- DOE SciDAC 2005 meeting, Journal of Physics: Conference Series 16, 305-309 (2005) -

Microbial Cell Modeling via Reacting Diffusive Particles

Steven J. Plimpton and Alex Slepoy
Sandia National Laboratories[1], Albuquerque, NM
E-mail: sjplimp@sandia.gov, aslepoy@sandia.gov

Abstract.
We describe a particle-based simulator called ChemCell that we are developing with the goal of modeling the protein chemistry of biological cells for phenomena where spatial effects are important. Membranes and organelle structure are represented by triangulated surfaces. Diffusing particles represent proteins, complexes, or other biomolecules of interest. Particles interact with their neighbors in accord with Monte Carlo rules to perform biochemical reactions which can represent protein complex formation and dissociation, ligand binding, etc. In this brief paper we give the motivation for such a model, describe a few of the code's features, and highlight interesting computational issues that arise in particle-based cell modeling.

1. Introduction

One of the next grand-challenge goals in biology, building on the deluge of genomics and proteomics data, is to understand how individual cells function as a collection of interacting biochemical molecules and molecular machines. Consider single-cell microbes, the simplest self-sufficient living organisms. At a macroscopic level they are a small bag of biochemical cytoplasm with little internal structure. Yet microbes create and use the fundamental molecular machinery common to all life (protein synthesis via DNA transcription and ribosomal mRNA translation, regulation of gene expression, membrane control of small molecule transport, signal propagation) to sense the state of their environment and respond appropriately (grow, move, reproduce). Eukaryotic cells execute enhanced versions of these regulatory, metabolic, and signaling networks within structures of rich spatial complexity (organelles, membranes, filaments, etc).

Such a picture poses a wide range of challenges for computational cell modeling which various groups are addressing in different ways. Typically, one key input to all such models is a set of reaction rate equations which specify the biochemistry to be modeled. Each equation has one or more reactants which are converted at a certain rate into one or more products. Complex networks may contain 100s or 1000s of such coupled equations. Using experiment and computation to deduce the relevant equations and their associated rate constants is its own area of active research.

Cell models can solve these equations as ODEs (effectively a zero-dimensional model) or in a spatial representation of the cell as reaction/diffusion PDEs. Examples of the former approach

[1] Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

are part of the Systems Biology Workbench (SBW) toolkit [8]; the Virtual Cell project [5] is a good example of the latter.

Cellular events can be triggered by the presence of one or a few copies of a biomolecule (e.g. a gene or protein), which is a concentration level not trackable by continuum models. Stochastic cell models, which have their origin in the work of Gillespie [3], track individual molecules while solving the set of coupled rate equations described above. Such models can be zero-dimensional or include spatial effects. Zero-dimensional solvers are part of the SBW. Several groups have been developing particle-based stochastic reaction models that track spatial diffusion in cellular geometries. These include MCell [6], Smoldyn [1], MesoRD [4], and our ChemCell project [7].

The motivation for such a model can be understood by considering Escherichia Coli (E. Coli). A single E. Coli cell is cylindrical in shape, a few cubic microns in size, with a dozen flagella. Its single strand of DNA has 3000 genes, most of whose function is known. It also contains a few million protein molecules, tens of thousands of ribosomes, similar numbers of tRNA and mRNA molecules, and tens of millions of small organic molecules and ions, with the remaining 70% of the cell volume being water [2]. From a high-performance computing standpoint, where simulations of many millions or even billions of particles or grid cells are becoming commonplace, a simulation of an E. Coli cell could potentially model all the biomolecules in the cell (excluding water) as individual particles, if the equations that describe their chemical interactions were known.

2. Model

In this section we describe a simple cellular model which can represent individual macromolecules and their interactions with each other. The model has several components: particles, cellular geometry, particle motion, and particle reactions. We discuss each of these in turn and indicate how they are combined in our ChemCell code to create a simulation.

In our model, an individual particle represents a single biomolecule. It could be a protein, a protein complex (e.g. a membrane-bound ABC transporter), a molecular machine (e.g. a ribosome), a portion of a DNA strand (e.g. a gene or operon), or a smaller organic molecule that will bind to a protein target. Particles in the model have an x,y,z position and a species type. A particular protein or complex that exists in several states (e.g. due to phosphorylation) can be represented by different species.

The geometry of the cell is represented by one or more triangulated surfaces (membranes) which serve as barriers to particle diffusion. These can be idealized spheres or ellipsoids, or can be reconstructed from actual image data. The collection of surfaces define the cell's and organelle's geometry as in the diagram at the left of Figure 1.

Each timestep in the simulation, particles diffuse independently via Brownian motion, within an implicit background of water and small organic molecules. The membrane surfaces affect the motion of particles as they can be semipermeable or impermeable to different species. Particles in the cytoplasm diffuse in 3d. Particles may also be bound to membranes and diffuse in 2d along the surface.

Each timestep, particles can also react with their neighbors to carry out a prescribed set of biochemical reactions. For example, the reaction $A + B \rightarrow C$ can occur if two particles of species A and B are near each other. This would replace particles A and B with a new particle C. This happens in a Monte Carlo sense where each reaction is assigned a probability based on the diffusion coefficients of the reactants, their relative separation, the reaction rate, and the timestep size. Random numbers are used to pick which reactions take place. The schematic at the right side of Figure 1 illustrates this idea. Were the densities of A and B particles uniform throughout the cell volume, the computed probability factors would produce the same number of reactions that would occur in a well-mixed system due to the Gillespie stochastic simulation algorithm. However, in our model there can be spatial gradients in particle density due to

diffusion effects. The model is thus a stochastic particle formulation of the reaction/diffusion equations used in continuum cell models.

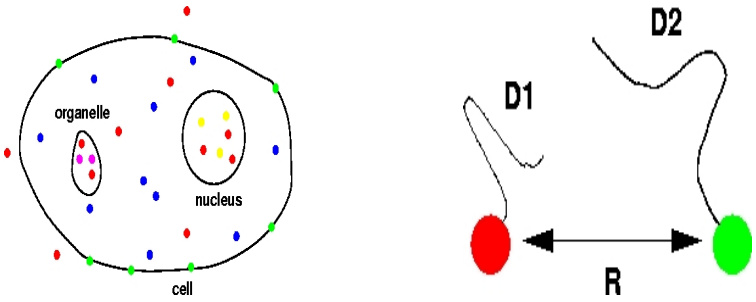

Figure 1. *(Left) A simplified 2d model of cell geometry where the cytoplasm is partitioned by membranes and particles diffuse in the cytoplasm or on the membranes. (Right) The reaction probability for 2 nearby particles is a function of their diffusion rates and the Gaussian distribution they sample over a diffusive timestep.*

The input to a ChemCell model includes the geometry of the cell and a list of chemical species and reactions with their associated diffusion coefficients and reaction rates. A simulation proceeds by alternating between diffusion moves and Monte Carlo reactions for all the particles each timestep. The output of the simulation is the species concentrations in time and space.

Figure 2 illustrates two different ChemCell models. The first is a plot of species concentration versus time for a simple model of a 3-stage protein kinase cascade (9 species, 7 reactions). As the first kinase species is activated, it's concentration spikes upward. It in turn activates the second kinase species, which activates the third. All three species eventually disappear as the receptor which triggers the cascade becomes deactivated. The figure plots the ChemCell spatial results on top of those from a Gillespie zero-dimensional calculation for the same set of reactions. When the diffusion coefficient in the spatial model is large, there is good agreement between the 2 models for the peak heights in both time and intensity, indicating the stochastic spatial model can reproduce the well-mixed results.

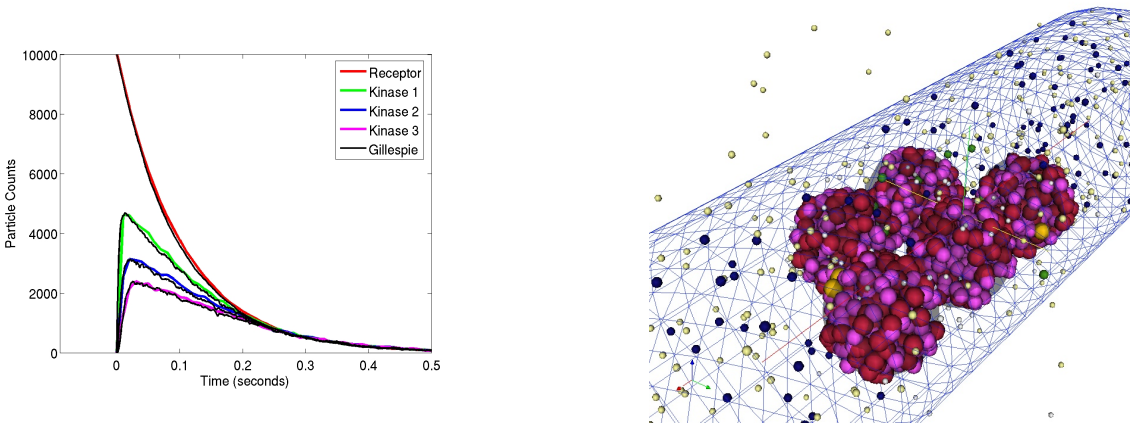

Figure 2. *(Left) Particle concentration versus time for several species involved in a protein kinase cascade. (Right) A snapshot from a ChemCell model of carboxysome organelles in a rod-shaped Synechococcus cyanobacterium.*

The right side of Figure 2 illustrates a more complex ChemCell model of a Synechococcus

cyanobacterium. The outer membrane is shown as transparent triangles; the inner clumps of particles are carboxysome organelles where carbon fixation reactions take place at catalytic sites on RuBisCO protein molecules. This is a snapshot from a simulation of a portion of the metabolic cycle that converts inorganic carbon to organic sugar in Synechococcus (16 species, 11 reactions) as an illustration of the spatial and chemical complexity such a particle model can encode.

3. Challenges

Cell models such as those described in this paper pose interesting algorithmic challenges for computational scientists. The issues highlighted in this section are active areas of research in both continuum and stochastic cell modeling.

From a biology standpoint, the challenge is to represent cellular function with model inputs (diffusion constants, lists of reactions, reaction rates) that enable both qualitative and quantitative comparison of simulation results with experiment, and which ultimately lead to biological insight. This is a difficult task, since model inputs are often not known to high fidelity. Algorithms are needed for performing parameter estimation and testing various models for stability, sensitivity, and robustness.

For reaction networks involving protein complexes with several constituents, each of which can be in several states, there can be a combinatorial explosion of intermediate species and reactions that are created as binding sites are occupied and large protein complexes form. In a model like ChemCell, these species and their associated reactions must be tracked, and diffusion and reaction rates assigned to them. Efficient ways are needed to represent these states and compute which will be needed either *a priori* or as they are generated during the simulation.

Algorithms are needed for coupling stochastic species (low-density) with continuum species (high-density) in a hybrid simulation model.

The algorithms used (in our code and others) for performing stochastic reactions between particles are heuristic approaches that attempt to mimic Gillespie's original stochastic simulation algorithm [3] which is derived from the chemical master equation. It is an open question what is the most accurate or computationally efficient such algorithm. Performing Monte Carlo reactions in parallel adds a new dimension to the question. Gillespie's original algorithm has limited inherent parallelism, since it performs reactions one at a time and updates all reaction probabilities that are changed by the occurrence of the reaction.

Additional parallel issues for simulations with millions or more particles include efficient neighbor finding and load-balancing of diffusion and reactions within a complex cellular geometry where particle densities can change in time and space. We have attempted to address this load-balancing issue in ChemCell by partitioning the cell domain spatially and adjusting the partitions either statically or dynamically to track density variations. Figure 3 shows parallel performance data for a simple benchmark problem in a spherical cell geometry. Timing data is converted to parallel efficiency for three fixed-size problems of varying particle count. The parallel efficiency (normalized to 100% on one processor) is plotted versus processor count for runs on the Intel ASCI Red Tflops machine at Sandia with load-balancing enabled and disabled.

From a computational standpoint, particle models like ChemCell are similar to large-scale parallel particle codes developed by many groups, e.g. molecular dynamics, direct-simulation Monte Carlo (DSMC), and electromagnetic particle-in-cell (PIC) models. The wealth of computational and algorithmic ideas that have been developed for those applications can hopefully be brought to bear on biological cell models as well. Since ChemCell is an open-source research tool, designed to allow for easy experimentation with new physical and computational algorithms, our hope is that it will be a useful platform for such efforts.

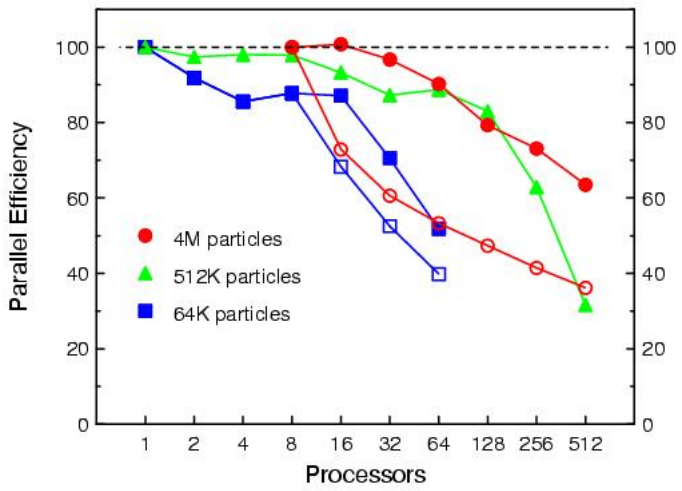

Figure 3. *Parallel efficiency for 3 fixed-size particle simulations performed with ChemCell. Shaded symbols are for runs with load-balancing turned on; open symbols are for runs with load-balancing turned off.*

Acknowledgments

This work has been funded by the US Department of Energy's Genomics: GTL program (www.doegenomestolife.org) under project, "Carbon Sequestration in *Synechococcus Sp.*: From Molecular Machines to Hierarchical Modeling," (www.genomes-to-life.org).

References

[1] S. S. Andrews and D. Bray, Phys. Biol., 1, 137-151 (2004), Smoldyn WWW site: www.anat.cam.ac.uk/ compcell/Smoldyn.html.

[2] D. S. Goodsell, The Machinery of Life, Springer-Verlag, NY, 1993.

[3] D. T. Gillespie, J. Phys. Chem., 81, 2340-2361 (1977).

[4] J. Hattne, D. Fange, and J. Elf, Bioinformatics, 21, 2923-2924 (2005), MesoRD WWW site: mesord.sourceforge.net.

[5] L. M. Loew and J. C. Schaff, Trends in Biotechnology, 19, 401-406 (2001), Virtual Cell WWW site: www.nrcam.uchc.edu.

[6] J. R. Stiles and T. M. Bartol, Computational Neuroscience: Realistic Modeling for Experimentalists (ed. by E. De Schutter) 87-127 (2001), MCell WWW site: www.mcell.cnl.salk.edu.

[7] S. J. Plimpton and A. Slepoy. Sandia Technical Report SAND2003-4509 (2003), available from http://www.genomes2life.org/publications.

[8] Systems Biology Workbench WWW site: http://sbw.sourceforge.net.