

Accurate particle-based simulation of adsorption, desorption and partial transmission

Steven S Andrews

Molecular Sciences Institute, 2168 Shattuck Avenue, Berkeley, CA 94704, USA

and

Fred Hutchinson Cancer Research Center, 1100 Fairview Avenue N., Seattle, WA 98109, USA

E-mail: sandrews@fhcrc.org

Received 30 June 2009


Accepted for publication 20 October 2009

Published 12 November 2009

Online at stacks.iop.org/PhysBio/6/046015

Abstract

Particle-based simulators represent molecules of interest with point-like particles that diffuse and react in continuous space. These simulators are often used to investigate spatial or stochastic aspects of biochemical systems. This paper presents new particle-based simulation algorithms for modeling interactions between molecules and surfaces; they address irreversible and reversible molecular adsorption to, desorption from and transmission through membranes. Their central elements are: (i) relationships between adsorption, desorption and transmission coefficients on the one hand, and simulator interaction probabilities on the other, and (ii) probability densities for initial placements of desorbed molecules. These algorithms, which were implemented and tested in the Smoldyn simulator, are accurate, easy to implement and computationally efficient. They allow longer time steps and better address reversible processes than an algorithm that Erban and Chapman recently presented (*Physical Biology* 4:16–28, 2007). This paper also presents a method for simulating unbounded diffusion in a limited spatial domain using a partially absorbing boundary, as well as new solutions to the diffusion differential equation with reversible Robin boundary conditions.

 This article has associated online supplementary data files

Nomenclature

Units are given in parentheses, assuming a three-dimensional system: L = length, T = time and 1 = unitless.

Roman symbols

$C(x, t)$	concentration in solution; time is omitted for steady state (L^{-3})
$\hat{C}(x, z)$	Laplace transform of solution concentration ($L^{-3} T$)
$C_0, C_{\pm\infty}$	concentration adjacent to and infinitely far from the surface (L^{-3})
$C_a(t)$	concentration adsorbed to the surface; time is omitted for steady state (L^{-2})

D, D'	normal and reduced diffusion coefficient ($L^2 T^{-1}, 1$)
$J, \mathbf{J}(\mathbf{r})$	flux parallel to the x -axis and flux vector ($L^{-2} T^{-1}$)
k, k'	normal and reduced desorption rate constant ($T^{-1}, 1$)
$\tilde{\mathbf{n}}(\mathbf{r})$	unit outward normal for the bounding surface (1)
P_a, P_d	adsorption and desorption probability (1)
P_{tF}, P_{tB}	transmission probability for front or back (1)
$p(x)$	spatial probability density (L^{-1})
q_j	emission rate of the source j (T^{-1})
\mathbf{r}, \mathbf{r}_j	position on the bounding surface and of the source j (L)
s	root mean square step length (L)
$t, \Delta t$	time and simulation time step (T)
x	position on the x -axis (L)
z	Laplace transform conjugate of time (T^{-1})

Greek symbols

κ	adsorption coefficient ($L T^{-1}$)
κ_F	permeability coefficient from the front side ($L T^{-1}$)
κ_B	permeability coefficient from the backside ($L T^{-1}$)
$\kappa', \kappa_F', \kappa_B'$	reduced adsorption or permeability coefficient (1)

1. Introduction

Cell modeling studies typically represent molecular interactions with mass action methods. These methods are simple, well supported by software (e.g. Copasi [21] and Virtual Cell [32]), and fully adequate for a wide variety of studies. However, they cannot represent the natural stochasticity that arises within and between cells [38], nor the intricate intracellular spatial organization that most cell systems exhibit. To investigate these topics, many scientists use more detailed modeling methods (reviewed in [4]).

Particle-based simulation is one of these more detailed methods. It represents individual molecules with point-like particles that behave in accordance with elementary processes, including diffusion and chemical reactions. For typical cell biology systems, particle-based simulation can support spatial resolution to scales of several nanometers [1, 14], can simulate up to tens of minutes of real time [23] and can represent hundreds of thousands of simulated molecules [2]. Freely available particle-based simulators include Cell++ [39], ChemCell [34], MCell [22] and Smoldyn [2]. Using these tools, researchers have shown, for example, that neural signaling may depend upon neurotransmitter release from sites that are away from the synapse [14] and that stable protein concentration gradients may arise in bacterial cells [31].

Recently, Erban and Chapman developed particle-based algorithms for simulating molecular adsorption to surfaces [18] (figure 1(a)). They addressed the following problem: given a typical particle-based simulator design [3, 22, 35] (figure 2), in which molecules diffuse in discrete steps and either adsorb to (with *adsorption probability* P_a) or reflect from any surface they encounter, how should the simulator compute this adsorption probability from the surface's physical *adsorption coefficient*? Their result (equation (10) of [18]) is

$$P_a = \kappa \sqrt{\frac{\pi \Delta t}{D}}, \quad (1)$$

where κ is the adsorption coefficient (defined below and in the glossary), Δt is the simulation time step and D is the molecular diffusion coefficient. Bartol and co-workers [7] derived essentially the same equation 16 years earlier (equation (6) of [7]) for the closely related problem of simulating reactions between solution-phase and surface-bound molecules. Also, Singer and co-workers recently generalized equation (1) to multi-dimensional systems with molecular drift and anisotropic diffusion matrices [41].

All of these derivations of equation (1) assumed *well-mixed* systems, in which molecular concentrations are spatially uniform throughout the solution. An assumption about the

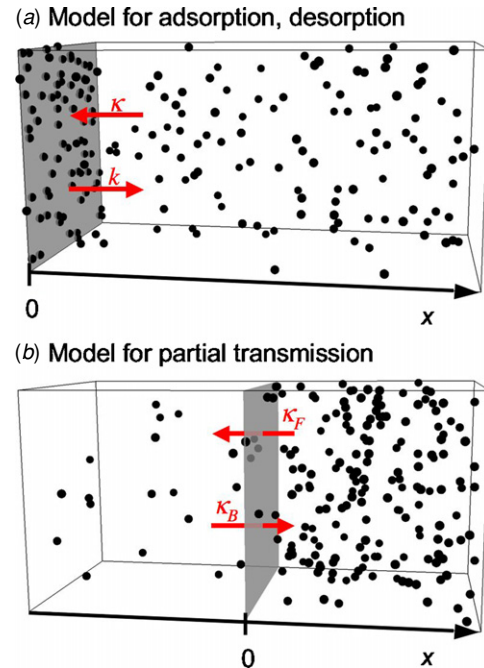


Figure 1. (a) Model system for adsorption and desorption. Molecules, shown with dots, adsorb to the gray surface with the adsorption coefficient κ and desorb with the rate constant k . (b) Model system for partial transmission. Molecules transmit from the front side of the surface ($x > 0$) to the back side ($x < 0$) with the permeability coefficient κ_F and transmit from back to front with the permeability coefficient κ_B .

average molecular positions is required for the adsorption probability to be independent of the precise position that each molecule diffused from. However, the *well-mixed assumption* is inconsistent with the problem definition because irreversible molecular adsorption to surfaces necessarily creates a local concentration gradient. Regardless of the starting state, this gradient develops rapidly in the immediate vicinity of the surface and approaches the time-invariant *steady state* (see section 3.3.1 of [16] and section 5, below). As a result, molecular concentrations near adsorbing surfaces, whether physical or simulated, are likely to spend the vast majority of their time much closer to the steady state than to the well-mixed state.

In this paper, we re-investigate the problem that Erban and Chapman posed, but for steady-state systems rather than well-mixed systems. The relationship we find between the adsorption probability and the adsorption coefficient enables substantially more accurate simulations than equation (1) does. We also develop related algorithms for simulating irreversible desorption, reversible adsorption and either irreversible or reversible transmission through permeable surfaces. Together, these algorithms enable quantitative simulation of most elementary molecule-surface interactions. Examples include peripheral membrane protein binding and release, membrane permeability and heterogeneous catalysis. Where coarse-grained approximations are appropriate, our algorithms also support simulation of ligand binding to receptors or molecular transport through channels or pores. In addition, our

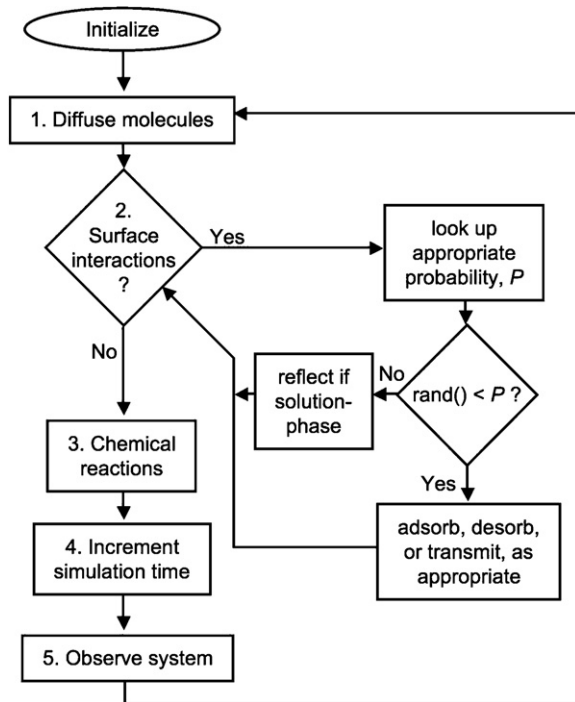


Figure 2. Particle-based simulator design assumed throughout this work. The probability, P , refers to the adsorption, desorption, or partial transmission probability, as appropriate, and ‘rand()’ is a uniformly distributed random number between 0 and 1. This paper presents molecule–surface interaction probabilities (P) with which simulators following this design will adsorb, desorb and transmit molecules at the correct rates.

adsorption algorithm enables a method for efficiently modeling unbounded diffusion in a limited spatial domain, which we discuss at the end of this paper.

A specific example that illustrates the need for accurate adsorption algorithms arises in the yeast pheromone response system (reviewed in [6]), a model eukaryotic signaling system. Upon pheromone stimulation, pheromone-sensitive cells execute a series of actions, of which a central one is that the Ste5 protein translocates from the cytoplasm to the plasma membrane [37]. Colman-Lerner and co-workers recently recorded microscopy image sequences that show this translocation [15]. Because their data are two-dimensional images of three-dimensional cells, including in-focus and out-of-focus portions, they require further analysis to determine the adsorption coefficient between Ste5 and the membrane, which could lead insights into the interaction mechanisms. This analysis could be performed relatively easily by adjusting a simple computer model until it agreed with the microscope images; the simulator that ran this model would require accurate adsorption algorithms. Afterward, the adsorption coefficient between Ste5 and the plasma membrane could be used in new spatial simulations of the pheromone response system. The resulting simulations, which would again require accurate adsorption algorithms, might lend further insights into the yeast signaling system dynamics or signal processing (see e.g. [24, 28]).

2. Model systems

We explore one model system for adsorption and desorption and another for partial transmission (figure 1). The adsorption/desorption system extends over all $x > 0$ while the partial transmission system extends over all space. Both active surfaces are at $x = 0$. While these models are three-dimensional for convenient visualization, they can be described equally well with fewer dimensions because their average molecular concentrations are translationally invariant parallel to the surfaces; the mathematics throughout this paper uses one dimension.

We assume that solution-phase molecules move solely by diffusion. This means we treat water and other unmodeled species only through their implicit contributions to the Brownian dynamics of the modeled molecules. It also means we ignore intermolecular interactions between modeled molecules, which is typically valid when their concentrations are low. These approximations allow us to describe diffusion with Fick’s laws [11, 16]:

$$J = -D \frac{\partial C(x, t)}{\partial x} \quad (2)$$

$$\frac{\partial C(x, t)}{\partial t} = D \frac{\partial^2 C(x, t)}{\partial x^2}, \quad (3)$$

where J is the average molecular flux toward positive x and $C(x, t)$ is the average concentration at position x and time t . We call $C(x, t)$, which is technically a distribution function [26], the *concentration profile*.

Adsorption/desorption model system definition. The net adsorption rate in the adsorption/desorption system is

$$\frac{\partial C_a(t)}{\partial t} = D \left. \frac{\partial C(x, t)}{\partial x} \right|_{x=0} = \kappa C(0, t) - k C_a(t), \quad (4)$$

where $C_a(t)$ is the average concentration of molecules that are adsorbed to the surface at time t and k is the *desorption rate constant*. The first equality expresses the conservation of molecules between the solution and surface phases. More precisely, it states that the rate of adsorption onto the surface equals the flux of solution-phase molecules toward the surface, measured adjacent to the surface (see equation (2)). The second equality expresses the adsorption rate in terms of the solution and surface concentrations. The first term on the right-hand side of the equation is the *Robin* or *radiation boundary condition* [12, 16, 19], in which the adsorption rate is proportional to the concentration of solution-phase molecules at the surface. The proportionality constant, κ , is the adsorption coefficient. In this model system, κ can range from zero, for a completely inert surface, to infinity, for a surface that adsorbs molecules immediately upon contact. Physically, κ is limited to about the thermal velocity of potential adsorbents [13], which is $(k_B T/m)^{1/2}$, where k_B is Boltzmann’s constant, T is the temperature and m is the molecule’s mass, because this velocity determines how frequently molecules collide with the surface (for a 50 kDa protein at 37 °C, $\kappa < 7 \times 10^6 \mu\text{m s}^{-1}$). Note that the Robin boundary condition does not depend on the surface-bound concentration, so it does not account for surface saturation (in contrast to the Langmuir

equation [27]). Also, note that authors frequently use the term ‘adsorption coefficient’ to mean other things, such as the adsorption equilibrium constant [9, 10, 40] (see below) or an exponent for a variant of the Langmuir equation [20]. The second term on the right-hand side of equation (4) is the desorption rate. Its proportionality constant, k , can also range from zero to infinity, now representing no desorption and instantaneous desorption, respectively. According to Chou and D’Orsogna [13], desorption rates are typically between 10^{-4} s^{-1} and 10^4 s^{-1} .

Partial transmission model system. The net flux of molecules through the permeable surface, from front to back, is

$$D \left. \frac{\partial C(x, t)}{\partial x} \right|_{x=-0} = D \left. \frac{\partial C(x, t)}{\partial x} \right|_{x=+0} = \kappa_F C(+0, t) - \kappa_B C(-0, t). \quad (5)$$

In analogy to prior definitions, κ_F and κ_B are the *permeability coefficients* from the front of the surface (which faces the positive x -axis) to the back, and from the back to the front, respectively. Equation (5) specifies the surface sides with $+0$ for the front and -0 for the back because the concentration profile is typically discontinuous at $x = 0$. As before, the first equality expresses the conservation of molecules and the second equality represents the Robin boundary condition. Again, this equation ignores saturation; for example, it does not account for ion channels operating at capacity. In general, permeability coefficients are large for molecules that tend to partition into the membrane as well as for molecules with high diffusion coefficients within the membrane (see [8]). Paula *et al* [33] found that permeability coefficients across a 27 Å phosphatidylcholine bilayer are about $3.5 \times 10^{-8} \mu\text{m s}^{-1}$ for potassium ions, $0.014 \mu\text{m s}^{-1}$ for urea, $0.027 \mu\text{m s}^{-1}$ for glycerol and $150 \mu\text{m s}^{-1}$ for water.

3. Simulator design

Figure 2 presents the simulator design assumed throughout this work. Smoldyn [2] follows this design, and MCell [22] and ChemCell [35] use similar approaches.

In step 1, the simulator diffuses molecules for a fixed amount of time with Gaussian-distributed random displacements, while ignoring any surfaces or bimolecular interactions. The standard deviation of these Gaussian distributions, called the *root mean square (rms) step length*, is

$$s = \sqrt{2D\Delta t}. \quad (6)$$

This is a characteristic length for the simulation, against which we previously compared intermolecular interaction distances [3] and surface curvature radii [2]. Molecules move about an rms step length at each time step, so the planar surfaces in our model systems accurately represent membranes that have curvature radii which are much larger than the rms step length, and are less accurate for more tightly curved membranes. For either model system, a diffusive step transforms a starting concentration profile, $C(x, t)$, to

$$C^{(1)}(x, t) = C(x, t) * \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2s^2}\right). \quad (7)$$

The star represents convolution and the term on its right is a Gaussian with unit area and standard deviation s .

In step 2, the simulator addresses any solution-phase molecules that diffused across the surface in step 1, as well as the possible desorption of any surface-bound molecules. The former molecules adsorb to the surface with probability P_a , transmit through it with probability P_t , and otherwise reflect off of it using ballistic reflection [3]. The latter molecules desorb with probability P_d ; the simulator displaces desorbed molecules away from the surface according to the probability density $p(x)$. We solve for P_a , P_t , P_d and $p(x)$ below. The simulator does not perform multiple surface interactions, such as both adsorption and desorption, on the same molecule so as to improve computational efficiency and to make their order of execution irrelevant (however, the algorithms do account for the multiple surface interactions that physical molecules may undergo). Step 2 transforms the adsorption/desorption system concentration profile to

$$C^{(2)}(x, t) = \begin{cases} C^{(1)}(x, t) & x \leq 0 \\ C^{(1)}(x, t) + (1 - P_a) C^{(1)}(-x, t) + P_d C_a(t) p(x) & x > 0 \end{cases}. \quad (8)$$

It also transforms the adsorbed concentration to

$$C_a^{(2)}(t) = (1 - P_d) C_a(t) + P_a \int_{-\infty}^0 C^{(1)}(x, t) dx. \quad (9)$$

The analogs of these equations for the partial transmission system are not shown because they are more complicated and less useful.

This design for step 2 incorporates two simplifications for good computational efficiency. First, P_a and P_t do not depend on how far molecules are from surfaces, but instead depend only on static properties, such as diffusion and adsorption coefficients. This allows a simulator to compute these probabilities just once for a simulation, rather than once for each molecule–surface interaction. Secondly, the simulator ignores all molecules in the underlying model systems that diffused across the surface during step 1, but that returned back to the side where they started before the end of the time step (see section 3.2 of [3] and section 2.2 of [18]). In preliminary work, accounting for these additional surface contacts proved computationally intensive, complicated to implement and minimally more accurate.

In step 3, the simulator performs chemical reactions [3]. We assume that any reactions are slow enough that their effects on concentrations near surfaces are negligible, where the rms step length provides the necessary length scale. In step 4, the simulator increments the simulation time. This updates the concentration profile and surface concentration to

$$C(x, t + \Delta t) = C^{(2)}(x, t) \quad (10)$$

$$C_a(t + \Delta t) = C_a^{(2)}(t). \quad (11)$$

Equation (10) applies to both model systems, while equation (11) only applies to the adsorption/desorption system. By definition, $C(x, t + \Delta t)$ equals the starting concentration profile, $C(x, t)$, if and only if they represent a steady state. Ordinarily, the adsorbed concentration, $C_a(t)$, would also have to be constant at steady state. In fact, we

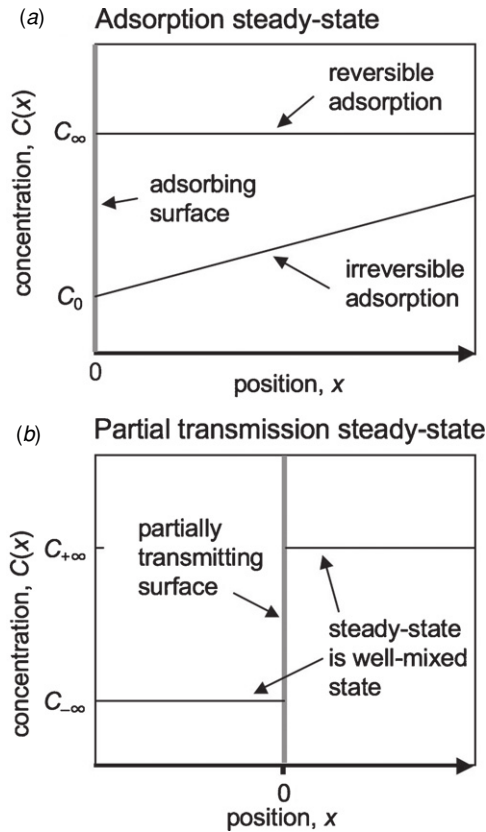


Figure 3. Steady-state concentration profiles for model systems that (a) reversibly or irreversibly adsorb molecules to a surface or (b) reversibly transmit molecules through a surface. The gray line at $x = 0$ represents the surface.

require this time independence for reversible adsorption but not for irreversible adsorption. We do not require it for irreversible adsorption because (i) we wish to investigate a constant adsorption flux and (ii) the adsorbed concentration does not affect other system aspects. Finally, the simulator performs any observations of the system in step 5. It then returns to step 1 to start the next time step.

4. Interaction probabilities

In this section, we relate the molecule–surface interaction probabilities to their respective coefficients for accurate steady-state simulation. We also derive the necessary probability densities for desorbed molecule displacements. These results are presented as five algorithms, along with several variants. Figure 5 presents the main results graphically, the supplementary information (available at stacks.iop.org/PhysBio/6/046015) presents them in tabular form, and several equations below approximate them with interpolating functions. Also, the supplementary information includes the C language library files that Smoldyn uses to perform the necessary table look-up and interpolation. Except as noted, most results in this section were computed with Mathematica software [44].

Unitless reduced variables, shown with a prime symbol, simplify several results. We use the simulation time step as the standard time unit and the rms step length as the standard

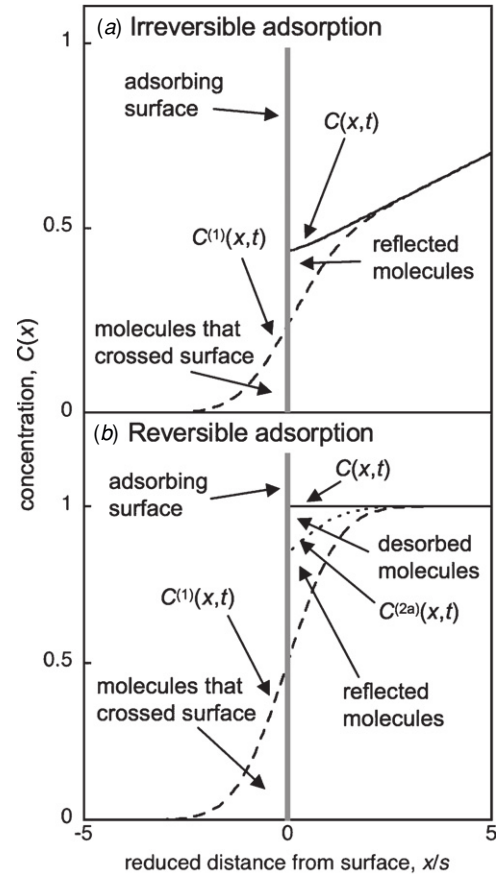


Figure 4. Steady-state concentration profiles at different steps of the simulation process. These apply to (a) irreversible adsorption and (b) reversible adsorption. Solid lines represent the profiles at both the beginning and end of time steps, long-dashed lines represent the profiles after the simulator’s step 1, and the short-dashed line in panel B represents the profile after the reflection and adsorption portions of step 2, but before desorption (this situation does not apply to panel A). Areas on the figure represent molecular populations, as labeled. The gray line at $x = 0$ represents the surface. Panel A results were computed with the simulation emulator (see the main text and supplementary information available at stacks.iop.org/PhysBio/6/046015) and panel (b) with Mathematica software [44].

length unit. From these, the reduced adsorption coefficient, desorption rate constant and diffusion coefficient are

$$\kappa' \equiv \frac{\kappa \Delta t}{s} = \frac{\kappa \sqrt{\Delta t}}{\sqrt{2D}} \quad (12)$$

$$k' \equiv k \Delta t \quad (13)$$

$$D' \equiv \frac{D \Delta t}{s^2} = \frac{1}{2}. \quad (14)$$

Reduced permeability coefficients, κ_F' and κ_B' , are analogous to the reduced adsorption coefficient (equation (12)).

4.1. Algorithm I: irreversible adsorption

Consider the adsorption/desorption system and assume irreversible adsorption ($\kappa > 0$ and $k = 0$). Solving equations (3) and (4) shows that the model steady-state concentration profile,

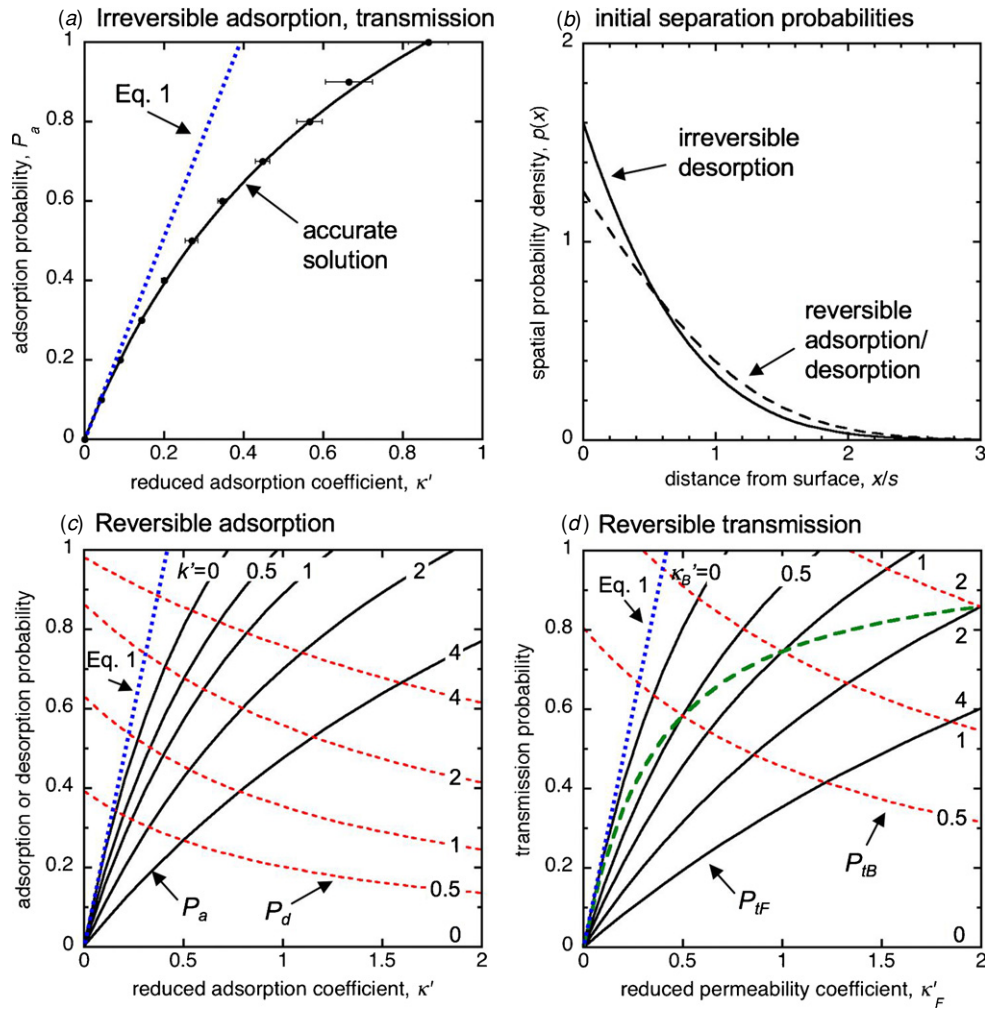


Figure 5. Relationships between transition coefficients and transition probabilities, and initial separations between surfaces and desorbed molecules. (a) Relationship between κ' and P_a for irreversible adsorption. The solid black line, calculated by the simulation emulator and approximated by equation (21), enables accurate adsorption simulation. Points with one standard deviation error bars represent results from stochastic simulations of a steady-state system (see supplementary information for details available at stacks.iop.org/PhysBio/6/046015). (b) Probability densities for the initial separation between a desorbed molecule and the surface that it desorbed from. The solid line represents the density that should be used for irreversible desorption (equation (25)) and the dashed line represents the density for reversible adsorption (equation (33)). (c) Relationships between P_a (solid black lines), P_d (dashed red lines), κ' (abscissa) and k' (marked on figure) for reversible adsorption; these are also given in equations (32) and (37). The $k' = 0$ desorption probability line is coincident with the abscissa. (d) Relationships between P_{tF} (solid black lines), P_{tB} (dashed red lines), κ'_F (abscissa) and κ'_B (marked on figure) for reversible transmission; these are also given in equations (47) and (48). The $\kappa'_B = 0$ transmission probability line is coincident with the abscissa. The green bold long-dashed line presents the transmission probabilities for the condition that $\kappa'_B = \kappa'_F$. Where present, the dotted blue line represents the relationship given by equation (1).

$C(x)$, increases linearly with distance away from the surface and the surface concentration increases at a constant rate (figure 3(a)),

$$C(x) = C_0 \left(\frac{\kappa}{D} x + 1 \right), \quad x > 0 \quad (15)$$

$$\frac{\partial C_a(t)}{\partial t} = \kappa C_0, \quad (16)$$

where C_0 is the solution-phase concentration at $x = 0$. (Equation (3) and the outer portions of equation (4) are the equations of motion for $C(x,t)$ and $C_a(t)$, respectively, while the latter two portions of equation (4) give the necessary boundary condition.) Equation (15) becomes physically unreasonable as the system increases in size because $C(x)$ is unbounded; however, this model is likely to be physically correct near the

surface, so it is adequate for our work. As before, note that the concentration profile and adsorption flux are time invariant at steady state.

In an *exact algorithm*, meaning one in which the simulated system is theoretically indistinguishable from the underlying model system, the average simulated steady-state concentration profile would equal equation (15). We determine whether this is possible by seeing if our simulator design can preserve the model steady-state profile over one time step. Step 1 (equation (7)) transforms equation (15) to

$$C^{(1)}(x, t) = \int_0^\infty C_0 \left(\frac{\kappa}{D} x' + 1 \right) \frac{1}{s\sqrt{2\pi}} \exp \left[-\frac{(x-x')^2}{2s^2} \right] dx'. \quad (17)$$

Steps 2–4 (equations (8) and (10)) then transform this profile to

$$C(x, t + \Delta t) = C_0 \left\{ \kappa' \sqrt{\frac{2}{\pi}} (1 + P_a) e^{-\frac{x^2}{2}} + \left[1 + 2\kappa' x - \frac{1 - P_a + 2\kappa' (1 + P_a) x}{2} \operatorname{erfc}\left(\frac{x}{\sqrt{2}}\right) \right] \right\} \quad (18)$$

for $x > 0$. Equation (18) is the concentration profile at the end of the time step. Comparing equations (15) and (18) shows that they cannot equal each other for arbitrary κ' and x values. Thus, equation (15) is a steady state for the model system but *not* for our simulator design, which means that our simulator design cannot treat irreversible adsorption exactly. (The simplifications in step 2, which we included for computational efficiency, cause this inexactness.)

Knowing that simulated concentration profiles would necessarily be inexact, I computed the adsorption probabilities that yield the correct adsorption rates with a ‘simulation emulator’ (see figure 4(a) and the supplementary information available at stacks.iop.org/PhysBio/6/046015). Each emulator run started with a fixed adsorption probability and a well-mixed numerical concentration profile (which extended from -6 to $+10$ rms step lengths and was tabulated at 401 equally spaced points, plus additional points immediately above and below 0 for higher accuracy). The emulator iterated steps 1 and 2, using equations (7)–(9), until the system was essentially at steady state (as determined by the adsorption rate changing by less than 0.01% between sequential iterations). It addressed the region with $x < -6s$ by extrapolating the concentration profile with an error function, and it addressed the region with $x > 10s$ by fixing these concentrations to 1. Once at steady state, it fit a straight line to the final concentration profile (x ranged from 3 s to 7 s to avoid edge effects), from which it calculated the effective reduced adsorption coefficient using

$$\kappa' = \frac{m}{2C_0}. \quad (19)$$

Equation (19) derives from equation (15), with m as the concentration profile slope ($C_0\kappa/D$) and C_0 as the intercept. Because the emulator could not actually reach the steady state, it performed the above routine twice, starting from different states; it started with $C(x, 0)$ equal to either 0 or 1 so that it would approach the steady-state profile from both above and below. Finally, the emulator averaged the two results to yield a final κ' value. These κ' values are almost certainly accurate to within $\pm 10^{-6}$ because (i) the two intermediate solutions never differed by more than 10^{-7} , and (ii) more conservative choices for emulator parameters (number of tabulated points, size of x -domain and number of iterations) affected results by even smaller amounts.

Figure 5(a) shows the emulator results, oriented so that P_a depends on κ' . Figure 5(a) also shows stochastic simulation results, generated by Smoldyn, for systems that were similar to those that the emulator quantified (supplementary information available at stacks.iop.org/PhysBio/6/046015). The close agreement between the two sets of results lends confidence in the emulator software, the Smoldyn software and the mathematical analysis. When reduced adsorption coefficients

are small, the emulator results agree with equation (1). This makes sense because small κ' values imply either: (i) small concentration gradients (because κ is small or D is large), so the steady state is similar to the well-mixed state, or (ii) a small simulation time step, so molecules diffuse over short distances and the concentration gradient affects the simulated adsorption rate minimally. On the other hand, the emulator results are up to a factor of 2 smaller than equation (1) for larger κ' values. This implies that simulators basing their adsorption probabilities on equation (1) would simulate adsorption too quickly in many situations. Finally, figure 5(a) shows that κ' values larger than about 0.86 cannot be modeled with our simulator design because the adsorption probability cannot exceed 1. This means that large adsorption coefficients can only be simulated quantitatively by using small time steps. (This limitation arises from simplifications that we made for computational efficiency in the simulator’s step 2.)

Polynomial fits to the emulator’s relationship between κ' and P_a , which may be more convenient than the raw data, are

$$\kappa' = \frac{1}{\sqrt{2\pi}} P_a + 0.247\ 61 P_a^2 + 0.006\ 16 P_a^3 + 0.203\ 84 P_a^4 \quad (20)$$

$$P_a = \kappa' \sqrt{2\pi} - 3.333\ 21 \kappa'^2 + 3.356\ 69 \kappa'^3 - 1.520\ 92 \kappa'^4. \quad (21)$$

The initial terms were constrained with equation (1) to make these fits exact as κ' or P_a tends toward 0. These equations agree with the tabulated data to within 1% over their entire domains.

4.2. Algorithm II: irreversible desorption

Continuing with the adsorption/desorption system, we now consider irreversible desorption ($\kappa = 0$ and $k > 0$). Desorption is a simple first-order process, like a first-order chemical reaction, so a molecule should desorb during a time step with probability [3, 42]

$$P_d = 1 - e^{-k\Delta t}. \quad (22)$$

This is exact for all possible desorption rate constants and simulation time steps.

The more difficult question concerns where the simulator should place desorbed molecules. In one algorithm variant, the simulator places them adjacent to the surface. This ignores diffusion occurring between when a molecule desorbs and the end of the time step, which does not affect the accuracy of the total surface- and solution-phase concentrations, but means that this algorithm variant yields inexact concentration profiles.

A better variant displaces desorbed molecules away from the surface. To calculate the probability density for this displacement, consider a molecule that desorbs at time t , where t is between the start of the time step at time 0 and the end at time Δt . The probability density for this molecule’s position at time Δt is [16]

$$p(x, \Delta t | \text{desorb at } t) = \frac{1}{\sqrt{\pi D (\Delta t - t)}} e^{-\frac{x^2}{4D(\Delta t - t)}}, \quad x > 0. \quad (23)$$

We denote the probability density that the molecule desorbs at time t , given that it desorbs between 0 and Δt , as $p_t(t)$. Treating $p_t(t)$ as a weighting factor for equation (23), the spatial probability density for a molecule that desorbs at an unknown time between 0 and Δt is

$$p(x, \Delta t) = \int_0^{\Delta t} \frac{p_t(t)}{\sqrt{\pi D(\Delta t - t)}} e^{-\frac{x^2}{4D(\Delta t - t)}} dt, \quad x > 0. \quad (24)$$

To be consistent with equation (22), $p_t(t)$ should equal $k/P_d \exp(-kt)$. However, this makes $p(x, \Delta t)$ depend on k , which is inconvenient. So, we make the approximation that the molecule is equally likely to desorb at any time during the time step, which simplifies $p_t(t)$ to $1/\Delta t$. This approximation is valid when $k\Delta t \ll 1$ because the exact exponential function approaches $1/\Delta t$ in this limit. Because equation (22) already ensures accurate surface and solution concentrations and we are just refining the solution concentration profile here, this approximation is also likely to be acceptable for substantially larger values of $k\Delta t$ as well. Using the approximation, equation (24) integrates to

$$p(x) = \frac{2\sqrt{2}}{s\sqrt{\pi}} e^{-\frac{x^2}{2s^2}} - \frac{2x}{s^2} \operatorname{erfc} \frac{x}{s\sqrt{2}}. \quad (25)$$

Figure 5(b) shows this displacement probability density with a solid line.

A final task makes equation (25) useful. We need a relationship with which a simulator can transform random numbers that are uniformly distributed between 0 and 1 into random numbers that obey the probability density in equation (25). We do this by integrating and inverting equation (25) [36]. The integral, from 0 to x , is

$$P(x) = \frac{x\sqrt{2}}{s\sqrt{\pi}} e^{-\frac{x^2}{2s^2}} - \frac{x^2}{s^2} + \left(1 + \frac{x^2}{s^2}\right) \operatorname{erf} \frac{x}{s\sqrt{2}}. \quad (26)$$

This cannot be solved for x analytically, so I inverted it numerically. A least-squares rational function fit to the result is

$$x = s \frac{0.571825P - 0.552246P^2}{1 - 1.53908P + 0.546424P^2}. \quad (27)$$

Equation (27) converts a uniform deviate, P , into the desired deviate, x . Compared to the numerical solution, it is in error by less than 0.02 s for nearly all P -values.

4.3. Algorithm III: reversible adsorption/desorption

If adsorption is reversible ($\kappa > 0$ and $k > 0$), the adsorption and desorption rates equal each other for the system to be at steady state. Adding this condition to equations (3) and (4) yields the steady-state concentrations (figure 3(a)),

$$C(x) = C_\infty, \quad x > 0 \quad (28)$$

$$C_a = \frac{\kappa}{k} C_\infty, \quad (29)$$

where C_∞ is the solution concentration. In contrast to the preceding investigations of irreversible processes, this steady state is a well-mixed state. This steady state is

also an equilibrium state because there are no net molecular fluxes. The ratio κ/k , which clearly equals C_a/C_∞ , is the adsorption equilibrium constant; it is analogous to the familiar equilibrium constant for reversible reactions [5].

As in section 4.1, we determine whether our simulator design can treat reversible adsorption exactly by seeing if it can preserve the model concentration profile over one time step (see figure 4(b)). Substituting equation (28) with equation (7) yields

$$C^{(1)}(x, t) = C_\infty \left(1 - \frac{1}{2} \operatorname{erfc} \frac{x}{2\sqrt{D\Delta t}}\right). \quad (30)$$

The adsorption and reflection portions of step 2 (see equation (8)) then transform the concentration profile to

$$C^{(2a)}(x, t) = C_\infty \left(1 - \frac{P_a}{2} \operatorname{erfc} \frac{x}{2\sqrt{D\Delta t}}\right). \quad (31)$$

Finally, the desorption portion of step 2 transforms this result back to the starting condition, if and only if

$$\frac{P_a}{P_d} = \frac{\kappa}{k} \sqrt{\frac{\pi}{D\Delta t}} \quad \text{and} \quad (32)$$

$$p(x) = \frac{1}{2} \sqrt{\frac{\pi}{D\Delta t}} \operatorname{erfc} \frac{x}{2\sqrt{D\Delta t}}. \quad (33)$$

Thus, our simulation design can simulate *reversible* adsorption exactly under equilibrium conditions. Equation (32), which relates the adsorption and desorption probabilities, arises from the constraint that equal numbers of molecules must adsorb and desorb during each time step, on average. Equation (33), which is the probability density for desorbed molecule displacements and is shown in figure 5(b) with a dashed line, causes desorbed molecules to exactly replace those that adsorbed. This density is the ratio of the function $C(x, t + \Delta t) - C^{(2a)}(x, t)$ and its integral (the ‘desorbed molecules’ region of figure 4(b)), so that the probability density has unit area.

We integrate and invert equation (33) to make it useful. Integration yields

$$P(x) = 1 - e^{-\frac{x^2}{2s^2}} + \frac{x}{s} \sqrt{\frac{\pi}{2}} \operatorname{erfc} \frac{x}{s\sqrt{2}}. \quad (34)$$

A rational function fit to the numerically calculated inverse function is

$$x = s \frac{0.729614P - 0.70252P^2}{1 - 1.47494P + 0.484371P^2}. \quad (35)$$

As before, P is a uniform deviate between 0 and 1 and x is the initial displacement for a desorbed molecule. This fit is in error by less than 0.02 s for nearly all P -values.

The remaining question is how P_a or P_d should be chosen. Their solutions for irreversible adsorption (equation (21)) and irreversible desorption (equation (22)) do not simultaneously satisfy the equilibrium condition of equation (32) because the prior results did not consider molecules that both adsorb and desorb during a single time step. We address them with the following conceptual scheme: we label all molecules as being solution-phase or surface-bound at time 0, the system evolves over one time step, and we then use the labels to see how many molecules had a net adsorption transition by time Δt . We solve this problem in appendix A. The answer,

equation (A.12), represents the *model* adsorbed molecules at the end of a time step. Meanwhile, the expectation *simulated* adsorbed molecules for one time step (see figure 4(b) and equation (9)) is

$$\text{adsorbed amount} = P_a \int_{-\infty}^0 C^{(1)}(x, t) dx = P_a \frac{s}{\sqrt{2\pi}}. \quad (36)$$

Equating these amounts and solving for P_a yields the adsorption probability,

$$P_a = \frac{\kappa' \sqrt{2\pi} [c_2 - c_1 - c_2 e^{c_1^2} \operatorname{erfc}(c_1) + c_1 e^{c_2^2} \operatorname{erfc}(c_2)]}{c_1 c_2 (c_2 - c_1)} \quad (37)$$

$$c_1 = \frac{\kappa' - \sqrt{\kappa'^2 - 2k'}}{\sqrt{2}} \quad (38)$$

$$c_2 = \frac{\kappa' + \sqrt{\kappa'^2 - 2k'}}{\sqrt{2}}. \quad (39)$$

Figure 5(c) shows this adsorption probability and the corresponding desorption probability (from equation (32)) for different κ' and k' values. As before, (i) the adsorption probability approaches equation (1) as κ' is reduced to zero, (ii) equation (1) simulates adsorption too quickly with larger κ' values and (iii) κ' values that exceed some cut-off, which now depends on k' , cannot be simulated quantitatively with our simulator design because adsorption probabilities cannot exceed 1. In the limit that κ' is reduced to zero, P_d approaches its value for irreversible desorption (equation (22)), as one would expect. In contrast though, P_a does *not* approach the irreversible adsorption probability (equation (21)) as k' is reduced to zero. This is because the concentration profile remains well mixed for the reversible adsorption case, even as k' is reduced toward zero, which is a consequence of the fact that the reversible adsorption steady state is an equilibrium state (as k' is reduced toward zero, the adsorption equilibrium constant and the adsorbed concentration increase toward infinity); in contrast, the concentration profile is sloped for the irreversible adsorption case.

4.4. Algorithm IV: irreversible transmission

The partially transmitting system (figure 1(b)), with irreversible transmission from the front face to the back face ($\kappa_F > 0$ and $\kappa_B = 0$), is essentially identical to the irreversible adsorption system. Its steady-state concentration profile is

$$C(x) = C_{\pm 0} \left(\frac{\kappa_F}{D} x + 1 \right), \quad (40)$$

where C_{+0} represents the concentration at the front of the surface and is used if $x > 0$, and C_{-0} represents the concentration at the back of the surface and is used if $x < 0$. As we found before, our simulation design cannot simulate concentration profiles exactly for steady-state irreversible transmission, but it can achieve accurate concentrations on each side of the surface; figure 5(a) shows the transmission probability as a function of the transmission coefficient, and equations (20) and (21) are polynomial fits to the relationship between P_t and κ_F .

The question remains where the simulator should place transmitted molecules. It can, of course, place them at

the backside of the surface, although this ignores diffusion occurring between transmission and the end of the time step. A better algorithm variant accounts for this diffusion using the probability density given in equation (25). Whereas this density was inexact before, it is exact now because a steady-state irreversibly transmitting surface acts as a constant molecular source to the region with $x < 0$. In yet a third algorithm variant, the simulator does not move transmitted molecules, but simply leaves them where they ended up after step 1 of the simulation process. The resulting probability density (which can be approximated with the $x < 0$ portion of equation (17)) is very similar to the exact result. This is the most computationally efficient variant.

4.5. Algorithm V: reversible transmission

Finally, consider the reversible partially transmitting system ($\kappa_F > 0$ and $\kappa_B > 0$). From equations (3) and (5), the model steady state is well mixed on each side of the surface but is likely discontinuous at the surface,

$$C(x) = \begin{cases} C_{+\infty} & x > 0 \\ C_{-\infty} & x < 0 \end{cases}, \quad (41)$$

$$\frac{C_{+\infty}}{C_{-\infty}} = \frac{\kappa_B}{\kappa_F}. \quad (42)$$

This steady state is an equilibrium state, for which each side of equation (42) represents the chemical partition coefficient [5]. Physically, the concentration discontinuity can be maintained actively, such as by trans-membrane molecular pumps, or passively, such as by chemical partitioning between different solvents.

Our simulator design can simulate reversible transmission exactly under equilibrium conditions. The starting concentration profile (equation (41)) diffuses to

$$C^{(1)}(x, t) = \frac{C_{-\infty}}{2} \operatorname{erfc} \frac{x}{2\sqrt{D\Delta t}} + \frac{C_{+\infty}}{2} \operatorname{erfc} \frac{-x}{2\sqrt{D\Delta t}}, \quad (43)$$

where the first term represents molecules that diffuse from $x < 0$ and the second term represents molecules that diffuse from $x > 0$. In step 2, the simulator transmits molecules that diffused across the surface with probabilities P_{tF} and P_{tB} for the front and back faces, respectively. The number of molecules that the simulator transmits from the front to the back is the integral of the second term of equation (43) from $-\infty$ to 0 (see equation (36)), times P_{tF} . This result, and the analogous one for transmission the other way, is

$$\text{amount from front} = P_{tF} C_{+\infty} \sqrt{\frac{D\Delta t}{\pi}} \quad (44)$$

$$\text{amount from back} = P_{tB} C_{-\infty} \sqrt{\frac{D\Delta t}{\pi}}. \quad (45)$$

These are equal at steady state. Combining them with equation (42) yields

$$\frac{P_{tB}}{P_{tF}} = \frac{\kappa_B}{\kappa_F}. \quad (46)$$

To return the concentration profile to the steady state by the end of the time step, the simulator can displace transmitted

molecules away from the surface according to equation (25) or just leave them where they ended up after step 1. These produce the same steady-state concentration profiles and so are equally accurate at steady state, although the latter variant is more computationally efficient.

We find P_{tF} much as we found P_a for the reversible adsorption problem. Now, we solve: given an initially well-mixed concentration profile for $x > 0$, no concentration for $x < 0$, and a reversible partially transmitting boundary, what is the total number of molecules in the region $x < 0$ at time Δt ? Appendix B presents the result. The solution, equation (B.12), represents the number of molecules that the simulator *should* transmit over one time step. We equate it to equation (44), which represents the number that the simulator *does* transmit, and solve for the transmission probabilities. They are

$$P_{tF} = \frac{\kappa'_F}{(\kappa'_B + \kappa'_F)^2} \left\{ 2(\kappa'_B + \kappa'_F) - \sqrt{\frac{\pi}{2}} + \sqrt{\frac{\pi}{2}} e^{2(\kappa'_B + \kappa'_F)^2} \operatorname{erfc}[\sqrt{2}(\kappa'_B + \kappa'_F)] \right\} \quad (47)$$

$$P_{tB} = P_{tF} \frac{\kappa'_B}{\kappa'_F}. \quad (48)$$

Figure 5(d) presents these probabilities.

The results are familiar: transmission probabilities approach equation (1) as κ'_F and κ'_B approach zero; equation (1) simulates transmission too quickly with larger κ'_F and κ'_B values, and, in most cases, a simulator can only quantitatively simulate transmission up to finite κ'_F and κ'_B values. The exception to the last result occurs if $\kappa'_F = \kappa'_B$ (the bold dashed line in figure 5(b)), which is typical for passive membranes.

5. Algorithm verification and comparison

All of the algorithms presented above rest on firm theoretical foundations that ensure high accuracy for steady-state systems. Here, we investigate their accuracy away from steady state. Figure 6 shows results from several algorithm tests, run with the Smoldyn simulator. Each test started with 20 000 molecules that were well mixed within box-shaped regions and that had $5 \mu\text{m}^2 \text{s}^{-1}$ diffusion coefficients (see figure 1; x extended from 0 to $2 \mu\text{m}$ and y and z were each $1 \mu\text{m}$ wide). These are reasonably typical parameters for proteins in bacteria [17, 43]. Initially, no molecules were adsorbed to a surface at $x = 0$ and there were no molecules in the region $x < 0$. Simulation time steps were 1 ms, which is larger than the $0.25 \mu\text{s}$ to 0.1ms time steps that modelers typically use [14, 25, 29–31], both because this tested the algorithms where they are likely to be weakest and because a goal of this work is to enable accurate simulations with longer time steps. Depending on the specific test, molecules interacted with the surface with irreversible adsorption, reversible adsorption or reversible transmission. These tests used either ‘fast’ rates, with the maximum possible adsorption or transmission coefficients that the time step allowed, or ‘slow’ rates, with coefficients that led to adsorption or transmission probabilities to be about 0.1 (the figure 6 caption presents the coefficients). Figure 6(a)

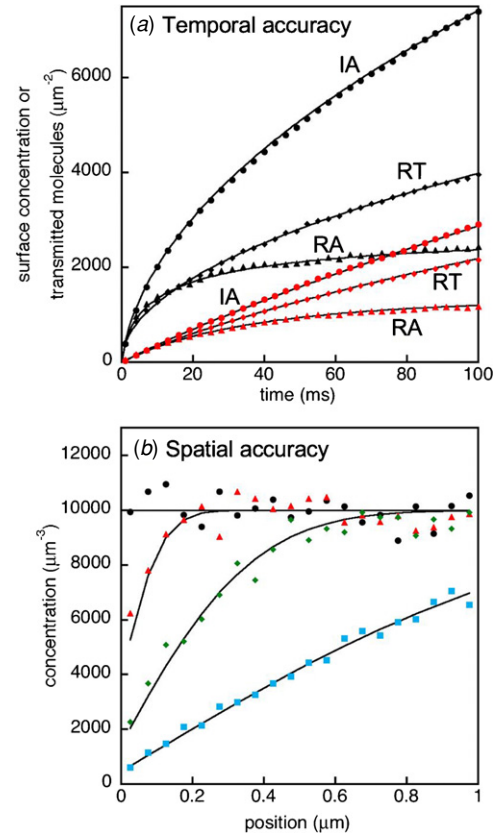


Figure 6. Algorithm tests for systems that started with well-mixed states. Dots represent stochastic simulation results and lines represent predictions for the corresponding model systems. (a) Number of molecules adsorbed to a surface or transmitted through the surface, as appropriate, in tests of irreversible adsorption (IA, circles), reversible adsorption (RA, triangles) or reversible transmission (RT, diamonds). Upper black points are for ‘fast’ rates: $\kappa = 85.9 \mu\text{m s}^{-1}$ for irreversible and reversible adsorption ($P_a = 1$), $k = 276 \text{s}^{-1}$ for reversible adsorption ($P_d = 0.128$) and $\kappa_F = \kappa_B = 5045 \mu\text{m s}^{-1}$ for reversible transmission ($P_{tF} = P_{tB} \approx 1$). Lower red points are for ‘slow’ rates: $\kappa = 4.23 \mu\text{m s}^{-1}$ for irreversible and reversible adsorption ($P_a = 0.1$), $k = 28 \text{s}^{-1}$ for reversible adsorption ($P_d = 0.026$), and $\kappa_F = \kappa_B = 4.36 \mu\text{m s}^{-1}$ for reversible transmission ($P_{tF} = P_{tB} = 0.1$). Every third simulation data point is shown for clarity. (b) Concentration profiles at several time points for the irreversible adsorption test with the ‘fast’ rate. Black circles represent time 0, red triangles represent time 1 ms (1 time step), green diamonds represent time 10 ms and blue squares represent time 100 ms. See supplementary information for details available at stacks.iop.org/PhysBio/6/046015.

compares these stochastic simulation test results with exact calculations, where the latter are

$$C_a(t) = \frac{C_\infty D}{\kappa} \left[e^{\frac{\kappa^2 t}{D}} \operatorname{erfc} \left(\kappa \sqrt{\frac{t}{D}} \right) + 2\kappa \sqrt{\frac{t}{\pi D}} - 1 \right] \quad (49)$$

(Equation 3.37 of [16]) for irreversible adsorption, equation (A.12) for reversible adsorption and equation (B.12) for reversible transmission. In all cases, the simulation results agreed essentially perfectly with the model ones. This is true at all time points, despite the fact that the system was far from steady state for the initial ones. Additional tests (not shown) performed equally well for smaller adsorption and transmission coefficients.

Figure 6(b) shows several snapshots of molecular spatial distributions from the irreversible adsorption test that used the ‘fast’ rate. A comparison with the model profiles,

$$C(x, t) = C_\infty \left[\operatorname{erfc} \frac{x}{2\sqrt{Dt}} - \exp^{\frac{\kappa x}{D} + \frac{\kappa^2 t}{D}} \operatorname{erfc} \left(\frac{x}{2\sqrt{Dt}} + \kappa \sqrt{\frac{t}{D}} \right) \right] \quad (50)$$

(equation 3.35 of [16]), again showed excellent agreement at all time points. As before, additional tests that used smaller adsorption coefficients performed equally well. Note that even though the system shown in figure 6(b) was well mixed initially, its concentration profile evolved rapidly in the immediate vicinity of the surface (the rms step length for this simulation was 0.1 μm) toward a steady-state profile; this is the reason why we developed our algorithms for steady-state systems.

These tests indicate that our irreversible adsorption, reversible adsorption and reversible transmission algorithms are accurate not only for steady-state systems, but also for those away from steady state. Furthermore, they are accurate over the entire range of adsorption or transmission coefficients accessible for a given simulation time step. Considering the other two algorithms, the irreversible desorption probability presented above (equation (22)) is accurate for all systems because it does not rely on the steady-state assumption, and the irreversible transmission algorithm is essentially identical to the irreversible adsorption algorithm, which we just verified.

In contrast, simulations based on equation (1) performed less well. First, they could not be tested with the ‘fast’ set of adsorption or transmission coefficients because equation (1) does support such large coefficients (equation (1) only allows κ' values up to 0.399, whereas they can reach 0.859 for our irreversible adsorption algorithm; see figure 5(a)). For the ‘slow’ set of coefficients, adsorption was about 6% too fast and partial transmission about 10% too fast, at all time points. With larger adsorption or transmission coefficients, use of equation (1) led simulations to irreversibly adsorb molecules up to 2.1 times too fast, or to reversibly adsorb or transmit molecules with even larger errors (the maximum errors depended on the rates of the reverse processes).

6. Simulating unbounded diffusion with partial absorption

Our final algorithm, which applies to all spatial simulation methods, builds on the irreversible adsorption algorithm. It addresses the following situation: suppose a modeler wants to investigate biochemical interactions in unbounded space, such as between extracellular pheromones and cell-surface receptors, but wants to limit the simulated spatial domain for computational efficiency [2]. The modeler needs to know how much space to simulate explicitly and how to treat its boundary. Clearly, the boundary should be neither completely reflective, which would keep molecules from diffusing away permanently, nor completely absorbing, which would inaccurately represent those molecules that diffuse away

and then return. We propose the intermediate solution that the boundary should absorb molecules with the absorption coefficient that causes the simulated concentrations to equal those for the unbounded model system. We solve for this absorption coefficient here.

We derive the absorption coefficient for three-dimensional systems with stationary sources that emit molecules at constant rates, without significant obstacles to diffusion and without significant molecular turnover through reactions. We number the sources with index j : source j is located at \mathbf{r}_j and emits molecules with rate q_j . For our unbounded model, the steady-state molecular concentration at location \mathbf{r} is [16]

$$C(\mathbf{r}) = \sum_j \frac{q_j}{4\pi D |\mathbf{r} - \mathbf{r}_j|}. \quad (51)$$

The flux at \mathbf{r} is computed from the concentration gradient using Fick’s law, as in equation (2),

$$\mathbf{J}(\mathbf{r}) = \sum_j \frac{q_j (\mathbf{r} - \mathbf{r}_j)}{4\pi |\mathbf{r} - \mathbf{r}_j|^3}. \quad (52)$$

The Robin boundary (equation (4)) relates the flux into the surface with the concentration at the surface, which we generalize to more than one dimension to yield

$$\mathbf{J}(\mathbf{r}) = \kappa(\mathbf{r}) C(\mathbf{r}) \tilde{\mathbf{n}}(\mathbf{r}), \quad (53)$$

where $\tilde{\mathbf{n}}(\mathbf{r})$ is the unit outward normal for the surface at \mathbf{r} ; also, note that κ , the absorption coefficient, is now made a function of \mathbf{r} . We equate the fluxes in equations (52) and (53) and solve for the absorption coefficient,

$$\kappa(\mathbf{r}) = D \sum_j \frac{q_j (\mathbf{r} - \mathbf{r}_j) \cdot \tilde{\mathbf{n}}(\mathbf{r})}{|\mathbf{r} - \mathbf{r}_j|^3} \bigg/ \sum_j \frac{q_j}{|\mathbf{r} - \mathbf{r}_j|}. \quad (54)$$

Thus, if a surface absorbs molecules with this absorption coefficient, the concentration profile will be the same as if the system were unbounded.

Figure 7 illustrates this algorithm. Using the Smoldyn program, I defined two molecular sources and bounded the system with a cube that was 5 μm on each side. Each cube face was tiled with 25 panels, each 1 μm square. Each of these panels absorbed molecules according to the coefficient given in equation (54), calculated with \mathbf{r} set to the panel center. The upper portion of figure 7 shows a slice through the middle of the system and the lower portion compares average steady-state molecular concentrations in a band through the system with model concentrations from equation (51). The simulated and model systems agree well.

7. Discussion and conclusions

Erban and Chapman’s adsorption algorithm for particle-based simulators [18], based on equation (1), is simple to understand, easy to implement and fast to simulate. These qualities make it ideal for ‘quick-and-dirty’ simulations. However, it is only quantitatively accurate if (i) the modeled system is kept well mixed, even in the immediate vicinity of adsorbing surfaces, or (ii) simulation time steps or adsorption coefficients are very small (more precisely, $\kappa' \ll (2\pi)^{-1/2}$). We built on their result here with algorithms that yield more accurate adsorbed

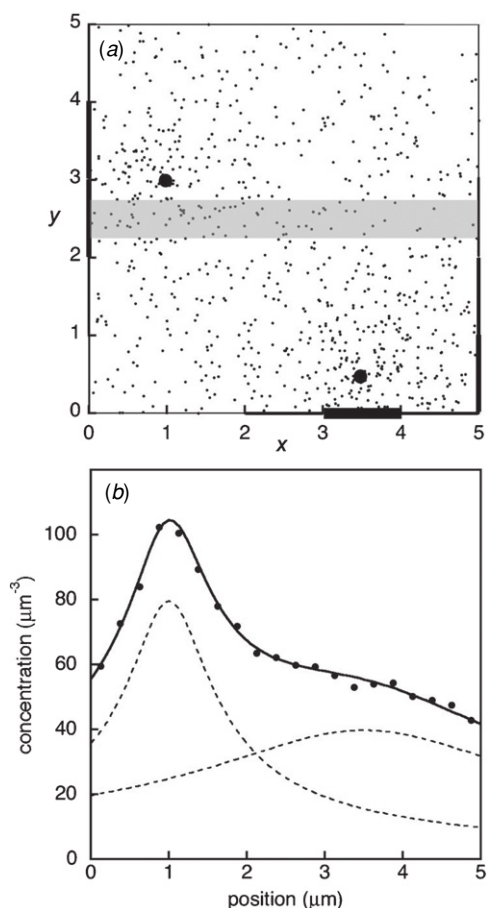


Figure 7. Simulation of effective unbounded diffusion with a partially absorbing boundary. (A) a $0.5 \mu\text{m}$ thick slice, taken parallel to the x,y -plane, through the middle of a cubical-simulated system. Large dots represent molecular emitters, small dots represent molecules and the gray band across the system ($5 \mu\text{m}$ long by $0.5 \mu\text{m}$ wide and thick) represents the volume in which molecules were counted. Each face of the system is assembled from a grid of 25 panels, each with an absorption coefficient from equation (54); panels with larger absorption coefficients are depicted with thicker boundary lines. (B) Dots represent simulated molecule concentrations, measured along the gray band from panel A, at steady state and averaged over 1000 time steps. Dashed lines represent theoretical concentrations from each individual source, along the gray band, for the unbounded model system. The solid line represents the sum of the dashed lines, which is the total theoretical concentration for the unbounded model system. See supplementary information for details available at stacks.iop.org/PhysBio/6/046015.

concentrations and solution-phase concentration distributions, and also with new algorithms for desorption and partial transmission. We derived these algorithms for steady-state systems, but also showed that they are very accurate for systems that start well mixed. They require essentially the same computational effort as Erban and Chapman's method at each simulation time step but may enable faster simulations overall because they maintain their accuracy as time steps are made longer.

This work shares two important themes with our prior work on simulating bimolecular reactions [3]. First, we designed both sets of algorithms to yield correct rates when

the systems are at steady state rather than when they are well mixed. We did this because the adsorption and reaction processes we wished to model cause systems to evolve rapidly away from well-mixed states (or other initial states) and toward steady states. As a result, portions of these systems close to adsorbing surfaces or reactive molecules, which are the regions that determine adsorption and reaction rates, spend nearly all of their time close to steady state. As we saw above and in prior work [3], steady-state investigations do not lead to the simple analytical equations that other researchers found using well-mixed assumptions [18, 22, 41], but are nevertheless necessary for accurate simulations. Secondly, we found that reversible interactions need to be considered as special cases, rather than as independent forward and reverse processes. We saw that here for reversible adsorption and reversible transmission, and previously for reversible bimolecular chemical reactions [1, 3]. Treating the processes as independent forward and reverse processes leads to incorrect transition rates and, more importantly, incorrect equilibrium constants.

The algorithms presented here will, hopefully, make it easier for researchers to quantify biological adsorption, desorption and transmission rates by using computational modeling. This will help address the current paucity of quantitative data for these interactions. Combining these algorithms with the better data should also enable particle-based biological simulations that are faster and more accurate than current ones. These investigations may help scientists better understand dynamic protein localization, intracellular molecular gradients and related topics. To promote these research directions, all algorithms presented here are implemented in a code library that is supplied with the supplementary information (available at stacks.iop.org/PhysBio/6/046015), and in the general-purpose Smoldyn biochemical simulator (available at <http://www.smoldyn.org>).

8. Supplementary information

The supplementary information for this paper (available at stacks.iop.org/PhysBio/6/046015) includes (i) details about the simulation emulator, (ii) tables that relate adsorption coefficients to adsorption probabilities for irreversible and reversible adsorption, (iii) details about stochastic simulations used in figures 5(a), 6 and 7, and (iv) C source code that converts between molecule-surface interaction coefficients and their simulation probabilities, and also includes the simulation emulator, and (v) documentation for the source code.

Acknowledgments

This publication was made possible by grant number P50 HG002370 from the NHGRI, and also by MITRE Corp. contract number 77459; R Brent was the principle investigator for both grants. Its contents are solely the responsibility of the author and do not necessarily represent the official views of the NIH. I thank Attila Szabo for help with the appendix results; Radek Erban and Jon Chapman for helpful discussions;

Nathan Addy, Esther Mecking and Orna Resnekov for editing assistance, and anonymous reviewers for exceptionally helpful reviews.

Appendix A.

This section solves the following problem. In the adsorption/desorption model system, defined by equations (3) and (4), suppose that the solution is well mixed with concentration C_∞ and there are no adsorbed molecules at time 0. What is the adsorbed concentration at time t ?

We Laplace transform equations (3) and (4) to yield

$$-C(x, 0) + z\hat{C}(x, z) = D \frac{\partial^2 \hat{C}(x, z)}{\partial x^2} \quad (\text{A.1})$$

$$\begin{aligned} -C_a(0) + z\hat{C}_a(z) &= D \left. \frac{\partial \hat{C}(x, z)}{\partial x} \right|_{x=0} \\ &= \kappa \hat{C}(0, z) - k\hat{C}_a(z). \end{aligned} \quad (\text{A.2})$$

A ‘hat’ symbol denotes a Laplace transformed function and z is the Laplace conjugate of the time (we use z instead of s , which is conventional, because we use s for the rms step length). In addition, three boundary conditions are

$$C(x, 0) = C_\infty \quad (\text{A.3})$$

$$C_a(0) = 0 \quad (\text{A.4})$$

$$\hat{C}(\infty, z) = \frac{C_\infty}{z}. \quad (\text{A.5})$$

Substitution of equation (A.3) into equation (A.1), followed by solution of the differential equation, yields the general solution

$$\hat{C}(x, z) = \frac{C_\infty}{z} + ae^{x\sqrt{\frac{z}{D}}} + be^{-x\sqrt{\frac{z}{D}}}, \quad (\text{A.6})$$

where a and b are constants of integration. Using equation (A.5), a equals 0. Meanwhile, the first and last portions of equation (A.2), with substitution from equation (A.4), yield

$$\hat{C}_a(z) = \frac{\kappa}{z+k} \hat{C}(0, z). \quad (\text{A.7})$$

Combination with equation (A.6) simplifies the result to

$$\hat{C}_a(z) = \frac{\kappa}{z+k} \left(\frac{C_\infty}{z} + b \right). \quad (\text{A.8})$$

Using the middle and last portions of equation (A.2), along with substitutions from equations (A.6) and (A.8), yields

$$-Db\sqrt{\frac{z}{D}} = \kappa \left(\frac{C_\infty}{z} + b \right) - k \frac{\kappa}{z+k} \left(\frac{C_\infty}{z} + b \right). \quad (\text{A.9})$$

We solve this for b to find

$$b = -\frac{C_\infty \kappa}{(z+k)\sqrt{Dz} + \kappa z}. \quad (\text{A.10})$$

This result is substituted into equations (A.6) and (A.8) to yield the Laplace transformed solution for the surface concentration,

$$\hat{C}_a(z) = \frac{C_\infty D \kappa}{z [D(z+k) + \kappa \sqrt{Dz}]}. \quad (\text{A.11})$$

Using the method of partial fractions, equation (A.11) inverse Laplace transforms to yield the surface concentration as a function of time

$$C_a(t) = \frac{C_\infty \kappa t \left[-c_1 + c_2 - c_2 e^{c_1^2} \operatorname{erfc}(c_1) + c_1 e^{c_2^2} \operatorname{erfc}(c_2) \right]}{c_1 c_2 (c_2 - c_1)} \quad (\text{A.12})$$

$$c_1 = \frac{\kappa - \sqrt{\kappa^2 - 4Dk}}{2\sqrt{D}} \sqrt{t} \quad (\text{A.13})$$

$$c_2 = \frac{\kappa + \sqrt{\kappa^2 - 4Dk}}{2\sqrt{D}} \sqrt{t}, \quad (\text{A.14})$$

where c_1 and c_2 are variables that the partial fraction expansion introduced, but which do not have a clear physical interpretation. This solution applies for all physically reasonable (i.e. non-negative) values of the parameters κ , k , D and t . However, it may not be particularly convenient for many parameter choices because the constants c_1 and c_2 are complex for many reasonable values of κ and k . Note that the solution-phase concentration profile can be found from results presented in section 12.4.III.iii of [12].

Appendix B.

This section is very similar to appendix A. In the partial transmission model system, defined by equations (3) and (5), suppose that the concentration is 0 for all $x < 0$ and is C_∞ for all $x > 0$ at time 0. How much substance is in the $x < 0$ portion of the system at time t ?

We Laplace transform equations (3) and (5) to yield

$$-C_B(x, 0) + z\hat{C}_B(x, z) = D \frac{\partial^2 \hat{C}_B(x, z)}{\partial x^2} \quad (\text{B.1})$$

$$-C_F(x, 0) + z\hat{C}_F(x, z) = D \frac{\partial^2 \hat{C}_F(x, z)}{\partial x^2} \quad (\text{B.2})$$

$$\begin{aligned} D \left. \frac{\partial \hat{C}_B(x, z)}{\partial x} \right|_{x=-0} &= D \left. \frac{\partial \hat{C}_F(x, z)}{\partial x} \right|_{x=+0} \\ &= \kappa_F \hat{C}_F(0, z) - \kappa_B \hat{C}_B(0, z). \end{aligned} \quad (\text{B.3})$$

$C_B(x, t)$ and $C_F(x, t)$, along with their Laplace transformed versions, are the concentration profiles on the back and front sides of the surface, respectively. Some additional boundary conditions are

$$C_B(x, 0) = 0 \quad (\text{B.4})$$

$$C_F(x, 0) = C_\infty \quad (\text{B.5})$$

$$\hat{C}_B(-\infty, z) = 0 \quad (\text{B.6})$$

$$\hat{C}_F(\infty, z) = \frac{C_\infty}{z}. \quad (\text{B.7})$$

The first two of these boundary conditions simplify the differential equations, equations (B.1) and (B.2). They are

then solved (see equation (A.6)) and simplified further with the latter two boundary conditions to yield

$$\hat{C}_B(x, z) = ae^x \sqrt{\frac{x}{D}} \quad (\text{B.8})$$

$$\hat{C}_F(x, z) = \frac{C_\infty}{z} + be^{-x} \sqrt{\frac{x}{D}}. \quad (\text{B.9})$$

In these equations, a and b are constants of integration. These general solutions are combined with the Robin boundary condition in equation (B.3) and simplified to yield solutions for a and b ,

$$a = -b = \frac{C_\infty \kappa_F}{z(\kappa_B + \kappa_F + \sqrt{Dz})}. \quad (\text{B.10})$$

Substitution of equation (B.10) into equations (B.8) and (B.9) yields the Laplace transforms of the concentration profiles. The inverse Laplace transform, which would yield the time-dependent concentration profiles, appears to be intractable. Instead, we work toward the desired solution by integrating the concentration profile on the backside of the surface to find the total number of transmitted molecules. Its Laplace transform is

$$\hat{M}(z) = \int_{-\infty}^0 \hat{C}_B(x, z) dx = \frac{C_\infty \kappa_F \sqrt{D}}{z^{3/2}(\kappa_B + \kappa_F + \sqrt{Dz})}. \quad (\text{B.11})$$

This inverse Laplace transforms to yield the desired solution

$$M(t) = \frac{C_\infty \kappa_F \sqrt{Dt}}{(\kappa_B + \kappa_F)^2 \sqrt{\pi}} \left\{ 2(\kappa_B + \kappa_F) \right. \\ \left. - \sqrt{\frac{\pi D}{t}} + \sqrt{\frac{\pi D}{t}} e^{\frac{(\kappa_B + \kappa_F)^2 t}{D}} \operatorname{erfc} \left[\frac{(\kappa_B + \kappa_F) \sqrt{t}}{\sqrt{D}} \right] \right\}. \quad (\text{B.12})$$

This solution applies to all physically reasonable values of the parameters ($\kappa_F > 0$, $\kappa_B > 0$, $t > 0$ and $D > 0$).

Glossary

Adsorption coefficient. The proportionality constant in the Robin boundary condition that expresses the rate at which solution-phase molecules that are adjacent to a surface adsorb to the surface. It is defined here by κ in equation (4) and has units of length/time. Other authors frequently define the term differently, depending on their adsorption model.

Adsorption probability. The probability with which molecules that diffused across a surface during the prior simulation time step adsorb to the surface.

Concentration profile. The expectation molecular concentration as a function of the distance away from the surface. It is a distribution function.

Desorption rate constant. The probability with which a molecule desorbs from a surface during a simulation time step.

Exact algorithm. An algorithm for which all of its results are theoretically indistinguishable from those of the underlying model system.

Particle-based simulation. A simulation method in which individual molecules of interest are represented with point-like particles in continuous space. This method accounts for stochastic and spatial detail.

Permeability coefficient. The proportionality constant in the Robin boundary condition that expresses the rate at which solution-phase molecules that are adjacent to a surface diffuse through the surface. It is defined here by κ_F and κ_B in equation (5) and has units of length/time.

Robin boundary condition. A weighted combination of Dirichlet and Neumann boundary conditions in which the gradient of a function at a surface is directly proportional to its value at the surface. This is also called the radiation boundary condition.

Root mean square (rms) step length. The average length of a step for a molecule in a Brownian dynamics simulation. This is often a characteristic length for the simulation algorithms.

Steady-state system. A system in which expectation molecular concentrations are constant with respect to time, although there may be constant molecular fluxes.

Well-mixed system. A system in which the expectation solution-phase molecular concentration is uniform throughout space.

References

- [1] Andrews S S 2005 Serial rebinding of ligands to clustered receptors as exemplified by bacterial chemotaxis *Phys. Biol.* **2** 111–22
- [2] Andrews S S, Addy N J, Brent R and Arkin A P 2009 Detailed simulation of cell biology with Smoldyn 2.1 *PLoS Comput. Biol.* provisionally accepted
- [3] Andrews S S and Bray D 2004 Stochastic simulation of chemical reactions with spatial resolution and single molecule detail *Phys. Biol.* **1** 137–51
- [4] Andrews S S, Dinh T and Arkin A P 2009 Stochastic modeling of biochemical reaction networks *Encyclopedia of Complexity and System Science* ed R A Meyers (Heidelberg: Springer)
- [5] Atkins P W 1986 *Physical Chemistry* 3rd edn (New York: Freeman)
- [6] Bardwell L 2005 A walk-through of the yeast mating pheromone response pathway *Peptides* **26** 339–50
- [7] Bartol T M J, Land B R, Salpeter E E and Salpeter M M 1991 Monte Carlo simulation of miniature endplate current generation in the vertebrate neuromuscular junction *Biophys. J.* **59** 1290–307
- [8] Bemporad D, Luttmann C and Essex J W 2004 Computer simulation of small molecule permeation across a lipid bilayer: dependence on bilayer properties and solute volume, size, and cross-sectional area *Biophys. J.* **87** 1–13
- [9] Ben-Tal N, Honig B, Bagdassarian C K and Ben-Shaul A 2000 Association entropy in adsorption processes *Biophys. J.* **79** 1180–7
- [10] Benz R and McLaughlin S 1983 The molecular mechanism of action of the proton ionophore FCCP (carbonylcyanide *p*-trifluoromethoxyphenylhydrazone) *Biophys. J.* **41** 381–98

- [11] Berg H C 1993 *Random Walks in Biology* 2nd edn (Princeton, NJ: Princeton University Press)
- [12] Carslaw H S and Jaeger J C 1959 *Conduction of Heat in Solids* 2nd edn (Oxford: Clarendon)
- [13] Chou T and D'Orsogna M R 2007 Multistage adsorption of diffusing macromolecules and viruses *J. Chem. Phys.* **127** 105101
- [14] Coggan J S *et al* 2005 Evidence for ectopic neurotransmission at a neuronal synapse *Science* **309** 446–51
- [15] Colman-Lerner A 2008 Personal communication
- [16] Crank J 1975 *The Mathematics of Diffusion* 2nd edn (Oxford: Oxford University Press)
- [17] Elowitz M B, Surette M G, Wolf P-E, Stock J B and Leibler S 1999 Protein mobility in the cytoplasm of *Escherichia coli* *J. Bacteriol.* **181** 197–203
- [18] Erban R and Chapman S J 2007 Reactive boundary conditions for stochastic simulations of reaction-diffusion processes *Phys. Biol.* **4** 16–28
- [19] Gustafson K and Abe T 1998 The third boundary condition—was it Robin's *Math. Intelligencer* **20** 63–71
- [20] Hlady V, Buijs J and Jennissen H P 1999 Methods for studying protein adsorption *Methods Enzymol.* **309** 402–29
- [21] Hoops S *et al* 2006 COPASI—a COmplex PATHway SIMulator *Bioinformatics* **22** 3067–74
- [22] Kerr R A *et al* 2008 Fast Monte Carlo simulation methods for biological reaction-diffusion systems in solution and on surfaces *SIAM J. Sci. Comput.* **30** 3126–49
- [23] Kerr R A, Levine H, Sejnowski T J and Rappel W-J 2006 Division accuracy in a stochastic model of Min oscillations in *Escherichia coli* *Proc. Natl Acad. Sci. USA* **103** 347–52
- [24] Kofahl B and Klipp E 2004 Modelling the dynamics of the yeast pheromone pathway *Yeast* **21** 831–50
- [25] Koh X, Srinivasan B, Ching H S and Levchenko A 2006 A 3D Monte Carlo analysis of the role of dyadic space geometry in spark generation *Biophys. J.* **90** 1999–2014
- [26] Kubo R 1965 *Statistical Mechanics. An Advanced Course with Problems and Solutions* (Amsterdam: North-Holland)
- [27] Langmuir I 1916 The constitution and fundamental properties of solids and liquids: I. Solids *J. Am. Chem. Soc.* **38** 2221–95
- [28] Levchenko A, Bruck J and Sternberg P W 2000 Scaffold proteins may biphasically affect the levels of mitogen-activated protein kinase signaling and reduce its threshold properties *Proc. Natl Acad. Sci. USA* **97** 5818–23
- [29] Lipkow K 2006 Changing cellular location of CheZ predicted by molecular simulations *PLoS Comp. Biol.* **2** e39
- [30] Lipkow K, Andrews S S and Bray D 2005 Simulated diffusion of CheYp through the cytoplasm of *E. coli* *J. Bacteriol.* **187** 45–53
- [31] Lipkow K and Odde D J 2008 Model for protein concentration gradients in the cytoplasm *Cell. Mol. Bioeng.* **1** 84–92
- [32] Loew L M and Schaff J C 2001 The virtual cell: a software environment for computational cell biology *Trends Biotech.* **19** 401–6
- [33] Paula S, Volkov A G, van Hoek A N, Haines T H and Deamer D W 1996 Permeation of protons, potassium ions, and small polar molecules through phospholipid bilayers as a function of membrane thickness *Biophys. J.* **70** 339–48
- [34] Plimpton S and Slepoy A 2003 ChemCell: a particle-based model of protein chemistry and diffusion in microbial cells *Technical Report SAND2003-4509* (Albuquerque, NM: Sandia National Laboratory)
- [35] Plimpton S J and Slepoy A 2005 Microbial cell modeling via reacting diffusive particles *J. Phys.: Conf. Ser.* **16** 305–9
- [36] Press W H, Flannery B P, Teukolsky S A and Vetterling W T 1988 *Numerical Recipes in C. The Art of Scientific Computing* (Cambridge: Cambridge University Press)
- [37] Pryciak P M and Huntress F A 1998 Membrane recruitment of the kinase cascade scaffold protein Ste5 by the G-beta-gamma complex underlies activation of the yeast pheromone response pathway *Genes Dev.* **12** 2684–97
- [38] Rao C V, Wolf D M and Arkin A P 2002 Control, exploitation, and tolerance of intracellular noise *Nature* **420** 231–7
- [39] Sanford C, Yip M L K, White C and Parkinson J 2006 Cell++—simulating biochemical pathways *Bioinformatics* **22** 2918–25
- [40] Schneider P and Smith J M 1968 Adsorption rate constants from chromatography *AIChE J.* **14** 762–71
- [41] Singer A, Schuss Z, Osipov A and Holcman D 2008 Partially reflected diffusion *SIAM J. Appl. Math.* **68** 844–68
- [42] Stiles J R and Bartol T M 2001 Monte Carlo methods for simulating realistic synaptic microphysiology using MCell *Computational Neuroscience: Realistic Modeling for Experimentalists* ed E De Schutter (Boca Raton, FL: CRC Press) pp 87–130
- [43] Sundararaj S *et al* 2004 The CyberCell Database (CCDB): a comprehensive, self-updating, relational database to coordinate and facilitate *in silico* modeling of *Escherichia coli* *Nucleic Acids Res.* **32** D293–5
- [44] Wolfram Research Inc. 2009 *Mathematica 7.0* (Champaign, IL: Wolfram Research Inc.)