COMPUTATIONAL CELL BIOLOGY:
Spatiotemporal Simulation of Cellular Events

Boris M. Slepchenko, James C. Schaff, John H. Carson, and
Leslie M. Loew
*Center for Biomedical Imaging Technology, University of Connecticut Health Center,
Farmington, Connecticut 06117; e-mail: les@volt.uchc.edu*

■ **Abstract** The field of computational cell biology has emerged within the past 5 years because of the need to apply disciplined computational approaches to build and test complex hypotheses on the interacting structural, physical, and chemical features that underlie intracellular processes. To meet this need, newly developed software tools allow cell biologists and biophysicists to build models and generate simulations from them. The construction of general-purpose computational approaches is especially challenging if the spatial complexity of cellular systems is to be explicitly treated. This review surveys some of the existing efforts in this field with special emphasis on a system being developed in the authors' laboratory, *Virtual Cell*. The theories behind both stochastic and deterministic simulations are discussed. Examples of respective applications to cell biological problems in RNA trafficking and neuronal calcium dynamics are provided to illustrate these ideas.

CONTENTS

INTRODUCTION

In the past 20 years, the advent and dissemination of revolutionary new technologies have permitted cell biologists to probe the physics and chemistry of living cells in situ. Confocal and two-photon excited fluorescence microscopies permit

423

investigators to study the structure and dynamics of living cells with submicro-meter three-dimensional (3D) spatial resolution and with time resolutions as fast as milliseconds. These quantitative microscopies can be combined with fluorescent indicators and fluorescent protein constructs to enable the study of the spatiotemporal behavior of individual molecules in cells. Patch clamp electrophysiological recording can be used to study ion currents through single-channel proteins or across the entire cell membrane. All these techniques can be further combined with methods to impart specific perturbations to cells such as photorelease of caged compounds to deliver controlled doses of second messengers or laser tweezer manipulations to determine the response of cells to mechanical stresses. Thus many of the cellular mechanisms, which in the past could only be studied in reconstituted artificial environments in test tubes, can now be studied in their native milieu and spatial organization.

Matching the advances in microscope-based technologies for studying living cells has been enormous progress in our cumulative knowledge of biomolecular structure and function. Massive structural biology efforts have produced extensive databases of 3D protein structures. High-throughput molecular biology and molecular genetics technologies have led to descriptions of the full genomes of several organisms, including, of course, the human genome. More recently, high-throughput proteomics technologies promise to catalog, for a given state of a given cell, the dynamic levels of and interactions between all proteins and their post-translational modifications. This wealth of molecular data has spawned the field of bioinformatics to provide computational tools for the organization, analysis, and synthesis of all this information. However, there is a critical need for new computational approaches that can link all the molecular-level data to the cellular processes that can be probed with the microscope. The nascent field of computational cell biology is emerging to fill this need and is the subject of this review.

The overall goal of computational cell biology is to enable cell biologists to build and exercise predictive models of cellular processes. An operational definition of the term model is most appropriately formulated in relation to the scientific method. A model, in this language, is simply a collection of hypotheses and facts brought together in an attempt to understand a phenomenon. Indeed, the choice of which hypotheses and facts to collect and the manner in which they are assembled themselves constitute additional hypotheses. For a cell biological model, the facts and hypotheses are composed of the molecular species and the biochemical or electrophysiological transformations that are presumed to underlie the cellular events. A prediction based on the model is in one sense most useful if it does not match the experimental details of the process—it then unequivocally tells us that the elements of the model are inaccurate or incomplete. Although such negative results are not always publishable, they are a tremendous aid in refining our understanding. If the prediction does match the experiment, it can never guarantee the truth of the model but should suggest other experiments that can test the validity of critical elements; ideally, it should also provide new predictions that can, in turn, be verified experimentally.

The very complexity of cell biological processes necessitates the development of new computational approaches to enable the application of the classical scientific method. A pair of separate factors contributes to this problem. First, the large number of interdependent chemical reactions and structural components that combine to affect and regulate a typical cell biological process forces one to seek the help of a computer to build a model and generate quantitative predictions from it. A structured computational framework is required to gather the relevant data about a cellular system and then execute mathematically rigorous simulations to establish whether the elements that were so identified are sufficient to produce an experimentally observed biological endpoint. The second factor recognizes that scientists trained in experimental cell biology are not typically equipped with sufficient mathematical, physical, or computational expertise to generate quantitative predictions from models. Conversely, theoretical biologists are often trained in the physical sciences and have difficulty communicating with experimentalists (bifurcation diagrams, for example, will not serve as a basis for a common language). Appropriate computational tools will therefore provide an interface that enables biologists to easily build models, run simulations, and visualize simulation results in a way that allows direct comparison to experiments. At the same time, these tools should be sufficiently sophisticated that they can facilitate the analysis of models by theorists and thus, ultimately, promote communication and collaboration between these communities.

This review describes current efforts to address this need, especially within the area of spatial modeling. We do not attempt to cover the large body of mathematical modeling studies that have used ad hoc approaches with either analytical solutions or numerical simulations that address only a specific problem. Rather, we focus on the design and application of several new general-purpose software tools for modeling intracellular physiology. The concentrations of reacting molecular species as a function of time in a well-mixed reactor can be obtained by solving ordinary differential equations (ODEs) that specify the rate of change of each species as a function of the concentrations of the molecules in the system. If membrane transport and electrical potential are to be included in the model, the rate expressions can become more complex but can still be formulated in terms of a system of ODEs. However, when diffusion of molecules within the complex geometry of a cell is also considered, the resultant "reaction/diffusion" system requires the solution of partial differential equations (PDEs) that describe variations in concentration over space and time. We discuss the difference between stochastic and deterministic approaches to solving reaction/diffusion systems to develop an understanding of when each is appropriate. We then briefly review existing tools for analyzing biochemical systems at the ODE level. Finally, three software systems for the construction and analysis of spatial models will be described, *StochSim, MCell*, and *Virtual Cell. Virtual Cell* was developed in our lab, and we describe it in the most detail including examples of both continuous and stochastic spatial models of cell biological processes that have been studied.

STOCHASTIC AND DETERMINISTIC DESCRIPTIONS OF CELLULAR PROCESSES

A continuous approach to spatially resolved biochemical systems based on reaction-diffusion PDEs provides a deterministic description in terms of average species concentration. This description is accurate and effective so long as the number of molecules in a system is macroscopically large. In this case, thermal stochastic fluctuations around average values are relatively small and can therefore be ignored. The complexity of most realistic models of cellular processes would require that the equations be solved numerically, so both a spatial domain and a time interval have to be sampled, and the equations should be correspondingly discretized. The resultant linear algebraic system is then solved by employing effective linear solvers. The most common approaches in engineering disciplines can be usually categorized as finite difference (37) or finite element methods (44). However, these approaches are difficult to implement within an automated software tool designed to solve problems on cells with arbitrary geometries.

The finite volume method, developed originally for problems in heat transfer (27), is especially well-suited to simulations in cell biological systems (31, 33). It is closely related to finite difference methods but allows for good control of boundary conditions and surface profile assumptions while preserving the conservative nature of the equations. Most importantly, the finite volume formalism accommodates the heterogeneous spatial organization of cellular compartments. As implemented in the *Virtual Cell*, the simulation geometry is composed of uniformly sampled rectangular volume elements. Piecewise linear interpolation functions are used to interpret the values of molecular concentrations and electric potentials between element centers. Within such elements, the rate of change of the concentration of a given molecular species is simply the sum of fluxes entering the volume element from its adjacent neighbors plus the rate of production of the given species via reactions. Appropriate jump boundary conditions are implemented at boundaries between dissimilar compartments (i.e., membrane transport conditions). The numerical formulation involves integrating the equations in time over each volume element using appropriate interpolation profiles and boundary conditions. The solution of each integration relates a small neighborhood of sample values over space and time. In choosing a solution method for the resulting system of algebraic equations, numerical stability, given the physically appropriate constraints associated with physiological models, must be considered. Accordingly, diffusion is treated implicitly, i.e. concentrations satisfy the system of simultaneous linear equations evaluated at the next time point to maintain stability. The iterative method originally chosen for solving the linear algebraic system builds on effective routines used for one-dimensional systems and is relatively easy to implement. However, there are some significant drawbacks, such as the need to solve for all the elements in the rectangular domain even if only some compartments are of interest. Moreover, in order for the method to converge in a reasonably small number of iterations, the ratio $\lambda = D\Delta t/\Delta x^2$, where D is the diffusion coefficient, Δt is the

time step, and Δx is the spatial step, must be kept small. This imposes severe restrictions on the time step if high-spatial resolution is required and creates significant problems for running realistic 3D simulations. These issues were resolved, at least partially, by using a different linear solver. Linear solvers based on Krylov space approximations, such as the conjugate gradient method, in conjunction with a preconditioner (an operator that approximates the inverse of the matrix but can be applied at a low computational cost), become powerful and robust. There are commercial packages that implement a range of Krylov space methods, as well as many of the well-known preconditioners (e.g., PCGPAK, Scientific Computing Associates, New Haven, Connecticut). This has produced significant improvements in computational times and enables the practical application of these methods to much larger systems.

If the number of molecules involved in a process is relatively small, the fluctuations can become important. In this case, the continuous description is no longer sufficient and stochastic effects have to be included in a model. Single-channel ionic currents are one such example. While predictions based on the deterministic Hodgkin-Huxley model (20) are usually good for macroscopic phenomena, because of the nonlinearity of the system, stochastic behavior of ion channels proves to be important in some circumstances even when the number of channels is relatively large (35). Similar issues arise in calcium dynamics where the calcium concentration (calcium sparks, calcium waves) (22) can be significantly affected by the stochastic firing of calcium channels. Stochastic fluctuations of macromolecules are crucial for understanding the dynamics of vesicles and granules driven by competing molecular motors. In the case of a relatively small number of participating particles, a system that would be described deterministically by reaction-diffusion PDEs requires fully stochastic treatment. In this approach, diffusion is described as Brownian random walks of individual particles, and chemical kinetics is simulated as stochastic reaction events. Numerical stochastic simulations in this case are based on pseudo-random-number generation (28). They are often called Monte Carlo simulations (the term, originally introduced by Ulam and von Neumann in the days of the Manhattan Project) since throwing a dice is actually a way to generate a random number. Because Monte Carlo methods are general, relatively simple, and straightforward to implement, they can provide a numerical solution of the original reaction-diffusion system even when stochastic fluctuations are not an issue (see the description of *MCell*, below).

In situations where one subsystem requires discrete stochastic formulation, whereas the other can be treated deterministically, the problem can be described in terms of stochastic differential equations (17). As an example, in the Hodgkin-Huxley model, the membrane voltage is treated as a continuous deterministic variable described through a set of differential equations, whereas the single channel behavior is random. A natural way to introduce stochasticity in the model is to replace open probabilities by the actual numbers of open channels (9, 13). In fact, Hodgkin and Huxley introduced variables in their model to represent the proportion of open gates for various ions. The number of open channels is random and is

governed by a corresponding Markov kinetic model that explicitly incorporates the internal workings of the ion channels. Mathematically, the membrane potential is now described by a stochastic differential equation with a discrete random process. Effects of stochasticity on nonlinear systems with excitable or bistable behavior and numerical approaches to stochastic differential equations are areas of active ongoing research (17, 41).

Numerical solution of stochastic differential equations includes the combination of numerical techniques commonly applied to regular differential equations and Monte Carlo methods employing random-number generators. But especially relevant for computational cell biology, the pioneering work by Gillespie (14, 15) on stochastic models for chemical reactions utilized an elegantly efficient algorithm in which the probabilities of each reaction are calculated from rate constants and numbers of substrate molecules. A stochastic method is used to determine which reaction will occur based on their relative probabilities. The time step is then adjusted to match the particular reaction that occurs. After the reaction is complete, the numbers of substrate molecules are readjusted prior to the next cycle. The Gillespie algorithm has been used extensively to analyze stochastic events in the field of biochemical kinetics. When combined with stable, accurate numerical schemes developed for the conventional differential equations, they can be applied for numerical solution of stochastic differential equations with discrete random processes. This type of approach has been utilized in the *Virtual Cell* to combine the deterministic description of a continuously distributed species (RNA) with the stochastic treatment of discrete particles (RNA granules) (see below).

MODELING SOFTWARE FOR NONSPATIAL BIOCHEMICAL SYSTEMS

This review does not cover the well-established field of molecular structure and dynamics simulations, as this is outside the problem domain we define as cell biology. However, another well-established target for computational approaches that is worthy of mention is neuroscience, where a number of software tools are available to simulate the electrophysiological behavior of single neurons and neuronal networks. Because these tools can treat membrane transport and electrical potential in cells, they are quite relevant to our problem domain. The two most prevalent programs are NEURON (18) and GENESIS (5). Both use cable theory (21) to treat the dynamics of electrical signals in the complex geometries of neurons. This theory solves the equation for membrane potential in a series of connected segments with the overall topology of the neuron. Ideally, each segment is small enough to be treated as a compartment that establishes electrical equilibrium rapidly on the timescale of the overall dynamics that are being modeled; in this way the problem can be reduced to solving ODEs, even for a geometry as complex as a neuron. Even more pertinent to cell biology are the recent efforts at generalizing these modeling tools to treat intracellular signaling networks. Hines & Carnevale (19) have added a model description language, NMODL, to

NEURON that accommodates this larger problem domain. Bhalla has developed a new interface called KINETIKIT that adapts GENESIS for chemical kinetics (http://www.ncbs.res.in/~bhalla/kkit/index.html); this tool was employed in an insightful analysis of the modularity of subsystems within complex signaling networks (4).

Several software tools have been developed from the ground up specifically to build complex biochemical reaction pathways and numerically simulate the time course of the individual molecular species within them. Each tool can translate reaction schemes into the corresponding system of ODEs and contains embedded numerical methods for their solution. GEPASI (24) (http://www.gepasi.org/) is one of the best established of these systems. It has an extensive list of predefined kinetic types that significantly aid in the construction of models. It also offers access to a number of optimization methods for deducing reasonable values of those parameters that are not well constrained from experimental determinations. Jarnac/Scamp (30) (http://members.tripod.co.uk/sauro/biotech.htm) is a scripting language that allows users to build, manipulate, and analyze metabolic pathway models with a syntax that is familiar to biologists and avoids having to deal with differential equations. DBSolve (16) (http://websites.ntl.com/~igor.goryanin/) is noteworthy because it has been designed to interface readily with pathway databases. It also contains mathematical tools such as bifurcation and metabolic control analysis. Berkeley Madonna was developed by Oster & Macey (http://www.berkeley madonna.com/) as a general purpose modeling and analysis tool that allows both graphical and text-based inputs of reaction kinetics; it contains several efficient solvers and provides capabilities for bifurcation and sensitivity analysis.

A different philosophy drives the ECELL Project (42) (http://www.e-cell.org/). This is a computational system for constructing whole-cell models, and a model of a self-sustaining primitive cell, based on a subset of 127 genes from the genome of a mycoplasma, has been completed. The model simulations are based on a series of reaction rules that are designed to rapidly calculate the effects of perturbations to individual components of the system. Models of red blood cells and mitochondria are currently under development. Ultimately, the aim of the project is to develop accurate computational models of complex cells such as cardiac myocytes.

At an early stage of development is another ambitious project, BioSpice, from the laboratory of Adam Arkin (http://www.lbl.gov/~aparkin/). While this laboratory has focused on analyzing prokaryotic genetic circuits (23), BioSpice is ultimately intended as a general purpose modeling framework for both genetic and biochemical networks. The open architecture of BioSpice is also intended to ease the incorporation of new software modules that can expand the capabilities of the system according to the needs of the modeling community. A similar philosophy is behind the development of a new software architecture, JSIM (http://nsr.bioeng.washington.edu/), for a simulation tool that models solute transport and exchange in the cardiovascular system; the aim here is also to produce a sufficiently open architecture so that the software can be easily enhanced and adapted to meet the needs of other modeling communities, including cell biology.

Although we have given references for all these software tools, the best information about the availability and current status of these packages is found on the developers' websites. Also, it is doubtful that this list is exhaustive; it is clear that the collection of tools for computational approaches to cell biology is rapidly growing.

SOFTWARE FOR SPATIAL MODELS: *StochSim,* *MCell,* and *Virtual Cell*

StochSim

The extraordinary efficiency of the Gillespie stochastic kinetics algorithm (14, 15) is achieved by restricting the decision process to selecting which reaction will occur and adjusting the time step accordingly. Focusing exclusively on the reaction avoids consideration of the properties of individual reactive species as discrete entities, which minimizes processing time when the number of reacting species is large. However, processing time increases in proportion to the number of different reactions. Furthermore, the Gillespie approach does not easily accommodate the existence of multiple states of different substrates, which may affect their reactivities, and since individual reactive species are not identified as discrete elements, their states, positions, and velocities within the reaction volume cannot be followed over time.

The *StochSim* program developed by Morton-Firth and Bray (24a–c) to analyze complex stochastic signaling pathways in bacterial chemotaxis addresses the "multi-state" problem by shifting the focus from the reaction to the individual reactive species. In this program individual molecules or molecular complexes are represented as discrete software objects or intracellular automata. The time step is set to accommodate the most rapid reaction in the system. At each time step two separate molecules are sequentially selected at random from the total population. Dummy "pseudo-molecules" are included in the population to simulate uni-molecular reactions. A random-number generator is used to determine if a reaction will occur between the two selected molecules by comparison to a look up table of probabilities of all possible reactions. When a reaction occurs the system is updated according to the stoichiometry of the reaction. Molecules that exist in more than one state are encoded as "multi-state molecules" using a series of binary flags to represent different states of the molecule such as conformation, ligand binding, or covalent modification. The flags can modify the reactivity of the molecule, and reactions can modify the flags associated with a multi-state molecule. If the number of reactions is small and the number of molecules large, *StochSim* may be less efficient than the Gillespie algorithm. However, in systems where molecules can exist in multiple states, *StochSim* is generally faster, with the added advantage of being able to track individual molecules over multiple time steps.

The initial version of *StochSim* (version 1.0) treated the system as a well-mixed solution, omitting spatial heterogeneity. A more recent version (22a) includes a simple two-dimensional molecular lattice where nearest neighbor interactions

can affect the reactivities of molecules in the lattice. Implementation of the lattice was used to test the hypothesis that the remarkable sensitivity and dynamic range of bacterial chemotaxis is achieved through adaptive receptor clustering where ligand-induced changes in the signaling activity of receptors are propagated throughout the cluster by nearest neighbor interactions (5a, 35a). This is a particularly compelling illustration of the importance of including spatial heterogeneity in stochastic modeling of intracellular reactions.

MCell

Another software package designed for realistic 3D simulations of cellular physiology, *MCell*, is described by its authors as a general Monte Carlo simulator of cellular microphysiology (www.MCell.cnl.salk.edu). It is written by T. M. Bartol Jr. and J. R. Stiles based on their initial code specifically tailored for simulating the generation of postsynaptic miniature endplate currents (1, 40). The user interaction with *MCell* is carried out through input files written in a special model description language (MDL) (38), an approach similar to the one implemented in GENESIS (5). MDL allows a user to create different types of diffusing ligand molecules with various initial distributions, define patterns of ligand release, specify arbitrary locations of surfaces representing membranes and their interaction with ligand molecules, define multiple types of ligand-binding sites and arbitrary chemical reaction mechanisms for different ligands and their binding sites, and select the type and format of output data that can be visualized.

The software then parses the MDL input files, creates the corresponding C++ objects, and executes a simulation according to the user instructions. *MCell* utilizes Monte Carlo random walk and chemical reaction algorithms using pseudo-random-number generation. One of *MCell*'s convenient features is checkpointing, which involves stopping and restarting a simulation as many times as desired. At each checkpoint, one or more modifications to MDL files can be made, including changing surface permeability and location (thus modeling moving surfaces), modifying reactants and reaction mechanisms, and varying the output type and format.

The simulation domain in *MCell* is a rectangular box containing arbitrary surfaces. Because Monte Carlo simulations do not require volume sampling, geometry treatment reduces to surface triangulation, which results in a list of polygons. 3D surface reconstruction is a preprocessing step that is not a part of a *MCell* simulation. Third-party software should be used for this purpose. Once a surface is generated, it is necessary to edit properties of the triangles that comprise the surface, e.g., add particular types of binding sites at different densities, or specify different permeability for different ligands. The automatic tools that would simplify and accelerate this step are under development.

As the number of participating objects grows, the Monte Carlo algorithms become slow. To speed up simulations, *MCell* is optimized by using 3D spatial partitioning that makes computing speed virtually independent of microdomain geometric complexity. Running parallel computations, another way to speed up

Monte Carlo simulations, is also being pursued in *MCell*. Although currently the successful applications of *MCell* are limited to microphysiology of synaptic transmission (39), other areas of possible *MCell* application include statistical chemistry, diffusion theory, single-channel simulation and data analysis, noise analysis, and Markov processes.

Virtual Cell

THE MODELING PROCESS WITHIN THE *VIRTUAL CELL* ENVIRONMENT   The *Virtual Cell* is a software environment that is being specifically developed to enable the use of modeling as an aid to the design and interpretation of experiments in cell biology (31–34). It can be accessed and run over the Internet from within a web browser (www.nrcam.uchc.edu). It is designed for biologists with little training in physics and math, as well as for experienced mathematical modelers. It achieves this by providing two separate workspaces for construction of models that are designed with the needs of each community in mind. The BioModel workspace abstracts the model components through inputs specifying compartmental topologies, molecular species and their location, kinetic expressions for the reactions and membrane fluxes, and the geometry. The Math workspace allows for the input of a model with a mathematics description language (VCMDL). Importantly, models developed through the BioModel workspace are used to generate a VCMDL version of the model that, if desired, may be modified or refined within the Math workspace; this can facilitate collaboration between biologists and modelers.

Simulations are performed and results are analyzed and visualized with tools that are common to both workspaces. Simulations of both nonspatial (i.e., ODEs) and spatial (PDEs) models can be performed. For nonspatial models, compartments are assigned appropriate volume fractions relative to their parents in the model topology and surface-to-volume ratios for the proper treatment of membrane fluxes. In spatial models, the segmented regions within a 1D, 2D, or 3D image are connected to the corresponding compartments in the topology. The geometry is prepared for a model in a separate Geometry workspace and can come from a segmented experimental image or can be defined analytically. Systems of ODEs are solved numerically with a choice of several solvers including variable time step stiff solvers. PDEs are solved via the finite volume method (27, 37) adapted for the inclusion of membrane transport processes as well as automated pseudo-steady-state approximations for fast reactions (31, 36). These are all implemented in an extensive C++ library that also includes software for stochastic simulations of particle motion and the reaction of individual molecules with continuously distributed species. At present, however, only deterministic models are fully treatable through the web-based interface.

Figure 1 schematizes the way in which the modeling process is structured within the *Virtual Cell*. This hierarchical structure emphasizes a general

physiology definition, the BioModel, that specifies the topology of the system, the identities and locations of molecular species, and reactions and membrane transport kinetics. The BioModel can then have several Applications that each specify a particular geometry, boundary conditions, default initial concentrations and parameter values, and whether any of the reactions are sufficiently fast to permit a pseudo-steady-state approximation. Also at the Application level, individual reactions can be disabled as an aid in determining the proper initial conditions for a prestimulus stable state. An application of a BioModel is sufficient to completely describe the governing mathematics of the model, and as noted above, a VCMDL file is generated at this point. The *Virtual Cell* is designed to maintain a separation between this mathematical description, generated either via a BioModel or a MathModel, and the details of how the simulations are implemented. As shown in Figure 1, several simulations can be spawned from a given Application. The simulation specifications include the choice of solver, time step, and mesh size for spatial simulations, and overrides of the default initial conditions or parameter values. A local sensitivity analysis service is also available at the simulation level to aid in parameter estimation and to determine which features of the model are most critical in determining its overall behavior.

The *Virtual Cell* software displays spatial and nonspatial simulation solutions for the variables over time. The spatial data viewer displays a single plane section of a 3D data set and can sample the solution along an arbitrary curve (piecewise linear or Bezier spline) or at a set of points. Membranes are displayed as curves superimposed on the volume mesh, and membrane variables are displayed along these curves. The nonspatial data viewer plots any number of variables over time on the same plot. All plot windows support a tabular display that allows the user to copy the data into any spreadsheet program. A completely integrated data export service provides for data retrieval in a number of formats (e.g., comma-separated value, gif images, Apple QuickTime movies, animated gif movies) and data reduction schemes (subset of variables, time, and space including data sampling at selected points and along selected curves). Because the model can be mapped to a geometry acquired directly from the microscope, many of the same image analysis tools used to analyze experiments can be applied directly to simulation results. A poor correspondence between simulation and the experiment indicates that the model must be either incomplete or incorrect; indeed such negative results are highly informative because they can suggest how the hypotheses underlying the model need to be modified to accommodate the experiments. When the simulations are consistent with experiment, the ability to visualize the behavior of experimentally inaccessible molecular species often provides important new insights and suggests new experiments to further test the model validity. To illustrate some of these concepts, we now summarize the results of two studies that utilized the *Virtual Cell* system. The second of these takes advantage of stochastic modeling capabilities included in the *Virtual Cell* C++ library but have not yet been migrated to the web-based user interface.

DETERMINISTIC *VIRTUAL CELL* MODEL OF CALCIUM DYNAMICS IN A NEURONAL
CELL   The *Virtual Cell* software is a valuable tool for formulating and testing
hypotheses on the behavior of complex intracellular reaction/diffusion systems. A
key to the successful application of this kind of modeling is that there be suf-
ficient data to formulate reasonable quantitative hypotheses and that there be
appropriate experimental methods available to validate or disprove the predic-
tions of a model. The study of intracellular calcium signals fulfills these require-
ments exceptionally. This is because of the ready availability of fluorescence
microscopy–based methods for following the spatial and temporal patterns of
calcium changes in living cells [e.g., (26, 43)]. Arguably, it is the availability of
these methods that has led to an explosion of interest in calcium dynamics and
that, in turn, has led to further studies of the biochemical and electrophysiological
events that lie upstream and downstream from the calcium signal. Ample data are
often available to begin the development of a *Virtual Cell* model with few addi-
tional experiments. On the other hand, the molecules and mechanisms that can
be involved in the control of intracellular calcium are sufficiently numerous and
complex so as to make it difficult to understand experimentally observed behavior
without the use of modeling.

In particular, inositol-1,4,5-triphosphate (InsP$_3$)-mediated calcium release from
the endoplasmic reticulum (ER) is a common mechanism for receptor-mediated
signaling in many cell types (2, 3, 29). We have used the *Virtual Cell* to help analyze
and interpret experimental data on the details of the calcium release process in dif-
ferentiated N1E-115 neuroblastoma cells (11, 12). In addition to providing insights
on how morphology controls spatiotemporal patterns of InsP3 signaling within the
cell, the needs of the calcium modeling have inspired significant improvements to
the *Virtual Cell* such as a generalized automated pseudo-steady-state treatment for
fast reactions (36) and a flux correction algorithm for the "staircase" membranes
in the finite volume method (33).

In our study, the neuromodulator bradykinin (BK) was the external stimulus that
set off InsP$_3$ production at the inner surface of the plasma membrane. Combining
calcium imaging, quantitative uncaging of microinjected InsP$_3$, and simulations
from the *Virtual Cell* led to the conclusion that BK triggers a buildup of InsP$_3$ in
the neurite at a rate and to an extent much greater than in the soma (11, 12). The
proximal segment of the neurite is the critical region for a response to a BK stimulus
and is necessary and sufficient to initiate and propagate the calcium signal to other
regions of the cell. The high surface-to-volume ratio in the neurite intensifies
the InsP$_3$ signal in this region. A high density of ER calcium stores in the soma
(predicted by the simulations and then confirmed by 3D immunofluorescence)
balances the rapid rise of [InsP$_3$]$_{cyt}$ in the neurite to explain the contrasting results
for both global and focal stimulations of these cells.

Figure 2 shows the results of an experiment and a simulation for the response
of a cell to uniform global application of BK. The experimental data were col-
lected on a fast digital-imaging microscope using the fluorescence from the in-
dicator fura-2 to record the spatiotemporal changes in [Ca^{2+}]$_{cyt}$. Because the

simulation was based on the cell geometry of the experiment, a direct comparison of the observed and predicted $[Ca^{2+}]_{cyt}$ dynamics can be made (first two columns of Figure 2). It must be emphasized that this close match between experiment and simulation was achieved only after many iterations between modeling and data collection, which helped to uncover previously unknown or misunderstood details of the components of the system (such as the high density of stores in the soma, as mentioned above). Also in Figure 2 are simulation results for the spatiotemporal pattern of InsP$_3$ and the open state of the InsP$_3$ receptor calcium channel in the ER. This ability to visualize molecular species that are inaccessible experimentally is one of the most valuable benefits of computational modeling.

STOCHASTIC MODELS FOR RNA TRAFFICKING IN THE *VIRTUAL CELL*   RNA trafficking targets expression of specific proteins to particular subcellular compartments and minimizes ectopic expression elsewhere in the cell. Studies of RNA trafficking in oligodendrocytes reveal that RNAs are assembled into trafficking intermediates termed granules (8), each containing multiple (approximately 30) RNA molecules (25), associated RNA-binding proteins, components of the translation machinery, and molecular motors (conventional kinesin and cytoplasmic dynein). RNAs with similar trafficking pathways are co-assembled into the same granules, whereas RNAs with different trafficking pathways are segregated into different granules. Conventional kinesin moves granules toward the plus ends of microtubules while cytoplasmic dynein moves granules toward the minus ends. The balance of power between kinesin and dynein activities in individual granules determines their direction and rate of movement along microtubules (6). Time lapse analysis of granule dynamics in living cells reveals rapid back-and-forth vibration along the axis of the microtubule, which is believed to reflect stochastic fluctuations in motor activities in individual granules. Biased activity of kinesin over dynein results in translocation of specific RNA granules to the periphery of the cell.

A stochastic model for granule assembly has been developed in the *Virtual Cell*. RNA is treated as a disperse chemical species that diffuses from a source in the nucleus, through the nuclear envelope into the cytoplasm where RNA molecules are captured, to core granules treated as discrete species moving stochastically throughout the cytoplasmic volume and undergoing elastic interactions with the cell membrane and nuclear envelope. The simulation is constrained within a 2D profile of an actual oligodendrocyte extracted from a confocal microscopic image data. The parameters used in the simulation—RNA concentrations in the nucleus and cytoplasm, number of granules in the cytoplasm, rate of movement of granules, rate of diffusion of RNA, rate of export of RNA from the nucleus, and number of RNA molecules per granule—are either experimentally determined or estimated based on observations in other systems. The simulation integrates chemical reactions and diffusion of a disperse species (RNA) with stochastic properties of mobile discrete species (granules) constrained by elastic interactions with immobile structures (membranes) within the cell. A representative image from the simulation is shown in Figure 3. The entire simulation is available

at http://www.nrcam.uchc.edu/rna_traffick_dir/rna_traffick.html. The kinetics of granule assembly in the *Virtual Cell* recapitulate the kinetics observed experimentally in microinjected oligodendrocytes. As the simulation proceeds, granules tend to accumulate in varicosities and diverticuli within the cell in a pattern reminiscent of the distribution of granules observed experimentally in microinjected oligodendrocytes. The unusual fractal geometry of the reaction volume, which is a realistic representation of the intracellular space in an oligodendrocyte, appears to constrain the movement of granules in ways that could not be predicted based on simulations in a symmetric reaction volume. This example illustrates the importance of using actual image data to define the geometry of the reaction volume.

A stochastic model for granule dynamics on microtubules has also been developed using the *Virtual Cell* (7). Microtubules are represented as one-dimensional, oriented contours of arbitrarily defined shapes decomposed into discrete contour elements. Granules are represented as zero-dimensional points, which can be captured to microtubules if one or more contour element is within a specified capture radius, corresponding to the actual size (approximately 0.5μm) of an RNA granule in a living cell. Each granule contains multiple plus end and minus end motors, each of which can be in one of three states relative to the microtubule (U, unbound; I, bound inactive; and A, bound active). Granule capture is a stochastic event mediated by binding of a specific motor molecule to a specific contour element. The type of motor that binds to the contour at each time step depends on the population of motors of each type associated with the granule and the corresponding motor on-rates multiplied by Δt. Monte Carlo methods are used to determine whether the granule will be captured within each time step, and if captured, which motor will bind to the contour element.

Once captured to a microtubule, granule movement along the axis of the microtubule is determined by stochastic state transitions for each motor associated with the granule. Because there are two types of motors in each granule, there are eight single-motor state transition rates: $k_{ai}^{\pm}$, $k_{ia}^{\pm}$, $k_{ui}^{\pm}$, $k_{iu}^{\pm}$. An important simplifying assumption is that the granule can have no more than one active motor at any time. The instantaneous behavior of the granule then depends on the type of the active motor (if there is one) and the state of the "cloud" of inactive motors. The discrete-valued variable s determines the state of the granule depending on the type of the active motor. This variable can take on one of the three values: $s = 0$, no active motors, the granule does not move; $s = 1$, a minus motor is active, the granule is moving toward the minus end with the velocity v_-; and $s = 2$, a plus motor is active, the granule is moving toward the plus end with the velocity v_+. Because the numbers of plus and minus motors in the cloud, $N_{\pm}$, are large, the state of the cloud can be described by two continuous variables, $P_{\pm}$, representing the probabilities for the plus and minus inactive motors to be in a bound state, so that the instantaneous number of inactive bound plus and minus motors are $N_+ P_+$ and $N_- P_-$. The discrete and continuous variables are separated in the sense that the dynamics of the discrete variable s is determined by the rates of the A $\leftrightarrow$ I transitions, whereas the dynamics of the continuous variables $P_{\pm}$ depend on the rates of the I $\leftrightarrow$ U transitions. However, the variables are also interdependent

because the active motor exerts strain on inactive bound motors, making it easier for them to detach from the microtubule, and the inactive bound motors exert drag on the active motor, creating an additional load on it. Granule dynamics in the *Virtual Cell* recapitulate experimentally observed granule dynamics. Moreover, because the positions and the velocities of individual granules and the states of motors within each granule can both be tracked over time, it is possible to correlate particular aspects of granule dynamics with specific state transitions of the motors within the granule leading to experimentally testable predictions concerning the way stochastic interactions of different molecular motors result in directional intracellular trafficking of RNA granules.

The *Virtual Cell* program has several important advantages for stochastic modeling in eukaryotic cells. First, realistic image-based cell geometries are used to define intracellular reaction volumes, which constrain the stochastic behavior of intracellular reactants in unexpected ways. Second, definitions of reactive species can include multiple states described as either discrete parameters or continuous variables, which provide extraordinary contextual richness and behavioral versatility. Third, dynamic transformation and translocation of multiple individual reactive species can be tracked over time, facilitating integration of spatially heterogeneous stochastic models with simultaneous deterministic reaction/diffusion models. A major future challenge for the *Virtual Cell* will be to integrate dynamic shape changes in the reaction volume within the powerful and flexible stochastic modeling platform already developed. If this can be accomplished, the holy grail of stochastic modeling of cell motility may be attainable using the *Virtual Cell*.

FUTURE CHALLENGES AND PROSPECTS

Future development and enhancement of computational tools for cell biology will provide opportunities to develop larger and more realistic models with a wider range of modeling capabilities. One advantage of using computer tools in cell modeling is the potential ability to deal with complex models. The only prudent way to construct complex models is to assemble them from smaller submodels (modeling cassettes) that are well understood and tested. To make sure that the model components are compatible, a careful analysis of assumptions is necessary. Ideally, the results of this analysis should be documented in some formal way so that the compatibility test could be made automatic. The analysis of the most successful mathematical models that are frequently used in cell modeling with respect to all the assumptions, both explicit and implicit, made in constructing those models is a first step in this direction.

As we move toward modeling complex systems on realistic 3D structures, computational efficiency of numerical algorithms becomes a critical issue. Numerical methods should be fully automated, reasonably accurate, stable, and fast. Because of variability in biological systems, it is particularly important in biological applications to be able to rerun simulations a number of times with varying parameters. Therefore, even 3D simulations of complex systems must complete within

a relatively short time. The numerical algorithms currently used in the *Virtual Cell* are applicable to a wide range of problems, particularly now that we have developed and implemented the automatic numerical approach to fast reactions in reaction/diffusion systems (36). However, we use a fixed time step spatial solver and leave it to the user to specify the time step. In combination with currently used explicit treatment of reactions and membrane fluxes, this may lead to numerical instability when the chosen time step is larger than characteristic times of explicitly treated processes. Also, it is currently the user's responsibility to distinguish the fast subsystem, although, in principle, it is possible to automatically detect the fast subsystems based on the values of reaction constants, initial concentrations, and other simulation parameters. Usually, the most time-consuming part of a simulation is solving the large linear algebraic system resulting from the discretization of governing PDEs. The optimal choice of a linear solver and the parameters associated with it is therefore critical to the efficiency and robustness of the overall package. To improve stability, accuracy, and overall efficiency of numerical simulations, the issues of reaction stiffness in the PDEs, more accurate representation of irregular boundaries, and choice of effective linear solvers need to be addressed.

Although software systems such as *StochSim, MCell*, and *Virtual Cell* provide modeling capabilities to cell biologists and biophysicists, which would previously have required highly specialized training in numerical methods and mathematical physics, their problem domains do not encompass all the areas of interest in the broad field of cell biology. In particular, the *Virtual Cell* features currently accessible from the user interface are limited to deterministic reaction/diffusion systems mapped to arbitrary, but fixed, geometries including arbitrary fluxes and reactions associated with membranes. Although a wide range of cellular processes falls into this category, additional features are being developed, including modeling membrane potential, stochastic processes, lateral diffusion in membranes, and one-dimensional structures such as microtubules and microfilaments. Currently some of these are only accessible through custom executables that call on these features through the C++ library. It remains to make them accessible through the user interface. Also needed are computational tools to treat cell structural dynamics to enable the construction of models of such processes as cell migration or mitosis. Such tools will be especially challenging because they will require new formulations for the physics underlying cell structural changes; in contrast to the firm theoretical foundation for reaction/diffusion processes, this physics has not yet been firmly established.

Visit the Annual Reviews home page at www.annualreviews.org

LITERATURE CITED

1. Bartol TM Jr, Land BR, Salpeter EE, Salpeter MM. 1991. Monte Carlo simulation of miniature endplate current generation in the vertebrate neuromuscular junction. *Biophys. J.* 59:1290–307

2. Berridge MJ. 1993. Inositol trisphosphate and calcium signalling. *Nature* 361:315–25

3. Berridge MJ. 1998. Neuronal calcium signaling. *Neuron* 21:13–26

4. Bhalla US, Iyengar R. 1999. Emergent properties of networks of biological signaling pathways. *Science* 283:381–87

5. Bower JM, Beeman D. 1998. *The Book of GENESIS: Exploring Realistic Neural Models with the General Neural Simulation System*. New York: Springer. 2nd ed.

5a. Bray D, Levin MD, Morton-Firth CJ. 1998. Receptor clustering as a cellular mechanism to control sensitivity. *Nature* 393:85–88

6. Carson JH, Cui H, Barbarese E. 2001. The balance of power in RNA trafficking. *Curr. Opin. Neurobiol.* 11:558–63

7. Carson JH, Cui H, Krueger W, Slepchenko B, Brumwell B, et al. 2001. RNA trafficking in oligodendrocytes. In *Cell Polarity and Subcellular Localization*, ed. D Richter, pp. 69–81. Berlin: Springer

8. Carson JH, Kwon S, Barbarese E. 1998. RNA trafficking in myelinating cells. *Curr. Opin. Neurobiol.* 8:607–12

9. Clay J, Defelice L. 1983. Relationship between membrane excitability and single channel open-close kinetics. *Biophys. J.* 42:151–57

10. Deleted in proof

11. Fink CC, Slepchenko B, Moraru II, Schaff J, Watras J, et al. 1999. Morphological control of inositol-1,4,5-trisphosphate-dependent signals. *J. Cell Biol.* 147:929–35

12. Fink CC, Slepchenko B, Moraru II, Watras J, Schaff J, et al. 2000. An image-based model of calcium waves in differentiated neuroblastoma cells. *Biophys. J.* 79:163–83

13. Fitzhugh R. 1965. A kinetic model of the conductance changes in nerve membrane. *J. Cell. Comp. Physiol.* 66:111–18

14. Gillespie DT. 1977. Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.* 81:2340–61

15. Gillespie DT. 2001. Approximate accelerated stochastic simulation of chemically reacting systems. *J. Chem. Phys.* 115:1715–33

16. Goryanin I, Hodgman TC, Selkov E. 1999. Mathematical simulation and analysis of cellular metabolism and regulation. *Bioinformatics* 15:749–58

17. Higham DJ. 2001. An algorithmic introduction to numerical simulation of stochastic differential equations. *SIAM Rev.* 43:525–46

18. Hines ML, Carnevale NT. 1997. The NEURON simulation environment. *Neural Comput.* 9:1179–209

19. Hines ML, Carnevale NT. 2000. Expanding NEURON's repertoire of mechanisms with NMODL. *Neural Comput.* 12:995–1007

20. Hodgkin AL, Huxley AF. 1952. A quantitative description of membrane current and its application to conduction and excitation in nerve. *J. Physiol.* 117:500–44

21. Jack JJB, Noble D, Tsien RW. 1975. *Electric Current Flow in Excitable Cells*. Oxford: Clarendon. 502 pp.

22. Keizer J, Smith GD, Ponce-Dawson S, Pearson JE. 1998. Saltatory propagation of Ca^{2+} waves by Ca^{2+} sparks. *Biophys. J.* 75:595–600

22a. Le Novere N, Shimizu TS. 2001. STOCHSIM: modelling of stochastic biomolecular processes. *Bioinformatics* 17:575–76

23. McAdams HH, Arkin A. 1998. Simulation of prokaryotic genetic circuits. *Annu. Rev. Biophys. Biomol. Struct.* 27:199–224

24. Mendes P, Kell D. 1998. Non-linear optimization of biochemical pathways: applications to metabolic engineering and parameter estimation. *Bioinformatics* 14:869–83

24a. Morton-Firth CJ. 1998. *Stochastic simulation of cell signaling pathways*. PhD thesis. Cambridge Univ. 263 pp.

24b. Morton-Firth CJ, Bray D. 1998. Predicting temporal fluctuations in an intracellular signalling pathway. *J. Theor. Biol.* 192:117–28

24c. Morton-Firth CJ, Shimizu TS, Bray D. 1999. A free-energy-based stochastic simulation of the Tar receptor complex. *J. Mol. Biol.* 286:1059–74

25. Mouland AJ, Xu H, Cui H, Krueger W, Munro TP, et al. 2001. RNA trafficking signals in human immunodeficiency virus type 1. *Mol. Cell. Biol.* 21:2133–43

26. Nuccitelli R, ed. 1994. *A Practical Guide to the Study of Calcium in Living Cells*, Vol. 40. *Methods in Cell Biology*. San Diego: Academic Press. 342 pp.

27. Patankar SV. 1980. *Numerical Heat Transfer and Fluid Flow*. Washington, DC: Taylor & Francis. 197 pp.

28. Press WH, Tuekolsky SA, Vetterling WT, Flannery BP. 1999. *Numerical Recipes in C* (*The Art of Scientific Computing*). Cambridge, UK: Cambridge Univ. Press. 2nd ed.

29. Putney JW Jr, Bird GS. 1993. The inositol phosphate-calcium signaling system in nonexcitable cells. *Endocr. Rev.* 14:610–31

30. Sauro HM. 1993. SCAMP: a general-purpose simulator and metabolic control analysis program. *Comput. Appl. Biosci.* 9:441–50

31. Schaff J, Fink CC, Slepchenko B, Carson JH, Loew LM. 1997. A general computational framework for modeling cellular structure and function. *Biophys. J.* 73:1135–46

32. Schaff J, Loew LM. 1999. The virtual cell. In *Biocomputing: Proceedings of the 1999 Pacific Symposium*, ed. RB Altman, AK Dunker, L Hunter, TE Klein, K Lauderdale, pp. 228–39. Singapore: World Sci.

33. Schaff JC, Slepchenko BM, Choi Y, Wagner JM, Resasco D, et al. 2001. Analysis of non-linear dynamics on arbitrary geometries with the Virtual Cell. *Chaos* 11:115–31

34. Schaff JC, Slepchenko BM, Loew LM. 2000. Physiological modeling with the Virtual Cell framework. *Methods Enzymol.* 321:1–23

35. Schneidman E, Freedman B, Segev I. 1998. Ion channel stochasticity may be critical in determining the reliability and precision of spike timing. *Neural Comput.* 10:1679–703

35a. Shimizu TS, Le Novere N, Levin MD, Beavil AJ, Sutton BJ, et al. 2000. Molecular model of a lattice of signalling proteins involved in bacterial chemotaxis. *Nat. Cell. Biol.* 2:792–96

36. Slepchenko BM, Schaff JC, Choi YS. 2000. Numerical approach to fast reaction-diffusion systems: application to buffered calcium waves in bistable models. *J. Comp. Phys.* 162:186–218

37. Smith GD. 1985. *Numerical Solution of Partial Differential Equations: Finite Difference Methods*. Oxf. Appl. Math. Comput. Sci. Ser. Oxford: Clarendon. 3rd ed.

38. Stiles JR, Bartol TM, Salpeter EE, Salpeter MM. 1998. Monte Carlo simulation of neurotransmitter release using MCell, a general simulator of cellular physiological processes. In *Computational Neuroscience*, ed. J Bower, pp. 279–84. New York: Plenum

39. Stiles JR, Kovyazina IV, Salpeter EE, Salpeter MM. 1999. The temperature sensitivity of miniature endplate currents is mostly governed by channel gating: evidence from optimized recordings and Monte Carlo simulations. *Biophys. J.* 77:1177–87

40. Stiles JR, Van HD, Bartol TM Jr, 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. USA* 93:5747–52

41. Sulis W, Trofimova I, eds. 2001. *Nonlinear Dynamics in the Life and Social Sciences*, Vol. 320. NATO Science Series: Life Sciences. Amsterdam: IOS

42. Tomita M, Hashimoto K, Takahashi K, Shimizu TS, Matsuzaki Y, et al. 1999. E-CELL: software environment for whole-cell simulation. *Bioinformatics* 15:72–84

43. Williams DA, Bowser DN, Petrou S. 1999. Confocal Ca^{2+} imaging of organelles, cells, tissues, and organs. *Methods Enzymol.* 307:441–69

44. Zienkiewicz OC, Taylor RL. 2000. *Finite Element Method: Vol. 1, The Basis.* London: Butterworth-Heinemann. 712 pp. 5th ed.
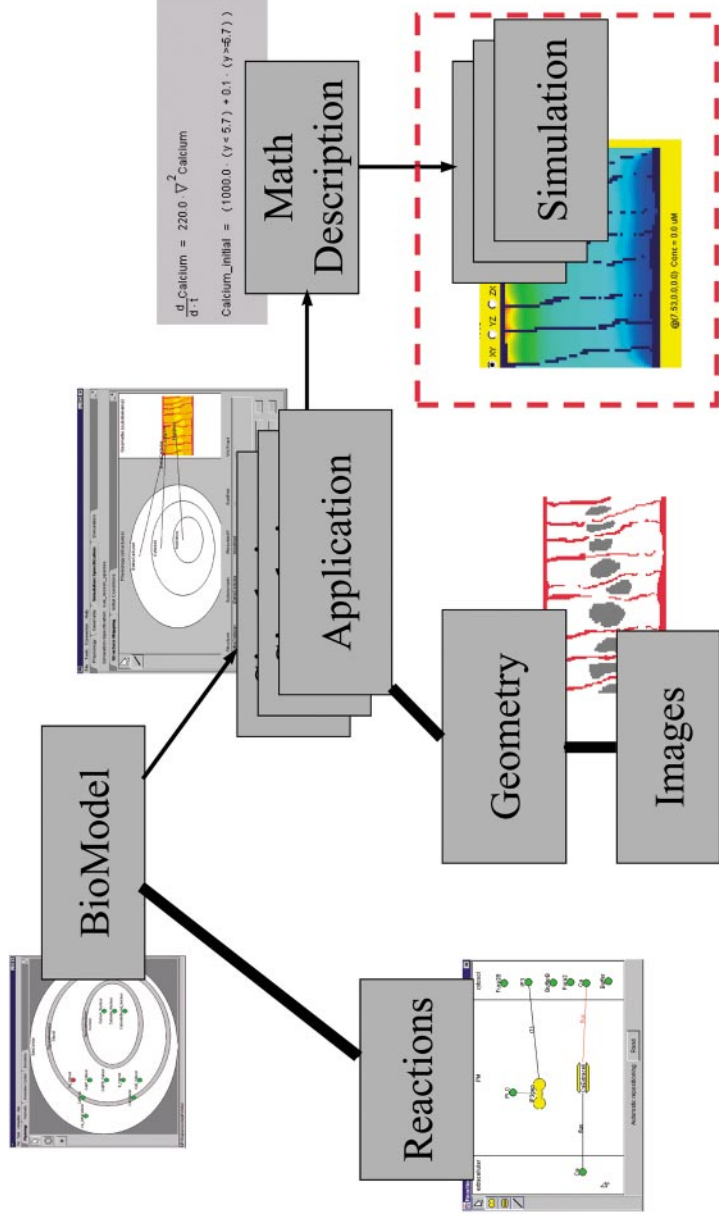
Figure 1 The modeling process within the *Virtual Cell* BioModel workspace. Each component of the overall model is labeled over a screen snapshot of the corresponding section of the user interface. The individual components and the flow of model and simulation specification are described in the text. A model entered through the MathModel workspace would have the geometry directly linked to the Math Description, but could still spawn multiple simulations.
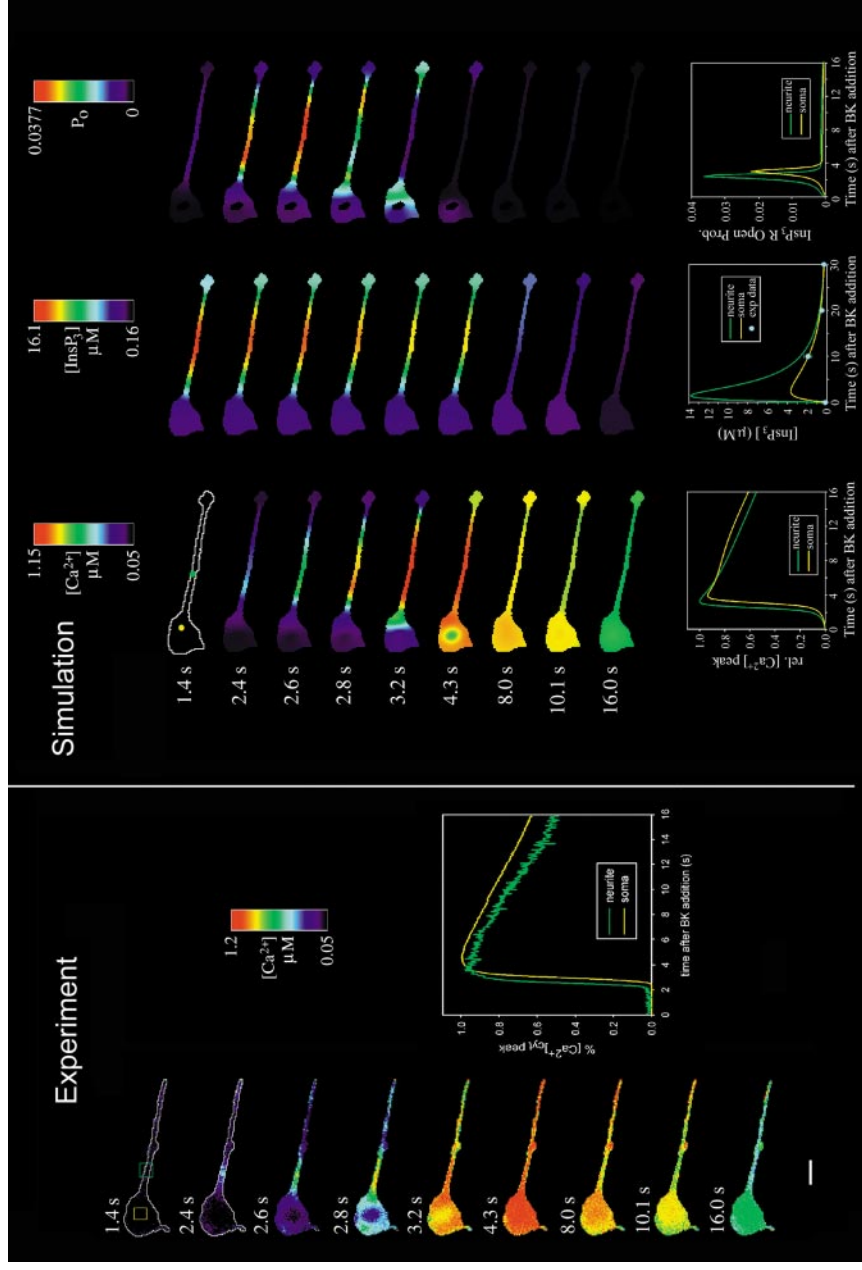
(See legend on next page)

Figure 2 (See figure on previous page) Experiment and simulation of calcium dynamics following BK stimulation of a N1E-115 neuroblastoma cell. A 250 nM solution of BK was applied uniformly to the cell at time 0, and the $[Ca^{2+}]_{cyt}$ is monitored with fura-2 to produce the experimental record shown in the left column. The data were collected through a microscope using a cooled ccd camera at 15 frames/sec. Representative frames are shown, and the change in calcium in the neurite (*green box*) and soma (*yellow box*) are plotted in the inset. The *Virtual Cell* simulation shown in the next column provides a good match to the experiment. The third and fourth columns display the simulation results for $[InsP_3]_{cyt}$ and P_o, the open probability of the $InsP_3$-sensitive calcium channel in the ER membrane. Details on the model components have been published (11, 12), and this figure has been adapted from that work (12).
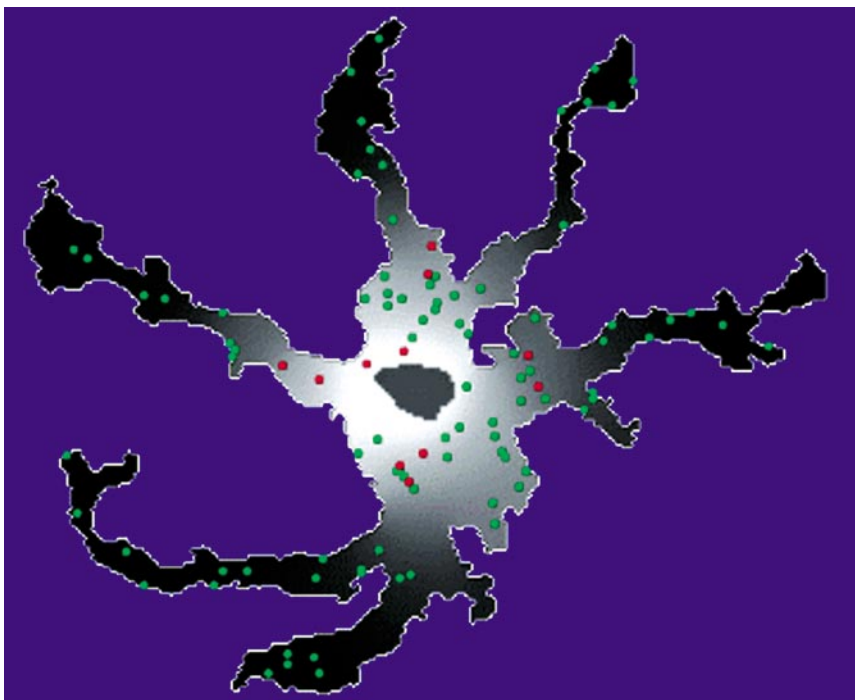
Figure 3 Stochastic modeling of granule assembly in the *Virtual Cell*. The external contour of an oligodendrocyte in culture was extracted from a confocal micrograph of an oligodendrocyte injected with fluorescent dextran to visualize the cytoplasmic volume. Dextran was size-excluded from the interior of the nucleus (shown as *dark gray*). RNA was set at a constant (high) concentration in the nucleus and could diffuse through the nuclear envelope as a disperse species creating a concentration gradient within the cytoplasm (shown in *gray scale*). Individual granules are represented as discrete particles that walk randomly throughout the cytoplasm and undergo elastic collisions at the plasma membrane and nuclear envelope. Core granules (lacking RNA) are shown in *green*. Diffusing RNA molecules are captured to individual granules. When the number of captured RNA molecules reaches a threshold, the granule is shown in *red*. Granules tend to get trapped in varicosities and diverticuli in the distal processes.

CONTENTS

ERRATA
  An online log of corrections to *Annual Review of Biophysics
  and Biomolecular Structure* chapters may be found
  at http://biophys.annualreviews.org/errata.shtml