P

Particle-Based Stochastic Simulators

Steven S. Andrews
Division of Basic Sciences, Fred Hutchinson Cancer Research Center, Seattle, WA, USA
Department of Physics, Seattle University, Seattle, WA, USA

Definition

A stochastic simulator that tracks the location and interactions of individual molecules of interest. Molecules are represented with minimal internal detail.

Detailed Description

Computational simulations are widely used in neuroscience, as in other branches of biology, to explore the implications of quantitative models. They are used to investigate the sequestration of calcium calmodulin kinase II (CamKII) in dendritic spines, the transmission of action potentials through networks of neurons, and the release of heterogeneous synaptic vesicles, among many other topics. Some simulations are used with abstract "toy models" to identify fundamental biological principles, such as which signaling network topologies can transmit information particularly accurately. Others are tightly integrated with experimental work to build better understandings of particular systems.

Neuroscience simulation methods can be broadly categorized by the level of detail that they represent. At the coarse end of the scale, one can numerically integrate the differential rate equations that represent the kinetics of biochemical reaction and transport processes (Aldridge et al. 2006). Such integration methods typically assume few spatial compartments and ignore stochasticity. At the other extreme lie molecular dynamics approaches, in which one simulates the intricate motions of individual atoms within potential energy landscapes (Rapaport 2004). In moving along the continuum from coarser to finer levels of detail, these simulation methods typically address smaller total volumes of space, simulate shorter amounts of time, require more simulation parameters, and become more computationally intensive. They also consider space more realistically, such as by moving from a compartment description to a lattice, and then on to continuous space. In addition, the simulators change from using deterministic computations, which are appropriate when enough molecules are considered that their fluctuations become irrelevant, to stochastic computations, which account for natural fluctuations and are essential when there are few molecules.

Particle-based simulations lie toward the detailed end of this scale. They track the locations and interactions of all individual molecules of interest, while ignoring water and other molecules

D. Jaeger, R. Jung (eds.), *Encyclopedia of Computational Neuroscience*,
https://doi.org/10.1007/978-1-4614-7320-6_191-2

that are not of interest. They typically address sizes that range from single nanometers to tens of microns, and time scales from tens of nanoseconds to minutes. This level of detail makes them good for simulating problems involving molecular diffusion, chemical reactions, macromolecular complex formation, and/or molecule-membrane interactions. Their level of detail would also make them appropriate for investigating dynamics involving the conformations of cytoskeletal and nucleic acid filaments (Andrews 2014), although those topics have received less attention so far and are minimally supported in current software. Particle-based methods lie between the coarser spatial Gillespie methods (e.g., Drawert et al. 2012) and the finer Brownian and molecular dynamics methods (Ermak and McCammon 1978; Rapaport 2004).

Naively, it might seem best to work at the highest possible level of detail because it is the most accurate. In fact though, the opposite is generally the case. It is typically best to work at the lowest level of detail that adequately addresses the research question at hand because doing so enables simpler models and faster simulations; the importance of faster simulations should not be underestimated because simulation speed largely dictates the pace of the research and determines the extent of parameter scanning and parameter fitting that are practical. Furthermore, there is little benefit in simulating a model with more detail than is in the model itself. For example, simulating encounters between structurally complicated proteins with subnanometer precision is not meaningful when those same proteins are being approximated as perfect isotropic spheres.

Many particle-based simulators have been developed over the past three decades (see reviews Blackwell 2013; Sokolowski and ten Wolde 2017; Schöneberg et al. 2014). Of these, we are aware of six that are being actively maintained, meaning in this case that they have had new releases in the past 3 years. These simulators are described below and in Table 1, listed roughly in order of increasing detail. The simulators that are not listed here (see Schöneberg et al. 2014) are generally less mature, with fewer features, less documentation, and/or more difficult installation.

MCell and Smoldyn

MCell (Stiles and Bartol 2001, http://mcell.org) and Smoldyn (Andrews et al. 2010, http://www.smoldyn.org) work at similar levels of detail. Both are able to represent molecules as point-like particles that diffuse freely and only interact when pairs of reactants collide with each other and then undergo a chemical reaction. These molecules can also adsorb to surfaces, desorb from surfaces, and diffuse along surfaces. Both programs are mature, with easy installation, good documentation, and many features. MCell was initially developed for modeling reactions and diffusion in the neuromuscular junction (Stiles et al. 1996) and continues to be used for a wide variety of neuroscience simulations. MCell supports molecules with multiple states, such as arise with phosphorylation (Stefan et al. 2014). It has a graphical user interface that works with the open-source Blender software and provides excellent graphical output.

Smoldyn does not support graphical input but reads text input files instead. Smoldyn's graphical output, which displays as the simulation runs, is good but inferior to MCell's. On the other hand, Smoldyn runs about three times as fast as MCell, is more accurate (Andrews et al. 2010; Andrews 2017), and offers more features. All of Smoldyn's algorithms approach exactness, meaning perfect agreement with the underlying conceptual model, as time steps are reduced towards zero and it also simulates reaction rates, on-surface diffusion, and molecule-surface interactions quite accurately even with very long time steps. Smoldyn allows molecules to have excluded volumes, enabling simulations of macromolecular crowding. Smoldyn offers rule-based modeling, with either BioNetGen or a new wildcards method, for modeling the dynamics of multimeric complexes. It also offers multiscale simulation using spatial Gillespie approaches and has been combined with partial differential equation (PDE) methods in the popular Virtual Cell software (Schaff et al. 2016). Smoldyn is used particularly in the biophysics and systems biology communities.

Particle-Based Stochastic Simulators, Table 1 Comparison of different simulators. See http://www.smoldyn.org/simulators.html for detailed notes about all table entries and any updates. System boundaries codes: R = reflecting, A = absorbing, P = periodic, T = transmitting, and I = interacting. *Algorithm is exact but software produces incorrect results. †These benchmark run times are not comparable with others due to differing levels of detail

	MCell	Smoldyn	eGFRD	SpringSaLaD	ReaDDy
Time steps	~ 1 μs	Ns to ms	Event-based	~10 ns	~0.1 ns to μs
Molecules	Points	Points, spheres	Spheres	Multi-spheres	Multi-spheres
Dimensions	2,3	1,2,3	3	3	3
System boundaries	R, A, P, T	R, A, P, T	P	R	P, I
Surfaces	Triangle mesh	Many primitives	–	1 flat surface	Plane, sphere
Surface molecules	1/tile, 2 states	Unlimit., 4 states	–	Unlimit., 3 states	–
Excluded volume	–	Excellent	Exact	Good	Excellent
Multimers	States only	Rule-based model	–	Explicit	Explicit
Allostery	–	Yes	–	Yes	–
Reaction accuracy	Very good	Excellent	Exact*	Excellent	Excellent
Dissociation products	Stochastic	Fixed separat.	Adjacent	Adjacent	Adjacent
Molec.-surf. interact.	Good	Excellent	–	To sites only	Potentials
Long-range interact.	–	–	–	–	Yes
Hybrid simulation	–	Sp. Gillespie, PDE	–	–	–
Benchmark run time	67 s	22 s	13 days†	9.1 months†	13 min.
Distribution	Executable	Executable	Self-compile	Java file	Self-compile
User interface	GUI, text	Text	Text	GUI	Python script
Graphical output	Excellent	Good	Partial support	Partial support	Good
Library interface	Python	C/C++	–	–	Python

E-Cell and eGFRD

The enhanced Green's Function Reaction Dynamics (eGFRD) algorithm (Takahashi et al. 2010) is an exact approach for simulating chemical reactions between molecules that can be represented as perfect spheres. Simulated molecules diffuse in agreement with ideal Brownian motion and nonreactive pairs of molecules reflect off of each other upon collision to account for excluded volume effects. Bimolecular reactions exactly obey the Collins and Kimball chemical reaction model, in which molecules are treated as hard spheres that have an "intrinsic reaction rate" when in contact (Rice 1985); this model is widely assumed in the particle-based simulation field. The eGFRD algorithm achieves this high level of detail by using an event-driven method in which it steps the simulation time from the moment of one interaction event to the moment of the next event. The two original algorithm developers, Takahashi and ten Wolde, have now implemented it in separate software: E-Cell (Tomita et al. 1999, http://www.e-cell.org), a platform that offers simulators that work at several levels of detail, and eGFRD (http://gfrd.org), which only runs the eGFRD algorithm.

The eGFRD algorithm is very efficient for dilute systems because it does not simulate multiple diffusive steps between molecule encounters. However, it is slow for dense systems because these have a large number of molecule encounters and the algorithm performs complicated calculations at each one. For example, I found that it was

several orders of magnitude slower than Smoldyn when simulating a Michaelis-Menten reaction that started with 10,000 molecules (Table 1). This result agreed with speed differences found when testing reaction accuracy (Table 1) and in a prior macromolecular crowding study (Andrews et al. 2015).

SpringSaLaD and ReaDDy

SpringSaLaD (Michalski and Loew 2016, http://health.uconn.edu/cell-analysis-modeling/ccam-software/) and ReaDDy (Schöneberg and Noé 2013, http://www.readdy-project.org) were designed to bridge the gap between simulation approaches that treat molecules as perfect spheres and molecular dynamics methods. These simulators represent individual molecules as collections of spheres that are bound together with either springs, in the case of SpringSaLaD, or with interaction potentials, in the case of ReaDDy. Both simulators move the separate spheres by numerically integrating the equations of overdamped Langevin dynamics, a physical theory that combines the fluctuating forces that drive Brownian motion and the viscous drag forces that slow it down. This integration requires very short time steps, typically on the order of 1–10 ns (although not its main purpose, ReaDDy can also ignore inter-particle potentials, resulting in much faster simulations). SpringSaLaD represents cell membranes and other surfaces minimally, using a single planar membrane within the simulation volume, while ReaDDy represents surfaces using a combination of planes or spheres, each of which can interact with molecules through an interaction potential.

SpringSaLaD and ReaDDy are particularly useful for simulating the dynamics of protein complexes, such as the multiple subunits of CamKII, along with their interactions with surrounding molecules. This greater level of detail makes these simulators more computationally intensive. They are generally best for systems with a thousand molecules or fewer and can simulate up to a few milliseconds of total time.

Conclusion

Particle-based stochastic simulators are powerful tools for exploring the roles of spatial organization and stochasticity in cell-scale systems. They are used widely in neuroscience research at present and will undoubtedly be used more in the future as the software improves, computers run faster, and biology research becomes more quantitative. Current simulators are able to accurately represent the diffusion, chemical reactions, complexation, and membrane interactions of individual molecules of interest. These simulators work at different levels of detail, ranging from MCell, which treats molecules as point-like particles, to ReaDDy, which addresses interaction forces between portions of molecules.

All of the tools described here are open source and have nonrestrictive licenses that allow for both commercial and noncommercial use. They are also works in progress that are frequently expanded and improved in response to users' needs. Although perhaps less evident, each simulator is also the product of many person-years of development and testing. If you need a capability that is not supported by one of these tools, I encourage contacting the teams that work on these programs for advice on working around limitations before spending time writing new code.

Cross-References

- ▶ MCell
- ▶ MOOSE, the Multiscale Object-Oriented Simulation Environment
- ▶ Stochastic Simulators

References

Aldridge BB, Burke JM, Lauffenburger DA, Sorger PK (2006) Physicochemical modelling of cell signalling pathways. Nat Cell Biol 8(11):1195–1203

Andrews SS (2014) Methods for modeling cytoskeletal and DNA filaments. Phys Biol 11(1):011001

Andrews SS (2017) Smoldyn: particle-based simulation with rule-based modeling, improved molecular interaction and a library interface. Bioinformatics 33(5):710–717

Andrews SS, Addy NJ, Brent R, Arkin AP (2010) Detailed simulations of cell biology with Smoldyn 2.1. PLoS Comput Biol 6:e1000705

Andrews SS, Arjunan SN, Balbo G, Bittig AT, Feret J, Kaizu K, Liu F (2015) Simulating macromolecular crowding with particle and lattice-based methods (Team 3). In: Gilbert D, Heiner M, Takahashi K, Uhrmacher AM (eds) Multiscale spatial computational systems biology. Dagstuhl Reports 4(11):170–187. http://drops.dagstuhl.de/opus/volltexte/2015/4972. https://doi.org/10.4230/DagRep.4.11.138

Blackwell K (2013) Approaches and tools for modeling signaling pathways and calcium dynamics in neurons. J Neurosci Methods 220(2):131–140

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

Ermak DL, McCammon J (1978) Brownian dynamics with hydrodynamic interactions. J Chem Phys 69(4):1352–1360

Michalski PJ, Loew LM (2016) SpringSaLaD: a spatial, particle-based biochemical simulation platform with excluded volume. Biophys J 110(3):523–529

Rapaport D (2004) The art of molecular dynamics simulation. Cambridge University Press, Cambridge

Rice SA (1985) Diffusion-limited reactions. Elsevier, Amsterdam

Schaff JC, Gao F, Li Y, Novak IL, Slepchenko BM (2016) Numerical approach to spatial deterministic-stochastic models arising in cell biology. PLoS Comput Biol 12(12):e1005236

Schöneberg J, Noé F (2013) ReaDDy-a software for particle-based reaction-diffusion dynamics in crowded cellular environments. PLoS One 8(9):e74261

Schöneberg J, Ullrich A, Noé F (2014) Simulation tools for particle-based reaction-diffusion dynamics in continuous space. BMC Biophys 7(1):1

Sokolowski TR, ten Wolde PR (2017) Spatial-stochastic simulation of reaction-diffusion systems. arXiv preprint arXiv:1705.08669

Stefan MI, Bartol TM, Sejnowski TJ, Kennedy MB (2014) Multi-state modeling of biomolecules. PLoS Comput Biol 10(9):e1003844

Stiles JR, Bartol TM (2001) Chapter 4, Monte Carlo methods for simulating realistic synaptic microphysiology using MCell. In: De Schutter E (ed) Computational neuroscience, realistic modeling for experimentalists. CRC Press, Boca Raton, pp 87–127

Stiles JR, Van Helden D, Bartol TM, Salpeter EE, Salpeter MM (1996) Miniature endplate current rise times less than 100 microseconds from improved dual recordings can be modeled with passive acetylcholine diffusion from a synaptic vesicle. Proc Natl Acad Sci 93(12):5747–5752

Takahashi K, Tănase-Nicola S, Ten Wolde PR (2010) Spatio-temporal correlations can drastically change the response of a MAPK pathway. Proc Natl Acad Sci 107(6):2473–2478

Tomita M, Hashimoto K, Takahashi K, Shimizu TS, Matsuzaki Y, Miyoshi F, Saito K, Tanida S, Yugi K, Venter JC et al (1999) E-cell: software environment for whole-cell simulation. Bioinformatics (Oxford, England) 15(1):72–84