

## Sweeny dynamics for the random-cluster model with small $Q$

Zirui Peng,<sup>1</sup> Eren Metin Elçi<sup>2</sup>, Youjin Deng,<sup>1,3,4,\*</sup> and Hao Hu<sup>5,†</sup>

<sup>1</sup>*Department of Modern Physics, University of Science and Technology of China, Hefei, Anhui 230026, China*

<sup>2</sup>*School of Mathematical Sciences, Monash University, Clayton, VIC 3800, Australia*

<sup>3</sup>*Hefei National Laboratory, University of Science and Technology of China, Hefei 230088, China*

<sup>4</sup>*MinJiang Collaborative Center for Theoretical Physics, College of Physics and Electronic Information Engineering, Minjiang University, Fuzhou 350108, China*

<sup>5</sup>*School of Physics and Optoelectronic Engineering, Anhui University, Hefei, Anhui 230601, China*



(Received 2 August 2023; accepted 25 October 2023; published 20 November 2023)

The Sweeny algorithm for the  $Q$ -state random-cluster model in two dimensions is shown to exhibit a rich mixture of critical dynamical scaling behaviors. As  $Q$  decreases, the so-called critical speeding-up for nonlocal quantities becomes more and more pronounced. However, for some quantity of a specific local pattern, e.g., the number of half faces on the square lattice, we observe that, as  $Q \rightarrow 0$ , the integrated autocorrelation time  $\tau$  diverges as  $Q^{-\zeta}$ , with  $\zeta \simeq 1/2$ , leading to the nonergodicity of the Sweeny method for  $Q \rightarrow 0$ . Such  $Q$ -dependent critical slowing-down, attributed to the peculiar form of the critical bond weight  $v = \sqrt{Q}$ , can be eliminated by a combination of the Sweeny and the Kawasaki algorithm. Moreover, by classifying the occupied bonds into bridge bonds and backbone bonds, and the empty bonds into internal-perimeter bonds and external-perimeter bonds, one can formulate an improved version of the Sweeny-Kawasaki method such that the autocorrelation time for any quantity is of order  $O(1)$ .

DOI: [10.1103/PhysRevE.108.055308](https://doi.org/10.1103/PhysRevE.108.055308)

### I. INTRODUCTION

The Hamiltonian of the  $Q$ -state Potts model [1] reads  $\mathcal{H} = -J \sum_{\langle ij \rangle} \delta_{\sigma_i, \sigma_j}$ , where  $J > 0$  is the ferromagnetic coupling constant,  $\langle ij \rangle$  stands for a pair of nearest-neighbor sites, and each lattice site  $i$  has a spin  $\sigma_i$ , taking one of the  $Q$  states  $\sigma_i = 1, 2, \dots, Q$ . For  $Q = 2$ , the Potts model corresponds to the famous Ising model. Given a spin configuration, under the Fortuin-Kasteleyn transformation [2,3], bonds between nearest-neighbor sites with the same spin value are independently occupied with the probability  $p = 1 - e^{-J/kT}$  ( $k$  is the Boltzmann constant,  $T$  is the temperature, and, for convenience, we set  $kT = 1$ ), and the resulting graph is a configuration of the random-cluster model [4]. For bond configurations  $A \subseteq E$  of a given graph  $G = (V, E)$ , the random-cluster model is defined by the partition function

$$Z = \sum_{A \subseteq E} Q^{k(A)} v^{|A|}. \quad (1)$$

Here  $A$  is the set of “occupied bonds,”  $|A|$  is the number of occupied bonds,  $k(A)$  is the number of clusters (connected components), and real positive numbers  $Q$  and  $v = p/(1-p)$  are the cluster and bond weights, respectively. For integer  $Q \geq 2$ , the random-cluster model serves as a graphical representation of the  $Q$ -state Potts model. As  $Q \rightarrow 1$ , the random-cluster model reduces to independent bond percolation [5].

In equilibrium, the weight ratio between two configurations is  $W(\beta)/W(\alpha) = Q^{\delta k(A)} v^{\delta |A|}$ , where  $\delta k(A)$  ( $\delta |A|$ ) represents the changes of  $k(A)$  ( $|A|$ ) from configuration  $\alpha$  to  $\beta$ . As  $Q \rightarrow 0$ , when  $v/Q$  is finite, a configuration consists of a spanning forest [4,6]. This is because in this case  $W(\beta)/W(\alpha) \sim Q^{\delta[k(A)+|A|]}$ , and spanning forests have the minimum  $k(A) + |A|$ . From a spanning forest  $\alpha$ , occupying any internal-perimeter bond (defined later in Fig. 2) leads to a nonforest  $\beta$  with  $\delta[k(A) + |A|] = 1$ , such that  $W(\beta)/W(\alpha) \sim Q \rightarrow 0$ . In the case  $Q \rightarrow 0$  with  $v/Q^\sigma$  ( $0 < \sigma < 1$ ) being finite, similarly the model consists of uniform spanning trees which have the smallest  $k(A) + \sigma |A|$  [4,6].

Markov chain Monte Carlo (MCMC) algorithms [7,8] are effective in simulating the random-cluster model, such as the Swendsen-Wang-Chayes-Machta algorithm [9,10] for  $Q > 1$  with nonlocal cluster updates, and the Sweeny algorithm [11] for  $Q > 0$  with local bond updates. To assess the efficiency of the MCMC algorithms, autocorrelation times are commonly used as a metric [12]. For any observable  $\mathcal{O}$ , one first defines the autocorrelation function:

$$\rho_{\mathcal{O}}(t) = \frac{\langle \mathcal{O}_s \mathcal{O}_{s+t} \rangle - \langle \mathcal{O} \rangle^2}{\text{var}(\mathcal{O})}. \quad (2)$$

Here expectations are taken after equilibrium is attained, and the time unit is one “hit,” in which only one bond is visited (a Monte Carlo sweep consists of  $|E|$  hits, where  $|E|$  is the number of all bonds). Then one defines the exponential autocorrelation time for an observable  $\mathcal{O}$ ,

$$\tau_{\text{exp}, \mathcal{O}} = \limsup_{t \rightarrow \pm\infty} \frac{|t|}{-\log |\rho_{\mathcal{O}}(t)|}, \quad (3)$$

\*yjdeng@ustc.edu.cn

†huhao@ahu.edu.cn

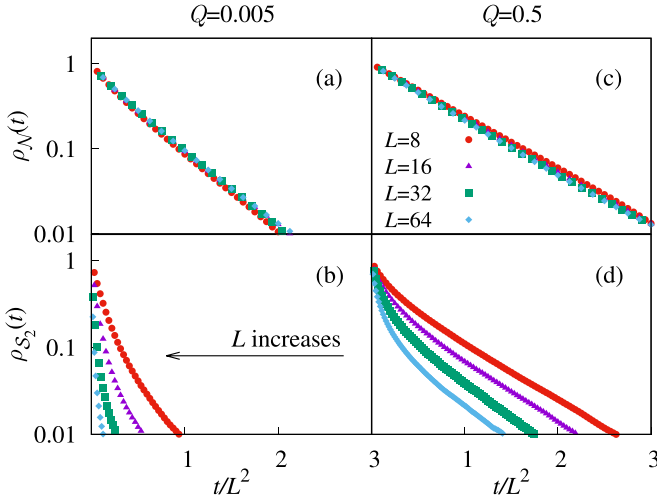


FIG. 1. Autocorrelation function  $\rho_{S_2}(t)$  and  $\rho_{\mathcal{N}}(t)$  for the critical two-dimensional random-cluster model at  $Q = 0.005$  and  $0.5$  on the square lattice. The unit of  $t$  is one hit.

and the overall exponential autocorrelation time:

$$\tau_{\text{exp}} = \sup_{\mathcal{O}} \tau_{\text{exp}, \mathcal{O}}. \quad (4)$$

The integrated autocorrelation time for an observable  $\mathcal{O}$  is defined as

$$\tau_{\text{int}, \mathcal{O}} = \frac{1}{2} \sum_{t=-\infty}^{\infty} \rho_{\mathcal{O}}(t). \quad (5)$$

Observables typically have  $\tau_{\text{exp}, \mathcal{O}} = \tau_{\text{exp}}$  and  $\tau_{\text{int}, \mathcal{O}} \leq \tau_{\text{exp}}$ . The autocorrelation times  $\tau_{\text{exp}}$  and  $\tau_{\text{int}, \mathcal{O}}$  play different roles in MCMC simulations [12]:  $\tau_{\text{exp}}$  is the relaxation time of the slowest mode in the system, and  $\tau_{\text{int}, \mathcal{O}}$  controls the statistical error in the measurement of  $\langle \mathcal{O} \rangle$ . Thus  $\tau_{\text{int}, \mathcal{O}}$  mainly determines the efficiency of MCMC sampling. For a time series of total length  $M$ , the sample mean

$$\hat{\mathcal{O}} = \frac{1}{M} \sum_{t=1}^M \mathcal{O}_t \quad (6)$$

has variance

$$\text{var}(\hat{\mathcal{O}}) \approx 2\tau_{\text{int}, \mathcal{O}} \frac{\text{var}(\mathcal{O})}{M}, \quad \text{when } M \rightarrow \infty. \quad (7)$$

The above Eq. (7) can be used to measure  $\tau_{\text{int}, \mathcal{O}}$ .

In MCMC simulations, the dynamic processes usually undergo *critical slowing-down* [13]: the autocorrelation time diverges as  $\tau/|E| \sim \xi^z$  when the critical point is being approached, where  $\xi$  is the spatial correlation length and  $z$  is called the dynamic exponent. For the two-dimensional Ising model, Metropolis-Hastings [14,15] and Glauber [8,16] algorithms have  $z \approx 2$ . To overcome the *critical slowing-down*, the Swendsen-Wang-Chayes-Machta algorithm [9,10] was proposed for the random-cluster model with  $Q > 1$ . The latter algorithm has  $z \approx 0.2$  for the two-dimensional Ising model, which represents a huge improvement. The gained efficiency is associated with nonlocal cluster updates.

However, some local algorithms may still be very efficient, such as the worm algorithm [17] and the Sweeny algorithm

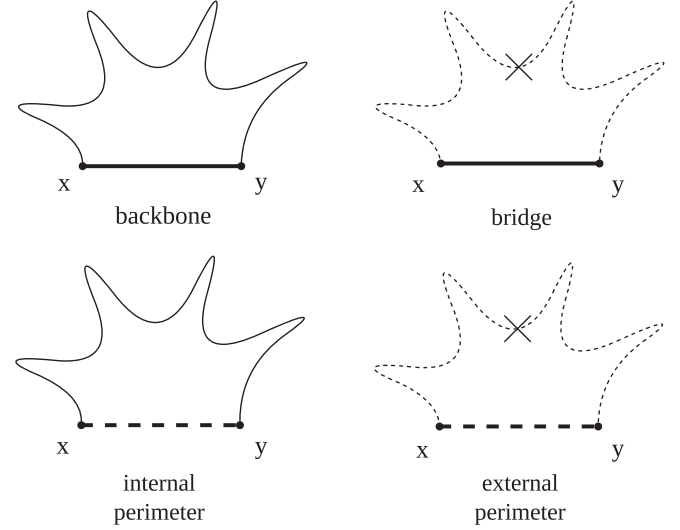


FIG. 2. Schematic diagrams for four kinds of bonds in the random-cluster model. They are categorized according to their occupation and connectivity: if bond  $xy$  is occupied and the two end points are connected (resp. not connected) via a path that does not include  $xy$ , then the occupied bond is called a *backbone bond* (resp. *bridge bond*); if bond  $xy$  is empty and the two end points are connected (resp. not connected) via a path that does not include  $xy$ , then the empty bond is referred to as an *internal-perimeter bond* (resp. *external-perimeter bond*).

[18]. The worm algorithm outperforms the cluster algorithms for the three-dimensional Ising model in measuring quantities such as the susceptibility and correlation length. With local single-bond updates, the Sweeny algorithm can surprisingly exhibit *critical speeding-up* for a suitable range of  $Q$  [18]: the relaxation time of some observables decreases significantly at time scales much shorter than one sweep as the lattice size  $L$  increases, and this phenomenon becomes even more pronounced as  $Q$  becomes smaller. We illustrate the latter property of the Sweeny algorithm by observing the dynamical behavior of two quantities, i.e., the number of occupied bonds  $\mathcal{N} = |A|$  and the second moment of cluster sizes  $S_2 = \sum |\mathcal{C}|^2$ . Hereafter, if not specified, all simulations were performed on the square lattice at the critical point  $v_c(Q) = \sqrt{Q}$ . The average  $\langle S_2 \rangle / |V|$  is a susceptibilitylike quantity of the random-cluster model [2,3], with  $|V|$  being the volume. For  $Q \geq 2$ , it is found [18] that the dynamic exponent  $z_{\text{exp}} \approx z_{\text{int}, \mathcal{N}}$ , and that both  $z_{\text{int}, \mathcal{N}}$  and  $z_{\text{int}, S_2}$  are only slightly larger than the lower bound  $\alpha/\nu$  [19], and could possibly be equal to it (here  $\alpha$  and  $\nu$  are critical exponents for the specific heat and correlation length, respectively). For  $Q < 2$ , the autocorrelation function of  $\mathcal{N}$  follows an almost purely exponential function, as demonstrated for  $Q = 0.005$  and  $0.5$  in Figs. 1(a) and 1(c), respectively. However, the dynamic behavior of  $S_2$  is richer [18],

(1) As shown in Figs. 1(b) and 1(d), the autocorrelation function  $\rho_{S_2}(t)$  decays very rapidly in much less than one sweep, as a prelude to the final exponential decay  $\sim e^{-t/\tau_{\text{exp}}}$ .

(2) The exponent  $z_{\text{int}, S_2}$  is a negative number, and it decreases with the increasing of lattice size  $L$ .

(3) From Figs. 1(d) to 1(b), it is seen that the critical speeding-up is more pronounced for smaller  $Q$ .

In this paper, we are interested in the dynamic behavior of the Sweeny algorithm for very small  $Q$  values. A  $Q$ -dependent critical slowing-down is found for the random-cluster model in two dimensions. The phenomenon can be observed in the dynamics of a specific local pattern on the square lattice, i.e., a unit square with configuration  $\begin{array}{|c|} \hline \text{I} \\ \hline \end{array}$  or  $\begin{array}{|c|} \hline \text{II} \\ \hline \end{array}$ , where solid lines represent occupied bonds. We call the local pattern a *half-face*. We find that, when  $Q \rightarrow 0$ , the integrated autocorrelation time of the number of half faces in terms of sweeps (i.e.,  $\tau_{\text{int}, \mathcal{N}_{\text{HF}}}/2L^2$ , and hereafter  $\tau_{\mathcal{N}_{\text{HF}}}/L^2$  is used for brevity) diverges as  $Q^{-1/2}$ . This indicates that the Sweeny algorithm becomes not ergodic as  $Q \rightarrow 0$ .

The above  $Q$ -dependent critical slowing-down can be explained by the acceptance probability of updates in the Sweeny algorithm. As illustrated in Fig. 2, considering different occupation and connectivity, the bonds can be classified into four types—the *backbone*, *bridge*, *internal-perimeter* and *external-perimeter* bonds. At  $v_c = \sqrt{Q}$ , when transforming an internal-perimeter bond into a backbone bond, one has  $\delta k(A) = 0$  and  $\delta|A| = 1$ , which yields the weight ratio  $W(\beta)/W(\alpha) = \sqrt{Q}$ , and transforming a bridge bond into an external-perimeter bond leads to  $\delta k(A) = 1$  and  $\delta|A| = -1$ , which also gives  $W(\beta)/W(\alpha) = \sqrt{Q}$ . It is shown in Sec. II that these weight ratios are associated with vanishing acceptance probabilities  $\sqrt{Q}$  as  $Q \rightarrow 0$ . To overcome the  $Q$ -dependent slowing-down, we first use the knowledge of four kinds of bonds in the detail balance condition and propose an improved Sweeny algorithm. The improved Sweeny algorithm largely alleviates the slowing-down but does not fully solve the problem. For example, at  $Q = 0$  where the graphs become spanning trees, the improved Sweeny algorithm is still invalid due to zero probability of deleting or adding a bond. We then employ the Kawasaki algorithm [20], which proposes the random exchange of an empty and an occupied bond. At  $Q = 0$  on spanning trees, the Kawasaki algorithm still has an  $L$ -dependent divergence  $\tau_{\mathcal{N}_{\text{HF}}}/L^2 \sim L^{3/2}$  which can also be related to the acceptance probability of updates. The latter divergence can be eliminated through an improved Kawasaki algorithm which uses the categories of bonds. Finally, we combine the use of the improved Sweeny and improved Kawasaki algorithms, and find no critical slowing-down for small  $Q$ , i.e., the autocorrelation time  $\tau/L^2$  is of order  $\mathcal{O}(1)$ , being independent of  $Q$ .

The remainder of this paper is organized as follows. Section II describes the Sweeny algorithm and the  $Q$ -dependent slowing-down. Section III presents an improved Sweeny algorithm which can greatly improve simulation efficiency. Section IV introduces an improvement of the Kawasaki algorithm and its combination with the improved Sweeny algorithm, which can completely eliminate the  $Q$ -dependent slowing-down. Section V contains a discussion and conclusion.

## II. SWEENEY ALGORITHM AND $Q$ -DEPENDENT SLOWING-DOWN

As mentioned in the Introduction, the Sweeny algorithm is a local single-bond update algorithm for the random-cluster model. Actually, it is the only known MCMC algorithm for

the model with  $0 < Q < 1$ . The original Sweeny algorithm [11] is presented in Algorithm 1. A basic step of the Sweeny algorithm is proposing to change the status of a randomly selected bond, and accepting the proposal according to the Metropolis criterion.

For an efficient implementation of the Sweeny algorithm, the main challenge is in checking the connectivity between the two end points of a chosen bond. There are three typical methods with different asymptotic run-time scaling: breadth-first search, union-and-find, and dynamic connectivity algorithms [21,22]. We use the method of simultaneous breadth-first searches starting at both end points of a bond  $xy$ , since it has very simple code and is effective enough for small and medium size  $L$  [23]. We note that a simple modification of Algorithm 1 can save some time by not requiring connectivity checking for a fraction of updates. The modification for  $Q \leq 1$  at  $v_c = \sqrt{Q}$  is shown in Algorithm 2. The basic idea is to consider together several probabilities in Algorithm 1: if the generated random number is less than the minimum of these probabilities, the update must be accepted, and if it is greater than the maximum of these probabilities, the update must be rejected. For  $Q \rightarrow 1$ , this small change in the algorithm can lead to a significant improvement of efficiency.

---

### Algorithm 1 Original Sweeny Algorithm.

---

```

1 Choose a bond  $xy$  uniformly at random
2 if  $xy$  is an external-perimeter bond then
3   | Occupy  $xy$  with probability  $\min\{1, v/Q\}$ 
4 else if  $xy$  is an internal-perimeter bond then
5   | Occupy  $xy$  with probability  $\min\{1, v\}$ 
6 else if  $xy$  is a bridge bond then
7   | Delete  $xy$  with probability  $\min\{1, Q/v\}$ 
8 else if  $xy$  is a backbone bond then
9   | Delete  $xy$  with probability  $\min\{1, 1/v\}$ 
10 end

```

---

As described in the Introduction, the Sweeny algorithm exhibits rich dynamics, including both critical slowing-down and critical speeding-up. However, its dynamics at small  $Q$  values are not uncovered fully. For Algorithms 1 and 2, one can observe that they encounter a problem when  $Q$  is small: at  $v_c = \sqrt{Q}$ , probabilities of some updates (occupying an internal-perimeter bond and deleting a bridge bond) are proportional to  $\sqrt{Q}$ , which causes the algorithm being not ergodic at  $Q \rightarrow 0$ . The problem was not severe in previous studies, since the studied  $Q$  values are not extremely small, and conventional observables such as  $\mathcal{N}$  and  $\mathcal{S}_2$  could not detect any slowing-down effect due to the  $\sqrt{Q}$ -form updating probabilities. Actually on the square lattice in the limit  $Q \rightarrow 0$ , a critical equilibrium configuration consists of a spanning tree, thus the above observables remain fixed as  $\mathcal{N} = L^2 - 1$  and  $\mathcal{S}_2 = L^4$  for a given  $L$ .

We find that the above problem can become dominant in sampling other quantities, such as the number of half faces  $\mathcal{N}_{\text{HF}}$ . Using the original Sweeny algorithm, we calculated  $\tau_{\mathcal{N}_{\text{HF}}}$  as plotted in Fig. 3. It can be seen that  $\tau_{\mathcal{N}_{\text{HF}}}/L^2$  diverges as  $Q^{-1/2}$ , independent of the system size  $L$ . We refer to this as  *$Q$ -dependent critical slowing-down*. This newly discovered phenomenon at small  $Q$  enriches the dynamics of the Sweeny

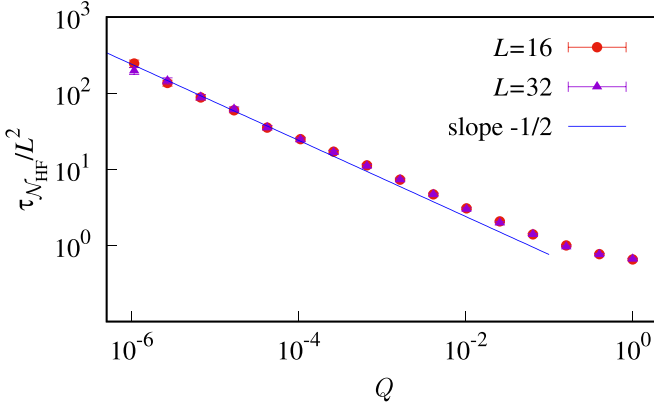


FIG. 3. Autocorrelation time of the number of half faces for the original Sweeny algorithm.  $\tau_{\mathcal{N}_{\text{HF}}}/L^2$  diverges as  $Q^{-1/2}$  when  $Q \rightarrow 0$ .

algorithm. It is important to note that, away from  $Q = 0$ , the original Sweeny algorithm is not inherently flawed for small  $Q$ . For example, when  $Q = 10^{-6}$ , the autocorrelation time  $\tau_{\mathcal{N}_{\text{HF}}}/L^2$  is of order  $O(10^2)$ , which means that  $\mathcal{N}_{\text{HF}}$  can still be measured rather efficiently.

To understand the  $Q$ -dependent critical slowing-down, we associate the above factor  $Q^{-1/2}$  to the updating probabilities in the Sweeny algorithm. In the following we measure the numbers of the aforementioned four kinds of bonds, and then give these updating probabilities.

On the square lattice in the limit  $Q \rightarrow 0$ , the number of bridge bonds ( $\mathcal{N}_{\text{BR}}$ ) and that of internal-perimeter bonds ( $\mathcal{N}_{\text{IP}}$ ) are  $\mathcal{N}_{\text{BR}} \rightarrow L^2 - 1$  and  $\mathcal{N}_{\text{IP}} \rightarrow |E| - \mathcal{N}_{\text{BR}}$ , respectively. These numbers can be obtained from the characteristics of a spanning tree: since every node (i.e., site) corresponds to an occupied bond except the root, there are  $L^2 - 1$  occupied bonds for a tree with  $L^2$  nodes; the  $L^2 - 1$  occupied bonds are all bridge bonds, and other (empty) bonds are internal-perimeter bonds. For finite small  $Q$ , changes of the numbers of four kinds of bonds with the decrease of  $Q$  are shown in Fig. 4, with fitting results for two of them in Table I. For  $L = 16$ , we find that for small  $Q$ , the average number of backbone bonds ( $\mathcal{N}_{\text{BB}}$ ) and that of external-perimeter bonds ( $\mathcal{N}_{\text{EP}}$ ) are, respectively,

**Algorithm 2** Modified Sweeny Algorithm (for  $Q \leq 1$  at  $v_c = \sqrt{Q}$ ).

- 
- 1  $P_{\min} = \sqrt{Q}$ ,  $P_{\max} = 1$
  - 2 Choose a bond  $xy$  uniformly at random
  - 3 Generate a random number  $P \in [0, 1)$
  - 4 **if**  $P \leq P_{\min}$  **then**
  - 5 Change the status of  $xy$
  - 6 **else if**  $xy$  is an external-perimeter bond or a backbone bond **then**
  - 7 Change the status of  $xy$
  - 8 **end**
- 

$$\langle \mathcal{N}_{\text{BB}} \rangle \simeq 0.94L^2\sqrt{Q}, \quad \langle \mathcal{N}_{\text{EP}} \rangle \simeq L^2\sqrt{Q}, \quad (8)$$

both being proportional to  $\sqrt{Q}$ , as plotted in Fig. 4(a). The average number of occupied bonds is found to be  $\langle \mathcal{N}_B \rangle \simeq L^2 - 1$

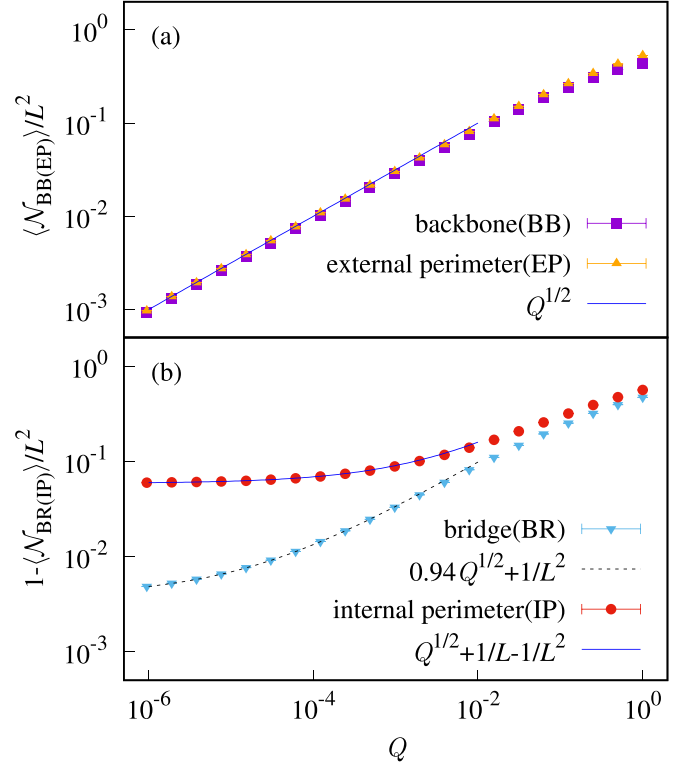


FIG. 4. Average numbers of four types of bonds vs  $Q$  at fixed  $L = 16$ . The  $Q^{1/2}$  behavior in panel (a) is confirmed by fits in Table I, and the curves in panel (b) can be derived from Eq. (9) in the main text.

for small  $Q$ , which is consistent with  $\mathcal{N}_{\text{BR}} \rightarrow L^2 - 1$  in the limit  $Q \rightarrow 0$ . In the simulations, for convenience of a loop construction which will be described in Sec. III, we use periodic boundary conditions in one direction of the square lattice and open boundary conditions in the other direction. The total number of bonds on the lattice is  $|E| = 2L^2 - L$ . Thus, considering  $\mathcal{N}_B = \mathcal{N}_{\text{BB}} + \mathcal{N}_{\text{BR}}$  and  $|E| - \mathcal{N}_B = \mathcal{N}_{\text{EP}} + \mathcal{N}_{\text{IP}}$ , we derive

$$\begin{aligned} \langle \mathcal{N}_{\text{BR}} \rangle &\simeq L^2 - 0.94L^2\sqrt{Q} - 1 \simeq L^2(1 - 0.94\sqrt{Q}), \\ \langle \mathcal{N}_{\text{IP}} \rangle &\simeq L^2 - L^2\sqrt{Q} - L + 1 \simeq L^2(1 - \sqrt{Q}). \end{aligned} \quad (9)$$

The above scaling behaviors are demonstrated by a plot of  $1 - \langle \mathcal{N}_{\text{BR(IP)}} \rangle / L^2$  as shown in Fig. 4(b). In the thermodynamic limit  $L \rightarrow \infty$ , from the self-duality of the square lattice, the critical numbers of four kinds of bonds can be obtained [24] (an alternative derivation is given in the Appendix of the present paper) as

$$\begin{aligned} \langle \mathcal{N}_{\text{BB}} \rangle / L^2 &= \langle \mathcal{N}_{\text{EP}} \rangle / L^2 = \frac{\sqrt{Q}}{1 + \sqrt{Q}}, \\ \langle \mathcal{N}_{\text{BR}} \rangle / L^2 &= \langle \mathcal{N}_{\text{IP}} \rangle / L^2 = \frac{1}{1 + \sqrt{Q}}. \end{aligned} \quad (10)$$

In the limit  $Q \rightarrow 0$ , the factor 0.94 in Eqs. (8) and (9) deviates from 1 in Eq. (10) due to finite-size effects.

With the above bond numbers for small  $Q$ , we can get the updating probabilities in the Sweeny algorithm as following: the probability of selecting a bridge bond or internal-perimeter

TABLE I. Fitting results for  $\langle \mathcal{N}_{EP} \rangle / L^2$  and  $\langle \mathcal{N}_{BB} \rangle / L^2$  at  $L = 16$ . The data were fitted by the formula  $Q^{y_Q}(a_0 + b_1 Q^{y_Q} + b_2 Q^{2y_Q})$  using the least-square method, with an upper cutoff  $Q_{\max}$  on included values of  $Q$ .

	$Q_{\max}$	$y_Q$	$a_0$	$b_1$	$b_2$	$\chi^2/\text{DF}$
$\langle \mathcal{N}_{EP} \rangle / L^2$	0.015625	0.50001(8)	0.9959(8)	-0.940(10)	0.762081(1)	10.2/11
	0.0078125	0.5000(1)	0.996(1)	-0.95(2)	0.803890(1)	10.1/10
	0.00390625	0.5000(1)	0.996(2)	-0.95(3)	0.840794(1)	10.0/9
	0.00195313	0.5002(2)	0.999(2)	-1.02(6)	1.775890(1)	8.0/8
$\langle \mathcal{N}_{BB} \rangle / L^2$	0.015625	0.49989(8)	0.9404(8)	-0.969(10)	0.796009(1)	14.7/11
	0.0078125	0.4999(1)	0.941(1)	-0.97(2)	0.815144(1)	14.6/10
	0.00390625	0.5001(1)	0.942(2)	-1.02(3)	1.284402(1)	11.3/9
	0.00195313	0.4999(2)	0.940(2)	-0.95(6)	0.401849(1)	9.5/8

bond is  $\approx 1$ , and the acceptance probability ( $Q/v$  or  $v$ ) is  $\sqrt{Q}$ , then the associated updating probability is  $\sim \sqrt{Q}$ ; the probability of selecting a backbone bond or external-perimeter bond is  $\sim \sqrt{Q}$ , and the acceptance probability is 1, then the associated updating probability is also  $\sim \sqrt{Q}$ . Thus the  $Q^{-1/2}$ -form critical slowing-down is associated with the  $\sqrt{Q}$ -form updating probabilities.

### III. AN IMPROVED SWEENEY ALGORITHM

To overcome the  $Q$ -dependent critical slowing-down, we introduce an improved Sweeny algorithm as shown in Algorithm 3. Instead of selecting a bond at random, the improved algorithm randomly proposes one of the following four operations: deleting a backbone bond (delete-backbone), deleting a bridge bond (delete-bridge), adding a backbone bond (add-backbone), and adding a bridge bond (add-bridge). Acceptance probabilities of the operations shall be given later. The algorithm consists of recording and dynamically updating the information of four kinds of bonds, which are directly used in proposing MCMC updates. To the end of this section, we shall see that the improved Sweeny algorithm can largely solve the  $Q$ -dependent critical slowing-down.

**Algorithm 3** Improved Sweeny Algorithm.

- ```

1 Randomly choose an operation: delete-backbone,
  delete-bridge, add-backbone, or add-bridge
2 if delete-backbone then
3   Randomly pick up a backbone bond and delete it
  with probability  $P_{BB}^{(-)}$ 
4 else if delete-bridge then
5   Randomly pick up a bridge bond and delete it
  with probability  $P_{BR}^{(-)}$ 
6 else if add-backbone then
7   Randomly pick up an internal-perimeter bond and
  occupy it (i.e., add a backbone bond) with
  probability  $P_{BB}^{(+)}$ 
8 else if add-bridge then
9   Randomly pick up an external-perimeter bond and
  occupy it (i.e., add a bridge bond) with
  probability  $P_{BR}^{(+)}$ 
10 end

```

For Algorithm 3, a list is created to dynamically keep the information for each kind of bonds. The lists record the

indices of corresponding bonds, and the numbers of four types of bonds, i.e.,  $\mathcal{N}_{BB}$ ,  $\mathcal{N}_{BR}$ ,  $\mathcal{N}_{EP}$ , and  $\mathcal{N}_{IP}$ . The bonds are classified by using the Baxter-Kelland-Wu loops [25] on the medial lattice, as exemplified in Fig. 5. The loops are constructed by assigning each occupied bond two loop arcs to its two sides, and assigning each unoccupied bond two loop arcs which cross the bond. The types of bonds are determined as following: an occupied bond is a backbone (bridge) bond if and only if the loop arcs on its two sides do not (do) belong to the same loop, and an unoccupied bond is an external (internal) perimeter bond if the two arcs crossing the bond do not (do) belong to the same loop. To avoid the case that an occupied bond is a backbone bond but has its two loop arcs in the same loop, the boundary conditions are chosen as periodic in one direction and open in the other direction such that the finite- $L$  square lattice is planar. During the update of a bond state, the connectivity checking and labeling of loop arcs in the loop configuration are performed by simultaneous breadth-first searches starting from two loop arcs of the bond [21,23]. When a loop is split into two or two loops are merged into one during the update, the simultaneous searching time is only proportional to the size of the smaller loop.

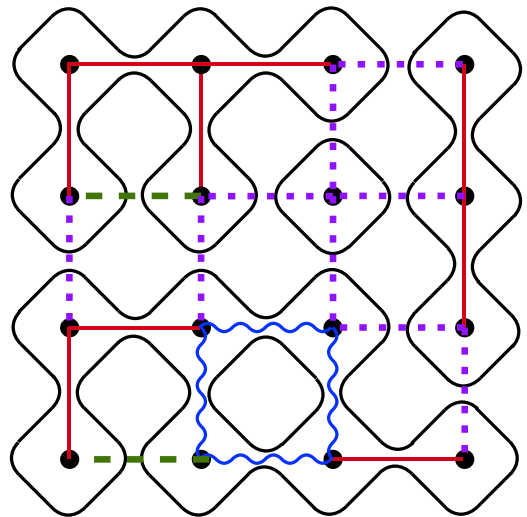
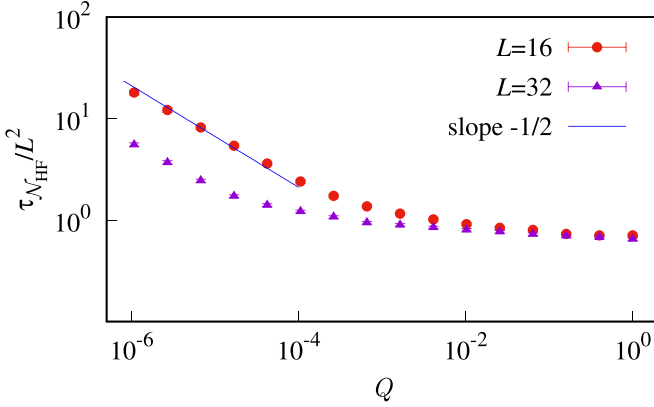


FIG. 5. Example bond configuration and corresponding Baxter-Kelland-Wu loops on the square lattice. Red lines are bridge bonds, blue wavy lines are backbone bonds, green dashed lines represent internal-perimeter bonds, and purple dotted lines stand for external-perimeter bonds.


 FIG. 6.  $\tau_{\mathcal{N}_{\text{HF}}}$  for the improved Sweeny algorithm.

The acceptance probabilities in Algorithm 3 are derived as follows. Using  $\alpha$  and  $\beta$  to represent two configurations that can transform into each other in an update, the detailed balance condition has the form

$$\begin{aligned} \pi(\alpha)P_{\text{pro}}(\alpha \rightarrow \beta)P_{\text{acc}}(\alpha \rightarrow \beta) \\ = \pi(\beta)P_{\text{pro}}(\beta \rightarrow \alpha)P_{\text{acc}}(\beta \rightarrow \alpha), \end{aligned} \quad (11)$$

where  $\pi$  is the configuration weight, and  $P_{\text{pro}}$  and  $P_{\text{acc}}$  are the proposal and acceptance probabilities, respectively. The detailed balance condition associated with the deletion or addition of a backbone bond can be written as

$$1 \times \frac{1}{\mathcal{N}_{\text{BB}}(\alpha)} \times P_{\text{BB}}^{(-)} = \frac{1}{v} \times \frac{1}{\mathcal{N}_{\text{IP}}(\beta)} \times P_{\text{BB}}^{(+)}, \quad (12)$$

and that associated with the deletion or addition of a bridge bond as

$$1 \times \frac{1}{\mathcal{N}_{\text{BR}}(\alpha)} \times P_{\text{BR}}^{(-)} = \frac{Q}{v} \times \frac{1}{\mathcal{N}_{\text{EP}}(\beta)} \times P_{\text{BR}}^{(+)}. \quad (13)$$

Following the Metropolis-Hastings criterion, the acceptance probabilities are calculated as

$$P_{\text{BB}}^{(-)} = \min \left\{ 1, \frac{\mathcal{N}_{\text{BB}}(\alpha)}{\mathcal{N}_{\text{IP}}(\beta)} \times \frac{1}{v} \right\}, \quad (14)$$

$$P_{\text{BB}}^{(+)} = \min \left\{ 1, \frac{\mathcal{N}_{\text{IP}}(\beta)}{\mathcal{N}_{\text{BB}}(\alpha)} \times v \right\}, \quad (15)$$

$$P_{\text{BR}}^{(-)} = \min \left\{ 1, \frac{\mathcal{N}_{\text{BR}}(\alpha)}{\mathcal{N}_{\text{EP}}(\beta)} \times \frac{Q}{v} \right\}, \quad (16)$$

$$P_{\text{BR}}^{(+)} = \min \left\{ 1, \frac{\mathcal{N}_{\text{EP}}(\beta)}{\mathcal{N}_{\text{BR}}(\alpha)} \times \frac{v}{Q} \right\}. \quad (17)$$

Our results of  $\tau_{\mathcal{N}_{\text{HF}}}$  at  $v_c = \sqrt{Q}$  for the improved Sweeny algorithm are shown in Fig. 6. When fixing the size  $L$ , it can be seen that, as  $Q$  decreases,  $\tau_{\mathcal{N}_{\text{HF}}}/L^2$  is almost independent of  $Q$  before a turning point  $Q_c(L)$ , after which  $\tau_{\mathcal{N}_{\text{HF}}}/L^2$  tends to increase as  $\sim Q^{-1/2}$ . The value of  $Q_c(L)$  decreases with increasing system size  $L$ , and it is expected that  $Q_c(L) \rightarrow 0$  in the limit  $L \rightarrow \infty$ . Thus, comparing with results of the original Sweeny algorithm (i.e., Fig. 3), the improved Sweeny algorithm greatly improves simulation efficiency. For  $Q = 10^{-6}$  and  $L = 32$ , the autocorrelation time for the improved algorithm is of order  $O(10)$ , and it decreases with  $L$ , making the algorithm sufficient for most studies of

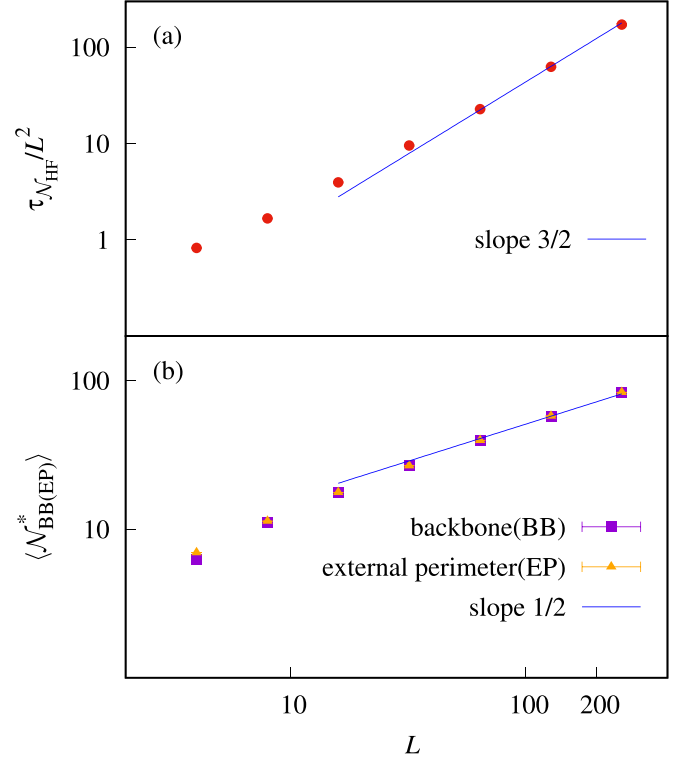


FIG. 7. Results for the uniform spanning trees on the square lattice. (a) The autocorrelation time of the Kawasaki algorithm  $\tau_{\mathcal{N}_{\text{HF}}}/L^2$  vs  $L$ . Here each swap of two bond states is regarded as the time unit hit. (b) The average numbers of bonds in the intermediate state  $\langle \mathcal{N}_{\text{BB}}^* \rangle$  and  $\langle \mathcal{N}_{\text{EP}}^* \rangle$  vs  $L$ . Deviations at small sizes represent finite-size corrections.

the random-cluster model with small  $Q$ . The performance of the improved Sweeny algorithm can be understood from the acceptance probabilities:  $P_{\text{BB}}^{(+)}$  and  $P_{\text{BR}}^{(-)}$  are of order  $O(1)$  for  $Q \gg Q_c(L)$ ;  $P_{\text{BB}}^{(-)}$  and  $P_{\text{BR}}^{(+)}$  are of order  $O(1)$  for all  $Q$ . The residual  $Q^{-1/2}$  behavior is attributed to the fact that  $P_{\text{BB}}^{(+)}$  and  $P_{\text{BR}}^{(-)}$  are of order  $O(\sqrt{Q})$  for  $Q \ll Q_c(L)$ . Detailed analyses of these acceptance probabilities will be presented in Sec. IV. In the limit  $Q \rightarrow 0$ , the improved Sweeny algorithm fails since  $P_{\text{BR}}^{(-)} = P_{\text{BB}}^{(+)} = 0$ , i.e., no bridge bond can be deleted from a uniform spanning tree and no backbone bond can be added to the tree.

#### IV. AN IMPROVED KAWASAKI ALGORITHM AND ITS COMBINATION WITH THE IMPROVED SWEENY ALGORITHM

In the limit  $Q \rightarrow 0$ , since both the original and improved Sweeny algorithms are invalid, one has to employ other methods. When  $Q \rightarrow 0$ , though no bond can be erased or occupied, bonds may exchange their states. Thus, to simulate models with  $Q \rightarrow 0$ , we can use the Kawasaki algorithm [20], in which a randomly selected empty bond is proposed to switch with a randomly selected occupied bond. We applied the Kawasaki algorithm to the uniform spanning trees, which appear in the limit of  $Q \rightarrow 0$  with  $v_c/Q^{1/2}$  being finite, for the critical square-lattice model. The results are shown in Fig. 7(a), where it can be seen that  $\tau_{\mathcal{N}_{\text{HF}}}/L^2$  diverges as  $\sim L^{3/2}$ .

We understand this  $L$ -dependent slowing-down as follows. A Kawasaki update can be performed in two steps: first deleting a randomly chosen occupied bond so that the spanning tree is split into two trees, then occupying a randomly chosen empty bond. We call the configuration between the two steps an intermediate state. For a uniform spanning tree, the overall update would be accepted if and only if the chosen empty bond is an external-perimeter bond in the intermediate state. The number of external-perimeter bonds in the intermediate states  $\mathcal{N}_{\text{EP}}^*$  can be counted by observing changes of the loops: on a spanning tree, the deletion of an occupied bond decomposes a large loop into two loops, and  $\mathcal{N}_{\text{EP}}^*$  is proportional to the size of the external perimeter associated with the smaller loop. The loop surrounding a cluster defines its hull, and the fraction of loop arcs in the hull corresponding to external-perimeter bonds is called the external perimeter of the cluster. The size of the smaller loop is determined as  $\sim L^{d_{\text{H}}-x_2}$  in Ref. [23], where  $d_{\text{H}}$  is the fractal dimension of the hull and  $x_2$  is the two-arm exponent [26]. Analogously, we conjecture that the size of the external perimeter associated with the smaller loop is  $\sim L^{d_{\text{EP}}-x_2}$ , where  $d_{\text{EP}}$  is the fractal dimension of the external perimeter. According to the literature [23,26–28], the fractal dimensions take values  $d_{\text{H}} = 1 + 2/g$  and  $d_{\text{EP}} = 1 + g/8$ ; the parameter  $g$  and  $k$ -arm exponents  $x_k$  are given by

$$Q = 2 + 2 \cos(g\pi/2), \quad g \in [2, 4], \quad (18)$$

$$x_k = (g/8)k^2 - (g-4)^2/(8g), \quad k \geq 2. \quad (19)$$

For  $Q \rightarrow 0$ , one gets  $g = 2$ ,  $x_2 = 3/4$ , and  $d_{\text{EP}} = 5/4$ . Then we expect  $\langle \mathcal{N}_{\text{EP}}^* \rangle \sim L^{5/4-3/4} \sim L^{1/2}$ , which is numerically confirmed in Fig. 7(b). Thus, the probability that a randomly selected empty bond is an external-perimeter bond is  $\langle \mathcal{N}_{\text{EP}}^* \rangle / 2L^2 \sim L^{-3/2}$ , which leads to  $\tau_{\mathcal{N}_{\text{HF}}} / L^2 \sim L^{3/2}$  on the uniform spanning trees. To eliminate the above  $L$ -dependent slowing-down, similar to the idea in improving the Sweeny algorithm, we make use of the real-time information of four kinds of bonds and propose an improved Kawasaki algorithm, as shown in Algorithm 4. This improved algorithm either exchanges a bridge bond with an external-perimeter bond, or exchanges a backbone bond with an internal-perimeter bond. Since the proposed update always exists and the acceptance probability is 1, the improved Kawasaki algorithm does not have the above  $L$ -dependent slowing-down. This is confirmed by our simulation results which have  $\tau_{\mathcal{N}_{\text{HF}}} / L^2 \simeq 0.55$  for  $L = 4$  to 128.

---

**Algorithm 4** Improved Kawasaki Algorithm.

---

- 1 Randomly choose an occupied bond and erase the bond, then we have an intermediate configuration
  - 2 **if** *The chosen bond is a bridge bond* **then**
  - 3 | Randomly occupy an external-perimeter bond in the intermediate configuration
  - 4 **else if** *The chosen bond is a backbone bond* **then**
  - 5 | Randomly occupy an internal-perimeter bond in the intermediate configuration
  - 6 **end**
- 

For nonzero small  $Q$ , since updates in the (improved) Kawasaki algorithm cannot change the total number of oc-

cupied bonds, one has to employ the (improved) Sweeny algorithm. In Sec. III, for the improved Sweeny algorithm at fixed  $L$ , we have shown the appearance of a  $Q$ -dependent slowing-down after a turning point  $Q_c(L)$ . This can be understood by observing the acceptance probabilities as follows. First, for updates of deleting a backbone bond or adding a bridge bond, substituting Eqs. (8) and (9) into Eqs. (14) and (17), one has

$$P_{\text{BB}}^{(-)} = \min \left\{ 1, \frac{\mathcal{N}_{\text{BB}}(\alpha)}{\mathcal{N}_{\text{IP}}(\alpha) + 1} \times \frac{1}{\sqrt{Q}} \right\} \simeq 1, \quad (20)$$

$$P_{\text{BR}}^{(+)} = \min \left\{ 1, \frac{\mathcal{N}_{\text{EP}}(\beta)}{\mathcal{N}_{\text{BR}}(\beta) + 1} \times \frac{1}{\sqrt{Q}} \right\} \simeq 1, \quad (21)$$

where  $\mathcal{N}_{\text{IP}}(\alpha) + 1 = \mathcal{N}_{\text{IP}}(\beta)$  and  $\mathcal{N}_{\text{BR}}(\beta) + 1 = \mathcal{N}_{\text{BR}}(\alpha)$  are used. Then, for updates of deleting a bridge bond or adding a backbone bond, more considerations are needed. For example, after deleting a bridge from configuration  $\alpha$ , there is no simple relation between  $\mathcal{N}_{\text{EP}}(\beta)$  and  $\mathcal{N}_{\text{EP}}(\alpha)$ . Instead, one has  $\mathcal{N}_{\text{EP}}(\beta) = \mathcal{N}_{\text{EP}}^*$ , which scales as  $\langle \mathcal{N}_{\text{EP}}^* \rangle \sim L^{d_{\text{EP}}-x_2}$  in the limit  $Q \rightarrow 0$  as previously discussed. For nonzero small  $Q$  and large  $L$ , we expect that Eq. (8) still holds, thus suppose that  $\langle \mathcal{N}_{\text{EP}}^* \rangle \sim L^2 \sqrt{Q} + aL^{d_{\text{EP}}-x_2}$ . Substituting this result and Eq. (9) into Eq. (16), we get the acceptance probability

$$P_{\text{BR}}^{(-)} = \frac{\mathcal{N}_{\text{BR}}}{\mathcal{N}_{\text{EP}}^*} \times \sqrt{Q} \sim \frac{\sqrt{Q}}{\sqrt{Q} + aL^{d_{\text{EP}}-x_2-2}}. \quad (22)$$

Thus, we expect that the autocorrelation time scales as the inverse of the above acceptance probability:

$$\tau_{\mathcal{N}_{\text{HF}}} / L^2 \sim \frac{\sqrt{Q} + aL^{d_{\text{EP}}-x_2-2}}{\sqrt{Q}}. \quad (23)$$

From the above equation, for a fixed  $Q$ , as  $L$  increases, since  $d_{\text{EP}} - x_2 - 2 < 0$ , one has  $\tau_{\mathcal{N}_{\text{HF}}} / L^2 \rightarrow \mathcal{O}(1)$ , and for a fixed  $L$ , as  $Q$  decreases,  $\tau_{\mathcal{N}_{\text{HF}}} / L^2 \rightarrow Q^{-1/2}$ . These imply the turning near  $Q_c(L)$  and explain the results of the improved Sweeny algorithm, i.e., Fig. 6. Finally, since the acceptance probabilities for operations in the improved Kawasaki algorithm are also 1 for nonzero  $Q$ , we can combine the improved

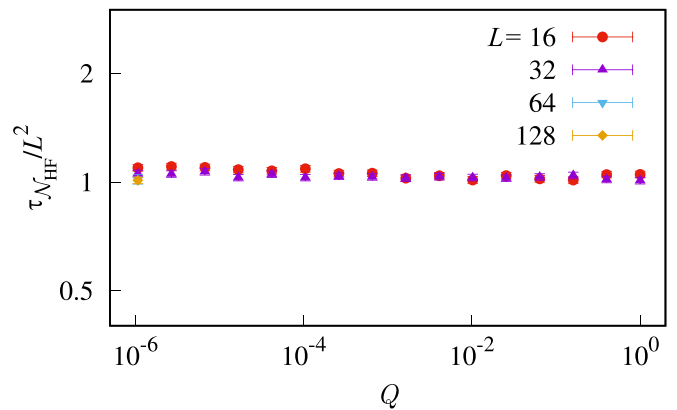


FIG. 8.  $\tau_{\mathcal{N}_{\text{HF}}}$  for the combined method of the improved Sweeny and improved Kawasaki algorithms. The  $L$  dependence of  $\tau_{\mathcal{N}_{\text{HF}}} / L^2$  is very small as demonstrated by data points at  $Q = 10^{-6}$ . Here in our simulation a step consists of a single-bond update in the improved Sweeny algorithm and a two-bond swap in the improved Kawasaki algorithm, which is counted as two hits.

Sweeny algorithm with the improved Kawasaki algorithm. Our simulation results for this combined method are shown in Fig. 8, where  $\tau_{\mathcal{N}_{\text{HF}}}/L^2$  versus  $L$  is almost a straight line. Thus the improved Kawasaki method eliminates the  $\sim Q^{-1/2}$  form slowing-down of the improved Sweeny algorithm at fixed  $L$ , and the combined improved Sweeny-Kawasaki method serves as a complete solution for simulating the random-cluster model with small  $Q$ .

## V. DISCUSSION AND CONCLUSION

To summarize, by introducing the number of half faces, we observe a  $Q$ -dependent slowing-down in two dimensions, which further enriches the dynamics of the Sweeny algorithm for the random-cluster model. The slowing-down is understood by observing the updating probabilities, which contains the critical bond weight  $v_c = \sqrt{Q}$  on the square lattice. By classifying the bonds into four types and dynamically updating their type information, we propose an improved Sweeny algorithm, which (at fixed  $L$ ) pushes the slowing-down to the much smaller  $Q$  region. Since the  $Q$ -dependent slowing-down renders the (improved) Sweeny algorithm non-ergodic for  $Q \rightarrow 0$ , we resort to the Kawasaki algorithm. The latter algorithm is found to exhibit an  $L$ -dependent slowing-down, and this slowing-down can be fully suppressed in an improved version of the algorithm by making use of the bond-type classification. Finally, we formulate an improved Sweeny-Kawasaki method which combines advantages of the two improved algorithms and eliminates the above critical slowing-down.

We note that updates in the improved algorithms proceed with the aid of Baxter-Kelland-Wu loops, as it is convenient to classify the bonds using the loops, and it is also faster to perform connectivity checking in the loops than in the clusters. The computational cost of the algorithms primarily stems from the connectivity checking: simultaneous breadth-first search [21,23] in the clusters has a runtime scaling of  $\sim L^{d_{\text{F}} - \chi_2}$ , while that in the loops has a runtime scaling of  $\sim L^{d_{\text{H}} - \chi_2}$  with  $d_{\text{H}} \leq d_{\text{F}}$ . For not too large  $L$ , the improved Sweeny-Kawasaki method in this paper should be adequate for simulating the random-cluster model. To speed up the simulations for large  $L$ , it should be worthwhile to study how to incorporate dynamic connectivity algorithms which have runtime scaling at most logarithmic in the system size [21,22]. It is also noted that, solely for the uniform spanning trees, there exist algorithms which have excellent performance. These include the celebrated Wilson's algorithm [29], which uniformly constructs a spanning tree from scratch by a loop-erased random walk, and a smart Kawasaki algorithm which uniformly generates spanning trees by using a fast link-cut tree data structure [30]. The latter is different from our improved Kawasaki algorithm, which constructs loops to classify the bonds and applies also to models with  $Q \neq 0$ . Furthermore, though our simulations were performed at  $v_c = \sqrt{Q}$  on the square lattice, for  $v = Q^\sigma$  with general values of  $0 < \sigma < 1$ , as  $Q \rightarrow 0$  there is also  $Q$ -dependent slowing-down for the Sweeny algorithm. This is because in the latter case the updating probabilities tend to zero as  $Q^\sigma$  or  $Q^{1-\sigma}$ . The improved Sweeny and Kawasaki algorithms can also solve this  $Q$ -dependent slowing-down, due to reasons

similar to those for  $v = \sqrt{Q}$ . Also, the  $Q^{-1/2}$ -form critical slowing-down for the random-cluster model is expected to exist on other two-dimensional lattices. For example, on the triangular ( $\mathbb{T}$ ) and honeycomb ( $\mathbb{H}$ ) lattices, when  $Q \leq 4$ , it is known [1,4,31] that

$$v_c^{\mathbb{T}} = 2 \cos \left[ \frac{2}{3} \cos^{-1} \left( \frac{\sqrt{Q}}{2} \right) \right] - 1, \quad (24)$$

$$v_c^{\mathbb{H}} = 2\sqrt{Q} \cos \left[ \frac{1}{3} \cos^{-1} \left( \frac{\sqrt{Q}}{2} \right) \right], \quad (25)$$

$$v_c^{\mathbb{T}} \cdot v_c^{\mathbb{H}} = Q. \quad (26)$$

These critical bond weights  $v_c^{\mathbb{T}}$  and  $v_c^{\mathbb{H}}$  also vanish as  $\sim \sqrt{Q}$  when  $Q \rightarrow 0$ . However, when the spatial dimension  $d \geq 3$ , for small  $Q$  the critical point behaves as  $v_c \sim Q$  (hence the limit  $Q \rightarrow 0$  corresponds to the spanning-forest model) [32]. Thus, the  $Q$ -dependent critical slowing-down does not occur for  $d \geq 3$ . Nevertheless, the improved Sweeny and Kawasaki algorithms may still help to improve the simulation efficiency. Finally, the number of half faces proves to be very useful in demonstrating the slowing-down of the unimproved algorithms. Further work may explore more quantities which have similar behavior. Despite the mechanism discussed in this paper, we cannot rule out any other property which may cause more severe slowing-down.

## ACKNOWLEDGMENTS

This work has been supported by the National Natural Science Foundation of China (under Grant No. 12275263 and No. 12375026), the Innovation Program for Quantum Science and Technology (under Grant No. 2021ZD0301900), and the Natural Science Foundation of Fujian Province of China (under Grant No. 2023J02032). We thank David B. Wilson for his early contributions on this work.

## APPENDIX: DENSITY OF BONDS IN THE THERMODYNAMIC LIMIT

In this Appendix, we give an alternative derivation of the results in Eq. (10). On the square lattice in the thermodynamic limit  $L \rightarrow \infty$ , the self-duality condition leads to the following relation between the critical numbers of bonds:

$$\langle \mathcal{N}_{\text{BB}} + \mathcal{N}_{\text{BR}} \rangle / 2L^2 = \langle \mathcal{N}_{\text{IP}} + \mathcal{N}_{\text{EP}} \rangle / 2L^2 = 1/2. \quad (\text{A1})$$

In the Fortuin-Kasteleyn transformation, the density of occupied bonds can be related to the nearest-neighbor connectivity  $g_{\text{n}}$  (i.e., the probability that two nearest-neighbor sites belong to the same cluster) as [33]

$$\langle \mathcal{N}_{\text{BB}} + \mathcal{N}_{\text{BR}} \rangle / 2L^2 = p_c [g_{\text{n}} + (1 - g_{\text{n}})/Q], \quad (\text{A2})$$

where  $p_c = \sqrt{Q}/(1 + \sqrt{Q})$ . Substituting Eq. (A1) into Eq. (A2), one gets [33]

$$g_{\text{n}} = \frac{2 + \sqrt{Q}}{2(1 + \sqrt{Q})}. \quad (\text{A3})$$

Letting  $P_{\text{n}}$  be the probability that two nearest-neighbor sites are connected by a path not including the bond between the



two sites, one can express the probability  $g_n$  as

$$g_n = P_n + (1 - P_n) \frac{v/Q}{1 + v/Q}. \quad (\text{A4})$$

Substituting  $v_c = \sqrt{Q}$  and Eq. (A3) into Eq. (A4) leads to  $P_n = 1/2$ . Finally, one can express the critical densities of four kinds of bonds as

$$\langle \mathcal{N}_{\text{BB}} \rangle / 2L^2 = P_n \frac{v}{1+v} = \frac{\sqrt{Q}}{2(1+\sqrt{Q})}, \quad (\text{A5})$$

$$\langle \mathcal{N}_{\text{BR}} \rangle / 2L^2 = (1 - P_n) \frac{v/Q}{1+v/Q} = \frac{1}{2(1+\sqrt{Q})}, \quad (\text{A6})$$

$$\langle \mathcal{N}_{\text{IP}} \rangle / 2L^2 = P_n \frac{1}{1+v} = \frac{1}{2(1+\sqrt{Q})}, \quad (\text{A7})$$

$$\langle \mathcal{N}_{\text{EP}} \rangle / 2L^2 = (1 - P_n) \frac{1}{1+v/Q} = \frac{\sqrt{Q}}{2(1+\sqrt{Q})}. \quad (\text{A8})$$

The above densities have been derived in Ref. [24], and they are consistent with  $\langle \mathcal{N}_{\text{BB}} \rangle / 2L^2 = \langle \mathcal{N}_{\text{BR}} \rangle / 2L^2 = 1/4$  for  $Q = 1$  in Ref. [34].

- 
- [1] F. Wu, *Rev. Mod. Phys.* **54**, 235 (1982); **55**, 315 (1983).  
 [2] P. W. Kasteleyn and C. M. Fortuin, *J. Phys. Soc. Jpn.* **26**(Suppl.), 11 (1969).  
 [3] C. M. Fortuin and P. W. Kasteleyn, *Physica* **57**, 536 (1972).  
 [4] G. Grimmett, *The Random-Cluster Model* (Springer, New York, 2006).  
 [5] D. Stauffer and A. Aharony, *Introduction to Percolation Theory*, revised 2nd ed. (Taylor & Francis, London, 1994).  
 [6] J. L. Jacobsen, J. Salas, and A. D. Sokal, *J. Stat. Phys.* **119**, 1153 (2005).  
 [7] M. E. Newman and G. T. Barkema, *Monte Carlo Methods in Statistical Physics* (Clarendon Press, Oxford, 1999).  
 [8] D. Landau and K. Binder, *A Guide to Monte Carlo Simulations in Statistical Physics*, 2nd ed. (Cambridge University, New York, 2005).  
 [9] R. H. Swendsen and J.-S. Wang, *Phys. Rev. Lett.* **58**, 86 (1987).  
 [10] L. Chayes and J. Machta, *Physica A* **254**, 477 (1998).  
 [11] M. Sweeny, *Phys. Rev. B* **27**, 4445 (1983).  
 [12] A. D. Sokal, in *Functional Integration: Basics and Applications*, edited by C. Witt-Morette, P. Cartier, and A. Follacci (Plenum, New York, 1997), pp. 131–192.  
 [13] P. C. Hohenberg and B. I. Halperin, *Rev. Mod. Phys.* **49**, 435 (1977).  
 [14] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, *J. Chem. Phys.* **21**, 1087 (1953).  
 [15] W. K. Hastings, *Biometrika* **57**, 97 (1970).  
 [16] R. J. Glauber, *J. Math. Phys.* **4**, 294 (1963).  
 [17] N. Prokof'ev and B. Svistunov, *Phys. Rev. Lett.* **87**, 160601 (2001).  
 [18] Y. Deng, T. M. Garoni, and A. D. Sokal, *Phys. Rev. Lett.* **98**, 230602 (2007).  
 [19] X.-J. Li and A. D. Sokal, *Phys. Rev. Lett.* **63**, 827 (1989).  
 [20] K. Kawasaki, *Phys. Rev.* **145**, 224 (1966).  
 [21] E. M. Elçi and M. Weigel, *Phys. Rev. E* **88**, 033303 (2013).  
 [22] E. M. Elçi, Algorithmic and geometric aspects of the random-cluster model, Ph.D. thesis, Coventry University, 2015.  
 [23] Y. Deng, W. Zhang, T. M. Garoni, A. D. Sokal, and A. Sportiello, *Phys. Rev. E* **81**, 020102(R) (2010).  
 [24] E. M. Elçi, M. Weigel, and N. G. Fytas, *Nucl. Phys. B* **903**, 19 (2016).  
 [25] R. J. Baxter, S. B. Kelland, and F. Y. Wu, *J. Phys. A* **9**, 397 (1976).  
 [26] H. Saleur and B. Duplantier, *Phys. Rev. Lett.* **58**, 2325 (1987).  
 [27] B. Nienhuis, *J. Stat. Phys.* **34**, 731 (1984).  
 [28] J. Cardy, *J. Phys. A* **31**, L105 (1998).  
 [29] D. B. Wilson, in *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing, STOC '96* (ACM Press, New York, NY, 1996), pp. 296–303.  
 [30] L. Russo, A. Teixeira, and A. Francisco, *Algorithms* **11**, 53 (2018).  
 [31] R. J. Baxter, H. Temperley, and S. E. Ashley, *Proc. R. Soc. A* **358**, 535 (1978).  
 [32] Y. Deng, T. M. Garoni, and A. D. Sokal, *Phys. Rev. Lett.* **98**, 030602 (2007).  
 [33] H. Hu, H. W. J. Blöte, R. M. Ziff, and Y. Deng, *Phys. Rev. E* **90**, 042106 (2014).  
 [34] X. Xu, J. Wang, Z. Zhou, T. M. Garoni, and Y. Deng, *Phys. Rev. E* **89**, 012120 (2014).