AIP | The Journal of Chemical Physics

Exact on-lattice stochastic reaction-diffusion simulations using partial-propensity methods

Rajesh Ramaswamy and Ivo F. Sbalzarini

Citation: J. Chem. Phys. **135**, 244103 (2011); doi: 10.1063/1.3666988

View online: http://dx.doi.org/10.1063/1.3666988

View Table of Contents: http://jcp.aip.org/resource/1/JCPSA6/v135/i24

Published by the American Institute of Physics.

AIPAdvances
Submit Now
Explore AIP's new open-access journal
• Article-level metrics now available
• Join the conversation! Rate & comment on articles

Exact on-lattice stochastic reaction-diffusion simulations using partial-propensity methods

Rajesh Ramaswamy[a)] and Ivo F. Sbalzarini[b)]
MOSAIC Group, Institute of Theoretical Computer Science and Swiss Institute of Bioinformatics, ETH Zurich, CH–8092 Zürich, Switzerland

(Received 7 September 2011; accepted 18 November 2011; published online 23 December 2011)

Stochastic reaction-diffusion systems frequently exhibit behavior that is not predicted by deterministic simulation models. Stochastic simulation methods, however, are computationally expensive. We present a more efficient stochastic reaction-diffusion simulation algorithm that samples realizations from the exact solution of the reaction-diffusion master equation. The present algorithm, called partial-propensity stochastic reaction-diffusion (PSRD) method, uses an on-lattice discretization of the reaction-diffusion system and relies on partial-propensity methods for computational efficiency. We describe the algorithm in detail, provide a theoretical analysis of its computational cost, and demonstrate its computational performance in benchmarks. We then illustrate the application of PSRD to two- and three-dimensional pattern-forming Gray-Scott systems, highlighting the role of intrinsic noise in these systems. © *2011 American Institute of Physics*. [doi:10.1063/1.3666988]

I. INTRODUCTION

Chemically reactive systems exhibiting spatial heterogeneity are often modeled using reaction-diffusion equations.[1–10] Reaction-diffusion models explicitly capture spatial variations of the concentration fields, accounting for diffusive transport of reactants and products to and from reaction sites. Spatial heterogeneity is sustained when diffusion of chemicals is slower than reactions between them. In the limit of large numbers of molecules, reaction-diffusion processes can be modeled continuously as systems of coupled partial differential equations (frequently called *reaction-diffusion equations* or *Fisher-KPP equations*[1,2]) governing the spatiotemporal evolution of the smooth concentration fields of all chemical species. Continuum reaction-diffusion models can exhibit nontrivial spatiotemporal dynamics, such as traveling concentration fronts[11] and inhomogeneous stationary concentration distributions ("Turing patterns").[3,12–14] These phenomena have been successful in explaining a number of experimental observations, including localization of cell division sites in *E. coli*[15] and "black eyes" patterns in the chlorite-iodide-malonic acid reaction.[16–18] For low molecular copy numbers, however, continuum models fail to provide an accurate description of the spatiotemporal dynamics of reaction-diffusion systems. In particular, intrinsic noise from the apparent molecular discreteness, leading to stochasticity of chemical reactions, alters front propagation dynamics[19] and Turing patterns[5,6] in a nontrivial way. This is because fluctuations in the molecule populations may no longer be negligible, and correlated fluctuations may lead to deviations from deterministic behavior.[5,6,8,20,21] These effects can be accounted for by stochastic reaction diffusion (SRD) simulations.

There are mainly two types of SRD simulations: on-lattice (or compartment-based) simulations and off-lattice (or particle-based) simulations. On-lattice simulations include the next subvolume method (NSM) (Ref. 5), whereas Greens-function reaction dynamics (GFRD) (Ref. 22) and Brownian dynamics (BD) (Ref. 23) are examples of off-lattice schemes. On-lattice SRD simulations[5,24–29] are based on dividing (discretizing) the computational domain into subvolumes, in each of which the chemical reaction system is assumed to be well mixed (spatially homogeneous). It is further postulated that only molecules within the same subvolume can react with each other, effectively treating molecules of the same chemical in different subvolumes as different species. Diffusion is modeled as unimolecular "diffusion reactions" representing jumps of molecules between neighboring subvolumes. The on-lattice approach hence describes the reaction-diffusion system as a large chemical reaction network with the number of species proportional to the product of the actual number of chemical species and the number of subvolumes used to discretize space. The kinetics of this enlarged reaction network can be mathematically described by the on-lattice reaction-diffusion master equation (RDME), analogous to the chemical master equation (CME).[30] Off-lattice SRD simulations[22,23,31,32] are based on computational particles mimicking the Brownian motion of molecules, whereby the molecules involved in a bimolecular reaction react with a certain probability when the distance between them is smaller than a pre-defined reaction radius.

Here, we focus on on-lattice SRD simulations in order to avoid computationally expensive collision detection and time-step adaptation mechanisms.[33] Since on-lattice SRD is described by a system of chemical reactions modeled by the RDME, it can be exactly simulated using Gillespie's stochastic simulation algorithm (SSA).[30,34] SSA samples trajectories from the exact solution of the master equation by sampling

[a)] Electronic mail: rajeshr@ethz.ch.
[b)] Author to whom correspondence should be addressed. Electronic mail: ivos@ethz.ch.

the index of the next reaction, the time to the next reaction, and updating the reaction probability rates (called "propensities"). Different SSA formulations are available that use different sampling and update algorithms, including the direct method (DM),[34,35] the first reaction method (FRM),[34,35] the next reaction method (NRM),[36] the optimized direct method (ODM),[37] the sorting direct method (SDM),[38] the SSA with composition-rejection sampling (SSA-CR),[39] and partial-propensity methods[40] such as the partial-propensity direct method (PDM),[41] the sorting PDM (SPDM),[41] and the partial-propensity SSA with composition-rejection sampling (PSSA-CR).[42] Directly using any of these SSA formulations for the RDME, without adapting it to the specifics of on-lattice SRD simulations, would be correct, but computationally and/or memory inefficient since the number of species and the number of reactions increase linearly with the number of subvolumes. A more efficient way of performing on-lattice SRD simulation is to first sample the subvolume in which the next reaction will happen and then sample the index of the reaction within that subvolume. This is, for example, done in NSM[5] as implemented in the MesoRD software package.[24] NSM uses NRM for sampling the subvolume and DM for sampling the reaction within that subvolume.[5,24] For a chemical reaction network with N species and M reactions, the computational cost (here formalized using the Bachmann-Landau "big-O" notation) of NSM to perform an on-lattice SRD simulation in a three-dimensional (3D) computational domain divided into N_v subvolumes is $O(\log_2 N_\mathrm{v} + Mf_\mathrm{r} + 6N(1 - f_\mathrm{r}))$,[5,24] where f_r is the fraction of firings accounted for by "real" reactions and $(1 - f_\mathrm{r})$ the fraction of firings of "diffusion reactions." This is composed of the $O(\log_2 N_\mathrm{v})$ cost for maintaining the subvolume priority queue and the $O(Mf_\mathrm{r} + 6N(1 - f_\mathrm{r}))$ cost for sampling the next reaction. $M + 6N$ is the number of reactions in each subvolume, composed of the M "real" chemical reactions and the $6N$ "diffusion reactions" to the 6 face-connected neighboring subvolumes in a uniform Cartesian 3D mesh (in 2D this would be $4N$).

If the molecular population increases, the time step of exact SSAs decreases, increasing the runtime of the simulations. This can be alleviated by approximate SSAs that use a fixed time step to sample trajectories from an approximate solution of the master equation. In this spirit, on-lattice SRD simulations involving larger population sizes can be accelerated using approximate SSAs.[25–29] Here, we focus on exact on-lattice SRD formulations since they are parameter free and do not require prescription of a time step size or a target error level.

We propose a novel exact on-lattice SRD simulation method, called PSRD, using partial propensities. Partial-propensity methods are exact SSAs with a computational cost that scales at most linearly with the number of species in the reaction network.[40–42] For weakly coupled reaction networks, where the number of reactions influenced by any other reaction is asymptotically independent of system size, the computational cost of partial-propensity methods is bounded by a constant.[42] This is particularly advantageous for on-lattice SRD simulations, where the number of neighboring subvolumes influenced by any subvolume is constant (6 in 3D, 4 in 2D), independent of the total number of subvolumes used to discretize space. PSRD hence uses composition-rejection sampling to find the next subvolume with an $O(G_\mathrm{a})$ cost, and then uses SPDM inside that subvolume to sample the next reaction with a cost of $O(N)$. G_a is the logarithm of the ratio of the maximum to the minimum nonzero subvolume propensities, which is at most $O(\log_2 N_\mathrm{v})$. PSRD thus has an overall computational cost of $O(G_\mathrm{a} + N)$, which is asymptotically bounded from above by $O(\log_2 N_\mathrm{v} + N)$ and independent of the ratio between "real" and "diffusion" reactions. This is achieved by restricting the method to elementary chemical reactions, under the premise that any non-elementary reactions can be equivalently decomposed into elementary reactions[30,43,44] at the expense of a linear (in the order of the highest non-elementary reaction) increase in network size. We demonstrate the scaling of the computational cost of PSRD on two types of reaction networks: one in which the number of reactions M increases super-linearly with the number of species N (a strongly coupled network), and a second in which M is proportional to N (a weakly coupled network). Finally, we demonstrate the application of PSRD to pattern-forming stochastic Gray-Scott systems,[3,45–48] highlighting the effect of intrinsic noise on the resulting Turing patterns.

II. ON-LATTICE STOCHASTIC REACTION-DIFFUSION

We recall the concept of on-lattice SRD simulations using an example reaction network. In the benchmarks presented below we assume that the boundary of the computational domain is reflective (no-flux boundary condition), except for the showcases in Sec. V, where we use periodic boundary conditions. Other boundary conditions can be treated as described by Erban and Chapman (2007)[49] (see Sec. 2 in their article). The scaling of the computational cost of on-lattice SRD simulations, however, is independent of the type of the boundary condition.

A. General concept

Consider the example of the following trimerization reaction in a 3D cuboidal reactor of dimension $L_x \times L_y \times L_z$ and volume $\Omega = L_x L_y L_z$:

$$\begin{aligned}
&\text{Reaction 1} : && \emptyset && \overset{k_1}{\rightarrow} && \mathrm{S}_1\\
&\text{Reaction 2} : && \mathrm{S}_1 + \mathrm{S}_1 && \overset{k_2}{\rightarrow} && \mathrm{S}_2\\
&\text{Reaction 3} : && \mathrm{S}_1 + \mathrm{S}_2 && \overset{k_3}{\rightarrow} && \mathrm{S}_3\\
&\text{Reaction 4} : && \mathrm{S}_3 && \overset{k_4}{\rightarrow} && \emptyset.
\end{aligned} \tag{1}$$

The k's are the macroscopic reaction-rate constants. This reaction network has $N = 3$ species and $M = 4$ reactions. We choose this reaction network as an example since it contains all types of elementary reactions: reaction 1 is a source reaction, reaction 2 a bimolecular reaction between the same species (homo-bimolecular reaction), reaction 3 a bimolecular reaction between two different species (hetero-bimolecular reaction), and reaction 4 is a unimolecular reaction. Any non-

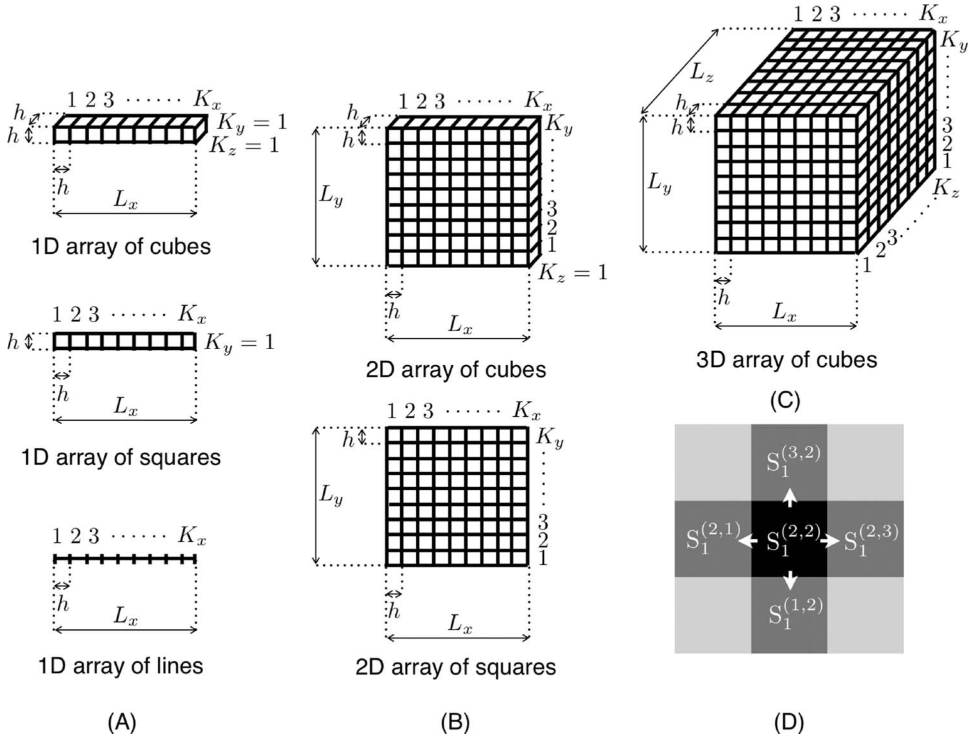

FIG. 1. Partitioning of the computation domain into subvolumes. (A–C) Different possibilities of a subdiving a box-shaped computational domain in one, two, and three dimensions, respectively. L_x, L_y, and L_z are the edge lengths of the computational domain in each direction. K_x, K_y, and K_z are the numbers of subvolumes of edge length h in each direction. (D) Diffusion is modeled as jump "reactions" to face-connected subvolumes. The same chemical in different subvolumes is treated as a different species. Unimolecular "diffusion reactions" convert species as shown.

elementary reaction involving $r > 2$ reactants can be broken down to a set of $2r - 3$ elementary reactions by introducing $r - 2$ additional species.[43,44] The reaction-propensity a_μ of reaction μ is defined as the probability rate of firing of that reaction. Each a_μ is computed as the product of the reaction degeneracy and the specific probability rate c_μ of that reaction. The reaction degeneracy is the number of distinct combinations (collision pairs) of reactant molecules that can be formed, and the specific probability rate is the probability rate of the reaction when only one molecule of each reactant is present. According to these definitions, the reaction propensities for the reaction network in Eq. (1) are as follows:

$$a_\mu = \begin{cases} c_\mu, & c_\mu = k_1\Omega, \quad \text{if } \mu = 1 \\ \frac{1}{2}n_1(n_1 - 1)c_\mu, & c_\mu = 2k_2\Omega^{-1}, \quad \text{if } \mu = 2 \\ n_1 n_2 c_\mu, & c_\mu = k_3\Omega^{-1}, \quad \text{if } \mu = 3 \\ n_3 c_\mu, & c_\mu = k_4, \quad \text{if } \mu = 4\,, \end{cases} \tag{2}$$

where n_i denotes the population of species S_i, i.e., the number of molecules of S_i present in the system.

If the characteristic time of diffusion of the species is comparable to or larger than the characteristic time of reaction, the system will exhibit spatial inhomogeneities and diffusion of the species in the reaction network needs to be explicitly accounted for. In on-lattice SRD methods, this is done by dividing the computational domain into subvolumes within which the system is assumed to be well mixed. The chemical species in each subvolume can (i) react with each other in bimolecular reactions, (ii) undergo unimolecular reactions, or (iii) appear from source reactions. In all cases, the products are formed in the same subvolume and species from different subvolumes can not react with each other. Diffusion of molecules is modeled as a jump process from a subvolume to any of the face-connected neighboring subvolumes.

Assume that we divide the 3D computational domain into $N_\text{v} = K_x K_y K_z$ equisized cubic subvolumes of edge length $h = L_x/K_x = L_y/K_y = L_z/K_z$ and volume $\Omega_\text{c} = h^3$ (see Fig. 1(c); the one- and two-dimensional cases are illustrated in Figs. 1(a) and 1(b), respectively). The subvolumes are indexed by their Cartesian mesh coordinates over the set[50]

$$\begin{aligned} I_\text{all} = \{(l, m, n) \mid l, m, n &\text{ are integers such that} \\ 1 \le l \le K_x\,;\; 1 \le m &\le K_y\,;\; 1 \le n \le K_z\,\} \end{aligned} \tag{3}$$

and the set of face-connected neighbors of a subvolume with index (l, m, n) is

$$C_{l,m,n} = \{(l, m, n) + \boldsymbol{\delta} \mid (l, m, n) + \boldsymbol{\delta} \in I_\text{all}\} \tag{4}$$

such that $\boldsymbol{\delta} \in E = \{(1, 0, 0), (-1, 0, 0), (0, 1, 0), (0, -1, 0), (0, 0, 1), (0, 0, -1)\}$. Hence, the on-lattice reaction-diffusion system of the reaction network in Eq. (1) can be written as

$$\left.\begin{array}{ccccc}
\varnothing & \xrightarrow{k_1^{(l,m,n)}} & \mathrm{S}_1^{(l,m,n)}, & & a_1^{(l,m,n)} \\
\mathrm{S}_1^{(l,m,n)}+\mathrm{S}_1^{(l,m,n)} & \xrightarrow{k_2^{(l,m,n)}} & \mathrm{S}_2^{(l,m,n)}, & & a_2^{(l,m,n)} \\
\mathrm{S}_1^{(l,m,n)}+\mathrm{S}_2^{(l,m,n)} & \xrightarrow{k_3^{(l,m,n)}} & \mathrm{S}_3^{(l,m,n)}, & & a_3^{(l,m,n)} \\
\mathrm{S}_3^{(l,m,n)} & \xrightarrow{k_4^{(l,m,n)}} & \varnothing, & & a_4^{(l,m,n)} \\
\mathrm{S}_1^{(l,m,n)} & \xrightarrow{D_1/h^2} & \mathrm{S}_1^{\mathbf{c}}, & & a_5^{(l,m,n)} \\
\mathrm{S}_2^{(l,m,n)} & \xrightarrow{D_2/h^2} & \mathrm{S}_2^{\mathbf{c}}, & & a_6^{(l,m,n)} \\
\mathrm{S}_3^{(l,m,n)} & \xrightarrow{D_3/h^2} & \mathrm{S}_3^{\mathbf{c}}, & & a_7^{(l,m,n)}
\end{array}\right\} \quad \forall (l,m,n)\in I_{\text{all}}, \quad \mathbf{c}\in C_{l,m,n}, \qquad (5)$$

where $\mathrm{S}_i^{(l,m,n)}$ denotes species S_i in subvolume (l, m, n), the k's the macroscopic reaction rates, and the a's the corresponding propensities. In general, the k's can be different in different subvolumes, which is explicitly shown in Eq. (5) by indexing them with the subvolume index. Diffusion of species S_i with diffusion constant D_i is modeled as jumps to face-connected neighboring subvolumes as illustrated in Fig. 1(d). Equation (5) models the on-lattice reaction-diffusion system as a system of chemical reactions composed of $3K_xK_yK_z = 3N_\text{v}$ species and $22K_xK_yK_z - 6(K_xK_y + K_yK_z + K_xK_z) = 22N_\text{v} - 6N_\text{v}(\frac{1}{K_x} + \frac{1}{K_y} + \frac{1}{K_z})$ reactions ($4K_xK_yK_z$ "real" reactions and $3(6K_xK_yK_z - 6(K_xK_y + K_yK_z + K_xK_z))$ "diffusion reactions", accounting for the missing neighboring subvolumes at the domain boundary). In general, the 3D SRD dynamics of N species and M reactions in a computational domain with reflective boundaries and $K_x \times K_y \times K_z$ subvolumes can be modeled by a chemical reaction network consisting of $NK_xK_yK_z = NN_\text{v}$ species and $(M + 6N)K_xK_yK_z - 2(K_xK_y + K_yK_z + K_xK_z)N = (M + 6N)N_\text{v} - 2(\frac{1}{K_x} + \frac{1}{K_y} + \frac{1}{K_z})N_\text{v}N$ reactions. For other boundary conditions, the number of reactions is $(M + 6N)N_\text{v}$, accounting for the diffusive fluxes across the boundary.

For inhomogeneous diffusion, D_i additionally depends on the subvolume index (l, m, n). For anisotropic diffusion, D_i depends on the direction of the jump reaction. These extensions are straightforward to include in any on-lattice SRD framework.

The propensities of the reactions in Eq. (5) are as follows:

$$a_\mu^{(l,m,n)} = \begin{cases}
c_\mu^{(l,m,n)}, & c_\mu^{(l,m,n)} = k_1^{(l,m,n)}\Omega_\text{c}, \quad \text{if } \mu = 1\\
\frac{1}{2}n_1^{(l,m,n)}(n_1^{(l,m,n)}-1)c_\mu^{(l,m,n)}, & c_\mu^{(l,m,n)} = 2k_2^{(l,m,n)}\Omega_\text{c}^{-1}, \quad \text{if } \mu = 2\\
n_1^{(l,m,n)}n_2^{(l,m,n)}c_\mu^{(l,m,n)}, & c_\mu^{(l,m,n)} = k_3^{(l,m,n)}\Omega_\text{c}^{-1}, \quad \text{if } \mu = 3\\
n_3^{(l,m,n)}c_\mu^{(l,m,n)}, & c_\mu^{(l,m,n)} = k_4^{(l,m,n)}, \quad \text{if } \mu = 4\\
n_1^{(l,m,n)}c_\mu^{(l,m,n)}, & c_\mu^{(l,m,n)} = D_1h^{-2}, \quad \text{if } \mu = 5\\
n_2^{(l,m,n)}c_\mu^{(l,m,n)}, & c_\mu^{(l,m,n)} = D_2h^{-2}, \quad \text{if } \mu = 6\\
n_3^{(l,m,n)}c_\mu^{(l,m,n)}, & c_\mu^{(l,m,n)} = D_3h^{-2}, \quad \text{if } \mu = 7\,,
\end{cases} \qquad (6)$$

where $n_i^{(l,m,n)}$ is the population of species $\mathrm{S}_i^{(l,m,n)}$ (i.e., species S_i in subvolume (l, m, n)) and $c_\mu^{(l,m,n)}$ is the specific probability rate of reaction μ in subvolume (l, m, n). These formulations for the propensities directly follow from the same argument as the propensities in Eq. (2) for the reaction system given in Eq. (1). The rates of the "diffusion reactions" always scale as h^{-2}, irrespective of the dimension of the subvolumes.

B. Discretization-corrected propensities

The propensity formulations in Eq. (6) may lead to artifacts in the kinetics introduced by the spatial discretization.[33,50] This is due to the subdivision of the reaction space into disjoint subvolumes. This subdivision is fundamentally different from the one used in spatial discretization of continuum models (e.g., finite-difference or finite-volume methods). While in discretizing continuum models more resolution (smaller subvolumes) is always better, this is not necessarily the case in SRD simulations. This is because in SRD simulations the subvolumes introduce spurious physical boundaries; molecules in one subvolume cannot react with molecules in a neighboring subvolume, even though for molecules close to a subvolume boundary the closest collision partner could be in a neighboring subvolume. The subvolumes thus define closed, well-mixed reaction spaces of volume $\Omega_c \ll \Omega$. In order for the reaction system to be well mixed within each subvolume, the subvolume edge lengths have to be much smaller than the Kuramoto length,[51] hence

$$h \ll h_{\max} = \sqrt{2dDt_r}\,, \qquad (7)$$

where d is the dimension of the subvolume. The characteristic time t_r of the fastest reactions in the system can be estimated from the time-autocorrelation function of species populations simulated using an exact SSA. While t_r depends on the reactor volume Ω, an estimate for it can be obtained from the linear-noise approximation of the CME.[52] On length scales larger than $h_{\max}$ the subvolumes are no longer spatially homogeneous (well mixed).

In addition to this upper bound on h, there may also be a lower bound. It is, for example, known that chemical kinetics in small volumes is quantitatively and qualitatively altered.[52,53] It has further been shown that the RDME has different solutions depending on the level of spatial discretization of the computational domain.[50] If the discretization becomes too fine, the RDME even yields unphysical results.[33,50] These artifacts are introduced by the artificial subdivision of space, which causes the propensities in the RDME to become inconsistent with Smoluchowski's microscopic reaction-diffusion framework.[33,54] The propensities in on-lattice SRD simulations hence need to be corrected for the spatial discretization.

At present, there are two strategies toward deriving discretization-corrected propensities. The first is based on the premise that for a well-stirred system of reactions the kinetics of the reaction-diffusion process should not depend on the resolution of the spatial discretization.[50] In this strategy, only the propensities of bimolecular reactions need to be corrected. This discretization-correction framework has only been derived for 3D reactors subdivided into an equal number of cubic subvolumes along each dimension, and it imposes a lower bound on the admissible subvolume size, given by the constraint that the corrected reaction propensities have to be nonnegative.[50]

The second strategy derives discretization-corrected propensities such that the RDME becomes consistent with Smoluchowski's microscopic reaction-diffusion framework.[33,54] In this strategy, the discretization-corrected propensities depend on the populations of reactant molecules in the neighboring subvolumes, rendering the correction non-local and reaction-dependent. This approach is valid also for non-cubic computational domains and non-3D simulations, and it does not impose any lower bound on the subvolume size h. It has been shown to provide a seamless transition between Smoluchowski's microscopic framework and that of on-lattice SRD as based on the RDME.[33,54]

We use the discretization-corrected propensities of Erban and Chapman (2009) wherever applicable. Extending the present on-lattice SRD method to the framework proposed by Fange *et al.* (2010)[33] should also be possible. The scaling of the computational cost of on-lattice SRD methods, however, is independent of the formulation used for the propensities.

We note that similar corrections are also necessary in off-lattice SRD simulations, where the artificial spatial discretization is introduced by the reaction radius.[50] Moreover, off-lattice simulation methods frequently use a fixed time step, which renders them more related to approximate SSAs than to exact ones and potentially hampers convergence toward the correct dynamics as the time-step size is reduced.[55]

C. The next subvolume method (NSM) for on-lattice stochastic reaction-diffusion simulations

NSM simulates the on-lattice SRD system by sampling from the conditional joint probability distribution function (PDF) for the time τ to the next reaction, the index μ of the next reaction, and the subvolume (l, m, n) containing the next reaction, given the current population $\mathbf{n}(t)$ at time t. This joint PDF results from the on-lattice RDME and is given by

$$p(\tau, \mu, l, m, n\,|\,\mathbf{n}(t)) = p(\tau)\; p(l, m, n)\; p(\mu\,|\,l, m, n)\,. \qquad (8)$$

Here, $p(\tau)$ is the continuous PDF for the time to the next reaction, τ, given by

$$p(\tau) = a\mathrm{e}^{a\tau}, \qquad (9)$$

where a is the total propensity of the system. The discrete PDF $p(l, m, n)$ for the subvolume (l, m, n) of the next reaction is given by

$$p(l, m, n) = \frac{a^{(l,m,n)}}{a}, \qquad (10)$$

where $a^{(l,\,m,\,n)}$ is the propensity of subvolume (l, m, n). The discrete PDF $p(\mu\,|\,l, m, n)$ for the next reaction μ within subvolume (l, m, n) is given by

$$p(\mu\,|\,l, m, n) = \frac{a_\mu^{(l,m,n)}}{a^{(l,m,n)}}, \qquad (11)$$

$a_\mu^{(l,m,n)}$ the propensity of reaction μ in subvolume (l, m, n).

Formally, $\mathbf{n}(t) = [n_1^{(1,1,1)}, \ldots, n_N^{(1,1,1)}, \ldots, n_1^{(K_x,K_y,K_z)}, \ldots, n_N^{(K_x,K_y,K_z)}](t)$, where $n_i^{(l,m,n)}(t)$ is the population of species S_i in subvolume (l, m, n) at time t, $a^{(l,m,n)} = \sum_\mu a_\mu^{(l,m,n)}$ the total propensity of all reactions in subvolume (l, m, n), and $a = \sum_{l=1}^{K_x} \sum_{m=1}^{K_y} \sum_{n=1}^{K_z} a^{(l,m,n)}$ the total propensity of all reactions across all subvolumes.

NSM (Ref. 5) is a popular and efficient algorithm for sampling trajectories of $\mathbf{n}(t)$ from the above PDF, which is the exact solution of the RDME. In NSM, the subvolume (l, m, n) in which the next reaction will occur is sampled first according to Eq. (10) and subsequently one of the reactions μ in that subvolume is sampled according to Eq. (11). The latter is done by first deciding whether the next reaction is a "real" or a "diffusion" reaction and then using linear search only over the corresponding reaction group.[24] The algorithm used in NSM to sample the next subvolume is inspired by the indexed priority queues used in the next reaction method.[36] Sampling a reaction within a subvolume is done using linear search as in Gillespie's original direct method.[34,35] The time to the next reaction is calculated from Eq. (9). After the chosen reaction fired, the population and the propensities of some of the reactions need to be updated. In NSM, the population is updated using a sparse representation of the stoichiometry matrix,[41] and the propensities are updated using a dependency graph.[36]

The computational cost of NSM is as follows: (i) $O(1)$ for sampling the subvolume; (ii) $O(Mf_{\mathrm{r}} + 6N(1 - f_{\mathrm{r}}))$ for sampling the next reaction within that subvolume, where M is the number of "real" reactions, $6N$ the number of "diffusion reactions" ($4N$ in 2D), and f_{r} the fraction of reaction firings accounted for by "real" reactions; (iii) $O(1)$ for updating the population; (iv) at most $O(M)$ for updating the propensities within a subvolume; (v) $O(\log_2 N_{\mathrm{v}})$ for updating the subvolume priority queue, where N_{v} is the number of subvolumes. The overall computational cost of NSM thus is $O(\log_2 N_{\mathrm{v}} + Mf_{\mathrm{r}} + 6N(1 - f_{\mathrm{r}}))$. The fraction f_{r} of "real" reaction firings decreases with increasing N_{v}. For small N_{v}, almost all reactions are "real" and the computational cost of NSM is $O(\log_2 N_{\mathrm{v}} + M)$. In particular, for $N_{\mathrm{v}} = 1$ the fraction $f_{\mathrm{r}} = 1$ and the computational cost of NSM is $O(M)$, as for Gillespie's DM.[34,35] For large N_{v}, the computational cost of NSM is $O(\log_2 N_{\mathrm{v}} + 6N)$ since $f_{\mathrm{r}} \ll 1$ and almost all reaction events pertain to "diffusion reactions."

III. THE PARTIAL-PROPENSITY STOCHASTIC REACTION-DIFFUSION METHOD

Combining ideas from NSM and partial-propensity SSAs,[41] we introduce a novel on-lattice SRD simulation method, the partial-propensity stochastic reaction-diffusion method (PSRD). PSRD is based on the idea of binning the subvolumes and determining the next subvolume using composition-rejection sampling.[39,42,56] Then, we use the concept of partial propensities[40–42] to sample the index of the next reaction within the selected subvolume.

A. General concept of PSRD

We summarize the general concepts of binned composition-rejection sampling and partial propensities. For a more detailed description, we refer to Sec. III B and to the corresponding original publications.[39–42]

1. Composition-rejection sampling to select the subvolume

Composition-rejection sampling[56–59] is an efficient algorithm to sample realizations of a random variable according to a given discrete probability distribution. In on-lattice SRD simulations, the discrete PDF for the subvolume index (l, m, n) is $p(l, m, n)$ (see Eq. (10)). The sampling process starts by binning the $a^{(l,m,n)}$ according to their values and then proceeds in two steps: The composition step is used to identify the bin by linear search, and the rejection step is used to identify the $a^{(l,m,n)}$, and hence the index of the subvolume (l, m, n), inside that bin.

2. Partial propensities to sample the next reaction within a subvolume

Partial propensities: The partial propensity of a reaction is defined as the propensity per molecule of one of its reactants.[41] For example, the partial propensity $\pi_\mu^{(l,m,n);(i)}$ of reaction μ within a subvolume (l, m, n) with respect to (perhaps the only) reactant $\mathrm{S}_i^{(l,m,n)}$ is $a_\mu^{(l,m,n)}/n_i^{(l,m,n)}$, where $a_\mu^{(l,m,n)}$ is the propensity of reaction μ in subvolume (l, m, n) and $n_i^{(l,m,n)}$ the population of $\mathrm{S}_i^{(l,m,n)}$ (i.e., the number of molecules of species S_i in subvolume (l, m, n)). The partial propensities of the three elementary reaction types within each subvolume (l, m, n) are:

- Bimolecular reactions $\mathrm{S}_i^{(l,m,n)} + \mathrm{S}_j^{(l,m,n)} \xrightarrow{c_\mu^{(l,m,n)}}$ Products: $a_\mu^{(l,m,n)} = n_i^{(l,m,n)} n_j^{(l,m,n)} c_\mu^{(l,m,n)}$ and $\pi_\mu^{(l,m,n);(i)} = n_j^{(l,m,n)} c_\mu^{(l,m,n)}$, $\pi_\mu^{(l,m,n);(j)} = n_i^{(l,m,n)} c_\mu^{(l,m,n)}$. If both reactants are of the same species, i.e., $\mathrm{S}_i^{(l,m,n)} = \mathrm{S}_j^{(l,m,n)}$, only one partial propensity exists, $\pi_\mu^{(l,m,n);(i)} = 1/2(n_i^{(l,m,n)} - 1)c_\mu^{(l,m,n)}$, because the reaction degeneracy is $1/2 n_i^{(l,m,n)}(n_i^{(l,m,n)} - 1)$.
- Unimolecular reactions $\mathrm{S}_i^{(l,m,n)} \xrightarrow{c_\mu^{(l,m,n)}}$ Products: $a_\mu^{(l,m,n)} = n_i^{(l,m,n)} c_\mu^{(l,m,n)}$ and $\pi_\mu^{(l,m,n);(i)} = c_\mu^{(l,m,n)}$. The "diffusion reactions" representing the jumps from a subvolume to one of its neighbors fall into this category.
- Source reactions $\emptyset \xrightarrow{c_\mu^{(l,m,n)}}$ Products: $a_\mu^{(l,m,n)} = c_\mu^{(l,m,n)}$ and $\pi_\mu^{(l,m,n);(0)} = c_\mu^{(l,m,n)}$.

We use the specific probability rates given in Eq. (6). In cases where the 3D computational domain is cubic with equal numbers of subvolumes in each dimension, we use the discretization-corrected specific probability rates as given by Erban and Chapman (2009)[50] for the bimolecular reactions. The computational cost and the formalism of PSRD, however, are independent of the formulation used for the specific probability rates.

We consider only elementary reactions under the premise that any reaction with three or more reactants can be decomposed into a set of elementary reactions.[30,43,44]

Sampling using partial propensities: Within the selected subvolume we use partial propensity methods to sample the next reaction according to Eq. (11). We group the partial propensities of all reactions within each subvolume according to the index of the factored-out reactant.[40–42] This results in at most $N + 1$ groups of size $O(N)$. Every reaction in a subvolume, and its corresponding partial propensity, are then identifiable by two indices: a *group index* and an *element index*. The group index identifies the partial-propensity group to which a reaction belongs and the element index identifies the position of the reaction inside that group. Determining the index of the next reaction is thus done by first sampling its group index and then the element index.

An intuitive way of interpreting this sampling scheme is to imagine the group index as the index of the first reactant of the next reaction (0 for source reactions) and the element index as the index of the second reactant, i.e., the reaction partner (none for unimolecular reactions). Partial-propensity methods hence sample reaction *partners* (group index: "what species is going to react next?" element index: "what species is it going to react with?") rather than reactions. This changes their computational cost from $O(M)$ to $O(N)$, which amounts to a cost reduction as M is usually larger than N. This is particularly true for strongly coupled networks, where M grows super-linearly with N.[41]

After the selected reaction has fired and the populations of the involved species have been updated according to the reaction stoichiometry, the affected partial propensities are updated using a dependency graph over species.[41] This dependency graph points to all partial propensities within the subvolume that need to be updated due to the change in population. If the executed reaction was a "diffusion reaction" modeling the jump of a molecule from a subvolume to one of its neighbours, we additionally update the population of that species in the corresponding neighboring subvolume and update the affected partial-propensities in the neighboring subvolume using the respective dependency over species. Since any partial propensity is a function of the population of at most one species, the number of updates is at most $O(N)$.

B. Detailed description of the PSRD algorithm

We provide a detailed description of the algorithms and data structures used in PSRD. The workflow of the algorithm is summarized in Table I.

1. Data structures

The population of species in each subvolume (l, m, n) is stored in an array $\mathbf{n}^{(l,m,n)}$. The partial propensities of the reactions within each subvolume (l, m, n) are stored in "partial-propensity structures" $\mathbf{\Pi}^{(l,m,n)} = \{\mathbf{\Pi}_i^{(l,m,n)}\}_{i=0}^{N}$ as one-dimensional arrays of one-dimensional arrays $\mathbf{\Pi}_i^{(l,m,n)}$. Each array $\mathbf{\Pi}_i^{(l,m,n)}$ contains the partial propensities belonging to group i in subvolume (l, m, n). The partial propensities of source reactions are stored as consecutive entries of the 0th array $\mathbf{\Pi}_0^{(l,m,n)}$. The partial propensities of all reactions in subvolume (l, m, n) that have species $\mathrm{S}_1^{(l,m,n)}$ as the factored-out reactant are stored as consecutive entries of $\mathbf{\Pi}_1^{(l,m,n)}$. In general, the ith array $\mathbf{\Pi}_i^{(l,m,n)}$ contains the partial propensities of all reactions in subvolume (l, m, n) that have $\mathrm{S}_i^{(l,m,n)}$ as the common factored-out reactant, provided these reactions have not yet been included in any of the previous $\mathbf{\Pi}_{j<i}^{(l,m,n)}$. That is, out of the two partial propensities of a bimolecular reaction with $\mathrm{S}_i^{(l,m,n)}$ and $\mathrm{S}_j^{(l,m,n)}$ as its reactants and $i < j$, $\pi_\mu^{(l,m,n);(i)}$ is part of $\mathbf{\Pi}_i^{(l,m,n)}$, and $\pi_\mu^{(l,m,n);(j)}$ is not stored anywhere. In order to save memory, we lump the "diffusion reactions" of each species within a subvolume into one reaction with no products. The specific probability rate of the lumped reaction is the sum of the specific probability rates of all "diffusion reactions" in that subvolume. Therefore, instead of storing 6 partial propensities in 3D (4 in 2D), we only store 1 partial propensity for the "diffusion reactions" of each species. This reduces the total number of reactions per subvolume from $M + 6N$ in 3D ($M + 4N$ in 2D) to $M + N$. For convenience, we define all reactions $\mu \leq M$ as "real" reactions and the reaction with index $\mu = M + i$ as the lumped "diffusion reaction" of species $\mathrm{S}_i^{(l,m,n)}$. See Sec. III B 2 for how the direction of a

TABLE I. The detailed algorithm of PSRD.

0. For a reaction network with N species and M reactions: Divide the computational box into $N_\mathrm{v} = K_x K_y K_z$ cubic subvolumes of edge length h. Formulate the reaction network modeling the reaction-diffusion system by including the jump reactions. The resulting reaction network for the reaction-diffusion system has at most $M + 6N$ reactions and N species in each subvolume. Lump the "diffusion reactions" of each species in each subvolume into one reaction with no products, such that number of reactions in each subvolume is $M + N$. Make sure that the reaction index of the lumped "diffusion reactions" is $>M$ and that reaction $\mu = M + i$ is the lumped "diffusion reaction" of species i.

1. Set time $t \leftarrow 0$. Initialize the data structures in each subvolume (l, m, n): the partial-propensity structure $\mathbf{\Pi}^{(l,m,n)}$, the group-sum array $\mathbf{\Lambda}^{(l,m,n)}$, $\mathbf{\Sigma}^{(l,m,n)}$, the population $\mathbf{n}^{(l,m,n)}$, the specific probability rates $\mathbf{c}^{(l,m,n)}$, and the total propensity in the subvolume, $a^{(l,m,n)}$. Also initialize the data structures global to all subvolumes containing the same reaction network: the look-up table $\mathbf{L}$, the sparse representation of the stoichiometry matrix $\mathbf{U}^{(1)}$ and $\mathbf{U}^{(2)}$, the dependency graph over species $\mathbf{U}^{(3)}$, and the total propensity of all subvolumes a. Bin the $a^{(l,m,n)}$ into G_a bins as described in Sec. III B 2.

2. While $t < t_\mathrm{f}$, repeat:

   2.1. Compute the time to the next reaction $\tau \leftarrow a^{-1} \ln(r^{-1})$, where a is the total propensity of all subvolumes and r a uniformly distributed random number in [0, 1).

   2.2. Use composition-rejection sampling to determine the subvolume (l, m, n) containing the next reaction. Use linear search (Eq. (12)) in the composition step to locate the bin containing $a^{(l,m,n)}$ and use the rejection step to locate $a^{(l,m,n)}$ inside that bin.

   2.3. Sample the next reaction μ in subvolume (l, m, n) by sampling its group and element indices. Sample the group index I using linear search over $\mathbf{\Sigma}^{(l,m,n)}$ (Eq. (13)). Subsequently, sample the element index J using linear search over $\mathbf{\Pi}_I^{(l,m,n)}$ (Eq. (14)). The reaction index μ is then identified from the look-up table as $\mu = \mathrm{L}_{I,J}$.

   2.4. Update the internal data structures in subvolume (l, m, n) and the total propensity a using $\mathbf{U}^{(1)}$, $\mathbf{U}^{(2)}$, and $\mathbf{U}^{(3)}$.

   2.5. Increase the number of bins G_a if necessary and update the bin membership of $a^{(l,m,n)}$.

   2.6. If $\mu > M$ (i.e., the sampled reaction is a lumped "diffusion reaction,") compute the index of the diffusing species as $i = \mu - M$. Resolve the diffusion event to identify the neighboring target subvolume (l', m', n') to which one molecule of species S_i from subvolume (l, m, n) is diffusing. Increment the population of species S_i in the target subvolume. Subsequently, update the other internal data structures of the target subvolume and the total propensity a using $\mathbf{U}_i^{(3)}$. Increase the number of bins G_a if necessary and update the bin membership of $a^{(l',m',n')}$.

   2.7. Advance time: $t \leftarrow t + \tau$.

3. Stop.

"diffusion jump" is resolved when a lumped "diffusion reaction" has been selected.

The reaction indices of the partial propensities in $\mathbf{\Pi}^{(l,m,n)}$ are stored in a look-up table $\mathbf{L} = \{\mathbf{L}_i\}_{i=0}^{N}$, which is also an array of arrays. For subvolumes containing the same reaction network, we store the look-up table only once. In case the reaction network is the same in all subvolumes, only a single, global look-up table is needed. Subvolumes that host different reaction networks have different look-up tables. The look-up table renders every reaction within each subvolume identifiable by a unique pair of indices, a group index I and an element index J, such that the partial propensity of reaction $\mu = \mathrm{L}_{I,J}$ is stored in $\Pi_{I,J}^{(l,m,n)}$ for subvolume (l, m, n).

The "group-sum array" $\mathbf{\Lambda}^{(l,m,n)}$ stores the sums of the partial propensities in each group $\mathbf{\Pi}_i^{(l,m,n)}$, i.e., $\Lambda_i^{(l,m,n)} = \sum_j \Pi_{i,j}^{(l,m,n)}$. We also store the total propensity of each group in an array $\mathbf{\Sigma}$, computed as $\Sigma_i^{(l,m,n)} = n_i^{(l,m,n)}\Lambda_i^{(l,m,n)}$, $i = 1, \ldots, N$, and $\Sigma_0^{(l,m,n)} = \Lambda_0^{(l,m,n)}$.[41] The total propensity of the reactions in subvolume (l, m, n) is then $a^{(l,m,n)} = \sum_{i=0}^{N} \Sigma_i^{(l,m,n)}$. The total propensity of all reactions across all subvolumes is stored in $a = \sum_{l=1}^{K_x}\sum_{m=1}^{K_y}\sum_{n=1}^{K_z} a^{(l,m,n)}$ and is used to calculate the time to the next reaction according to Eq. (9) as $\tau = -a^{-1}\log r_0$, where r_0 is a uniformly distributed random number in [0, 1).

2. Algorithms

In PSRD, like in NSM, the subvolume containing the next reaction is sampled first. To this end, the total propensities $a^{(l,\,m,\,n)}$ of all subvolumes are sorted into $G_{\mathrm{a}} = \log_2(a_{\max}/a_{\min}) + 1$ bins, such that bin b contains all $a^{(l,\,m,\,n)}$ in the interval $2^{b-1}a_{\min} \leq a^{(l,\,m,\,n)} < 2^b a_{\min}$. The bounds $a_{\min}$ and $a_{\max}$ are the smallest non-zero and the largest value that any of the $a^{(l,\,m,\,n)}$ can assume during the simulation. They are determined as follows: The lower bound $a_{\min}$ is the minimum propensity of any reaction in any subvolume when the number of molecules of all reactants is one (minimum non-zero population). For elementary reactions, this is the smallest specific probability rate across all subvolumes. The largest possible value of $a^{(l,\,m,\,n)}$ may be ascertained using physical reasoning or prior knowledge about the reaction-diffusion system. In cases where this cannot be evaluated *a priori*, PSRD initially sets $a_{\max}$ to the maximum $a^{(l,\,m,\,n)}$. If during the course of the simulation the maximum $a^{(l,\,m,\,n)}$ increases, PSRD updates $a_{\max}$ and G_{a}, and the corresponding data structures are dynamically enlarged.

PSRD uses composition-rejection sampling to determine the subvolume of the next reaction in two steps: (i) composition step to find the bin b and (ii) rejection step to find $a^{(l,\,m,\,n)}$ inside that bin. The composition step uses linear search to determine

$$b = \min\left[b' \;:\; r_1 a < \sum_{i=1}^{b'} \alpha_i\right], \tag{12}$$

where r_1 is a uniform random number in $[0, 1)$ and α_i is the total propensity in bin i computed by summing up the $a^{(l,\,m,\,n)}$ in that bin. The rejection step samples the subvolume (l, m, n) among the entries in the selected bin b. For this, we first generate a uniformly distributed random number r_2 in $[0, 2^b a_{\min})$ and a uniformly distributed random integer r_3 between 1 and the number of entries in bin b. If the r_3-th element in bin b is greater than or equal to r_2, the corresponding subvolume is selected. If the inequality is not satisfied, the rejection step is repeated. This procedure is illustrated in Fig. 2 for a computational domain divided into 4 subvolumes with indices (1, 1, 1), (2, 1, 1), (1, 2, 1) and (2, 2, 1). Assume that the composition step has chosen bin 1 as the bin containing the next subvolume. The rejection step then samples uniformly random points inside the rectangle defining the range of this bin (bold rectangle). A sample is accepted if it falls inside one of the shaded bars representing the $a^{(l,\,m,\,n)}$'s. If the first sample (point A in Fig. 2 with $r_3 = 2$ and $r_2 > a^{(1,\,2,\,1)}$) is rejected, sampling is repeated until the point falls inside one of the shaded bars (point B in Fig. 2 with $r_3 = 1$ and $r_2 < a^{(2,\,1,\,1)}$). By binning the $a^{(l,\,m,\,n)}$ as described above, it is guaranteed that the area covered by the $a^{(l,\,m,\,n)}$ bars in each bin is at least 50% of the bin's total area. This ensures that the expected number of rejection steps required is ≤ 2. The probability of needing more than k rejection steps is $\leq 2^{-k}$ and hence exponentially small.

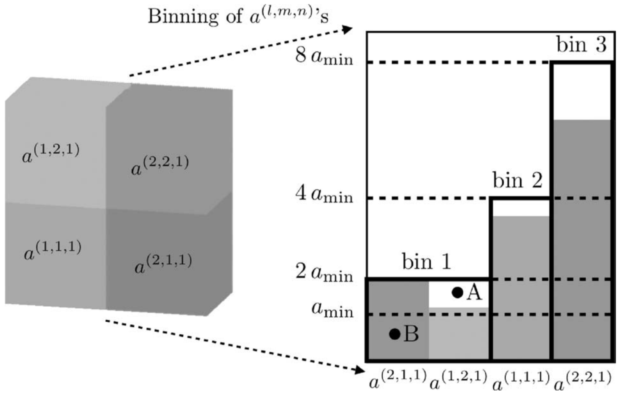

FIG. 2. Illustration of the binning of the total propensities of the subvolumes used for composition-rejection sampling of the next subvolume. The illustration shows a computational domain divided into 4 subvolumes. Points A and B refer to the example in main text used to explain rejection sampling.

Once the subvolume (l, m, n) containing the next reaction has been chosen, PSRD samples the index of the next reaction μ within that subvolume in two steps: (i) perform linear search for the group index I such that

$$I = \min\left[I' \;:\; r_4 a^{(l,m,n)} < \sum_{i=0}^{I'} \Sigma_i^{(l,m,n)}\right] \tag{13}$$

and (ii) perform linear search for the element index J inside group $\mathbf{\Pi}_I^{(l,m,n)}$ such that

$$J = \min\left[J' \;:\; r_4 a^{(l,m,n)} < \sum_{j=1}^{J'} n_I^{(l,m,n)}\Pi_{I,j}^{(l,m,n)} + \left(\sum_{i=0}^{I}\Sigma_i^{(l,m,n)}\right) - \Sigma_I^{(l,m,n)}\right], \tag{14}$$

where r_4 is a uniform random number in [0,1). The indices I and J are then translated to the reaction index μ in subvolume (l, m, n) using the look-up table $\mathbf{L}$, thus $\mu = \mathrm{L}_{I,J}$. It has earlier been shown that sampling the reaction according to Eqs. (13) and (14) is algebraically equivalent to the linear search over propensities used in Gillespie's original direct method,[41] i.e., for the same random number stream the two methods produce the exact same sequence of reaction firings provided the reactions are indexed in the same order in both methods. In order to reduce the average search depth, the group and element indices are dynamically rearranged such that frequent reactions accumulate at the beginning of the list, i.e., have low index values. This is done by dynamically bubbling up a reaction whenever it fires by performing a single iteration of a bubble-sort algorithm. The permutation lists for the reordered indices in each subvolume are stored in an array for the I's, and one-dimensional array of one-dimensional arrays of the size of $\mathbf{\Pi}^{(l,m,n)}$ for the J's. PSRD thus uses the sorting partial-propensity direct method (SPDM) to sample the next reaction within a subvolume. This renders the sampling procedure more efficient (in the sense that it reduces the prefactor in the scaling of the computational cost) when the reaction network in a subvolume is multi-scale (stiff), without compromising on the efficiency in non-stiff cases.[41] In SRD simulations the reaction networks inside the subvolumes tend to be stiff since the specific probability rates of bimolecular reactions scale as h^{-3} (in 3D subvolumes) whereas those of source reactions scale as h^3 (see Eq. (6)). Using SPDM instead of PDM may hence lead to significant computational savings.

Once a reaction has been executed, $\mathbf{n}^{(l,m,n)}$, $\mathbf{\Pi}^{(l,m,n)}$, $\mathbf{\Lambda}^{(l,m,n)}$, and $\mathbf{\Sigma}^{(l,m,n)}$ need to be updated. This is efficiently done using three update structures. If the reaction network is the same in each subvolume, the same update structures can be used for all subvolumes and they do not have to be stored separately for different subvolumes. Subvolumes containing different reaction networks have different update structures.

$\mathbf{U}^{(1)}$ is an array of M arrays, where the i^{th} array contains the indices of all species involved in the i^{th} "real" reaction. The index of the species involved in the i^{th} lumped "diffusion reaction" does not need to be stored as it is simply i itself.

$\mathbf{U}^{(2)}$ is a array of M arrays containing the corresponding stoichiometries (the change in population of each species upon reaction) of the species stored in $\mathbf{U}^{(1)}$. The stoichiometries of the "diffusion reactions" are not stored since they are all -1.

$\mathbf{U}^{(3)}$ is a array of N arrays, where the i^{th} array contains the indices of all entries in the $\mathbf{\Pi}^{(l,m,n)}$'s that depend on $n_i^{(l,m,n)}$.

When a reaction is executed in subvolume (l, m, n), the populations of the species involved in this reaction change. Hence, all entries in $\mathbf{\Pi}^{(l,m,n)}$ that depend on these populations need to be updated. After each reaction, we use $\mathbf{U}^{(1)}$ to determine the indices of all species involved in this reaction. The stoichiometry is then looked up in $\mathbf{U}^{(2)}$ and the population $\mathbf{n}^{(l,m,n)}$ is updated. Subsequently, $\mathbf{U}^{(3)}$ is used to locate the affected entries in $\mathbf{\Pi}^{(l,m,n)}$ and recompute them. The two data structures $\mathbf{U}^{(1)}$ and $\mathbf{U}^{(2)}$ hence amount to a sparse representation of the stoichiometry matrix; $\mathbf{U}^{(3)}$ represents the dependency graph over species. Since the partial propensities of unimolecular and source reactions are constant and never need to be updated, $\mathbf{U}^{(3)}$ only contains the indices of the partial propensities of bimolecular reactions. Along with updating the partial propensities in subvolume (l, m, n), the change in the total propensity of that subvolume is also calculated and incrementally applied to $a^{(l,m,n)}$. This may require the bin membership of $a^{(l,m,n)}$ to be updated, for which the current bin assignment of $a^{(l,m,n)}$ must be known. We implement this by storing two additional integers for every $a^{(l,m,n)}$: one for the bin membership and the other for the location inside that bin. Depending on its new value, $a^{(l,m,n)}$ may remain in the same bin or move to a different one. Removal of an element from a bin is done by replacing it with the last element in that bin and reducing the bin size by one. Addition of an element into a bin is done by appending it at end of the bin. The computational cost of both of these operations is $O(1)$.[39,42]

If the index of the next reaction is greater than M, then the sampled reaction is a lumped "diffusion reaction" and additional steps need to be taken to resolve the direction of the jump as follows: First, the index of the species undergoing diffusion is computed as $i = \mu - M$. Second, a uniform random number between 0 and the lumped specific probability rate of the lumped "diffusion reaction" is generated. Third, linear search over the specific probability rates of individual directional diffusion events is used to determine the target subvolume of the jump. The jump is executed by increasing the population of species S_i in the target subvolume by 1 (the reduction in the source subvolume has already been done above) and updating the entries in the partial propensity structure of the target subvolume as given by the indices in $\mathbf{U}_i^{(3)}$. Finally, the total propensity of the target subvolume and its bin membership are updated.

Figure 3 summarizes the data structures used in PSRD for the example reaction network given in Eq. (1). The complete algorithm is given in Table I. The computational cost of PSRD to sample the subvolume is $O(1)$ if the ratio of maximum to minimum non-zero total propensity in each subvolume is independent of the number of subvolumes and of the size of the reaction network. In cases where this ratio is not bounded by a constant, the computational cost to sample the subvolume is $O(G_{\mathrm{a}})$, where the total number of bins G_{a} depends on the logarithmic span of the subvolume propensities as $G_{\mathrm{a}} = \log_2(a_{\max}/a_{\min}) + 1$. The computational cost of sampling the index of the next reaction within a subvolume is $O(N)$.[41] The overall computational cost of PSRD hence is $O(G_{\mathrm{a}} + N)$, which is at most $O(\log_2 N_{\mathrm{v}} + N)$. The memory requirement of PSRD is $O((M+N)N_{\mathrm{v}})$. For more details on the computational cost and the memory requirement, see Appendix.

IV. BENCHMARKS

We analyze the computational cost of PSRD as quantified by the average simulation (CPU) time Θ taken per reaction

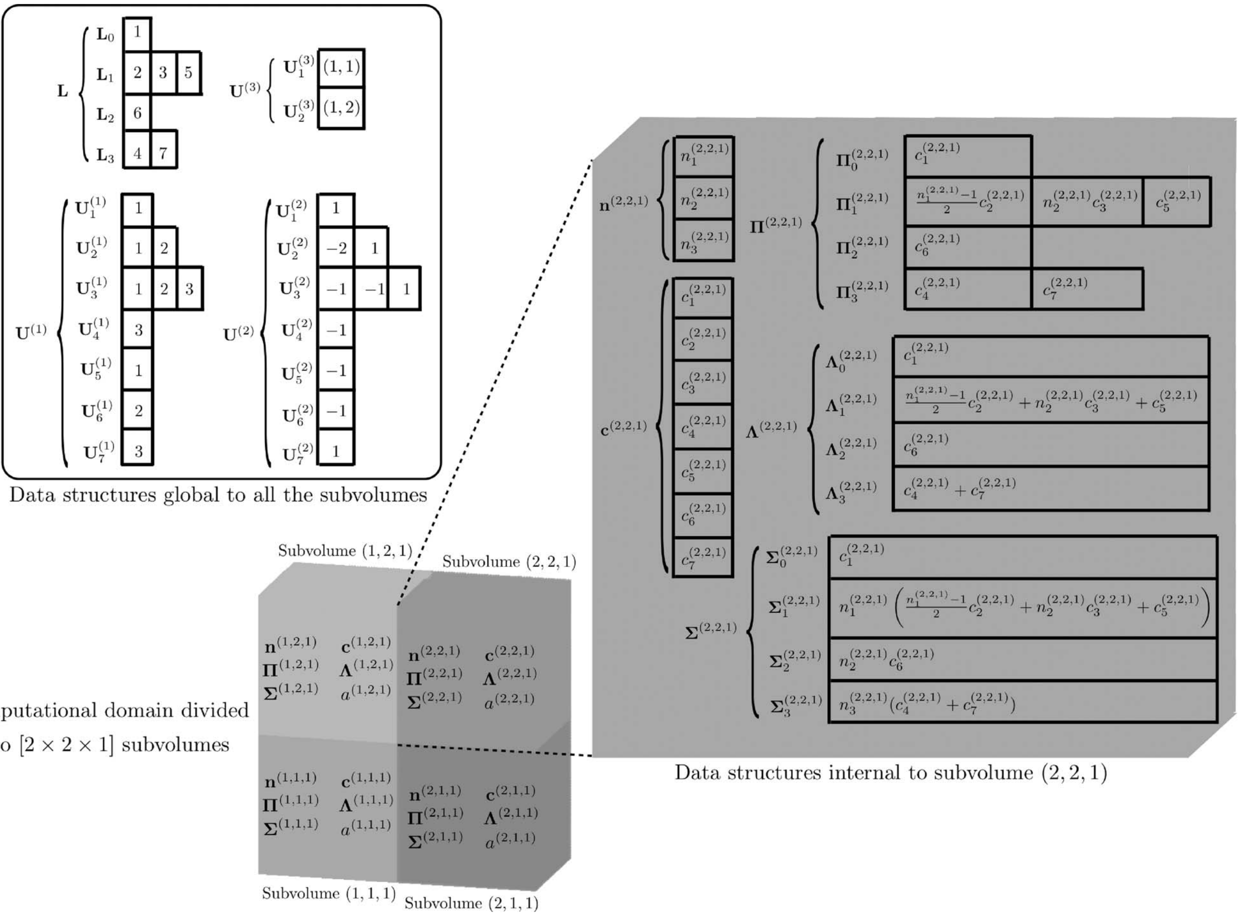
L
$\mathbf{L}_0$ 1
$\mathbf{L}_1$ 2 3 5
$\mathbf{L}_2$ 6
$\mathbf{L}_3$ 4 7
$\mathbf{U}^{(3)}$
$\mathbf{U}_1^{(3)}$ (1, 1)
$\mathbf{U}_2^{(3)}$ (1, 2)
$\mathbf{U}^{(1)}$
$\mathbf{U}_1^{(1)}$ 1
$\mathbf{U}_2^{(1)}$ 1 2
$\mathbf{U}_3^{(1)}$ 1 2 3
$\mathbf{U}_4^{(1)}$ 3
$\mathbf{U}_5^{(1)}$ 1
$\mathbf{U}_6^{(1)}$ 2
$\mathbf{U}_7^{(1)}$ 3
$\mathbf{U}^{(2)}$
$\mathbf{U}_1^{(2)}$ 1
$\mathbf{U}_2^{(2)}$ −2 1
$\mathbf{U}_3^{(2)}$ −1 −1 1
$\mathbf{U}_4^{(2)}$ −1
$\mathbf{U}_5^{(2)}$ −1
$\mathbf{U}_6^{(2)}$ −1
$\mathbf{U}_7^{(2)}$ 1
Data structures global to all the subvolumes
Subvolume (1, 2, 1)
Subvolume (2, 2, 1)
$\mathbf{n}^{(1,2,1)}$ $\mathbf{c}^{(1,2,1)}$ $\mathbf{\Pi}^{(1,2,1)}$ $\mathbf{\Lambda}^{(1,2,1)}$ $\mathbf{\Sigma}^{(1,2,1)}$ $a^{(1,2,1)}$
$\mathbf{n}^{(2,2,1)}$ $\mathbf{c}^{(2,2,1)}$ $\mathbf{\Pi}^{(2,2,1)}$ $\mathbf{\Lambda}^{(2,2,1)}$ $\mathbf{\Sigma}^{(2,2,1)}$ $a^{(2,2,1)}$
$\mathbf{n}^{(1,1,1)}$ $\mathbf{c}^{(1,1,1)}$ $\mathbf{\Pi}^{(1,1,1)}$ $\mathbf{\Lambda}^{(1,1,1)}$ $\mathbf{\Sigma}^{(1,1,1)}$ $a^{(1,1,1)}$
$\mathbf{n}^{(2,1,1)}$ $\mathbf{c}^{(2,1,1)}$ $\mathbf{\Pi}^{(2,1,1)}$ $\mathbf{\Lambda}^{(2,1,1)}$ $\mathbf{\Sigma}^{(2,1,1)}$ $a^{(2,1,1)}$
Computational domain divided into [2 × 2 × 1] subvolumes
Subvolume (1, 1, 1)
Subvolume (2, 1, 1)
$\mathbf{n}^{(2,2,1)}$: $n_1^{(2,2,1)}$ $n_2^{(2,2,1)}$ $n_3^{(2,2,1)}$
$\mathbf{c}^{(2,2,1)}$: $c_1^{(2,2,1)}$ $c_2^{(2,2,1)}$ $c_3^{(2,2,1)}$ $c_4^{(2,2,1)}$ $c_5^{(2,2,1)}$ $c_6^{(2,2,1)}$ $c_7^{(2,2,1)}$
$\mathbf{\Pi}^{(2,2,1)}$
$\mathbf{\Pi}_0^{(2,2,1)}$ $c_1^{(2,2,1)}$
$\mathbf{\Pi}_1^{(2,2,1)}$ $\frac{n_1^{(2,2,1)}-1}{2}c_2^{(2,2,1)}$ $n_2^{(2,2,1)}c_3^{(2,2,1)}$ $c_5^{(2,2,1)}$
$\mathbf{\Pi}_2^{(2,2,1)}$ $c_6^{(2,2,1)}$
$\mathbf{\Pi}_3^{(2,2,1)}$ $c_4^{(2,2,1)}$ $c_7^{(2,2,1)}$
$\mathbf{\Lambda}^{(2,2,1)}$
$\mathbf{\Lambda}_0^{(2,2,1)}$ $c_1^{(2,2,1)}$
$\mathbf{\Lambda}_1^{(2,2,1)}$ $\frac{n_1^{(2,2,1)}-1}{2}c_2^{(2,2,1)} + n_2^{(2,2,1)}c_3^{(2,2,1)} + c_5^{(2,2,1)}$
$\mathbf{\Lambda}_2^{(2,2,1)}$ $c_6^{(2,2,1)}$
$\mathbf{\Lambda}_3^{(2,2,1)}$ $c_4^{(2,2,1)} + c_7^{(2,2,1)}$
$\mathbf{\Sigma}^{(2,2,1)}$
$\mathbf{\Sigma}_0^{(2,2,1)}$ $c_1^{(2,2,1)}$
$\mathbf{\Sigma}_1^{(2,2,1)}$ $n_1^{(2,2,1)}\left(\frac{n_1^{(2,2,1)}-1}{2}c_2^{(2,2,1)} + n_2^{(2,2,1)}c_3^{(2,2,1)} + c_5^{(2,2,1)}\right)$
$\mathbf{\Sigma}_2^{(2,2,1)}$ $n_2^{(2,2,1)}c_6^{(2,2,1)}$
$\mathbf{\Sigma}_3^{(2,2,1)}$ $n_3^{(2,2,1)}(c_4^{(2,2,1)} + c_7^{(2,2,1)})$
Data structures internal to subvolume (2, 2, 1)

FIG. 3. The data structures in PSRD. The contents of the data structures shown corresponds to the example reaction network in Eq. (1) with 3 species and 4 reactions. We assume that the computational domain is divided into 4 subvolumes. In the illustration, $c_5^{(2,2,1)}$, $c_6^{(2,2,1)}$, and $c_7^{(2,2,1)}$ are the lumped specific probability rates of the "diffusion reactions" of species 1, 2, and 3 respectively. See main text for details.

event of the chemical reaction model of a reaction-diffusion system. We compare it to the time expected from the theoretical cost analysis (see Sec. III B) for two different types of reaction networks: (i) an aggregation model where the number of reactions increases super-linearly with the number of species and (ii) a linear chain model where the number of reactions is almost the same as the number of species. We simulate the corresponding reaction-diffusion processes in a three-dimensional cubic computational domain (reactor) of dimensions $L \times L \times L$ and volume $\Omega = L^3$ from a initial time $t = 0$ until a final time $t = t_{\mathrm{f}}$. For simulating the reaction-diffusion process we divide the computational domain into $N_{\mathrm{v}} = K^3$ equi-sized cubic subvolumes of edge length $h = L/K$, such that K is the number of subvolumes along each spatial dimension.

For each of these networks we report Θ as a function of the number of subvolumes N_{v} for a fixed size of the reaction network and as a function of the reaction network size for a fixed number of subvolumes. We use the number of species N in the reaction network to quantify the size of the network. All timings are compared to those obtained on the same systems and the same computer using NSM.

Both PSRD and NSM were implemented in C++ using the random number generator of the GSL library and compiled using the Intel C++ compiler version 12.0.2 with the O3 optimization flag. NSM is implemented according to the details provided on the MesoRD webpage (Algorithm 7 in Ref. 60). All timings were measured on a Linux 2.6 workstation with a 2.8 GHz quad-core Intel Xeon E5462 processor, 8 GB of memory and 4 MB L2 cache. For all test cases, we simulate until a final time t_{f} and report the average CPU time Θ per reaction event. The time Θ does not include the initialization of the data structures as this is done only once. We explain the measurements by least-squares fits of $\Theta(N_{\mathrm{v}}, N)$ with the corresponding theoretical cost models. For PSRD and NSM, we hence fit Θ with $\alpha_1 \log_2 N_{\mathrm{v}} + \alpha_2 N$ and $\alpha_1 \log_2 N_{\mathrm{v}} + \alpha_2 f_{\mathrm{r}} M + \alpha_3 (1 - f_{\mathrm{r}}) 6N$, respectively. Before fitting, we estimate the functional dependence of f_{r} on N_{v} or N by performing simulations. Subsequently, we fit Θ to determine the coefficients α_i.

All simulations are run without any *a priori* estimate of the maximum total propensity $a_{\max}$ across all subvolumes. Instead, $a_{\max}$ is constantly updated during a simulation and the number of bins G_{a} is dynamically increased when required (see Sec. III B 2).

A. Colloidal aggregation model

We consider the non-equilibrium colloidal aggregation model as a prototype of a strongly coupled reaction network in which the number of reactions increases super-linearly with the number of species:

$$\begin{array}{cccl} \emptyset & \xrightarrow{k_0} & \mathrm{S}_1 & \\ \mathrm{S}_i + \mathrm{S}_j & \xrightarrow{k_{i,j}} & \mathrm{S}_{i+j} & i+j = 2,\ldots,N \\ \mathrm{S}_{i+j} & \xrightarrow{\bar{k}_{i,j}} & \mathrm{S}_i + \mathrm{S}_j & i+j = 2,\ldots,N \\ \mathrm{S}_i & \xrightarrow{k_i} & \emptyset & i = 1,\ldots,N. \end{array} \tag{15}$$

The k's are the macroscopic reaction rates. This system of reactions models the influx of monomers (S_1) into a reactor where multimers (S_i) fuse with each other to form larger multimers. Multimers in the reactor also break to form smaller units in all possible combinations, and all of the multimers can leave the reactor. Such a system of reactions models driven colloidal aggregation and is relevant for a variety of phenomena of practical importance, e.g., nano-particle clustering and colloidal crystallization (nanotechnology), emulsification and emulsion stabilization in porous media (oil industry), and oligomerization of proteins (biochemistry). For N chemical species, the aggregation reaction network consists of $M = \lfloor N^2/2 \rfloor + N + 1$ reactions.

We divide the cubic computational domain (reactor) into $N_{\mathrm{v}} = K^3$ subvolumes, such that the on-lattice reaction-diffusion process with reflective boundary conditions is described by the following set of "reactions":

$$\left.\begin{array}{cccc} \emptyset & \xrightarrow{k_0} & \mathrm{S}_1^{(l,m,n)} & \\ \mathrm{S}_i^{(l,m,n)} + \mathrm{S}_j^{(l,m,n)} & \xrightarrow{k_{i,j}} & \mathrm{S}_{i+j}^{(l,m,n)} & i+j = 2,\ldots,N \\ \mathrm{S}_{i+j}^{(l,m,n)} & \xrightarrow{\bar{k}_{i,j}} & \mathrm{S}_i^{(l,m,n)} + \mathrm{S}_j^{(l,m,n)} & i+j = 2,\ldots,N \\ \mathrm{S}_i^{(l,m,n)} & \xrightarrow{k_i} & \emptyset & i = 1,\ldots,N \\ \mathrm{S}_i^{(l,m,n)} & \xrightarrow{D_i/h^2} & \mathrm{S}_i^{\mathbf{c}} & i = 1,\ldots,N \end{array}\right\} \quad \forall\ (l,m,n) \in I_{\mathrm{all}}, \quad \mathbf{c} \in C_{l,m,n}, \tag{16}$$

where D_i is the diffusion constant of species S_i and h the edge length of the cubic subvolumes. The propensities of these reactions are computed as described in Sec. II. For all bimolecular reactions, we use discretization-corrected propensities.[50] The above network consists of $NK^3 = NN_{\mathrm{v}}$ species and $MK^3 + N(6K^3 - 6K^2) = (\lfloor N^2/2 \rfloor + 1)N_{\mathrm{v}} + N(7N_{\mathrm{v}} - 6N_{\mathrm{v}}^{2/3})$ reactions.

For the present benchmarks, we set the macroscopic reaction rates $k_{i,\,i} = 0.5$, all other rates and all diffusion constants to 1, and the reactor volume to $\Omega = 10$. At time $t = 0$, the populations of all species in all subvolumes, i.e., all $n_i^{(l,m,n)}$, are set to 0. From this initial condition we simulate the reaction-diffusion system until $t_{\mathrm{f}} = 100$.

Figure 4(a) shows the computational cost Θ as a function of the number of subvolumes N_{v} using PSRD and NSM for two fixed-size aggregation networks with $N = 10$ and $N = 100$, respectively. The corresponding numbers of reactions M are 61 and 5101, respectively. In both cases we estimate f_{r} and use it for fitting Θ. We observe that f_{r} decreases as $N_{\mathrm{v}}^{-0.34}$ with increasing N_{v}. For PSRD, $\Theta(N_{\mathrm{v}}, N = 10) \approx 0.02861 \log N_{\mathrm{v}}$ at large N_{v}. This scaling of Θ is caused by the dynamic increase in the number of bins G_{a}. For NSM, $\Theta(N_{\mathrm{v}}, N = 10) \approx 0.1095 \log N_{\mathrm{v}}$ at large N_{v}. For the larger network with $N = 100$, we find for PSRD $\Theta(N_{\mathrm{v}}, N = 100) \approx 0.04401 \log N_{\mathrm{v}}$ at large N_{v}. For NSM, $\Theta(N_{\mathrm{v}}, N = 100) \approx 0.288 \log N_{\mathrm{v}}$ at large N_{v}. For smaller $1 \leq N_{\mathrm{v}} < 512$, Θ of NSM decreases with increasing N_{v}. This is mediated by the decrease in f_{r}. At $N_{\mathrm{v}} = 1$, $f_{\mathrm{r}} = 1$ and the cost is dominated by that of sampling the next "real" reaction. As N_{v} increases, f_{r} decreases. This decrease in f_{r} progressively reduces the cost of sampling a reaction in a subvolume from being linear in M to linear in $6N$. At large-enough N_{v}, the cost of sampling a reaction in a subvolume is dominated by the cost of sampling "diffusion reactions." For a fixed network size, the increase in $\Theta(N_{\mathrm{v}}, N = 100)$ at large N_{v} is thus primarily due to the increasing cost to sample the next subvolume. In summary, the scaling of the computational cost of PSRD with respect to the number of subvolumes N_{v} is $O(\log_2 N_{\mathrm{v}})$. This scaling is asymptotically (for large N_{v}) the same as that of NSM, but with a smaller prefactor.

Figure 4(b) shows the computational cost Θ as a function of the size N of the aggregation reaction network using PSRD and NSM with $N_{\mathrm{v}} = 512$ and $N_{\mathrm{v}} = 1000$ subvolumes. We observe that for both N_{v} the ratio f_{r} does not depend on the size N of network. For $N_{\mathrm{v}} = 512$, $f_{\mathrm{r}} = 0.04$, decreasing to $f_{\mathrm{r}} = 0.02$ for $N_{\mathrm{v}} = 1000$. For PSRD, $\Theta(N_{\mathrm{v}} = 512, N) \approx 0.002258\,N$, confirming the linear dependence on N predicted by the theoretical cost analysis. For NSM, $\Theta(N_{\mathrm{v}} = 1000, N) \approx 0.00011M + 0.0152N$. For the larger number subvolumes, $N_{\mathrm{v}} = 1000$, $\Theta(N_{\mathrm{v}} = 1000, N) \approx 0.002777N$ for PSRD. For NSM, $\Theta(N_{\mathrm{v}} = 1000, N) \approx 0.000055M + 0.0186N$. In summary, the scaling of the computational cost of PSRD with respect to the size N of the reaction network is $O(N)$.

B. Linear chain model

As a prototypical reaction network in which the number of reactions is almost the same as the number of species, we consider the non-equilibrium linear chain model:

$$\begin{array}{cccl} \emptyset & \xrightarrow{k_0} & \mathrm{S}_1 & \\ \mathrm{S}_i & \xrightarrow{k_i} & \mathrm{S}_{i+1} & i = 1,\ldots,N-1 \\ \mathrm{S}_N & \xrightarrow{k_N} & \emptyset. & \end{array} \tag{17}$$

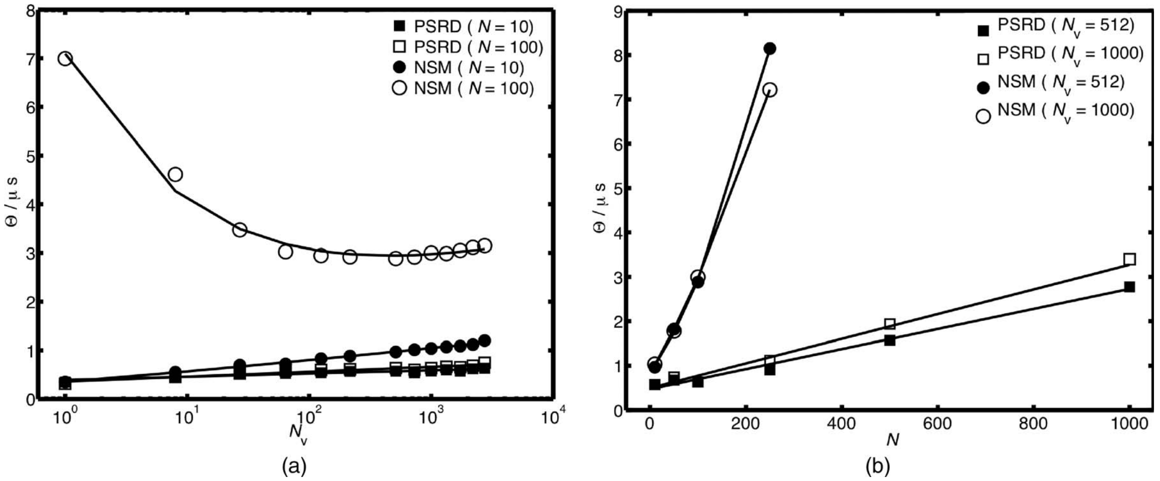

FIG. 4. Computational cost of PSRD and NSM for the aggregation model (Eq. (15)). (A) Computational cost Θ of PSRD (squares) and NSM (circles) as a function of the number of subvolumes N_v with the size of the reaction network fixed to $N = 10$ (filled symbols) and $N = 100$ (empty symbols), respectively. The solid lines show the corresponding least-squares fits of the theoretical cost models: For $N = 10$, $\Theta^{\mathrm{PSRD}} \approx 0.02861 \log N_v + 0.03925\, N$, $\Theta^{\mathrm{NSM}} \approx 0.1095 \log N_v + 0.00581\, f_r M + 0.00481(1 - f_r)6N$; for $N = 100$, $\Theta^{\mathrm{PSRD}} \approx 0.04401 \log N_v + 0.003579\, N$, $\Theta^{\mathrm{NSM}} \approx 0.288 \log N_v + 0.001375\, f_r M + 0.001418(1 - f_r)6N$. We estimate $f_r \approx 1.096 N_v^{-0.3353} - 0.08263$ for $N = 10$ and $f_r = 1.097 N_v^{-0.3372} - 0.0825$ for $N = 100$. (B) Computational cost Θ of PSRD (squares) and NSM (circles) as a function of the number of species N in the reaction network with the number of subvolumes fixed to $N_v = 512$ (filled symbols) and $N_v = 1000$ (empty symbols), respectively. The solid lines show the corresponding least-squares fits of the theoretical cost models: For $N_v = 512$, $\Theta^{\mathrm{PSRD}} \approx 0.07559 \log N_v + 0.002258\, N$, $\Theta^{\mathrm{NSM}} \approx 0.1356 \log N_v + 0.002784\, f_r(\lfloor N^2/2 \rfloor + N + 1) + 0.002633(1 - f_r)6N$; for $N_v = 1000$, $\Theta^{\mathrm{PSRD}} \approx 0.07205 \log N_v + 0.002777\, N$, $\Theta^{\mathrm{NSM}} \approx 0.1198 \log N_v + 0.002762\, f_r(\lfloor N^2/2 \rfloor + N + 1) + 0.003163(1 - f_r)6N$. The fraction $f_r = 0.04$ for $N_v = 512$ and $f_r = 0.02$ for $N_v = 1000$.

Again, the k's are the macroscopic reaction rates. This linear chain of reactions can, e.g., be used to model signal transduction pathways in biological cells.[61,62] For N species, this network contains $M = N + 1$ reactions.

Again dividing the cubic computational domain into $N_v = K^3$ subvolumes, the resulting reaction-diffusion system with reflective boundary conditions is given by:

$$\left.\begin{array}{llll} \emptyset & \xrightarrow{k_0} & \mathrm{S}_1^{(l,m,n)} & \\ \mathrm{S}_i^{(l,m,n)} & \xrightarrow{k_i} & \mathrm{S}_{i+1}^{(l,m,n)} & i = 1, \ldots, N-1 \\ \mathrm{S}_N^{(l,m,n)} & \xrightarrow{k_N} & \emptyset & \\ \mathrm{S}_i^{(l,m,n)} & \xrightarrow{D_i/h^2} & \mathrm{S}_i^{\mathbf{c}} & i = 1, \ldots, N \end{array}\right\} \forall\ (l, m, n) \in I_{\mathrm{all}},\ \mathbf{c} \in C_{l,m,n}, \qquad (18)$$

where D_i is the diffusion constant of species S_i and h is the edge length of the cubic subvolumes. The propensities of these reactions are computed as described in Sec. II. There are no bimolecular reactions in this network, and we do not use discretization-corrected propensities.[50] This system of reactions modeling the reaction-diffusion process contains $NK^3 = NN_v$ species and $MK^3 + N(6K^3 - 6K^2) = (7N + 1)N_v - 6NN_v^{2/3}$ reactions.

For the benchmarks we set all macroscopic reactions rates and all diffusion constants D_i to 1, and the volume of the reactor to $\Omega = 100$. At time $t = 0$, the populations of all species in all subvolumes are 0, and the simulation is run until $t_f = 100$.

Figure 5(a) shows the computational cost Θ as a function of the number of subvolumes N_v using PSRD and NSM for two fixed-size linear chain networks with $N = 10$ and $N = 100$. The corresponding numbers of reactions M are 11 and 101, respectively. In both cases we estimate f_r and use it for fitting Θ. We observe that f_r decreases as $N_v^{-0.22}$ with increasing N_v. For PSRD, $\Theta(N_v, N = 10) \approx 0.03312 \log N_v$. This scaling of Θ is caused by the increase in the number of bins G_a. For NSM, $\Theta(N_v, N = 10) \approx 0.08256 \log N_v$. For the larger network with $N = 100$, the computational cost of PSRD is $\Theta(N_v, N = 100) \approx 0.04842 \log N_v$ for $N_v \lessapprox 512$ and $\Theta(N_v, N = 100) \approx 0.2923 \log N_v$ for $N_v \gtrapprox 512$. For NSM, $\Theta(N_v, N = 100) \approx 0.1428$

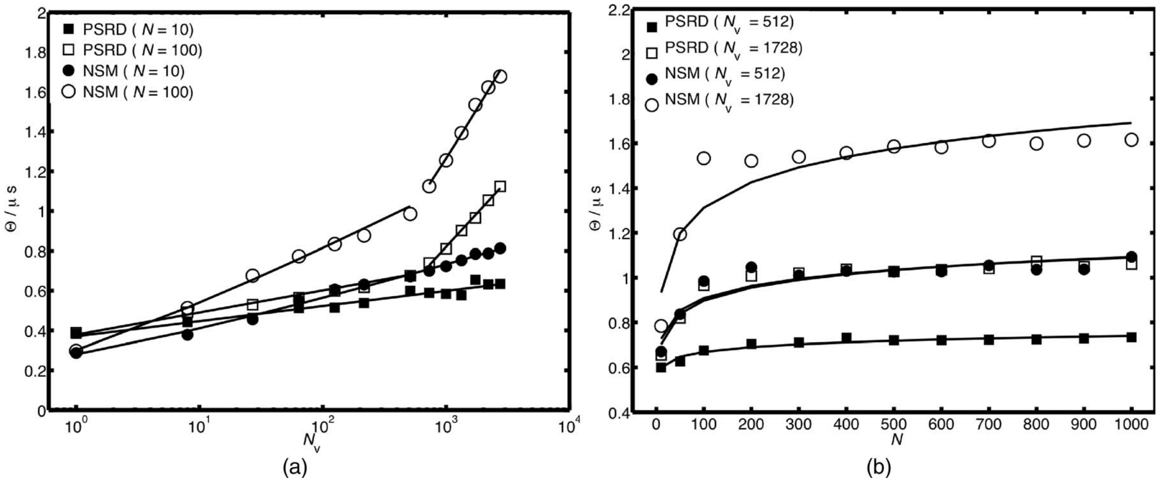

FIG. 5. Computational cost of PSRD and NSM for the linear chain model (Eq. (17)). (A) Computational cost Θ of PSRD (squares) and NSM (circles) as a function of the number of subvolumes N_v with the size of the reaction network fixed to $N = 10$ (filled symbols) and $N = 100$ (empty symbols), respectively. The solid lines show the corresponding least-squares fits of the theoretical cost models: For $N = 10$, $\Theta^{\mathrm{PSRD}} \approx 0.03312 \log N_\mathrm{v} + 0.03703N$, $\Theta^{\mathrm{NSM}} \approx 0.08256 \log N_\mathrm{v} + 0.02504 f_\mathrm{r}M + 0.002615(1 - f_\mathrm{r})6N$; for $N = 100$, $\Theta^{\mathrm{PSRD}} \approx 0.04842 \log N_\mathrm{v} + 0.003786N$, $\Theta^{\mathrm{NSM}} \approx 0.1428 \log N_\mathrm{v} + 0.002934 f_\mathrm{r}M + 0.0001978(1 - f_\mathrm{r})6N$ for $N_\mathrm{v} \lessapprox 512$ and $\Theta^{\mathrm{PSRD}} \approx 0.2923 \log N_\mathrm{v} - 0.01199N$, $\Theta^{\mathrm{NSM}} \approx 0.5929 \log N_\mathrm{v} + 0.0000008 f_\mathrm{r}M - 0.004924(1 - f_\mathrm{r})6N$ for $N_\mathrm{v} \gtrapprox 512$. We estimate $f_\mathrm{r} \approx 1.281N_\mathrm{v}^{-0.2121} - 0.2505$ for $N = 10$ and $f_\mathrm{r} \approx 1.241N_\mathrm{v}^{-0.2291} - 0.2138$ for $N = 100$. (B) Computational cost Θ of PSRD (squares) and NSM (circles) as a function of the number of species N in the reaction network with the number of subvolumes fixed to $N_\mathrm{v} = 512$ (filled symbols) and $N_\mathrm{v} = 1728$ (empty symbols), respectively. The solid lines show the corresponding least-squares fits of the theoretical cost models: For $N_\mathrm{v} = 512$, $\Theta^{\mathrm{PSRD}} \approx 0.03051 \log N + 0.5291$, $\Theta^{\mathrm{NSM}} \approx 0.07885 \log N + 0.5458$; for $N_\mathrm{v} = 1728$, $\Theta^{\mathrm{PSRD}} \approx 0.08479 \log N + 0.5073$, $\Theta^{\mathrm{NSM}} \approx 0.1642 \log N + 0.5561$. The fraction $f_\mathrm{r} = 0.06$ for $N_\mathrm{v} = 512$ and $f_\mathrm{r} = 0.03$ for $N_\mathrm{v} = 1728$.

$\log N_\mathrm{v}$ for $N_\mathrm{v} \lessapprox 512$ and $\Theta(N_\mathrm{v}, N = 100) \approx 0.5929 \log N_\mathrm{v}$ for $N_\mathrm{v} \gtrapprox 512$. The abrupt increase in the prefactor of the scaling around $N_\mathrm{v} \approx 512$ is likely caused by cache-memory effects. In summary, the scaling of the computational cost of PSRD with respect to the number of subvolumes N_v is $O(\log_2 N_\mathrm{v})$. Again, this is the same scaling as that of NSM, but with a smaller prefactor.

Figure 5(b) shows the computational cost Θ as a function of the size N of the linear chain network using PSRD and NSM with $N_\mathrm{v} = 512$ and $N_\mathrm{v} = 1728$ subvolumes. We observe that for both N_v the ratio f_r is independent of the size N of the network. For $N_\mathrm{v} = 512, f_\mathrm{r} = 0.06$, decreasing to $f_\mathrm{r} = 0.03$ for $N_\mathrm{v} = 1728$. We observe that the scaling of Θ is slower than predicted by the theoretical cost analysis. This is not a violation of the theory since the theoretical analysis only provides an upper bound for the scaling. The slower scaling in the present case is specific to the particular reaction network. We obtain reasonable fits with a function linear in $\log N$. The asymptotic plateau is due to "diffusion reactions" of species S_1 accounting for the majority of all reaction firings. Since this reaction is on top of the list (species index 1), it is found in $O(1)$ time. For PSRD, $\Theta(N_\mathrm{v} = 512, N) \approx 0.03051 \log N$. For NSM, $\Theta(N_\mathrm{v} = 512, N) \approx 0.07885 \log N$. For the larger number subvolumes $N_\mathrm{v} = 1728$, Θ of PSRD is $\Theta(N_\mathrm{v} = 1728, N) \approx 0.08479 \log N$. For NSM, $\Theta(N_\mathrm{v} = 1728, N) \approx 0.1642 \log N$. In summary, the scaling of the computational cost of PSRD with respect to the size N of the reaction network is $O(N)$, since $\log N \in O(N)$.

V. TWO- AND THREE-DIMENSIONAL SRD SIMULATIONS USING PSRD

As an example application we use PSRD for two- and three-dimensional SRD simulations of the Gray-Scott reaction system,[3,45–48] given by:

$$\begin{aligned}
\emptyset &\xrightarrow{Fk_1u^3} \mathrm{S}_1\\
\mathrm{S}_1 &\xrightarrow{Fk_1u^2} \emptyset\\
\mathrm{S}_1 + 2\mathrm{S}_2 &\xrightarrow{k_1} 3\mathrm{S}_2\\
\mathrm{S}_2 &\xrightarrow{kk_1u^2} \emptyset\\
\mathrm{S}_2 &\xrightarrow{Fk_1u^2} \emptyset.
\end{aligned} \tag{19}$$

This system is widely used to study the formation of Turing patterns[12] in reaction-diffusion systems.[3] The trivial steady state of the system is $n_1 = u\Omega$ and $n_2 = 0$, where Ω is the volume of the reactor. For a fixed reactor volume, a larger u hence corresponds to a larger number of molecules in the reactor, reducing the effect of noise. In the limit of very large u, the kinetics of the stochastic system tends to that of the deterministic one.

The third reaction in the system is not elementary since it involves three reactant molecules. We therefore extend PSRD to also handle tri-molecular reactions by using a three-dimensional partial-propensity structure and factoring out two other reactants. We choose this strategy over expanding the network into elementary reactions in order to render the parameters k and F comparable to the deterministic limit case. We do not use discretization-corrected propensities since no theoretical framework is available for tri-molecular reactions.[33]

For the simulations we fix the dimensionless constants such that $F = 0.04$ and $k = 0.06$, and we choose the macroscopic rate $k_1 = 1$. In 2D we simulate the reaction-diffusion system in a computational domain of area $\Omega = 0.64^2$, divided into $K^2 = 64^2$ subvolumes (or subareas) of edge length $h = 0.01$. At the boundary of the computational domain, periodic boundary conditions are used for the jump reactions. The resulting reaction-diffusion system in 2D thus is:

$$\left.\begin{array}{ccc}
\emptyset & \xrightarrow{Fk_1u^3} & \mathrm{S}_1^{(l,m)} \\
\mathrm{S}_1^{(l,m)} & \xrightarrow{Fk_1u^2} & \emptyset \\
\mathrm{S}_1^{(l,m)} + 2\mathrm{S}_2^{(l,m)} & \xrightarrow{k_1} & 3\mathrm{S}_2^{(l,m)} \\
\mathrm{S}_2^{(l,m)} & \xrightarrow{kk_1u^2} & \emptyset \\
\mathrm{S}_2^{(l,m)} & \xrightarrow{Fk_1u^2} & \emptyset \\
\mathrm{S}_1^{(l,m)} & \xrightarrow{D_1/h^2} & \mathrm{S}_1^{(l',m')} \\
\mathrm{S}_2^{(l,m)} & \xrightarrow{D_2/h^2} & \mathrm{S}_2^{(l',m')}
\end{array}\right\} \quad \forall\ (l,m) \in I_{\mathrm{all}}\,, \qquad (20)$$

where I_{all} is the set of all possible subvolume indices in 2D and (l', m') are the neighboring subvolumes of (l, m) taking into account the periodic boundary conditions, hence $l' \in \{[((l-1) \pm 1 + 2K) \mathrm{mod}\ K] + 1\}$ and $m' \in \{[((m-1) \pm 1 + 2K) \mathrm{mod}\ K] + 1\}$. At $t = 0$, the initial population is:

$$n_1^{(l,m)} = \begin{cases} \frac{uh^2}{2} + \lfloor 0.04(r-0.5)uh^2 + 0.5 \rfloor, & \text{for } 24 \le l, m \le 40 \\ uh^2, & \text{otherwise.} \end{cases}$$

$$n_2^{(l,m)} = \begin{cases} \frac{uh^2}{4} + \lfloor 0.02(r-0.5)uh^2 + 0.5 \rfloor, & \text{for } 24 \le l, m \le 40 \\ 0, & \text{otherwise,} \end{cases} \qquad (21)$$

where r is a uniform random number in [0, 1) that acts as an initial perturbation. We use the diffusion constants $D_1 = 2 \times 10^{-5}$ and $D_2 = D_1/2$.

Figure 6 shows the 2D simulation results from PSRD and from a deterministic simulation. It shows the spatial concentration distribution of species S_1, normalized with u, at time $t_{\mathrm{f}} = 2000/(k_1u^2)$. Figures 6(a) and 6(b) show the normalized concentration distributions for $u = 10^6$ and 10^7, respectively, as obtained using PSRD. The maximum number of molecules of S_1 in any subvolume is on the order of $h^2u = 0.01u$. For $u = 10^6$, approximately 0.3×10^9 reaction events are simulated until t_{f} with $f_{\mathrm{r}} \approx 0.12$ and a total runtime (CPU time) of 157 s for PSRD and 200 s for NSM. For $u = 10^7$, the number of reaction events happening during the simulation increases to $\approx 3 \times 10^9$ with $f_{\mathrm{r}} \approx 0.14$ and a total runtime of 1854 s for PSRD and 2290 s for NSM. All timings were done on the same computer and using the same software as those presented in Sec. IV

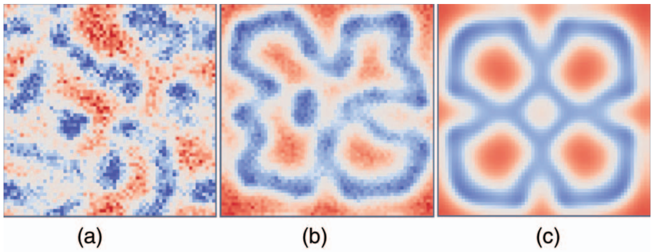

FIG. 6. Normalized spatial concentration distribution of species S_1 in the two-dimensional Gray-Scott reaction-diffusion system (Eq. (20)) for $F = 0.04$, $k = 0.06$, $k_1 = 1$, and $D_1 = 2D_2 = 2 \times 10^{-5}$ in a square computational domain of area 0.64^2, divided into $N_{\mathrm{v}} = 64^2$ subvolumes (or subareas) of edge length $h = 0.01$. The concentration in each subvolume is shown as a color ranging from blue (concentration zero) to red (concentration one). (A, B) Concentration distributions, normalized by u, obtained using PSRD for $u = 10^6$ (A) and $u = 10^7$ (B), respectively. (C) Normalized concentration distribution obtained from a deterministic simulation using the same parameters, simulated using second-order finite differences. All snapshots are taken at final time $t_{\mathrm{f}} = 2000/(k_1u^2)$.

Increasing u increases the total number of molecules in the reactor and hence decreases the noise in the system. The normalized concentration distribution obtained from a deterministic simulation is independent of u and is shown in Fig. 6(c). The deterministic simulation is done using the same numerical scheme as Pearson[3] in order to render the results comparable. This is a second-order finite-difference discretization of the Laplacian for the diffusion part and a first-order explicit Euler scheme with time-step size $\Delta t = 1.125$ for time stepping. The results show that as the number of molecules in the reactor increases with increasing u, the stochastic spatial pattern tends toward the deterministic one. The intrinsic noise in the stochastic system, however, breaks the symmetry of the pattern.

We also simulate the reaction-diffusion system in Eq. (20) in 3D (using triplet indices for the subvolumes) in a computational domain of volume $\Omega = 0.64^3$, divided into $K^3 = 64^3$ subvolumes of edge length $h = 0.01$. We use periodic boundary conditions in all three dimensions. At $t = 0$, the initial population is:

$$n_1^{(l,m,n)} = \begin{cases} \frac{uh^3}{2} + \lfloor 0.04(r-0.5)uh^3 + 0.5 \rfloor, & \text{for } 24 \le l, m, n \le 40 \\ uh^3, & \text{otherwise.} \end{cases}$$

$$n_2^{(l,m,n)} = \begin{cases} \frac{uh^3}{4} + \lfloor 0.02(r-0.5)uh^3 + 0.5 \rfloor, & \text{for } 24 \le l, m, n \le 40 \\ 0, & \text{otherwise,} \end{cases} \qquad (22)$$

where r is a uniform random number in [0, 1) that acts as an initial perturbation. We use the same diffusion constants as in the 2D case above.

Figure 7 shows the 3D concentration distribution of species S_1 at time $t_f = 2000/(k_1 u^2)$, normalized with $u = 10^8$. For these parameters, the maximum number of molecules of species S_1 in any subvolume is $uh^3 = 100$ and hence the intrinsic noise breaks the symmetry of the Turing pattern. Approximately 36×10^9 reaction events are simulated until t_f with $f_r \approx 0.1$. The total runtime (CPU time) for PSRD is 77 413 s, for NSM it is 100 636 s (extrapolated).

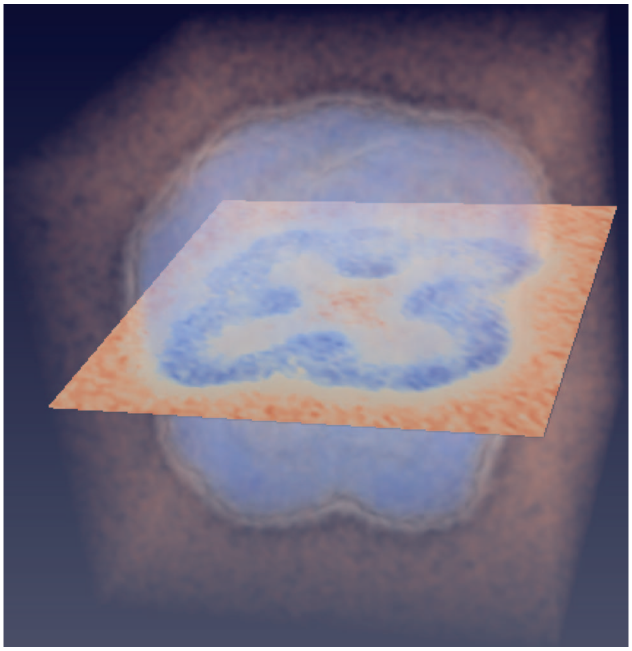

FIG. 7. Normalized spatial concentration distribution of species S_1 in the three-dimensional Gray-Scott reaction-diffusion system for $F = 0.04$, $k = 0.06$, $k_1 = 1$, and $D_1 = 2D_2 = 2 \times 10^{-5}$ in a cubic computational domain of volume 0.64^3, divided into $N_v = 64^3$ subvolumes of edge length $h = 0.01$. The concentration in each subvolume, normalized by $u = 10^8$, is shown as a color ranging from blue (concentration zero) to red (concentration one). The snapshot is taken at final time $t_f = 2000/(k_1 u^2)$.

VI. CONCLUSIONS AND DISCUSSION

We have introduced the on-lattice partial-propensity stochastic reaction-diffusion (PSRD) method. PSRD proceeds by dividing the computational domain into N_v subvolumes. The chemical reaction system in each subvolume is assumed to be well mixed and it is imposed that molecules can only react with partners within the same subvolume. Diffusion is modeled by jump "reactions" between neighboring subvolumes. PSRD combines composition-rejection sampling[39,42,56] with the concept of partial propensities.[41] Computational efficiency is achieved by binning the subvolumes and using partial propensities to group the reactions within each subvolume.

PSRD samples trajectories from the exact solution of the reaction-diffusion master equation for on-lattice reaction-diffusion systems, provided the subvolume sizes are within admissible bounds.[33,50,51] This is done by first sampling the subvolume using composition-rejection sampling, and then sampling the index of the next reaction within that subvolume using linear search over the dynamically grouped partial propensities, analogous to the sorting partial-propensity direct method (SPDM).[41] The computational cost of PSRD to sample the next subvolume is $O(G_a)$, where the number of bins is $G_a = \log_2(a_{\max}/a_{\min}) + 1$, $a_{\max}$ is the maximum total propensity in any subvolume, and $a_{\min}$ is the smallest non-zero total propensity in any subvolume. In any simulation, the number G_a scales at most as $O(\log_2 N_v)$. If the logarithmic span of the propensities can be*a priori*bounded by a constant, the cost of sampling the subvolume reduces to $O(1)$.[39] The computational cost to sample the index of the next reaction within a subvolume is $O(N)$, where N is the number of species in the reaction network. Thus, the overall computational cost of PSRD is $O(G_a + N)$, which is bounded in the worst case by $O(\log_2 N_v + N)$. This cost of PSRD is independent of whether the SRD simulation is dominated by "real" reactions or by "diffusion reactions." We demonstrated this scaling of the computational cost using prototypical benchmark cases for both types of reaction networks: strongly coupled and weakly coupled. For the former, the number of reactions scales super-linearly with the number of species. For the latter, the number of reactions is almost the same as the number of species.

PSRD inherits the limitations of partial-propensity methods.[40–42] It is hence limited to reaction networks comprising only elementary reactions. Non-elementary reactions can be broken down into sets of elementary ones at the expense of an increased network size.[30,43,44] For spatiotemporal reaction-diffusion simulations, however, including non-elementary reactions is of questionable value since no kinetic-theoretical framework exists for them.[33,54] It is hence unclear how the propensity functions of non-elementary reactions should be correctly formulated in a discretized space.[33]

Due to the more complex data structures used in partial-propensity methods, we do not expect PSRD to offer significant speed-ups for small ($N \lessapprox 10$) chemical reaction networks. In these cases, the next subvolume method (NSM)[5,24] can be as efficient or faster than PSRD. The data structures of PSRD also have a larger memory footprint than those of NSM, even though the scaling of the memory usage with problem size is the same for the two methods when not using a dependency graph in NSM (see Appendix). In addition, PSRD is restricted to chemical reaction networks that do not involve time delays. This could be overcome by using dPDM (delay PDM)[63] instead of SPDM inside each subvolume in PSRD. Our current software implementation of PSRD is moreover limited to rectangular computational domains. This limitation, however, is not inherent to the method as such and future developments will consider extending the method to computational domains of arbitrary shape,[64,65] e.g., using unstructured meshes.[66]

While we have described the basic version of PSRD for simplicity and conciseness of the presentation, the algorithm can be further improved in efficiency using standard techniques. Using a binary tree search instead of linear search over subvolume bins,[36] the computational cost of sampling the next subvolume can, for example, be reduced to $O(\log_2 G_a)$, rendering the overall computational cost of such a variant of PSRD $O(\log_2 G_a + N)$ and in the worst-case $O(\log_2 \log_2 N_v + N)$. Moreover, for weakly coupled reaction networks[41,42] the computational cost of sampling the next reaction within a subvolume can be reduced to $O(G_r)$ using the partial-propensity method with composition-rejection sampling (PSSA-CR) within each subvolume. G_r is the logarithmic span of non-zero propensities within the subvolume. In summary, the computational cost of PSRD can be reduced to $O(\log_2 G_a + N)$ or even $O(\log_2 G_a + \log_2 G_r)$ for certain classes of reaction networks and when using a binary search tree also within PSSA-CR. These improvements can be realized at the expense of larger memory requirements, which is why we did not include them in the presentation here. Their implementation, however, is straightforward and they will be included in future versions of the PSRD software package.

PSRD uses dynamic bubble sort for the reactions within each subvolume. This is inspired by the sorting direct method (SDM)[38] and its partial-propensity variant SPDM.[41] Sorting SSAs have been shown to be particularly efficient on multiscale (stiff) reaction networks where the propensities of different reactions are orders of magnitude apart. This means that a small fraction of reactions can potentially account for the majority of reaction events. The dynamic "bubbling up" of these reactions in the reaction list reduces the average search depth when sampling the next reaction as it accumulates the most frequent reactions at the top of the list. Using a sorting SSA inside each subvolume of an on-lattice SRD simulation is particularly advantageous since the propensities of different reaction types scale differently with subvolume size (see Eq. (2)). While the propensities of bimolecular reactions scale as Ω_c^{-1}, those of source reactions scale as Ω_c, and the propensities of unimolecular reactions are independent of Ω_c. The propensities of "diffusion reactions" scale as h^{-2}. Reducing the grid spacing h thus renders the reaction network increasingly multi-scale with the propensity ratio between the fastest and slowest reactions scaling at most as h^6 in 3D subvolumes (h^4 in 2D subvolumes).

Taken together, PSRD offers an improved scaling of the computational cost for exact on-lattice SRD simulations. This can lead to significant performance improvements when simulating strongly coupled spatiotemporal processes, such as colloidal aggregation and scale-free biochemical networks.[61,62,67,68]

A C++ software implementation of PSRD and its efficiency-improved variants, including delay-reaction versions thereof, will be made available as open source on the web page of the authors. We hope they will provide a useful tool for stochastic reaction-diffusion simulations in various disciplines, including chemical physics, chemical engineering, and systems biology.

ACKNOWLEDGMENTS

R.R. was funded by a grant from the Swiss SystemsX.ch initiative, grant WingX, evaluated by the Swiss National Science Foundation. This project was also supported with a grant from the Swiss SystemsX.ch initiative, grant LipidX-2008/011, to I.F.S.

APPENDIX: COMPUTATIONAL COST OF PSRD

The steps that define the scaling of the computational cost of PSRD are the sampling of the subvolume containing the next reaction, the sampling of the next reaction within that subvolume, and the update of the data structures after firing the sampled reaction.

The computational cost of the composition-rejection sampling of the next subvolume is $O(G_a)$. This is because (i) the composition step is a linear search over G_a bins, and (ii) the rejection step is $O(1)$ since the average number of iterations for this step is bounded by a constant thanks to the dyadic binning.[39,42]

The computational cost for sampling the next reaction within the selected subvolume (l, m, n) is $O(N)$. This step involves sampling the group index I and the element index J of the next reaction in the partial-propensity structure. Sampling the group index involves a linear search over the at most $N + 1$ elements of $\mathbf{\Sigma}^{(l,m,n)}$ and hence has a computational cost of $O(N)$. Sampling the element index involves a linear search over the $O(N)$ elements of $\mathbf{\Pi}_I^{(l,m,n)}$ and hence has a computational cost of $O(N)$ as well.

The computational cost for updating the data structures within a subvolume is at most $O(N)$. Assuming that the number of distinct species involved in any one chemical reaction is $O(1)$ (i.e., does not increase beyond a constant bound as the number of species in the network increases), the cost of updating the population of species is $O(1)$. Under the same assumption, the number of entries in $\mathbf{\Pi}^{(l,m,n)}$ that need to be updated after any reaction has fired is at most $O(N)$.[41,42] By the same argument, the cost of updating the partial-propensity structure of any neighboring subvolume upon firing of a "diffusion reaction" is at most $O(N)$.

Overall, the computational cost of PSRD thus is $O(G_\mathrm{a} + N)$, irrespective of the fraction f_r of "real" reaction firings. The asymptotically (for large N_v) worst case for PSRD is when half of the subvolumes contain bimolecular reactions and the other half source reactions. In 3D subvolumes, the propensity of the bimolecular reactions is proportional to h^{-3} whereas that of the source reactions is proportional to h^3, where h is the edge length of the subvolumes. As N_v increases, the logarithmic span of the propensities in the system hence increases. This leads to an increase in the number of bins G_a that is proportional to $\log_2 h^{-6} = 2\log_2 N_\mathrm{v} - 2\log_2 \Omega$, where Ω is the (constant) volume of the reactor. Therefore, $G_\mathrm{a} \in O(\log_2 N_\mathrm{v})$, rendering the computational cost of PSRD $O(\log_2 N_\mathrm{v} + N)$ in the worst case, independent of f_r. This worst-case scaling of PSRD's computational cost can be reduced to $O(\log_2 \log_2 N_\mathrm{v} + N)$ by using a tree search[36] to sample b in Eq. (12).

The data structures of PSRD require $O(M + N)$ memory per subvolume. Therefore, the total memory requirement of PSRD is $O((M + N)N_\mathrm{v})$.

[1] R. A. Fisher, Ann. Eugenics **7**, 355 (1937).

[2] A. N. Kolmogorov, I. G. Petrovsky, and N. S. Piskunov, Bull. Univ. d'Etat à Moscou (Bjul. Moskowskogo Gos. Univ.) **Série internationale A 1**, 1 (1937).

[3] J. E. Pearson, Science **261**, 189 (1993).

[4] M. A. J. Chaplain, M. Ganesh, and I. G. Graham, J. Math. Biol. **42**, 387 (2001).

[5] J. Elf and M. Ehrenberg, Syst. Biol. **1**, 230 (2004).

[6] D. Fange and J. Elf, PLoS Comput. Biol. **2**, 637 (2006).

[7] S. J. Altschuler, S. B. Angenent, Y. Wang, and L. F. Wu, Nature (London) **454**, 886 (2008).

[8] K. Takahashi, S. Tanase-Nicola, and P. R. ten Wolde, Proc. Natl. Acad. Sci. U.S.A. **107**, 2473 (2010).

[9] M. Bergdorf, I. F. Sbalzarini, and P. Koumoutsakos, J. Math. Biol. **61**, 649 (2010).

[10] D. M. Holloway, F. J. P. Lopes, L. da Fontoura Costa, B. A. N. Travencolo, N. Golyandina, K. Usevich, and A. V. Spirov, PLoS Comput. Biol. **7**, e1001069 (2011).

[11] R. D. Benguria and M. C. Depassier, Phys. Rev. Lett. **77**, 1171 (1996).

[12] A. M. Turing, Phil. Trans. R. Soc. London **B237**, 37 (1952).

[13] A. Gierer and H. Meinhardt, Kybernetik **12**, 30 (1972).

[14] A. Koch and H. Meinhardt, Rev. Mod. Phys. **66**, 1481 (1994).

[15] H. Meinhardt and A. J. d. B. Piet, Proc. Natl. Acad. Sci. U.S.A. **98**, 14202 (2001).

[16] C. X. Zhou, H. Y. Guo, and Q. Ouyang, Phys. Rev. E **65**, 036118 (2002).

[17] L. Yang, M. Dolnik, A. M. Zhabotinsky, and I. R. Epstein, Phys. Rev. Lett. **88**, 208303 (2002).

[18] G. H. Gunaratne, Q. Ouyang, and H. L. Swinney, Phys. Rev. E **50**, 2802 (1994).

[19] D. Panja, Phys. Rep. **393**, 87 (2004).

[20] C. W. Gardiner, K. J. McNeil, D. F. Walls, and I. S. Matheson, J. Stat. Phys. **14**, 307 (1976).

[21] M. Springer and J. Paulsson, Nature (London) **439**, 27 (2006).

[22] J. S. van Zon and P. R. ten Wolde, J. Chem. Phys. **123**, 234910 (2005).

[23] S. S. Andrews and D. Bray, Phys. Biol. **1**, 137 (2004).

[24] J. Hattne, D. Fange, and J. Elf, Bioinformatics **21**, 2923 (2005).

[25] D. Rossinelli, B. Bayati, and P. Koumoutsakos, Chem. Phys. Lett. **451**, 136 (2008).

[26] K. A. Iyengar, L. A. Harris, and P. Clancy, J. Chem. Phys. **132**, 094101 (2010).

[27] L. Ferm, A. Hellander, and P. Lötstedt, J. Comput. Phys. **229**, 343 (2010).

[28] W. Koh and K. T. Blackwell, J. Chem. Phys. **134**, 154103 (2011).

[29] M. Jeschke, R. Ewald, and A. M. Uhrmacher, J. Comput. Phys. **230**, 2562 (2011).

[30] D. T. Gillespie, Physica A **188**, 404 (1992).

[31] M. J. Morelli and P. R. ten Wolde, J. Chem. Phys. **129**, 054112 (2008).

[32] S. Hellander and P. Lotstedt, J. Comput. Phys. **230**, 3948 (2011).

[33] D. Fange, O. G. Berg, P. Sjoberg, and J. Elf, Proc. Natl. Acad. Sci. U.S.A. **107**, 19820 (2010).

[34] D. T. Gillespie, J. Comput. Phys. **22**, 403 (1976).

[35] D. T. Gillespie, J. Phys. Chem. **81**, 2340 (1977).

[36] M. A. Gibson and J. Bruck, J. Phys. Chem. A **104**, 1876 (2000).

[37] Y. Cao, H. Li, and L. Petzold, J. Chem. Phys. **121**, 4059 (2004).

[38] J. M. McCollum, G. D. Peterson, C. D. Cox, M. L. Simpson, and N. F. Samatova, Comput. Biol. Chem. **30**, 39 (2006).

[39] A. Slepoy, A. P. Thompson, and S. J. Plimpton, J. Chem. Phys. **128**, 205101 (2008).

[40] R. Ramaswamy and I. F. Sbalzarini, "Fast exact stochastic simulation algorithms using partial propensities," in *Proc. ICNAAM, Numerical Analysis and Applied Mathematics, International Conference* (AIP, New York, 2010), pp. 1338–1341.

[41] R. Ramaswamy, N. González-Segredo, and I. F. Sbalzarini, J. Chem. Phys. **130**, 244104 (2009).

[42] R. Ramaswamy and I. F. Sbalzarini, J. Chem. Phys. **132**, 044102 (2010).

[43] T. Wilhelm, J. Math. Chem. **27**, 71 (2000).

[44] K. R. Schneider and T. Wilhelm, J. Math. Biol. **40**, 443 (2000).

[45] P. Gray and S. K. Scott, Chem. Eng. Sci. **38**, 29 (1983).

[46] P. Gray and S. K. Scott, Chem. Eng. Sci. **39**, 1087 (1984).

[47] P. Gray and S. K. Scott, J. Phys. Chem. **89**, 22 (1985).

[48] K. J. Lee, W. D. McCormick, Q. Ouyang, and H. L. Swinney, Science **261**, 192 (1993).

[49] R. Erban and S. J. Chapman, Phys. Biol. **4**, 16 (2007).

[50] R. Erban and S. J. Chapman, Phys. Biol. **6**, 046001 (2009).

[51] Y. Kuramoto and T. Yamada, Progr. Theoret. Phys. **56**, 724 (1976).

[52] R. Ramaswamy, I. F. Sbalzarini, and N. González-Segredo, PLoS ONE **6**, e16045 (2011).

[53] R. Ramaswamy, N. González-Segredo, I. F. Sbalzarini, and R. Grima, "Discreteness-induced concentration inversion in mesoscopic chemical systems," Nat. Commun. (unpublished).

[54] M. von Smoluchowski, Z. Phys. Chem **92**, 129 (1917).

[55] J. A. Helmuth, S. Reboux, and I. F. Sbalzarini, J. Comput. Sci. **2**, 324 (2011).

[56] L. Devroye, *Non-Uniform Random Variate Generation* (Springer-Verlag, New York, 1986).

[57] B. L. Fox, ORSA J. Comput. **2**, 126 (1990).

[58] S. Rajasekaran and K. W. Ross, ACM Trans. Model. Comput. Simul. **3**, 1 (1993).

[59] T. Hagerup, K. Mehlhorn, and I. Munro, "Optimal algorithms for generating time-varying discrete random variates," in *Lecture Notes Computational Science*, Vol. 700 (Springer, New York, 1993), pp. 253–264.

[60] J. Hattne, "The algorithms and implementation of MesoRD," Master's thesis, Department of Information Technology, Uppsala University, 2006.

[61] R. Albert and A.-L. Barabási, Rev. Mod. Phys. **74**, 47 (2002).

[62] R. Albert, J. Cell Sci. **118**, 4947 (2005).

[63] R. Ramaswamy and I. F. Sbalzarini, J. Chem. Phys. **134**, 014106 (2011).

[64] I. F. Sbalzarini, A. Mezzacasa, A. Helenius, and P. Koumoutsakos, Biophys. J. **89**, 1482 (2005).

[65] I. F. Sbalzarini, A. Hayer, A. Helenius, and P. Koumoutsakos, Biophys. J. **90**, 878 (2006).

[66] S. Engblom, L. Ferm, A. Hellander, and P. Lötstedt, SIAM J. Sci. Comput. **31**, 1774 (2009).

[67] S. H. Strogatz, Nature (London) **410**, 268 (2001).

[68] A. L. Barabási and Z. N. Oltvai, Nat. Rev. Genet. **5**, 101 (2004).