A Gaussian jump process formulation of the reaction–diffusion master equation enables faster exact stochastic simulations

Cite as: J. Chem. Phys. 157, 194110 (2022); doi: 10.1063/5.0123073
Submitted: 28 August 2022 • Accepted: 1 November 2022 •
Published Online: 17 November 2022

Tina Subic[1,2,3] and Ivo F. Sbalzarini[1,2,3,4,a)]

AFFILIATIONS

[1] Technische Universität Dresden, Faculty of Computer Science, Dresden, Germany
[2] Max Planck Institute of Molecular Cell Biology and Genetics, Dresden, Germany
[3] Center for Systems Biology Dresden, Dresden, Germany
[4] Cluster of Excellence Physics of Life, TU Dresden, Dresden, Germany

[a)] Author to whom correspondence should be addressed: sbalzarini@mpi-cbg.de

ABSTRACT

We propose a Gaussian jump process model on a regular Cartesian lattice for the diffusion part of the Reaction–Diffusion Master Equation (RDME). We derive the resulting Gaussian RDME (GRDME) formulation from analogy with a kernel-based discretization scheme for continuous diffusion processes and quantify the limits of its validity relative to the classic RDME. We then present an exact stochastic simulation algorithm for the GRDME, showing that the accuracies of GRDME and RDME are comparable, but exact simulations of the GRDME require only a fraction of the computational cost of exact RDME simulations. We analyze the origin of this speedup and its scaling with problem dimension. The benchmarks suggest that the GRDME is a particularly beneficial model for diffusion-dominated systems in three dimensional spaces, often occurring in systems biology and cell biology.

I. INTRODUCTION

Reaction–diffusion models are a workhorse of mathematical modeling in chemical physics and cell biology. They describe the dynamics of interacting chemical species, or their concentration fields, in space and in time in the absence of any active transport mechanism or flow. They are thus often the simplest possible model of spatially resolved chemical dynamics. In biology, reaction–diffusion models have, for example, been sufficient to describe intra-cellular symmetry breaking and asymmetric cell division (Min system in *E. coli*,[1] Cdc42 system in *S. cerevisiae*,[2] and PAR system in *C. elegans*[3]), as well as chemotaxis.[4]

Despite their success in recapitulating spatiotemporal dynamics, deterministic reaction–diffusion models are inappropriate at scales where molecular discreteness becomes apparent. In biological systems, this is typically the case at sub-cellular scales where copy numbers of molecular can be low. The inherent stochasticity of chemical kinetics at the molecular scale then starts to dominate, as has, for example, been shown for the spatial patterns emerging during cell polarity establishment,[1,5] cell fate determination,[6] or in cellular signaling pathways.[7–9] In these cases, intrinsic molecular noise can lead to qualitatively different system behaviors than what would be predicted by a mean-field model.[10]

The combination of intrinsic molecular noise and spatially heterogeneous concentration fields can, among other methods, be described at the mesoscopic scale by the Reaction–Diffusion Master Equation (RDME).[11,12] This equation governs the space-time evolution of the localization probability of the molecules on a regular Cartesian lattice. It is mesoscopic in the sense that it accounts for molecular discreteness and stochastic kinetics, yet does not include molecular details, such as atom composition or electronic structure of the molecules, nor their localization within a lattice cell.

Neglecting the localization of individual molecules, the RDME tracks the space-time evolution of the numbers of molecules (i.e., the molecular population) in each lattice cell. As individual cells are assumed to be well-mixed, stochastic reaction kinetics are described by one Chemical Master Equation (CME)[13] in each cell. These CMEs are extended by additional *jump reactions* between face-connected adjacent cells in order to model diffusion of molecules between neighboring cells.[11] This can then directly be simulated, for example, by the Next Subvolume Method (NSM),[14] which extends the next reaction method[15] to RDME models by introducing a priority queue for the time of the next event in each lattice cell.

While NSM performs "exact" (in the sense that each and every event—reaction or jump—is simulated sequentially) simulations, it is computationally costly as diffusion jumps account for the vast majority of events and their frequency increases with the increasing spatial resolution of the lattice. Several attempts have been made to reduce the computational cost of the diffusion part of simulation. This included the binomial tau-leaping spatial stochastic simulation algorithm[16] and the Multinomial Simulation Algorithm (MSA),[17] both of which aggregate multiple diffusion jumps into single events, as well as the time dependent propensity function method,[18] which models the effect of diffusion on reactions. A similar approach using operator splitting has also been used for RDME on unstructured meshes.[19] Another approach is to aggregate over lattice cells instead of over jump events. This can be done, for example, by allowing diffusion jumps to non-adjacent cells[20] and is a good approximation for sufficiently dilute systems, where the probability of encountering a reaction partner along the jump is small. While all these algorithms demonstrated significant speedups, sometimes by orders of magnitude, they no longer resolve every stochastic event and are therefore not exact.

In the RDME, every diffusion event is modeled as a jump to an adjacent cell. This is akin to how the compact finite-difference stencil discretizes the continuous Laplace operator in deterministic mean-field diffusion processes.[14,20] This analogy is valid since Green's function of the continuous diffusion equation is formally identical to the Markov propagator density of Brownian motion, albeit corresponds to a different physical interpretation.[21,22] This suggests then that other numerical discretization schemes for the continuous diffusion operator can inspire alternative formulations of the RDME when interpreted in probability space.[23]

Here, we exploit this analogy to derive an RDME formulation that uses Gaussian jump processes to model diffusion to non-adjacent, larger neighborhoods of cells. We derive the corresponding jump rates by analogy to a discretization method for the continuous diffusion equation, called Particle Strength Exchange (PSE).[24] This results in a method that is in the same spirit as the Unstructured Reaction Diffusion Master Equation (URDME) method, where jump rates on unstructured meshes have been derived by analogy with finite element methods,[23,25–27,55] but makes the size of the neighborhood a user-controllable parameter. The present analogy with PSE is most natural for the classic PSE formulation, which uses Gaussian kernels for simulating diffusion. These kernels depend on the distance between the two points and are symmetric and mass conserving. Their coefficients are systematically derived by satisfying physical moment conditions on the diffusing (probability) distribution.[28] This solves the problem of time-dependent variance in the Gaussian transition kernel of Brownian motion and allows us to derive an alternative formulation of the RDME with larger diffusion neighborhoods, which we call Gaussian RDME (GRDME).

The GRDME can then be directly and exactly simulated using an extension of the NSM to larger jump neighborhoods. We present the resulting algorithm, which we call Gaussian NSM (GNSM). We show that the diffusion jumps to non-adjacent lattice cells accelerate the simulation roughly two-, four-, or six-fold in one-, two-, or three-dimensional domains, respectively. It also enables increasing the size of the diffusion jump neighborhood when refining the lattice so that jump reactions do not abound. We derive theoretical bounds for these speedups and confirm them in numerical experiments for different problem sizes and space dimensions. Finally, we study the accuracy of the GRDME in comparison with the classic RDME and characterize their limits of validity, providing guidelines for when to choose which formulation.

II. GAUSSIAN REACTION–DIFFUSION MASTER EQUATION

We start by deriving the GRDME from the analogy with the PSE method for simulating diffusion in the continuum.[24] For this, we first formulate the GRDME, then briefly review PSE, and finally derive the diffusion jump rates of the GRDME using PSE. Consider a system with C chemical species and R reaction channels. In reaction channel $j = 1, \ldots, R$, $\chi_{i,j}$ reactants of species X_i, $i = 1, \ldots, C$, interact with each other and get converted to $\chi'_{i,j}$ products. The reaction channels fire randomly and independently.

If the system is not dilute and well-mixed in a larger domain Ω, finite diffusion times of molecules have to be taken into account. The RDME model does so by partitioning Ω into $p = 1, \ldots, M$ cubic cells of edge length h such that the system within each individual cell can be assumed dilute and well-mixed. Each cell stores the molecular population within so that X_i^p is species i in cell p. Only molecules within the same cell can react with each other. This results in a much larger system of $R \cdot M$ reactions, as given in the following equation:

$$\begin{gathered}\chi_{1,j}X_1^p + \cdots + \chi_{C,j}X_C^p \xrightarrow{k_j} \chi'_{1,j}X_1^p + \cdots + \chi'_{C,j}X_C^p \\ \forall j = 1, \ldots, R, \qquad p = 1, \ldots, M.\end{gathered} \tag{1}$$

Diffusion between cells is modeled as first-order "jump" reactions of molecules between cells. This results in an additional set of first-order reactions, where a molecule in cell p is "converted" to a molecule of the same species in another cell q in the neighborhood $\mathcal{N}(p)$ of cell p,

$$X_i^p \xrightarrow{k_{D_i}^q} X_i^q \quad \forall p = 1, \ldots M,\ q \in \mathcal{N}(p). \tag{2}$$

The resulting system of compartmentalized reactions and diffusion jump events can be modeled with the RDME[11,29] in Eq. (3), where a is the reaction propensity.[13] This master equation describes the time evolution of the probability distribution $P()$ of the state $[\vec{s}^1, \ldots, \vec{s}^M]$, where $\vec{s}^p$ is the vector of the number of molecules of each species in cell p, i.e., the "molecular population" in cell p. The shift operator $E_{i,j}^k$ subtracts k from the state s_i^p of species i in cell p. The evolution of the state due to the compartmentalized reactions is described by the first row of the equation, and the evolution of the state due to the diffusion jumps is described by the second row,

$$\frac{\mathrm{d}}{\mathrm{d}t}P(\vec{s}^{1},\ldots,\vec{s}^{M},t) = \sum_{p=1}^{M}\sum_{j=1}^{R}\left(\prod_{i=1}^{C}E_{i,p}^{\chi_{i,j}-\chi'_{i,j}} - 1\right)a\left(\vec{s}^{p},\frac{\Omega}{M}\right)$$
$$\times P(\vec{s}^{1},\ldots,\vec{s}^{M},t)$$
$$+ \sum_{p=1}^{M}\sum_{q\in\mathcal{N}(p)}\sum_{i=1}^{C}\left(E_{i,p}^{1}E_{i,q}^{-1} - 1\right)k_{D_i}s_i^p$$
$$\times P(\vec{s}^{1},\ldots,\vec{s}^{M},t). \qquad (3)$$

Note that unlike the classic RDME, Eq. (3) allows for diffusion jumps within a potentially larger neighborhood $\mathcal{N}(p)$ of each cell p. This means that we cannot use the classic RDME[11] jump rates $k_D^i = D_i/h^2$ for diffusion constant D_i of species i and regular Cartesian cells with all edge lengths equal to h. We need to derive different rates.

The diffusion jump neighborhood in the original RDME[11] consists of the face-connected adjacent cells and, as such, is equivalent to the support of the compact $(2d+1)$-point finite-difference stencil to discretize the Laplace operator on a regular Cartesian lattice in d dimensions. The jump rates $k_{D,i} = D_i/h^2$ can, therefore, be derived from the probabilistic interpretation of this stencil.[23]

Here, we consider larger neighborhoods, as illustrated in Fig. 1. A molecule from the center cell can jump to any other cell within this larger neighborhood in one step/event. We derive the corresponding jump rates k_D^q for all $q \in \mathcal{N}(p)$ by analogy with another discretization method for the continuous Laplacian, Particle Strength Exchange (PSE).[24] PSE operators have larger support than finite-difference stencils.

PSE discretizes the Laplace operator Δ over a continuous concentration field f by

$$\Delta f(\vec{x}_p) \approx \frac{1}{\varepsilon^2}\sum_{q\in\mathcal{N}(p)} V_q(f(\vec{x}_q) - f(\vec{x}_p))\eta_\varepsilon(\vec{x}_p - \vec{x}_q), \qquad (4)$$

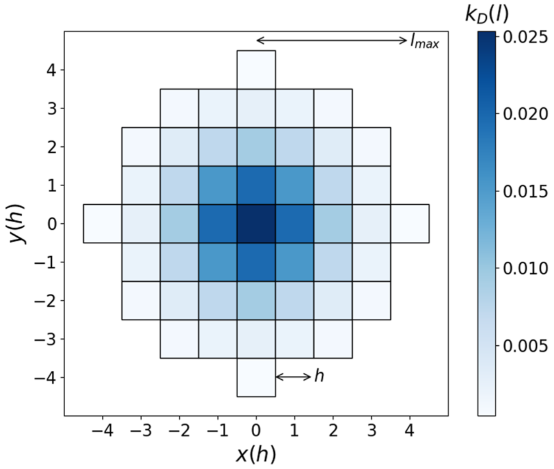

FIG. 1. Neighborhood $\mathcal{N}^{(0,0)}$ for a two-dimensional system with grid resolution h, smoothing length $\varepsilon = h$ (i.e., $\varepsilon' = 1$), and a cutoff radius $l_{\max} = 4h$. The center cell has coordinate (0, 0). Grid cells with the same $k_D(l)$, where l is the distance from the center, are filled with the same color [see Eq. (8)].

where $\vec{x}_p \in \mathbb{R}^d$ are the locations of the collocation points (here, lattice cell centers) and V_p is the volume of cell p. The operator kernel $\eta_\varepsilon(z) = \frac{1}{\varepsilon^d}\eta(\frac{z}{\varepsilon})$ has characteristic width ε and converges to the Dirac delta measure for $\varepsilon \to 0$. For PSE to be consistent, we must require $\varepsilon \geq h$, which is known as the "overlap condition".[24] It is, therefore, convenient to measure ε in units of h, $\varepsilon' = \varepsilon/h$. The kernel η_ε is a local (but not necessarily compact) radially symmetric function such that the neighborhood $\mathcal{N}(p)$ has a finite cutoff radius $l_{\max}$. There are many different choices of η_ε that lead to valid approximations in Eq. (4) with different orders of accuracy, cutoff radii, and dispersion properties. They all have in common that they fulfill a set of well-defined moment conditions,[28] which define a valid PSE operator.

A particularly natural choice for simulating diffusion is the Gaussian kernel,

$$\eta_\varepsilon = \left(\frac{1}{4\pi\varepsilon^2}\right)^{\frac{d}{2}}\exp\left(\frac{-\|\vec{x}_p - \vec{x}_q\|_2^2}{4\varepsilon^2}\right). \qquad (5)$$

When used in PSE for simulating continuous deterministic isotropic diffusion, this kernel results in second-order consistency of the approximation in Eq. (4) and exact (to machine precision) mass conservation if the collocation points are arranged on a regular Cartesian lattice.

Similar to how RDME jump rates can be derived from finite-difference stencils,[23] we here derive the jump rates for larger neighborhoods as illustrated in Fig. 1 by analogy to PSE with Gaussian kernels. Since the collocation points form a regular Cartesian lattice with resolution h, all lattice cells have identical volumes $V_p = V_q = h^d$. This allows us to rewrite the PSE scheme for the extensive quantity, such as mass or the molecular population, $s^p = h^d f(\vec{x}_p)$ for a well-mixed grid cell p. For isotropic, homogeneous, and normal diffusion in $\mathbb{R}^d$, this results in the dynamics

$$\frac{\mathrm{d}s^p}{\mathrm{d}t} = \frac{h^d D}{\varepsilon^2}\sum_{q\in\mathcal{N}(p)}(s^q - s^p)\left(\frac{1}{4\pi\varepsilon^2}\right)^{\frac{d}{2}}\exp\left(\frac{-\|\vec{x}_p - \vec{x}_q\|_2^2}{4\varepsilon^2}\right) \qquad (6)$$

at each grid cell p.

Splitting the sum over the neighborhood into influx and outflux terms,

$$\frac{\mathrm{d}s^p}{\mathrm{d}t} = \frac{h^d D}{\varepsilon^2}\sum_{q\in\mathcal{N}(p)} s^q\left(\frac{1}{4\pi\varepsilon^2}\right)^{\frac{d}{2}}\exp\left(\frac{-\|\vec{x}_p - \vec{x}_q\|_2^2}{4\varepsilon^2}\right)$$
$$- \frac{h^d D}{\varepsilon^2}\sum_{q\in\mathcal{N}(p)} s^p\left(\frac{1}{4\pi\varepsilon^2}\right)^{\frac{d}{2}}\exp\left(\frac{-\|\vec{x}_p - \vec{x}_q\|_2^2}{4\varepsilon^2}\right), \qquad (7)$$

provides a direct link to the diffusion jump rates from the outflux term in the second line of the equation for the first-order reactions from cell p to cell $q \in \mathcal{N}(p)$ [Eq. (2)]. We see from Eq. (7) that the jump rates have a Gaussian dependence on the Euclidean distance between cells p and q, $l_q = \|\vec{x}_p - \vec{x}_q\|_2$.

This results in the following rates for the diffusion jump reactions in Eq. (2):

$$k_{D_i}^q(l_q) = \frac{h^d D_i}{\varepsilon^2}s^p\left(\frac{1}{4\pi\varepsilon^2}\right)^{\frac{d}{2}}\exp\left(\frac{-l_q^2}{4\varepsilon^2}\right). \qquad (8)$$

The total propensity of the outflux of molecules of X_i from p to anywhere in $\mathcal{N}(p)$ is obtained by the following summation:

$$a_i^p = s_i^p \frac{D_i}{\varepsilon^2} \sum_{q \in \mathcal{N}(p)} \left(\frac{1}{4\pi\varepsilon^2}\right)^{\frac{d}{2}} \exp\left(\frac{-l_q^2}{4\varepsilon^2}\right) h^d \approx s_i^p \frac{D_i}{\varepsilon^2}. \tag{9}$$

The expected waiting time until the next diffusion jump event of any molecule of species i is thus $\tau = \frac{\varepsilon^2}{D_i}$. The PSE kernel width can therefore be expressed as $\varepsilon = \sqrt{D_i \tau}$. Replacing $\varepsilon^2 = D_i \tau$ in Eq. (8) recovers the discretized Green's function of diffusion. The GRDME therefore defines the rates for on-lattice diffusion jumps equivalently to Green's function of the continuous diffusion process.

III. LIMITS OF VALIDITY

For RDME and GRDME to describe physically consistent and "useful" processes, it is important to choose an appropriate grid resolution h. A good h must be small enough to resolve spatial inhomogeneities in the system, but large enough to avoid artificial loss of bimolecular reactions.[30–37] This constrains the grid resolution between $h_{\min}$ and $h_{\max}$ for which RDME models are valid. While $h_{\max}$ depends on the Kuramoto length of the system being modeled, $h_{\min}$ is an intrinsic property of the specific master equation model used. Previous works defined and provided physical reasons for the $h_{\min}$ of the RDME model,[30–32] proposed modifications/corrections to the bimolecular reaction propensities that reduce $h_{\min}$,[33,34] and combine the RDME with microscopic frameworks to assert convergence.[35–37] Since GRDME and RDME use different diffusion models, however, we expect their $h_{\min}$ to differ.

We therefore derive a guideline for $h_{\min}$ with which we can compare RDME and GRDME. For this, we build on top of the approaches of Isaacson[30] and Gillespie *et al.*[31] and design a heuristic guideline based on macroscopic reaction rates and diffusivities.

We start from the Kuramoto length of the system $L_K = \sqrt{2dD\tau}$, which is the average distance travelled by a molecule of species X_i during its lifetime τ.[38] The lifetime of X_i depends on the total propensity of all reactions that consume X_i, $\sum_{j \in R_2} s_j k_j + \sum_{j \in R_1} k_j$, where R_1 and R_2 are the sets of first and second order reactions, respectively, in which X_i participates. The Kuramoto length of X_i in a system of volume Ω is then given by

$$L_K^{\Omega,i} = \sqrt{2dD_i\,|\Omega| \left(\sum_{j \in R_2} s_{i'} k_j + |\Omega| \sum_{j \in R_1} k_j\right)^{-1}}, \tag{10}$$

where D_i is the diffusion constant of X_i and i' is the index of the species reacting with X_i in bimolecular reaction j. We approximate $h_{\max}$ as the smallest Kuramoto length of the system for any i,

$$h_{\max} = \min_{i=1,\ldots,C} L_K^{\Omega,i}. \tag{11}$$

To find an approximate $h_{\min}$, we consider the Kuramoto length of an individual grid cell. The RDME with conventional bimolecular reaction rates is consistent for grid resolutions for which the system is dilute, and hence well-mixed, within individual cells.[31] A cell p is well-mixed with respect to a bimolecular reaction j if the probability of the reaction occurring is the same anywhere in p. In other words, the Kuramoto length of a cell has to be much larger than $h_{\min}$: $L_K^p = \theta h_{\min}$, $\theta \gg 1$.

To estimate τ required to compute L_K^p, we assume that there are at most two molecules of any species in the cell p at any given time. Since the average number of molecules in the compartment p tends to zero as $h \to 0$, the probability of finding more than two molecules in the cell tends to zero for small enough h. This approximation holds well for sufficiently dilute systems. As there is then at most one reactant pair in cell p when h is small enough, the propensity of the bimolecular reaction is given by $a_j = 1 \cdot k_j/h^d$. This yields $\tau_j = h^d/k_j$ and

$$\theta h_{\min}^j = \sqrt{2d\hat{D}_j \frac{(h_{\min}^j)^d}{k_j}}, \quad \theta \gg 1. \tag{12}$$

Since the reaction system has to be well-mixed and dilute within the cell for the mass action assumption to hold, we take k_j to be the macroscopic rate constant in the reaction-limited regime. Then, $\hat{D}_j = D_i + D_{i'}$ is the total diffusivity of the reactant pair. $h_{\min}$ of the model is the largest $h_{\min}^j$ of any bimolecular reaction j,

$$h_{\min} = \max_{j \in R_2} h_{\min}^j.$$

We expect GRDME and RDME to differ in the value of θ since the two diffusion models have different propensities. To see this, we express $L_K^p = \theta h_{\min}$ in terms of the propensity of the diffusion reactions. Since diffusion events of different molecules are mutually independent, the total propensity is the sum $a_{\hat{D}} = a_D^i + a_D^{i'}$. For the RDME and GRDME, this is $a_D = cD/h^2$, with $c = 2d$ for RDME and $c = 1/\varepsilon'^2$ for GRDME, where $\varepsilon' = \varepsilon/h$ is the relative smoothing length. Expressing $\hat{D} = a_{\hat{D}} h^2/c$ and $a_{\hat{D}} = 1/\tau_j$, we get $\theta h = \sqrt{h^2/c}$. Therefore, θ of the GRDME is a factor of $\sqrt{2d}\varepsilon'$ larger than θ of the RDME.

From Eq. (12), we can find approximations for $h_{\min}$ for any dimension. For one-dimensional systems $(d = 1)$, both the RDME and GRDME are consistent for all choices of h. For two-dimensional systems $(d = 2)$, we find that the two models are consistent if the ratio between the diffusivity and the reaction rate is high enough, specifically if

$$k < \frac{4\hat{D}}{\theta^2}. \tag{13}$$

For three-dimensional systems $(d = 3)$, the criterion leads to[39]

$$h_{\min} = \frac{k\theta^2}{6\hat{D}}, \tag{14}$$

a rule of thumb similar to the one provided by Isaacson.[30] $h_{\min}$ for three-dimensional systems is the most restrictive. Since θ of GRDME is $\sqrt{2d}\varepsilon'$ larger than that of the RDME, we therefore expect $h_{\min}$ of the GRDME to be $6\varepsilon'^2$-times larger than for the RDME. We verify this in numerical experiments in Sec. V C for $\varepsilon' = 1$.

The analysis here shows that both RDME and GRDME experience loss of bimolecular reactions with the decreasing cell size, but for the GRDME, this already happens at a $2d$-fold larger grid resolution. If one knows $h_{\min}$ for one diffusion model, one can estimate

h_{min} for the other model for a given reaction system. Since h_{min} depends on the propensity of the diffusion model, it can be estimated for any kernel width ε with h_{min} quadratically increasing with ε, yielding

$$h_{\text{min}}^{\text{GRDME}} = 2d\varepsilon'^2\, h_{\text{min}}^{\text{RDME}}. \tag{15}$$

IV. GAUSSIAN NEXT SUBVOLUME METHOD

The GRDME in Eq. (3) with jump rates as given in Eq. (8) can be simulated exactly by extending the NSM[14] algorithm to enable diffusion jumps to a larger neighborhood, leading to the GNSM algorithm. GNSM differs from NSM in the way it stores the information about the neighboring cells and in the way it finds the target cell q in the diffusion event.

In the original NSM, information about cell adjacency and the boundary conditions is precomputed and stored in a connectivity matrix (cf. supplementary materials of Ref. 14). Using a similar approach for the extended neighborhoods of GNSM would lead to prohibitively large memory consumption, especially for three-dimensional domains. Therefore, GNSM computes this information on the fly from a dedicated neighborhood data structure $\mathcal{N}$ when executing a diffusion jump.[40] This data structure stores the translation-invariant relative index shifts of all cells within a d-dimensional ball with center 0^d and radius l_{max}. These represent the potential jump vectors (from any p) in cell grid units. In order to build this data structure, the algorithm proceeds in rings of increasing radius l since diffusion jump rates only depend on the distance between the two cells. For each radius l, all cell shifts $w(l)$ for that radius are collected in a tuple v, along with their total number $|w(l)|$, the radius l, and the corresponding diffusion jump rate $k_D(l)$. Final $\mathcal{N}$ is then the tuple of all tuples v, sorted by increasing l.

This procedure is detailed in Algorithm 1 for $d = 3$. The procedures for $d = 1, 2$ are analogous, but simpler. Since the Gaussian PSE kernel has infinite support, there is no actual cutoff radius. Therefore, we start in line 1 by determining l_{max} from the smallest k_D the user still wants to consider (in our benchmarks, we use $k_{\text{min}} = 10^{-10}$, which is the value below which the cumulative sum of the diffusion rates $k_D(l)|w(l)|$ saturates on a 64-bit computer due to finite-precision arithmetic). Line 2 then uses Algorithm 3 (see Appendix A) to create an array MS of all multisets m of cardinality d taken from the set $\{0, 1, \ldots, \lceil l_{\text{max}} \rceil\}$. Each element of MS is a tuple (m, l, nmp), where l is the radius of all coordinates that can be built from m and nmp is the number of multiset permutations of m. This creates an array of all index multisets in a cube with edge length $2l_{\text{max}}$. To obtain the index multisets of all cell coordinates within a ball of radius l_{max}, we remove the elements of MS with l larger than l_{max} in line 3. The elements are then sorted by increasing l in line 4. For each element in MS, we then create an array MP of all multiset permutations of m using Algorithm 4 (see Appendix A) in line 6. Finally, in line 8, we create an array SP for each permutation, where we use Algorithm 5 (Appendix A) to generate all sign permutations of the index vectors in mp. These coordinates are appended to w in line 9. The diffusion jump rate corresponding to this radius is computed in line 11; the tuple v is built in line 12 and appended to the neighborhood $\mathcal{N}$ in line 13. This data structure provides direct access to all information needed to sample diffusion events in Algorithm 2.

Algorithm 1. Construct neighborhood $\mathcal{N}$ ($d = 3$).

```
Require: $k_{\min}$, $\varepsilon' = \varepsilon/h$
 Ensure: $\mathcal{N}$
 1: $l_{\max} = 2\varepsilon'\sqrt{-\ln k_{\min} + \ln((2\sqrt{\pi\varepsilon'})^d)}$
 2: Use Algorithm 3 to build an array MS of elements $(m, l,\ \text{nmp})$
 3: Remove elements in MS with $l$ larger than $l_{\max}$
 4: Sort elements in MS by increasing $l$
 5: for all ms in MS do
 6:     Use Algorithm 4 to create an array MP with permutations of ms.$m$
 7:     for all mp in MP do
 8:         Use Algorithm 5 to create an array SP with permutations of the signs of mp
 9:         Append all elements in SP to $w$
10:     end for
11:     $k_D = (\frac{1}{4\pi\varepsilon'^2})^{d/2}\ \exp(-\,\text{ms}.l^2/4\varepsilon'^2)$
12:     Create a tuple $v = (w, |w|, k_D,\ \text{ms}.l)$
13:     Append $v$ to $\mathcal{N}$
14: end for
```

For each diffusion event, Algorithm 2 is used to find the target cell q. It first samples the index β in the tuple v corresponding to the jump radius (lines 2 to 5) using a uniformly distributed random number ω between 0 and 1 and computing the cumulative sum of the diffusion rates. The diffusion rates sum to 1 by construction and therefore do not have to be further normalized before computing the cumulative sum. Once the index β is identified, the random number ω is linearly rescaled in line 6 to find the jump index shift $\vec{x}^{\,\text{jump}}$ from the candidates in the tuple $w(l(\beta))$ of the data structure $\mathcal{N}$ in line 7. The coordinates of the source cell are then computed in line 8, and the jump shift is added in line 9 to compute the coordinates of the tentative target cell. In line 10, the method of images is used to apply appropriate boundary conditions (e.g., reflection or absorption) before the index of the target cell is computed from its coordinates in line 11. The way reaction events are handed is identical to how NSM[14] does it and is therefore not described here.

Algorithm 2. Diffusion event: find target cell q.

```
Require: $\mathcal{N}$, $\omega \sim U_{[0,1]}$, $p$
 Ensure: $q$
 1: $\beta = 0$, $c = 0$
 2: while $c < \omega$ do \(▷\) find index $\beta$ in $v$ with jump radius $l(\beta)$
 3:     $c = c + |w(l(\beta))| k_D(l(\beta))$
 4:     $\beta = \beta + 1$
 5: end while
 6: Rescale $\omega$ to $\omega_\beta = \frac{\omega - c}{|w(l(\beta))| k_D(l(\beta))} + 1$
 7: Use $\omega_\beta$ to find $\vec{x}_{\text{jump}}$ from $w(l(\beta))$
 8: Compute the coordinate $\vec{x}_p$ of the center point of cell $p$
 9: $\vec{x'}_q = \vec{x_p} + \vec{x}_{\text{jump}}$
10: Find $\vec{x}_q$ by reflecting $\vec{x'}_q$ using the method of images
11: Compute cell index $q$ for location $\vec{x}_p$
```

TABLE I. The size of elements in $\mathcal{N}$ for $\varepsilon = h$ ($\varepsilon' = 1$) and $k_{\min} = 10^{-10}$.

d	$l_{\max}$	$\lvert\mathcal{N}\rvert$	$\sum_\beta \lvert w\rvert$
1	10.751	11	21
2	10.984	56	373
3	11.212	191	5887

In order to compare the computational complexity of NSM and GNSM, we note that the size of $\mathcal{N}$ in GNSM depends on $l_{\max}$, but is independent of h and of the edge length L of the simulation domain Ω. The connectivity matrix in NSM, on the other hand, contains the neighbors of every cell and hence scales with the size of the domain as $2d(L/h)^d$. The total number of coordinates $\sum_\beta |w|$ in $\mathcal{N}$ can be approximated by the volume of a d-dimensional ball with radius $l_{\max}$ (giving $2l_{\max}$, $\pi l_{\max}^2$, and $4/3\pi l_{\max}^3$ in one-, two-, and three-dimensions, respectively) and hence scales exponentially with dimension (Table I, column 4). Since the coordinates in w use the majority of the memory in $\mathcal{N}$, this data structure becomes more memory efficient than a connectivity matrix when the number of cells per edge length exceeds $\sum_\beta |w|^{1/d} \approx 20$.

Each diffusion jump event in GNSM consists of more operations than in NSM. Finding q in NSM only requires sampling an element in row p of the connectivity matrix using a random number. This is comparable in complexity to line 7 in Algorithm 2. All operations before line 7 provide additional complexity for computing a diffusion event in GNSM. The maximum number of operations to find the index β in lines 2 to 5 is bounded by the size of $\mathcal{N}$ (column 3 of Table I). This is approximately $|\mathcal{N}| \approx V_{\text{ball}}/V_{\text{cube}}\binom{\lceil l_{\max}\rceil + d}{d}$, where V_{ball} is the volume of a d-dimensional ball with radius $l_{\max}$ and V_{cube} is the volume of a d-dimensional cube with length $2\lceil l_{\max}\rceil$. Large $|\mathcal{N}|$ imply a large average number of loop iterations in lines 2–5 of Algorithm 2. However, the cumulative sum c grows fastest for the smallest l, as seen in Table II, which shows the index β at different values of the cumulative sum c in different dimensions. In the best-case scenario, when the loop is short, the complexity of a diffusion event in GNSM is thus approximately an order of magnitude larger than in NSM. In the worst-case scenario, when the number of loop iterations is equal to $|\mathcal{N}|$, a diffusion event in GNSM is up to 200 times costlier than in NSM. Since the majority of the probability mass of the Gaussian kernel is concentrated close to the source of the jump (see Fig. 1), the average cost is closer to the optimal case than to the worst case.

While the computational cost for executing one diffusion jump is much larger in GNSM than in NSM, we show in Sec. V B that GNSM is overall still computationally more efficient because diffusion jumps in the GRDME happen less frequently (i.e., diffusion "reactions" have a much smaller propensity than in the RDME).

TABLE II. Values of the index β at different values of the cumulative sum c in Algorithm 2 and for $l_{\max}$ in $\mathcal{N}$ in dimensions $d = 1, 2, 3$.

d	$c = 0.5$	$c = 0.68$	$c = 0.95$
1	1	2	3
2	2	4	9
3	3	7	24

V. NUMERICAL EXPERIMENTS

Here, we compare the GRDME and RDME models with respect to their accuracy and benchmark the computational performance of GNSM in comparison to NSM. Finally, we characterize the limits of validity of the GRDME and RDME models with respect to the lattice cell size h. In order to ensure a fair comparison, we implemented both NSM and GNSM in the same software environment. Both methods were implemented in C++ using the Eigen[41] library to represent arrays and the Boost[42] library to implement the priority queues and compute linear least squares regressions of the Mean Squared Displacement (MSD). Statistical analysis and plotting of the results were done in Python using the pandas,[43] numpy,[44] Matplotlib,[45] and Seaborn libraries.[46] By default, the width of the PSE kernel is set to the edge length of the lattice cells, i.e., $\varepsilon = h$ ($\varepsilon' = 1$). This choice of ε leads to not only the highest accuracy of the diffusion model but also the highest computational cost of GNSM. It therefore provides a lower bound for the speedup GNSM provides over NSM. Results for $\varepsilon' > 1$ are provided in Appendix B.

A. Accuracy of the diffusion model

We start by comparing the accuracy of the diffusion models in the RDME and GRDME as this is the only difference between the two. To do so, we simulate a case with a known analytical solution: diffusion from a point source in a one-, two-, and three-dimensional open domain in the absence of any chemical reactions. Therefore, these benchmarks consider only a single species and track the evolution of its molecular population across lattice cells.

Since the models are stochastic, we consider convergence in both the strong and the weak sense. To quantify convergence in the strong sense (i.e., in value), we simulate many repetitions of a single particle diffusing out from the point source and compute the L_2 error in the mean squared displacement across the so-obtained trajectories. To quantify weak convergence, we consider the errors in the first four moments of the probability distribution of a population of N molecules diffusing out from a point source at the origin.

1. Strong convergence: Mean squared displacement

The Mean Squared Displacement (MSD) of a traveling molecule or particle is defined from its trajectory over time as MSD $:= \langle \|\vec{x}(t) - \vec{x}_0\|_2^2 \rangle$, where $\vec{x}(t)$ is the location of the particle at time t for a process that started from location $\vec{x}_0$ at initial time 0. The angle brackets denote an ensemble average over many *i.i.d.* repetitions of the process. For a particle undergoing normal, isotropic, and homogeneous diffusion with diffusion constant D in an open d-dimensional space, the MSD is a linear function of time: MSD $= 2dDt$.

We compute MSD curves from RDME and GRDME simulations of diffusion of a single particle in an open system in $d = 1, 2, 3$ dimensions until final time $T = 10^7$ with $D = 1$ and $h = 1$ (all units dimensionless). The MSD averages are computed in a sliding window along the trajectories at time intervals of 1 until a final time shift of 100. The slope α_1 and offset α_0 of a linear regression to the MSD

TABLE III. L_2 errors and their 95% confidence intervals (over 10 independent repetitions of each experiment) for the offset α_0 and slope α_1 coefficients of the linear fit to the MSD curve of a particle diffusing in open space in one, two, and three dimensions according to RDME and GRDME.

d	α	RDME	GRDME
1	0	0.089 676 ± 0.058 389	0.106 235 ± 0.054 984
	1	0.005 956 ± 0.003 575	0.007 367 ± 0.003 594
2	0	0.216 079 ± 0.100 957	0.172 402 ± 0.070 996
	1	0.014 352 ± 0.006 606	0.010 178 ± 0.005 423
3	0	0.121 224 ± 0.050 473	0.155 069 ± 0.066 789
	1	0.010 434 ± 0.003 732	0.010 846 ± 0.005 069

curves have been obtained by a linear least-squares fit of the MSD values using the Boost library. To obtain the confidence intervals of the errors, we repeated the whole procedure 10 times. Table III summarizes the L_2 errors of the α_0 and α_1 coefficients for RDME and GRDME with respect to the ground truth $\hat{\alpha}_0 = 0$ and $\hat{\alpha}_1 = 2dD$. The errors in both coefficients are comparable between RDME and GRDME and mutually contained within their confidence intervals. We thus conclude that both models are equally accurate in the strong sense.

2. Weak convergence: Probability distribution moments

We study the convergence of the models in the weak sense by measuring the L_2 error in the first four moments of the localization probability distribution of a population of diffusing molecules. The ground-truth probability distribution is a Gaussian with variance $\sigma^2 = 2dDT$. We consider the first moment (mean) μ_1', central second moment (variance) μ_2, standardized third moment (skewness) μ_3/σ^3, and standardized fourth moment (kurtosis) μ_4/σ^4. The expected convergence order of the error in the moments is $O(N^{-1/2}) + O(h^2)$, with the increasing molecule number N and decreasing lattice cell size h, reducing the error. The first term, $O(N^{-1/2})$, is a consequence of the central limit theorem when computing the moments over the population of N molecules. The second term, $O(h^2)$, corresponds to the (asymptotic) convergence rate of the deterministic discretization underlying the diffusion jumps, i.e., second-order central finite differences for RDME and the second-order accurate Gaussian PSE kernel for GRDME. The two errors are additive because we assume their sources to be statistically independent of each other.

We first study convergence with respect to N by running the same point-source diffusion simulation for $N = 10^k$, $2 \cdot 10^k$, and $5 \cdot 10^k$ for $k = 2, \ldots, 7$ until final time $T = 100$ with diffusion constant $D = 1$, grid resolution $h = 1$, and $\varepsilon' = 1$ (dimensionless units).

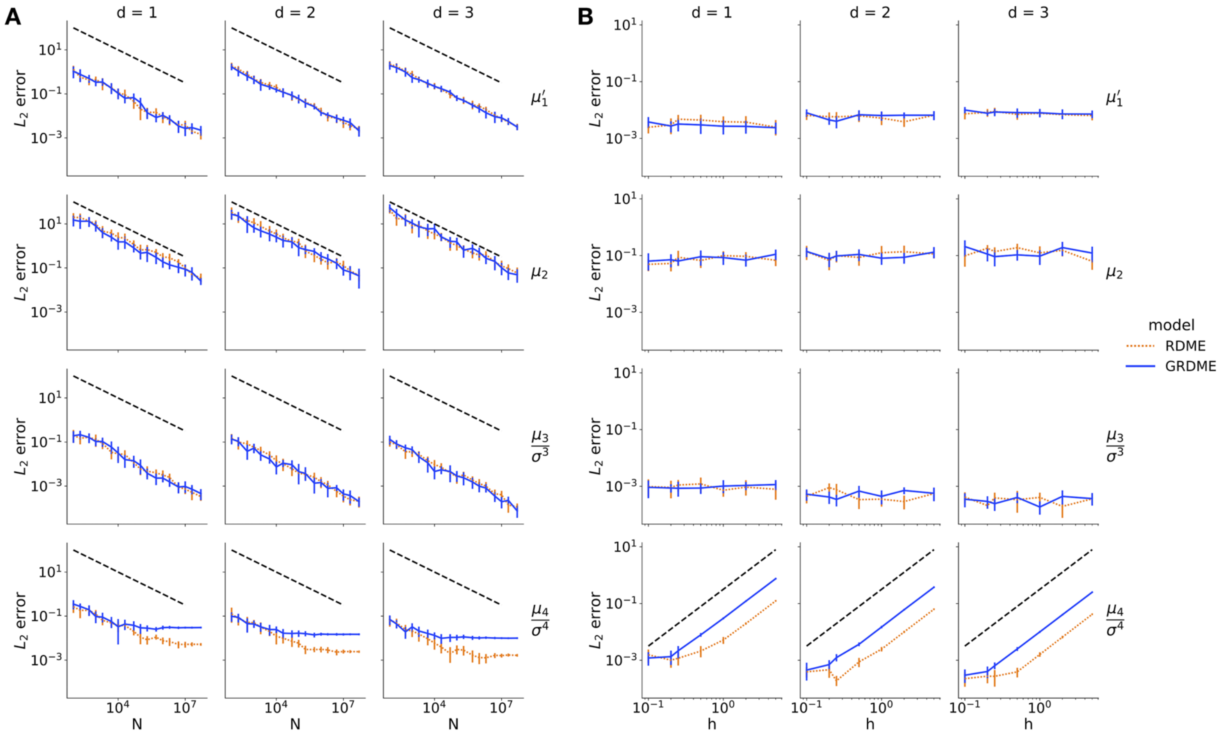

FIG. 2. L_2 error convergence of the first four moments μ_1', μ_2, μ_3/σ^3, and μ_4/σ^4 in dimensions $d = 1, 2, 3$. (a) Convergence with respect to the number of molecules N for RDME (orange dotted lines) and GRDME (blue solid lines) for point-source diffusion in an open domain with $h = 1$ and $D = 1$ at final time $T = 100$. The black dashed lines indicate the theoretically expected convergence order $-1/2$. (b) Convergence with respect to the lattice resolution h for RDME (orange dotted lines) and GRDME (blue solid lines) for the same problem with $N = 10^7$ and $D = 1$ at time $T = 100$. The black dashed lines indicate the theoretically expected convergence order 2.

L_2 errors are computed with respect to the moments of the ground-truth Gaussian distribution with mean zero and variance $\sigma^2 = 2dDT$. We simulate ten *i.i.d.* repetitions for each N to obtain confidence intervals of the errors. The results are shown in Fig. 2(a) along with dashed lines indicating the theoretically expected convergence order of $-1/2$ with respect to N. For large N and high moments, however, the error from the grid cell size h becomes dominant, leading to the plateaus seen in the bottom row of Fig. 2. Table IV compares the values of the L_2 errors in all moments for $N = 5 \cdot 10^7$. The errors in the first three moments are comparable between RDME and GRDME, whereas the error in the fourth moment is approximately six times larger for GRDME than for RDME once the plateau is reached. This is consistent with the error stemming from h being about six times larger in GRDME than in RDME because of the larger neighborhood considered.

To study the convergence of the GRDME with respect to h, we perform ten *i.i.d.* simulations $h = 0.1$, 0.2, 0.25, 0.5, 1.0, 2.0, and 5.0 for a large population size, $N = 10^7$ with $T = 100$, $D = 1$, and $\varepsilon' = 1$. The results are shown in Fig. 2(b). Reducing h does not reduce the L_2 error for the first three moments, confirming that the dominant error term for these moments is $O(N^{-1/2})$. For the N simulated, as already seen in Fig. 2(a), however, the dominant error in the fourth moment stems from h. This is confirmed in the bottom panel of Fig. 2(b), where the dashed line indicated the theoretically expected convergence order of 2 with respect to h. For small h, we again observe a plateau where the error from N becomes dominant again.

Despite the different propensities of the diffusion models of GRDME and RDME, the two models are comparable in their error for small population sizes. Since stochastic models are predominantly used to study the effect of intrinsic noise in diffusion-limited regimes, which implies small population sizes, the two models are of comparable accuracy for practical applications. The error for small N is dominated by the noise generated by the Brownian motion of the diffusing molecules. The main difference between the two models is in the error of the fourth moment for large population sizes, where the error for GRDME is larger. For both models, the error converges correctly with order 2 for decreasing h, which matches the convergence order numerical schemes used to derive the models. Both the errors in h and in N, however, cause plateaus in the convergence of the other respective error when they become limiting.

TABLE IV. L_2 errors and their 95% confidence intervals (over 10 independent repetitions of each experiment) for the first four moments of the probability distribution of a population of $N = 5 \cdot 10^7$ molecules diffusing out from a point source with $D = 1$, $h = 1$, at $T = 100$, compared between RDME and GRDME for different space dimensions d.

Moment	d	RDME	GRDME
μ'_1	1	0.001 454 ± 0.000 584	0.002 249 ± 0.000 884
	2	0.002 520 ± 0.000 769	0.002 105 ± 0.000 808
	3	0.003 238 ± 0.000 845	0.003 136 ± 0.000 650
μ_2	1	0.034 944 ± 0.015 896	0.025 486 ± 0.007 220
	2	0.045 222 ± 0.021 068	0.044 089 ± 0.034 839
	3	0.059 043 ± 0.033 842	0.049 188 ± 0.026 956
$\frac{\mu_3}{\sigma^3}$	1	0.000 336 ± 0.000 103	0.000 460 ± 0.000 218
	2	0.000 205 ± 0.000 103	0.000 197 ± 0.000 069
	3	0.000 113 ± 0.000 037	0.000 074 ± 0.000 039
$\frac{\mu_4}{\sigma^4}$	1	0.005 198 ± 0.000 311	0.030 215 ± 0.000 291
	2	0.002 415 ± 0.000 101	0.014 866 ± 0.000 178
	3	0.001 648 ± 0.000 096	0.009 977 ± 0.000 069

The results in Fig. 5 for $\varepsilon' > 1$ further show that the error increases quadratically for increasing ε', consistent with the theoretical prediction of an error order $O(h^2 \varepsilon r^2)$ for the GRDME.

B. Computational cost

The computational cost of an exact stochastic reaction–diffusion simulation depends both on the propensities of the model and on the algorithms used to sample and execute the events. In the RDME diffusion model, each cell has $2d$ possible outgoing diffusion jumps, each with a rate of $k_D = \frac{D}{h^2}$. The total propensity of diffusion jumps for one cell is thus $a_p^{\text{diff}} = s^p \frac{2dD}{h^2}$. As derived in Eq. (9), in GRDME, the total diffusion propensity is $a_p^{\text{diff}} = s^p \frac{D}{\varepsilon^2}$ since molecules can jump further and therefore do so less frequently. For $\varepsilon = h$, we therefore expect GNSM to have a roughly $2d$-fold lower total diffusion propensity than NSM.

It is not clear, though, whether this reduction in propensity directly translates to a proportional saving in algorithm runtime since, as we have outlined in Sec. IV, each individual diffusion jump is costlier to compute in GNSM than in NSM. We therefore conduct numerical experiments to benchmark the runtimes of the two algorithms. For this, we simulate diffusion of $N = \{1 \cdot 10^k, 2 \cdot 10^k, 5 \cdot 10^k\}_{k=1,2,3,4}$ molecules for a system with no-flux boundary conditions in one-, two-, and three-dimensional space with $D = 1$, $\varepsilon = h = 1$, $L = 10$, and $T = 10^4$. Simulations were per-

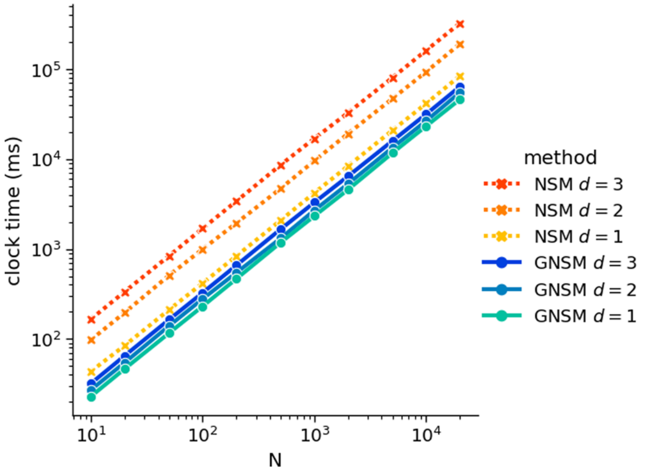

FIG. 3. Total runtimes of simulations of diffusion in a bounded domain (see the main text for the simulation setup) with N molecules, simulated using NSM (dotted lines in shades of orange) and GNSM (solid lines in shades of blue). The speedups $\rho_d = \text{time(NSM)}/\text{time(GNSM)}$ are $\rho_1 = 1.81$, $\rho_2 = 3.47$, and $\rho_3 = 5.06$. Therefore, GNSM achieves 85% of the maximally possible speedup of $2d$-fold. The error bars over 10 independent repetitions of each simulation are smaller than the symbol size.

formed using the GCC 10.3.0 compiler with the o3 optimization flag on the AMD Ryzen Threadripper 3990X 64-Core processor with the Ubuntu 20.04.4 LTS operating system. The results are shown in Fig. 3. As expected, both methods have a computational cost that scales linearly with the number of molecules, and GNSM has a lower computational cost than NSM in all dimensions. We also find that the runtimes of GNSM less strongly depend on the dimensionality of the space than those of NSM, which is in line with the above propensity estimates. As expected from those estimates, the speedup afforded by GNSM is larger in higher dimensions. Because each individual diffusion event is costlier to compute in GNSM, though, the speedups are smaller than the maximum possible ones of $2d$. We empirically find them to be 1.81-fold for $d = 1$, 3.47-fold for $d = 2$, and 5.06-fold for $d = 3$, which is about 85% of $2d$-fold.

This speedup can be further increased, at the expense of a larger error, by increasing ε'. Indeed, as confirmed by the results in Fig. 6, the computational cost of GNSM is up to $2d\varepsilon'^2$ times smaller than that of NSM overall.

C. Limits of validity for a homobimolecular reaction

The theoretical considerations in Sec. III predicted that the smallest allowed cell size $h_{\min}$ for the GRDME is about a factor of $\sqrt{6}$ larger than for the RDME. We verify this experimentally by considering the bimolecular reaction system,

$$\begin{aligned} \emptyset &\xrightarrow{k_1} A, \\ A + A &\xrightarrow{k_2} \emptyset, \end{aligned} \tag{16}$$

with propensities $a_1 = k_1$, $a_2 = \frac{k_2}{2} s_A(s_A - 1)$. The same system has previously been used to demonstrate the limits of validity of the RDME model.[12] We simulate this system for $k_1 = 0.3$ and $k_2 = 0.02$

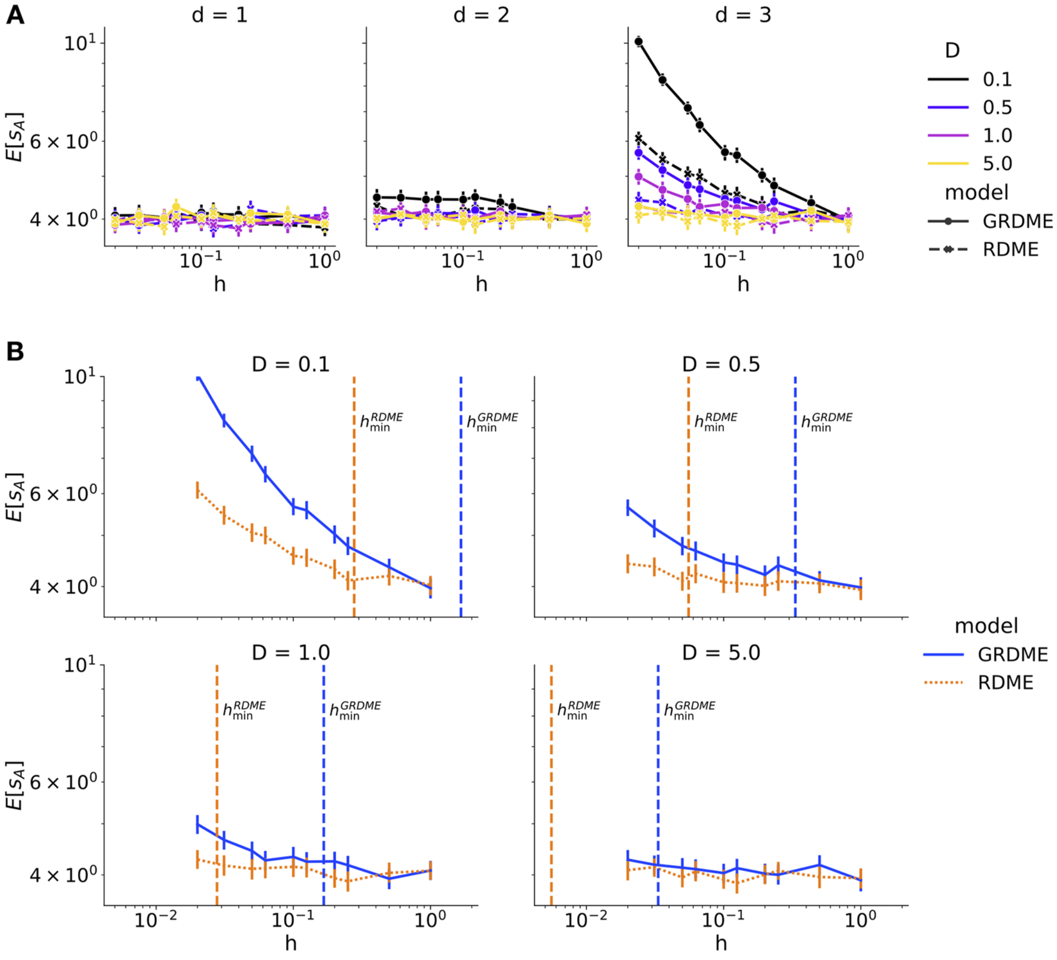

FIG. 4. (a) Expectation value of s_A for the reaction system from Eq. (16) simulated at different grid resolutions h = 0.02, 0.03125, 0.05, 0.0625, 0.1, 0.125, 0.2, 0.25, 0.5, 1.0 and diffusivities D = 0.1, 0.5, 1.0, 5.0 in dimensions d = 1, 2, 3 for 500 repetitions (error bars) of each combination. (b) The curves for d = 3 plotted separately for each diffusivity D. The analytically predicted $h_{\min}$ from Eq. (14) for GRDME and RDME are shown as vertical dashed lines. They are roughly a factor of six apart.

TABLE V. Values of $h_{\min}$ for different diffusivities D for RDME and GRDME with $\varepsilon' = 1$.

D	$h_{\min}$ RDME	$h_{\min}$ GRDME
0.1	0.277 778	1.666 667
0.5	0.055 556	0.333 333
1.0	0.027 778	0.166 667
5.0	0.005 556	0.033 333

and four different diffusivities, $D = 0.1$, 0.5, 1.0, and 5.0. To simplify the analysis, we assume k_2 to be at the ballistic limit of the diffusional propensity[47] and hence to not change with varying D. The Kuramoto lengths $L_K^{\Omega}(D) = \sqrt{2dD|\Omega|/(k_2(\mathbb{E}[s_A] - 1))}$ for the diffusivities used are $L_K^{\Omega}(0.1) = 3.2$, $L_K^{\Omega}(0.5) = 7.2$, $L_K^{\Omega}(1.0) = 10.2$, and $L_K^{\Omega}(5.0) = 22.8$ such that the system is well-mixed in a domain with edge length $L = 1$ and steady state $\mathbb{E}[s_A] = \sqrt{k1/k2} \approx 3.8730$. This ensures that any resolution-dependent loss of bimolecular reactions is unphysical, rather than a true result of the reaction dynamics in a diffusion-limited system. We confirm in the leftmost plot of Fig. 4(a) that both models are indeed consistent for all choices of h in one dimension.

The results in two dimensions are consistent for all choices of h and D except the smallest $D = 0.1$ in GRDME, confirming that θ for the GRDME is larger than for the RDME such that for $D = 0.1$, Eq. (13) is not satisfied anymore. However, we do not see the error in $\mathbb{E}[s_A]$ diverge as $\ln(h)$, as predicted by Isaacson[30] and shown by Hellander *et al.*[33] Further analysis is required to see if this would be that case for smaller h. Another reason could be that here we measure the average number of molecules in the system rather than the average time until the first reaction, and the $h_{\min}$ heuristic has been developed for macroscopic parameters. The results for the same reaction system with $\varepsilon' > 1$ are given in Fig. 7 in Appendix B.

Consistently, both the RDME and GRDME models experience reaction loss in the three-dimensional system at a grid resolution ~10 times larger than the Kuramoto length of a cell. The proportionality constant θ at which this happens is larger for GRDME than for RDME. This case is analyzed in more detail in Fig. 4(b). The dashed vertical lines indicate $h_{\min}$ computed using Eq. (14). The θ coefficient for GRDME is $\theta = 10$, and for RDME, $\theta = 10/\sqrt{6}$. The corresponding values of $h_{\min}$ are given in Table V. While for the present choices of k_2 and D, $h_{\min} < h_{\max}$, further decreasing D or increasing k_2 would lead to $h_{\min} > h_{\max}$ such that there would be no h for which either RDME or GRDME is consistent (see Table VI). The numbers confirm the theoretical prediction of Sec. III that RDME and GRDME experience the same level of loss of bimolecular reactions at grid resolutions approximately six fold apart in three dimensions.

TABLE VI. Values of $h_{\min}$ for different diffusivities D for RDME and GRDME with different $\varepsilon' > 1$.

$h_{\min}$	$D = 0.5$	$D = 5$	$D = 50$
RDME	0.055 556	0.005 556	0.000 556
GRDME, $\varepsilon' = 1.0$	0.333 333	0.033 333	0.003 333
GRDME, $\varepsilon' = 1.5$	0.750 000	0.075 000	0.007 500
GRDME, $\varepsilon' = 2.0$	1.333 333	0.133 333	0.013 333
GRDME, $\varepsilon' = 3.0$	3.000 000	0.300 000	0.030 000
GRDME, $\varepsilon' = 5.0$	8.333 333	0.833 333	0.083 333

Thus, for a given $h_{\min}$, molecules with diffusivity six times higher than the critical diffusivity that determines $h_{\min}$ can be simulated with GRDME, reducing the computational cost of simulations.

VI. CONCLUSIONS AND DISCUSSION

We presented the Gaussian Reaction–Diffusion Master Equation (GRDME) model, which generalizes the Reaction–Diffusion Master Equation (RMDE) to larger diffusion jump neighborhoods. Analogous to how the classic RDME is derived from a finite-difference interpretation of diffusion, we derived the GRDME from the Particle Strength Exchange (PSE) discretization method, providing a straightforward second-order scheme with a larger support. The diffusion transfer rates in the present GRDME model have a Gaussian form, and the diffusion jump neighborhood is extended to nonadjacent cells without introducing additional state dependences. The GRDME in the presented form can therefore be exactly (in the sense that all events are explicitly executed) simulated by an extension of the Next Subvolume Method (NSM) called the Gaussian Next Subvolume Method (GNSM).

Extending the classic RDME model to larger diffusion neighborhoods would require the use of finite-difference stencils with larger support. While such schemes exist (e.g., WENO finite differences[48]), they have solution-dependent weights that cannot be precomputed. We also note that an extension to orders of accuracy larger than two seems impossible, both for the RDME and for the GRDME. This is because any finite-difference stencil and any PSE kernel or order larger than two have to have negative weights on some of the grid cells in order to cancel the odd terms in the Taylor expansion of the operator. In any scenario where the molecular population in a cell is zero, this could lead to negative molecule numbers being computed. It therefore seems that extending the RDME model to larger diffusion neighborhoods, or extending either the RDME or GRDME to higher orders of convergence, is at least difficult if not impossible.

We have shown in numerical experiments that the GRDME diffusion model is of comparable accuracy to the RDME model and displays the expected error convergence with respect to population size and grid resolution. GRDME, however, has smaller total diffusion propensity, which reduces the computational cost of an exact simulation by at most a factor of $2d\varepsilon'^2$ in a d-dimensional space and when using a PSE kernel of normalized width ε'. The presented benchmarks showed that our implementation of GNSM sustained a speedup over NSM that is close to this upper bound. The price to pay for this speedup is a smaller range of length scales in which the model is valid. Indeed, while the maximum admissible resolution $h_{\max}$ is the same for both models, our analysis has shown that $h_{\min}$ of the GRDME is $2d\varepsilon'^2$ times larger than that for the RDME, reflecting a proportional trade-off between computer time and accuracy. Therefore, it might be interesting in the future to develop hybrid approaches, where the diffusion model switches between RDME and GRDME on a per-species basis depending on the diffusion constant of the species, while the reaction parts of the two models remain identical. Modeling rapidly diffusing species with GRDME and slowly diffusing ones with RDME could still achieve lower computational cost overall, while maintaining the larger validity range of grid resolutions of the RDME.

An important limitation of both the RDME and the GRDME models is that they are derived under the assumption of dilute molecular populations in each grid cell. While this is mostly not the case in biological cells, we agree with previously established arguments[29,31] that RDME and, therefore, also GRDME are valid models if (i) the molecules undergo normal, Brownian diffusion, (ii) the *reactant* molecules of all modeled reactions are sufficiently dilute, (iii) $h_{\text{min}} < h_{\text{max}}$, and (iv) the diffusional bimolecular reaction propensity[47] is applicable. Under these conditions, the crowders mainly increase the effective viscosity of the medium, but do not participate in the chemical kinetics modeled. This must always be carefully decided on a case-by-case basis before deciding on whether to use any master-equation model. For example, macromolecular crowding can reduce diffusivity and increase bimolecular reaction rates,[49] possibly violating requirement (iii). Excluded volume effects may violate requirement (iv). Highly concentrated systems as seen in liquid–liquid phase separation[50] may violate all four requirements. In such cases, additional microscopic detail is required and, thus, modeling frameworks, such as Brownian dynamics,[51] vRDME,[52] or SPT-RDME,[53] should be used. In many cases, however, solutes display normal diffusion[54] and reactants are rare in comparison to inert molecules. Then, master-equation models, such as the RDME or GRDME, could be appropriate, provided that requirement (iii) holds, which is easy to verify for any given system using the known expressions for h_{min} and h_{max}.

Taken together, the GRDME is comparably accurate as the classic RDME, but permits faster exact simulations with a suitably extended version of the NSM algorithm. Due to its larger minimum grid cell size, it is best suited for systems with sufficiently high diffusivity, and the computational savings are largest for three-dimensional systems.

ACKNOWLEDGMENTS

We thank Professor Andreas Deutsch (TU Dresden), Ankit Gupta (ETH Zurich), and the MOSAIC group (TU Dresden, MPI-CBG, and CSBD) for helpful discussions and advise on our work. This work was supported, in part, by the German Research Foundation (Deutsche Forschungsgemeinschaft, DFG) under Grant No. SB/350008342 (project "OpenPME").

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Tina Subic: Data curation (lead); Formal analysis (lead); Investigation (equal); Methodology (equal); Software (lead); Validation (equal); Visualization (lead); Writing – original draft (lead); Writing – review & editing (supporting). **Ivo F. Sbalzarini**: Conceptualization (lead); Funding acquisition (lead); Investigation (equal); Methodology (equal); Project administration (lead); Resources (lead); Supervision (lead); Validation (equal); Visualization (supporting); Writing – original draft (supporting); Writing – review & editing (lead).

DATA AVAILABILITY

The code and input files to reproduce the results shown in this paper, as well as a reference implementation of GNSM, are provided as open-source at https://git.mpi-cbg.de/mosaic/GRDME.

APPENDIX A: ALGORITHMS FOR $\mathcal{N}$

Algorithm 3 constructs an array MS of all multisets m of cardinality d from the set $\{0, 1, \ldots, \lceil l_{\text{max}} \rceil\}$. MS additionally contains the radius l of the coordinates, which will be built from the multisets m, and the number of multiset permutations of each m, nmp. A multiset allows several instances of each element. The number of instances of an element is called its *multiplicity*. Multisets containing elements with large multiplicity have a smaller number of permutations. The array "multiplicity" of size K stores the largest multiplicity of each m in MS. Line 1 of Algorithm 3 computes K, which is the number of elements in MS. Lines 2 and 3 allocate arrays MS and multiplicity of size K and initialize all values to 1. The multisets m are constructed in the nested loops starting in lines 5 through 7. The inner loops progress until they reach the value of the iterator of the loop within which they are nested. The outermost loop iterates until $\lceil l_{\text{max}} \rceil$. In line 8, the multiset m is set to the values of the iterators x, y, and z. The radius l is computed in line 9. At the end of each inner loop, the value of the iterator of the inner loop and the iterator of the outer loop are the same, increasing the multiplicity value of m in line 12. The final multiplicities are computed in lines 15–18. The multiplicity of the elements is used to compute the number

Algorithm 3. Construct all multisets of cardinality d from the set $\{0, 1, \ldots, \lceil l_{\text{max}} \rceil\}$

```
 1: $K = \binom{\lceil l_{\text{max}} \rceil + d}{d}$
 2: Initialize an array MS of $K$ elements $(m, l, \text{nmp})$, setting all to 1
 3: Initialize an array multiplicity of $K$ integers, setting all to 1
 4: $x = y = z = 0, i = 0$
 5: for $x = 0, x \le \lceil l_{\text{max}} \rceil$ do
 6:     for $y = 0, y \le x$ do
 7:         for $z = 0, z \le y$ do
 8:             $\text{MS}[i].m = [x, y, z]$
 9:             $\text{MS}[i].l = \sqrt{x^2 + y^2 + z^2}$
10:             $i = i + 1$
11:         end for
12:         $\text{Multiplicity}[i-1] = \text{Multiplicity}[i-1] + 1$
13:         $\text{MS}[i-1].\text{nmp} = \text{MS}[i-1].\text{nmp} \cdot \text{Multiplicity}[i-1]$
14:     end for
15:     for $j = i - 1 - x, j \le i - 1$ do
16:         $\text{Multiplicity}[j] = \text{Multiplicity}[j] + 1$
17:         $\text{MS}[j].\text{nmp} = \text{MS}[j].\text{nmp} \cdot \text{Multiplicity}[j]$
18:     end for
19: end for
20: for all ms in MS do
21:     $\text{ms. nmp} = \frac{d!}{\text{ms. nmp}}$
22: end for
```

Algorithm 4. Find all permutations of a multiset of cardinality *d*.

```
Require: ms
 1: $\delta = 1$
 2: initialize ms.nmp sized array MP of permutations of ms.$m$
 3: MP.append($m$)
 4: P1 = P2 = $m$
 5: if $m[1] = m[2]$ then
 6:     $\delta = -1$
 7: end if
 8: for $i = 0, i < \text{ms.nmp} - 1$ do
 9:     P1 = P2
10:     P1[1] = P2[$1 + \delta$]
11:     P1[$1 + \delta$] = P2[1]
12:     $\delta = -\delta$
13:     MP.append(P1)
14: end for
```

of multiset permutations for each m, required as an input for Algorithm 4.

Algorithm 4 creates an array MP from all multiset permutations of a given element ms in MS. The first element in MP is initialized with the input sequence in m (line 3). Permutations are created by swapping the position of the temporary arrays P1 and P2 nmp times using multipliers $\delta \in \{-1, 1\}$ (lines 8 to 14), starting with $\delta = 1$. In the special case where the second and third element of m are the same, δ is initialized to -1 (lines 5–7). At each iteration, δ is mirrored to $-\delta$ (line 12), and the value of P1 is appended to MP (line 13).

Algorithm 5 creates an array SC of all permutations of the sign values of the x, y, and z coordinates of a given permutation mp of a multiset m. The number of such permutations depends on the value of mp and is computed in lines 3–7: only nonzero elements have two sign versions. The permutations are created in the nested for loops in lines 10 through 22 by multiplying each element in mp with the sign value (lines 13 to 15). At the end of each loop, the sign along the axis over which the loop iterated is flipped (lines 17, 19, and 21).

Algorithm 5. Compute all sign permutations of mp.

```
Require: mp
 1: $i = 0$, sign$x$ = sign$y$ = sign$z$ = 1
 2: nsign = [1, 1, 1]
 3: for ax = 0, ax < $d$ do
 4:     if |mp.$m$[ax]| > 0 then
 5:         nsign[ax] = 2
 6:     end if
 7: end for
 8: n = nsign[0] · nsign[1] · nsign[2]
 9: create an array SC of size n containing sequences $[x, y, z]$
10: for $x = 0, x <$ nsign[0] do
11:     for $y = 0, y <$ nsign[1] do
12:         for $z = 0, z <$ nsign[2] do
13:             SC[$i$][0] = mp.$m$[0] · sign$x$
14:             SC[$i$][1] = mp.$m$[1] · sign$y$
15:             SC[$i$][2] = mp.$m$[2] · sign$z$
16:             $i = i + 1$
17:             sign$z$ = −sign$z$
18:         end for
19:         sign$y$ = −sign$y$
20:     end for
21:     sign$x$ = −sign$x$
22: end for
```

APPENDIX B: NUMERICAL EXPERIMENTS FOR $\varepsilon' > 1$

The normalized (to the grid spacing) PSE kernel width ε' is a free parameter in the GRDME model. It can be used to tune between accuracy, resolution requirement, and computational cost of the model and its exact simulation. Our theoretical analysis shows that the error of the diffusion model scales as $O(h^2\varepsilon'^2)$. The computational cost of GNSM is up to a factor of $2d\varepsilon'^2$ times lower than that of NSM, but the lower grid resolution limit of GRDME is $2d\varepsilon'^2$ times larger than that of the RDME. We empirically confirm this by repeating the numerical experiments from Sec. V for different $\varepsilon' > 1$.

1. Probability distribution moments

We repeat the measurements of the L_2 error in the first four moments of the spatial distribution of a population of diffusing molecules from Sec. V A 2 for $\varepsilon' = 1, 1.5, 2, 3, 5$ for varying N and h using the same experimental conditions as in Sec. V A 2. The expected convergence order of the error in the moments is $O(N^{-1/2}) + O(h^2\varepsilon'^2)$ with increasing particle number N and decreasing lattice cell size h and kernel width ε', reducing the error. We expect the error to increase quadratically with increasing ε' for fixed N and h. Figures 5(a) and 5(b) show the results. As expected, increasing ε' increases the error in the moments of the distribution. For large ε', the discretization error becomes the dominant term even for small N [Fig. 5(a)]. Indeed, when both ε' and h are large (for $h = 5$ and $\varepsilon' = 5$, $\varepsilon = 25$), the discretization error also dominates for the second and third moment [Fig. 5(b)]. Figure 5(c) shows the dependence of the L_2 error on ε' for $N = 10^7$ and different h. As long as the discretization error dominates, the L_2 error increases quadratically with increasing ε', as expected from theory. For comparison, the L_2 errors for RDME are indicated by dots. The dots are placed at $\varepsilon' = 1/\sqrt{6}$, which is the value GRDME would have to use in order to match the accuracy of the RDME model. This value was found by extrapolation of the numerical results and not through theoretical error analysis. Since $\varepsilon' \geq 1$ by the overlap condition of PSE (see Sec. II), the GRDME diffusion model can never match the accuracy of the RDME diffusion model when using a Gaussian PSE kernel.

2. Computational cost

The computational cost of GNSM can be up to $2d\varepsilon'^2$ times smaller than that of NSM. However, the computation of each diffusion event is costlier in GNSM than in NSM. Since the average number of diffusion events decreases for larger kernels (i.e., larger ε'), we expect that increasing ε' further reduces the computational cost of GNSM compared to NSM. Since the neighborhood to be searched for each individual diffusion jump grows with ε'^2, we also expect that the practically achieved speedup will be farther from the theoretically possible $2d\varepsilon'^2$ for larger ε'. Figure 6(a) shows the measured computational costs of NSM and GNSM for $\varepsilon' = 1, 1.5, 2, 3, 5$ and $d = 1, 2, 3$. As expected, increasing ε' increases the speedup of

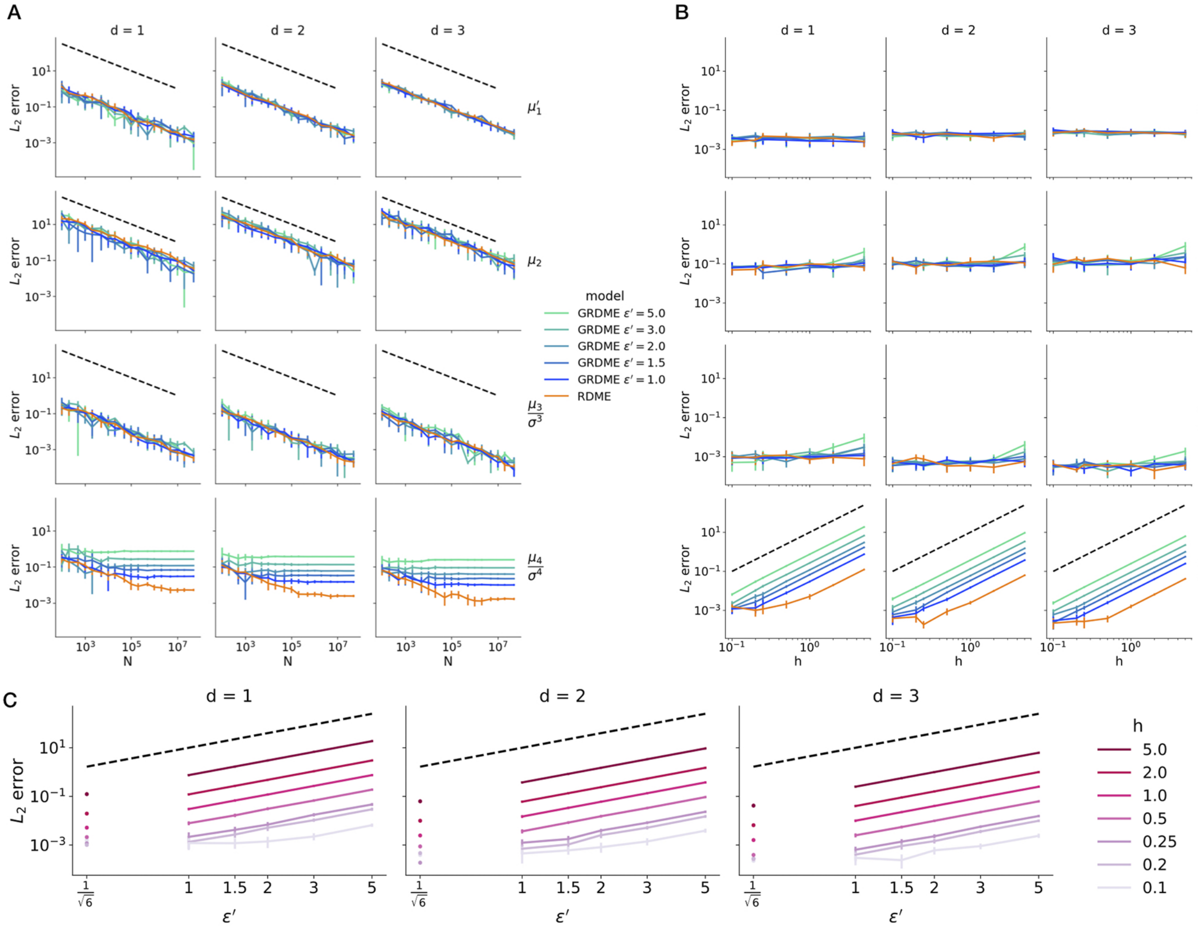

FIG. 5. L_2 error convergence of the first four moments μ_1', μ_2, μ_3/σ^3, and μ_4/σ^4 in dimensions $d = 1, 2, 3$ for the RDME diffusion model and the GRDME diffusion model with $\varepsilon' = 1, 1.5, 2, 3, 5$. (a) Convergence with respect to the number of molecules N for RDME (orange) and GRDME (green–blue and darker shades indicate smaller ε') for point-source diffusion in an open domain with $h = 1$ and $D = 1$ at final time $T = 100$. The black dashed lines indicate the theoretically expected convergence order $-1/2$. (b) Convergence with respect to the lattice resolution h for RDME (orange) and GRDME (green–blue and darker shades indicate smaller ε') for the same problem with $N = 10^7$ and $D = 1$ at time $T = 100$. The black dashed lines indicate the theoretically expected convergence order 2. (c) Convergence with respect to the normalizes kernel width $\varepsilon' = \varepsilon/h$ for the GRDME with different lattice resolutions h (darker hue indicates larger h), $N = 10^7$, $D = 1$, and $T = 100$. The dots show the L_2 errors for RDME for the same h, indicated at the value of $\varepsilon' = 1/\sqrt{6}$; the GRDME model *would have to* be use in order to achieve the same error. The black dashed lines indicate the theoretically expected convergence order 2.

GNSM over NSM. However, as shown in Fig. 6(b), increasing ε' too much leads away from the quadratic speedup predicted by theory, as seen for $d = 3$ and $\varepsilon' = 5$. Nonetheless, GNSM with $\varepsilon' = 5$ is ~70-fold faster than NSM in three dimensions.

3. Limits of validity for a homo-bimolecular reaction

The lower limit on the grid resolution, $h_{\min}$, is also a function of ε'. In Sec. III, we have shown that one-dimensional systems should not possess a limit on h. Two- and three-dimensional systems, however, must have a sufficiently large ratio of diffusivity over reaction rate. For two-dimensional systems for which this ratio is satisfied, we expect there to be no lower limit on grid resolution, whereas $h_{\min}$ always exists for three-dimensional systems.

We repeat the numerical experiment from Sec. V C using the same reaction network and reaction parameters, but using the diffusion coefficients $D = 0.5, 5, 50$, resulting in $L_K^{\Omega}(0.5) = 7.2$, $L_K^{\Omega}(5) = 22.8$, and $L_K^{\Omega}(50) = 72.3$. The results in the top row of Fig. 7 imply that, indeed, there is no lower limit on the grid resolution for $d = 1$ even for the GRDME models with $\varepsilon' > 1$. The results in the middle row of Fig. 7 show that a two-dimensional system *can* experience unphysical loss of bimolecular reactions if the propensity of the diffusion model is too small compared to the propensity of the bimolecular reactions, confirming the results in Fig. 4(a). The bottom row confirms the existence of $h_{\min}$ for a three-dimensional system, where the loss of bimolecular reactions is apparent for small diffusivity and increases with increasing ε'. The numerical values for this case are given in Table VI. Indeed, for $D = 0.5$ and $\varepsilon' = 5$,

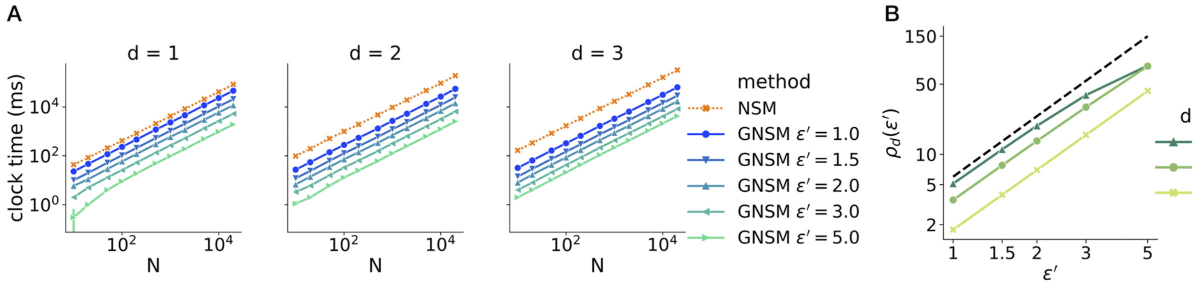

FIG. 6. (a) Total runtimes of simulations of diffusion in a bounded domain (see Sec. V B for the simulation setup) with N molecules for $d = 1, 2, 3$, simulated using NSM (orange dotted lines) and GNSM with $\varepsilon' = 1, 1.5, 2, 3, 5$ (solid lines in shades of green–blue and darker shades indicate smaller ε'). The error bars over 10 independent repetitions of each simulation are smaller than the symbol size. (b) The speedups $\rho_d(\varepsilon') = \text{time}(\text{NSM})/\text{time}(\text{GNSM}(\varepsilon'))$ plotted vs ε'. The black dashed line indicates that $\rho_3 = 6\varepsilon'^2$ indicates the upper bound of the speedup in three dimensions $(d = 3)$. The error bars over 10 independent repetitions of each simulation are smaller than the symbol size.

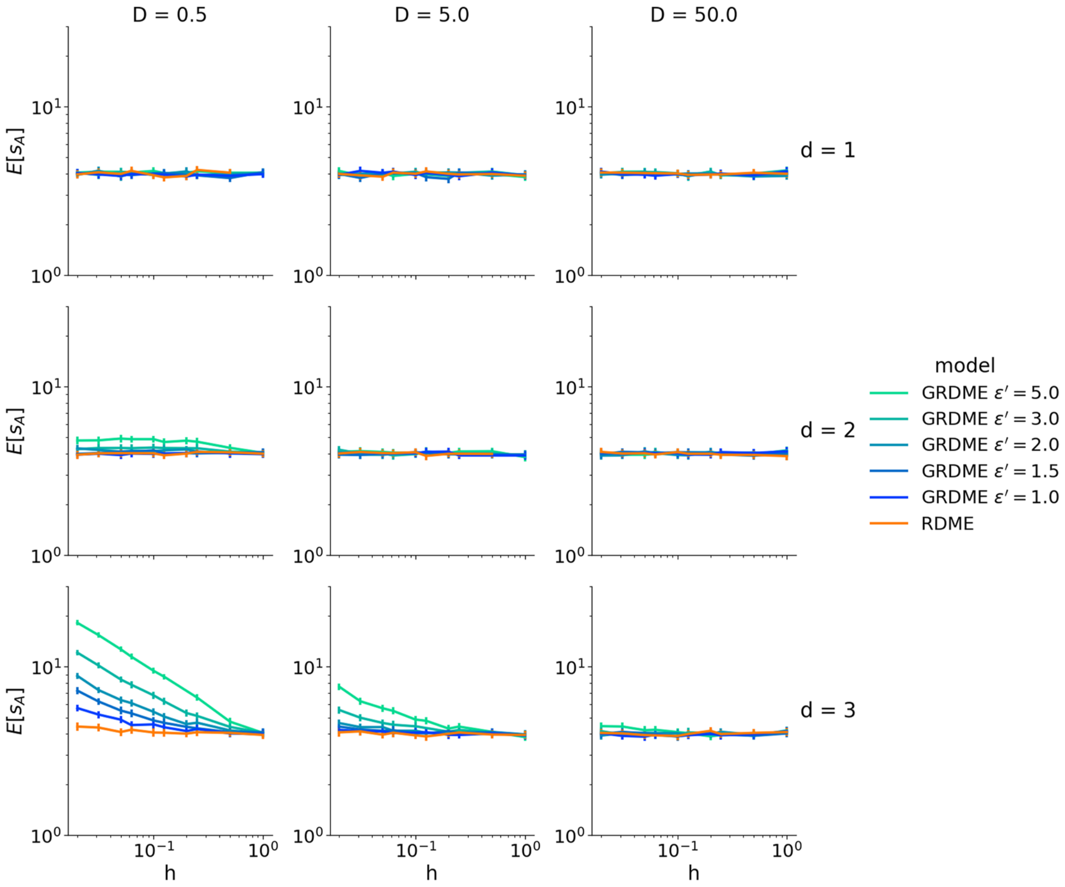

FIG. 7. Expectation value of s_A for the reaction system from Eq. (16) simulated at different grid resolutions $h = 0.02, 0.03125, 0.05, 0.0625, 0.1, 0.125, 0.2, 0.25, 0.5, 1.0$ and diffusivities $D = 0.5$, 5, 50 in dimensions $d = 1, 2, 3$. To simulate diffusion, we used both the RDME model (orange) and GRDME models with $\varepsilon' = 1$, 1.5, 2, 3, 5 (shades of green–blue and darker shades indicate smaller ε'). The error bars show the standard deviations of the measurements over 500 independent repetitions of each combination.

there exists no h for which the model would be consistent since $h_{\min} > h_{\max} = L_K^{\Omega}$. However, any system with sufficiently large diffusion coefficients can be, in principle, modeled by a GRDME with larger ε' since the increase in $h_{\min}$ with increasing $\varepsilon' > 1$ becomes more and more negligible for larger D.

REFERENCES

[1] D. Fange and J. Elf, "Noise-induced min phenotypes in E. coli," PLOS Comput. Biol. **2**, e80 (2006).

[2] T. Freisinger, B. Klünder, J. Johnson, N. Müller, G. Pichler, G. Beck, M. Costanzo, C. Boone, R. A. Cerione, E. Frey, and R. Wedlich-Söldner, "Establishment of a robust single axis of cell polarity by coupling multiple positive feedback loops," Nat. Commun. **4**, 1807 (2013).

[3] N. W. Goehring, P. K. Trong, J. S. Bois, D. Chowdhury, E. M. Nicola, A. A. Hyman, and S. W. Grill, "Polarization of PAR proteins by advective triggering of a pattern-forming system," Science **334**, 1137 (2011).

[4] M. Onsum and C. V. Rao, "A mathematical model for neutrophil gradient sensing and polarization," PLoS Comput. Biol. **3**, e36 (2007).

[5] S. J. Altschuler, S. B. Angenent, Y. Wang, and L. F. Wu, "On the spontaneous emergence of cell polarity," Nature **454**, 886 (2008).

[6] K. R. Duffy, C. J. Wellard, J. F. Markham, J. H. S. Zhou, R. Holmberg, E. D. Hawkins, J. Hasbold, M. R. Dowling, and P. D. Hodgkin, "Activation-induced B cell fates are selected by intracellular stochastic competition," Science **335**, 338 (2012).

[7] J. Paulsson, O. G. Berg, and M. Ehrenberg, "Stochastic focusing: Fluctuation-enhanced sensitivity of intracellular regulation," Proc. Natl. Acad. Sci. U. S. A. **97**, 7148 (2000).

[8] M. Sturrock, A. Hellander, A. Matzavinos, and M. A. J. Chaplain, "Spatial stochastic modelling of the Hes1 gene regulatory network: Intrinsic noise can explain heterogeneity in embryonic stem cell differentiation," J. R. Soc., Interface **10**, 20120988 (2013).

[9] K. Takahashi, S. Tănase-Nicola, and P. R. ten Wolde, "Spatio-temporal correlations can drastically change the response of a MAPK pathway," Proc. Natl. Acad. Sci. U. S. A. **107**, 2473–2478 (2010).

[10] R. Ramaswamy, N. González-Segredo, I. F. Sbalzarini, and R. Grima, "Discreteness-induced concentration inversion in mesoscopic chemical systems," Nat. Commun. **3**, 779 (2012).

[11] C. W. Gardiner, K. J. McNeil, D. F. Walls, and I. S. Matheson, "Correlations in stochastic theories of chemical reactions," J. Stat. Phys. **14**, 307 (1976).

[12] S. Smith and R. Grima, "Spatial stochastic intracellular kinetics: A review of modelling approaches," Bull. Math. Biol. **81**, 2960 (2019).

[13] D. T. Gillespie, "Exact stochastic simulation of coupled chemical reactions," J. Phys. Chem. **81**, 2340 (1977).

[14] J. Elf and M. Ehrenberg, "Spontaneous separation of bi-stable biochemical systems into spatial domains of opposite phases," Syst. Biol. **1**, 230 (2004).

[15] M. A. Gibson and J. Bruck, "Efficient exact stochastic simulation of chemical systems with many species and many channels," J. Phys. Chem. A **104**, 1876 (2000).

[16] T. T. Marquez-Lago and K. Burrage, "Binomial tau-leap spatial stochastic simulation algorithm for applications in chemical kinetics," J. Chem. Phys. **127**, 104101 (2007).

[17] S. Lampoudi, D. T. Gillespie, and L. R. Petzold, "The multinomial simulation algorithm for discrete stochastic simulation of reaction-diffusion systems," J. Chem. Phys. **130**, 094104 (2009).

[18] J. Fu, S. Wu, H. Li, and L. R. Petzold, "The time dependent propensity function for acceleration of spatial stochastic simulation of reaction–diffusion systems," J. Comput. Phys. **274**, 524 (2014).

[19] L. Ferm, A. Hellander, and P. Lötstedt, "An adaptive algorithm for simulation of stochastic reaction–diffusion processes," J. Comput. Phys. **229**, 343 (2010).

[20] P. R. Taylor, R. E. Baker, and C. A. Yates, "Deriving appropriate boundary conditions, and accelerating position-jump simulations, of diffusion using non-local jumping," Phys. Biol. **12**, 016006 (2014).

[21] S. A. Isaacson and Y. Zhang, "An unstructured mesh convergent reaction–diffusion master equation for reversible reactions," J. Comput. Phys. **374**, 954 (2018).

[22] J. S. van Zon and P. R. ten Wolde, "Green's-function reaction dynamics: A particle-based approach for simulating biochemical networks in time and space," J. Chem. Phys. **123**, 234910 (2005).

[23] L. Meinecke and P. Lötstedt, "Stochastic diffusion processes on cartesian meshes," J. Comput. Appl. Math. **294**, 1 (2016).

[24] P. Degond and S. Mas-Gallic, "The weighted particle method for convection-diffusion equations. Part 1: The case of an isotropic viscosity," Math. Comput. **53**, 485 (1989).

[25] S. Engblom, L. Ferm, A. Hellander, and P. Lötstedt, "Simulation of stochastic reaction-diffusion processes on unstructured meshes," SIAM J. Sci. Comput. **31**, 1774 (2009).

[26] P. Lötstedt and L. Meinecke, "Simulation of stochastic diffusion via first exit times," J. Comput. Phys. **300**, 862 (2015).

[27] L. Meinecke, S. Engblom, A. Hellander, and P. Lötstedt, "Analysis and design of jump coefficients in discrete stochastic diffusion models," SIAM J. Sci. Comput. **38**, A55 (2016).

[28] J. D. Eldredge, A. Leonard, and T. Colonius, "A general deterministic treatment of derivatives in particle methods," J. Comput. Phys. **180**, 686 (2002).

[29] S. Smith and R. Grima, "Breakdown of the reaction-diffusion master equation with nonelementary rates," Phys. Rev. E **93**, 052135 (2016).

[30] S. A. Isaacson, "The reaction-diffusion master equation as an asymptotic approximation of diffusion to a small target," SIAM J. Appl. Math. **70**, 77 (2009).

[31] D. T. Gillespie, L. R. Petzold, and E. Seitaridou, "Validity conditions for stochastic chemical kinetics in diffusion-limited systems," J. Chem. Phys. **140**, 054111 (2014).

[32] S. Hellander, A. Hellander, and L. Petzold, "Reaction rates for mesoscopic reaction-diffusion kinetics," Phys. Rev. E **91**, 023312 (2015).

[33] S. Hellander, A. Hellander, and L. Petzold, "Reaction-diffusion master equation in the microscopic limit," Phys. Rev. E **85**, 042901 (2012).

[34] R. Erban and S. J. Chapman, "Stochastic modelling of reaction–diffusion processes: Algorithms for bimolecular reactions," Phys. Biol. **6**, 046001 (2009).

[35] D. Fange, O. G. Berg, P. Sjöberg, and J. Elf, "Stochastic reaction-diffusion kinetics in the microscopic limit," Proc. Natl. Acad. Sci. U. S. A. **107**, 19820 (2010).

[36] A. Hellander, S. Hellander, and P. Lötstedt, "Coupled mesoscopic and microscopic simulation of stochastic reaction-diffusion processes in mixed dimensions," Multiscale Model. Simul. **10**, 585 (2012).

[37] S. A. Isaacson, "A convergent reaction-diffusion master equation," J. Chem. Phys. **139**, 054101 (2013).

[38] R. Grima and S. Schnell, "Modelling reaction kinetics inside cells," Essays Biochem. **45**, 41 (2008).

[39] This expression for hmin holds for the general diffusion rate $k = \frac{k_b k_d}{k_b + k_d}$, where k_b describes the dynamics at the site of collision and $k_d = 4\pi\sigma$ is the diffusional rate.[48] In the diffusion limit of the reaction rate, $h_{\min}$ can be expressed in terms of the reaction radius σ, for $d = 3$,

$$h_{\min} = \frac{4\pi\sigma\hat{D}\theta^2}{6\hat{D}} \approx 2\theta^2\sigma,$$

which is similar to the expression for $h_{\min}$ given in Ref. 31.

[40] We note that it is also possible to use such a neighborhood structure in NSM, instead of a connectivity matrix. While this would reduce the memory requirements of NSM, it would increase its computational cost since the information would need to be computed on the fly while executing a diffusion jump. Since NSM executes many more diffusion jumps than GNSM, the connectivity matrix seems the better choice for NSM.

[41] G. Guennebaud, B. Jacob *et al.*, Eigen v3, http://eigen.tuxfamily.org, 2010.

[42] Boost, Boost C++ Libraries, http://www.boost.org/, 2022.

[43] The Pandas Development Team, pandas-dev/pandas: Pandas, 2020.

[44] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus,

S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, "Array programming with NumPy," Nature **585**, 357 (2020).

[45] J. D. Hunter, "Matplotlib: A 2D graphics environment," Comput. Sci. Eng. **9**, 90 (2007).

[46] M. L. Waskom, "seaborn: statistical data visualization," J. Open Source Software **6**, 3021 (2021).

[47] D. T. Gillespie, "A diffusional bimolecular propensity function," J. Chem. Phys. **131**, 164109 (2009).

[48] X.-D. Liu, S. Osher, and T. Chan, "Weighted essentially non-oscillatory schemes," J. Comput. Phys. **115**, 200–212 (1994).

[49] R. J. Ellis, "Macromolecular crowding: Obvious but underappreciated," Trends Biochem. Sci. **26**, 597 (2001).

[50] S. F. Banani, H. O. Lee, A. A. Hyman, and M. K. Rosen, "Biomolecular condensates: Organizers of cellular biochemistry," Nat. Rev. Mol. Cell Biol. **18**, 285 (2017).

[51] G. A. Huber and J. A. McCammon, "Brownian dynamics simulations of biological molecules," Trends Chem. **1**, 727 (2019).

[52] C. Cianci, S. Smith, and R. Grima, "Molecular finite-size effects in stochastic models of equilibrium chemical systems," J. Chem. Phys. **144**, 084101 (2016).

[53] C. Cianci, S. Smith, and R. Grima, "Capturing Brownian dynamics with an on-lattice model of hard-sphere diffusion," Phys. Rev. E **95**, 052118 (2017).

[54] J. A. Dix and A. S. Verkman, "Crowding effects on diffusion in solutions and cells," Annu. Rev. Biophys. **37**, 247 (2008).

[55] B. Drawert, S. Engblom and A. Hellander, "URDME: a modular framework for stochastic simulation of reaction-transport processes in complex geometries," BMC Syst. Biol. **6**(1), 76 (2012).