Computational Tool

SpringSaLaD: A Spatial, Particle-Based Biochemical Simulation Platform with Excluded Volume

Paul J. Michalski[1,*] and Leslie M. Loew[1]
[1]Richard D. Berlin Center for Cell Analysis and Modeling, University of Connecticut Health Center, Farmington, Connecticut

ABSTRACT We introduce Springs, Sites, and Langevin Dynamics (SpringSaLaD), a comprehensive software platform for spatial, stochastic, particle-based modeling of biochemical systems. SpringSaLaD models biomolecules in a coarse-grained manner as a group of linked spherical sites with excluded volume. This mesoscopic approach bridges the gap between highly detailed molecular dynamics simulations and the various methods used to study network kinetics and diffusion at the cellular level. SpringSaLaD is a standalone tool that supports model building, simulation, visualization, and data analysis, all through a user-friendly graphical user interface that should make it more accessible than tools built into more comprehensive molecular dynamics infrastructures. Importantly, for bimolecular reactions we derive an exact expression relating the macroscopic on-rate to the various microscopic parameters with the inclusion of excluded volume; this makes SpringSaLaD more accurate than other tools, which rely on approximate relationships between these parameters.

INTRODUCTION

Computational modeling has become indispensable for elucidating the properties of complex biochemical networks. In a traditional modeling approach the modeler defines the various chemical species of interest and the reaction kinetics between them, which defines a closed network of reactions that can be simulated in a number of ways. If the copy number of the individual species is low, the kinetics should be simulated by solving the chemical master equation for the stochastic population probabilities. If the number of molecules is large enough and the system is well mixed, then deterministic simulations can be performed by numerically solving ordinary differential equations (ODEs). Copasi (copasi.org) is a popular example of a simulator for ODEs and stochastic network models (1). Spatial dynamics within an explicit geometry can be modeled by adding diffusion terms to the reaction equations and solving the resulting partial differential equations. MCell (mcell.org (2)) and Smoldyn (smoldyn.org (3)) are popular software platforms for modeling discrete Brownian motion and stochastic reactions. Virtual Cell (vcell.org) is a comprehensive software environment for modeling and simulating reaction networks and membrane transport either stochastically or deterministically, with the option of accounting for diffusion in realistic cellular geometries (4,5). But there are two problems with these commonly used tools for computational systems biology: they cannot readily handle models with combinatorial complexity, and, for spatial models, they do not accurately model the excluded volume and spatial extent of interacting molecules or clusters.

A combinatorially complex system makes it difficult or impossible for the modeler to specify a reaction network without additional computational tools. For example, a receptor with 10 phosphorylation sites can exist in $2^{10} = 1028$ states, with an even larger number of reactions needed to describe the transitions between these states. In general, the size of the reaction network grows exponentially with the number of potential protein modifications or binding sites, and this phenomenon is termed "combinatorial complexity". Tools such as BioNetGen (6,7) automatically generate reaction networks from a small number of defined rules, but the exponential increase in network size ultimately limits the practical simulation of such large networks. In a recent study (8), we estimated that it would take a 2.54 GHz Intel Xeon processor 290 years to generate the reaction network for simple model of the synaptic kinase CaMKII. Even if network generation completed in a reasonable amount of time, we would not have the computational resources required to simulate such a large system of differential equations or store the results of such calculations.

Some particle-based methods circumvent the problems of combinatorial complexity by avoiding the process of enumerating all species and reactions (9–12). Instead, these methods (also known as agent-based or network-free simulators) use reaction rules to probabilistically spawn the states of individual molecules during the simulation, and therefore only need to consider the potential reactions for species (i.e., states) that are actually present in the system. These methods are implemented in several openly accessible software tools (11,13,14).

Submitted September 11, 2015, and accepted for publication December 23, 2015.

*Correspondence: paul.michalski@jax.org

Editor: Stanislav Shvartsman.

0006-3495/16/02/0523/7

http://dx.doi.org/10.1016/j.bpj.2015.12.026

A major limitation of most spatial simulators is that they treat all molecules, even large multicomponent complexes, as point particles. Various approaches exist to model the effects of excluded volume, such as adding density-dependent terms in reaction-diffusion equations (15,16) or by defining a reaction that moves nearby particles apart (as implemented in Smoldyn), but in general such methods are unable to model important properties that depend on the spatial extent and composition of the complex, such as the effects of spatial orientation and the reduced diffusion of larger complexes. One biologically important example of such a system is ligand-induced membrane receptor clustering, as illustrated in Fig. 1, which shows a system of trivalent receptors clustered by interacting with bivalent ligands. Such interactions lead to the formation of molecular clusters that increase local concentration of biomolecules, potentially triggering signaling events. We have included such complexes in a class of biophysical structures called pleomorphic ensembles (PEs) (12,17,18), because these complexes are often plastic, with dynamic and variable composition. Accurate modeling of PEs requires tools that account for their extended spatial structure and excluded volume, as these properties underlie essential features of PEs, such as reduced access to open binding sites on the interior of the cluster or rebinding of newly dissociated monomers. Furthermore, because the sizes and compositions of pleomorphic ensembles are open-ended (i.e., sampling an infinite number of states), a particle-based algorithm is required. For these reasons, pleomorphic ensembles cannot be modeled by computational tools such as MCELL, Smoldyn, or Virtual Cell.

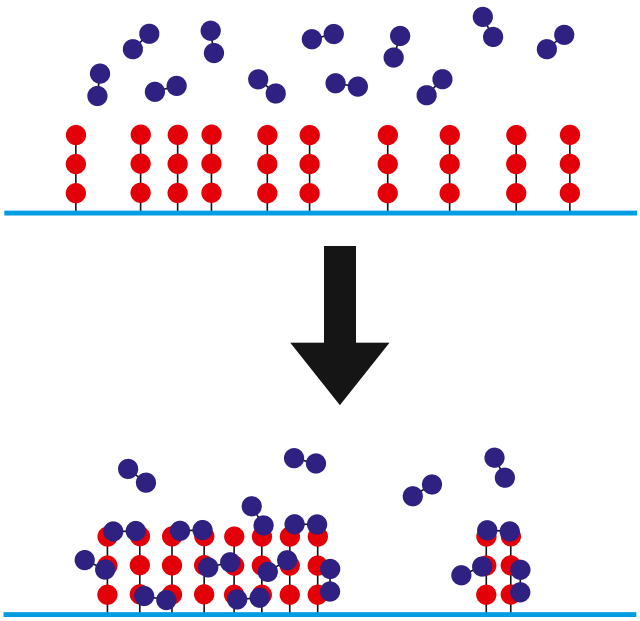

FIGURE 1 Receptor clustering example. Ligand-induced clustering of membrane receptors is one example of a system with combinatorial complexity where spatial organization and excluded volume can strongly modify system dynamics and equilibrium organization. Bivalent ligands are shown in blue, trivalent receptors are shown in red, and the cyan line represents the membrane. To see this figure in color, go online.

In principle, PEs could be modeled with molecular dynamics (MD) simulations, but the large sizes of these systems, which consist of tens to hundreds of proteins, make such simulations computationally impractical. These systems are best modeled using a mesoscopic, coarse-grained approach, where individual proteins are modeled as a single site or a collection of linked sites. In recent years a number of programs have become available for such mesoscopic modeling, most notably SRSim (19) and ReaDDyMM (20,21). However, these and other programs of this class require a level of computational expertise that makes them inaccessible to the typical biologist. Moreover, all available programs implement bimolecular reactions by relating macroscopic on-rates to the microscopic parameters using formulae that are only approximate in the presence of excluded volume, or which were derived assuming well-mixed conditions that are invalid for the typical systems studied.

Here we introduce a standalone modeling and simulation package, SpringSaLaD, which implements spatial, particle-based models with excluded volume and accurate biochemical reactions, including the treatment of allostery. Notably, we derive and implement an exact formula to relate bimolecular macroscopic on-rates to the various microscopic parameters. The platform offers a simple but powerful user interface and allows for the simulation of receptor clustering and other multistate pleomorphic ensembles. It will run in a reasonable time on any modern personal computer (hours to a few days, depending on the number of molecules simulated). The software is written in Java and is freely available as a standalone .jar file for Windows, Mac, or Linux at www.ccam.uchc.edu/resources/ccam_software.html#SpringSaLaD. The simulation code is bundled as an independent .jar file, and may be used on the command line to, for example, run hundreds of simultaneous simulations on a Linux cluster. A User's Guide and Tutorial are included in the downloadable zip file, and are also available in the Supporting Material. Source code is freely available on Github at pjmichalski/SpringSaLaD (GUI components) and pjmichalski/LangevinNoVis01 (simulation components).

MATERIALS AND METHODS

Please see the Supporting Material.

RESULTS

Molecule description

Springs, Sites, and Langevin Dynamics (SpringSaLaD) describes the molecules in the system as a set of distinct spherical sites connected by links, which are modeled with stiff springs. The sites may represent different domains within

the macromolecule, and may be selected as binding sites for potential reactions with other binding sites. Many biological molecules of interest can be described in such a manner, and two of these are shown in Fig. 2: the hemoglobin tetramer and a GFP-tagged α-tubulin. Models that seek to describe finer details, such as the motion of individual amino acids, are more appropriately simulated with MD, while models with less detail will run faster with simulators that assume molecules to be infinitesimal points.

We will use a model of protein kinase A (PKA) to illustrate the three steps involved in molecule construction. These steps and the GUI for molecule construction are shown in Fig. 3. Firstly, we define the types of sites in the molecule. For example, PKA contains two types of sites: regulatory and catalytic domains. Each type can have associated with it an arbitrary number of internal states with their own associated set of biochemical reactions (reactions are described below). For example, the regulatory domain could have three states describing the number of bound cAMP molecules (0, 1, or 2), and the catalytic domain could have an inactive and active state reflecting its kinase activity. Each type also has an associated physical size, diffusion constant, and color (for visualization purposes).

Secondly, sites are added to the molecule, and each site is assigned one of the previously defined types. To construct PKA we would add four sites, assigning two of them to be regulatory domains and two to be catalytic domains. Physically, sites are modeled as impenetrable spheres to accurately capture excluded volume effects.

Thirdly, links are added to connect the sites to each other. Each site can be linked to an arbitrary number of other sites in either two or three dimensions. The only requirement is that sites cannot overlap. Links are stiff and thus define an intersite distance, but are free to rotate around the sites. For example, a triangular molecule with three sites will not maintain its geometry in the simulation with only two links, but will if a third bond is added to enforce a distance between the two outer sites. Links are only used to control the distance between sites and do not occupy physical space, and both sites and other links are free to pass through a link. Excluded volume is only enforced by sites.

To specify membrane-bound molecules, a special anchor designation is assigned to a site, which restricts it to two-dimensional diffusion within the membrane; the anchor site can then be linked to sites that diffuse in the adjacent volume, effectively restricting those sites to be part of a membrane-bound molecule.

SpringSaLaD provides several convenience methods for constructing large linear polymers, and other convenience methods for constructing large arrays of linked receptors. Molecules must be defined as intracellular, extracellular, or membrane-bound. These and other details can be found in the User's Guide and Tutorial, which are available in the Supporting Material.

Geometry

SpringSaLaD currently only supports a rectangular geometry with reflecting boundary conditions. The rectangular geometry is partitioned into an extracellular space, a planar membrane, and an intracellular space. The size of the membrane and the depths of the intra- and extracellular spaces are user-defined. Future versions of SpringSaLaD will support a wider variety of geometries.

Particle motion: diffusion and constraints due to binding

Particle motion is influenced by two classes of forces: random forces that lead to diffusional motion, and interparticle forces that impose the constraints from intra- and intermolecular bonds. These forces are incorporated in the overdamped Langevin equation (22),

$$\zeta \vec{v} = \vec{F}_{\text{rand}} + \vec{F}_{\text{bonds}}, \tag{1}$$

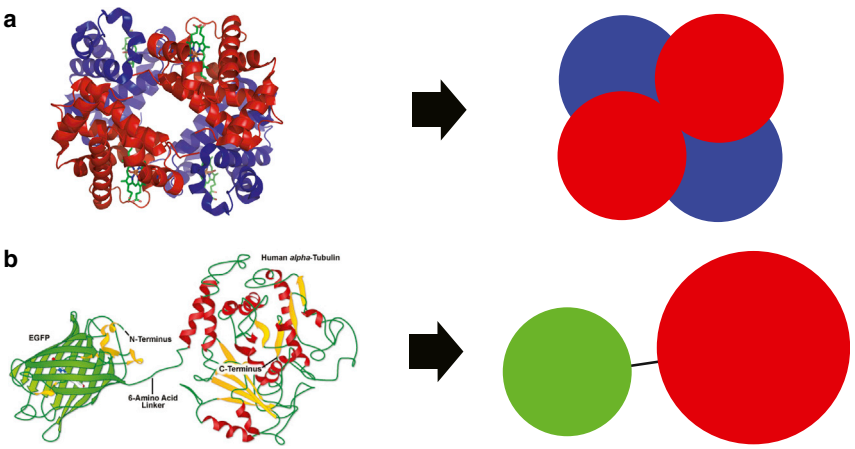

FIGURE 2 Examples of coarse-grained models. Two examples of proteins that can be readily decomposed into nanometer-sized domains appropriate for coarse-grained modeling with SpringSaLaD. (*a*) The hemoglobin tetramer can be coarse-grained into four independent biochemical sites. (*b*) Any globular GFP-tagged protein can be represented as two connected sites. Here we show GFP-tagged α-tubulin. The images on the left are modified from the original images available at (*a*) http://zeiss-campus.magnet.fsu.edu/articles/probes/fpintroduction.html, and (*b*) http://upload.wikimedia.org/wikipedia/commons/3/3d/1GZX_Haemoglobin.png. To see this figure in color, go online.

FIGURE 3 PKA model. (*Top*) A description of PKA in terms of SpringSaLaD model components. PKA is composed of two regulatory sites and two catalytic sites. The state of a regulatory site is defined by the number of bound cAMP molecules, while the catalytic subunit can be in either an inactive or an active state. (*Bottom*) The SpringSaLaD molecule editor GUI, as used to construct the PKA molecule described on the top. To see this figure in color, go online.

where ζ is the coefficient of viscous friction, $\vec{v}$ is the particle velocity, and $\vec{F}_{\text{rand}}$ and $\vec{F}_{\text{bonds}}$ represent the random forces and the total force of bonds. The random forces are guaranteed to recapitulate the desired diffusion, provided they are chosen from normal distribution with variance

$$\left\langle \vec{F}_{\text{rand}}(t) \cdot \vec{F}_{\text{rand}}(t') \right\rangle = 2 n_d D \zeta^2 \, \delta(t - t'), \tag{2}$$

where D is the desired diffusion coefficient, n_d is the dimension of the system (here $n_d = 3$), and the delta-function simply states that the random force is uncorrelated in time. The bonds are modeled as stiff springs,

$$\vec{F}_{\text{bonds}} = \sum_i k_i \left(r_i - r_{0,i}\right) \hat{r}_i, \tag{3}$$

where the sum runs over all bonds, k_i is a spring constant, $\hat{r}_i$ is the unit vector pointing from the particle to the neighbor with which it shares a bond, r_i is the current interparticle distance, and $r_{0,i}$ is the equilibrium bond distance. The exact value of the spring constant is not important, provided the

spring is stiff enough to keep sites near the expected distances but not so stiff that accurate motion requires unreasonably small time steps. In practice, only the ratio k/ζ is required, and in SpringSaLaD this ratio is the same for all bonds and links, namely, $k/\zeta = 10^8\ \mathrm{s}^{-1}$; this allows us to use time steps of ~1–100 ns for biologically relevant diffusion constants.

Reactions

Zeroth-order

SpringSaLaD supports particle creation reactions, which may be used to buffer the concentration of a species. Given a macroscopic creation rate, k_{create}, with units μM/s, a single molecule is added to the system at each time step with probability $k_{\mathrm{create}}Vdt$, where V is either the intra- or extracellular volume depending on the location of the molecule. A molecule is added by testing random positions in the system until a position is found that does not overlap with other particles. Note that this mechanism will fail in extremely dense systems. In such a system a simple, constant-rate zeroth order reaction is not consistent with physical constraints, and is not an appropriate modeling construct.

First-order

SpringSaLaD supports three general types of first-order reactions, all of which are described by a reaction rate, r, with units s^{-1}, and which occur at each time step with probability rdt. These are:

1) Bond dissociation reactions: when a dissociation reaction occurs, the bond is simply removed from the system.
2) Internal state transitions: these describe the transitions between the internal states of each site, as defined by the type of that site. The probability of these transitions can depend on the identities of binding partners or the states of other sites in the same molecule. The former dependency would be used, for example, to prevent a transition from an unphosphorylated to a phosphorylated state unless the site is bound to a kinase. The latter dependency can be used to model allosteric interactions.
3) Decay reactions: these are used to remove molecules from the system.

Second-order

SpringSaLaD supports binding reactions between two sites. A bond is modeled by the creation of a new link between the reacting sites. The link is modeled identically to the links that hold molecules together, except it has an associated off-rate controlling molecular dissociation. Particle-based simulations often use the Smoluchowski approach (23) to model bimolecular reactions, but such an approach is incompatible with excluded volume. Instead, we modified the approach described in Erban and Chapman (24) to account for excluded volume. Each site is associated with two radii: the physical radius, ρ_i ($i = 1,2$), which is defined in molecule construction and enforces excluded volume, and a slightly larger reaction radius, R_i. Two reactive sites undergo a binding reaction with probability $\lambda\, dt$ per time step when their reaction radii overlap. The reaction rate λ is related to the macroscopic on-rate, k_{on}, with units of $\mu\mathrm{M}^{-1}\ \mathrm{s}^{-1}$, through

$$k_{\mathrm{on}} = \frac{4\pi RD\phi}{1 - \frac{R}{a}\phi}, \tag{4}$$

where

$$\phi = 1 - \left(\frac{r_0}{R}\right)\frac{r_0 \sinh\left(\frac{R-\rho}{r_0}\right) + \rho \cosh\left(\frac{R-\rho}{r_0}\right)}{\rho \sinh\left(\frac{R-\rho}{r_0}\right) + r_0 \cosh\left(\frac{R-\rho}{r_0}\right)} \tag{5}$$

and $r_0 = \sqrt{D/\lambda}$, $D = D_1 + D_2$, $\rho = \rho_1 + \rho_2$, $R = R_1 + R_2$, and a is the dissociation radius.

The derivation of this equation and additional discussion of bimolecular reactions can be found in the Supporting Material.

Data analysis

SpringSaLaD comes packaged with an interactive three-dimensional viewer to visualize simulation results. The viewer is implemented in Java3D and provides methods for saving images and generating movies in a variety of formats. (Currently, the viewer will only work on Windows and Linux machines. The viewer is disabled on Macs because of an incompatibility between Java3D and the latest versions of Java on Macs.) Two example videos (Movies S1 and S2) that show clustering in the Nck-nephrin-Nwasp system are available in the Supporting Material. The program keeps track of a variety of observables, such as the number and types of bound particles or the number of sites in a particular state, and automatically displays this data in a convenient tabular format. Stochastic simulations require many runs to compute average properties. SpringSaLaD provides the option to launch runs in sequence or in parallel, depending on the user's computational resources, and automatically computes averages and standard deviations of the observables in the data tables. Instructions for the viewer and data analysis are provided in the User's Guide and Tutorial.

Algorithm implementation verification

We constructed simple models of analytically tractable systems to verify the accuracy and implementation of all simulation algorithms. These included various tests of diffusion and all implemented reactions. As an example, we modeled

a system consisting of a single species undergoing a creation and decay reaction. In this case SpringSaLaD reproduces the full population distribution predicted by the time-dependent solution to the chemical master equation. In the Supporting Material, we provide a discussion of all the tests, the models used, and the simulation results.

DISCUSSION

The software presented here fills an important gap in the spectrum of publicly available biochemical simulation platforms, allowing the simulation of systems that exhibit combinatorial complexity, are too large for MD simulations, and are influenced by spatial effects. Many important biological systems fall into this category, such as chromatin dynamics in the confines of a crowded nucleus, signaling in the dense and diverse postsynaptic density of dendritic spines, signaling in the narrow foot processes of kidney podocytes, and the large variety of structures, such as receptor clusters, cell adhesion complexes, or mRNA granules, which we have described as PE (17). The primary conceptual requirement is that molecules must be described in a coarse-grained manner as a linked collection of biochemically distinct sites, with radii of a few nanometers. Modeling at a finer scale requires MD simulations, while coarser models can be simulated more efficiently with other methods. The description in terms of linked sites automatically captures emergent properties of a dynamic system, such as the reduced diffusion coefficient of larger clusters because of the disordered individual motions of its components.

There are three significant but unavoidable drawbacks to our approach. Firstly, there is the increased computational cost associated with tracking the hundreds of individual sites. The simulation runs time scales linearly in the number of particles (for a fixed concentration), which puts practical limits on the number of molecules in the system. Secondly, the simulation must use time steps of 10 ns to accurately enforce excluded volume and prevent overextending the springs holding sites together. This is orders-of-magnitude smaller than the 100 μs time steps typical in Smoldyn and the 1 ms to 1 s time steps common in ODE or partial differential equation simulations. On the other hand, it is many orders-of-magnitude larger than the femtosecond time steps used in MD simulations. We find that simulations following up to 1000 particles over the course of 1–10 s will run in 1–5 days. We have several improvements planned, such as a port to C++ and parallelization of individual simulations, which will allow simulation of larger systems.

Thirdly, there is the practical difficulty in relating some microscopic parameters required in the model to macroscopic parameters measured in the lab. For example, SpringSaLaD requires the user to define a diffusion constant for each site in a molecule, whereas the experimentally accessible parameter is the diffusion coefficient of the molecule as a whole. In some cases, such as linear polymers or spherical globules, the two parameters can be related, but for an arbitrary geometry there is no exact formula to relate the macroscopic and microscopic diffusion coefficients, and the modeler may have to try several microscopic values to find one that is appropriate. A similar difficulty arises when defining the on-rate of a bimolecular reaction. If the reactive site is buried in a relatively inaccessible pocket of a large molecule, then the true microscopic on-rate may be many orders-of-magnitude larger than the macroscopic value. Again, no exact relation exists for an arbitrary geometry, and a parameter scan may be required to find a suitable value. These limitations are not unique to SpringSaLaD, but are faced in any coarse-grained modeling approach.

To our knowledge, the one truly new feature described here is the Smoluchowski-like algorithm used to model bimolecular interactions with excluded volume. Although Eq. 4 is derived with standard methods, we are not aware of its previous publication, and it is not used in any other biochemical simulation platform. SpringSaLaD packages these methods together with standard treatments of first-order reactions, allosteric state transitions, and Langevin dynamics within a convenient, user-friendly interface, which will make these methods available to a broader community of biochemical modelers. In the next year we plan to integrate SpringSaLaD with the Virtual Cell modeling and simulation platform (4,5,25,26), where it will be offered as an additional tool to complement the wide variety of modeling methods supported by VCell.

SUPPORTING MATERIAL

Supporting Materials and Methods, Supporting Results, twelve figures, one table, SpringSaLaD User's Guide and Tutorial, and two movies are available at http://www.biophysj.org/biophysj/supplemental/S0006-3495(15)04810-9.

AUTHOR CONTRIBUTIONS

P.J.M. and L.M.L. conceived the project; P.J.M. wrote the software code and validated the algorithms; L.M.L. tested the software; and L.M.L. and P.J.M. wrote the article.

ACKNOWLEDGMENTS

We thank Marc Rigatti for extensively testing the development version of SpringSaLaD, and Ahmed Elmokadem for helpful comments on the user's manual.

This work was supported by National Institutes of Health grant No. P41 GM103313.

REFERENCES

1. Mendes, P., S. Hoops, …, U. Kummer. 2009. Computational modeling of biochemical networks using COPASI. *Methods Mol. Biol.* 500:17–59.
2. Stiles, J. R., T. M. Bartol, Jr., …, M. M. Salpeter. 1998. Monte Carlo simulation of neurotransmitter release using MCell, a general simulator

of cellular physiological processes. *In* Computational Neuroscience. J. Bower, editor. Plenum Press, New York, pp. 279–284.

3. Robinson, M., S. S. Andrews, and R. Erban. 2015. Multiscale reaction-diffusion simulations with Smoldyn. *Bioinformatics.* 31:2406–2408.
4. Schaff, J., C. C. Fink, …, L. M. Loew. 1997. A general computational framework for modeling cellular structure and function. *Biophys. J.* 73:1135–1146.
5. Cowan, A. E., I. I. Moraru, …, L. M. Loew. 2012. Spatial modeling of cell signaling networks. *Methods Cell Biol.* 110:195–221.
6. Blinov, M. L., J. R. Faeder, …, W. S. Hlavacek. 2004. BioNetGen: software for rule-based modeling of signal transduction based on the interactions of molecular domains. *Bioinformatics.* 20:3289–3291.
7. Faeder, J. R., M. L. Blinov, and W. S. Hlavacek. 2009. Rule-based modeling of biochemical systems with BioNetGen. *Methods Mol. Biol.* 500:113–167.
8. Michalski, P. J., and L. M. Loew. 2012. CaMKII activation and dynamics are independent of the holoenzyme structure: an infinite subunit holoenzyme approximation. *Phys. Biol.* 9:036010.
9. Stefan, M. I., T. M. Bartol, …, M. B. Kennedy. 2014. Multi-state modeling of biomolecules. *PLOS Comput. Biol.* 10:e1003844.
10. Schöneberg, J., A. Ullrich, and F. Noé. 2014. Simulation tools for particle-based reaction-diffusion dynamics in continuous space. *BMC Biophys.* 7:11.
11. Sneddon, M. W., J. R. Faeder, and T. Emonet. 2011. Efficient modeling, simulation and coarse-graining of biological complexity with NFsim. *Nat. Methods.* 8:177–183.
12. Falkenberg, C. V., M. L. Blinov, and L. M. Loew. 2013. Pleomorphic ensembles: formation of large clusters composed of weakly interacting multivalent molecules. *Biophys. J.* 105:2451–2460.
13. Sorokina, O., A. Sorokin, …, V. Danos. 2013. A simulator for spatially extended κ-models. *Bioinformatics.* 29:3105–3106.
14. Colvin, J., M. I. Monine, …, R. G. Posner. 2010. RuleMonkey: software for stochastic simulation of rule-based models. *BMC Bioinformatics.* 11:404.
15. Painter, K. J., and T. Hillen. 2002. Volume-filling and quorum-sensing in models for chemosensitive movement. *Can. Appl. Math. Q.* 10: 501–543.
16. Hillen, T., and K. J. Painter. 2009. A user's guide to PDE models for chemotaxis. *J. Math. Biol.* 58:183–217.
17. Mayer, B. J., M. L. Blinov, and L. M. Loew. 2009. Molecular machines or pleiomorphic ensembles: signaling complexes revisited. *J. Biol.* 8:81.
18. Suderman, R., and E. J. Deeds. 2013. Machines vs. ensembles: effective MAPK signaling through heterogeneous sets of protein complexes. *PLOS Comput. Biol.* 9:e1003278.
19. Gruenert, G., B. Ibrahim, …, P. Dittrich. 2010. Rule-based spatial modeling with diffusing, geometrically constrained molecules. *BMC Bioinformatics.* 11:307.
20. Schöneberg, J., and F. Noé. 2013. ReaDDy—a software for particle-based reaction-diffusion dynamics in crowded cellular environments. *PLoS One.* 8:e74261.
21. Biedermann, J., A. Ullrich, …, F. Noé. 2015. ReaDDyMM: fast interacting particle reaction-diffusion simulations using graphical processing units. *Biophys. J.* 108:457–461.
22. Snook, I. 2006. The Langevin and Generalised Langevin Approach to the Dynamics of Atomic, Polymeric and Colloidal Systems. Elsevier, Dordrecht, The Netherlands.
23. Andrews, S. S., and D. Bray. 2004. Stochastic simulation of chemical reactions with spatial resolution and single molecule detail. *Phys. Biol.* 1:137–151.
24. Erban, R., and S. J. Chapman. 2009. Stochastic modelling of reaction-diffusion processes: algorithms for bimolecular reactions. *Phys. Biol.* 6:046001.
25. Moraru, I. I., J. C. Schaff, …, L. M. Loew. 2008. Virtual Cell modelling and simulation software environment. *IET Syst. Biol.* 2:352–362.
26. Slepchenko, B. M., J. C. Schaff, …, L. M. Loew. 2003. Quantitative cell biology with the Virtual Cell. *Trends Cell Biol.* 13:570–576.

SpringSaLaD: a spatial, particle-based biochemical simulation platform with excluded volume

P. J. Michalski and L. M. Loew

SUPPORTING MATERIAL

SUPPLEMENTARY SECTION I: Validation Simulations

Here we present the results of simulations used to validate various components of SpringSaLaD.

Diffusion

The Langevin dynamics algorithms underlying SpringSaLaD are guaranteed to reproduce diffusive motion with the given diffusion constant. We verified this with a model consisting of a single molecule species, A, composed of one inert site, with diameter 1 nm and diffusion constant $D = 1\ \mu m^2/s$. We calculated the mean-squared-displacement as a function of time and compared it to the expected short- and long-time behavior.

At short times we expect $msd(t) \approx 2nDt$, where n is the dimensionality of the system. To verify this behavior, we simulated 2000 A molecules diffusing in a cube of side length 2001 nm, which is effectively 2000 nm after accounting for the radius of *A*. The occupied volume fraction, ϕ, is

$$\phi = \frac{2000\ V_{particle}}{V_{system}} = 1.31 \text{ x } 10^{-7},$$

which allows us to ignore excluded volume effects and treat this as a system of non-interacting particles. Short time behavior is expected for $t \ll \frac{L^2}{\pi^2 D} = 0.4$ s, and we simulated diffusion for 10 μs, well-within the short-time regime. In Fig. 1 we plot the msd as a function of time, and use a linear fit to determine the "measured" diffusion constant, which gives $D_{\text{measured}} = 1.0008\ \mu m^2/s$, which is within 0.1% of the expected value.

At long times we expect to observe confined diffusion. In 1D, the long-time behavior is given by

$$msd(t) = \frac{L^2}{6} - \frac{16L^2}{\pi^4} exp\left(-\frac{D\pi^2 t}{L^2}\right), \quad (1)$$

where L is the length of the system. To verify this behavior, we simulated 200 *A* molecules diffusing in a cube of side length 101 nm for 6 ms. The occupied volume fraction is $\phi = 1.0 \text{ x } 10^{-4}$, so we can treat this as a system of non-interacting particles. Figure 2 shows a plot of the msd along the x-coordinate as a function of time, along with a best fit to equation (1). The fit gives $D_{\text{measured}} = 0.984\ \mu m^2/s$, which is within 2% of the expected value.

SpringSaLaD can also be used to investigate the effects of crowding on diffusional motion. For 2- and 3-dimensional motion crowding simply renormalizes the diffusion constant, but in 1D crowding also changes the scaling exponent such that at high densities $msd(t) \sim t^{1/2}$. We investigated if SpringSaLaD could reproduce this behavior by using a model consisting of 200 membrane-embedded molecules with a diameter of 4 nm and $D = 1\ \mu m^2/s$. The system has a width of 4.5 nm, which effectively restricts motion to one dimension. The

length was varied from 80000 nm to 1000 nm to study system densities from 0.01, where we expect free diffusion, to 0.8, where we hoped to see crowding behavior. The system was simulated for $20\ \mu s$.

Figure 3 shows a plot of the msd as a function of time for each of the simulated systems. As the system density increases the msd is reduced, and at large densities the relationship no longer appears to be linear. We quantified the effects of crowding by fitting $msd(t) = at^b$ for each curve, and Fig. 4 shows a plot of the exponent, b, as a function of system density. At low densities $b \approx 1$, as expected, and b decreases as the density increases. At a density of 0.7 we found $b = 0.54$, and at a density of 0.8 we found $b = 0.46$, both of which are in good agreement with the theoretical result.

Creation/Decay Reactions

SpringSaLaD supports zeroeth-order particle creation reactions defined by a rate constant k_c , with units $\mu M\ s^{-1}$, and particle decay reactions defined by a rate constant k_d, with units s^{-1}. These reactions were verified using a system with the same composition as that used to verify diffusion (single site molecule A, diameter 1 nm, $D = 1\ \mu m^2/s$). If we let $p_m(t)$ be the probability of finding m Amolecules at time t, then the chemical master equation is

$$\dot{p}_m = k_d[(m+1)p_{m+1} - mp_m] + r_c[p_{m-1} - p_m], \tag{2}$$

where $r_c$has units of s^{-1} and is given by $r_c = 602\ V\ k_c$, where V is the volume of the system in μm^3, and the factor of 602 comes from Avogadro's constant and unit conversion factors. The macroscopic quantity of interest is the average number of particles at time t,

$$\langle m(t) \rangle = \sum_{m=0}^{\infty} m p_m(t)\ . \tag{3}$$

Equation (2) can be solved exactly for certain limiting cases.

1) If $k_d = 0$ then the equation describes a Poisson process. Assuming the system begins with no particles, $p_m(t) = \delta_{m,0}$, then the complete solution is

$$p_m(t) = \frac{(r_c t)^m}{m!}\ e^{-r_c t}\ , \tag{4}$$

$$\langle m(t) \rangle = r_c t\ . \tag{5}$$

We ran 500 simulations with $k_c = 0.5\ \mu M\ s^{-1}$ in a cube of side length 2000 nm, giving $r_c = 2408\ s^{-1}$. Simulations were run for 10 ms. Figure 5(a) shows the average number of particles as a function of time compared to the prediction of Eq. (5), and the results are seen to be in excellent agreement. As a more stringent test, Fig. 5(b) shows the full distribution at 10 ms compared to the predicted distribution of Eq. (4). The distributions are statistically equivalent.

2) If $r_c = 0$ then the equation describes a simple decay process. Assuming the system begins with M particles at time t=0, then the complete solution is

$$p_m(t) = \binom{M}{m} \; e^{-m\,k_d t}(1 - e^{-k_d t})^{M-m}\,, \tag{6}$$

$$\langle m(t)\rangle = M e^{-k_d t}\,. \tag{7}$$

We ran 500 simulations with $k_d = 100\ s^{-1}$ in a cube of side length 2000 nm. Simulations were run for 100 ms. Figure 6(a) shows the average number of particles as a function of time compared to the prediction of Eq. (6), and the results are seen to be in excellent agreement. As a more stringent test, Fig. 6(b) shows the full distribution at 10 ms compared to the predicted distribution of Eq. (7). The distributions are statistically equivalent.

3) If both creation and decay processes are allowed, then we can solve for the stationary distribution and the time-dependent average. Assuming the system begins with no particles, we obtain

$$p_m = \frac{\lambda^m}{m!}\, e^{-\lambda}\,, \tag{8}$$

$$\langle m(t)\rangle = \lambda(1 - e^{-k_d t}), \tag{9}$$

where $\lambda = r_c/k_d$. We ran 500 simulations with $k_c = 1\ \mu M\ s^{-1}$ ($r_c = 4816\ s^{-1}$) and $k_d = 250\ s^{-1}$, giving $\lambda = 19.264$, in a cube of side length 2000 nm. Simulations were run for 50 ms, and equilibrium was achieved at approximately 15 ms. Figure 7(a) shows the average number of particles as a function of time compared to the prediction of Eq. (9), and the results are seen to be in excellent agreement. As a more stringent test, Fig. 7(b) shows the full equilibrium distribution (averaged over the distributions at 20, 30, 40, and 50 ms) compared to the expected distribution from Eq. (8). The distributions are statistically equivalent.

Bimolecular Reactions

SpringSaLaD supports bimolecular association/dissociation reactions which are defined by an on rate, k_{on}, with units $\mu M^{-1}s^{-1}$, and an off rate, k_{off}, with units s^{-1}.

Single Species Combination

Consider a molecular species, A, which can dimerize to form a complex, *C*. Mass conservation implies $A_{\text{tot}} = A(t) + 2C(t)$, which shows that the system is completely characterized by the concentration of *A*. Let m denote the total number of free *A* molecules at time *t*, and assume there are M total molecules of A in the system. If we let n be the total number of C molecules, then $n = \frac{1}{2}(M - m)$. The chemical master equation is

$$\dot{p}_m = \frac{1}{2}k_{\text{off}}[(M-m+2)p_{m-2} - (M-m)p_m] + \frac{(k_{\text{on}}/N_A)}{V}[(m+2)(m+1)p_{m+2} - m(m-1)p_m], \tag{10}$$

where N_A is Avogadro's constant. It is straightforward to show that the average number of particles obeys

$$\frac{d\langle m\rangle}{dt} = k_{\text{off}}(M - \langle m\rangle) - \frac{2k_{\text{on}}}{VN_A}[\langle m^2\rangle - \langle m\rangle]\,. \tag{11}$$

We try to recover the deterministic limit by letting $A_{\text{tot}} = \frac{M}{VN_A}$, $= \frac{\langle m\rangle}{VN_A}$, and assuming zero variance, $\langle m^2\rangle = \langle m\rangle^2$, and find

$$\frac{dA}{dt} = k_{\text{off}}(A_{\text{tot}} - A) - 2k_{\text{on}}A^2 + 2k_{\text{on}}\frac{A}{VN_A}, \tag{12}$$

which only agrees with the classical result when $V \to \infty$.

The steady-state solutions to the master equation are

$$p_m = \frac{\left(\frac{M-1}{2}\right)!}{\left(\frac{M-m}{2}\right)!\,m!}\,\frac{(2r)^{(m-1)/2}}{{}_1F_1\left(-\frac{M-1}{2},\frac{3}{2},-\frac{r}{2}\right)} \quad \text{for } M \text{ and } m \text{ odd}, \tag{13}$$

$$p_m = \frac{\left(\frac{M}{2}\right)!}{\left(\frac{M-m}{2}\right)!\,m!}\,\frac{(2r)^{m/2}}{{}_1F_1\left(-\frac{M}{2},\frac{1}{2},-\frac{r}{2}\right)} \quad \text{for } M \text{ and } m \text{ even}, \tag{14}$$

where $r = \frac{VN_Ak_{\text{off}}}{2k_{\text{on}}}$ is a dimensionless parameter and ${}_1F_1(a,b,c)$ is the Kummer confluent hypergeometric function.

The complete time dependent solution can be obtained in the deterministic limit. If we define $a(t) = 2A(t)/K_D$, where $K_D = k_{\text{off}}/k_{\text{on}}$ is the equilibrium dissociation constant, then we find

$$a(t) = -\frac{1}{2} + p\,\frac{(a_0+1/2)\cosh(pk_{\text{off}}t) + p\sinh(pk_{\text{off}}t)}{(a_0+1/2)\sinh(pk_{\text{off}}t) + p\cosh(pk_{\text{off}}t)}, \tag{15}$$

where $p = \sqrt{a_0 + 1/4}$.

We modeled *A* as a molecule with a single site of radius 1 nm and $D = 2\ \mu m^2/s$, which could undergo a dimerization reaction with $k_{\text{on}} = 25\ \mu M^{-1}s^{-1}$ and $k_{\text{off}} = 50\ s^{-1}$. The simulation volume was a cube with side length 500 nm ($V = 0.125\ \mu m^3$), giving $r = 75.25$. The stochastic result only differs from the classical deterministic result when $r \ll 1$, and thus we expect the average concentrations to follow the deterministic results. The simulations consisted of 150 initially free A molecules ($A = 1.99\ \mu M$), and simulations were run for 400 ms. Figure 8(a) shows the average number of particles as a function of time compared to the prediction of Eq. (15), and Fig. 8(b) shows the full equilibrium distribution (averaged over the distributions at 100, 150, 200, 250, 300, 350, and 400 ms) compared to the expected distribution from Eq. (14). The observations match the expected values, except the average number of particles is larger than expected at short time scales. This is a consequence of the fact that the on-rate can be calculated to match the initial time

course or the equilibrium values, but not both, and we choose to match the on-rate to the equilibrium value. A further discussion of this issue can be found in Supplementary Section II.

Two Species Combination

We now consider the situation where two species, $A$and B, dimerize to form a complex, C. Mass conservation implies $A_{tot} = A(t) + C(t)$ and $B_{tot} = B(t) + C(t)$. Letting $A(0) = A_0, B(0) = B_0,$ and $C(0) = C_0$, then we must have $A(t) - A_0 = B(t) - B_0 = C_0 - C(t)$, and we see that only one of these species can be taken as an independent variable. To simplify the analysis we assume $C_0 = 0$, giving $A_0 = A_{tot}\ and\ B_0 = B_{tot}$. Let $m$be the number of A molecules, and let m_0 and n_0 be the initial numbers of A and B, respectively. Then the chemical master equation is

$$\begin{aligned}\dot{p}_m = {} & k_{\text{off}}\big[(m_0 - (m-1))p_{m-1} - (m_0 - m)p_m\big] \\ & + \frac{(k_{\text{on}}/N_A)}{V}[(m+1)(m+1-m_0+n_0)p_{m+1} - m(m-m_0+n_0)p_m],\end{aligned} \tag{16}$$

and the average number of particles obeys

$$\frac{d\langle m\rangle}{dt} = k_{\text{off}}[m_0 - \langle m\rangle] - \frac{k_{\text{on}}}{N_A V}\left[\langle m^2\rangle - (m_0 - n_0)\langle m\rangle\right]. \tag{17}$$

The same relations used to go from Eq. (11) to Eq. (12) now give

$$\frac{dA}{dt} = k_{\text{off}}[A_0 - A] - k_{\text{on}}[A^2 - (A_0 - B_0)A]. \tag{18}$$

The steady state solutions to the master equation are

$$p_m = \binom{m_0}{m}\frac{(n_0 - m_0)!}{(n_0 - m_0 + m)!}\frac{r^m}{{}_1F_1(-m_0, n_0 - m_0 + 1, -r)} \quad \text{for } m_0 < n_0\ (0 \le m \le m_0), \tag{19}$$

$$p_m = \binom{n_0}{m - (m_0 - n_0)}\frac{(m_0 - n_0)!\ r^{m-(m_0-n_0)}}{m!\ {}_1F_1(-n_0, m_0 - n_0 + 1, -r)} \quad \text{for } m_0 > n_0\ (m_0 - n \le m \le m_0), \tag{20}$$

where $r = VN_Ak_{\text{off}}/k_{\text{on}}$.

The complete time dependent solution can be obtained in the deterministic limit. Define $a(t) = A(t)/K_D$, then

$$a(t) = \frac{a_0 - b_0}{2} - \frac{1}{2} + s\frac{(s+u)e^{2sk_{\text{off}}t} - (s-u)}{(s+u)e^{2sk_{\text{off}}t} + (s-u)}, \tag{21}$$

where $u = (a_0 + b_0 + 1)/2$ and $s^2 = \frac{(a_0-b_0)^2}{4} + \frac{1}{4} + \frac{a_0+b_0}{2}$.

We modeled both A and B as molecules with a single site of radius 1 nm and $D = 1\ \mu m^2/s$, which could undergo a dimerization reaction with $k_{\text{on}} = 25\ \mu M^{-1}s^{-1}$ and $k_{\text{off}} = 50\ s^{-1}$. The simulation volume was a cube with side length 500 nm ($V = 0.125\ \mu m^3$), giving $r = 150.5$. The stochastic result only differs from the classical deterministic result when $r \ll 1$, and thus we expect the average concentrations to follow the deterministic results. The simulations consisted of 75 initially free A molecules ($A = 0.997\ \mu M$) and 150 initially free B molecules ($B = 1.99\ \mu M$), and simulations were run for 400 ms. Figure 9(a) shows the average

number of particles as a function of time compared to the prediction of Eq. (21), and Fig. 9(b) shows the full equilibrium distribution (averaged over the distributions at 100, 150, 200, 250, 300, 350, and 400 ms) compared to the expected distribution from Eq. (19). The observations match the expected values, except the average number of particles is larger than expected at short time scales, as discussed above.

Transition Reactions

SpringSaLaD supports transition reactions between states for each site, characterized by a single rate constant r with units s^{-1}. A transition reaction could be used to model the phosphorylation of a site (conversion between an "unphosphorylated" and "phosphorylated" state), the activation of an enzyme (conversion between "off" and "on" state), or many other types of reactions. These reactions may be restricted such that they only occur under certain conditions, and SpringSaLaD currently supports four such conditions.

1) No condition. Transition can always occur.
2) Free. Transition can only occur if the site is not bound by another site. This is useful if binding tends to lock a molecule in a particular state and thus prevent the transition.
3) Bound. Transition can only occur if the site is bound by specific site. A phosphorylation reaction would use this condition to ensure phosphorylation only occurs when the site is bound by the kinase.
4) Allosteric. Transition can only occur if a specific site *in the same molecule* is in a particular state.

Allosteric Activation

We test our implementation of conditions (1) and (4) using a simple model of allosteric activation of an enzyme. Consider an enzyme with two sites: an allosteric site, A, which can be either "off" or "on", and an enzymatic site, E, which can also be either "off" or "on." We assume the allosteric site can spontaneously turn on, and the enzymatic site can only turn on once the allosteric site is activated. We assume all molecules begin as $A_{\text{off}}E_{\text{off}}$, and assume there are no back reactions, which keeps this model simple and analytically tractable. The full reaction diagram is thus

$$A_{\text{off}}E_{\text{off}} \xrightarrow{r_o} A_{\text{on}}E_{\text{off}} \xrightarrow{r_1} A_{\text{on}}E_{\text{on}},$$

and we will assume $r_0 \neq r_1$to simplify the discussion.

Let m be the number of $A_{\text{off}}E_{\text{off}}$, n be the number of $A_{\text{on}}E_{\text{off}}$, and $k = M - m - n$ be the number of $A_{\text{on}}E_{\text{on}}$ (only m and n are independent variables), where M is the total number of molecules in the system. The chemical master equation for the joint probability distribution is

$$\dot{p}_{m,n} = r_0\big[(m+1)p_{m+1,n-1} - mp_{m,n}\big] + r_1\big[(n+1)p_{m,n+1} - np_{m,n}\big], \tag{22}$$

with solution

$$p_{m,n}(t) = \frac{M!}{m!\,n!\,(M-m-n)!}[f(t)]^m[g(t)]^n[h(t)]^{M-m-n}, \tag{23}$$

where

$$f(t) = e^{-r_0 t}, \tag{24}$$

$$g(t) = \frac{r_0}{r_1 - r_0}(e^{-r_0 t} - e^{-r_1 t}), \tag{25}$$

$$h(t) = \frac{r_1}{r_1 - r_0}(1 - e^{-r_0 t}) - \frac{r_0}{r_1 - r_0}(1 - e^{-r_1 t}). \tag{26}$$

The individual probability distributions may be obtained by summing over the joint probability distribution, for example, $p_m(t) = \sum_{n=0}^{M-m} p_{mn}(t)$, gives the probability distribution for $A_{\text{off}}E_{\text{off}}$. We obtain

$$p_m(t) = \binom{M}{m}[f(t)]^m[1 - f(t)]^{M-m} \quad \text{for } A_{\text{off}}E_{\text{off}} \tag{27}$$

$$p_n(t) = \binom{M}{n}[g(t)]^n[1 - g(t)]^{M-n} \quad \text{for } A_{\text{on}}E_{\text{off}}, \tag{28}$$

$$p_k(t) = \binom{M}{k}[h(t)]^k[1 - h(t)]^{M-k} \quad \text{for } A_{\text{on}}E_{\text{on}}, \tag{29}$$

with average values

$$\langle m \rangle = f(t) \quad \text{for } A_{\text{off}}E_{\text{off}} \tag{30}$$

$$\langle n \rangle = g(t) \quad \text{for } A_{\text{on}}E_{\text{off}}, \tag{31}$$

$$\langle k \rangle = h(t) \quad \text{for } A_{\text{on}}E_{\text{on}}. \tag{32}$$

We simulated 200 molecules in a cubic container with side length 500 nm for a total of simulation time of 5 ms. The reaction rates were $r_0 = 500\ s^{-1}$ and $r_1 = 300\ s^{-1}$, and we ran 500 independent simulations. Figures 10(a) and 10(c) plot the time dependence of the average values of $A_{\text{off}}E_{\text{off}}$ and $A_{\text{on}}E_{\text{on}}$, respectively, and compares them to the predictions of Eqs. (30) and (32). The results are in excellent agreement with the expected values. Figures 10(b) and 10(d) show the full probability distributions at 1.5 ms and compare them to the predicted distributions from Eqs. (27) and (29). The distributions are statistically equivalent.

Enzymatic Activation

We test our implementation of conditions (2) and (3) with a simple model of ligand-mediated enzymatic activation. Consider a system with two species, a ligand which can be either inactive or active, and an enzyme which can be either off or on. Denote the ligand by L_k , for k="inact" or "act", and denote the enzyme by E_j , where j="off" or "on". Assume that the ligand can convert between its active and inactive states when it is not bound to the enzyme, and assume that the enzyme can turn on only when bound by the ligand. To simplify the system, assume that the ligand cannot dissociate from the active enzyme. Then the complete set of reactions is as follows:

1) Two reactions describing ligand activation and deactivation. We assume that the ligand can only switch states when it is free (unbound).

$$L_{inact} \xrightarrow{k\_L\_act} L_{act}$$

$$L_{act} \xrightarrow{k\_L\_inact} L_{inact}$$

2) Two reactions describing E activation and deactivation. We assume that E can only switch states while it is bound by ligand.

$$L_{act} \cdot E_{off} \xrightarrow{k\_E\_act} L_{act} \cdot E_{on}$$

$$L_{act} \cdot E_{on} \xrightarrow{k\_E\_inact} L_{act} \cdot E_{off}$$

3) Two reactions (association and dissociation) describing ligand binding and unbinding. We assume that binding is reversible when E is off, but dissociation is prevented when E is on.

$$L_{act} + E_{off} \xleftrightarrow{k_{on},\ k_{off}} L_{act} \cdot E_{off}$$

It is unlikely that analytic solutions exist for this system. Thus, we implemented this model in Virtual Cell as a non-spatial stochastic simulation. The VCell solvers have been well-validated and we take this result as the "expected" solution. The VCell model is publicly available as "Enzymatic-Activation" under username "pjmichal." The model parameters are listed in Table S1.

The simulation consisted of 60 ligand molecules initially in the inactive state, and 40 enzyme molecules initially in the off state. The system consisted of a cube with side length 200 nm, and the total simulation time was 0.2 seconds. Averages were taken over 500 independent simulations. Figures 11(a) and 11(c) plot the time dependence of L_{act} and E_{on}, respectively. The results are in excellent agreement with the VCell simulations. Figures 11(b) and 11(d) show the full equilibrium probability distributions averaged over the distributions at 40, 80, 120, 160, and 200 ms, along with the distribution calculated over 10,000 non-spatial stochastic VCell simulations. The distributions are statistically equivalent.

SUPPLEMENTARY SECTION II: Derivation of On-Rate Parameter

Reversible vs Irreversible Reactions: Discussion of the Smoluchowski Model

As we mentioned in Supplementary Section I, SpringSaLaD uses a relation between the macroscopic on-rate and the corresponding microscopic reaction rate which gives the expected equilibrium distribution but not the correct initial dynamics for a bimolecular reaction. Surprisingly, it is impossible to choose a microscopic rate which will recapitulate both the initial dynamics and equilibrium distributions. We will illustrate this result with the simple and well-known Smoluchowski model.

The Smoluchowski model assumes two spherical molecules, A and B, react immediately upon collision to form a complex. The aim of the model is to relate the reaction radii to the macroscopic on-rate. Let species A (B) have radius R_A (R_B) and diffusion constant D_A (D_B). We assume A and B are dilute, and choose a coordinate system centered on a single B molecule. Through straightforward rescaling we can consider an equivalent system where A is a point particle with diffusion constant $D = D_A + D_B$ and B is a sphere of radius $R = R_A + R_B$ fixed at the origin. If we let $c(r)$ be the concentration of A molecules, then $c(r)$ satisfies the diffusion equation, $\partial c/\partial t = D\nabla^2 c$. The surface of B is an absorbing boundary for A because the particles react immediately upon collision, and thus one boundary condition is $c(r = R) = 0$. The conventional Smoluchowski model assumes that the concentration of A goes to a constant at infinity, $c(r = \infty) = c_\infty$. With these boundary conditions the solution to the diffusion equation is

$$c(r) = c_\infty(1 - R/r).$$

The flux density of A at the surface of this single B molecule is $j_R = -D\left(\frac{\partial c}{\partial r}\right)_{r=R} = -c_\infty D/R$, and the total flux at a single B molecule is thus $J = \int d^2 r j_R = -4\pi DRc_\infty$. In the dilute limit each B molecule is assumed to be independent, and thus the total rate of destruction of A molecules is

$$\frac{dN_A}{dt} = JN_B = -4\pi DRN_B c_\infty.$$

We divide through by the volume of the system to convert from numbers to concentrations, and we associate $c_\infty = A$ to arrive at

$$\frac{dA}{dt} = -4\pi DR\ A\ B.$$

Comparison with the macroscopic rate equation, $\dot{A} = -k_{on}AB$ shows that the macroscopic on-rate is given by

$$k_{on} = 4\pi DR,$$

which is the classical Smoluchowski result.

Now consider the same system in equilibrium, where a dissociation reaction produces an A and B molecule from the complex. Assume, for simplicity, that upon dissociation the newly separated A and B molecules are exactly a distance $a > R$ apart. The important observation is that for $r > a$ the fluxes due to the forward and backward reactions must cancel in equilibrium, and thus $c(r) = c_0$ for $r \geq a$. This condition must replace the boundary condition at infinity which was used above, and the new solution to the diffusion equation is

$$c(r) = \frac{ac_0}{a-R}\left(1 - \frac{R}{r}\right).$$

Following the same steps as in the preceding paragraph to relate k_{on} and R , we find

$$k_{on} = \frac{4\pi DR}{1 - R/a}.$$

Thus, the reaction radius which reproduces the equilibrium (reversible) distribution is not the same as the reaction radius for the irreversible reaction considered in the preceding paragraph. In fact, we find

$$R_{rev} = \frac{R_{irr}}{1 + R_{irr}/a},$$

showing that the reversible reaction radius is always smaller than the irreversible reaction radius. If R_{rev} is used then the initial rate of reaction will be slower than expected from the macroscopic k_{on}, but the equilibrium distribution will be correct. Conversely, if R_{irr} is used then there will be more complex than expected in "equilibrium," but the transient dynamics will be correct.

The decision between the two radii must be made based on how the macroscopic on-rate was measured. A review of the literature suggests that the on-rate is usually inferred by directly measuring k_{off} and K_D, and then using $k_{on} = \frac{k_{off}}{K_D}$. In this case the measured on-rate is an equilibrium rate, and it is correct to use R_{rev}.

SpringSaLaD relates the macroscopic on-rate to a first order reaction rate, λ, instead of a reaction radius, but the same principle applies, namely, there is a different relationship between k_{on} and λ for irreversible and reversible reactions. SpringSaLaD uses the relationship which matches equilibrium, and this is the reason the transient dynamics show a lower reaction rate than might be expected, as shown in 8(a) and 9(a).

Derivation of SpringSaLaD bimolecular on-rate

SpringSaLaD enforces excluded volume, and to allow this feature each site must be characterized by two radii, as illustrated in Figure 12: 1) a physical radius, ρ, which represents the excluded volume, and 2) a reaction radius, $R > \rho$, which determines the maximum distance between two sites such that they can undergo a binding reaction. Specifically, if the reaction radii of two binding partners overlap, then those sites can undergo a binding reaction with probability $P = \lambda\, dt$, where λ is a first-order reaction rate. The goal here is to relate λ to the macroscopic on-rate, k_{on}.

We make the same transformations as in the Smoluchowski model, which allows us to consider A as a point particle with diffusion constant $D = D_A + D_B$, and B as fixed at the origin with physical radius $\rho = \rho_A + \rho_B$ and reaction radius $R = R_A + R_B$. The diffusion equation is

$$\frac{\partial c}{\partial t} = D\nabla^2 c - \lambda c \quad \text{for} \quad r \le R,$$

$$\frac{\partial c}{\partial t} = D\nabla^2 c \quad \text{for} \quad r > R,$$

with the following four boundary conditions:

1) A reflecting boundary at $r = \rho$ to account for excluded volume, $\left(\partial c/\partial r\right)_{r=\rho} = 0$.
2) The equilibrium condition forces no flux beyond the dissociation distance, and thus $c(r = a) = c_0$, and for simplicity we assume $a \geq R$. (It is also possible to solve the system with $\rho \leq a \leq R$, but SpringSaLaD uses $a = R$ and thus only the former solution is required.)
3) Continuity of concentration at $r = R$, $c(R_+) = c(R_-)$, where $R_\pm = R \pm \varepsilon$ as $\varepsilon \to 0$.
4) Continuity of flux at $r = R$, $\left(\partial c/\partial r\right)_{R_+} = \left(\partial c/\partial r\right)_{R_-}$.

The general solution to the diffusion equation is

$$c(r) = \begin{cases} \frac{b_0}{r}\sinh\left(\frac{r-\rho}{r_0}\right) + \frac{b_1}{r}\cosh\left(\frac{r-\rho}{r_0}\right) & \text{for } \rho \leq r \leq R \\ b_2 + \frac{b_3}{r} & \text{for } R < r \leq a \end{cases},$$

where $r_0^2 = D/\lambda$ and the b_i are constants to be fit by the boundary conditions.

Boundary conditions (1) and (2) can be used to eliminate b_1 and b_2, giving

$$c(r) = \begin{cases} \frac{b_0}{r}\left[\sinh\left(\frac{r-\rho}{r_0}\right) + \frac{\rho}{r_0}\cosh\left(\frac{r-\rho}{r_0}\right)\right] & \text{for } \rho \leq r \leq R \\ c_0 + \frac{b_3}{a}\left(1 - \frac{a}{r}\right) & \text{for } R < r \leq a \end{cases}.$$

Boundary condition (3) gives

$$\frac{b_0}{R}\left[\sinh\left(\frac{R-\rho}{r_0}\right) + \frac{\rho}{r_0}\cosh\left(\frac{R-\rho}{r_0}\right)\right] = c_0 + \frac{b_3}{a}\left(1 - \frac{a}{R}\right),$$

while boundary condition (4) gives

$$b_3 = -b_0\left\{\sinh\left(\frac{R-\rho}{r_0}\right) + \frac{\rho}{r_0}\cosh\left(\frac{R-\rho}{r_0}\right) - \frac{R}{r_0}\left[\cosh\left(\frac{R-\rho}{r_0}\right) + \frac{\rho}{r_0}\sinh\left(\frac{R-\rho}{r_0}\right)\right]\right\}.$$

These can be solved to eliminate b_0 and b_3, and we finally arrive at

$$c(r) = \begin{cases} c_0\frac{a}{rQ_1}\left[r_0\sinh\left(\frac{r-\rho}{r_0}\right) + \rho\cosh\left(\frac{r-\rho}{r_0}\right)\right] & \text{for } \rho \leq r \leq R \\ c_0\left[1 - \frac{Q_0}{Q_1}\left(1 - \frac{a}{r}\right)\right] & \text{for } R < r \leq a \end{cases},$$

where

$$Q_0 = r_0\sinh\left(\frac{R-\rho}{r_0}\right) + \rho\cosh\left(\frac{R-\rho}{r_0}\right) - \frac{R}{r_0}\left[\rho\sinh\left(\frac{R-\rho}{r_0}\right) + r_0\cosh\left(\frac{R-\rho}{r_0}\right)\right],$$

and

$$Q_1 = r_0\sinh\left(\frac{R-\rho}{r_0}\right) + \rho\cosh\left(\frac{R-\rho}{r_0}\right) - \left(\frac{R}{r_0}\right)\left(1 - \frac{a}{R}\right)\left[\rho\sinh\left(\frac{R-\rho}{r_0}\right) + r_0\cosh\left(\frac{R-\rho}{r_0}\right)\right].$$

We now proceed as in the Smoluchowski case: calculate the total flux at the surface of B, write an expression for the rate of change of the concentration of A, and compare that to the macroscopic kinetic equation. We find

$$k_{on} = \frac{4\pi RD\phi}{1 - \frac{R}{a}\phi},$$

where

$$\phi = 1 - \left(\frac{r_0}{R}\right)\frac{r_0 \sinh\left(\frac{R-\rho}{r_0}\right) + \rho \cosh\left(\frac{R-\rho}{r_0}\right)}{\rho \sinh\left(\frac{R-\rho}{r_0}\right) + r_0 \cosh\left(\frac{R-\rho}{r_0}\right)}.$$

We can check that this reduces to the Smoluchowski result and Erban-Chapman results in the appropriate limits. The Smoluchowski model is recovered by letting $\lambda \to \infty$, in which case the particles react immediately upon collision of their reaction radii. This corresponds to taking $r_0 \to 0$, whereupon $\varphi \to 1$ and we find

$$k_{on,Smol} = \lim_{\phi \to 1} k_{on} = \frac{4\pi RD}{1 - \frac{R}{a}},$$

which is indeed the Smoluchowski result derived above. The Erban-Chapman limit corresponds to letting $\rho \to 0$, which gives

$$\phi = 1 - \frac{r_0}{R}\tanh\left(\frac{R}{r_0}\right),$$

which is the expected result.

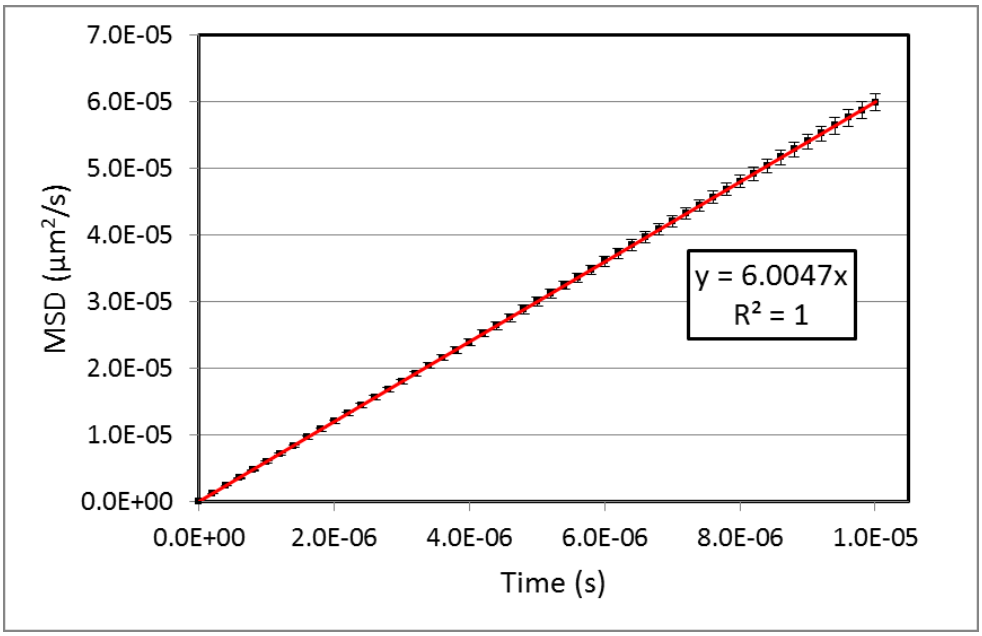

Figure 1: Validation of short-time diffusion. A plot of MSD vs time for the system described in the text. The plot shows the average and standard deviation from 100 simulations.

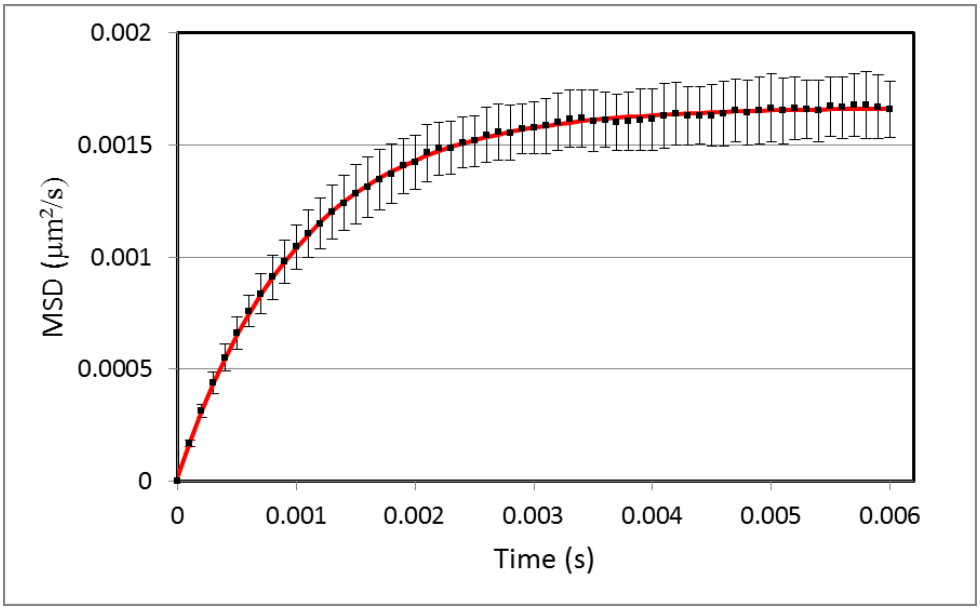

Figure 2: Validation of long-time diffusion. A plot of MSD vs time for the system described in the text. The plot shows the average and standard deviation from 100 simulations.

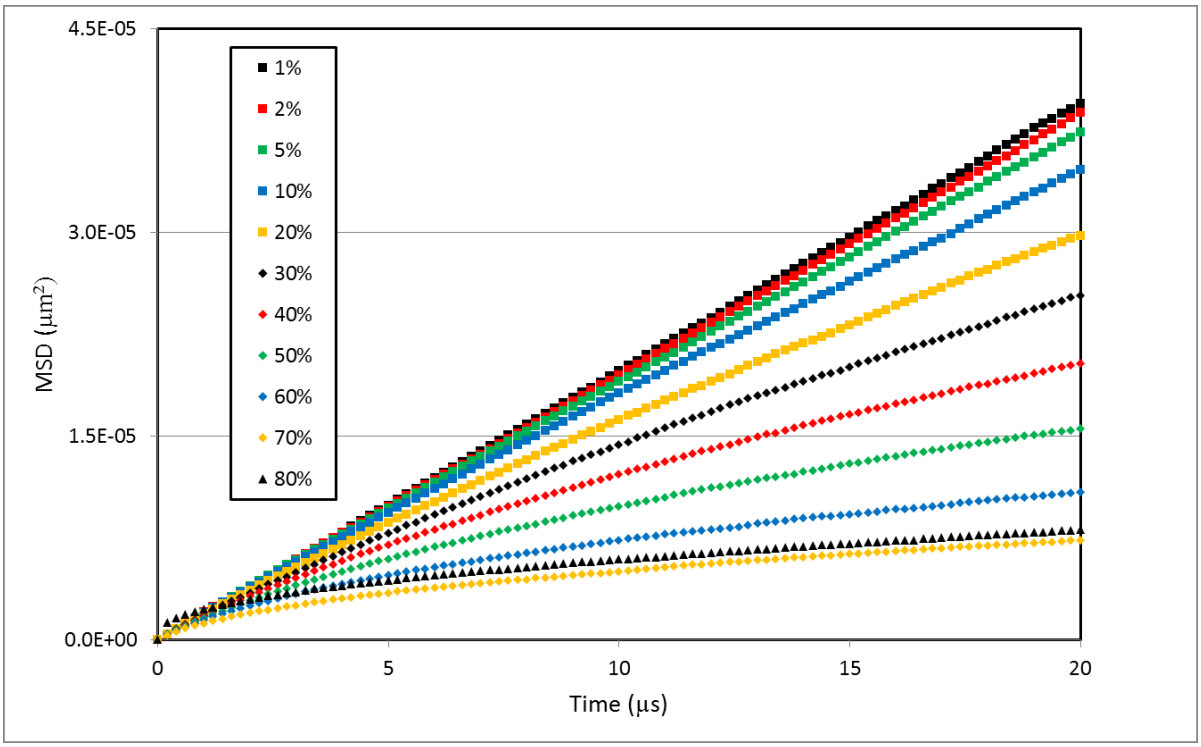

Figure 3: Effects of crowding on 1D diffusion. A plot of MSD vs time for the system described in the text as the density is varied, as indicated in the legend.

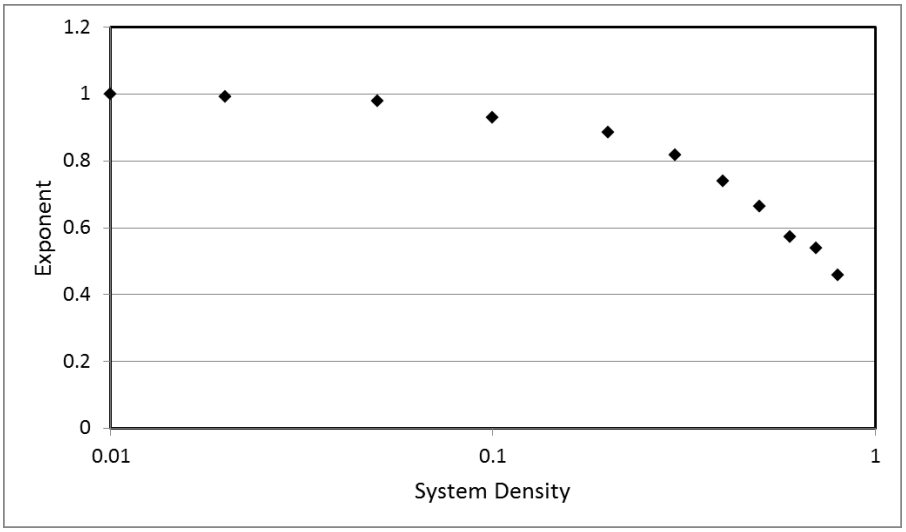

Figure 4: Effects of crowding on the scaling exponent. Each curve in Fig. 3 was fit to $msd(t) = at^b$, and here we plot b as a function of system density. At low densities b matches the exponent for free diffusion, while at large densities b approaches the theoretical result for crowded diffusion.

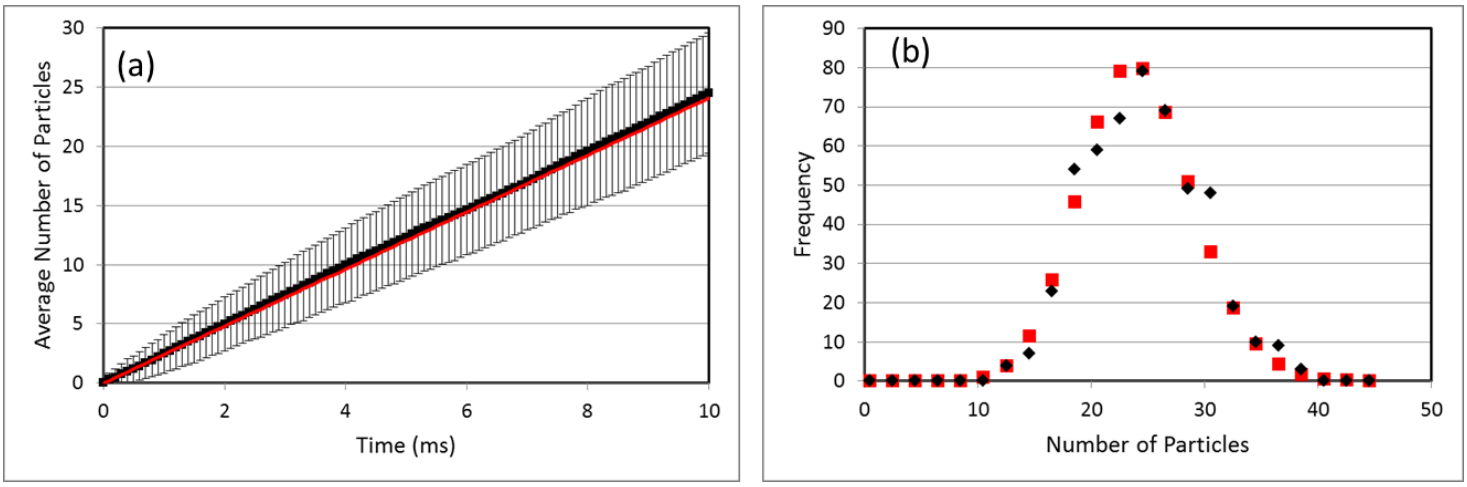

Figure 5: Validation of creation reactions. (a) Average number of particles as a function of time for the system described in the text. Average is taken over 500 independent simulations, error bars are standard deviation. The red line is the expected solution, Eq. (5). (b) The full distribution at 10 ms calculated over 500 runs (black diamonds) and the expected distribution according to Eq. (4) (red squares). The two distributions are statistically equivalent ($\chi^2 = 18.52, p = 0.07 > 0.05$).

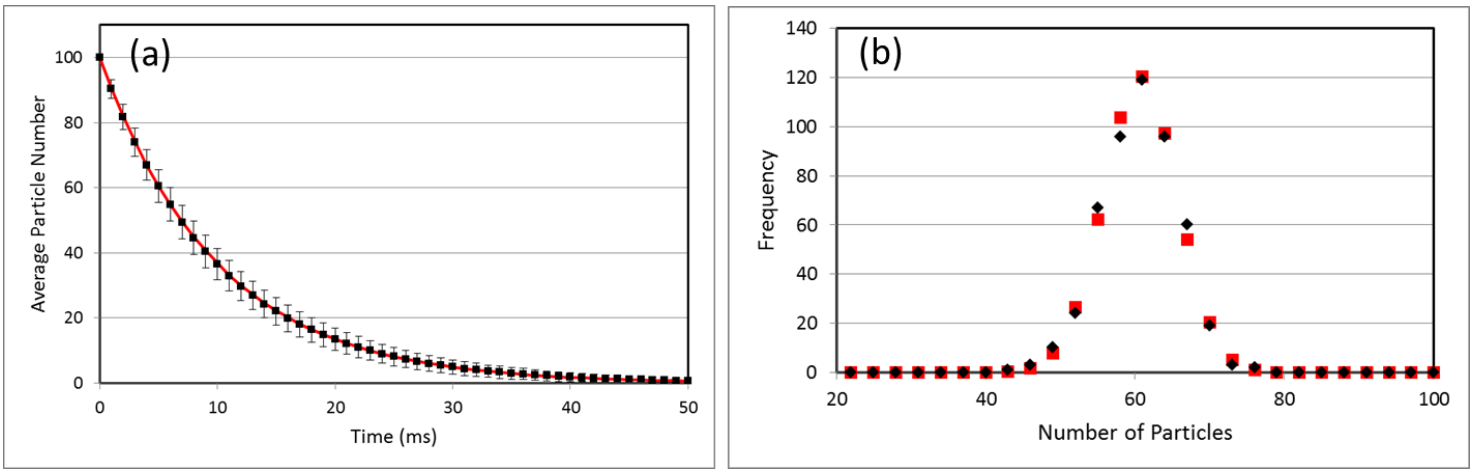

Figure 6: Validation of decay reactions. (a) Average number of particles as a function of time for the system described in the text. Average is taken over 500 independent simulations, error bars are standard deviations. The red line is the expected solution, Eq. (7). (b) The full distribution at 5 ms calculated over 500 run (black diamonds) and the expected distribution according to Eq. (6) (red squares). The two distributions are statistically equivalent ($\chi^2 = 3.30, p = 0.86 > 0.05$).

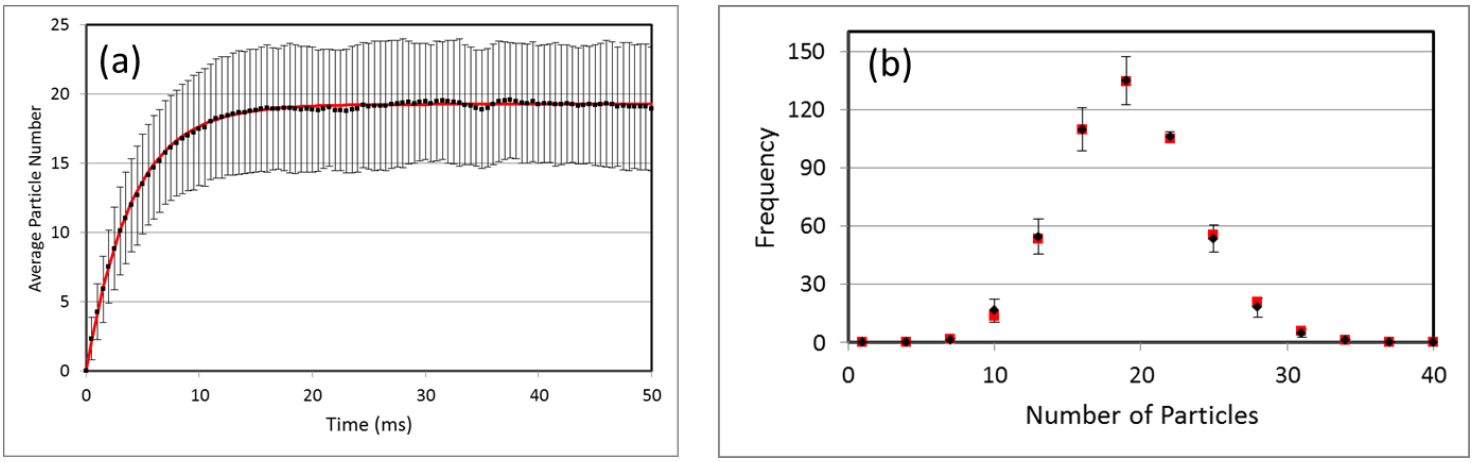

Figure 7: Validation of creation and decay reactions. (a) Average number of particles as a function of time for the system described in the text. Average is taken over 500 independent simulations, error bars are standard deviation. The red line is the expected solution, Eq. (9). (b) The full equilibrium distribution (black diamonds) compared to the expected distribution (red squares) from Eq. (8). The plot shows the average and standard deviation of the full distributions at 20, 30, 40, and 50 ms. The two distributions are statistically equivalent ($\chi^2 = 1.20,\ p = 0.98 > 0.05$).

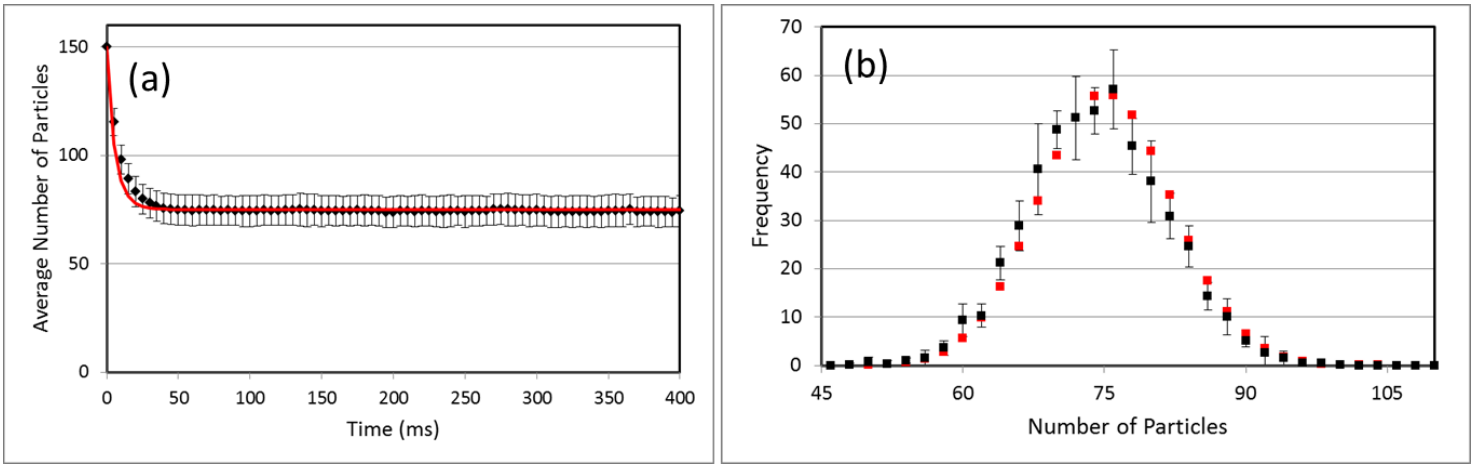

Figure 8: Validation of single species bimolecular reactions. (a) Average number of particles as a function of time for the system described in the text. Average is taken over 500 independent simulations, error bars are standard deviation. The red line is the expected solution, Eq. (15). (b) The full equilibrium distribution (black squares) compared to the expected distribution (red squares) from Eq. (14). The plot shows the average and standard deviation of the full distribution at 100, 150, 200, 250, 300, 350, and 400 ms. The two distributions are statistically equivalent ($\chi^2 = 10.21,\ p = 0.81 > 0.05$).

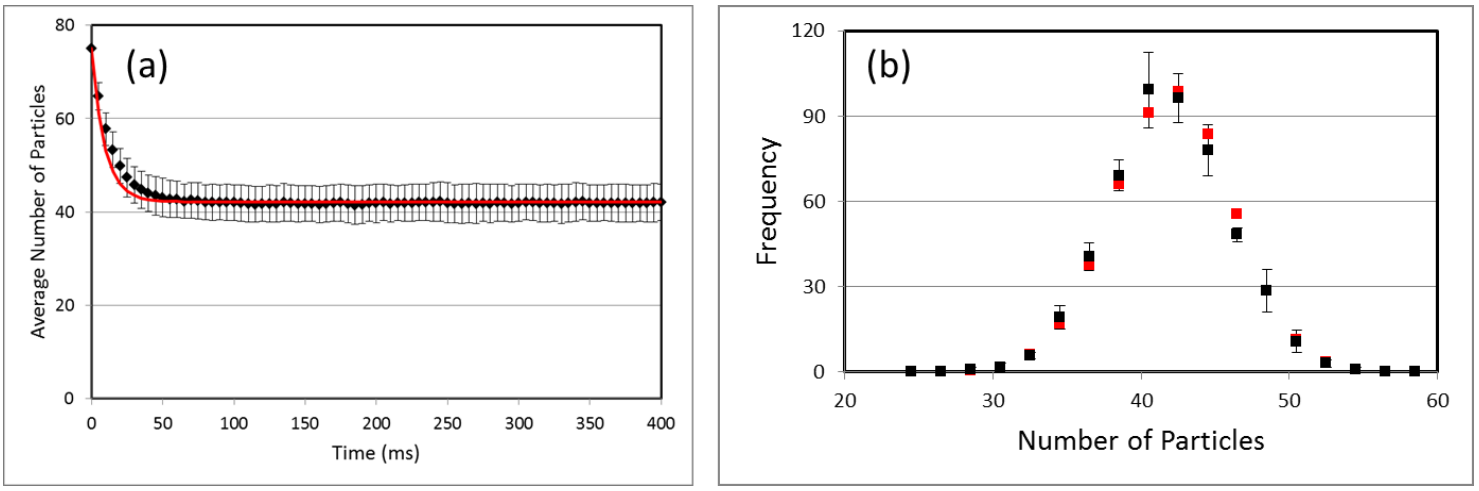

Figure 9: Validation of two species bimolecular reactions. (a) Average number of A particles as a function of time for the system described in the text. Average is taken over 500 independent simulations, error bars are standard deviation. The red line is the expected solution, Eq. (21). (b) The full equilibrium distribution (black squares) compared to the expected distribution (red squares) from Eq. (19). The plot shows the average and standard deviation of the full distribution at 100, 150, 200, 250, 300, 350, and 400 ms. The two distributions are statistically equivalent ($\chi^2 = 2.92,\ p = 0.98 > 0.05$).

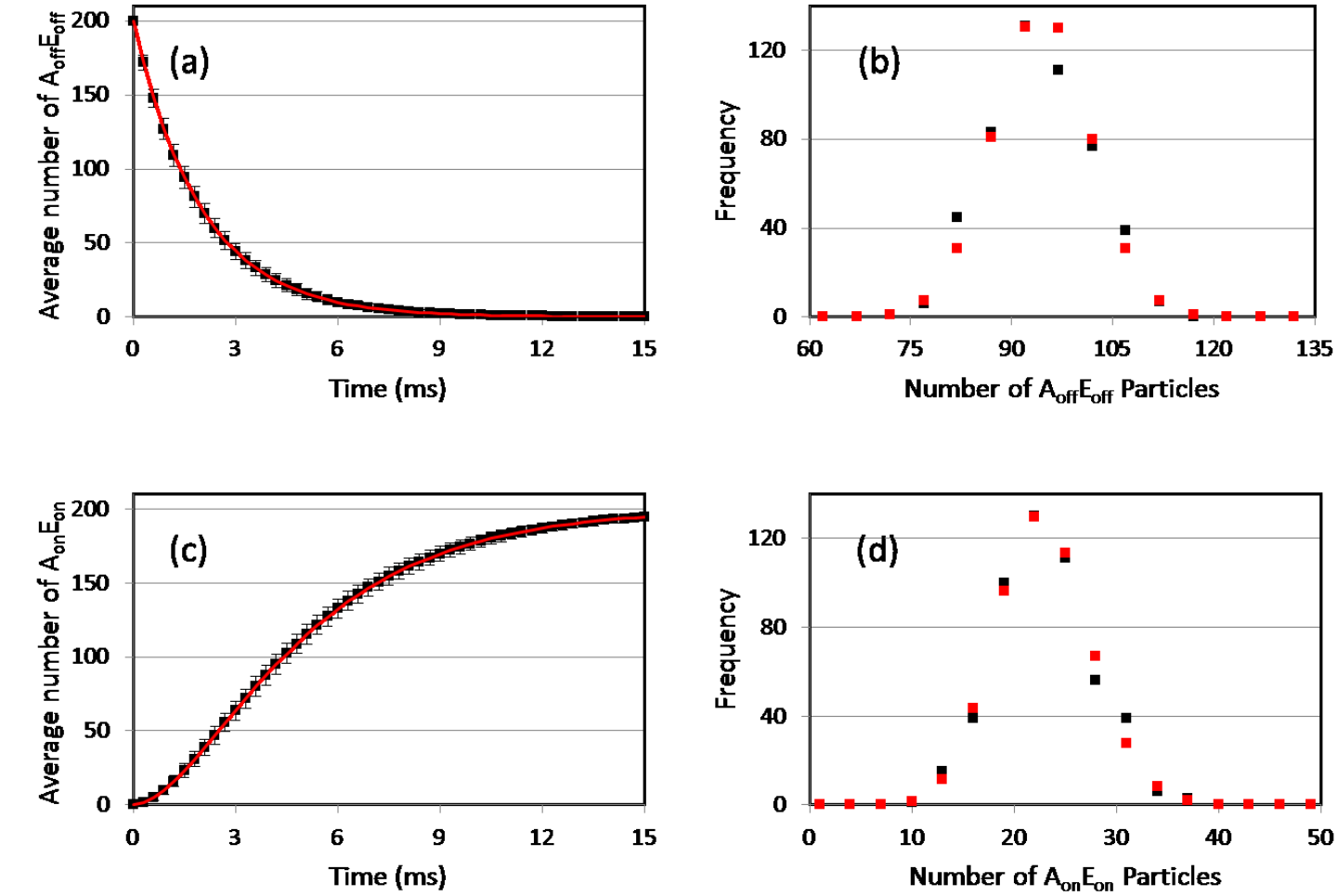

Figure 10: Validation of allosteric transition reactions. (a,c) Average number of $A_{\text{off}}E_{\text{off}}$ (a) and $A_{\text{on}}E_{\text{on}}$ (c) molecules as a function of time, for the system described in the text. The average is taken over 500 independent simulations, and the error bars are standard deviation. The red lines are the predictions from Eqs. (30) and (32). (b,d) The full distributions at 1.5 ms for $A_{\text{off}}E_{\text{off}}$ (b) and $A_{\text{on}}E_{\text{on}}$(d) (black squares) compared to the expected distributions (red squares) from Eqs. (27) and (29). Both observed distributions are statistically equivalent to their expected distributions ($A_{\text{off}}E_{\text{off}}$: $\chi^2 = 11.80,\ p = 0.11 > 0.05$, $A_{\text{on}}E_{\text{on}}$: $\chi^2 = 9.15,\ p = 0.24 > 0.05$).

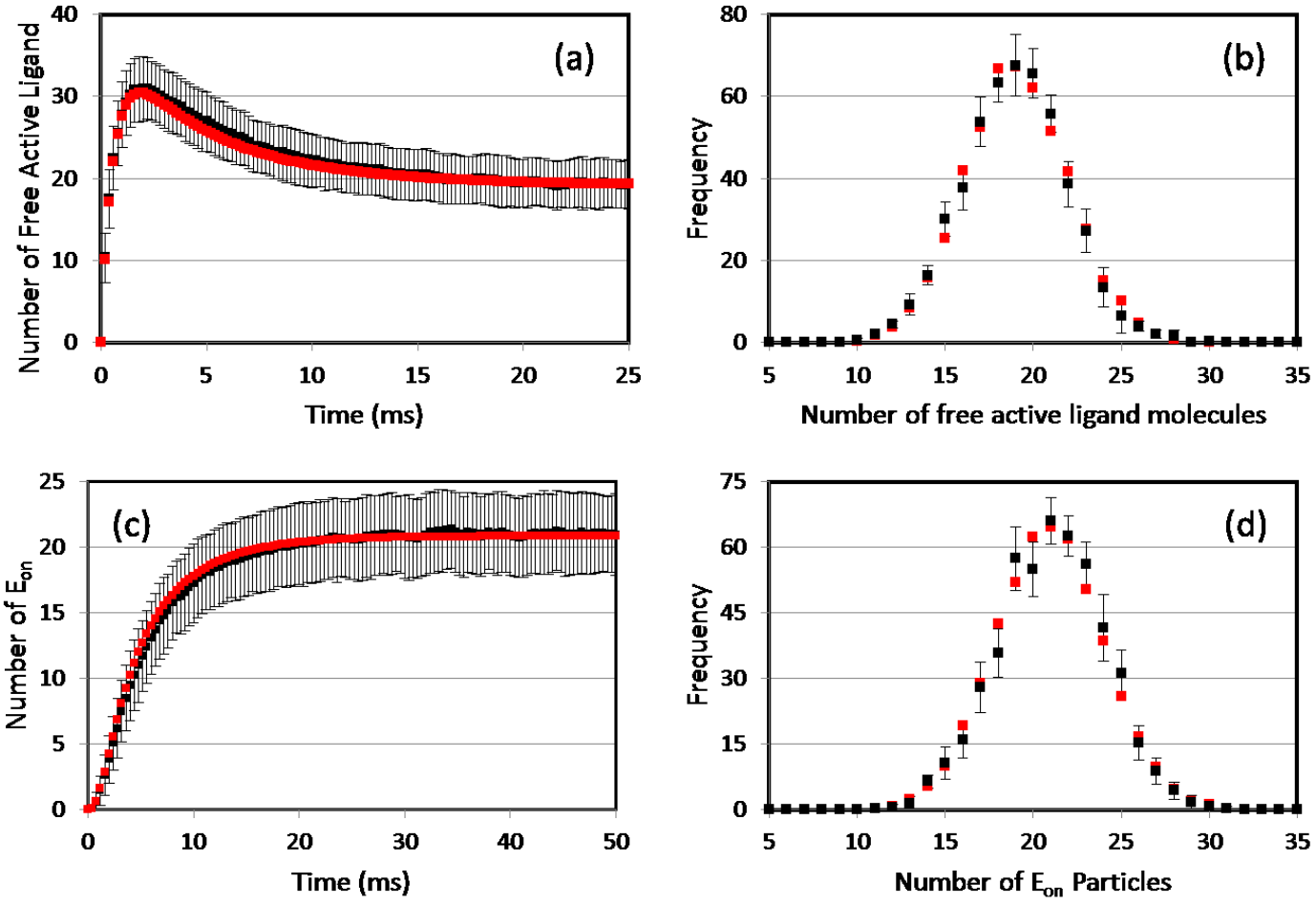

Figure 11: Validation of enzymatic activation reactions. (a,c) Average number of L_{act} (a) and E_{on} (c) molecules as a function of time, for the system described in the text. The average is taken over 500 independent simulations, and the error bars are standard deviation. The red lines are the results of deterministic VCell simulations. (b,d) The full equilibrium distribution for L_{act} (b) and E_{on}(d) (black squares) compared to the expected distributions. The expected distributions represent the results of 10000 VCell non-spatial stochastic simulations. The observed distributions are the average over the distributions at 40, 80, 120, 160, and 200 ms. Both observed distributions are statistically equivalent to their expected distributions (L_{act}: $\chi^2 = 3.90,\ p = 0.99 > 0.05$, E_{on}: $\chi^2 = 5.55,\ p = 0.96 > 0.05$).

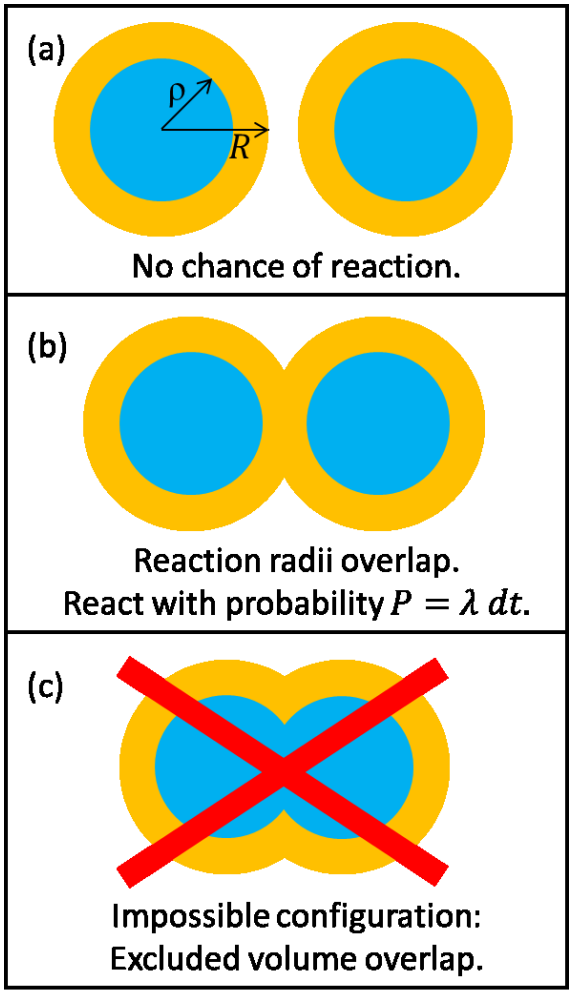

Figure 12: Schematic illustration of the relationship between the physical radius, ρ, and reaction radius, R, used in defining the bimolecular reaction rate λ.

Parameter Name	Value
k_L_act	$1000\ s^{-1}$
k_L_inact	$500\ s^{-1}$
k_E_act	$1000\ s^{-1}$
k_E_inact	$500\ s^{-1}$
k_{on}	$30\ \mu M^{-1}s^{-1}$
k_{off}	$100\ s^{-1}$

Table S1: Parameters used in the "Enzymatic Activation" model, which is used to validate the transition reactions. Identical parameters were used in both the VCell and SpringSaLaD models.