Monte Carlo simulation and linear stability analysis of Turing pattern formation in reaction-subdiffusion systems

J. W. Chiu* and K.-H. Chiam†

*A*STAR Institute of High Performance Computing, 1 Fusionopolis Way, #16-16 Connexis, Singapore 138632, Singapore*

(Received 4 September 2008; published 24 November 2008; publisher error corrected 26 November 2008)

Subdiffusion is an important physical phenomenon observed in many systems. However, numerical techniques to study it, especially when coupled to reactions, are lacking. In this paper, we develop an efficient Monte Carlo algorithm based on the Gillespie algorithm and the continuous-time random walk to simulate reaction-subdiffusion systems. Using this algorithm, we investigate Turing pattern formation in the Schnakenberg model with subdiffusion. First, we show that, as the system becomes more subdiffusive, the homogeneous state becomes more difficult to destablize and Turing patterns form less easily. Second, we show that, as the number of particles in the system decreases, the magnitude of fluctuations increases and again the Turing patterns form less easily. Third, we show that, as the system becomes more subdiffusive, the ratio between the two diffusive constants must be higher in order to observe Turing patterns. Finally, we also carry out linear stability analysis to validate the results obtained from our algorithm.

DOI: 10.1103/PhysRevE.78.056708 PACS number(s): 02.70.Uu, 05.40.−a, 47.54.−r, 82.40.Ck

I. INTRODUCTION

Since Turing discovered that the homogeneous steady state of a chemical reaction can be destabilized by diffusion [1], many studies have been carried out on reaction-diffusion models which exhibit patterns. Such models have applications in a number of diverse fields, such as biology [2,3], physics [4,5], and optics [6].

However, experiments have shown that, in many cases, the diffusion is non-Fickian and the standard reaction-diffusion model cannot be used. What is observed is that the mean square displacement of the particles $\langle x^2 \rangle$ is proportional to time t^α, where $0 < \alpha < 1$ is the diffusion exponent. This is known as subdiffusion. For example, using fluorescence correlation spectroscopy, Weiss *et al.* [7] showed that the motion of macromolecules in the cytoplasm is subdiffusive, with $\alpha \approx 0.526$. Other examples include charge carrier transport in amorphous semiconductors [8], solute transport in porous systems [9–12], etc. Also, space-fractional reaction-diffusion equations have been considered in Ref. [13] with applications to population biology [14]. Therefore, to study pattern formation in these systems, it is important to consider reaction-subdiffusion systems.

The canonical model for explaining subdiffusion is the continuous-time random walk (CTRW) model [15–17]. In this model, every particle makes a jump of distance x after a waiting time of τ according to a probability distribution $W(x,\tau)$. The latter is often decoupled as $\varphi(x)f(\tau)$ with $\varphi(x)$ being a normal distribution and $f(\tau)$ being a heavy-tailed distribution such that its Laplace tranform $\hat{f}(s) \sim 1-(\eta s)^\alpha$. It can then be shown that the long-time limit of the CTRW is governed by [18]

$$\frac{\partial N}{\partial t} = K\left({}_0D_t^{1-\alpha} \frac{\partial^2 N(x,t)}{\partial x^2} \right), \tag{1}$$

where N is the particle number density, K is the effective diffusion constant, and ${}_0D_t^{1-\alpha}$ is the Riemann-Liouville operator [19,20].

As a side note, Eq. (1) is also equivalent to the more familiar time-fractional form,

$$\frac{\partial^\alpha N}{\partial t^\alpha} = K \frac{\partial^2 N}{\partial x^2} \tag{2}$$

where the operator on the left-hand side of the equation is the Caputo fractional derivative. This form more closely resembles the usual diffusion equation. The use of the Caputo fractional derivative allows the initial condition at $t=0$ to be handled in a more natural way, similar to integer order derivatives in time. The reader may refer to Ref. [21] on the role of such fractional kinetic equations in modeling Hamiltonian chaos.

When the particles are also reacting with one another [22,23], it is not clear how the CTRW should be extended. Many proceed directly to writing down differential equations for reaction-subdiffusion systems. For example, for a system with two reacting species, it is tempting to write, by comparing with standard reaction-diffusion equations,

$$\frac{\partial N_i}{\partial t} = K_i\left({}_0D_t^{1-\alpha_i} \frac{\partial^2 N_i(x,t)}{\partial x^2} \right) + F_i(N_1,N_2), \tag{3}$$

for $i=1,2$. Here, N_1 and N_2 are particle number densities for the two species, F_1 and F_2 are functions specifying the reaction kinetics, K_1 and K_2 are the diffusion constants, and α_1 and α_2 are the diffusion exponents.

Unfortunately, these equations do not work for even the simplest system: $X_1 \to X_2$ [24,25]. The contributions of subdiffusion and reactions simply cannot be separated as in Eq. (3). The correct equations governing reaction-subdiffusion are derived in Ref. [26], but they seem too difficult to be evaluated numerically.

*Present address: Department of Mathematics, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA.

†ChiamKH@MailAPS.ORG

In Ref. [27], Weiss designed a hybrid algorithm to simulate reaction-subdiffusion and to study pattern formation for the Schnakenberg model [28]. His method divides space into bins, and simultaneously performs the CTRW and solves the differential equations numerically for each bin. This method is then employed to study Turing pattern formation in systems with noise induced by low particle numbers, an important problem because biological systems typically involve only low number of particles. However, by evolving the particle counts in each bin numerically, the above algorithm ignores the effect of noise in individual bins, which has even fewer particles.

Motivated by the lack of a stochastic algorithm to simulate reaction-subdiffusion systems and to study the effects of noise on such systems, we combine the Gillespie algorithm [29] and the CTRW, and arrive at an algorithm that can simulate reaction-subdiffusion systems. In Sec. III, we apply our algorithm to study the Schnakenberg model with subdiffusion. In particular, we are interested in Turing pattern formation in this system, and investigate how it could occur as parameters such as the diffusion exponent, total number of particles in the system, and the ratio of the diffusion constants of the reacting species are varied. Finally, we perform linear stability analysis for this model to validate our algorithm. Our analysis will be based on the mean field solution established in Ref. [26] and is similar to Ref. [30].

II. MATHEMATICAL MODEL AND A MONTE CARLO ALGORITHM

A. The Gillespie algorithm

We will begin with a review of the Gillespie algorithm [29]. The setup is as follows. Let A be a set of reactions, namely, $A=\{R_1,R_2,\ldots\}$ where each R_i is a reaction. Suppose these reactions involve S different species of reactants $X_1,\ldots,X_S$. Say the rate constant of R_i is c_i and the number of X_i particles present is N_i. Given a state $(N_1,\ldots,N_S)$ at time T, we can define the joint probability distribution $P_A(t,i)$ such that

$$P_A(t,i)dt = \Pr[\text{reaction } R_i \text{ takes place next among reactions in } A \text{ in the time interval } (T+t,T+t+dt)]. \tag{4}$$

If P_A is known, we can compute the marginal distribution $p_A(i)=\int_0^\infty P_A(t,i)dt$, and use it to identify the next reaction to happen. If some reaction R_a is selected, we can then sample t from the conditional distribution $P_A(t|a)=P_A(t,a)/p_A(a)$. This basic framework of the Gillespie algorithm is shown in Table I.

TABLE I. The basic framework of the Gillespie algorithm.

```
function GillespieBasicFramework ($A$)
  1. Let $T=0$.
  2. While $T<T_{\max}$,
      (a) Compute $p_A(i)=\int_0^\infty P_A(t,i)dt$ for each $R_i$ in $A$.
      (b) Pick reaction $R_a$ to happen next with probability $p_A(a)$.
      (c) Carry out reaction $R_a$ and update the $N_i$'s.
      (d) Sample $t$ from the conditional distribution
          $P_A(t|a)=P_A(t,a)/p_A(a)$.
      (e) Increment $T$ by $t$.
```

It remains to determine P_A in terms of the rate constants c_i and the state $(N_1,\ldots,N_S)$, so that the algorithm in Table I evolves the system according to the rate equations. For this purpose, Gillespie introduced the quantity h_i as the number of distinct R_i reacting combinations given the state $(N_1,\ldots,N_S)$. For example, if R_1 is "$2X_1+3X_2\to X_3$," then $h_1=\binom{N_1}{2}\binom{N_2}{3}$. Further, assuming that $c_i dt$ is the average probability that a reacting combination of R_i reacts in a time interval dt, Gillespie showed that

$$P_A(t,i) = \alpha_i e^{-\alpha_0 t} \tag{5}$$

where $\alpha_i=h_i c_i$ is known as the propensity of reaction R_i, and $\alpha_0=\Sigma_{R_i\in A}\alpha_i$. From Eq. (5), we obtain $p_A(i)=\alpha_i/\alpha_0$ and $P_A(t|a)=\alpha_0 e^{-\alpha_0 t}$, and translate Table I into the Gillespie algorithm in Table II.

To incorporate diffusion, the usual practice is to divide the whole space into bins or reaction chambers, and add reactions to allow particles to jump between adjacent bins [31]. For convenience, we call these additional reactions "diffusive reactions" and the reactions within each bin "nondiffusive reactions." Performing the Gillespie algorithm on such an expanded system of reactions has worked well for many reaction-diffusion models [32–35].

However, when particles undergo subdiffusion instead, it is not clear what the rate constants of diffusive reactions should be. They cannot be fixed with respect to time; otherwise standard diffusion would be reproduced. Yet, deciding how these rate constants should vary with time is not trivial. One underlying problem is that if we assume the subdiffusion to be accounted by the CTRW, then every subdiffusing particle has a nonexponential waiting time distribution [15] which is not memoryless. Each particle thus possesses memory and behaves uniquely, but the Gillespie algorithm does not distinguish between particles of the same species in the same bin. Another way to see this is to examine Eq. (1). The equation contains the Riemann-Liouville operator [19,20], and implies that the evolution of the system depends

TABLE II. The Gillespie algorithm.

```
function Gillespie ($A$)
  1. Let $T=0$.
  2. While $T<T_{\max}$,
      (a) Compute $\alpha_i=h_i c_i$ for each $R_i$ in $A$.
      (b) Let $\alpha_0=\Sigma_{R_i\in A}\alpha_i$.
      (c) Pick reaction $R_a$ to happen next with probability
          $\alpha_a/\alpha_0$.
      (d) Carry out reaction $R_a$ and update the $N_i$'s and $h_i$'s.
      (e) Sample $t$ from the distribution $\alpha_0 e^{-\alpha_0 t}$. (Specifically,
          let$t=\alpha_0^{-1}\log(1/r)$, where $r$ is sampled uniformly from
          $[0,1]$.)
      (f) Increment $T$ by $t$.
```

TABLE III. A direct method for simulating the basic model in Sec. II B. Note that it resembles the continuous-time random walk.

```
function GillespieCTRW ($A$,$A'$)
  1. Let $T=0$.
  2. Sample $\tau_{i,j}$ from $f_i(\tau)$ for every $R_{i,j}$ in $A'$. Store $T_{i,j}=0+\tau_{i,j}$.
  3. While $T<T_{\max}$,
      (a) Select among $A'$ the reacting combination $R_{a,b}$ with the minimum $T_{a,b}$.
      (b) Set $T$ to $T_{a,b}$.
      (c) Execute the reaction of combination $R_{a,b}$. If a new reacting combination
          $R_{i,j}$ is born, sample $\tau_{i,j}$ and store $T_{i,j}$ as $T+\tau_{i,j}$.
```

on its previous states. However, the Gillespie algorithm does not have any mechanism to remember the system's previous states. Our aim is therefore to modify the Gillespie algorithm so that it can support such a system with reasonable extra computational cost.

B. Extending Gillespie algorithm to continuous-time random walk

Before modifying the Gillespie algorithm, we present the basic mathematical model that it rests on. In this model, we assign every reacting combination a random waiting time. Let A be a set of reactions and A' be its reacting combinations. Denote a reacting combination of reaction R_i as $R_{i,j}$ where $j=1,\ldots,h_i$. Let $\tau_{i,j}$ be its waiting time, sampled from some probability distribution $f_i(\tau)$. If all the reactions are "diffusive reactions," then this model is essentially the CTRW. The most direct method to simulate such a model is to store the absolute time a reacting combination $R_{i,j}$ is going to happen as $T_{i,j}$, and repeatedly find the minimum $T_{i,j}$ among all the reacting combinations. This is illustrated in Table III.

For selecting the reacting combination with the minimum $\tau_{i,j}$, a heap data structure can be used [36]. However, due to combinatorial explosion, the algorithm in Table III remains infeasible. Here is an alternative approach. First, determine the distribution $P_A(t,i,j)$ defined such that

$$P_A(t,i,j)dt = \Pr[\text{reacting combination } R_{i,j} \text{ takes place next among reacting combinations in } A' \text{ in the time interval } (T+t,T+t+dt)]. \qquad (6)$$

This resembles Eq. (4). Second, define $p_A(i,j) = \int_0^\infty P_A(t,i,j)dt$ as the probability that reacting combination $R_{i,j}$ happens next. Third, modify the basic framework of the Gillespie algorithm in Table I such that, instead of choosing the next reaction R_a, choose the next *reacting combination* $R_{a,b}$ with probability $p_A(a,b)$. This is shown in Table IV. The question is, what is $P_A(t,i,j)$ given the waiting time distributions $f_i(\tau)$?

At any point in time, we let $t_{i,j}$ be the time that $R_{i,j}$ has already waited. Note that $0 \leqslant t_{i,j} < \tau_{i,j}$. For some $R_{a,b}$ to take place next after a further wait of t, i.e. $\tau_{a,b}$ is in the time interval $(t_{a,b}+t, t_{a,b}+t+dt)$, it must be that all other reacting combinations $R_{i,j}$ has $\tau_{i,j} > t_{i,j}+t$. By conditional probability, we can write

TABLE IV. Generalization of the basic framework of Gillespie algorithm in Table I.

```
function GillespieExtendedFramework ($A$,$A'$)
  1. Let $T=0$.
  2. While $T<T_{\max}$,
      (a) Compute $p_A(i,j)=\int_0^\infty P_A(t,i,j)dt$ for each $R_{i,j}$ in $A'$.
      (b) Pick reacting combination $R_{a,b}$ to happen next with probability $p_A(a,b)$.
      (c) Execute the reaction of combination $R_{a,b}$.
      (d) Sample $t$ from the conditional distribution
          $P_A(t|a,b)=P_A(t,a,b)/p_A(a,b)$.
      (e) Increment $T$ by $t$.
```

$$P_A(t,a,b)dt = \left(\prod_{\substack{R_{i,j}\in A' \\ (i,j)\neq(a,b)}} \Pr(\tau_{i,j} > t_{i,j}+t | \tau_{i,j} \rangle t_{i,j}) \right) \times \Pr(t_{a,b}+t < \tau_{a,b} < t_{a,b}+t+dt | \tau_{a,b} \rangle t_{a,b}). \qquad (7)$$

Define the upper tail probability of $f_i(\tau)$ as $g_i(\tau) = \int_{\tau'=\tau}^{\infty} f_i(\tau')d\tau'$, and rewrite Eq. (7) as

$$P_A(t,a,b)dt = \left(\prod_{\substack{R_{i,j}\in A' \\ (i,j)\neq(a,b)}} \frac{g_i(t_{i,j}+t)}{g_i(t_{i,j})} \right) \left(\frac{f_a(t_{a,b}+t)dt}{g_a(t_{a,b})} \right). \qquad (8)$$

Let $e_i(\tau) = f_i(\tau)/g_i(\tau)$ and simplify Eq. (9) into

$$P_A(t,a,b)dt = \left(\prod_{R_{i,j}\in A'} \frac{g_i(t_{i,j}+t)}{g_i(t_{i,j})} \right) e_a(t_{a,b}+t). \qquad (9)$$

If the f_i of all reactions in A are exponential distributions, i.e. of the form $c_i e^{-c_i \tau}$, then the memoryless property leads to very simple g_i's and e_i's: $g_i(\tau)=e^{-c_i\tau}$ and $e_i(\tau)=c_i$. Plugging these into Eq. (9), we obtain

$$P_A(t,a,b) = \left(\prod_{R_{i,j}\in A'} e^{-c_i t} \right) c_a = c_a e^{-\alpha_0 t}, \qquad (10)$$

where $\alpha_0 = \sum_{i=1}^{M} \alpha_i$ and $\alpha_i = h_i c_i$ are the propensities seen earlier in Sec. II A. Notice that $p_A(a,b) = \int_0^\infty P_A(t,a,b)dt = c_a/\alpha_0$. This is independent of b. Thus, for this special case, there is no need to distinguish between reacting combinations of the same reaction type and the method of selecting the next reacting combination in Table IV can be reduced to selecting the next reaction R_a with probability $p_A(a) = \sum_b p_A(a,b) = h_a c_a/\alpha_0 = \alpha_a/\alpha_0$. We arrive again at the Gillespie algorithm in Table II.

However, for reaction-subdiffusion systems, the f_i's for diffusive reactions cannot be exponential distributions. Otherwise, we will obtain standard diffusion according to CTRW theory [15]. For this more general case, the algorithm in Table IV is not feasible because integrating Eq. (9) to

TABLE V. Our algorithm for simulating the mathematical model in Sec. II B. Note that A_1 is the set of reacting combinations with f_i's being exponential distributions, and represents nondiffusive reactions, while A_2 is A_1's complement and represents diffusive reactions.

```
function GillespieReactionSubdiffusion $(A_1, A_2)$
  1. Let $T=0$.
  2. While $T<T_{\max}$,
       (a) For each $R_i$ in $A_1$, compute $\alpha_i=h_ic_i$.
       (b) Let $\alpha_0=\Sigma_{R_i \in A_1}\alpha_i$.
       (c) Sample the time delay $t_1$ from the distribution $\alpha_0 e^{-\alpha_0 t}$.
       (d) Let $T_1=T+t_1$.
       (e) Let $T_{c,d}$ be the smallest $T_{i,j}$ among reacting combinations of reactions in
             $A_2$. Let $T_2=T_{c,d}$.
       (f) Set $T$ to $\min(T_1, T_2)$.
       (g) If $T_1<T_2$, pick $R_a$ from $A_1$ with probability $\alpha_a/\alpha_0$ and execute it.
       (h) Otherwise, execute the reaction of combination $R_{c,d}$.
```

obtain $p_A(i,j)$ for *every* reacting combination seems computationally intractable.

C. Our Monte Carlo algorithm

So far, we have two candidates for simulating the basic mathematical model in Sec. II B. They are shown in Table III and Fig. 4. Unfortunately, neither is feasible due to combinatorial explosion. Our approach is to divide the set of reactions A into two sets A_1 and A_2, where A_1 is the set of reactions with f_i's being exponential distributions, and $A_2=A-A_1$. For A_1, which corresponds to nondiffusive reactions, we will apply the algorithm in Table IV. Since the f_i's for these reactions are exponential distributions, there is no need to distinguish between reacting combinations of the same reaction, and the algorithm in Table IV reduces to the dramatically faster Gillespie algorithm in Table II. For A_2, which corresponds to diffusive reactions, we will apply the algorithm in Table III. Since each of these diffusive reactions involves only one species, the number of reacting combinations for A_2 increases only linearly with the total number of particles, and can be handled well by the algorithm in Table III. Each algorithm will report their earliest reacting combination and we will pick the earlier one. This approach is outlined in Table V.

The correctness of this approach can be proven by considering a more general set-up. Suppose the set of reactions A is partitioned into L subsets $A_1, \ldots, A_L$. Let A_k' denote the reacting combinations of reactions in subset A_k. Suppose for each A_k, there is some algorithm $\mathcal{A}_k$ which can single out the earliest reacting combination in A_k'. To be precise, we mean that $\mathcal{A}_k$ selects reacting combination $R_{i,j}$ in A_i' to happen next at time interval $(T+t, T+t+dt)$ with probability $P_{A_k}(t,i,j)dt$, as in Eq. (9). Say each $\mathcal{A}_k$ reports an absolute time T_k for their "earliest" reacting combination, and we pick the reacting combination with the smallest T_k.

Obviously, for some $R_{a,b}$ in A_c to be selected and executed, it must be that $T_k>T_c$ for all $k\neq c$. Let $t_c=T_c-T$. By the assumption about $\mathcal{A}_k$ in the previous paragraph, this probability is $\Sigma_{R_{i,j}\in A_k}\int_{t_c}^{\infty}P_{A_k}(t,i,j)dt$.

Substitute in Eq. (8), bring the summation into the integrand and apply the product rule to obtain

$$\int_{t_c}^{\infty}\frac{d}{dt}\left(\prod_{R_{i,j}\in A_k}\frac{g_i(t_{i,j}+t)}{g_i(t_{i,j})}\right)=\prod_{R_{i,j}\in A_k}\frac{g_i(t_{i,j}+t_c)}{g_i(t_{i,j})}. \qquad (11)$$

This approach therefore chooses the combination $R_{a,b}$ from subset A_c to react in time interval $(T+t_c, T+t_c+dt)$ with probability $[P_{A_c}(t_c,a,b)dt]\prod_{k\neq c}\Pi_{R_{i,j}\in A_k}g_i(t_{i,j}+t_c)/g_i(t_{i,j})$.

Refer to Eq. (9) and recognize that the above is exactly $P_A(t_c,a,b)$. Hence, the algorithm in Table V, which combines two correct algorithms for subsets A_1, A_2, works. Note that for subset A_1, we can determine T_1 before deciding which reaction in A_1 is the earliest. This saves time whenever $T_1>T_2$.

III. SIMULATION OF THE SCHNAKENBERG MODEL AND RESULTS

A. Simulation set-up

In this section, we apply our algorithm to simulate the Schnakenberg model with subdiffusion. We want to investigate the conditions for Turing pattern formation when the particles undergo subdiffusion instead of normal diffusion. The Schnakenberg model is chosen as a simple and prototypical model for Turing pattern formation. It comprises only four reactions: $A\rightleftharpoons U$, $B\rightarrow V$, and $2U+V\rightarrow 3U$. Assume space to be one dimensional, $x\in[0,1]$ with periodic boundaries. Let $u(x)$ and $v(x)$ be the number density of particles of U and V respectively, at point x. In the absence of diffusion, the law of mass action dictates that

$$\frac{\partial u}{\partial t}=aM+\frac{vu^2}{M^2}-u$$

$$\frac{\partial v}{\partial t}=bM-\frac{vu^2}{M^2}, \qquad (12)$$

where a, b, and M are positive constants, i.e., we have assumed that the densities of species A and B are very much higher than those of species U and V and can therefore be treated as constants. In our simulations, we fix $a=0.1$ and $b=0.9$. Observe that, at steady state, we expect $u=M$ and $v=bM$. Since the space has unit length, we expect the system to have a total of $(b+1)M$ particles. The parameter M therefore specifies the number of particles in our simulations. We are also interested to study Turing pattern formation as a function of the number of particles in the system, i.e., to study the effects of fluctuations induced by small numbers of particles.

Next, subdivide the space evenly into N bins. Let x_i be the center of bin i and u_i, v_i be the number of particles of U, V in bin i, respectively. Approximately, $u_i=u(x_i)/N$ and $v_i=v(x_i)/N$. Substitute these into Eq. (12) to obtain

$$\frac{\partial u_i}{\partial t} = a\left(\frac{M}{N}\right) + v_i u_i^2\left(\frac{N}{M}\right)^2 - u_i,$$

$$\frac{\partial v_i}{\partial t} = b\left(\frac{M}{N}\right) - v_i u_i^2\left(\frac{N}{M}\right)^2. \qquad (13)$$

From the above equations, we can read off the propensities to use for nondiffusive reactions within each bin. For example, the propensity for $A \rightarrow U$ is $a(M/N)$ while the propensity for $2U+V \rightarrow 3U$ is $v_i u_i(u_i-1)(N/M)^2$.

Now, to incorporate subdiffusion, we let $u_i \rightarrow u_{i\pm 1}$ (and similarly for the v_i's) and let the f_i's of these diffusive reactions be the Pareto distribution, which is heavy tailed. This distribution has two parameters α, β: $f_{\alpha,\beta}(\tau)=\alpha\beta^{\alpha}\tau^{-\alpha-1}$, supported on $[\beta,\infty)$. Its Laplace transform is $\hat{f}_{\alpha,\beta}(s) \sim 1-(\eta s)^{\alpha}$ where $\eta=\beta(-\alpha\Gamma(-\alpha))^{1/\alpha}$ and Γ is the Gamma function.

By CTRW theory, if particles move by distance x after waiting for time τ with probability $\varphi(x)f(\tau)$, where the Fourier transform of $\varphi(x)$ is $\hat{\varphi}(k) \sim 1-\Sigma k^2$ and the Laplace transform of $f(\tau)$ is $\hat{f}(s) \sim 1-(\eta s)^{\alpha}$, then the particles subdiffuse with diffusion exponent α and diffusion constant $K=\Sigma/\eta^{\alpha}$ [15]. In our case, since particles move left or right with equal probability, $\varphi(x)=\frac{1}{2}[\delta(x-1/N)+\delta(x+1/N)]$ where δ is the Dirac delta function, and $\Sigma=1/(2N^2)$.

We focus on the case where the two species of particles have the same diffusion exponents but different diffusion constants. Thus, for the two Pareto waiting time distributions, we use the same α but different β. Specifically, when given α and K_i, let

$$\beta_i = \left(\frac{\Sigma}{-K_i\alpha\Gamma(-\alpha)}\right)^{1/\alpha}. \qquad (14)$$

In addition, we fix K_1 at 10^{-4} and vary only the ratio $D=K_2/K_1$.

To summarize, there are three parameters that we vary in our study of the Schnakenberg model, namely the number of particles in the system M which characterizes the extent of "noisiness" in the system, the diffusion exponent α which characterizes the extent of subdiffusion, and the ratio of diffusion constants D. We use the algorithm developed in Sec. II to study the formation of Turing patterns in the Schnakenberg model with subdiffusion as these parameters are varied.

B. Measuring the strength of patterns

Next, we define a metric to characterize the strength of patterns. Let $u_i(t)$, $v_i(t)$ be the number of particles of U, V in bin i at time t respectively, and $\mathbf{u}(t)$ denote the spatial state $\{u_i(t): i=1,\ldots,N\}$. The metric we use is the short-range correlation

$$\rho(t) = \frac{\sum_{i=1}^{N}[u_i(t)-\mu][u_{i+1}(t)-\mu]}{\sum_{i=1}^{N}[u_i(t)-\mu]^2} \qquad (15)$$

where μ is the mean of $\mathbf{u}(t)$. There is no need to consider $\mathbf{v}(t)=\{v_i(t): i=1,\ldots,N\}$ because it largely mirrors $\mathbf{u}(t)$. The metric differs from what is found in Ref. [27] and has the advantage of being normalized. In our simulations, we run from time $t=0$ to $t=300$, and find that $\rho(t)$ reaches steady state quickly. This is shown in Fig. 1. Hence, we take the average of $\rho(t)$ over $t=250$ to $t=300$ and over 20 runs with randomly chosen initial conditions. This quantity, denoted as ρ^*, is the actual metric used.

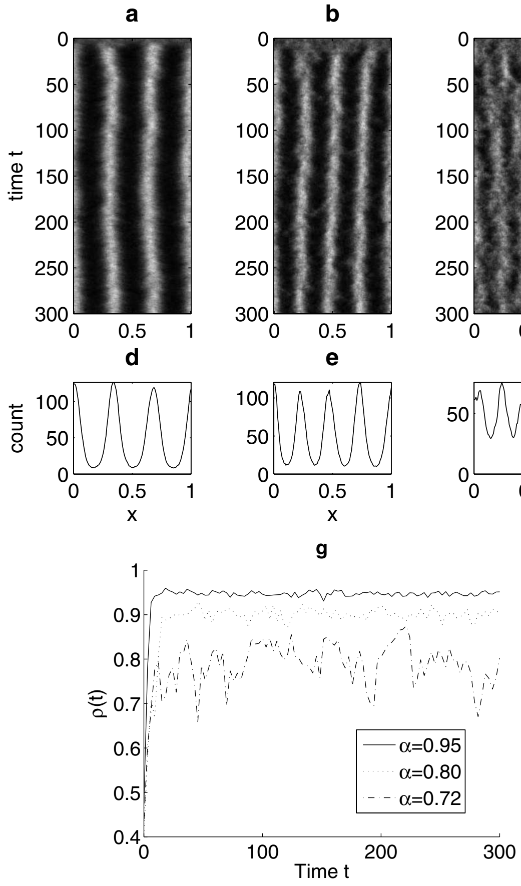

FIG. 1. (a)–(c) Evolution of number of U particles over time. (d)–(f) Variation of the number of U particles over space, averaged over $t=250-300$. (g) $\rho(t)$ [see Eq. (15) on this metric on the strength of patterns] versus time t, for one randomly chosen initial condition. From left to right, the subdiffusion exponent $\alpha=0.95,0.80$, 0.72; other parameters are the ratio between diffusion constants $D=100$, and the total number of U particles at steady state $M=6428$.

In Sec. III D we will explain that the threshold value of ρ^* for which Turing patterns appear will be $\rho^* \approx 0.82$, i.e., we consider Turing patterns to have formed when $\rho^* \gtrsim 0.82$.

C. Turing pattern formation

First, we vary the diffusion exponent α and investigate whether Turing pattern formation occurs when normal diffusion is replaced with subdiffusion. Recall that diffusion destabilizes the homogeneous state and leads to pattern formation. Therefore, when α decreases for both reactants, we expect the diffusion to become weaker and the patterns to diminish and disappear eventually. This is indeed what we found, as shown in Fig. 2. We see that when $\alpha \rightarrow 1$ (i.e.,

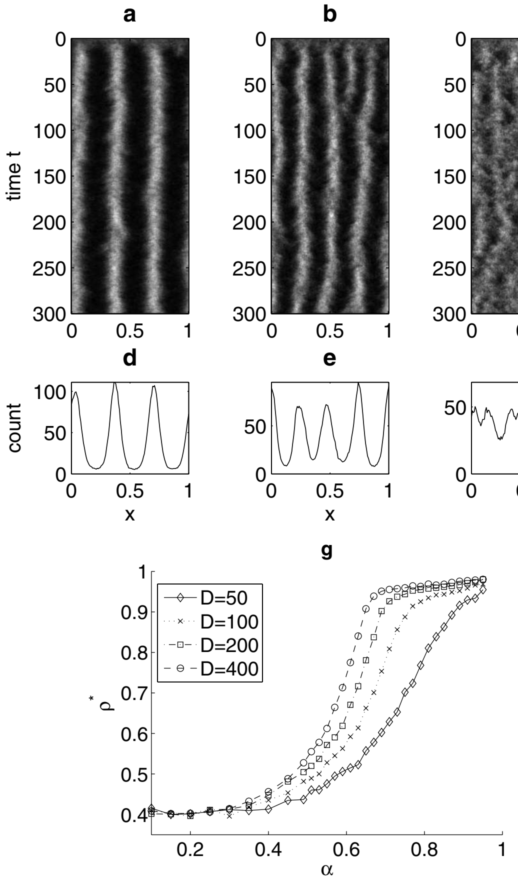

FIG. 2. (a)–(c)Evolution of number of U particles over time. (d)–(f) Variation of number of U particles over space, averaged over t=250–300. (g) ρ^*, the strength of patterns, versus time α, with various D's, the ratio between diffusion constants. From left to right, the diffusion exponent α=0.9,0.8,0.7; the total number of U particles at steady state M=6428.

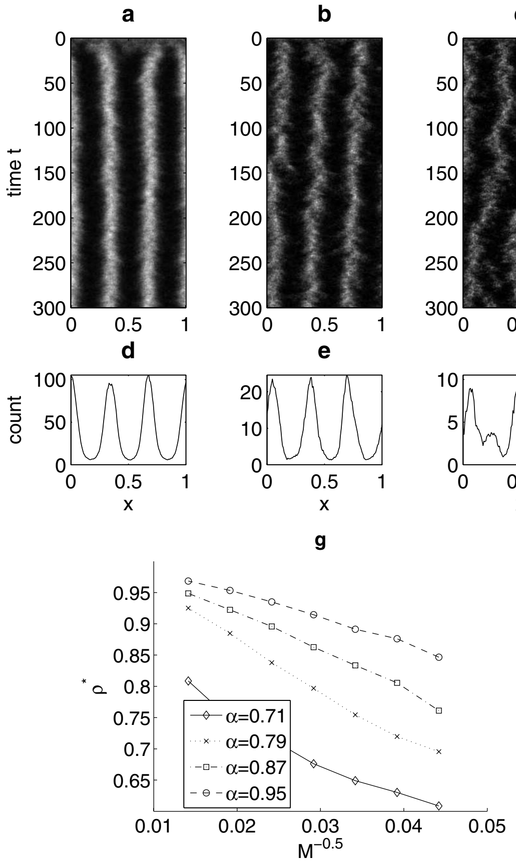

FIG. 3. (a) to (c) show how the number of U particles evolve over time. (d) to (f) show how the number of U particles vary over space, averaged over t=250–300. (g) ρ^*, the strength of patterns, versus $M^{-0.5}$, with various α's, the diffusion exponent. From left to right, M=5000,1175,512; the ratio between diffusion constants D=100.

approaching normal diffusion), the values of D=100, 200, 300, 400 and M=5000 are such that $\rho^* \to 1$, i.e., that patterns do occur. But when α is decreased, we see that ρ^* decreases as well. As will be explained in the next section, a value of $\rho^* \gtrsim 0.82$ denotes pattern formation. Thus, we see that pattern formation occurs in the Schnakenberg model when $\alpha \gtrsim 0.6$.

Second, we vary the total number of particles M and investigate how Turing pattern formation varies. In a system of M particles, we expect that variables that depend on M will have their variance scale like $\sim M^{-1}$. Thus, we expect the parameter $M^{-0.5}$ to control the magnitude of fluctuations in the system, or equivalently, the extent of "noisiness" in the system. In Fig. 3, we see that ρ^* decreases approximately linearly with $M^{-0.5}$, for several values of α and fixing D =100. This result is again intuitive: as noise in the system increases, patterns become weaker and eventually disappear.

These two results can be summarized in a parameter space plot of ρ^* as a function of both α and $M^{-0.5}$, as is shown in Fig. 4. In addition, we can also compute the width in α over which the transition from a homogeneous state to a pattern takes place. This width is denoted in the figure by the dotted lines drawn where ρ^*=0.77,0.82,0.87. Let the width w be the horizontal distance between the leftmost and rightmost lines. In the bottom diagram of Fig. 4, we see that w increases linearly with $M^{-0.5}$, suggesting that noise tends to smear out the transition to pattern formation that, by extrapolation, is sharp in the deterministic (no noise, $M^{-0.5}$=0) regime.

Third, we vary the ratio of diffusion constants $D=K_2/K_1$ and see how Turing pattern formation varies. It has been established that for normal reaction-diffusion systems, Turing patterns form only when $D \gg 1$. For example, when a =0.1 and b=0.9 as we use throughout here, we expect Turing patterns to occur when $D \gtrsim 8.5$ [27]. In Fig. 5, we see that ρ^* decreases as D decreases for several different values of α. In fact, the smaller α is, the larger D needs to be in order for Turing patterns to occur. Figure 6 illustrates how ρ^* varies with both D and α.

D. Linear stability analysis

We will now perform linear stability analysis to verify that the aforementioned results obtained from our simulations are correct. First, we will describe the procedure of the stability analysis for an arbitrary reaction-subdiffusion system, with two species of reactants having the same diffusion

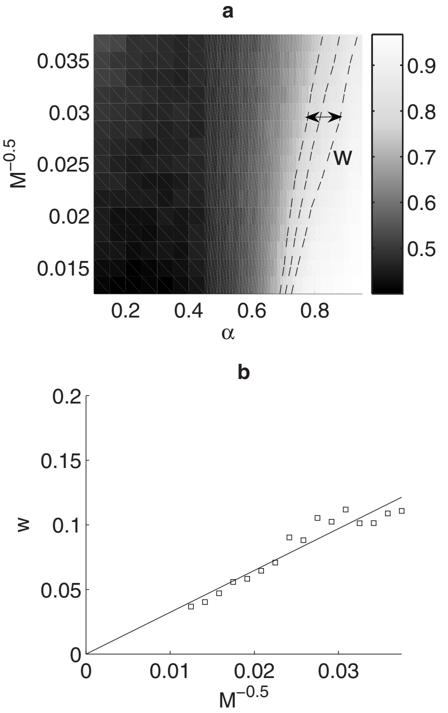

FIG. 4. In (a), the shading denotes the strength of patterns or ρ^*. The ordinate denotes $M^{-0.5}$ while the abscissa denotes α. There are three dotted lines marking where $\rho^*=0.77,0.82,0.87$ (from left to right). w is the horizontal distance between the leftmost and rightmost lines. (b) w versus $M^{-0.5}$ and the line of best fit, which shows that $w \propto M^{-0.5}$. For both diagrams, D is fixed at 100.

exponent. Then, we will apply this procedure to the Schnakenberg model to obtain α_L, the threshold value of α above which Turing patterns are expected to form. Finally, we will compare α_L with the results of the simulations.

Denote $\mathbf{N}(x,t)=\{N_1(x,t),N_2(x,t)\}$ as the number of particles of two species of reactants. The rate equations can be put as $d\mathbf{N}/dt=\mathbf{F}(\mathbf{N})$ where $\mathbf{F}$ is some vector-valued function of $\mathbf{N}$. For our Schnakenberg model,

$$\mathbf{F}(\mathbf{N}) = \begin{pmatrix} aM + N_1^2N_2/M^2 - N_1 \\ bM - N_1^2N_2/M^2 \end{pmatrix}. \tag{16}$$

Let $\mathbf{N}_0$ be the stationary state. Consider perturbing it by some small $\mathbf{n}$. Observe that

$$\frac{d\mathbf{n}}{dt} = \mathbf{A}\mathbf{n} \quad \text{where } A_{i,j} = \frac{\partial F_i}{\partial N_j}. \tag{17}$$

Assume that $\mathbf{A}$ can be diagonalized and expressed as $\mathbf{A} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^{-1}$ for some diagonal matrix $\mathbf{\Lambda}$. Define

$$\mathbf{B}(t) = \exp(\mathbf{A}t) = \mathbf{V}\exp(\mathbf{\Lambda}t)\mathbf{V}^{-1}. \tag{18}$$

The solution for Eq. (17) is

$$\mathbf{n}(t) = \mathbf{B}(t-t')\mathbf{n}(t'). \tag{19}$$

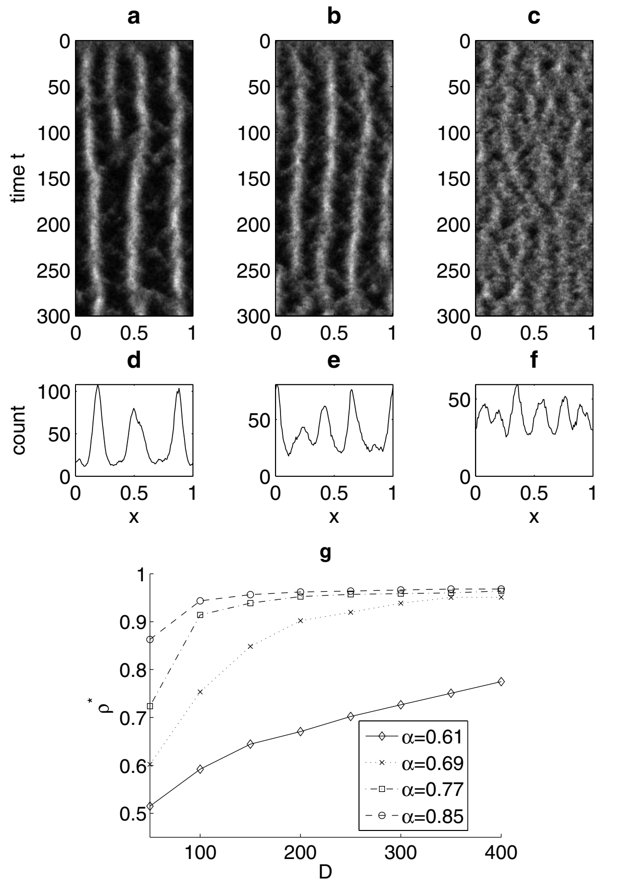

FIG. 5. (a)–(c)Evolution of the number of U particles over time. (d)–(f) Variation of the number of U particles over space, averaged over $t=250-300$. (g) ρ^*, the strength of patterns, versus D, the ratio between diffusion constants, with various α's, the diffusion exponent. From left to right, $D=300,200,100$; the number of U particles at steady state $M=5000$.

This describes how $\mathbf{n}$ changes due to reactions that happen between t' and t. Now, turn off reactions and consider just a CTRW. Define $n_i(x,t)$ as the number of species i particles that just appear at x at time t. Without reactions, we see that

$$n_i(x,t) = \int_{x'}\int_{t'=0}^{t} \varphi(x-x')f_i(t-t')n_i(x',t'). \tag{20}$$

Turn on reactions. Since $\mathbf{n}$ is a small perturbation to $\mathbf{N}_0$, we can modify $n_i(x',t')$ in Eq. (20) according to (19) and obtain

$$n_i(x,t) = \int_{x'}\int_{t'=0}^{t} \varphi(x-x')f_i(t-t')\sum_{j=1}^{n} B(t-t')_{i,j}n_j(x',t'). \tag{21}$$

Define $\mathbf{n}_0(x)$ as $\mathbf{n}(x,0)$ and $\mathbf{f}(t)$ as $\mathrm{diag}(f_1(t),f_2(t))$. Rewrite Eq. (21) in vector form,

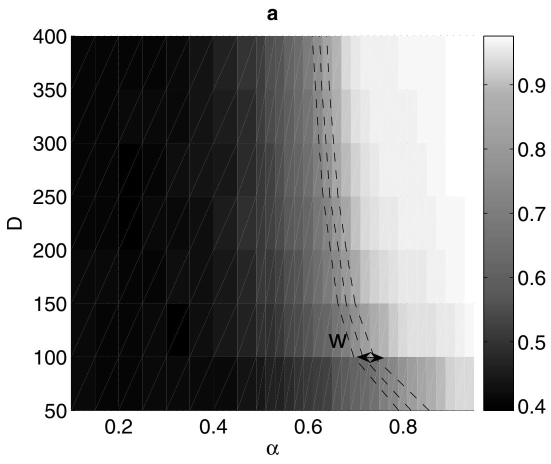

FIG. 6. The shading denotes the strength of patterns or ρ^*. The ordinate denotes D, the ratio between diffusion constants, while the abscissa denotes α, the diffusion exponent. There are three dotted lines marking where $\rho^*=0.77, 0.82, 0.87$ (from left to right). M is fixed at 5000.

$$\mathbf{n}(x,t) = \mathbf{n}_0(x)\,\delta(t) + \int_{x'}\int_{t'=0}^{t} \varphi(x-x')\mathbf{f}(t-t') \times \mathbf{B}(t-t')\mathbf{n}(x',t'). \tag{22}$$

Apply the Fourier transform to obtain

$$\hat{\mathbf{n}}(q,t) = \hat{\mathbf{n}}_0(q)\,\delta(t) + \hat{\varphi}(q)\int_{t'=0}^{t} \mathbf{f}(t-t')\mathbf{B}(t-t')\hat{\mathbf{n}}(q,t'). \tag{23}$$

Apply the Laplace transform to obtain

$$\hat{\hat{\mathbf{n}}}(q,s) = \hat{\mathbf{n}}_0(q) + \hat{\varphi}(q)\mathbf{C}(s)\hat{\hat{\mathbf{n}}}(q,s), \tag{24}$$

where

$$\mathbf{C}(s) = \mathcal{L}(\mathbf{f}(t)\mathbf{B}(t))(s). \tag{25}$$

Rearrange Eq. (24) to get

$$[\mathbf{I} - \hat{\varphi}(q)\mathbf{C}(s)]\hat{\hat{\mathbf{n}}}(q,s) = \hat{\mathbf{n}}_0(q). \tag{26}$$

When $\mathbf{n}_0=\mathbf{0}$, we expect nontrivial solutions to Eq. (24). Hence,

$$\det[\mathbf{I} - \hat{\varphi}(q)\mathbf{C}(s)] = 0. \tag{27}$$

It remains to determine $\mathbf{C}(s)$. Note that $\mathbf{V}$ is independent of t. Let $\mathbf{\Lambda}=\mathrm{diag}(\lambda_1,\lambda_2)$. By Eqs. (25) and (18),

$$C_{i,j}(s) = \mathcal{L}\left(\sum_k f_i(t)V_{i,k}\exp(\lambda_k t)V^{-1}_{k,j}\right). \tag{28}$$

Since $\hat{f}_i(s)=1-(\eta_i s)^{\alpha}$, we have

$$C_{i,j}(s) = \sum_k V_{i,k}V^{-1}_{k,j}[1-\eta_i^{\alpha}(s-\lambda_k)^{\alpha}]. \tag{29}$$

Let $\mathbf{\Omega}=\mathrm{diag}(\eta_1,\eta_2)$. Rewrite Eq. (29) in matrix form,

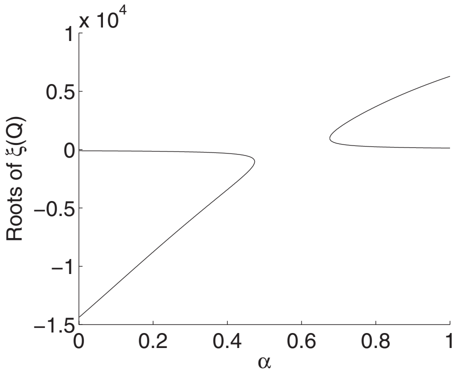

FIG. 7. Roots of $\xi(Q)$ [see Eq. (31)] versus α. D, the ratio between diffusion constants, is fixed at 100. When the roots of $\xi(Q)$ are positive, patterns can form. In this case, when $\alpha>0.676$, patterns start forming according to the linear stability analysis.

$$\mathbf{C} = \mathbf{I} - \mathbf{\Omega}^{\alpha}\mathbf{V}(s\mathbf{I}-\mathbf{\Lambda})^{\alpha}\mathbf{V}^{-1}. \tag{30}$$

Substitute the above into Eq. (27) to obtain an expression of the form $q^4+J_1(s)q^2+J_2(s)=0$. Set $s=0$ since we are interested in the asymptotic behavior. Let $Q=q^2$. This gives us a polynomial

$$\xi(Q) = Q^2 + J_1(0)Q + J_2(0). \tag{31}$$

Patterns can form only when there are real positive solutions for $\xi(Q)$. Now apply this to the Schnakenberg model. From Eq. (16), it can be shown that

$$\mathbf{N}_0 = \{(a+b)M = M, b(a+b)^{-2}M = bM\},$$

$$\mathbf{A} = \begin{pmatrix} -1+2b & 1 \\ -2b & -1 \end{pmatrix},$$

$$\mathbf{V} = \begin{pmatrix} 2b-2\sqrt{b^2-2b} & 2b+2\sqrt{b^2-2b} \\ -4b & -4b \end{pmatrix},$$

$$\lambda_1 = b-1-\sqrt{b^2-2b},$$

$$\lambda_2 = b-1+\sqrt{b^2-2b}. \tag{32}$$

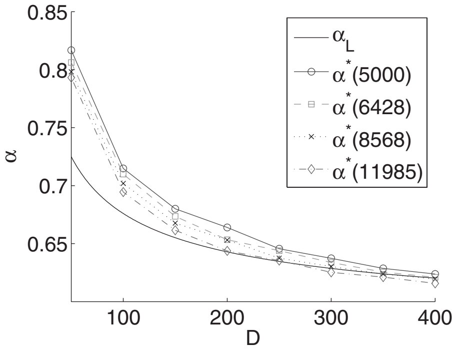

FIG. 8. α_L and $\alpha^*(m)$ versus D, for various values of m. $\alpha^*(m)$ is the α at which $\rho^*=0.82$, when the total number of U particles, M, is equal to m. α_L is the α at which patterns stop forming, according to linear stability analysis.

$\xi(Q)$ is now completely determined. For illustration, we plot its solutions with $D=100$, versus α in Fig. 7. When $\alpha \approx 0.68$, $\xi(Q)$ begins having positive real roots. Therefore, α_L, the value of α below which patterns cannot form, is approximately 0.68. It turns out that α_L is always a root of the discriminant $J_1(0)^2-4J_2(0)$ and can be obtained numerically.

Consider a plot of ρ^* versus α obtained from the simulations, for example Fig. 2. Define the function $\alpha^*(m)$ to be the α at which $\rho^*=0.85$, when $M=m$. We shall compare $\alpha^*(m)$ for various values of m, with α_L. Figure 8 shows that for large D, α_L and α^* coincide, suggesting that there is agreement between linear stability analysis and our simulations. (As to why there are deviations at small D, we postulate that finite-size effects are the cause: as M gets smaller, the mean field approximation made by the linear stability analysis becomes less reasonable, and α_L and α^* will begin to disagree.)

IV. CONCLUSION

We have designed a Monte Carlo algorithm for simulating reaction-subdiffusion systems, and used it to study the Schnakenberg model. The two reacting species are assumed to have the same diffusion exponents, and we find that when the diffusion exponent α, the total number of particles controlled by M, and the ratio between the diffusion constants D are large enough, patterns can be formed. We have also systematically worked out the parameter space in which patterns may form. A linear stability analysis is performed to obtain critical values of α, which we call α_L, below which there are no patterns, and it agrees with our simulation results.

In future, our algorithm can be used to study other systems as well, for example the Brusselator or the Gierer-Meinhardt models, or even growth and dispersal of biological species [14]. Superdiffusion can also be implemented by allowing particles to jump more than one bin away [13].

ACKNOWLEDGMENT

We acknowledge discussions with Grant Lythe and Chee-Meng Tan.

[1] A. M. Turing, Philos. Trans. R. Soc. London, Ser. B **237**, 37 (1952).
[2] K. Kondo and R. Asai, Nature (London) **376**, 765 (1995).
[3] I. Salazar-Ciudad and J. Jernvall, Proc. Natl. Acad. Sci. U.S.A. **99**, 8116 (2002).
[4] E. Ammelt, Yu. A. Astrov, and H.-G. Purwins, Phys. Rev. E **55**, 6731 (1997).
[5] Yu. A. Astrov, I. Müller, E. Ammelt, and H.-G. Purwins, Phys. Rev. Lett. **80**, 5341 (1998).
[6] F. T. Arecchi, S. Boccaketti, and P. Ramazza, Proc. Natl. Acad. Sci. U.S.A. **318**, 1 (1999).
[7] M. Weiss, M. Elsner, F. Kartberg, and T. Nilsson, Biophys. J. **87**, 3518 (2004).
[8] H. Scher and E. W. Montroll, Phys. Rev. B **12**, 2455 (1975).
[9] G. Drazer and D. H. Zanette, Phys. Rev. E **60**, 5858 (1999).
[10] M. O. Vlad, G. Cerofolini, and J. Ross, Phys. Rev. E **62**, 837 (2000).
[11] B. Berkowitz and H. Scher, Water Resour. Res. **31**, 1461 (1995).
[12] D. Benson, S. Wheatcraft, and M. Meerschaert, Water Resour. Res. **36**, 1413 (2000).
[13] D. del-Castillo-Negrete, B. A. Carreras, and V. E. Lynch, Phys. Rev. Lett. **91**, 018302 (2003).
[14] M. K. B. Baeumer and M. M. Meerschaert, Bull. Math. Biol. **69**, 2281 (2007).
[15] R. Metzler and J. Klafter, Phys. Rep. **339**, 1 (2000).
[16] J. Klafter, A. Blumen, and M. F. Shlesinger, Phys. Rev. A **35**, 3081 (1987).
[17] R. Metzler and J. Klafter, J. Phys. A **37**, R161 (2004).
[18] M. M. Meerschaert, D. Benson, H. Scheffler, and B. Baeumer, Phys. Rev. E **65**, 041103 (2002).
[19] K. S. Miller and B. Ross, *An Introduction to the Fractional Calculus and Fractional Differential Equations* (Wiley, New York, 1993).
[20] K. B. Oldham and J. Spanier, *The Fractional Calculus* (Academic Press, New York, 1974).
[21] G. Zaslavsky, Physica D **76**, 110 (1994).
[22] S. B. Yuste and K. Lindenberg, Chem. Phys. **284**, 169 (2002).
[23] S. B. Yuste, L. Acedo, and K. Lindenberg, Phys. Rev. E **69**, 036126 (2004).
[24] I. M. Sokolov, M. G. W. Schmidt, and F. Sagues, Phys. Rev. E **73**, 031102 (2006).
[25] S. B. Yuste, K. Lindenberg, and J. J. Ruiz-Lorenzo, in *Anomalous Transport: Foundations and Applications*, edited by R. Klages, G. Radons, and I. M. Sokolov (Wiley-VCH, Weinheim, 2007).
[26] A. Yadav and W. Horsthemke, Phys. Rev. E **74**, 066118 (2006).
[27] M. Weiss, Phys. Rev. E **68**, 036213 (2003).
[28] J. Schnakenberg, J. Theor. Biol. **81**, 389 (1979).
[29] D. T. Gillespie, J. Phys. Chem. **81**, 2340 (1977).
[30] Y. Nec and A. A. Nepomnyashchy, J. Phys. A: Math. Theor. **40**, 14687 (2007).
[31] K.-H. Chiam, C. M. Tan, V. Bhargava, and G. Rajagopal, Phys. Rev. E **74**, 051910 (2006).
[32] D. Bernstein, Phys. Rev. E **71**, 041103 (2005).
[33] J. Elf, A. Dončić, and M. Ehrenberg, Proc. SPIE **5110**, 114 (2003).
[34] T. S. Shimizu, S. V. Aksenov, and D. Bray, J. Mol. Biol. **329**, 291 (2003).
[35] A. B. Stundzia and C. J. Lumsden, J. Comput. Phys. **127**, 196 (1996).
[36] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *An Introduction to Algorithms* (MIT Press, Cambridge, MA, 2001).