tau_{\exp}(L)) \propto e^{-t/\tau_{\exp}(L)} \quad \text{with } \tau_{\exp}(L) \propto L^{z_{\exp}}. \qquad (25)$$

We analyzed $A_{\mathcal{O}}(t,L)$ by attempting to collapse the data onto a single curve according to Eq. (24). The data collapses for
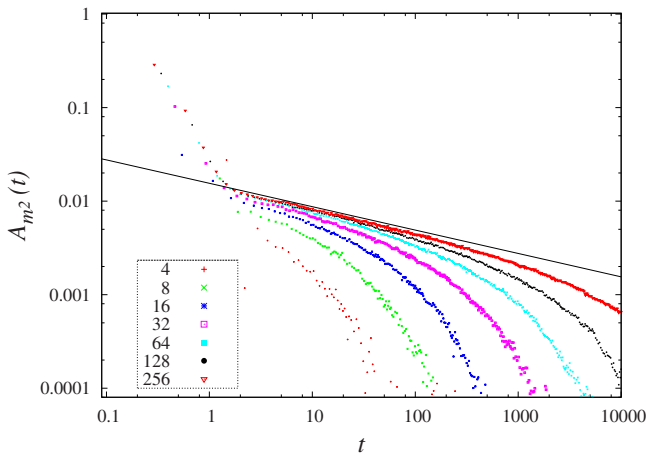

FIG. 4. (Color online) Autocorrelation function A_{m^2} for $q=1.25$ versus time t, using logarithmic scales. These data apply to the single-cluster simulation of the $q=1.25$ random-cluster model. The straight line is only for the purpose of illustration, and has slope −0.25.
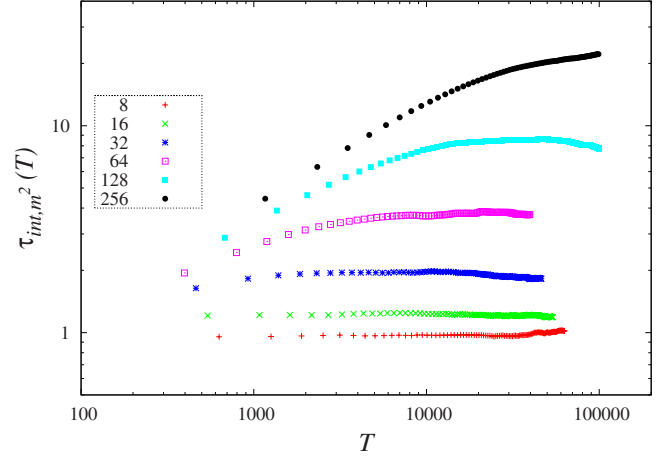

FIG. 5. (Color online) Indefinite integral $\tau_{int,m^2}(T)$ of the magnetic autocorrelation function $A_{m^2}(t)$ over the time interval $0 < t < T$. These results apply to the single-cluster simulation of the $q=1.25$ random-cluster model.

$\mathcal{O}=m^2$ and S work only approximately. This might be due to finite-size corrections. The results are shown in Table III.

The power-law dependence of the amplitude of the exponential decay of autocorrelations means that, in terms of a measure of the efficiency of the algorithm, the significance of the dynamic exponent $z_{\exp}$ is limited. The exponent z_{int} is a better measure of the L dependence of the rate of decay of correlations because it includes the size dependence of the amplitude of the decay. The unusual behavior in Fig. 4 may be expected to lead to significant differences between $z_{\exp}$ and z_{int}. This expectation is verified by integration of Eq. (24), which yields that

$$\tau_{int,\mathcal{O}} \propto L^{z_{int}}, \quad z_{int} = (1-r_{\mathcal{O}}) z_{\exp} - s_{\mathcal{O}}. \qquad (26)$$

Inspection of the numerical results for $z_{\exp}$, $r_{\mathcal{O}}$, and $s_{\mathcal{O}}$ in Table III shows that $z_{\exp}$ and z_{int} must have different values. For a numerical analysis of z_{int}, we have, in line with Eq. (19), integrated the autocorrelation functions for m^2 and S according to

$$\tau_{int,\mathcal{O}}(T) \equiv \frac{1}{2} + \sum_{t=1}^{T} A_{\mathcal{O}}(t), \qquad (27)$$

where, presently, T assumes the meaning of a time variable. The integrated autocorrelation times of the $q=1.25$ model for $\tau_{int,m^2}(T)$ and $\tau_{int,S}(T)$ are shown in Figs. 5 and 6, respectively. The lines for large L are approximately straight, which reflects the algebraic decay of the autocorrelation functions as a function of t. As a consequence of the exponential decay at large t, $\tau_{int}(T)$ approaches a constant. However, integration of random correlations at large t eventually affects the accuracy of the numerical result for $\tau_{int}(T)$ so that a cutoff has to be applied for optimal results. For this reason, the integrated autocorrelation times for $L=256$ could not be accurately determined and were skipped from the analysis. The remaining data were fitted by
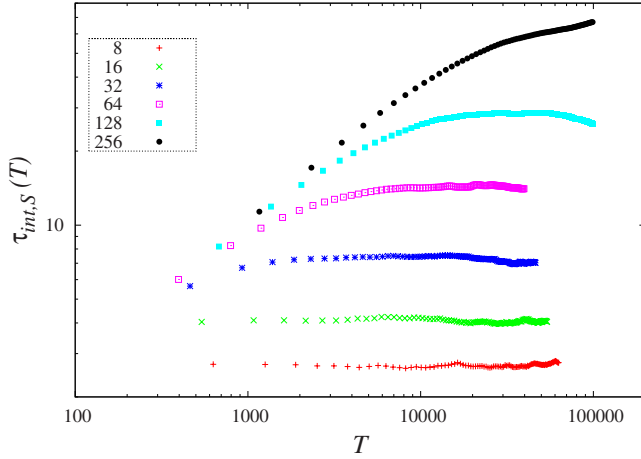
FIG. 6. (Color online) Indefinite integral $\tau_{\mathrm{int},S}(T)$ of the autocorrelation function $A_S(t)$ for the single-cluster size over the time interval $0 < t < T$. These results apply to the single-cluster simulation of the $q=1.25$ random-cluster model.

$$\tau_{\mathrm{int},\mathcal{O}}(L) = A + BL^{z_{\mathrm{int},\mathcal{O}}}, \qquad (28)$$

where A and B are unknown constants. The fits for $q=1.25$ yield $z_{\mathrm{int},m^2}=1.4$ (1), and $z_{\mathrm{int},S}=1.1$ (1). Table IV includes the results for the $z_{\mathrm{int},\mathcal{O}}$ for several other values of q.

IV. DISCUSSION

As stated in Sec. I, one might expect that the present single-cluster algorithm would have a dynamic exponent that is about the same as that of the SWCM algorithm [9]. However, after comparing the dynamic exponents of both algorithms, we find that this expectation is not justified for noninteger q. The single-cluster algorithm formulated in this work represents a new dynamic universality class. Finding the reasons behind this curious fact should help us to better understand from where critical slowing down arises, and tell us something about how one can further develop efficient Monte Carlo algorithms in statistical physics.

The single-cluster algorithm described above is obviously related to the Wolff [4] algorithm as defined for integer-q Potts models; it can reduce to the Wolff method if q is an integer. On the other hand, it is different in the sense that the single-cluster algorithm acts on a mixed configuration of site variables and random-cluster variables.
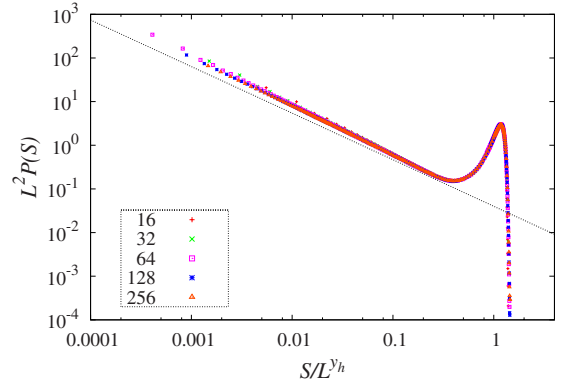

FIG. 7. (Color online) Data collapse for the single-cluster distribution $P(S)$ as a function of the cluster size S for the critical $q=2$ random-cluster model. The dashed line illustrates the asymptotic slope $-2/y_h=16/15$ which applies to $1 \ll S \ll L^{y_h}$. Data are shown for system sizes $L=16$, 32, 64, 128, and 256. The quantity P represents the probability that a randomly chosen site belongs to a cluster of size S.

This mixture of different types of variables is essentially the reason that the present single-cluster algorithm is relatively slow. In this algorithm, a number of lattice sites belongs to random clusters of type 0 with weight $q-[q]$, while the remaining sites are decorated with a Potts variable in one of $[q]$ Potts states.

As described in Sec. II, the only process that can change a type-0 cluster back into an integer spin state, depends on the random selection of a site in that cluster in the beginning of each cluster step. Thus, large clusters of type 0 are short lived and small ones are long lived. It is illustrated in Fig. 7 that the single-cluster distribution for the case $q=2$ displays a wide range of algebraic decay and an additional maximum at large cluster sizes of order L^{y_h}, preceding a rapid decay at even larger sizes. The distribution shown in Fig. 7 represents a time average. Individual cluster decompositions deviate because of thermal fluctuations. The lifetime of these deviations will naturally depend on the cluster size. The smaller the type-0 clusters are, the longer they will persist, and this will be reflected in the decay of the autocorrelation functions. The pronounced maximum in Fig. 7 at $S \approx L^{y_h}$ can thus be associated with a rapid initial decay of the autocorrelations. Once the largest clusters of type 0 are updated, some autocorrelations are still persisting due to the thermal fluctuations of the numbers of smaller clusters that remain to be

TABLE IV. Dynamic exponent z_{int} of the single-cluster cluster algorithm. This exponent describes the scaling behavior of τ_{int}, the integrated autocorrelation function. For a negative exponent z_{int}, the τ_{int} data approach a constant when $L \to \infty$. The values of $z^*_{\mathrm{int},\mathcal{O}}$ are calculated from Eq. (26) and Table III, while those of $z_{\mathrm{int},\mathcal{O}}$ follow from the fits using Eq. (28). Some data are included for integer q; these results apply to the Wolff algorithm.

q	1.25	1.50	1.75	2.25	2.50	2.75	2	3	4
z_{int,m^2}	1.4(1)	1.5(1)	1.6(1)	1.9(1)	1.9(1)	2.0(1)	$-0.16(2)$	0.485(7)	1.005(9)
z^*_{int,m^2}	1.5(2)	1.6(2)	1.7(2)	1.7(2)	1.8(2)	2.0(2)			
$z_{\mathrm{int},S}$	1.1(1)	1.1(1)	1.3(1)	1.6(1)	1.7(1)	1.8(1)	$-0.4(1)$	0.16(4)	0.72(5)
$z^*_{\mathrm{int},S}$	1.3(2)	1.4(2)	1.6(2)	1.4(2)	1.6(2)	1.7(3)			

updated. After t' single-cluster steps, the autocorrelations of the numbers of clusters with sizes $S > L^2/t'$ will be strongly reduced, while the clusters with sizes $S < L^2/t'$ will mostly be unaffected. Since the cluster-size distribution decays algebraically in a range of S, it is natural that autocorrelations associated with clusters that are not yet updated display a corresponding power-law decay in time, as long as the smallest clusters survive. After a number of steps of order L^2 also the clusters of size 1 will be updated. This somewhat qualitative reasoning, which neglects any persisting correlations after all clusters are visited, would mean that the longest autocorrelation time, expressed in single-cluster updates, is of order L^2, after which the autocorrelations will decay exponentially. Expressed in units of t as defined in Sec. III A, this corresponds with autocorrelations scaling as L^{2y_h-2} at criticality. Our numerical results suggest that the dynamic exponent is slightly larger, namely, $z_{exp} \approx 2$, but the data do not allow a more firm statement.

The persistence of the smallest clusters during a time of approximate order L^2 leads to a long-time "tail" during simulations using the single-cluster method. It is this effect that we hold responsible for the relatively large dynamic exponent z_{exp} of the single-cluster method. However, the amplitude of the algebraic decay of the autocorrelation functions still depends with a factor $L^{-s_{\mathcal{O}}}$ on the system size L. A positive value of $s_{\mathcal{O}}$, therefore, means that the critical slow-

ing down is less severe than suggested by the value of z_{exp}, in agreement with the smaller values of z_{int} as shown in Table IV.

Nevertheless our findings indicate that the single-cluster algorithm, apart from displaying interesting dynamic behavior, is not an efficient tool to investigate the two-dimensional random-cluster model. In higher-dimensional systems we have similar expectations. But there still seems to be a possibility that a number of single-cluster steps alternating with a full-cluster decomposition, which takes advantage of the fast initial decay of autocorrelations of the single-cluster algorithm as well as of the absence of a long-time tail in the SWCM cluster algorithm, will be relatively efficient in higher-dimensional systems.

[1] P. W. Kasteleyn and C. M. Fortuin, J. Phys. Soc. Jpn. **26**, 11 (1969).

[2] R. B. Potts, Proc. Cambridge Philos. Soc. **48**, 106 (1952).

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

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

[5] C. F. Baillie and P. D. Coddington, Phys. Rev. B **43**, 10617 (1991).

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

[7] M. Sweeny, Phys. Rev. B **27**, 4445 (1983).

[8] C.-K. Hu, Phys. Rev. Lett. **69**, 2739 (1992).

[9] L. Chayes and J. Machta, Physica A **254**, 477 (1998).

[10] X. Qian, Y. Deng, and H. W. J. Blöte, Phys. Rev. E **71**, 016709 (2005).

[11] Y. J. Deng, T. M. Garoni, J. Machta, G. Ossola, M. Polin, and A. D. Sokal, Phys. Rev. Lett. **99**, 055701 (2007); Y. J. Deng, T. M. Garoni, and A. D. Sokal, *ibid.* **98**, 230602 (2007).

[12] F. Y. Wu, Rev. Mod. Phys. **54**, 235 (1982).

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