CHAPTER 25

Methods for Simulating the Dynamics of Complex Biological Processes

Maria J. Schilstra,[*] Stephen R. Martin,[†] and Sarah M. Keating[*]

[*]Biological and Neural Computation Group
Science and Technology Research Institute
University of Hertfordshire
College Lane, Hatfield AL10 9AB
United Kingdom

[†]Division of Physical Biochemistry
MRC National Institute for Medical Research
The Ridgeway
Mill Hill, London NW7 1AA
United Kingdom

METHODS IN CELL BIOLOGY, VOL. 84

0091-679X/08 $35.00
DOI: 10.1016/S0091-679X(07)84025-8

Abstract

In this chapter, we provide the basic information required to understand the central concepts in the modeling and simulation of complex biochemical processes. We underline the fact that most biochemical processes involve sequences of interactions between distinct entities (molecules, molecular assemblies), and also stress that models must adhere to the laws of thermodynamics. Therefore, we discuss the principles of mass-action reaction kinetics, the dynamics of equilibrium and steady state, and enzyme kinetics, and explain how to assess transition probabilities and reactant lifetime distributions for first-order reactions. Stochastic simulation of reaction systems in well-stirred containers is introduced using a relatively simple, phenomenological model of microtubule dynamic instability *in vitro*. We demonstrate that deterministic simulation [by numerical integration of coupled ordinary differential equations (ODE)] produces trajectories that would be observed if the results of many rounds of stochastic simulation of the same system were averaged. In Section V, we highlight several practical issues with regard to the assessment of parameter values. We draw some attention to the development of a standard format for model storage and exchange, and provide a list of selected software tools that may facilitate the model building process, and can be used to simulate the modeled systems.

I. Introduction

A biological cell is a highly dynamic environment. It takes up energy and uses it to create complex molecules, and to control their interactions, assembly, and destruction. Even structures that were once thought to be relatively stable and inert, such as interphase DNA, or cell walls and membranes, are continuously being remodeled, repaired, and eventually, bit by bit, entirely replaced with new material. All of these processes happen in a highly coordinated fashion. ''Output'' molecules of particular processes function as ''input'' to other processes—acting as fuel, signals, or information, and the input–output chains frequently fork, converge, interlace, and form cycles. Even when the machinery and workings of a particular process or pathway have been studied in great detail, it is usually very difficult, if not impossible, to understand intuitively the timing and interrelationship of the component steps and to make predictions about the ways in which the whole system will react.

To aid in understanding of such complex systems, one may construct ''working models'' in which all components that are thought to be important to the functioning of the whole are represented, and act and interact in a manner that resembles the original. The model can then be made to ''simulate'' the real system. If the behavior of the model resembles that of the real system, one may, at least

temporarily, assume that the model is a reasonable representation of reality, and begin to make predictions by studying the responses of the model to changes in input, or changes in the structure of the model itself. If the predictions from the model are not borne out by the behavior of the real system, it will be necessary to reassess the model, or at least parts of it.

Model building is a somewhat subjective pursuit, and its success depends to some extent on the knowledge, experience, and intuition of the modeler. The models themselves can take many forms, and there are no rules that dictate the use of any particular medium: for example, a very informative working model of the UK economy was once implemented using hydraulic devices (pipes, valves, and pumps) and water (Swade, 1995). Such an analogue computing device could, in principle, very well be used to model metabolic flows in cells, or physiological processes in whole organisms. The advent of systems biology, which is linked to the spectacular advances in high-throughput technology of the last decade, has kindled an interest in the functioning and dynamics of complex biochemical systems among experts in dynamical systems from different fields, such as mathematics and electronic engineering. When modeling techniques from fields, such as electronics, are applied to biochemical systems, they tend be accompanied by their own suggestive jargon (“circuit,” “switch,” “design pattern”). While nontraditional modeling has its place and can certainly lead to new insights, it may also cause confusion as to cause and effect, or actor and role, and lead to erroneous conclusions. The novice modeler is advised to always bear in mind that models of biochemical processes should first and foremost adhere to the laws of thermodynamics, and that role and function are merely labels that do not imply anything more than what is explicitly expressed in the model.

In this chapter we shall, therefore, concentrate on modeling and simulation of biochemical processes in the conventional way in which the models describe the chemical reaction kinetics of the system, are expressed in mathematical terms, and are made to run on digital computers.

II. Rationale

Model building and testing—conjecture and attempted refutation—form the basis of all research in the natural sciences. Biochemical reaction systems are, in general, far too complex to allow predictions about their behavior solely on the basis of mental arithmetic or intuition. The title of this chapter may suggest a focus on large systems with many participants and many interactions, but in fact small systems, encompassing only two or three reactions and as few participants, may already exhibit behavior that is far too complex to be understood intuitively. Larger systems are just that—larger—but the same ground rules apply. We believe that it is of utmost importance that modelers understand what these ground rules are, and try to picture biochemical processes as interactions between individual

molecules or assemblies that are associated with particular kinetic constants and affinities.

Although modeling solely in these terms ignores important questions related to structure and function, it is important to remember that the values of rate constants, which are assessed experimentally on the basis of models built on the premises outlined in this chapter, may give important clues about the molecular characteristics of the reactants. In addition, rate constants also provide information that allows prediction of, for instance, the amount of time individual species will remain as part of a particular complex, which, in turn, may have important consequences for the behavior of the whole system.

III. Modeling

Because this volume is dedicated to techniques and tools that are used to study biochemical processes *in vitro*, we shall focus on the modeling and simulation of reactions in "well-stirred containers." In a well-stirred container, all conditions—temperature, pH, and concentrations—are homogeneous throughout. Cells, of course, do not fall into this category, and modeling of processes that occur *in vivo* may have to take account of gradients and other local conditions. This introduces many more parameters and degrees of freedom into a model, and turns simulation of the system into a much more complicated task (see, for instance, Andrews and Bray, 2004; Kruse and Elf, 2006; Meyers *et al.*, 2006). Local conditions may have a dramatic effect on the overall dynamics of a system; they do not, however, change its basic chemistry.

A. Running Example

In this chapter, we shall illustrate the most important concepts in modeling and simulation using the phenomenon of microtubule dynamic instability as a running example. Microtubules are long, hollow cylinders constructed from the tubulin $\alpha\beta$-heterodimer. The α subunit of the dimer contains nonexchangeably bound GTP; whereas the β subunit can contain GTP (Tu-GTP), GDP (Tu-GDP), or be nucleotide free (Tu). Following addition of Tu-GTP to a growing microtubule, the GTP is hydrolyzed to GDP and, as a result, Tu-GDP is the major component of microtubules.

In 1984, Michison and Kischner first reported that "microtubules *in vitro* coexist in growing and shrinking populations which interconvert rather infrequently," and proposed that "this dynamic instability is a general property of microtubules." Horio and Hotani (1986) were the first to observe experimentally that single microtubules were indeed either in a slowly growing state (slow tubulin dimer addition) or in a rapidly shortening or shrinking state (rapid tubulin dimer loss). Shrinking microtubules were occasionally "rescued," and became growing ones, whereas "catastrophe" sometimes befell growing microtubules, whereupon

they entered the shortening state. State interconversion was infrequent, and appeared to happen randomly. Although the two ends, "plus" and "minus," of the microtubules could be distinguished by their dynamics (plus-ends were much more dynamic than minus-ends), both ends appeared to behave in essentially the same way.

Horio and Hotani were able to determine values for the growth, shrinkage, and interconversion rates, and we shall use these original data to calculate the parameter values required to perform the simulations. Under the conditions used in their study, the plus-ends of shrinking microtubules were found to shorten for an average of 18 sec, before they were rescued, and lost about 2.4 μm in length over that period. Occurrences in which microtubules shrunk to zero length, and disappeared completely, were not counted. Microtubules have 1625 subunits per micrometer, so that 2.4 μm corresponds to 3900 (tubulin dimer) subunits. This means that they were shrinking at a rate of about 220 subunits per second. Growing microtubules continued to grow for an average of approximately three minutes, during which they gained about 1.8 μm: a growth rate of about 16 subunits per second.

These values were, of course, obtained under certain conditions, using a particular tubulin preparation, and similar measurements, probably more accurate, and under much wider sets of conditions, have been carried out over the years. Ideas about the origins and consequences of this behavior have been extensively discussed in the literature. Here, however, we will simply use a model that describes the phenomenon itself—rather than its origin—which reproduces the above observations, and is capable of predicting the overall behavior of microtubules under different conditions.

B. From Cartoons to Dynamic Model Diagrams

Descriptions of biochemical processes are often accompanied by cartoons in which aspects of their organization and behavior are illustrated. Such cartoons may show the molecules that participate in the process, their function, their chemistry, their interactions, the assemblies that they form, the dynamics of these assemblies, the cellular compartments in which the process occurs, and so on. Although cartoons can be very helpful in conveying ideas, they are, by and large, unsuitable as a basis for a quantitative description, not so much because they are not associated with numbers or equations, but mainly because the use of symbols is ambiguous. In Fig. 1A, the concept of microtubule dynamic instability (see Section III.A) is illustrated using a cartoon.

Three different diagram types can be used as the basis for a quantitative dynamic model, and these are also illustrated in Fig. 1. These dynamic model diagrams also contain boxes and arrows, just like cartoons, but here the use of symbols is unequivocal: they have an exact interpretation, and an equivalent mathematical representation. Such diagrams leave out all information that is irrelevant to the modeled dynamics, and therefore seldom show all of the ideas that are expressed in

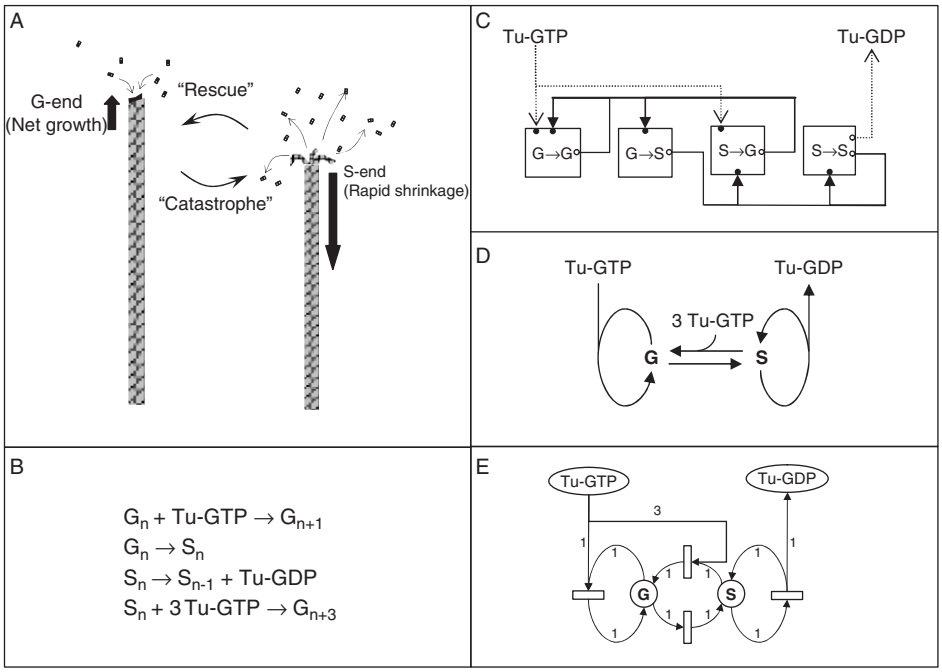


Fig. 1 Diagrams depicting microtubule dynamic instability: (A) cartoon, (B) chemical reaction scheme, (C) data-flow or process diagram, (D) state-transition diagram, (E) Petri-net diagram. The use of boxes, arrows, and other symbols in the cartoon (A) is ambiguous. Textual symbols in the equivalent chemical reaction scheme (B) and state-transition diagram (D) represent chemical or physical states; the arrows represent reactions. The arrows may be associated with rate equations, which express the dependence of the reaction rate on the concentrations of the reaction participants. The boxes in the data-flow diagram (C) represent processes (transformation of input into output); the arrows symbolize data such as concentrations of particular molecules or complexes. The small circles at the edges of the boxes indicate whether the data are input (filled circles) or output (open circles) to the processes. Each process is associated with a transfer function, which describes how its input is transformed into its output. In the Petri-net notation (E) ellipses and boxes represent states and reactions, respectively. The arrows indicate the function of the states in the reaction (reactant or product), and have a weight (here shown as a number next to the arrow) which relate to the stoichiometry of the associated reactants or products.

a typical cartoon. The three different diagram types are described below. Although the interpretation of dynamic model diagrams must be unambiguous, boxes and arrows denote different entities in these three different contexts.

1. Data-Flow Diagrams

In data-flow diagrams (Fig. 1C), boxes represent processes, and arrows may be seen as channels that shunt material or data from process to process. Electronic circuits are often represented as data-flow diagrams, and attempts have been made

to depict and model particular biochemical reaction networks (suggestively labeled "control circuits") in the similar way (e.g., Gilman and Arkin, 2002). Obviously, a box is a box and does not carry any information about the dynamics of input–output transformation that occurs "inside." In electronic diagrams, these boxes therefore have a certain shape, or carry a label that indicates which input–output relation, or "transfer function," should be used to describe its behavior. Such diagrams are particularly useful for a hierarchical description of complex systems in which the components have well-defined modes of operation and distinct functions. In that case, the description of a particular type of box can be used over and over, in the same way that particular types of electronic components can be used in different circuits.

2. State-Transition Diagrams

Chemical reaction schemes, written in the classical way (Fig. 1B), are easily transformed into state-transition diagrams (Fig. 1D). In this approach, the boxes (or symbols) and arrows are interpreted differently. In the dissociation reaction $S \rightarrow S + \text{Tu-GDP}$, for instance, in which Tu-GDP dissociates from S, a shrinking microtubule end, the symbols S and Tu-GDP represent biochemical species, or more precisely, *potential states* of biochemical species (species can be molecules, ions, complexes, macromolecular assemblies, etc.). The arrow represents a reaction—a physical or chemical *state transition*.

3. Hybrid Diagrams

The hybrid Petri-net[1] notation combines aspects of the data-flow and state-transition diagramming styles (Fig. 1E). It is equivalent to the standard chemical reaction notation (Fig. 1B) except that reactions and states are both represented explicitly as boxes and circles, respectively. The arrows that connect them indicate which states carry a reaction's input—its reactants—and output, the reaction products. The arrows have weights that represent the stoichiometry of the reactants and products in the reaction: in the reaction $S + 3\text{Tu-GTP} \rightarrow G$, the weights of the arrows that connect the input states S and Tu-GTP to the reaction box are 1 and 3, respectively, and that of the arrow that connects G is 1. A state has a population, formed by the number of "items" (i.e., molecules, complexes, assemblies) that are currently in that state. If there are not too many in a particular state, individual items may be depicted as small black circles inside the circle that represents the state.

[1] Named after its inventor, C.A. Petri; also called Place/Transition graphs. Petri-nets are used generally to model event sequences in concurrent processes. Here, we shall use some of the more evocative Petri-net terminology (transition, firing) but will avoid expressions that may lead to confusion (place, token).

C. Compartments

Each state is, by definition, restricted to a particular compartment. Thus, a molecular species that is present in the cytoplasm and in the nucleus has items in both the "cytoplasmic" and "nucleic" states, and movement of items from one compartment to another is modeled as a reaction. Concentrations are calculated from the number of items n_X in a particular state X and the volume V_X of the compartment associated with that state:

$$[\mathrm{X}] = \frac{n_\mathrm{X}}{N_\mathrm{A} V_\mathrm{X}} \tag{1}$$

where N_A is Avogadro's number (6.022×10^{23}). Here, we shall focus on reactions that take place *in vitro*, where there is usually just one compartment, with a constant volume, but it should be kept in mind that this is not necessarily the case. Ample consideration should be given to the effect of volume on concentrations and rate constants when introducing more than one compartment or variable volumes into a model.

D. Events and State Transitions

State-transition diagrams and Petri nets form the basis for "discrete event modeling." Discrete event modeling is used to examine how externally applied changes percolate through a system over time—not only in chemical kinetics, but also for instance in control engineering, operations management, and many other fields. An event is defined as something that happens instantaneously. The state transitions that are associated with chemical reactions (including biochemical ones) are usually modeled as events, because the time it takes for molecular species to change state is generally very short in comparison with the amount of time they spend being in a particular state.

Upon a state-transition event, reactant populations lose items, while product populations gain new items. The numbers of items lost and gained in an event are given by the weights of the arrows that connect the input and output states to the reaction box. We will refer to the occurrence of a state-transition event and the associated movement of items as the "firing" of the reaction. If the population of any of the input nodes to a transition is insufficient—that is, smaller than the weight of the connecting arc—the reaction can not fire and therefore will not occur. However, if the population of all input states is sufficiently high, a reaction will fire sooner or later. In Section III.F, we shall explain how the transition firing probability relates to the chemical rate constants that are associated with the transitions. However, before we can do that, we first need to discuss the basic concepts of chemical reaction kinetics and thermodynamics in well-stirred containers, which are central to the rest of the argument.

E. Reaction Kinetics

1. Basic Chemical Kinetics

In a chemical reaction, reactants are consumed, and products formed. The "instantaneous reaction rate" expressed as the quantity of reactant consumed or product formed per unit of time, depends on the concentrations of the reactants and on other factors, such as temperature.

Almost all chemical reactions have either one or two reactants. Unimolecular reactions, reactions with a single reactant, such as dissociations or isomerizations, appear to occur spontaneously, without the involvement of other molecules, and are associated with first-order rate constants, which have units of frequency (sec^{-1}, min^{-1}, etc.).

To calculate the instantaneous rate of a unimolecular reaction, the rate constant associated with the reaction, k_I (the subscript I is used here to indicate that the rate constant is first order), is multiplied by the number of items in the reactant state, n_X, or with the reactant's concentration, [X]:

$$v_{nX} = \frac{dn_X}{dt} = -k_I n_X \tag{2}$$

$$v_{[X]} = \frac{dX}{dt} = -k_I[X] \tag{3}$$

Note that $v_{nX} = v_{[X]} N_A V$, where V is the volume of the compartment or container in which the reaction takes place.

Bimolecular reactions have two reactants, which must collide to react. The frequency with which one molecule of the first reactant undergoes collisions with molecules of the second reactant increases with the *concentration* of the second reactant. The number of collisions, and therefore the reaction rate, decreases when the volume increases and the number of reactant molecules remains the same. Bimolecular reactions are associated with second-order rate constants (here indicated with the subscript II), which have units of frequency per unit of concentration (e.g., $M^{-1}\ sec^{-1} = L{\cdot}mol^{-1}\ sec^{-1}$), and *always* include the volume. The instantaneous rate for a bimolecular reaction is calculated by multiplying the number, n_X, *or* concentration [X] of one reactant, with the concentration [Y] of the other:

$$v_{nX} = \frac{dn_X}{dt} = -k_{II} n_X[Y] = -k_{II} \frac{n_X n_Y}{N_A V} \tag{4}$$

$$v_{[X]} = \frac{dn_X}{dt} = -k_{II}[X][Y] \tag{5}$$

Association reactions, such as cofactor binding or dimer formation, are examples of bimolecular reactions. Note that the rates at which X and Y disappear are the same.

If the population of state Y is much larger—say, 1000 times—than that of state X, the number of items in Y will not change significantly, even when the reaction goes to completion (1000 − 1 is still 99.9% of 1000). When such conditions apply, the highest reactant concentration is often taken to be constant, and incorporated in the rate constant of the reaction. The resulting product of the second-order rate constant for the reaction and the concentration of the most abundant reactant has units of frequency, and is referred to as a pseudo-first-order rate constant:

$$v_{[X]} = -k_{II}[X][Y] \approx -k_{II}Y_0[X] = -k_I'[X] \tag{6}$$

Here the pseudo-first-order rate constant $k_I' = k_{II}Y_0$, where Y_0 is the concentration of Y at the start of the reaction.

If the reaction is a dimerization ($2X \rightarrow X_2$), two items disappear from the X-state each time the reaction fires. A molecule of X cannot form a dimer with itself, and therefore has $n_X - 1$ other molecules of X to choose from:

$$v_{n_X} = \frac{dn_x}{dt} = 2k_{II}\frac{n_X(n_X - 1)}{N_A V} \tag{7}$$

but because $n_X - 1 \approx n_X$ for $n_X \gg 1$:

$$v_{[X]} = \frac{d[X]}{dt} = -2k_{II}[X]^2 \tag{8}$$

Reactions of a higher order,[2] in which more than two concentrations are multiplied with a rate constant to calculate the rate, also exist. Although these seldom involve the simultaneous encounter of more than two reactants, they are frequently used as an approximation of the real kinetics of a more complex process. We shall use the rescue reaction, S + 3Tu-GTP → G, as an example of a higher order reaction. The fact that its reaction rate is calculated as $v_{[S]} = d[S]/dt = -k_{IV}[S][\text{Tu-GTP}]^3$ does not mean that three Tu-GTPs must collide and bind simultaneously with a shrinking microtubule to turn it into a growing one. This form is merely used to approximate the kinetics of a much more complex mechanism.[3] Reactions of order zero, in which the reaction rate is independent of any concentration, also occur (at least conceptually, see Section III.E.3), and have rate equations of the form $v_{[X]} = d[X]/dt = -k_0$, where k_0 is a zero-order rate constant with units of concentration per unit of time.

[2] A reaction's molecularity is related to its stoichiometry, whereas its order is determined by the number of reactant concentrations that is taken into account to calculate the overall reaction rate, and is a kinetic concept; see Cornish-Bowden (1995) for a full explanation.

[3] In fact neither the stoichiometry nor the order of the rescue reaction follows from Horio and Hotani's data; a higher order dependency was used to explain the oscillations in the microtubule mass observed under conditions of extremely fast growth (Bayley *et al.*, 1989).

Irrespective of the reaction type, rates are expressed in number of items, mol, or concentration units per unit of time [e.g., sec^{-1}, $mol{\cdot}sec^{-1}$, or $M{\cdot}sec^{-1}$ where M (molar) = $mol{\cdot}L^{-1}$]. The reaction *firing frequency*, or flux, J, is related to the rate at which its reactant X disappears as:

$$J = -\frac{v_{n\mathrm{X}}}{s_{\mathrm{X}}} \tag{9}$$

where s_{X} is the number of items that are removed from the population of X when the reaction fires once (equal to the stoichiometry of reactant X). The general expression for the dependence of the reaction firing frequency on the reactant concentrations in a chemical reaction is:

$$J = k \times N_{\mathrm{A}} V \prod_{i=1}^{n} [\mathrm{R}_i]^{S_i} \tag{10}$$

Here n is the total number of different reactants, k is the rate constant, and $[\mathrm{R}_i]$ and S_i the concentration and stoichiometry of reactant R_i. This dependence is also known as the "law of mass action."

Instantaneous rates and firing frequencies represent the current state of the system, and may be calculated at any time during a reaction. In general, reactant concentrations are not constant. They may change during the course of a reaction because a reaction is out of equilibrium (see below, Section III.E.2), because an experimenter changes them in some way, for instance by increasing the volume, or because they are added or removed by coupled processes.

2. Equilibrium and Steady State

Most biochemical reactions are reversible and therefore tend toward an equilibrium state, where the forward rate equals the reverse rate, and the average concentration of each participant remains constant. The rate constants for the individual forward and reverse reactions reflect the free energy of activation for the reaction, $\Delta G^{\ddagger}$, the minimum free energy the reactants must have to reach the transition state (which must be crossed to form products). Rate constants are proportional to $T{\cdot}\exp[-\Delta G^{\ddagger}/RT]$, where R is the universal gas constant and T is the absolute temperature. The thermodynamic equilibrium constant, K, which is equal to the quotient of the rate constants k_{F} and k_{R} for the forward and reverse reactions, is related to the overall difference in free energy ΔG^{0} between the reactant and product states (under defined standard conditions, indicated by the superscript 0):

$$K = \mathrm{e}^{-\Delta G^{\circ}/RT} \tag{11}$$

which follows from

$$\frac{k_F}{k_R} = e^{-(\Delta G_F^{\ddagger} - \Delta G_R^{\ddagger})} \tag{12}$$

where the indices F and R refer to the forward and reverse reactions. If the forward and reverse reactions are of the same order, as in a reversible isomerization, the equilibrium constant is dimensionless. If the forward reaction is an association, and the reverse one a dissociation then, K, often referred to as an *equilibrium association constant*, indicated as K_a, has units of reciprocal concentration. Conversely, if the forward reaction is a dissociation and the reverse one an association, the equilibrium constant has units of concentration, and is called an *equilibrium dissociation constant*, and indicated as K_d. A high K_a indicates that the affinity is high; a high K_d indicates the opposite. The actual amount of complex formed depends, of course, on the concentrations of the participants. Note that *equilibrium* association and dissociation constants should not be confused with second- and first-order rate constants for association and dissociation processes, which are sometimes called association and dissociation *rate* constants.

It is important to appreciate that equilibrium is a *dynamic* state where the forward and reverse reactions fire at rates that are determined by the equilibrium concentrations of the participants. When an equilibrium is "perturbed," because the concentration of any of its participants change (for instance, because the experimenter adds some more), or because the equilibrium constant alters (for instance, as a result of a change in temperature), the forward and reverse rate are no longer equal. As a result, there will be a net flow of material, from reactants to products or vice versa, until a new equilibrium is reached. The process by which a system moves toward a new equilibrium position following a perturbation is called *relaxation*. Relaxation processes proceed at an overall rate that is, in reversible single-step reactions, determined by the sum of the forward *and* reverse reaction rates, irrespective of the actual direction of the flow. The larger the rates, the faster the relaxation, and the more rapidly the new equilibrium is reached. Thus, two different reversible reactions may have the same equilibrium constant, but very different relaxation rates. The relaxation characteristics of more complex systems depend on the rates of all reactions that are involved in coupled equilibria (further discussed in Chapter 15 by Eccleston *et al.*, this volume).

Equilibrium is a state that is often encountered *in vitro*, but hardly ever *in vivo*, where most reaction products are consumed by other reactions. When a reversible reaction, or a sequence of reversible reactions, is provided with a constant influx of reactants and efflux of products, the reaction will generally reach a "steady" state in which there is a net flux of material, which is constant, but not zero (in which case the reaction would be at equilibrium). Steady-state conditions may be perturbed because influx or efflux rates, or one or more reaction rate constants change. The response rate—the rate at which the system adapts to the

new conditions—depends, again, on both the forward and reverse reaction rates, with more dynamic systems having faster response times. Furthermore, the difference between steady state and equilibrium concentrations of the participants in the reaction depends on the ratio of the response rate to the influx and efflux rates: the faster the response, the closer the system can get to equilibrium.

3. Enzyme Kinetics

Enzyme catalyzed reactions are chemical reactions, and subject to the laws of thermodynamics. Therefore, they are modeled in exactly the same way as other chemical reaction systems, as combinations of association, isomerization, and dissociation events, with the appropriate kinetics. However, *in vitro* studies of enzymic reactions are often carried out under conditions where the response rate of the enzyme subsystem[4] is fast with respect to the rate at which the concentrations of its reactants change. Under such conditions, the enzyme will operate close to steady state. The steady-state distribution of enzyme species depends on the enzyme's mechanism, on the values of the rate constants, and on the concentrations of the reactants, but not on time, and not on the total enzyme concentration. The rate at which the reaction product is formed depends on the *absolute* concentrations of one or more enzyme species, and can be calculated from the distribution of enzyme species and the total enzyme concentration. The steady-state rate equation for a particular enzyme mechanism—sometimes called the kinetic law—describes the steady-state rate of reactant consumption or product formation in terms of kinetic constants, and enzyme and reactant concentrations. A derivation of the rate equation for the Henri–Michaelis–Menten mechanism can be found in any biochemistry textbook. The steady-state rate equation for this mechanism, $E + S \Leftrightarrow ES \rightarrow E + P$, where E is the enzyme, S is its substrate (reactant), and P is its product, is

$$v_{[P]} = \frac{d[P]}{dt} = k_2 e_0 \frac{[ES]}{e_0} = k_2 e_0 \frac{[S]}{((k_{-1} + k_2)/k_1) + [S]} \tag{13}$$

where k_1 and k_{-1} are the second-order forward, and first-order reverse rate constants for reactant binding to and dissociation from the enzyme, and k_2 combines the transformation of S into P and the dissociation of P from the enzyme into a single first-order rate constant. As e_0 is the total enzyme concentration, $[ES]/e_0$ is the fraction of the enzyme that is in the ES state, and has a dimensionless value between 0 and 1. The denominator $(k_{-1} + k_2)/k_1$ forms the Michaelis constant, K_M, and the product $k_2 e_0$ is the maximum rate, V_{max}, at which the reaction can proceed.

Steady-state rate equations can be derived for any enzyme mechanism, irreversible or reversible, with any number of reactants, products, or modifiers (see below). Techniques that facilitate the derivation of steady-state rate equations, notably

[4] The enzyme subsystem includes all enzyme species: free enzyme, enzyme-substrate, enzyme-product, and other complexes, but not unbound reactants or products.

the King and Altman method, are outlined in many monographs on enzyme kinetics (e.g., Cornish-Bowden, 2001), and some present the expressions for a large number of mechanisms (particularly Segel, 1975). Steady-state rate equations are all of the form

$$v_{[P]} = \frac{d[P]}{dt} = \kappa \times e_0 \times \varphi \tag{14}$$

The apparent first-order rate constant κ is a combination of rate constants, whereas φ, the fraction of enzyme species that determine the overall product formation rate, is a combination of rate constants, and reactant, product, and modifier concentrations. The precise forms of κ and φ depend on the enzyme mechanism. Rate equations for reversible reactions are of the form $v = v_F - v_R = e_0(\kappa_F\varphi_F - \kappa_R\varphi_R)$, where the indices F and R refer to the appropriate expressions for the forward and reverse reactions.

When an enzyme with Michaelis–Menten kinetics is operating under conditions where [S] is much smaller than K_M, the rate of the overall reaction S → P is proportional to [S] (provided, of course, that e_0 is constant), and the reaction behaves as a first-order reaction, with k_2e_0/K_M, which has units of frequency, as its rate constant. When [S] is much larger than K_M, however, [ES]/e_0 is very close to one, and the enzyme is "saturated." In this regime, the reaction S → P proceeds at a constant rate, independent of [S], and the kinetics look like those of a zero-order reaction. Reactions catalyzed by enzymes whose mechanism is more complex often exhibit a similar dependence of their apparent kinetic order on reactant concentration.

4. Beyond Mass Action

Enzyme catalyzed reactions may be described as sets of coupled reactions with mass-action kinetics in which the basic reactions each have their own (mass action) rate equation. However, the Henri–Michaelis–Menten scheme can be reduced simply to S → P, with Eq. (13) ($v = V_{max}[S]/(K_m + [S])$, provided steady-state conditions apply) describing the dependence of its rate on the reactant concentration [S], and the same holds for more complex mechanisms. In general, any equation that describes the dependence of the rate at which reactants are consumed (or products appear) on concentrations of the reactants may be used to calculate the reaction firing frequency. However, when derived rate equations are used, it is very important to be aware of the assumptions that were made during their derivation, and of any restrictions to their validity.

Derived equations may also describe the dependence of the rate on components that are not consumed in the reaction, or on other conditions. Such components or conditions function as rate *modifiers*. Modifiers may form one to one complexes with reactants or products, or act in other fixed proportions, but are not associated

with the stoichiometry of the reaction (as they are neither consumed, nor produced). In Eqs. (13) and (14), for instance, the value of e_0, the total enzyme concentration, determines the overall reaction rate, but E is neither a reactant, nor a product in the S → P reaction. Therefore E functions as a modifier in this reaction. Modifiers that decrease the reaction rate may be called inhibitors or repressors, and those that increase the rate may be activators, stimulators, catalysts, promoters, etc. Under different conditions, modifiers may have different, even opposite, effects, and it is important to recognize that labels, such as inhibitor or catalyst, do not imply a mechanism of action. The modifier concept is particularly useful in systems in which the concentration of the modifier is variable: coupled futile cycle systems, such as MAPK cascades, are often simplified using steady-state rate equations with variable enzyme concentrations (as in, for instance, Markevich *et al.*, 2004).

F. Timing of Events in Chemical Reactions

1. Transition Probabilities

Although it is not possible to predict *exactly* when any *individual* molecule will undergo a chemical reaction, the *likelihood* that it will do so within the next second (or minute, or year) can be calculated. To illustrate this point, we use the example of the dissociation of the Tu-GDP complex (Tu-GDP → Tu + GDP). If there is no reverse reaction (for instance, because Tu or GDP are removed rapidly via other processes), the rate at which the number of items in the Tu-GDP state, n, changes is $dn/dt = -k \times n$ [Eq. (2)]. This is an example of an ordinary[5] differential equation (ODE), which describes the relation between n (which is a function of time) and its derivative with respect to t. This particular ODE has an "analytical" solution, an algebraic equation that explicitly describes the dependence of n on t. The analytical solution is $n = C \times e^{-kt}$, where C is an arbitrary constant. If the number of items in the Tu-GDP state is n_0 at $t = 0$, then $C = n_0$, and the dependence of n on t can be written as $n = n_0 \times e^{-kt}$. Thus, the *fraction* of items still in the Tu-GDP state at time t is equal to $n/n_0 = e^{-kt}$. Conversely, the fraction that has already reacted is $1 - n/n_0 = 1 - e^{-kt}$.

The dissociation rate constant k for Tu-GDP has been measured to be about 0.09 sec^{-1} (at 298 K, Engelborghs and Eccleston, 1982). If there were 10^6 Tu-GDP complexes at $t = 0$, one second later there would have been about 0.914×10^6, or 91.4% remaining, that is, 8.6% of the initial population would have dissociated. Thus, the *probability* that any individual item present in the Tu-GDP state at $t = 0$ will disappear from that state within one second is 0.086, or 8.6%.

Essentially the same holds for bimolecular reactions under pseudo-first-order conditions. The (second-order) association rate constant for the reaction Tu + GTP → Tu-GTP was estimated to be 2.2×10^6 M^{-1} sec^{-1} (Brylawski and Caplow, 1983).

[5] An "ordinary" differential equation contains the derivative of a function to a single variable. Partial differential equations (PDEs) involve functions that are dependent on more than one variable (e.g., time and position).

If 10^6 items are initially in the nucleotide-free Tu state, and there are 10^9 GTP molecules, all in a volume of 1 pl (i.e., [GTP] = 1.66 mM), the pseudo-first-order rate constant k' is $2.2 \times 10^6 \times 1.66 \times 10^{-3} = 3.6 \times 10^3\ \text{sec}^{-1}$, which can substitute for k in the above equations. In the absence of a reverse reaction, the probability that any particular item has disappeared from the Tu state one millisecond after the start of the reaction is then calculated to be 97%.

2. Lifetime Distributions

As noted above, the general expression for the probability P that a particular item that was in the reactant state of a first-order reaction at t_0 has disappeared from that state at a given time t is:

$$P = 1 - \mathrm{e}^{-k(t-t_0)} \tag{15}$$

On the basis of the above example for Tu-GDP dissociation, we can calculate the number of items that stay in the Tu-GDP state for longer than 2 sec, but shorter than 3 sec, in other words, items that have lifetimes between 2 and 3 sec. At $t = 2$ sec, 16.5% will have already reacted, and have therefore spent less than 2 sec as Tu-GDP. At $t = 3$ sec, 23.7% will have reacted. Therefore, 7.2% have lifetimes between 2 and 3 sec. Similar calculations can be carried out for other intervals—0–1 sec: 8.6%, 1–2 sec: 7.9%, and so on, and the results may be plotted as a histogram that shows the fraction of items with lifetimes within certain intervals. Furthermore, it will be possible to calculate the average lifetime of an item in the Tu-GDP state, and the standard deviation on this average.

It turns out that this distribution of lifetimes, also called the *probability density function, f*, follows the first derivative of the expression for the probability P that, at time t, an item has disappeared from the state it was in at time t_0, that is, Eq. (15):

$$f = \frac{\mathrm{d}P}{\mathrm{d}t} = k \times \mathrm{e}^{-kt} \tag{16}$$

The expression for P in Eq. (15) is called the *cumulative distribution function* for the probability density function in Eq. (16). The type of distribution expressed in Eq. (16) is called an exponential distribution. If the lifetimes are exponentially distributed, the average lifetime, $\langle\tau\rangle$, as well as its standard deviation, $\sigma(\tau)$ are known to be equal to the reciprocal of the rate constant $\langle\tau\rangle = \sigma(\tau) = 1/k$. Thus, the value of a first-order rate constant—and the same holds for a pseudo-first-order one—predicts how long it will be, on average, before a reaction occurs. Therefore, the complex Tu-GDP in the example will exist for an average of 11 sec (±11 sec, because $\sigma(\tau) = \langle\tau\rangle$) after it has been formed.

It is difficult, or impossible, to derive an algebraic expression for P for higher order reactions, or for reaction sequences. However, the equation $f = \mathrm{d}P/\mathrm{d}t$ is generally true, also for lifetime distributions with different shapes (for instance, the

well-known Gaussian or normal distribution). If the probability density function is known or can be measured, the above relationship [Eq. (16)] may be used to calculate the cumulative distribution function.

IV. Simulation

To simulate the behavior of a biochemical reaction system, at least three pieces of information must be provided:

1. A model that describes the *structure* of the system (states and compartments, reactions, the connections between states and reactions, and the connection weights)
2. *Rate equations* for all reactions (expressions defining the rates at which the reactions fire, given a certain input) and values for the associated parameters
3. The *initial state* of the system (the population of each state and the volume(s) of the compartments in which the reactions take place).

Once the necessary data have been provided, time series—state populations recorded over time—can be generated computationally. In *stochastic* simulations, a random number generator is used to produce possible time courses, or trajectories, of individual items. Because of the randomness in the timing of individual events, individual stochastic simulation rounds have different outcomes, even if the initial conditions are exactly the same in each round. By contrast, *deterministic* simulations always give the same result for the same set of initial conditions. Deterministic simulations usually yield trajectories that are equal to the average of many rounds of stochastic simulation of the same system, given identical initial conditions.

A. Stochastic Methods: The Behavior of Individual Entities

We shall illustrate the principles of stochastic simulation using models for microtubule dynamic instability (see Section III.A) that increase in complexity as we introduce new concepts. In all of these models, the dissociation of a tubulin subunit from a shrinking end, $S_n \rightarrow S_{n-1}$ + Tu-GDP, and the transition from a growing to a shrinking end, $G_n \rightarrow S_n$, where n indicates the number of subunits in the microtubule, will be modeled as first-order processes, with rate constants $k_{SS} = 220$ and $k_{GS} = 0.0056$ sec^{-1}, respectively. The binding of Tu-GTP to a growing end, G_n + Tu-GTP $\rightarrow G_{n+1}$, and the rescue of a shrinking end, S_n + 3Tu-GTP $\rightarrow G_{n+3}$, will initially be modeled as pseudo-first-order processes, with rate constants k'_{GG} of 16 sec^{-1} and $k'_{SG} = 0.056$ sec^{-1}. The values for k_{GS} and k'_{SG} are derived from the average lifetimes of the growing and shrinking states, $\langle \tau_G \rangle = 180$ sec, and $\langle \tau_S \rangle = 18$ sec, using the relationship $k = 1/\langle \tau \rangle$, and assuming that the lifetimes observed by Horio and Hotani (1986) were exponentially distributed.

In the next stage, we shall model the association reactions as higher order processes (second and fourth order, respectively). To derive the values of the second- and fourth-order rate constants from the values of their pseudo-first-order counterparts, we need to know at which Tu-GTP concentration they were obtained. We shall assume that it was about 10 μM. If k_{GG}·[Tu-GTP] = 16, and k_{SG}·[Tu-GTP]^3 = 0.056 sec^{-1} at [Tu-GTP] = 10 μM, then $k_{GG} = 1.6 \times 10^6$ M^{-1} sec^{-1}, and $k_{SG} = 5.6 \times 10^{13}$ M^{-3} sec^{-1}.

To keep the models simple, we will assume that there is no subunit dissociation in the growth phase, or binding in a shrinking phase, and that Tu-GDP is not able to bind to either type of microtubule end. Furthermore, we shall disregard the fact that microtubule growth and shrinkage are probably associated with more than two rate constants, and that catastrophe and rescue are much more complex processes. These conditions need to be relaxed in more complex models (see Janulevicius *et al.*, 2006; Martin *et al.*, 1993) but the principles of the simulation remain the same.

1. The Behavior of a Single Item

First, we shall concentrate on the behavior of a *single* microtubule end under pseudo-first-order growth and rescue conditions (i.e., at a constant Tu-GTP concentration, for instance, because a GTP regenerating system is used). To start a simulation, the initial state—G or S—of the system must be chosen. Here, we arbitrarily choose that the end is in the G state, and set the total simulated time t_{sim} to 0.

One of two things will happen to a growing microtubule end: it may gain another subunit and remain in the growing state or it may change state and become a shrinking end. Either possibility has a finite (i.e., nonzero) chance of happening. To decide which of these two events will happen on this occasion, and when, we simply "draw straws," but make the odds reflect the very different probabilities for the two transitions. To do this, one random number between 0 and 1 is generated for each possible event (r_1 for $G_n \rightarrow G_{n+1}$, and r_2 for $G_n \rightarrow S_n$), and used to calculate the time at which each of the events would happen. This is done by substituting r_1 or r_2 for P, and k'_{GG} or k_{GS} for k in Eq. (15):

$$r_1 = 1 - e^{-k'_{GG}(t-t_0)}, \quad r_2 = 1 - e^{-k'_{GS}(t-t_0)}$$

and solving for $\Delta t = t - t_0$ in both cases:

$$\Delta t_1 = \frac{-\ln(1 - r_1)}{k'_{GG}} \text{ for the } G_n \rightarrow G_{n+1} \text{ reaction}$$

$$\Delta t_1 = \frac{-\ln(1 - r_2)}{k_{GS}} \text{ for the } G_n \rightarrow S_n \text{ reaction}$$

If this process was to be repeated many times, and r_1 and r_2 plotted against Δt, the cumulative distribution functions [Eq. (15)] for the two reactions would emerge.

The reaction with the smallest Δt is then selected as the one that will occur, the other is ignored. Suppose, for example, that the values for r_1 and r_2 were 0.44313 and 0.17621, then Δt_1 and Δt_2 would be 0.036 and 34.6 sec, and the continued growth reaction would be selected. The transition that is *most likely* to occur is, of course, always the one with the largest rate constant, that is, $G_n \rightarrow G_{n+1}$ and this is why the microtubule will continue to grow for a long time. However, there is a $0.0056/16 \times 100\% = 0.04\%$ chance that Δt_2 will be smaller than Δt_1, and that the $G_n \rightarrow S_n$ transition will be selected. For example, had r_1 and r_2 been 0.99927 and 0.00024, Δt_1 and Δt_2 would have been 0.451 and 0.043 sec, and the catastrophe reaction would have been selected.

Having decided which reaction will occur ($G_n \rightarrow G_{n+1}$ in this case), the simulation is updated by increasing the simulation time t_{sim} by the selected Δt (Δt_1 in this case) and letting the associated reaction fire at the new t_{sim}. The microtubule remains in the same state (G) but increases in length by one subunit.

To continue the simulation, this whole process must be repeated. If, as here, the microtubule remains in the growing state then k'_{GG} or k_{GS} are again used to evaluate newly drawn values of r_1 and r_2 for selecting the next step. However, had the $G_n \rightarrow S_n$ transition "won," then the microtubule would have switched states and the next event would be either shortening ($S_n \rightarrow S_{n-1}$) or rescue ($S_n \rightarrow G_{n+3}$) and k_{SS} or k'_{SG} would be used in choosing between them. The simulation is continued in this way until a preset end time is exceeded.

Figure 2 shows a typical simulated time course—a trajectory—of the microtubule end. It is clear from this figure that the spread in the lengths of both the growth and shrinkage periods is large, but that the latter are, on average, much shorter than the former. The close-up of a $G \rightarrow S$ transition in which the individual events are resolved shows that the average time interval between two subunit additions to a growing end is also much larger than that between two subunit dissociations from a shrinking end. The histogram shows that these intervals (lifetimes) are indeed exponentially distributed, with measured averages of 0.064 ($\pm$0.065) and 0.0044 ($\pm$0.0043) sec. The average growth and shrinkage rate constants that are calculated from these lifetimes are 15.6 and 227 sec, very close indeed to the 16 and 220 that were used as input.

To illustrate the simulation of higher order reactions, we shall now model Tu-GTP binding to a growing microtubule end as a second-order process ($G_n + \text{Tu-GTP} \rightarrow G_{n+1}$), and rescue of a shrinking end as a fourth-order process ($S_{n+3} + 3\text{Tu-GTP} \rightarrow G_{n+3}$). Initially we will, again, be looking at the behavior of a single microtubule end. We shall start the simulation with 50 μM Tu-GTP in a volume of 0.01 pl, that is, with about 3×10^5 molecules, with the microtubule end (whose concentration is 0.17 nM) in the growing state.

In this model, one molecule of Tu-GTP is removed from the reaction volume by each Tu-GTP association event, and three molecules are used for the rescue reaction. As GTP is hydrolyzed after incorporation of Tu-GTP in the growing microtubule, and not regenerated, the Tu-GTP concentration is variable. However, in stochastic simulations, the state of the system does not change between two

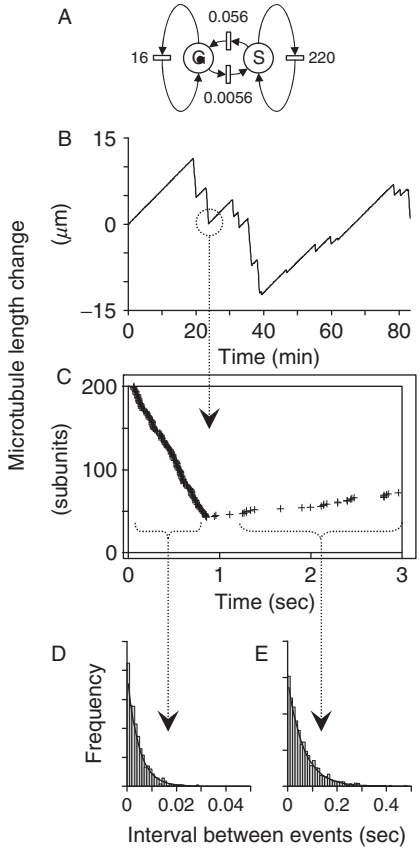


Fig. 2 Dynamic instability of a single microtubule end with growth, shrinkage, and state interconversion modeled as first-order processes, that is, under conditions in which [Tu-GTP] is constant (for instance, because an enzymic GTP regenerating system is used). (A) Petri-net diagram of the model. The numbers are the values for the first-order rate constants associated with the reactions. The small filled circle inside the larger open circle indicates the situation at the start of the simulation: one item (i.e., a microtubule end) in state G. (B) Trajectory of the length changes of the microtubule end during one stochastic simulation round of 5000 sec. (C) Close-up of the events (crosses) close to one S $\rightarrow$ G state transition. Note that the dissociation events in the shrinking phase are more closely spaced than the association events in the growing phase. This is borne out by the average duration of intervals between successive events in the S-phase, 0.0044 ± 0.0043 sec, and in the G-phase, 0.064 ± 0.065 sec. In both cases, the distribution of the intervals observed in the simulation is exponential, as shown in the histograms in (D) and (E). The solid lines represent the lifetime distributions according to Eq. (16), with $k_{SS} = 220$ (D) and k_{GG} 16 (E) sec^{-1}.

successive events, and is reassessed after each event. Therefore, exactly the same method of drawing a random number for each reaction with a finite firing probability, and calculating the associated Δt values, can be applied to decide which reaction will occur next, and when. However, rather than using constant

pseudo-first-order rate constants to assess the Δt values, the values for $k'_{GG} = k_{GG}[\text{Tu-GTP}]$ and $k'_{SG} = k_{SG}[\text{Tu-GTP}]^3$ are reassessed each time a molecule of Tu-GTP disappears from the reaction volume. Thus, under the initial conditions of the simulation, when [Tu-GTP] is 50 μM, k'_{GG} is $1.6 \times 10^6 \times 50 \times 10^{-6} = 80\ \text{sec}^{-1}$, and k'_{SG} is $5.6 \times 10^{13} \times (50 \times 10^{-6})^3 = 7\ \text{sec}^{-1}$. When, after some 2.4×10^5 association events, only 10 μM Tu-GTP is left, these rates are (as expected at 10 μM Tu-GTP, see the earlier description) 16 and 0.056 sec^{-1}.

Figure 3 shows a typical trajectory, and it can be clearly seen that the end grows more and more slowly, whereas the shrinking episodes increase in length as [Tu-GTP] decreases. The combined result is a dramatic disassembly when [Tu-GTP] decreases below about 10 μM.

2. Modeling the Behavior of Multiple Items

To model the individual behavior of more than one microtubule end, we simply create as many copies of the GS elements as necessary, where a GS element consists of a G and an S state that are connected via G → S and S → G reactions in the same way as in the previous model. However, Tu-GTP is supplied to these elements from a single

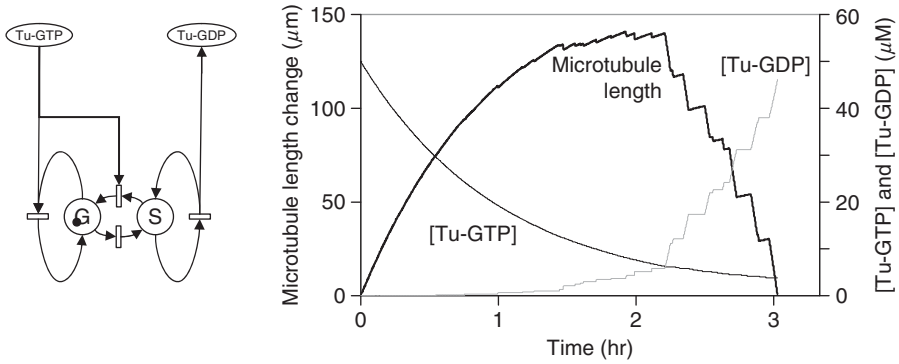


Fig. 3 Dynamic instability of a microtubule end in which growth (G_n + Tu-GTP → G_{n+1}) and rescue (S_n + 3Tu-GTP → G_{n+3}) are modeled as second- and fourth-order reactions, respectively. The Petri-net diagram of the model (left) indicates that the simulation was started with a single item in state G. The initial Tu-GTP concentration was 50 μM (300,000 molecules in a volume of 0.01 pl). The graph (right) shows the trajectories obtained during one stochastic simulation round over about 10,000 sec. In this model, one Tu-GTP item is consumed by each association event in the growth phase, and one Tu-GDP item is produced per dissociation event during shrinkage. However, Tu-GTP is not regenerated, so that [Tu-GTP] decreases, and [Tu-GDP] increases during the simulation (concentrations indicated on the right axis). In the first 90 min of the simulation, any shrinking microtubules are rapidly rescued (at a rate of k_{SG}·[Tu-GTP]^3), and the microtubule gains about 140 μm in length (left axis). As [Tu-GTP] decreases, the growth rate (k_{GG}·[Tu-GTP]), as well as the rescue rate decrease, resulting in the saw-tooth trajectories that are observed for microbubules under (near) steady-state conditions. When [Tu-GTP] falls below a certain critical concentration, (~10 μM), subunit loss during the increasingly long periods of shrinkage outweighs the slower and slower gain during the growths phases, and the polymerized state is no longer permanently sustained.

pool, as shown in the 3-microtubule model in Fig. 4. It is now necessary to set the initial state of each individual element, and in the example we have chosen to start the simulation with all three growing. The initial Tu-GTP concentration is, again, 50 μM.

In the previous models, there were always two reactions to choose from; now there are six, two for each microtubule end. The simulation principle remains the same: a random number is drawn for each of the six possible reactions, and the reaction that produces the smallest Δt is selected. Figure 4 shows the three trajectories that result. Note that the Tu-GTP pool is exhausted three times more quickly than in the example with a single microtubule (Fig. 3).

If the individual microtubule end trajectories are of no particular interest, it is also possible to just use a single GS element, as in the model in Fig. 3, but simply distribute the required number of items—three in this example—over the states in

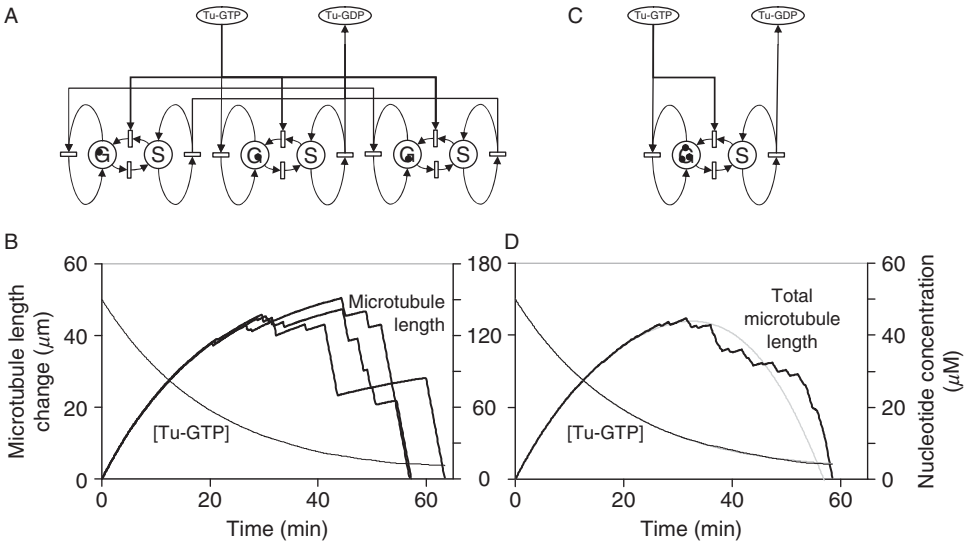


Fig. 4 Behavior of three microtubule ends, modeled as three different GS elements, each containing one item (A, B), or as one GS element containing three independent items (C, D). At the start of the simulations, all items (i.e., microtubule ends) are in a G state, and [Tu-GTP] = 50 μM (300,000 molecules in a volume of 0.01 pl). (A, C) Petri-net diagrams, indicating the state of the models at the start of the simulation; (B, D) microtubule length change and [Tu-GTP] trajectories observed in one simulation round over about 4000 sec. Because there are now three microtubule ends in both simulations, rather than one, Tu-GTP is consumed three times faster than in the simulation shown in Fig. 3, and the individual microtubules gain maximally about 45 μm in length before the critical Tu-GTP concentration is reached. Each microtubule is modelled as a single, distinguishable element in model (A), their individual paths in one stochastic simulation round are recorded and depicted in (B). Multiple items in a single state are simply counted, not individually identified, and the trajectory obtained in one stochastic simulation round (D, black lines) is therefore the *sum* of three trajectories (note the scale of the left axis). The gray lines in (D) are the trajectories for [Tu-GTP] (almost indistinguishable from the black line) and the total microtubule length, obtained by numerical integration (deterministic simulation) of the model in (C).

the GS element. Here, we have again placed all three items in the G state, and started the simulation with 50 μM Tu-GTP.

With all three microtubules growing, the probability that one of them will acquire a new subunit within the next second is, of course, three times as large as that for a single growing microtubule. Thus, the value of k'_{GG} is now calculated as $3 \times k_{GG} \times$ [Tu-GTP]. Likewise, the chance that any one of the items will change state is also three times that of a single microtubule under the same conditions. The simulation is, again, carried out along the same lines: two random numbers are chosen for the two possible reactions (GG and GS), and used to calculate values for Δt, whereupon the smallest value and its associated reaction are selected. However, rather than using $k_{GG} \times$ [Tu-GTP] and k_{SG}, the firing frequencies [Eq. (10)], $J_{GG} = 3 \times k_{GG} \times$ [Tu-GTP] and $J_{GS} = 3 \times k_{GS}$ are used to substitute k in Eq. (15).

If all microtubules continue growing, the distribution of items over the two states will remain the same (i.e., all three items in the G state); if the selected reaction is GS, however, one item will move to the S state. If that happens, all four reactions (SS, SG, GS, and GG) may occur in the next step, because all reaction firing frequencies now have finite values ($J_{GG} = 2 \times k_{GG} \times$ [Tu-GTP], $J_{GS} = 2 \times k_{GS}$, $J_{SG} = 1 \times k_{SG}$[GTP]^3, and $J_{SS} = 1 \times k_{SS}$). In that case, random numbers are drawn for all possible reactions, and the appropriate firing frequencies are used to substitute k in Eq. (15). The simulation is continued further as outlined above.

Multiple items in one state do not have individual identities, and it is, therefore, not possible to follow the trajectory of a single microtubule end in this way. Instead, the distribution of items over the two states is recorded, and the resulting trajectory, also shown in Fig. 4, is the sum of the individual trajectories.

3. Recipe for Performing Stochastic Simulations

Thus, the standard procedure for the stochastic simulation of a chemical reaction system comprises the following steps:

1. Choose a start time t_0 and end time t_{end}. Set the number of items in each of the states to their values at t_0, and the simulation time t_{sim} to t_0
2. Calculate the firing frequency J_i for each reaction i
3. For each transition i for which $J_i > 0$, draw a random number r_i ($0 \leq r_i < 1$) from a uniform distribution,[6] and compute the time t_i at which it will fire:

$$t_i = t_{sim} + \frac{-\ln(1-r)}{J_i} \tag{17}$$

4. Select the reaction R for which t_i is smallest as the one that will occur next
5. Set t_{sim} to the smallest value of t_i and let R proceed
6. Repeat from (2), until t_{sim} exceeds t_{end}.

[6] Be aware that random number generators may limit the time resolution: if, for instance, r is drawn from a sequence of 8-bit integers, the smallest finite value obtainable is 2^{-8}.

This procedure is at the heart of the algorithm for exact stochastic simulation of coupled chemical reactions known as Gillespie's direct method (Bortz *et al.*, 1975; Gillespie, 1977).

4. Speeding up Stochastic Simulations

Stochastic simulations are inherently slow, because each event is taken into consideration, and often multiple calculations must be carried out to decide on the timing of the next event. It is sometimes possible to increase the efficiency of the calculations without losing the intrinsic precision of this type of simulation. Below we have given a short description of some of the methods that have been proposed. However, these methods have either extra computational overhead or reduced precision. If one's interest is in the behavior of a single or a few items in a limited number of states, and a choice of a few transitions at any one time (e.g., a microtubule end), the simplest method in which a random number is drawn for each transition with a finite firing frequency may well be the most efficient.

1. The original algorithm proposed by Gillespie (1977) required the drawing of only two random numbers per step, rather than one for each reaction. The first random number is used to establish at what time any reaction will fire, and the second one to decide which reaction this will be. It involves the firing frequencies J for each reaction, and the overhead consists of dividing this sum into sectors proportional to the individual firing frequencies.

2. If the firing probability of reaction A is unaffected by the firing of reaction B, values for t_i that are calculated for A are not invalidated by the firing of B. In the "Next Reaction Method," proposed by Gibson and Bruck [who also coined the label "First Reaction" method for the Gillespie algorithm, Gibson and Bruck, (2000)], all valid values for t_i are kept in a priority queue, together with a reference to their associated reactions, which are fired at the appropriate times. The queue needs to be maintained and adjusted after each firing, but if the degree of independence in the system is high (i.e., there are many reactions whose firing frequency is not affected by the firing of many other reactions), this method can be much more efficient than the First Reaction Method.

3. Tau-leaping (Gillespie, 2001) is an approximate method in which the simulation time is advanced by a preselected "leap" interval in which one or more transitions are likely to fire significantly more often than once. The actual number of firings is chosen, again, on the basis of a random number. If the length of the leap interval is chosen so that the total flux in the system is predicted to change by a negligible amount over this period, this technique may result in a greater simulation speed, at the cost of some of the accuracy of the First and Next Reaction methods.

4. We have so far only considered transitions that represent simple mass-action processes, whose firing frequency J, under given constant input conditions, is independent of time. However, to save on computation time, a transition may be defined to represent a sequence or combination of separate steps. In this case, its

reactants may have a lifetime expectancy that changes as they get older, and the cumulative distribution function has a form that is different from Eq. (3). If the shape of the cumulative distribution function can somehow be assessed,[7] the procedure outlined in Section IV.A.3 can be followed. An example is given in Gibson and Bruck (2000) and the method is explained in greater detail in Schilstra and Martin (2006).

B. Deterministic Methods: The Average Behavior of Many Entities

The behavior of the three microtubule ends in the example in Section IV.A.2 illustrates the point that the trajectories of individual entities often deviate significantly from the average paths of many, but, nonetheless, that such an average does exist, and that its time course is determined by the structure and parameter values of the systems, and by the initial conditions. The average path may be assessed by repeating the stochastic simulation many times, or by the simultaneous inclusion of many particles in a single simulation, but the computational cost of such simulations tends to be very high, and may often be prohibitive.

1. Numerical Integration of Ordinary Differential Equations

There is a much faster and more efficient way of assessing average trajectories: numerical integration of the system of coupled ordinary differential equations (ODEs) that describe how the number of items in each state changes over time. To set up such a system of equations, the expressions for flux of items into and out of each state are combined to give the overall rate at which the number of items in that state changes. Reactions that add items to a particular state make a positive contribution, and reactions that remove items make a negative one. Thus, the overall change $\mathrm{d}(n_S)$ in the number of items in state S (n_S) over a very small time interval $\mathrm{d}t$ is equal to:

$$\begin{aligned}\frac{\mathrm{d}(n_S)}{\mathrm{d}t} &= -v_{SS} + v_{SS} - v_{SG} + v_{GS} \\ &= -s^S_{SG} J_{SG} + s^S_{GS} J_{GS} \\ &= -1 \times k_{SG} \times n_S \times [\mathrm{Tu-GTP}]^3 + 1 \times k_{GS} \times n_G\end{aligned}$$

Here, v_{SG}, J_{SG}, and k_{SG} are the rate at which items are removed from S by the S → G reaction, the firing frequency of this reaction and the reaction's rate constant, respectively. The number of items in the G and Tu-GTP states are n_G and $n_{\text{Tu-GTP}}$, and s^S_{SG} and s^S_{GS} represent the stoichiometry of S in the S → G, and

[7] For instance by numerical integration of Eq. (2) in which k is a function of time; further discussion of this subject is beyond the scope of this chapter.

G → S reactions (both 1). Note that $-v_{SS}$ and v_{SS} cancel (as the number of items in S stays the same when the S → S reaction fires), and that the rate $d(n_S)/dt$ is expressed as number of items per unit of time. The rate at which the number of items in S (and in the other states) changes may also be written in terms of concentration: $d[S]/dt = -k_{SG}[S][\text{Tu-GTP}]^3 + k_{GS}[G]$.

The full model used in Section IV.A.2 corresponds to the following system of coupled[8] ODEs:

$$\begin{aligned}
\frac{d[G]}{dt} &= -k_{GS}[G] + k_{SG}[S][\text{Tu-GTP}]^3 \\
\frac{d[S]}{dt} &= k_{GS}[G] - k_{SG}[S][\text{Tu-GTP}]^3 \\
\frac{d[\text{Tu-GTP}]}{dt} &= -k_{GG}[G][\text{Tu-GTP}] - 3k_{SG}[S][\text{Tu-GTP}]^3 \\
\frac{d[\text{Tu-GDP}]}{dt} &= k_{SS}[S]
\end{aligned}$$

There is no analytical solution (an expression that describes the concentrations as a function of time) to this and most other coupled ODE systems. However, if an initial set of concentrations is provided, it is possible to create a *numerical* solution. In the simplest implementation, a time step Δt is chosen over which none of the concentrations are expected to change by more than a certain very small percentage. The concentration changes over Δt are calculated by multiplying the expressions for the rate by the time interval. For example

$$\Delta[G] = [G]_{t+\Delta t} - [G]_t \approx (-k_{GS}[G]_t + k_{SG}[S]_t[\text{Tu-GTP}]_t^3)\Delta t$$

where the subscripts t and $t + \Delta t$ indicate current concentration, and predicted concentration after the time step Δt, respectively. The new concentrations are then calculated by adding these changes to their (known) present values:

$$[G]_{t+\Delta t} \approx [G]_t + (-k_{GS}[G]_t + k_{SG}[S]_t[\text{Tu-GTP}]_t^3)\Delta t$$

This is done for all equations in the set, and the process is repeated until a preset end time or other end point, such as a certain concentration level, is reached. This accumulation process is called numerical integration.

Smaller time steps result in smaller relative changes, and in more accurate solutions, but also in an increased total simulation time. With time steps approaching zero, the trajectories obtained with this method will approach the paths that

[8] They are coupled because the variables in the left-hand-side of the equations return in the right-hand-side of other equations in the same system.

would result from averaging an infinite number of stochastic simulations of the same system (under the same initial conditions). If the time steps taken are too large, the solution will not only lose accuracy but may also become unstable. In an unstable solution, the calculated values typically oscillate wildly with amplitudes that increase with every new time step.

2. Algorithms

Many algorithms have been developed to improve the accuracy, stability, and efficiency of numerical solutions to coupled ODE systems. The procedure outlined above, which is the oldest, simplest, but least efficient and stable, is known as Euler's method. The popular Runge–Kutta method is an example of a so-called higher order algorithm in which information from several substeps is used to match a Taylor expansion of order greater than one to create a more accurate estimate. So-called multistep, or predictor–corrector algorithms, such as the Adams–Bashford method, estimate the size of the changes on the basis of more than one previously computed step (Press *et al.*, 1989).

The above algorithms are examples of *forward* or *explicit* methods. In *backward* or *implicit* methods, such as the backward Euler algorithm, equations must be solved that express the estimates of the concentrations at $t + \Delta t$ as a function of the current concentrations. To obtain estimates for $[\mathrm{G}]_{t+\Delta t}$, $[\mathrm{S}]_{t+\Delta t}$, and $[\text{Tu-GTP}]_{t+\Delta t}$, for example, the set of simultaneous equations that describe the current concentrations in terms of their expected values, such as

$$[\mathrm{G}]_t \approx [\mathrm{G}]_{t+\Delta t} + (-k_{\mathrm{GS}}[\mathrm{G}]_{t+\Delta t} + k_{\mathrm{SG}}[\mathrm{S}]_{t+\Delta t}[\text{Tu-GTP}]^3_{t+\Delta t})\Delta t$$

must be solved.

Apart from the backward Euler algorithm, there are implicit methods that use Taylor expansion series, predictor–corrector methods, or a combination to improve the accuracy of the estimates. Frequently used implicit algorithms are those formulated by Adams–Moulton and Gear (1971).

Implicit methods are of special interest to modelers of biochemical systems, because they tend to produce the most efficient stable solutions of so-called stiff systems of differential equations. A system is stiff when the concentrations of its various components change on very different time scales. In enzyme catalyzed reactions, for example, the concentrations of the various enzyme species may reach steady state in milliseconds, whereas substrate and product concentrations may take minutes or longer to respond. Explicit algorithms must take very small time steps to keep track of the fast reactions, even when the concentrations of their participants are hardly changing. In implicit algorithms, the computation is much less sensitive to these fast reactions, and as a result much larger time steps can be taken. Because implicit methods usually require the solution of a set of nonlinear algebraic equations, they are more difficult to implement, and computationally more expensive than explicit methods. However, for certain systems, the saving in

computational time can be significant because they require far fewer steps to produce a stable and accurate solution.

Most computer programs that are designed solve systems of coupled ODEs use, in addition to one or more of the above algorithms, methods that control the size of the time step, adapting it to an optimal value on the basis of estimates of its accuracy at the current stage of the simulation. Optimizing the step size also has a certain computational cost, but, again, may have improved overall efficiency. Some also adaptively control the order of the method (broadly speaking, the higher the order, the more substeps or previous steps are taken into account), and some packages even provide an option to automatically choose the most efficient algorithm.

3. Recipe for Performing Deterministic Simulations

Thus, the standard procedure for the stochastic simulation of a chemical reaction system comprises the following steps:

1. Combine the expressions for influx and efflux into ODEs that describe how the population of each state changes, given the overall state (i.e., the population of all states) of the system
2. Choose a start time t_0 and end time t_{end}. Set the concentrations or number of items in each of the states to their initial values, and set the current simulation time, t_{sim} to t_0
3. Based on the expressions for the population change in each state, and given the state of the system at t_{sim}, estimate the population of the states after a certain finite time interval Δt, using an appropriate algorithm, desired accuracy, and advanced features such as adaptive step size control
4. Set the population of all states to their estimated values for $t_{sim} + \Delta t$, and set t_{sim} itself to $t_{sim} + \Delta t$
5. Repeat from (3), until $t_{sim} + \Delta t$ exceeds t_{end}.

Even though there is a choice of computational tools that will carry out numerical integration, it is relatively easy to use a spreadsheet program to numerically solve a simple ODE (such as $dn/dt = k \times n$) with the forward Euler method. We recommend this as a starting point because it helps in developing a sound understanding of the underlying processes, and enables one to assess the influence of step size, and so on.

C. Comparison of Stochastic and Deterministic Simulation

Figure 4 also shows the average microtubule end behavior and Tu-GTP concentration obtained by numerical solution of the ODE system for the simple model used here. Because the system has a high degree of dependence, and because many events occur over the simulated period, the stochastic simulations are

computationally very intensive, particularly when they are used to obtain an average time course. Numerical integration is generally a much more efficient procedure for calculating average trajectories. However, the improved efficiency comes at the price of losing all information about the behavior of individual entities, especially about the level of dispersion, the deviation of individual trajectories from the average.

V. Modeling and Simulation in Practice

A. Modeling

Modeling is sometimes said to be art rather than science. Although there is ongoing research into automated model building, in practice creating a model is done "by hand," on the basis of a combination of intuition and experience. There are no hard and fast rules that can be applied to the process of model building, but there are certain issues, particularly with regard to the assessment of parameter values, that modelers of biochemical reaction systems need to take into consideration, and these are addressed in Section V.A.1.

There are, however, quite a few software tools that *facilitate* the building of biochemical reaction networks by providing the option to enter networks as sets of states, reactions, compartments, rate equations and associated functions, parameter and initial values, and allowing the user to use, modify, and store these data. A relatively new development, discussed in Section V.A.2 , is the specification of a standard format for storage and exchange of biochemical reaction network models.

1. Obtaining and Assessing Parameter Values

The numbers used in the examples in this chapter were taken from reports that describe in detail how, and under which conditions, the experiments that yielded these values were carried out, and which assumptions and equations were used to extract the values from the observed data. Extracting values from data often involves a significant amount of model construction and data fitting, and the assumptions that were made along the way may restrict the application of the results. When setting up a quantitative model of a system it is, therefore, crucial to be aware of the origin of the parameters that are obtained from the literature. Furthermore, parameter values may have been obtained under conditions that were different from those simulated in the models, and in other cases the available data may not contain any information about parameters that are required in the simulation. Therefore, it is often necessary to guess the values of certain parameters, and different sets of values may be tried to find the combination that "fits" the observations best. Parameter optimization is the subject of another chapter in this volume, but here we need to make the following point. Large and complex systems require large numbers of parameters to be estimated, and it is highly likely

that, for a given model, many combinations of parameters will describe the system equally satisfactorily. Therefore, it is advisable not to be overambitious, keep the number of unknowns as small as possible, and try to assess how much changes in individual parameters influence the overall behavior of the model system. It is also important to remember that the behavior of any particular model may be relatively or even totally insensitive to the value of particular parameters.

a. Limits to Rate Constants

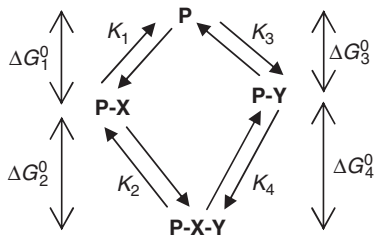
As noted above, it will frequently be necessary to guess the values of particular rate constants. There are no lower limits for rate constants and although essentially any value is possible the rate constants for slow steps must be selected with the required overall system response time in mind. In contrast, both first- and second-order rate constants have well-defined upper limits. The fastest reactions in biochemistry, energy, photon, electron, and proton transfer between well-positioned donors and acceptors, happen on femtosecond to nanosecond time scales and have rate constants of the order of 10^{13} to 10^{7} sec^{-1} (Zewail, 2003). All other processes are highly likely to proceed more slowly. Bimolecular reaction rates are limited by the diffusion rate of the reactants, and therefore depend on the viscosity of the medium. The maximum value for a second-order rate constant in water at 25 °C is estimated to be of the order of 10^{9} M^{-1} sec^{-1} (Atkins, 1994).

b. Overall Free Energy Conservation in Chains of Reversible Reactions

The equilibrium constant for a reversible reaction is determined by the free energy difference between the reactants and products. If two or more reversible reactions occur in sequence, the overall energy difference between the states at the two ends of the sequence is obtained by addition of the free energy differences for each of the individual steps. This sum corresponds in turn to an overall equilibrium constant, which is the product of the equilibrium constants for the individual steps:

$$\Delta G_1^\circ + \Delta G_2^\circ + \Delta G_3^\circ + \cdots + \Delta G_n^\circ = -RT \times \ln[K_1 \times K_2 \times K_3 \times \cdots \times K_n].$$

There are frequently two or more possible paths that lead from one end of the sequence to the other. For example, a ternary complex of a protein P and its ligands X and Y, P-X-Y, may be formed through two pathways, which, together with the associated free energy changes and equilibrium association constants, are indicated in Scheme 1. Because the free energy change must be independent of the pathway, $\Delta G_1^\circ + \Delta G_2^\circ$ must be equal to $\Delta G_3^\circ + \Delta G_4^\circ$, so that $K_1 \times K_2$ must be equal to $K_3 \times K_4$. Therefore, if any three of these constants are known then the fourth can be calculated. Two things should be noted. First, if the binding of X to P increases the affinity of P for Y ($K_2 > K_3$), then binding of Y to P must also increase the affinity of P for X ($K_4 > K_1$). Second, $K_1 \times K_2$ can only be not equal to $K_3 \times K_4$, if there is a continuous flux of material through the system. A model may



Scheme 1.

include many cycles of this type and careful attention must be given to ensure that all of constants used obey the appropriate relationship.

2. Standard Formats for Model Exchange and Storage

There are many software tools, proprietary or freely available, high or low level, than can be used to simulate biochemical reaction systems (Section V.B.3). Improved versions of existing tools and completely new ones are being released regularly, whereas others are no longer updated or maintained, and may become unusable. Obviously, all these programs allow their users to store the networks that have been created, together with parameter values, an initial (or current) state, and sometimes with simulation or other instructions. The file with the stored data can then be read into the same program at a later time, to be reused in new simulations, amended, and so on. Because each program has its own storage format, models that have been built using one tool cannot, in general, be read by another. This may become a problem when a tool ceases to function on the user's computer (for instance, after an operating system update) or when the user wants to use particular functions offered by another tool. Moreover, users may wish to store their models in a publicly accessible repository, so that they can be inspected and analyzed by others.

Attempts to create a standardized storage format have probably been undertaken several times, but two particular formats have recently become established and accepted as standards. CellML, the Cell Markup Language (Cuellar *et al.*, 2003; Lloyd *et al.*, 2004), is the oldest and has the widest scope of the two. SBML, the Systems Biology Markup Language (Hucka *et al.*, 2003, 2004), however, has the widest application base, with currently over a hundred tools supporting its format in one way or another. CellML and SBML are XML-based, which means that they adhere to a standard document format.[9] Both have their own rules for naming and defining states, reactions, compartments, and parameters, but use the same rules for expressing the associated math.[10] The CellML and SBML formats are text-based, and may be understood and written by humans, but are designed to be written, read, and interpreted primarily by machines. Further information is

[9] Specified in the XML (eXtended Markup Language) schema.
[10] Using MathML, the Mathematics Markup Language (XML-based).

found on the Web sites dedicated to these formats (CellML, 2006; SBML, 2006), and examples of stored models in the CellML repository (CellML Model Repository, 2006) and in the BioModels database (Le Novere *et al.*, 2006).

B. Simulation

Simulation, as opposed to modeling, is a mechanical process that lends itself very well to automation. In fact, simulation, even of relatively simple systems, would hardly be possible without computers.

1. Numerical Solution of ODEs

Today it is hardly ever necessary to write computer programs that implement algorithms for numerical solution of systems of coupled ODEs, such as the ones mentioned in Section IV.B.2. Sophisticated, well-tested software libraries are available (often freely), often packaged with routines that automatically select an appropriate algorithm, estimate simulation parameters such as accuracy and initial step size, and return the simulation results as lists of equidistant time points. It is still necessary to address the routines in these libraries in an appropriate programming language (see Section V.B.3), but, again, there are many software tools in which this has already been done. These tools offer an environment in which a user can enter ODEs or reaction networks in algebraic or graphical form, set parameter and initial values, perform simulations, and specify a desired form for the output. Examples are listed in Section V.B.3.

2. Stochastic Simulation

There are several tools that have been designed to perform general stochastic simulation of biochemical reaction networks (see Section V.B.3 for examples). However, the aim of a stochastic simulation may be to study the dynamics of relatively complex molecular assemblies that have important structural characteristics or other features that are not easily expressed in a standard way (e.g., see Chen and Hill, 1985; Duke, 1999; Janulevicius *et al.*, 2006; Martin *et al.*, 1993; Schilstra and Martin, 2006; VanBuren *et al.*, 2005). In this case, it is often easiest, sometimes essential, to use a high-level programming language to specify the system and its interactions, and to produce a purpose-built simulator. Fortunately, implementation of the recipe outlined in Section IV.A.3 is extremely simple; specification of the rest of the model may present a greater challenge.

3. Selected Simulation Tools

It is impossible to give an exhaustive list of software tools for simulation of biochemical reaction systems, and the selection presented here, which has been compiled partly on the basis of the software tools listed on the SBML Web site, may inadvertently have omitted some important or popular ones.

Simulations may be performed on the basis of a system specification in a compiled or interpreted high-level computer language, such as C/C++, Fortran, Java, Lisp, Perl, Python, or Visual Basic. There are software libraries that contain implementations of multiple ODE solvers (e.g., SUNDIALS, Hindmarsh *et al.*, 2005), which can be addressed from one or more of these languages. LibSBML (Bornstein, 2006) is a software library that assists software developers in the reading, writing, and validation of the SBML format, and has bindings to all of the above languages except Fortran and Visual Basic.

Mathematica, MATLAB, Maple (all proprietary), and R (free) are examples of high-level mathematics and statistics packages that can be used to solve ODEs, or to perform stochastic simulations. MathSBML (free, Shapiro *et al.*, 2004) is a Mathematica package, and SBMLToolbox (Keating *et al.*, 2006) and SBToolbox (Schmidt and Jirstrand, 2006) are free MATLAB toolboxes that can read, write, and apply the SBML format to build, store, use, and analyze models of biochemical reaction networks.

XPP (free, Ermentrout, 2002), Berkeley Madonna (Macey and Oster, 2001), Dymola (Dynasim, 2006), and ModelMaker (ModelKinetix, 2006) (all proprietary) are tools designed to facilitate the study of various types of dynamical systems, and can be applied to model, simulate, and analyze biochemical networks in various ways.

CellDesigner (Funahashi *et al.*, 2003; Kitano *et al.*, 2005), DBSolve (Goryanin *et al.*, 1999), Dizzy (Ramsey *et al.*, 2005), E-Cell (Takahashi *et al.*, 2003), Gepasi/COPASI (Mendes, 1997; Mendes and Kummer, 2006), Jarnac/JDesigner (Sauro, 2000), Promot/DIVA (Ginkel *et al.*, 2003), PySces (Olivier *et al.*, 2005), and Virtual Cell (Slepchenko *et al.*, 2003) are freely available tools that have been designed specifically for modeling and simulation of biochemical or reaction networks or physiological systems. Most execute deterministic simulation methods but Dizzy uses stochastic techniques, and COPASI implements both. All of these allow the user to enter a network as a collection of states, reactions, and reaction mechanisms, and automatically build the appropriate structures that are required for simulation. Several also allow direct entry of the ODEs. Some have user interfaces that allow the entry of models in graphical form; many provide extra tools to analyze aspects of the network structure and dynamic properties.

4. Selected Further Reading

All biochemistry textbooks (e.g., Berg *et al.*, 2000) have sections dedicated to the basics of thermodynamics and enzyme kinetics, whereas more specialized monographs present more comprehensive theoretical and practical background information (Cornish-Bowden, 2001; Fell, 1997). Textbooks on physical chemistry (Atkins, 1994) contain more theory on chemical reaction kinetics and thermodynamics than most people will ever need. Information on various algorithms that is reasonably accessible to nonspecialists is found in (Press *et al.*, 1989), and also on

several internet sites [e.g., Wikipedia (Wikipedia contributors, 2006), Mathworld (Weisstein, 2006)]. General theory on modeling and simulation is found in (Zeigler *et al.*, 2000), and on the use of diagrams in (Rumbaugh *et al.*, 1991). Last, but not least, several books have appeared recently which deal specifically with dynamic modeling in a biological context (Ellner and Guckenheimer, 2006; Szalasi *et al.*, 2006; Wilkinson, 2006).

VI. Concluding Remarks

We hope that we have provided the basic information required to understand the most important concepts in modeling and simulation. More information can be obtained from the recommended books and other literature. However, by far, the best way to develop further understanding is to set up a model of your favorite system, and try to simulate its behavior. You might be surprised at what you discover.

Acknowledgments

The preparation of this review was supported in part by grant 072930/Z/03/Z from the Wellcome Trust, United Kingdom, (to M.J.S.) and by grant R01 GM070923 from the National Institute for General Medical Sciences, USA (to S.M.K.).

References

Andrews, S. S., and Bray, D. (2004). Stochastic simulation of chemical reactions with spatial resolution and single molecule detail. *Phys. Biol.* **1,** 137–151.

Atkins, P. W. (1994). "Physical Chemistry." Oxford University Press, Oxford, UK.

Bayley, P. M., Schilstra, M. J., and Martin, S. R. (1989). A simple formulation of microtubule dynamics: Quantitative implications of the dynamic instability of microtubule populations *in vivo* and *in vitro*. *J. Cell Sci.* **93,** 241–254.

Berg, J. M., Tymoczko, J. L., and Stryer, L. (2002). "Biochemistry." Freeman and Company, New York, NY.

Bornstein, B. J. (2006). "LibSBML": SBML, The Systems Biology Markup Language. http://www.sbml.org/software/libsbml/

Bortz, A. B., Kalos, M. H., and Lebowitz, J. L. (1975). A new algorithm for Monte Carlo simulation of Ising spin systems. *J. Comput. Phys.* **17,** 10–18.

Brylawski, B. P., and Caplow, M. (1983). Rate for nucleotide release from tubulin. *J. Biol. Chem.* **258,** 760–763.

CellML. (2006). "The CellML Project: Overview—Portal." http://www.cellml.org

CellML Model Repository. (2006). "Model Repository—Portal." http://www.cellml.org/models

Chen, Y.-D., and Hill, T. L. (1985). Monte Carlo study of the GTP cap in a five-start helix model of a microtubule. *Proc. Natl. Acad. Sci. USA* **82,** 1131–1135.

Cornish-Bowden, A. (1995). "Fundamentals of Enzyme Kinetics." Portland Press, London.

Cornish-Bowden, A. (2001). "Fundamentals of Enzyme Kinetics." Portland Press, London, UK.

Cuellar, A. A., Lloyd, C. M., Nielsen, P. F., Bullivant, D. P., Nickerson, D. P., and Hunter, P. J. (2003). An overview of CellML 1.1, a biological model description language. *Simulation* **79,** 740–747.

Duke, T. A. J. (1999). Molecular model of muscle contraction. *Proc. Natl. Acad. Sci. USA* **96,** 2770–2775.

Dynasim, A. B. (2006). "Dymola." Dynasim, http://www.dynasim.se

Ellner, S. R., and Guckenheimer, J. (2006). "Dynamic Models in Biology." Princeton University Press, Princeton, NJ.

Engelborghs, Y., and Eccleston, J. (1982). Fluorescence stopped-flow study of the binding of S6-GTP to tubulin. *FEBS Lett.* **141,** 78–81.

Ermentrout, B. (2002). "Simulating, Analyzing, and Animating Dynamical Systems: A Guide to XPPAUT for Researchers and Students." SIAM, Philadelphia, PA.

Fell, D. (1997). "Understanding the Control of Metabolism." Portland Press, London.

Funahashi, A., Tanimura, N., Morohashi, M., and Kitano, H. (2003). CellDesigner: A process diagram editor for gene-regulatory and biochemical networks. *Biosilico* **1,** 159–162.

Gear, C. W. (1971). "Numerical Initial Value Problems in Ordinary Differential Equations." Prentice-Hall, Englewood Cliffs, NJ.

Gibson, M. A., and Bruck, J. (2000). Efficient exact stochastic simulation of chemical systems with many species and many channels. *J. Phys. Chem. A.* **104,** 1876–1889.

Gillespie, D. T. (1977). Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.* **81,** 2340–2361.

Gillespie, D. T. (2001). Approximate accelerated stochastic simulation of chemically reacting systems. *J. Chem. Phys.* **115,** 1716–1733.

Gilman, A., and Arkin, A. P. (2002). Genetic "code": Representations and dynamical models of genetic components and networks. *Annu. Rev. Genom. Hum. Genet.* **3,** 341–369.

Ginkel, M., Kremling, A., Nutsch, T., Rehner, R., and Gilles, E. D. (2003). Modular modeling of cellular systems with ProMoT/Diva. *Bioinformatics* **19,** 1169–1176.

Goryanin, I., Hodgman, T., and Selkov, E. (1999). Mathematical simulation and analysis of cellular metabolism and regulation. *Bioinformatics* **15,** 749–758.

Hindmarsh, A. C., Brown, P. N., Grant, K. E., Lee, S. L., Serban, R., Shumaker, D. E., and Woodward, C. S. (2005). SUNDIALS: Suite of nonlinear and differential/algebraic equation solvers. *ACM Trans. Math. Software* **31,** 363–396.

Horio, T., and Hotani, H. (1986). Visualization of the dynamic instability of individual microtubules by dark-field microscopy. *Nature* **321,** 605.

Hucka, M., Finney, A., Sauro, H. M., Bolouri, H., Doyle, J. C., Kitano, H., and the rest of the SBML Forum, Arkin, A. P., Bornstein, B. J., Bray, D., Cornish-Bowden, A., Cuellar, A. A., Dronov, S., *et al.* (2003). The systems biology markup language (SBML): A medium for representation and exchange of biochemical network models. *Bioinformatics* **19,** 524–531.

Hucka, M., Finney, A., Bornstein, B., Keating, S. M., Shapiro, B. E., Matthews, J., Kovitz, B., Schilstra, M. J., Funahashi, A., Doyle, J. C., and Kitano, H. (2004). Evolving a lingua franca and associated software infrastructure for computational systems biology: The systems biology markup language (SBML) project. *Syst. Biol.* **1,** 41–53.

Janulevicius, A., van Pelt, J., and van Ooyen, A. (2006). Compartment volume influences microtubule dynamic instability: A model study. *Biophys. J.* **90,** 788–798.

Keating, S. M., Bornstein, B. J., Finney, A., and Hucka, M. (2006). SBMLToolbox: An SBML toolbox for MATLAB users. *Bioinformatics* **22**(10), 1275–1277; doi:10.1093/bioinformatics/btl111.

Kitano, H., Funahashi, A., Matsuoka, Y., and Oda, K. (2005). Using process diagrams for the graphical representation of biological networks. *Nat. Biotechnol.* **23,** 961–966.

Kruse, K., and Elf, J. (2006). Kinetics in Spatially Extended Systems. *In* "Systems Modeling in Cellular Biology" (Z. Szalasi, J. Stelling, and V. Periwal, eds.), pp. 177–198. MIT Press, Cambridge, MA.

Le Novere, N., Bornstein, B., Broicher, A., Courtot, M., Donizelli, M., Dharuri, H., Li, L., Sauro, H., Schilstra, M., Shapiro, B., Snoep, J. L., and Hucka, M. (2006). BioModels database: A free, centralized database of curated, published, quantitative kinetic models of biochemical and cellular systems. *Nucl. Acids Res.* **34,** D689–D691.

Lloyd, C. M., Halstead, M. D. B., and Nielsen, P. F. (2004). CellML: Its future, present and past. *Prog. Biophys. Mol. Biol.* **85,** 433–450.

Macey, R. I., and Oster, G. F. (2001). "Berkeley Madonna." http://www.berkeleymadonna.com

Markevich, N. I., Hoek, J. B., and Kholodenko, B. N. (2004). Signaling switches and bistability arising from multisite phosphorylation in protein kinase cascades. *J. Cell Biol.* **164,** 353–359.

Martin, S. R., Schilstra, M. J., and Bayley, P. M. (1993). Dynamic instability of microtubules: Monte-Carlo simulation and application to different types of microtubule lattice. *Biophys. J.* **65,** 578–596.

Mendes, P. (1997). Biochemistry by numbers: Simulation of biochemical pathways with Gepasi 3. *Trends Biochem. Sci.* **22,** 361–363.

Mendes, P., and Kummer, U. (2006). "COPASI." http://www.copasi.org/tiki-index.php

Meyers, J., Craig, J., and Odde, D. J. (2006). Potential for control of signaling pathways via cell size and shape. *Curr. Biol.* **16,** 1685–1693.

ModelKinetix. (2006). "ModelMaker." http://www.modelkinetix.com/modelmaker/index.htm

Olivier, B. G., Rohwer, J. M., and Hofmeyr, J.-H. S. (2005). Modelling cellular systems with PySCeS. *Bioinformatics* **21,** 560–561.

Press, W. H., Flannery, B. P., Teukolsky, B. P., and Vetterling, W. T. (1989). "Numerical Recipes: The Art of Scientific Computing. Fortran Version." Cambridge University Press, Cambridge, UK.

Ramsey, S., Orrell, D., and Bolouri, H. (2005). Dizzy: Stochastic simulation of large-scale genetic regulatory networks. *J. Bioinform. Comput. Biol.* **3,** 415–436.

Rumbaugh, J., Blaha, M., Premerlani, W., Eddy, F., and Lorensen, W. (1991). "Object-Oriented Modelling and Design." Prentice-Hall, London.

Sauro, H. M. (2000). "Animating the Cellular Map. 9th International BioThermoKinetics Meeting" (J.-H. S. Hofmeyr, J. M. Rohwer, and J. L. Snoep, eds.), pp. 221–228, Stellenbosch University Press, Stellenbosch, South Africa.

SBML. (2006). "SBML.org—The home site for the Systems Biology Markup Language." http://sbml.org.

Schilstra, M. J., and Martin, S. R. (2006). An elastically tethered viscous load imposes a regular gait on the motion of myosin-V. Simulation of the effect of transient force relaxation on a stochastic process. *J. R. Soc. Interf.* **3,** 153–165.

Schmidt, H., and Jirstrand, M. (2006). Systems biology toolbox for MATLAB: A computational platform for research in systems biology. *Bioinformatics* **22,** 514–515.

Segel, I. H. (1975). "Enzyme Kinetics: Behavior and Analysis of Rapid Equilibrium and Steady-State Enzyme Systems." John Wiley & Sons Inc., USA.

Shapiro, B. E., Hucka, M., Finney, A., and Doyle, J. C. (2004). MathSBML: A package for manipulating SBML-based biological models. *Bioinformatics* **20,** 2829–2831.

Slepchenko, B. M., Schaff, J., Macara, I. G., and Loew, L. M. (2003). Quantitative cell biology with the virtual cell. *Trends Cell Biol.* **13,** 570–576.

Swade, D. (1995). When money flowed like water *Inc. Magazine* September 1995. http://pf.inc.com/magazine/19950915/2624.html

Szalasi, Z., Stelling, J., and Periwal, V. (eds). (2006). "Systems Modeling in Cellular Biology." MIT Press, Cambridge, MA.

Takahashi, K., Ishikawa, N., Sadamoto, Y., Ohta, S., Shiozawa, A., Miyoshi, F., Naito, Y., Nakayama, Y., and Tomita, M. (2003). E-Cell 2: Multi-platform E-cell simulation system. *Bioinformatics* **19,** 1727–1729.

VanBuren, V., Cassimeris, L., and Odde, D. J. (2005). Mechanochemical model of microtubule structure and self-assembly kinetics. *Biophys. J.* **89,** 2911–2926.

Weisstein, E. (2006). "Mathworld". Wolfram Research, http://mathworld.wolfram.com/

Wikipedia Contributors. (2006). "Wikipedia." Wikipedia, The Free Encyclopedia, http://en.wikipedia.org

Wilkinson, D. J. (2006). "Stochastic Modelling for Systems Biology." Chapman & Hall/CRC, London.

Zeigler, B. P., Praehofer, H., and Kim, T. G. (2000). "Theory of Modelling and Simulation: Integrating Discrete Event and Continuous Complex Dynamic Systems." Academic Press, San Diego.

Zewail, A. (2003). "Nobel Lectures, Chemistry 1996–2000" (I. Grenthe, ed.), pp. 274–367. World Scientific Publishing Co., Singapore.