Reaction rates for reaction-diffusion kinetics on unstructured meshes

Stefan Hellander and Linda Petzold
Department of Computer Science, University of California, Santa Barbara, California 93106-5070, USA

(Received 23 September 2016; accepted 17 January 2017; published online 8 February 2017)

The reaction-diffusion master equation is a stochastic model often utilized in the study of biochemical reaction networks in living cells. It is applied when the spatial distribution of molecules is important to the dynamics of the system. A viable approach to resolve the complex geometry of cells accurately is to discretize space with an unstructured mesh. Diffusion is modeled as discrete jumps between nodes on the mesh, and the diffusion jump rates can be obtained through a discretization of the diffusion equation on the mesh. Reactions can occur when molecules occupy the same voxel. In this paper, we develop a method for computing accurate reaction rates between molecules occupying the same voxel in an unstructured mesh. For large voxels, these rates are known to be well approximated by the reaction rates derived by Collins and Kimball, but as the mesh is refined, no analytical expression for the rates exists. We reduce the problem of computing accurate reaction rates to a pure preprocessing step, depending only on the mesh and not on the model parameters, and we devise an efficient numerical scheme to estimate them to high accuracy. We show in several numerical examples that as we refine the mesh, the results obtained with the reaction-diffusion master equation approach those of a more fine-grained Smoluchowski particle-tracking model. *Published by AIP Publishing.* [http://dx.doi.org/10.1063/1.4975167]

I. INTRODUCTION

Spatial stochastic modeling is a tool frequently used to study biochemical reaction networks in cells where the spatial distribution of molecules is non-uniform.[1–5] For instance, reactions can be localized to only a few sites, or molecules may take part in a sequence of reactions where the spatial correlation of the newly created molecules affects the dynamics of the whole system.[6–8]

The models considered when studying systems on the scale of a living cell are often divided into three levels: the macroscopic level, the mesoscopic level, and the microscopic level. On the macroscopic level the system is modeled by a deterministic partial differential equation (PDE). On the mesoscopic level, the system is modeled by the reaction-diffusion master equation (RDME). Exact trajectories can be generated by the next subvolume method (NSM),[9] in which molecules diffuse between voxels at some given intensity, and react at some given intensity when occupying the same voxel. On the microscopic level, we track the continuous position and movement of individual molecules, and the molecules can react when they are sufficiently close. In this paper we are concerned only with the mesoscopic and microscopic levels.

The RDME is a popular model, evidenced by the number of simulation tools available. While a less detailed model than microscopic models, it has the advantage of being orders of magnitude faster for many biologically relevant problems. Software packages implementing solvers for the RDME include MesoRD,[10] PyURDME, StochSS (http://www.stochss.org), NeuroRD,[11] E-Cell,[12] and STEPS.[13] Microscale simulations are suitable when high accuracy is needed, if we need to simulate very diffusion-limited reactions, or if reactions are localized near or on a complex geometrical structure. Software packages implementing microscale solvers include Smoldyn,[14] MCell,[15] and E-Cell.[12]

Many realistic biological systems exist within complex geometries, and a tractable approach to resolving complex geometries is to discretize space using an unstructured mesh. This approach was studied in Ref. 16, in which the authors devise a method to obtain accurate diffusive jump rates. The problem of obtaining accurate reaction rates on Cartesian meshes is well studied, see, e.g., Refs. 7, 8, and 17–19, but the problem of obtaining accurate reaction rates for a wide range of voxel sizes for unstructured meshes has not been as thoroughly studied. Isaacson derives a convergent RDME in Ref. 20 where the RDME is extended to allow nonlocal reactions and where convergence is to the microscale model proposed by Doi in Ref. 21. Here we consider local reactions only and the Smoluchowski microscale model where molecules are modeled by hard spheres[22] and react according to a Robin boundary condition.[23] Most of our results could be extended to different microscale models.

It is easy to see that for "large enough" voxels on Cartesian meshes we get reaction rates that are simply the effective rate scaled by the volume of the voxel,[18] and it is reasonable to assume that the same will be true on unstructured meshes. We show in Sec. IV that this is indeed true, but also that if the reactions are diffusion limited this may only be true for quite large voxels (analogous to the Cartesian case), and simulations will be inaccurate as we refine the mesh. A desirable property of numerical methods is convergence as we refine the mesh, at least up to some well-defined maximum spatial resolution.[17–19]

One approach to get around this problem is to consider hybrid methods, where mesoscopic simulations can be performed on fairly coarse meshes for most species, combined with accurate microscopic simulations for some species whose dynamics need to be simulated at a high spatial resolution.[24–27]

In this paper we instead develop a method to efficiently compute accurate reaction rates for the RDME on unstructured meshes for a wide range of voxel sizes and for diffusion-limited reactions, thus facilitating accurate simulations of biochemical systems in complex geometries. It was shown in Refs. 17–19 that the common approach of simply scaling the bimolecular reaction rates by the volume of the voxels leads to inaccurate results for some systems on Cartesian meshes. We show here that, as expected, the same holds true on unstructured meshes, and that by following the approach outlined in this paper, the RDME, for many problems, converges to the corresponding microscale results.

The remainder of the paper is organized as follows. In Sec. II we introduce the mesoscopic and microscopic modeling frameworks. In Sec. III we describe a method to compute mesoscopic reaction rates, and in Sec. IV we show with numerical examples first how to choose the parameters of the method and then that the method itself yields accurate reaction rates leading to convergence of the RDME.

II. BACKGROUND

In this section we introduce the mesoscopic RDME and the microscopic Smoluchowski models. Throughout this paper, we consider the accuracy of the RDME relative to the Smoluchowski model, but note that most of the derivations in Sec. III would be valid also for a different microscale model; we select a specific model for convenience.

As we will see in Sec. III, to derive accurate reaction rates for the RDME it is useful to consider individual trajectories of the stochastic system instead of the full probability distribution. In addition to introducing the full models, we therefore also introduce methods for generating exact trajectories of a system.

A. Mesoscopic simulations

Consider a volume Ω divided into N non-overlapping voxels, and a reaction network consisting of S species. We denote the $N \times S$ state matrix by $\mathbf{x}$, where the ith row, $\mathbf{x}_{i\cdot}$, gives the species copy numbers of voxel i, while the jth column, $\mathbf{x}_{\cdot j}$, gives the copy numbers of species j for each voxel. Assume that the system has M reactions. We denote the propensity function for reaction r in voxel i by $a_{ir}(\mathbf{x}_{i\cdot})$, and the stoichiometry vector for reaction r in voxel i by $\boldsymbol{\mu}_{ir}$. We denote the propensity function for a diffusive jump by species j from voxel i to k by $d_{ijk}(\mathbf{x}_{i\cdot})$, and the associated stoichiometry vector by $\boldsymbol{\nu}_{ijk}$.

Let $p(\mathbf{x}, t)$ be the probability of the system to be in state $\mathbf{x}$ at time t. The RDME describes the time evolution of $p(\mathbf{x}, t)$,

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}p(\mathbf{x},t) = &\sum_{i=1}^{N}\sum_{r=1}^{M} a_{ir}(\mathbf{x}_{i\cdot}-\boldsymbol{\mu}_{ir})p(\mathbf{x}_{1\cdot},\ldots,\mathbf{x}_{i\cdot}-\boldsymbol{\mu}_{ir},\ldots,\mathbf{x}_{N\cdot},t) - \sum_{i=1}^{N}\sum_{r=1}^{M} a_{ir}(\mathbf{x}_{i\cdot})p(\mathbf{x},t)\\ &+\sum_{j=1}^{M}\sum_{i=1}^{N}\sum_{k=1}^{N} d_{jik}(\mathbf{x}_{\cdot j}-\boldsymbol{\nu}_{ijk})p(\mathbf{x}_{\cdot 1},\ldots,\mathbf{x}_{\cdot j}-\boldsymbol{\nu}_{ijk},\ldots,\mathbf{x}_{\cdot S},t) - \sum_{j=1}^{M}\sum_{i=1}^{N}\sum_{k=1}^{N} d_{ijk}(\mathbf{x}_{\cdot j})p(\mathbf{x},t).\end{aligned} \tag{1}$$

The RDME is in general too high-dimensional to be solved by direct methods; a common approach is instead to generate statistics of the system using a Monte Carlo scheme. Exact trajectories of the RDME can be generated as follows. A molecule with diffusion rate D can jump to adjacent voxels at intensity γ_D, where $\gamma_D = 2dD/h^2$ on a Cartesian mesh in d dimensions where h is the width of the voxels, and where γ_D can be determined from, e.g., a finite element discretization in the case of unstructured meshes.[16] Molecules can undergo unimolecular and bimolecular reactions; a pair of molecules is only allowed to react when occupying the same voxel.

The next event of a system can be determined by sampling the tentative time of every possible diffusion and reaction event; every tentative event time is assumed to be exponentially distributed, and the smallest tentative event time determines which event will fire next.

An efficient algorithm to generate exact trajectories is the next subvolume method (NSM).[9]

B. Microscopic simulations

On the microscopic scale two molecules A and B, modeled as hard spheres with radii σ_A and σ_B, diffuse with diffusion constants D_A and D_B and react according to the Smoluchowski equation with a Robin boundary condition at the reaction radii $\sigma = \sigma_A + \sigma_B$. Let $\mathbf{r} = \mathbf{x}_1 - \mathbf{x}_2$ be the relative position of the two molecules (molecule A has position $\mathbf{x}_1$ and molecule B has position $\mathbf{x}_2$). Then the equation governing the relative position of the molecules is

$$\frac{\partial p}{\partial t} = D\Delta p(\mathbf{r}, t), \tag{2}$$

with the Robin boundary condition[23,28] given by

$$K\frac{\partial p}{\partial n}\Big|_{|\mathbf{r}|=\sigma} = k_a p(|\mathbf{r}| = \sigma, t), \tag{3}$$

where

$$K = \begin{cases} 4\pi\sigma^2 D, & (3D) \\ 2\pi\sigma D, & (2D) \end{cases}. \tag{4}$$

It can be shown that $\mathbf{R} = \mathbf{x}_1 + \mathbf{x}_2$ moves according to normal diffusion.[29]

A system of more than two molecules becomes an intractable many-body problem. A popular method to simulate such systems is the GFRD algorithm.[29,30] Instead of considering the full many-body problem, the system is divided into subsets of one-body and two-body problems by selecting a time step Δt during which each such subset is unlikely to interact with any other subset of molecules. Now, during Δt, molecules are either propagated by normal diffusion, or in the case of pairs of molecules, by sampling a new $\mathbf{r}$ and $\mathbf{R}$. All

microscale results in Sec. IV were obtained with the algorithm developed in Ref. 31.

Another approach to simulating Smoluchowski dynamics is to select a fixed time step during which individual molecules are propagated by normal diffusion. Given information about each molecule's initial and final positions, it is possible to compute the probabilities of molecules reacting.[14,15,32]

III. REACTION RATES

It has been shown in the case of structured Cartesian meshes that it is important to choose the reaction rates carefully to obtain accurate simulations.[8,17,18] For Cartesian meshes, the reaction rates derived in Refs. 17–19 for the standard RDME, and in Ref. 8 for a generalized RDME, were shown to yield accurate results down to a lower bound on the mesh resolution on the order of a few times the reaction radius for the standard RDME, and down to a mesh resolution on the order of the reaction radius in the case of the generalized RDME (in which molecules occupying neighboring voxels can react).

While Cartesian meshes are good for simple simulation volumes, they are less suitable for complex geometries. In that case, unstructured triangular (2D) or tetrahedral (3D) meshes are preferable. However, the analytical expressions for the reaction rates derived in Refs. 8 and 17–19 depend on the assumption of a Cartesian mesh.

In this section we devise an efficient way to numerically compute accurate reaction rates for the RDME on unstructured meshes.

We consider a system of three species A, B, and C undergoing the single irreversible reaction

$$\mathrm{A} + \mathrm{B} \xrightarrow{\mathrm{k_a}} \mathrm{C}$$

in a domain Ω, where k_a is the microscopic reaction rate. The A molecule does not diffuse, while the B molecule diffuses with diffusion rate D. We denote the reaction radius of the A and the B molecule by σ. Given these microscopic parameters, we seek to obtain the corresponding mesoscopic reaction rate k_a^{meso}, where k_a^{meso} is the rate, in units of s^{-1}, at which molecules react when they occupy the same voxel.

To this end, we make the assumption that the mean binding time on the microscale, $\tau_{\mathrm{micro}}(k_a)$, and the mesoscale, $\tau_{\mathrm{meso}}(k_a^{\mathrm{meso}})$, should match. For a given k_a, we want to find k_a^{meso} such that

$$\tau_{\mathrm{micro}}(k_a) = \tau_{\mathrm{meso}}(k_a^{\mathrm{meso}}) \tag{5}$$

holds.

A. Cartesian meshes

We first briefly summarize the derivation of reaction rates on Cartesian meshes, as some of the methodology carries over to the case of unstructured meshes. Throughout this section the domain will be a cube of volume V, discretized into N voxels, and where h denotes the width of a voxel.

1. Microscopic effective binding time

The effective binding time, τ_{micro}, can be divided into two parts: an initial diffusion part and a reaction part. We assume that the B molecule has a uniform initial distribution and that the fixed A molecule is some distance away from the boundary, and we denote the time it takes until the molecules are in contact for the first time by $\tau_{\mathrm{diff}}^{\mathrm{micro}}$. The reaction part is defined to be the time that remains until they react; that is, the time until the molecules react given that they start in contact. We denote the reaction part by $\tau_{\mathrm{react}}^{\mathrm{micro}}$. By definition we have

$$\tau_{\mathrm{micro}} = \tau_{\mathrm{diff}}^{\mathrm{micro}} + \tau_{\mathrm{react}}^{\mathrm{micro}}. \tag{6}$$

Let σ be the reaction radius, and D the diffusion constant of the B molecule. We know that for V large enough[7,18]

$$\tau_{\mathrm{diff}}^{\mathrm{micro}} \approx \begin{cases} \dfrac{V}{4\pi\sigma D}, & (3D) \\ \dfrac{V\left\{\log\left(\pi^{-1}\frac{V^{1/2}}{\sigma}\right)\right\}}{2\pi D}, & (2D) \end{cases} \tag{7}$$

and that

$$\tau_{\mathrm{react}}^{\mathrm{micro}} \approx \frac{V}{k_a} \quad (3\mathrm{D},\ 2\mathrm{D}). \tag{8}$$

Let $k_{\mathrm{CK}} = 4\pi\sigma D k_a/(4\pi\sigma D + k_a)$. In 3D we obtain the well-known expression[33,34] for the mean binding time, given by

$$\tau_{\mathrm{micro}} \approx \frac{V}{4\pi\sigma D} + \frac{V}{k_a} = \frac{4\pi\sigma D + k_a}{4\pi\sigma D k_a} V =: \frac{V}{k_{\mathrm{CK}}}, \tag{9}$$

where k_{CK} is the Collins and Kimball rate.[33] A common approach in mesoscopic simulations is to let the mesoscopic rate k_a^{meso} be the Collins and Kimball rate scaled by the volume of the voxel, V_{vox}, that is $k_a^{\mathrm{meso}} = k_{\mathrm{CK}}/V_{\mathrm{vox}}$.

2. Mesoscopic effective binding time

The mean binding time on the mesoscopic scale can be divided into two parts in a way analogous to the microscale case. The diffusion part, $\tau_{\mathrm{diff}}^{\mathrm{meso}}$, is the average time until the B molecule reaches the voxel that is occupied by the A molecule. The reaction part, $\tau_{\mathrm{react}}^{\mathrm{meso}}$, is the time until the A and the B molecule react, given that they start in the same voxel. Thus

$$\tau_{\mathrm{meso}} = \tau_{\mathrm{diff}}^{\mathrm{meso}} + \tau_{\mathrm{react}}^{\mathrm{meso}}. \tag{10}$$

These quantities are both known analytically on a Cartesian mesh[17,18]

$$\tau_{\mathrm{diff}}^{\mathrm{meso}} = \begin{cases} \dfrac{C_3 V}{6Dh} + O\left(N^{\frac{1}{2}}\right) & (3D) \\ \dfrac{V}{4\pi D}\log(N) + \dfrac{C_2 V}{4D} + O\left(N^{-1}\right) & (2D) \end{cases} \tag{11}$$

and

$$\tau_{\mathrm{react}}^{\mathrm{meso}} = \frac{N}{k_a^{\mathrm{meso}}}. \tag{12}$$

Now let

$$C_d \approx \begin{cases} 0.1951, & d = 2 \\ 1.5164, & d = 3 \end{cases}. \tag{13}$$

By inserting (7), (8), (11), and (12) into (5), and solving for k_a^{meso}, we obtain expressions for the mesoscopic reaction rate k_a^{meso},

$$k_a^{\mathrm{meso}} \approx \frac{k_a}{h^d}\left(1 + \frac{k_a}{D}G(h,\sigma)\right)^{-1}, \tag{14}$$

where

$$G(h,\sigma) = \begin{cases} \dfrac{1}{4\pi\sigma} - \dfrac{C_3}{6h} & (3D), \\ \dfrac{1}{2\pi}\log\left(\pi^{-\frac{1}{2}}\dfrac{h}{\sigma}\right) - \dfrac{1}{4}\left(\dfrac{3}{2\pi} + C_2\right) & (2D). \end{cases} \tag{15}$$

Note that (14) is an accurate approximation for $V \gg h$.

In Ref. 18 we showed that, in general, the most accurate simulations in 2D and 3D are obtained for

$$h_\infty^* = \begin{cases} \dfrac{2C_3}{3}\pi\sigma \approx 3.2\sigma, & (3D) \\ \sqrt{\pi}e^{\frac{3+2C_2\pi}{4}}\sigma \approx 5.1\sigma, & (2D) \end{cases}, \tag{16}$$

where the RDME may get less accurate as the mesh is refined further. The lower limit h_∞^* is also the smallest mesh size for which $k_a^{\text{meso}} > 0$ as $k_a \to \infty$.

Note that by (10) and (12), the mesoscopic rate can be written as

$$k_a^{\text{meso}} = \frac{N}{\tau_{\text{micro}} - \tau_{\text{diff}}^{\text{meso}}}. \tag{17}$$

If $\tau_{\text{micro}} \gg \tau_{\text{diff}}^{\text{meso}}$, then $k_a^{\text{meso}} \approx k_{\text{CK}}/h^3$ in 3D, and this is an assumption sometimes made in software implementations of the RDME. As we will see in Sec. IV, this assumption is not always satisfied and when not, this rate yields inaccurate results.

B. Unstructured meshes

In the case of Cartesian meshes we have analytical expressions for the reaction rates, where the derivation is based on analytical expressions for $\tau_{\text{diff}}^{\text{meso}}$ and $\tau_{\text{react}}^{\text{meso}}$. For unstructured meshes we can consider the same setup as for Cartesian meshes, but we do not have analytical expressions for $\tau_{\text{diff}}^{\text{meso}}$ and $\tau_{\text{react}}^{\text{meso}}$, and instead we need to compute them numerically.

The quantity $\tau_{\text{diff}}^{\text{meso}}$ can be computed independently of the microscopic reaction rates, and depends only on the diffusion rate and the mesh.

Now assume that the A molecule has diffused such that it occupies the same voxel as the B molecule. Denote by d_{tot} the total diffusion rate out of the voxel, as obtained by, e.g., a finite element discretization of the diffusion equation. The molecules then react with probability

$$p_r = \frac{k_a^{\text{meso}}}{k_a^{\text{meso}} + d_{\text{tot}}}, \tag{18}$$

and diffuse apart with probability $1 - p_r$. The molecules, on average, occupy the same voxel p_r^{-1} times before they react.

Given that the molecules occupy the same voxel, the average time until the next event will be given by

$$\frac{1}{k_a^{\text{meso}} + d_{\text{tot}}}. \tag{19}$$

They react with probability p_r. If they do not react, they consequently diffuse apart, and will then occupy adjacent voxels. Denote by t_1 the average time until the molecules again occupy the same voxel. We can summarize the above process as follows:

Assume that the molecules occupy the same voxel.

(1) With probability p_r they react after an average time of $1/(k_a^{\text{meso}} + d_{\text{tot}})$.

(2) With probability $1 - p_r$ they do not react. They occupy the same voxel once again after an average time of $1/(k_a^{\text{meso}} + d_{\text{tot}}) + t_1$.

The molecules react after occupying the same voxel on average p_r^{-1} times.

We obtain

$$\tau_{\text{react}}^{\text{meso}} = \frac{1}{p_r}\left[p_r\frac{1}{k_a^{\text{meso}} + d_{\text{tot}}} + (1-p_r)\left(\frac{1}{k_a^{\text{meso}} + d_{\text{tot}}} + t_1\right)\right], \tag{20}$$

which, after some straightforward algebra, yields

$$\tau_{\text{react}}^{\text{meso}} = \frac{1}{k_a^{\text{meso}}}(1 + d_{\text{tot}}t_1). \tag{21}$$

Now, to satisfy $\tau_{\text{meso}} = \tau_{\text{micro}}$, we should find k_a^{meso} such that

$$\tau_{\text{micro}} = \tau_{\text{diff}}^{\text{meso}} + \tau_{\text{react}}^{\text{meso}}, \tag{22}$$

which holds if and only if

$$k_a^{\text{meso}} = \frac{1 + d_{\text{tot}}t_1}{\tau_{\text{micro}} - \tau_{\text{diff}}^{\text{meso}}}. \tag{23}$$

Thus, to obtain the reaction rate k_a^{meso} for the molecules in a voxel V_i, we must compute $\tau_{\text{diff}}^{\text{meso}}$ and t_1.

C. Computing $\tau_{\text{diff}}^{\text{meso}}$ and t_1

Assume that the A molecule occupies a voxel V_i. We first compute $\tau_{\text{diff}}^{\text{meso}}$. A straightforward approach would be a simple Monte Carlo procedure:

However, for fine mesh resolutions the naive approach becomes computationally expensive; on a Cartesian mesh we know that the average number of steps required to find V_i scales proportionally to the number of voxels.

Instead we propose the following algorithm: In step 2, instead of always simulating the B molecule until it finds V_i, we note that if the B molecule has not reached V_i after some time Δt_1, where $\sqrt{2dD\Delta t_1} \sim |\Omega|^{\frac{1}{d}}$, its position can be approximated by a uniform distribution. Thus, the average time remaining until the B molecule finds V_i, after it has been diffusing for some time $\Delta t_1 \sim \frac{|\Omega|^{\frac{2}{d}}}{2dD}$, can be approximated by $\tau_{\text{diff}}^{\text{meso}}$. Let $\tau_{t\le\Delta t_1}^{\text{diff}}$ denote the average time required to find V_i, given that the B molecule reaches V_i before time Δt_1. Let q denote the probability that the B molecule finds V_i before Δt_1. Then

$$\tau_{\text{diff}}^{\text{meso}} \approx q\tau_{t\le\Delta t_1}^{\text{diff}} + (1-q)\left(\Delta t_1 + \tau_{\text{diff}}^{\text{meso}}\right). \tag{24}$$

By solving for $\tau_{\text{diff}}^{\text{meso}}$ we obtain

$$\tau_{\text{diff}}^{\text{meso}} \approx \tau_{t\le\Delta t_1}^{\text{diff}} + \frac{1-q}{q}\Delta t_1. \tag{25}$$

Thus, by letting $\Delta t_1 = c_1\frac{|\Omega|^{\frac{2}{d}}}{2dD}$ for some suitable constant c_1, we obtain an estimate of $\tau_{\text{diff}}^{\text{meso}}$ by computing $\tau_{t\le\Delta t_1}^{\text{diff}}$ and q. We discuss how to choose c_1 in Sec. IV.

We now wish to estimate t_1. Again we could approach this with a naive Monte Carlo approach:

However, again we note that if the B molecule does not reach V_i after $\Delta t_2 = c_2\frac{|\Omega|^{\frac{2}{d}}}{2dD}$, for some suitably chosen constant c_2, it will be approximately uniformly distributed on Ω. Thus, after a time Δt_2 the time remaining can be approximated by

$\tau_{\text{diff}}^{\text{meso}}$. Now, in step 2 above, we see that it is enough to simulate a trajectory until time Δt_2; if the B molecule has not visited V_i, we simply add $\tau_{\text{diff}}^{\text{meso}}$ to the total time.

We summarize the algorithm for computing $\tau_{\text{diff}}^{\text{meso}}$ and t_1 in Algorithm 3.

Note that for a given mesh we only need to compute $\tau_{\text{diff}}^{\text{meso}}$ and t_1 once, even if different species have different diffusion rates. This is because $\tau_{\text{diff}}^{\text{meso}}$ and t_1 are inversely proportional to the diffusion constant.

D. Linear approximation of $1/k_a^{\text{meso}}$

In principle we should compute k_a^{meso} for each voxel of the mesh, but for a fine mesh this will be prohibitively expensive computationally, as the number of voxels can be on the order of 10^5 or more. We know that $1/k_a^{\text{meso}}$ has a nonlinear dependence on the volume of the voxels, but by assuming that the distribution of voxel volumes is not too wide, we can, assuming that $1/k_a^{\text{meso}}$ is continuous, make a linear approximation locally. That is, by assuming that the distribution of volumes in a mesh is not too wide, we can estimate the mesoscopic reaction rate by

$$(k_a^{\text{meso}})^{-1} \approx k_0 + k_1 V_{\text{vox}}, \tag{26}$$

where V_{vox} is the volume of a voxel. For (26) to be useful, we also have to assume that for a given mesh, k_a^{meso} is approximately the same for two different voxels of the same volume. It is reasonable to assume that this will be true if the distribution of voxel volumes is similar throughout the domain. That may not be true for very complex domains, in which case we could compute the coefficient of determination to evaluate how good of an approximation the linear regression provides. In Sec. IV we show that assumption (26) is a good approximation for a few different common geometries.

For very small voxels in a mesh we may have

$$V_{\text{vox}} < -\frac{k_0}{k_1}, \tag{27}$$

leading to a negative estimate of k_a^{meso}. If this happens for many voxels in a mesh, then the mesh is over resolved, analogously to how a Cartesian mesh can be over resolved for $h < h_\infty^*$. It may however happen for a few voxels in a mesh, since the voxel volume is non-uniform, without the mesh overall being over resolved. In this case, we somewhat arbitrarily compute the rate for that voxel with $V_{\text{vox}} = -\frac{k_0}{k_1} + \epsilon$ for some small positive ϵ, to force the rate to be positive. This will introduce a small error, but as we will see in the second example in Sec. IV, it can be neglected even for very fine meshes.

IV. NUMERICAL RESULTS

A. Accuracy of the method

We need to determine suitable values for the constants c_1 and c_2. To that end, we consider one A molecule fixed to a single voxel near the center of a sphere of radius 1. One B molecule reacts with the A molecule and diffuses with a diffusion constant $D = 1$. Note that c_1 and c_2 are independent of the specific parameters of the reaction, and only depend on the diffusion of the B molecule.

Algorithm 1

1. Initialize the B molecule according to a uniform distribution on the mesh.
2. Simulate the system until the B molecule finds V_i.
3. Repeat N_1 times and compute the mean.

Algorithm 2

1. Sample the initial position of the B molecule by letting it diffuse from V_i to an adjacent voxel.
2. Diffuse the B molecule until it finds V_i and record the time t.
3. Repeat (1) and (2) N_2 times and compute the mean.

We first compute highly accurate approximations, $\bar{t_1}$ and $\overline{\tau_{\text{diff}}^{\text{meso}}}$, of t_1 and $\tau_{\text{diff}}^{\text{meso}}$ using Algorithms 1 and 2. We then proceed to compute approximations of t_1 and $\tau_{\text{diff}}^{\text{meso}}$, $\tilde{t_1}$ and $\widetilde{\tau_{\text{diff}}^{\text{meso}}}$, using Algorithm 3 while varying c_1 and c_2. The relative errors $E_1 = |\tilde{t_1} - \bar{t_1}|/|\bar{t_1}|$ and $E_2 = |\widetilde{\tau_{\text{diff}}^{\text{meso}}} - \overline{\tau_{\text{diff}}^{\text{meso}}}|/|\overline{\tau_{\text{diff}}^{\text{meso}}}|$ are plotted as heat maps in Fig. 1. In Fig. 2 we show that the approximate method (Algorithm 3) gives a speed-up of up to an order of magnitude compared to the exact approach of Algorithms 1 and 2.

We repeated the computations for a sequence of meshes of different resolutions (utilizing the tool Gnu Parallel[35]), and as we can see, $c_1 = 5$ and $c_2 = 5$ give errors on the order of 1%–2%, thus being reasonable choices.

It is reasonable to assume that Eqs. (13)–(15), with h substituted for $V_{\text{vox}}^{1/3}$, where V_{vox} is the volume of a voxel, will agree

Algorithm 3

Assume that the A molecule occupies voxel V_i.

1. Initialize the B molecule uniformly on Ω.
2. Let the B molecule diffuse until:
   (a) It finds V_i, or
   (b) $t = c_1 \frac{|\Omega|^{\frac{2}{d}}}{2dD}$.
3. Repeat (1) and (2) N_1 times.
4. Let q denote the proportion of trajectories that ended in 2 (a), and let $\tau_{t\le\Delta t_1}^{\text{diff}}$ denote the average time of the trajectories that ended in 2 (a). Estimate $\tau_{\text{diff}}^{\text{meso}}$ by
$$\widetilde{\tau_{\text{diff}}^{\text{meso}}} = \tau_{t\le\Delta t_1}^{\text{diff}} + \frac{1-q}{q}\Delta t_1.$$
5. Initialize the B molecule in a voxel adjacent to V_i, proportionally to the diffusion rates out of V_i.
6. Let the B molecule diffuse until:
   (a) It reaches V_i after some time $t_{\text{reac}} \le \Delta t_2$. Let $\tilde{t_1} = t_{\text{reac}}$ be an approximation of t_1.
   (b) $t = c_2 \frac{|\Omega|^{\frac{2}{d}}}{2dD}$. Let $\tilde{t_1} = \Delta t_2 + \widetilde{\tau_{\text{diff}}^{\text{meso}}}(k_a^{\text{meso}})$ be an approximation of t_1.
7. Repeat (5) and (6) N_2 times.
8. Estimate t_1 with the mean of $\tilde{t_1}$.
9. Estimate k_a^{meso} by
$$\tilde{k}_a^{\text{meso}} = \frac{1 + d_{\text{tot}}\tilde{t_1}}{\tau_{\text{micro}} - \widetilde{\tau_{\text{diff}}^{\text{meso}}}}.$$

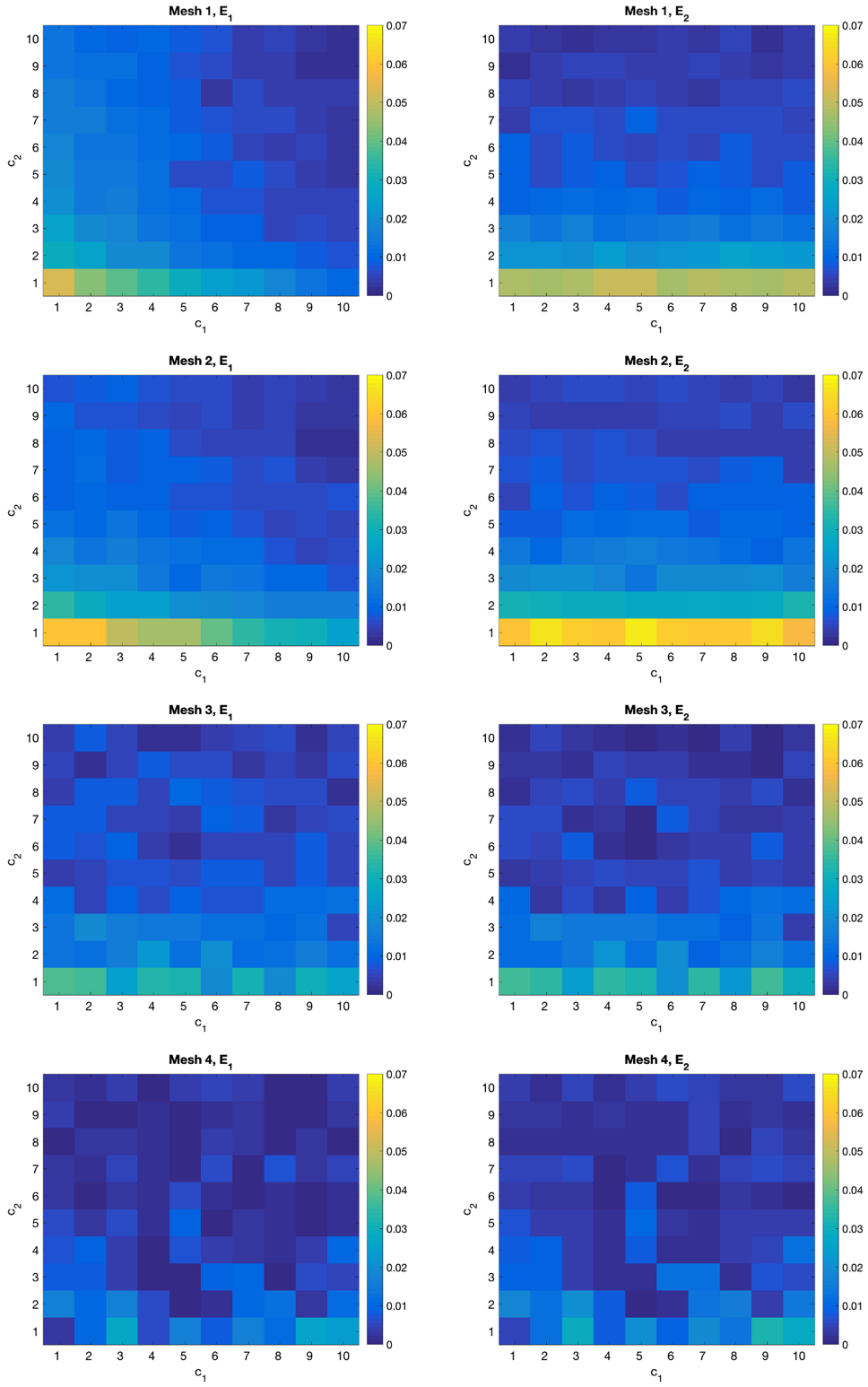

FIG. 1. The errors E_1 and E_2 as functions of c_1 and c_2, for four different meshes of increasing resolution. From top to bottom: mesh 1: 358 voxels, mesh 2: 1185 voxels, mesh 3: 22 644 voxels, and mesh 4: 98 206 voxels. We compute $\tilde{t}_1$, $\bar{t}_1$, $\overline{\tau_{\text{diff}}^{\text{meso}}}$, and $\overline{\tau_{\text{diff}}^{\text{meso}}}$ as the mean of 10^6 trajectories. The stochastic error is fairly small, and as we can see, E_1 and E_2 are generally on the order of 1%-2% for $c_1 = c_2 = 5$, which we therefore argue is a reasonable choice. For some mesh sizes it seems that smaller values of c_1 and c_2 would provide a similar accuracy, but the above choice should serve as a good general recommendation.

quite well for spatial reactions in simpler geometries. To test this hypothesis, as well as to show that the linearity assumption in Sec. III D is reasonable, we let the microscopic reaction rate be 1 and the reaction radius $5 \cdot 10^{-3}$ and then computed the rates numerically according to Algorithm 3. We then compared the results with the rates computed according to (13)–(15). In Fig. 3 we see that the rates do indeed agree quite well, while the rates computed as $k_{\text{CK}}/V_{\text{vox}}$ become increasingly incorrect as the mesh is refined.

Note that we cannot expect the numerical approach to always agree with Eqs. (13)–(15). The reason is that the formula (13)–(15) does not take into account that voxels in an unstructured mesh may be of different sizes; the rate for a small voxel within a mesh of mostly larger voxels may be incorrectly approximated by (13)–(15).

We should also note that the microscopic mean binding time for voxels close to a reflective boundary will not be given by τ_{micro} as computed by Eqs. (6)–(8) (the mean binding time

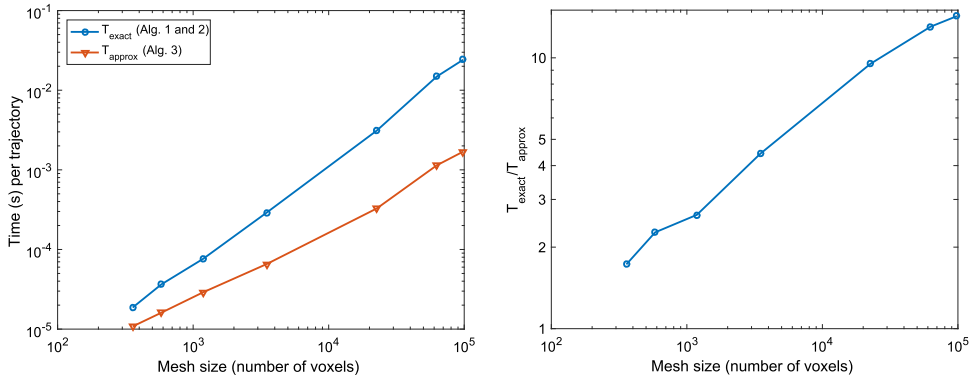

FIG. 2. To the left we plot the time in seconds per trajectory for the exact method, T_{exact}, outlined in Algorithms 1 and 2 (blue line with circles) and the time per trajectory of the approximate method, T_{approx}, outlined in Algorithm 3 (red line with triangles) with $c_1 = c_2 = 5$. To the right we plot the relative speed-up with $c_1 = c_2 = 5$. As we can see, for coarse meshes the speed-up is fairly modest (about two times), but as we refine the mesh the speed-up becomes significant. For a fine mesh consisting of 10^5 voxels, we obtain a speed-up of almost 15. Nevertheless, it is noteworthy that even with the exact method we can expect reasonable execution times, often on the order of the total simulation time, if we do not need excessive amounts of samples. Thus, if very high accuracy is required, the exact approach is computationally viable.

for a voxel far from the boundary). Thus, if very high accuracy is needed, we should for those voxels compute also τ_{micro} numerically. However, since we take a sample of voxels and perform linear regression, the effect of that error will generally be small, and as we show in the next example, we obtain very accurate simulations also when neglecting this error.

B. Dissociation with fast rebinding

To demonstrate the applicability of the method, we consider a system where the microscale model displays dynamics different from the mesoscopic model. A simple example is given by

$$\mathrm{S}_1 \xrightarrow{\mathrm{k}_1} \mathrm{S}_{11} + \mathrm{S}_{12} \xrightarrow{\mathrm{k}_2} \mathrm{S}_2. \tag{28}$$

On the microscopic scale, following a dissociation of S_1, the products S_{11} and S_{12} are placed in contact. Thus, the probability of S_{11} and S_{12} to rebind quickly and form the complex S_2 is higher than on the mesoscopic scale, where S_{11} and S_{12} are assumed to be well-mixed inside a voxel immediately following a dissociation. Note that all parameter values below are given in SI units.

With the RDME we expect to approach the microscale results as we refine the mesh, and ideally, for the finest mesh sizes, we hope to reproduce the results of the microscale simulations to high accuracy.

We simulate the system defined by Eq. (28) for 2 s, sampling the state at $M = 200$ uniformly distributed time points with $t_1 = 0.01$ and $t_{200} = 2.0$, and compute the average relative l_1 error of the final product S_2 as

$$E = \frac{1}{M}\sum_{i=1}^{M}\frac{|z_i - y_i|}{|y_i|}, \tag{29}$$

where z_i is the average of 5000 mesoscopic trajectories at time t_i, and y_i is the average of 10 000 trajectories on the microscopic scale at time t_i. The initial number of S_1 molecules

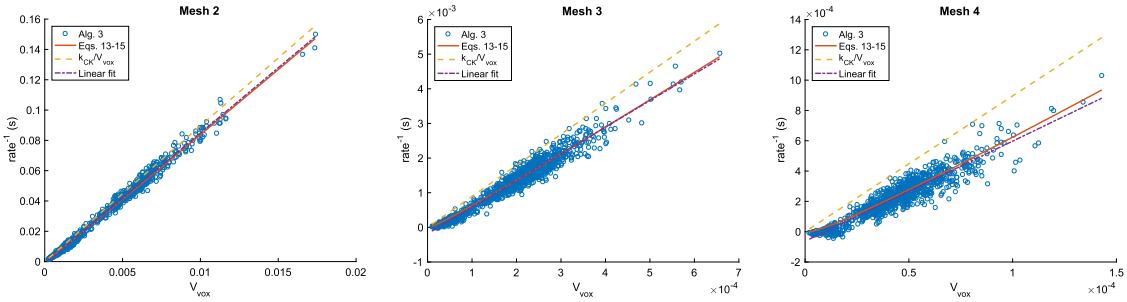

FIG. 3. The inverse of the rates sampled according to Algorithm 3 (blue circles), the linear fit of these samples (dashed-dotted purple line), rates computed according to Eqs. (13)–(15) with $h = V_{\text{vox}}^{1/3}$ (solid red line), and the effective rate scaled by the volume of the voxels (dashed yellow line). We computed 10^3 samples, where for each sample we place the A molecule in a random voxel, with 10^3 trajectories for each sample. Each voxel has a different volume V_{vox}. The total execution time on a desktop computer with an Intel i7-4770 CPU at 3.50 GHz running Ubuntu 14.04 was 8 s for the coarse mesh (mesh 2, $\sim 10^3$ voxels)), 83 s for the intermediate mesh (mesh 3, $\sim 23 \cdot 10^3$ voxels), and 339 s for the fine mesh (mesh 4, $\sim 10^5$ voxels). Again, $c_1 = c_2 = 5$. As we can see, the numerically computed rates agree reasonably well with the rates obtained with Eqs. (13)–(15) for all three mesh sizes, but the difference does increase as the mesh is refined. For the coarser mesh we see that $k_{\text{CK}}/V_{\text{vox}}$ provides a decent approximation, but as we refine the mesh it becomes a poor approximation of the reaction rates, in agreement with the theory outlined in Sec. II. Note that while the spread around the linear fit is quite large for the finer meshes, we do not see a large error in actual simulations (see Fig. 5). As we estimate the rate for each voxel with only 10^3 trajectories, we can expect a fairly large variance for fine meshes, but the error in the linear regression is not too large to still yield accurate enough rates for simulations of the system in Sec. IV B. We can also see that in particular for Mesh 4, we get some estimates <0. This also introduces a small error. Since all molecules diffuse around in this example, these errors are all small enough not to make a large impact on the overall error. If, however, we were to place a stationary molecule inside such a voxel, then we expect the error to be more significant.

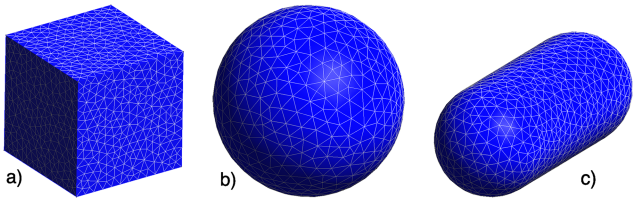

FIG. 4. We consider three different geometries: (a) a cube, (b) a sphere, and (c) two half-spheres connected by a cylinder. Each has a total volume of 1, and in geometry (c) the ratio of the radius to the length of the cylinder is 3. The geometries are discretized into tetrahedral meshes using Gmsh.[36]

is 100, and the initial number of S_{11}, S_{12}, and S_2 molecules is 0.

The microscale parameters are given by

$$\begin{cases} \sigma_1 = \sigma_{11} = \sigma_{12} = \sigma_2 = 2.5 \cdot 10^{-3} \\ D_1 = D_{11} = D_{12} = D_2 = 1.0 \\ k_1 = 10.0 \\ k_2 = 1.0 \end{cases}, \tag{30}$$

and the volume of each domain is $V = 1$ (see Fig. 4).

To demonstrate the flexibility of the algorithm, we consider three different geometries: (a) a cube, (b) a sphere, and (c) two half-spheres connected by a cylinder. We start out with a coarse mesh, and then consider successively finer meshes. In Fig. 5 we show that for the finest meshes the relative error is on the order of a few percent, in contrast to the constant high error when the rate is given by k_{CK}/V_{vox}. We can also see that the convergence is similar for all three geometries. When computing the rates, we neglected the error introduced by approximating the mean binding time close to boundaries by τ_{micro}, and as we can see, it did not introduce a large error in the simulated dynamics of the system.

It is noteworthy that, for this particular system, the microscale simulations will be fairly efficient in comparison to the mesoscopic simulations for the finest mesh sizes. For instance, for the finest cubic mesh (~240 000 voxels), a single

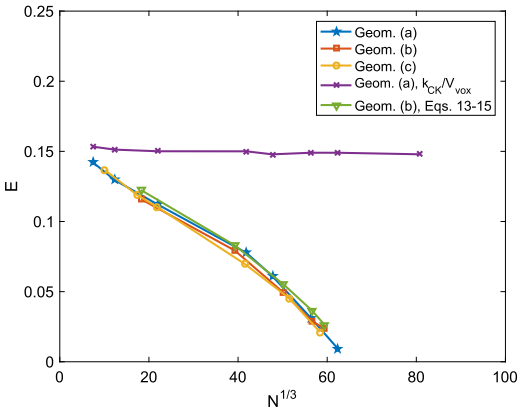

FIG. 5. The relative error E as defined by (29), as a function of the total number of voxels N. With the reaction rate k_2^{meso} computed according to Algorithm 3, the error decreases as we increase the resolution of the mesh (geometry (a): blue line with stars, geometry (b): red line with squares, and geometry (c) yellow line with circles), while we see no convergence if we choose the reaction rate to be the effective rate scaled by the volume of the voxel (purple line with crosses). We also see that for this problem we obtain accurate simulations also with the rates computed according to Eqs. (13)–(15) with $h = V_{vox}^{1/3}$ (green line with triangles).

trajectory on the mesoscopic scale (excluding the preprocessing time) is more expensive to simulate than the corresponding microscale simulation of a trajectory (7.35 s vs 5.83 s on a desktop computer with an Intel i7-4770 CPU at 3.50 GHz running Ubuntu 14.04). However, this system has a fairly low number of molecules, making it well suited for microscale simulations. Also, the mesoscale simulations scale quadratically with the number of voxels. Thus, for systems with more molecules or for less resolved geometries, the RDME will be competitive.

C. Reaction close to a boundary

We consider the system given by

$$S_1 + S_2 \underset{k_2}{\overset{k_1}{\rightleftharpoons}} S_3, \tag{31}$$

$$S_3 \xrightarrow{k_3} S_3 + P, \tag{32}$$

$$P \xrightarrow{k_4} \emptyset, \tag{33}$$

where we are interested in computing the population of species P to high accuracy. The domain in 3D is a large sphere with radius 0.8, with an inner boundary defined by a smaller sphere of radius 0.4 (see Fig. 6). This represents a simple model of a cell with a nucleus, where the nucleus occupies a volume fraction of 12.5% of the cell. We let the total volume be denoted by V.

Let k_1 (volume/s), k_2 (s^{-1}), k_3 (s^{-1}), and k_4 (s^{-1}) be the macroscopic reaction rates. Furthermore, assume that the diffusion of the S_2 molecule is negligible during the time interval of interest, so that it is fixed inside a given voxel. All we know is the rate of reaction,[33] without any detailed spatial information about the location of the S_2 molecule. We let $k_1/V = k_2 = 0.0308\ s^{-1}$. The other reaction rates are given by $k_3 = 20\ s^{-1}$ and $k_4 = 5.0\ s^{-1}$. The diffusion rate of S_1 is $D = 1.0$. We initialize one S_1 molecule according to a uniform distribution on the mesh, and fix one S_2 molecule to a specific voxel in the domain.

If the S_2 molecule is located away from the boundary, somewhere in the interior of the domain, then the analytical approach of Refs. 17 and 18 will provide accurate estimates of the reaction rates. By formulating the mesoscopic rates given in Eqs. (13)–(15) in terms of the macroscopic rates, by simply noting that the macroscopic rate will be V/τ_{micro}, we can accurately estimate the mesoscopic rates for this case. In Ref. 17 we show that the mesoscopic reaction rate can be written as $N/(\tau_{micro} - \tau_{diff}^{meso})$, where N is the number of voxels. If, however, the S_2 molecule happens to be located near a boundary, the assumptions made in deriving Eqs. (13)–(15) are not satisfied, and we cannot expect them to give accurate results. Instead we will have to resort to the numerical approach of Algorithm 3.

When the S_2 molecule is fixed, it is sufficient to compute the reaction rate for the specific voxel that it occupies. We can therefore afford to compute the reaction rate for that voxel to high accuracy; we compute it using 25 000 trajectories in our estimates of t_1 and τ_{diff}^{meso}.

Reaction (31) is reversible, and thus we must determine also the dissociation rate. In the case of a Cartesian mesh we could derive the dissociation rate using analytical expressions

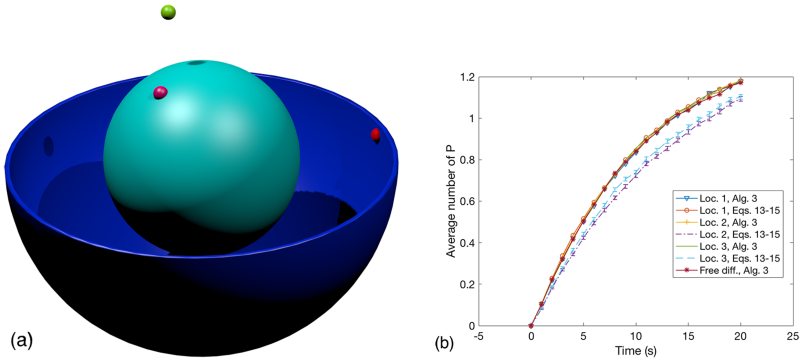

FIG. 6. We fix the S_2 molecule at three different locations (panel (a), to the left): away from both boundaries (green sphere, location 1) in the interior of the domain, close to the outer boundary (red sphere, location 2), and close to the inner boundary (purple half sphere, location 3). The spheres do not mark the exact location of the S_2 molecule; they show the location of the voxel holding the S_2 molecule. The nodes of the voxels coincide with the centers of the spheres. To the right, in panel (b), we plot the average number of P molecules over time. While all versions of the system will eventually settle into the same steady state, the transient phases do not agree. The reason is that the average binding time of the S_1 and S_2 molecules depends on the specific spatial location of the S_2 molecule when we use the analytical approach. When the stationary molecule is placed inside the domain, far away from the boundaries, both approaches yield accurate results. When the stationary molecule is placed near a boundary, only the numerical approach reproduces the correct behavior.

for all quantities and matching the steady state on the mesoscopic scale to the steady state on the microscopic scale.[18] In the case of unstructured meshes, we cannot do that since we do not have analytical expressions for the relevant quantities. Instead, we determine the dissociation rate by simply matching the equilibrium constant on the mesoscopic and macroscopic levels, yielding $k_d^{\text{meso}} = Vk_a^{\text{meso}}k_2/k_1$.

As the macroscale reaction rates are given independently of the location of the reactive molecule (that information is hidden in the rate), we want to see the same behavior regardless of where we happen to place the reactive S_2 molecule. In Fig. 6 we show that the analytical formulas fail to reproduce the correct behavior when the reactive molecule is close to a boundary. This is expected, since the formulas are derived under the assumption that the reactive molecule is located sufficiently far away from the boundaries. Thus, given knowledge of the spatial location of reactive molecules, the numerical approach may be preferable for matching a given macroscopic reaction rate.

For reference, we also simulated the system with both molecules diffusing freely with a diffusion rate of $D/2$. The reaction rates are estimated with Algorithm 3, and the results, as expected, agree with the case of the S_2 molecule fixed to a voxel inside the domain but a distance away from the boundaries. This demonstrates that the approach of Algorithm 3 with a linear approximation in the volume yields accurate reaction rates also when the surface-to-volume ratio is relatively high.

In Fig. 7 we consider the same reactions, but now on a 2D unit disk. The macroscopic reaction rates for the reversible reaction are $k_1/A = k_2 = 0.1818\,\text{s}^{-1}$ (where $A = 1$ is the area of the disk), and all the other parameters are the same as before. Again we see the same behavior. The population of P is consistent regardless of where we place the stationary molecule when we compute the mesoscopic reaction rates with Algorithm 3, while the analytical approach fails to reproduce the expected behavior when the stationary molecule is placed close to the boundary.

V. DISCUSSION

It has been shown that it is crucial for the accuracy of the RDME that reaction rates are selected with care when reactions are diffusion limited or when the spatial resolution is high. For structured Cartesian meshes, this problem has been studied in some detail, but not for simulations on unstructured meshes.

We have devised a method to compute accurate rates on unstructured meshes, and shown in numerical examples that with these rates the RDME is accurate also on unstructured meshes, for a wide range of mesh sizes and reaction rates. Furthermore, the method reduces to a pure preprocessing step with a computational cost that tends to be on the order of the total simulation time or less.

We have also shown for a few different geometries in 3D that the numerically computed rates agree well with the

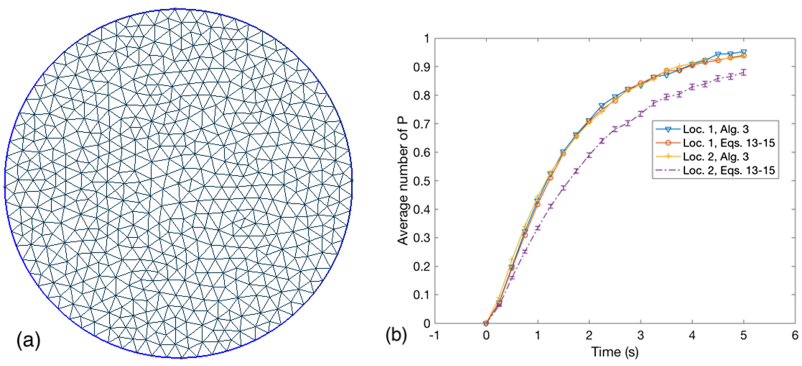

FIG. 7. We simulate the system on a triangulated unit disk (panel (a) to the left). The system displays the same behavior in 2D as it does in 3D: when the stationary molecule is placed near the center of the disk (Loc. 1), we get the same result with both approaches. When the stationary molecule is placed inside a voxel near the boundary (Loc. 2), we get the correct number of P only when we compute the reaction rates with Algorithm 3.

corresponding rates on Cartesian meshes, suggesting that for many systems the analytically derived rates on Cartesian meshes will provide sufficient accuracy. Conversely, in some cases where the system does not satisfy the assumptions made in deriving the analytical expressions, the reaction rates need to be computed numerically. We have shown that this can be made efficiently and accurately for such a system in both 3D and 2D geometry.

It is worth noting that the numerical approach outlined in this paper can be generalized to other types of reactions; we may for instance consider reactions between molecules in 3D and complex surfaces or 1D objects. It is straightforward to extend the algorithm to such reactions where analytical approaches are less likely to be successful.

Another interesting problem is the case when the computational domain is non-static. In Ref. 37 we consider simulations on an unstructured mesh that changes over time; to compute the reaction rates numerically in that case would, strictly speaking, require that we recompute the rates each time the domain changes. This would become prohibitively expensive. Instead we will have to assume that the analytical expressions are sufficiently accurate, but in general we cannot, and thus the case of moving domains is a problem that requires further study.

ACKNOWLEDGMENTS

This work was supported by NIBIB of the NIH under Grant No. R01-EB014877-01, the Institute for Collaborative Biotechnologies through Grant No. W911NF-09-D-0001 from the U.S. Army Research Office, and U.S. DOE Grant No. DE-SC0008975. The content of the information does not necessarily reflect the position or the policy of the Government, and no official endorsement should be inferred.

[1] M. J. Lawson, B. Drawert, M. Khammash, and L. Petzold, "Spatial stochastic dynamics enable robust cell polarization," PLoS Comput. Biol. **9**(7), e1003139 (2013).
[2] A. D. Rutenberg and M. Howard, "Pattern formation inside bacteria: Fluctuations due to the low copy number of proteins," Phys. Rev. Lett. **90**(12), 128102 (2003).
[3] J. Elf and D. Fange, "Noise induced min phenotypes in *E. coli*," PLoS Comput. Biol. **2**(6), e80 (2006).
[4] M. Sturrock, A. Hellander, A. Marsavinos, and M. Chaplain, "Spatial stochastic modeling of the Hes1 pathway: Intrinsic noise can explain heterogeneity in embryonic stem cell differentiation," J. R. Soc., Interface **10**(80), 20120988 (2013).
[5] M. Sturrock, A. Hellander, S. Aldakheel, L. Petzold, and M. Chaplain, "The role of dimerisation and nuclear transport in the Hes1 gene regulatory network," Bull. Math. Biol. **76**(4), 766–798 (2013).
[6] 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**(6), 2473–2478 (2010).
[7] 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**(46), 19820–19825 (2010).
[8] S. Hellander and L. R. Petzold, "Reaction rates for a generalized reaction-diffusion master equation," Phys. Rev. E **93**(1), 013307 (2016).
[9] J. Elf and M. Ehrenberg, "Spontaneous separation of bi-stable biochemical systems into spatial domains of opposite phases," Syst. Biol. **1**(2), 230–236 (2004).
[10] J. Hattne, D. Fange, and J. Elf, "Stochastic reaction-diffusion simulation with MesoRD," Bioinformatics **21**(12), 2923–2924 (2005).
[11] R. F. Oliveira, A. Terrin, G. Di Benedetto, R. C. Cannon, W. Koh, M. Kim, M. Zaccolo, and K. T. Blackwell, "The role of type 4 phosphodiesterases in generating microdomains of camp: Large scale stochastic simulations," PLoS One **5**(7), e11725 (2010).
[12] M. Tomita, K. Hashimoto, K. Takahashi, T. Shimizu, Y. Matsuzaki, F. Miyoshi, K. Saito, S. Tanida, K. Yugi, J. Craig, V. Clyde, and A. Hutchison, "E-Cell: Software environment for whole cell simulation," Bioinformatics**15**(1), 72 (1999).
[13] I. Hepburn, W. Chen, S. Wils, and E. De Schutter, "STEPS: Efficient simulation of stochastic reaction-diffusion models in realistic morphologies," BMC Syst. Biol. **6**(1), 36 (2012).
[14] S. S. Andrews and D. Bray, "Stochastic simulation of chemical reactions with spatial resolution and single molecule detail," Phys. Biol. **1**(3), 137–151 (2004).
[15] R. A. Kerr, T. M. Bartol, B. Kaminsky, M. Dittrich, J.-C. Jack Chang, S. B. Baden, T. J. Sejnowski, and J. R. Stiles, "Fast Monte Carlo simulation methods for biological reaction-diffusion systems in solution and on surfaces," SIAM J. Sci. Comput. **30**(6), 3126–3149 (2008).
[16] S. Engblom, L. Ferm, A. Hellander, and P. Lötstedt, "Simulation of stochastic reaction-diffusion processes on unstructured meshes," SIAM J. Sci. Comput. **31**(3), 1774–1797 (2009).
[17] S. Hellander, A. Hellander, and L. R. Petzold, "Reaction-diffusion master equation in the microscopic limit," Phys. Rev. E **85**(4), 042901 (2012).
[18] S. Hellander, A. Hellander, and L. R. Petzold, "Reaction rates for mesoscopic reaction-diffusion kinetics," Phys. Rev. E **91**(2), 023312 (2015).
[19] R. Erban and J. Chapman, "Stochastic modelling of reaction-diffusion processes: Algorithms for bimolecular reactions," Phys. Biol. **6**(4), 046001 (2009).
[20] S. A. Isaacson, "A convergent reaction-diffusion master equation," J. Chem. Phys. **139**(5), 054101 (2013).
[21] M. Doi, "Second quantization representation for classical many-particle system," J. Phys. A: Math. Gen. **9**(9), 1465–1477 (1976).
[22] M. V. Smoluchowski, "Versuch einer mathematischen theorie der koagulationskinetik kolloider lösungen," Z. Phys. Chem. **92**(2), 129–168 (1917).
[23] N. Agmon and A. Szabo, "Theory of reversible diffusion-influenced reactions," J. Chem. Phys. **92**(9), 5270 (1990).
[24] A. Hellander, S. Hellander, and P. Lötstedt, "Coupled mesoscopic and microscopic simulation of stochastic reaction-diffusion processes in mixed dimensions," Multiscale Model. Simul. **10**(2), 585–611 (2012).
[25] M. B. Flegg, S. Hellander, and R. Erban, "Convergence of methods for coupling of microscopic and mesoscopic reaction-diffusion simulations," J. Comput. Phys. **289**, 1–17 (2015).
[26] M. Flegg, J. Chapman, and R. Erban, "The two-regime method for optimizing stochastic reaction-diffusion simulations," J. R. Soc., Interface **9**(70), 859–868 (2012).
[27] M. Robinson, S. Andrews, and R. Erban, "Multiscale reaction-diffusion simulations with smoldyn," Bioinformatics **31**(14), 2406–2408 (2015).
[28] J. C. Jaeger and H. S. Carslaw, *Conduction of Heat in Solids* (Oxford University Press, 1959).
[29] J. S. van Zon and P. R. ten Wolde, "Simulating biochemical networks at the particle level and in time and space: Green's-function reaction dynamics," Phys. Rev. Lett. **94**(12), 128103 (2005).
[30] 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**(23), 234910 (2005).
[31] S. Hellander and P. Lötstedt, "Flexible single molecule simulation of reaction-diffusion processes," J. Comput. Phys. **230**(10), 3948–3965 (2011).
[32] M. J. Morelli and P. R. ten Wolde, "Reaction Brownian dynamics and the effect of spatial fluctuations on the gain of a push-pull network," J. Chem. Phys. **129**(5), 054112 (2008).
[33] F. C. Collins and G. E. Kimball, "Diffusion-controlled reaction rates," J. Colloid Sci. **4**(4), 425–437 (1949).
[34] D. T. Gillespie, "A diffusional bimolecular propensity function," J. Chem. Phys. **131**(16), 164109 (2009).
[35] O. Tange, "Gnu parallel: The command-line power tool," USENIX Mag. **36**(1), 42–47 (2011).
[36] C. Geuzaine and J.-F. Remacle, "Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities," Int. J. Numer. Methods Eng. **79**(11), 1309–1331 (2009).
[37] B. Drawert, S. Hellander, M. Trogdon, and T.-M. Yi, "A framework for discrete stochastic simulation on 3D moving boundary domains," J. Chem. Phys. **145**(18), 184113(2016).