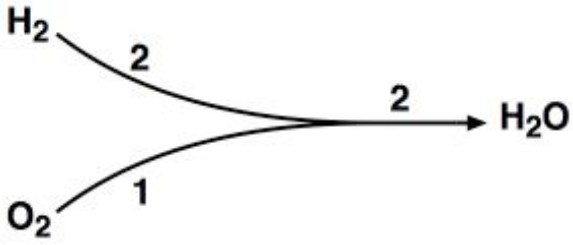
CHEMICAL SYSTEMS

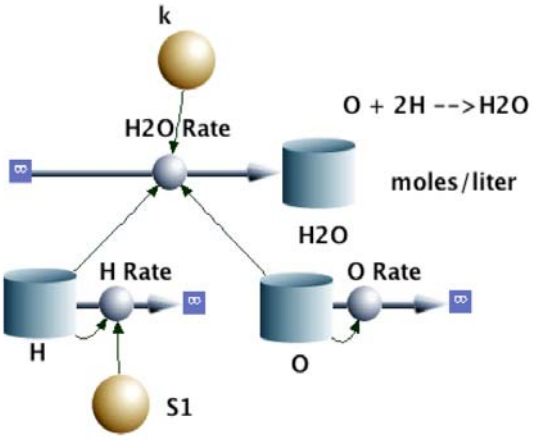
Introduction

In this chapter we will use Berkeley Madonna to model systems of chemical reactions. For simple systems the Flowchart is an adequate interface. However, for systems with more than a few reactions the equation interface quickly becomes easier and faster.

If you try and construct the Flowchart for the water formation reaction above, you will see that, even for this simple reaction, it is cumbersome and not very intuitive. This is because each reactant has its own reservoir and flow, but they cannot be interconnected into an intuitive diagram such as this one→



This is a limitation of the Berkeley Madonna Flowchart which is designed to keep track of conserved quantities such as mass or number (e.g. moles). But chemical reactions convert one substance to another according to the stoichiometric coefficients, and so *mole numbers are not conserved* except in the special case of unimolecular reactions with unit stoichiometric coefficients. However, such models are an important subclass which we will deal with first. In the Appendices we use the bimolecular reaction A + B ↔ C to illustrate the difficulties in modeling reactions with the Flowchart.



Linear reaction chains: Markov models

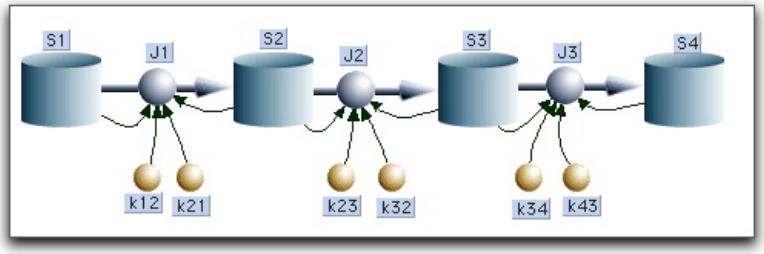
A commonly encountered situation is a chain of reservoirs where transitions, or flows, are unimolecular and have unit stoichiometry:

Equation 1.

$$S_1 \underset{k_{-1}}{\overset{k_1}{\rightleftharpoons}} S_2 \underset{k_{-2}}{\overset{k_2}{\rightleftharpoons}} \cdots \underset{k_{-2}}{\overset{k_2}{\rightleftharpoons}} S_N$$

The Flowchart corresponding to Equation 1 is shown in Figure 1 along with the equations of motion.

Exercise. GRAPH THE SOLUTION FOR THE 4-STATE REACTION CHAIN IN Figure 1.



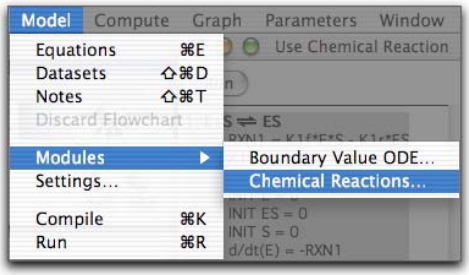
{Reservoirs} d/dt (S1) = - J1 INIT S1 = 10 d/dt (S2) = + J1 - J2 INIT S2 = 0 d/dt (S3) = + J2 - J3 INIT S3 = 0 d/dt (S4) = + J3 INIT S4 = 0	{Flows} J1 = k12*S1-k21*S2 J2 = k23*S2-k32*S3 J3 = k34*S3-k43*S4 {Functions} k12 = 2 k23 = 2 k34 = 2 k21 = 1 k32 = 1 k43 = 1

Figure 1. Linear chain of compartments modeled by Equation 1.

Linear chains of compartments can be used to model many situations other than chemical reaction chains. For long chains, the Flowchart is unwieldy; however, we will see later that Berkeley Madonna's **Array** notation makes handling long chains easy.

Using the chemical reaction module

Flowcharts for complex reaction schemes are confusing, so we need a better way of generating the model equations. For modeling systems of mass action kinetics, Berkeley Madonna has a simple interface that enables one to set up and solve complicated reaction schemes quickly and easily using ordinary chemical notation.



1. Open the Chemical Reactions dialog by selecting it from the **Model** menu: **Modules** popup (Figure 6a).
2. Using ordinary chemical notation, enter the first reaction in the Reactants: and Products: boxes (Figure 6b).
3. Enter numerical values for the forward rate constant, **Kf**, and the reverse rate constant, **Kr**, as shown.
4. Click **Add**, and the reaction will be recorded in the **Reactions**: list.
5. Enter the **Initial Concentrations**.
6. Click **OK** and the equations appear in the Equation window.

You can use the check boxes in the dialog box to control how much of the model equations are displayed in the Equation window. Note that the equations are highlighted indicating that you cannot edit them directly. However, double-clicking in the highlighted area re-opens the dialog box for editing. When a reaction is changed it is entered with the **Modify** button. A summary of the Chemical Reaction Module procedure is always available under the **Help** > **How do I…** menu.

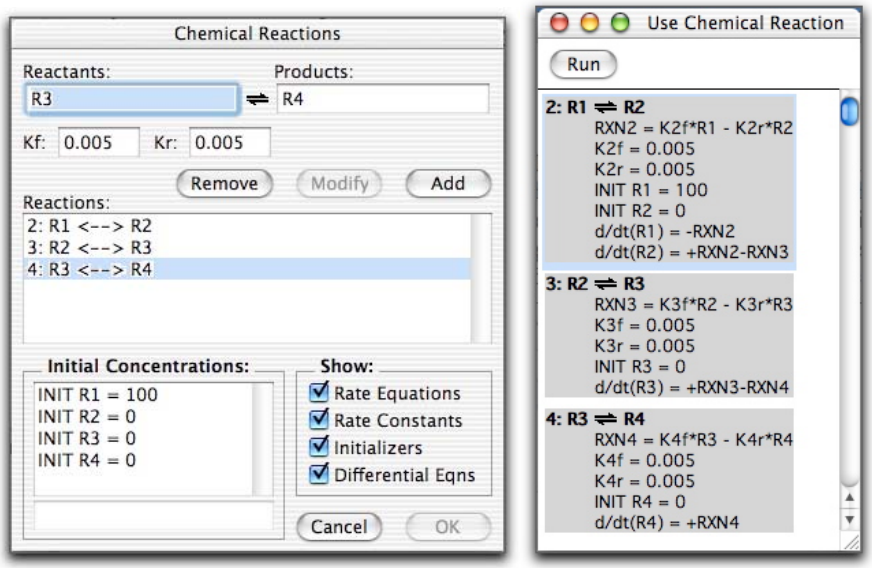


Figure 2. Chemical reaction module dialog box and equations.

Exercise: Simulate the 4-compartment reaction chain for 700 time units with an initial condition of 100 in the first compartment and the other compartments initially empty, as shown in Figure 2. Set all reaction rates = 0.005.

Introduction to parameter plots

Consider the simple function y = ymax*x/(b + x), ymax = 1, b = 1. You can easily plot this in Berkeley Madonna by typing in the Equation window as shown in Figure 3a. You could investigate how the shape of the curve changes when you vary, say, the parameter b by constructing a Slider. However, you can summarize many manipulations of the slider by making a plot of, say, the maximum value of the curve (ymax) as a function of the parameter b. Go to the Parameter Plot… dialog under the Parameters window and choose b as the parameter, and the maximum value of y as the value plotted (see Figure 3b).

Exercise. Plot the logistic equation of population growth: $dn/dt = an - bn^2$, where n(0) = 1, using sliders to vary a and b. Explain what happens when b > a. Now make a parameter plot of the final population for $1 \le b \le 10$ when a = 5.

Optional: Look up the DELAY function in the Equation Help and write the delayed logistic equation $dn_d/dt = an_d(b - n_d(t - T)^2)$. Make a slider to vary the time delay, T, and watch what happens as T increases from zero. Make a parameter plot of the oscillation amplitude as a function of the delay, T.

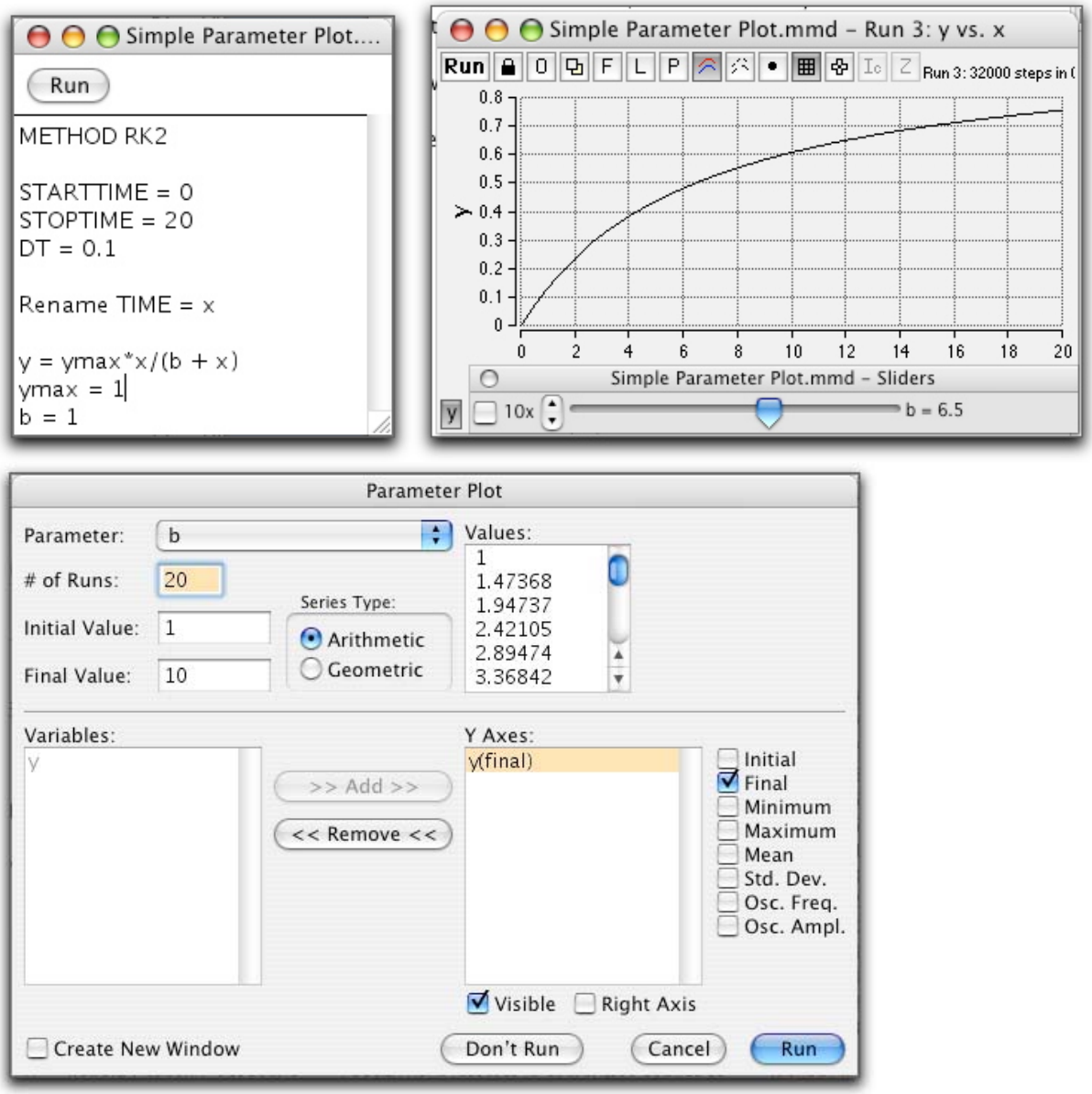


Figure 3. Plotting a function and making a Parameter plot.

Cooperative Binding of Oxygen to Hemoglobin

Binding of O_2 to hemoglobin (Hb) of red blood cells increases the O_2 carrying capacity of the blood by a factor of nearly 70. However, the binding properties must be delicately balanced so that Hb binds O_2 in the lungs and releases it in the tissues. The following model captures some of the major characteristics of this reversible binding reaction. We simulate the time course of binding when totally deoxygenated Hb is suddenly exposed to a constant concentration of O_2. In this case the hemoglobin binds four O_2 molecules sequentially:

$$Hb + O_2 \underset{k_{rf}}{\overset{k_{1f}}{\rightleftarrows}} HbO_2 \qquad HbO_2 + O_2 \underset{k_{2r}}{\overset{k_{2f}}{\rightleftarrows}} HbO_4$$

$$HbO_4 + O_2 \underset{k_{3f}}{\overset{k_{3f}}{\rightleftarrows}} HbO_6 \qquad HbO_6 + O_2 \underset{k_{4f}}{\overset{k_{4f}}{\rightleftarrows}} HbO_8$$

Exercise. Use the Chemical reaction module to construct the model for this process using the measured parameter values shown in Table 1

k_f [Mol^{-1}•msec^{-1}]	k_r [msec^{-1}]	Residence time [msec]
k_{f1} = 2,100	k_{r1} = 0.03	$\tau_{Hb} = \left(k_{f1}[O_2]\right)^{-1} = 47.6$
k_{f2} = 8,400	k_{r2} = 0.06	$\tau_{HbO2} = \left(k_{r1} + k_{f2}[O_2]\right)^{-1} = 8.8$
k_{f3} = 12,600	k_{r3} = 0.09	$\tau_{HbO4} = \left(k_{r2} + k_{f3}[O_2]\right)^{-1} = 5.4$
k_{f4} = 40,000	k_{r4} = 0.04	$\tau_{HbO6} = \left(k_{r3} + k_{f4}[O_2]\right)^{-1} = 2.0$
…	…	$\tau_{HbO8} = \left(k_{r4}\right)^{-1} = 25.0$

Table 1 Parameters for Hb-O_2 dissociation. Residence time τ is calculated for $[O_2]$ = 10^{-5} M

Note that the Reaction module inserts O_2 as a *variable*; to keep it constant, insert the equation d/dt (O_2) = 0 *after* the module equations. (Berkeley Madonna sets all quantities to the last values entered in the Equation window.) Use INIT O_2 = 10^{-5}, INIT Hb = 10^{-4}. At the beginning of the program use the following integration scheme and variables: METHOD STIFF, STARTTIME = 0, STOPTIME = 500, DTMIN = 1e-6, DTMAX = 1, TOLERANCE = 0.01.

Plot the O_2 dissociation curve by running a **Parameter Plot** with INIT O_2 on the x axis from O_2 = 10^{-6} to 3×10^{-5}. Use about 30 points spaced *geometrically*, and Plot FINAL values on the y-axis.

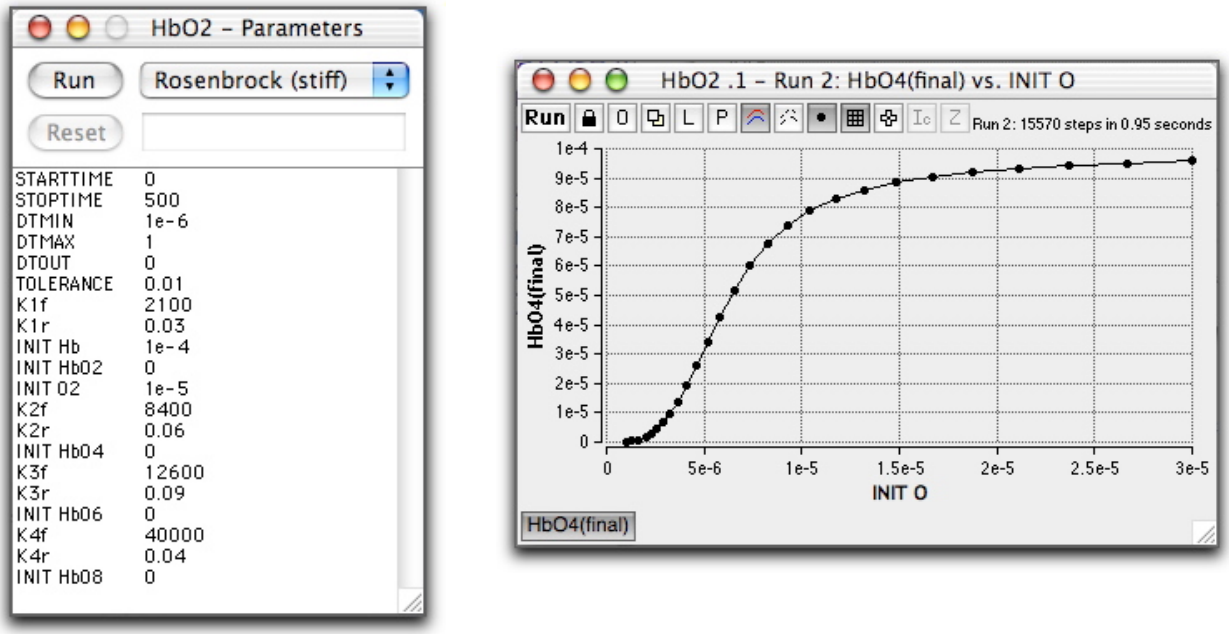


Figure 4. Hemoglobin model (a) Parameter window. (b) O_2 dissociation curve of FINAL HbO_4 vs. INIT O_2.

The O_2 dissociation curve reflects the cooperative binding of O_2: binding of each O_2 molecule induces a conformational change making binding easier for the next molecule. The curve is flat at very low O_2 concentrations because binding the first oxygen is difficult. Binding the second, third and final oxygens are progressively easier, so that the curve changes rapidly within the physiological range of O_2 concentrations encountered in arterial and venous blood. Consequently, Hb serves as an effective buffer for the O_2 concentration in the blood. This cooperative behavior is apparent from the model parameters in Figure 4: the magnitudes of the rate constants for the forward reactions, increase progressively as more O_2 molecules bind.

The parameter table also shows that the intermediates, HbO_2, HbO_4,and HbO_6, have short lifetimes (residence times $\tau \sim 1/k_f$) when compared to the initial, Hb, and final, HbO_8, reactants. Moreover, plots of these intermediates against time shows that their concentrations rarely rise to significant proportions. These considerations suggest the reaction scheme can be approximated by ignoring the intermediates and, since the entire reaction is fast, assuming the whole system is in equilibrium i.e. $nO_2 + Hb \leftrightarrow HbO_8$

where n = 4. The solution to this chemical equilibrium is:

Equation 2 $$[HbO_8] = [HbO_8]_{\max} \frac{[O_2]^n}{K_{50}{}^n + [O_2]^n}$$

The validity of this approximation can be evaluated by curve fitting Equation 2 to the results of the original simulation. To do this follow these steps:

1. Activate the graph window showing the O_2 dissociation curve obtained by the parameter plot. Use the Chemical Reaction module using the parameter values in Table 1.

2. Click on the **Table** button on the graph toolbar to obtain a numerical tabulation of the simulation results. The table should contain only two columns: (INIT O2, HbO_8). If there are other data on the table remove them.
3. Save the table. (Select (*File >Save Table as…*)
4. Close the model and open a *New* model (Select *File> New*)
5. Import the tabular data that you just saved as a new dataset (Select *File> Import Dataset*)
6. Type Equation 2 into the equation window (omit the square "concentration" brackets). Use the RENAME command to re-label the x-axis to *O2.*
7. Set STARTTIME = 1e-6, STOPTIME =3e-5, n=4, HbO_{max} = 1e-5, and K_{50} = 1e-5. Find the best fit for the parameters $HbO_{max,}$ and $K_{50.}$

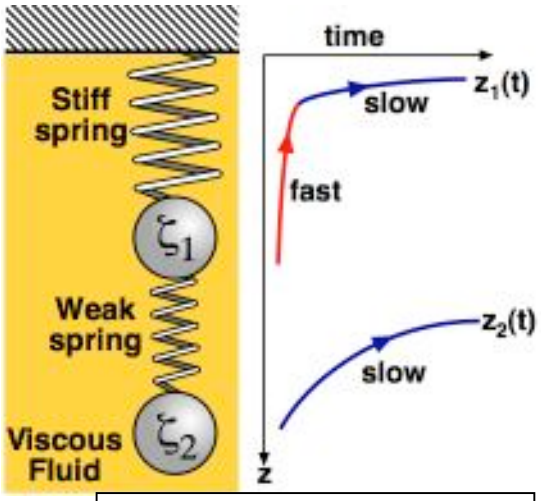
The fit is a fairly reasonable approximation. The advantage of using it over the original scheme is that we have an explicit expression for the result, so that the number of computations is dramatically reduced. Most importantly, the number of unknown parameters has been reduced from eight to two!

An even better approximation is suggested by noting that the plot of Equation 2 clearly has a long delay, and then rises faster than the original data. In other words the sigmoid shape of the approximation is more pronounced than the data. This indicates that the cooperativity within the original reaction scheme is not complete. We can accommodate this feature by relaxing our restriction on the parameter n and allowing it's value to be determined by the data. To do this, refit Equation 2 to the results, but this time allow all three parameters $HbO_{max,}$ K_{50}, and n to vary. This fit is excellent with a value of n = 3.2. In this type of reaction, the fitted value n, is called the **Hill Coefficient** and is often used as a measure of a reaction's cooperativity; the larger the value, the more the cooperativity (with n = 1 there is no cooperativity). Of course, the moment we depart from $n \approx 4$, we are abandoning the model. In this case ($n \neq 4$) becomes an empirical equation that is very useful for describing cooperative reaction data, but its physical interpretation is lost.

Although our results capture the primary features of the O_2 dissociation curve, it should be noted that the rate constants were determined using a purified preparation of sheep Hb at room temperature (15–20°C) and at pH 9.3. Therefore, we cannot expect the results to apply directly to physiological conditions of oxygen transport in normal mammals. In fact, the oxygen affinity predicted by these rate constants is much too strong so that the whole curve is shifted to the left. Hb under these conditions would not work; it would pick up O_2 in the lungs but wouldn't release it in the tissues. Raising the temperature, increasing the acidity, and incorporating normal amounts of DPG (2,3, diphophoglyceraldehyde) in the blood all lower the O_2 affinity substantially, pushing the dissociation curve back towards normal. It should also be noted that our model has not taken account of the numerous conformational changes (usually denoted 'Tense' and 'Relaxed') that occur with each stage of binding.

Michaelis-Menten kinetics: Fast and slow variables

'Stiff' systems



One of the most important methods of simplifying a complicated system is to separate the processes into 'fast' and 'slow' variables. The term arose from mechanical systems where 'stiff' springs respond very fast, while weak springs respond much more slowly. In the system shown to the right, two bodies are suspended in a very viscous fluid by stiff and floppy springs. (The viscous drag force on each is ζv_1 and ζv_2, where $v_{1,2} = dz_{1,2}/dt$ is the vertical velocity.) Then the displacement, $z_1(t)$ of the upper body, can be broken into an initial 'fast' regime, followed by a long-time 'slow' regime.

In many situations we care only about the long-time behavior, not the initial transient,. This is frequently the case in biochemical systems, where some reactions are fast and others are slow. The classical example is the Michaelis-Menten system, which we study below.

$$\zeta_1 \frac{dz_1}{dt} = -K_1\left(z_1 - z_1^0\right) + K_2\left(z - z_1 - z_2^0\right)$$
$$\zeta_2 \frac{dz_2}{dt} = -K_2\left(z_2 - z_1 - z_2^0\right)$$
$$k_1 \equiv K_1/\zeta_1, \qquad k_2 \equiv K_2/\zeta_2$$

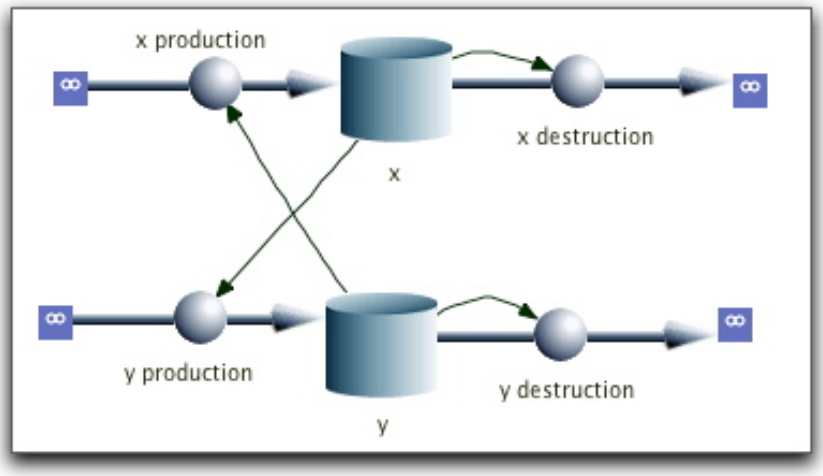
Exercise. Consider the simple system:

d/dt (x) = y - 2*x

INIT x = 1

d/dt (y) = -100*(y - x)

INIT y = 0



We can rewrite the equation for y as: $\tau \cdot dy/dt = x - y$, where $\tau = 1/100$ is the *time constant*. Because τ is small, y(t) rises very fast at first, then settles into the same decay rate as x.

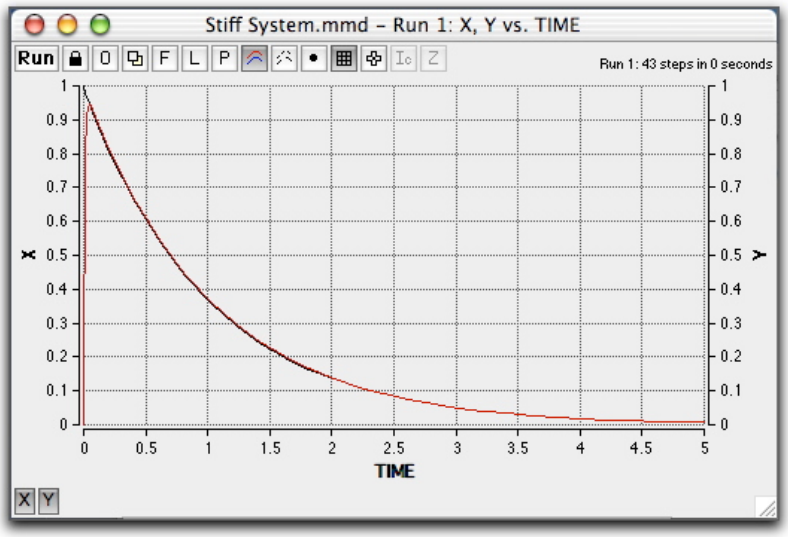


Figure 5. A 'stiff' system. The 'stiff' variable, y, has an initial rapid rise, while the 'soft' variable, x, dominates the long-time motion.

Use the STIFF solver with DTMIN = 0.01 and DTMAX = 0.5 to solve the system. Notice that y(t) rises very fast at the beginning (zoom in and see!), and thereafter changes slowly, along with x(t).

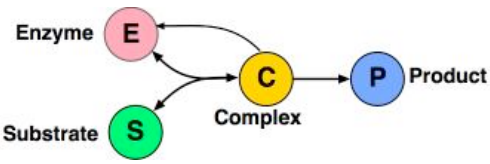
Suppose y changed so rapidly that it was always near its steady state: dy/dt = 0. Then you could solve the second equation for y = x and substitute it into the first equation for x, so that it became dx/dt = -2x+x = -x. To see how good this approximation is, define a third variable, z by dz/dt = -x, x(0) = 0, and plot it along with x and y.

Now solve the same system using the Euler method: in the Parameter window, select EULER from the drop-down menu. Show the data points by clicking the (•) button on the Graph window and see how many more steps Berkeley Madonna has to take to get a solution.

The STIFF (Rosenbrock) algorithm deals with the problem of fast and slow variations by looking at the slope of the solution. If it is large, then the step size is decreased, and if the slope is small (so that the solution isn't changing very fast) then the step size increases. There is no hard and fast 'rule' of when a stiff solver is better. If you see that there are fast and slow variables, try it out to see if it decreases the computation speed and/or the number of step.

Enzyme catalyzed reactions can be stiff

Simple enzyme catalyzed reactions obey the Michaelis-Menten (MM) kinetic equations:



Equation 3 $$E + S \underset{k_{-1}}{\overset{k_1}{\rightleftarrows}} C \xrightarrow{k_2} E + P$$

where E is the enzyme, S the substrate, P the product, and C ≡ E·S, the 'enzyme-substrate complex'. Many other kinetic systems obey the same reaction scheme; for example:

Membrane Transport Carrier, C: $C_{memb} + S_{in} \underset{k_{-1}}{\overset{k_1}{\rightleftharpoons}} C_{memb}S \xrightarrow{k_2} C_{memb} + S_{out}$

Ligand-Receptor Binding: $R + S \underset{k_{-1}}{\overset{k_1}{\rightleftharpoons}} RS \xrightarrow{k_2} Action$

Exercise: Use the Chemical Reaction module to generate the differential equations for the MM reaction scheme:

Equation 4 (a) Enzyme: $\frac{dE}{dt} = k_{-1}C + k_2C - k_1E \cdot S$ (c) Complex: $\frac{dC}{dt} = k_1E \cdot S - (k_{-1} + k_2)C$

(b) Substrate: $\frac{dS}{dt} = k_{-1}C - k_1E \cdot S$ (d) Product: $\frac{dP}{dt} = k_2C$

Not all of the differential equations are independent. Since the total amount of enzyme is constant: $E + C = E_{Total}$; so C can be eliminated from the equations . Also, the last equation for P(t) is completely determined once C is known, so it is superfluous. Therefore, the four differential equations can be reduced to only two *independent* differential equations.

Exercise: Solve the equations using the parameters given in Figure 6 using the Euler and Runge-Kutta methods with DT < 0.01. Then switch to the Rosenbrock (STIFF) solver with DTMIN = 1e-6, DTMAX = 1, DTOUT = 0, and TOLERANCE = 1e-4 (Press 1997). Use the Show Data button (•) on the graph to compare how efficient this integration method is to the Euler and Runge-Kutta. Berkeley Madonna shows how many time steps it took to solve the equations in the upper left of the graph.

The graph of the solution shown in Figure 6 shows that *there are two 'time scales'*: (i) a fast time scale wherein C is in a steady state at all levels of S (Figure 6c), and (ii) a slow time scale characterizing the appearance of the product (Figure 6d).

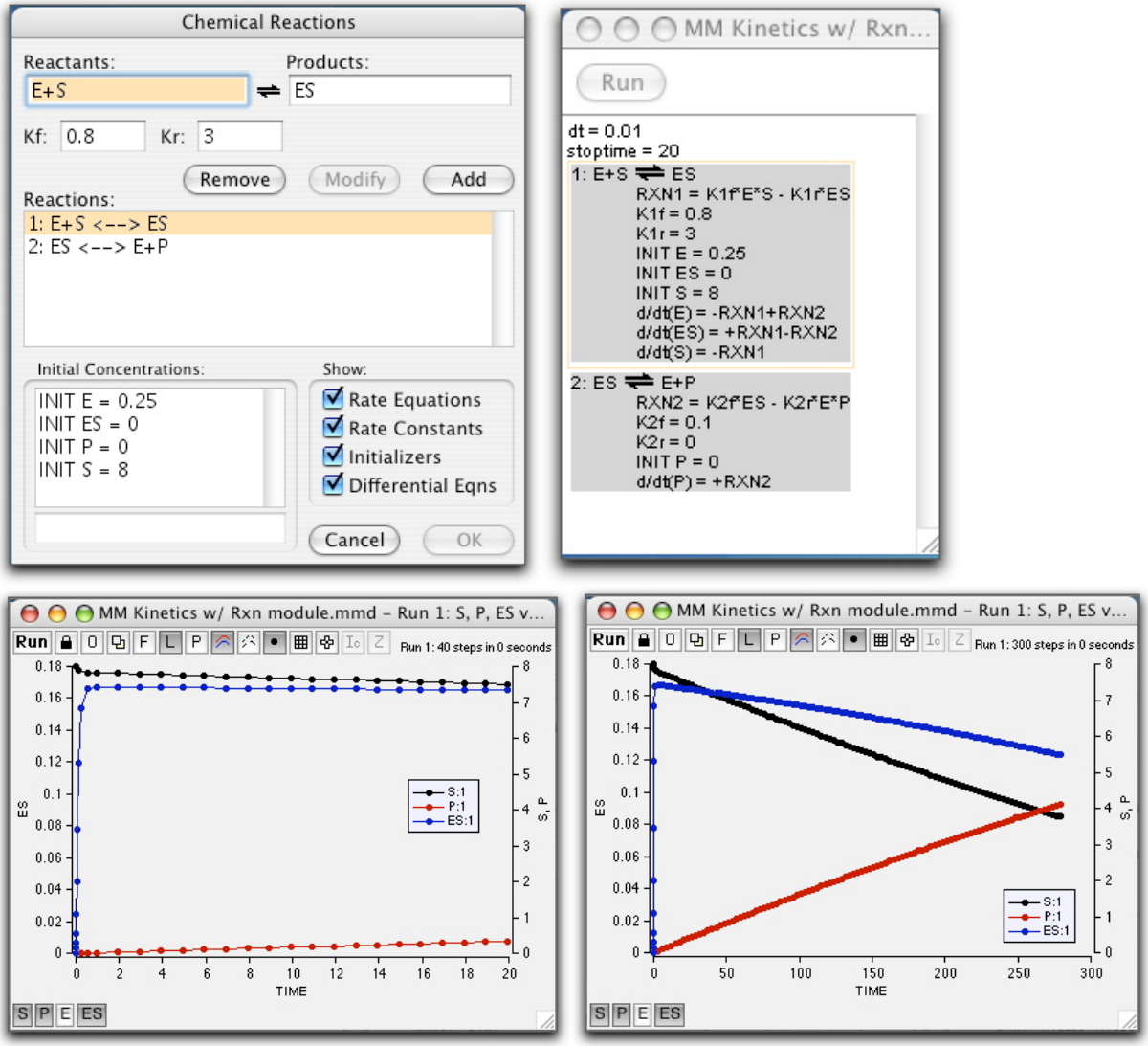


Figure 6. The Michaelis-Menten equations using the chemical reaction module. (a) Module window, (b) Equations; (c) Short time graph showing the initial rapid rise of the complex C = ES and the gradual rise of P and fall of S. (d) Long time plot showing the rise of P and fall of S; C is also falling gradually as substrate is consumed.

When the concentration of the complex is nearly constant, and we can set $dC/dt \approx 0$. Then C can be eliminated so that the system reduces to a single equation for the velocity of the reaction, $v = -dS/dt = dP/dt$:

Equation 5

$$v = \frac{v_{max} S}{K_m + S}$$

where $v_{max} \equiv k_2 E_{total}$, $K_m = (k_{-1} + k_2)/k_1$.

The function $v(S)$ is the Michaelis-Menten hyperbola. K_m is the *Michaelis constant.* It approximates the dissociation constant only when $k_{-1} >> k_2$. The value of S when the velocity of

the reaction is half its maximum, V_{max}, and the initial slope of the $v(S)$ curve is V_{max}/K_m. K_m is a dissociation constant: $K_m = k_{release}/k_{capture}$.[1] One can interpret V_{max}/K_m as an *effective rate of capture* of substrate by the enzyme (Northrop 1998; Northrop 1999).

Exercise: Use Madonna to plot the MM function by writing the equation directly into the Equation Window and setting S = TIME (or RENAME TIME = S). Use the BATCH mode so that you can plot several different curves with different parameters. Start by holding K_m constant at 10 mM for different values of V_{max}, and show that $v = v_{max}/2$ only when $S = K_m$. Next hold V_{max} fixed at 10 and allow K_m to vary. The higher the K_m the slower the rise in V—i.e. the longer it takes to get half way. In fact, $K_m = S_{1/2}$ = concentration when the reaction velocity is $V_{max}/2$. To see how K_m affects the shape of the curve, (i) make a slider to vary K_m, (ii) Use the BATCH RUNS option make a parameter plot with various K_m.

Exercise: In practice, biochemists determine the MM function by plotting the initial appearance of product. Plot the reaction velocity (i.e. the flow into the P reservoir, Reaction 2) for 30 values of the initial substrate concentration (INIT S) (see Figure 7).

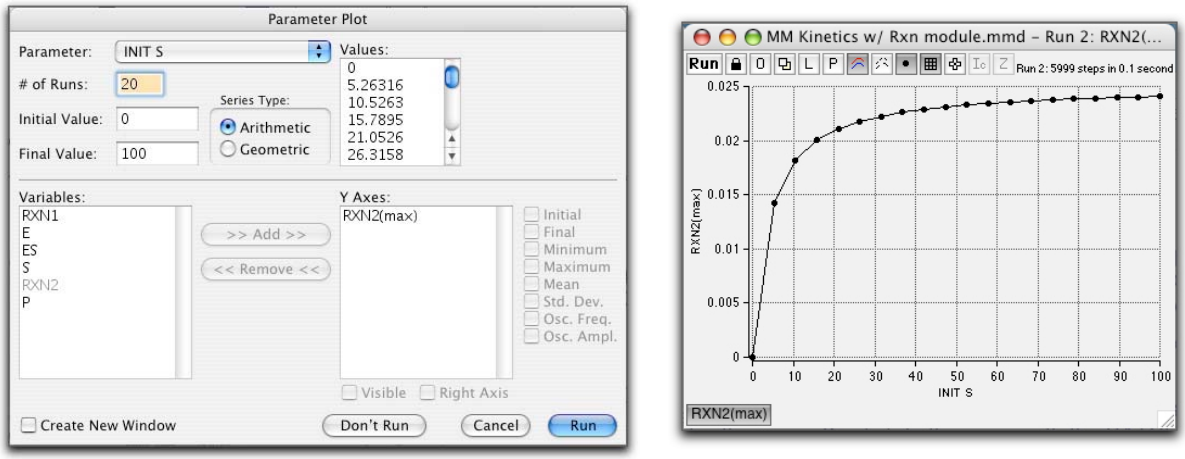
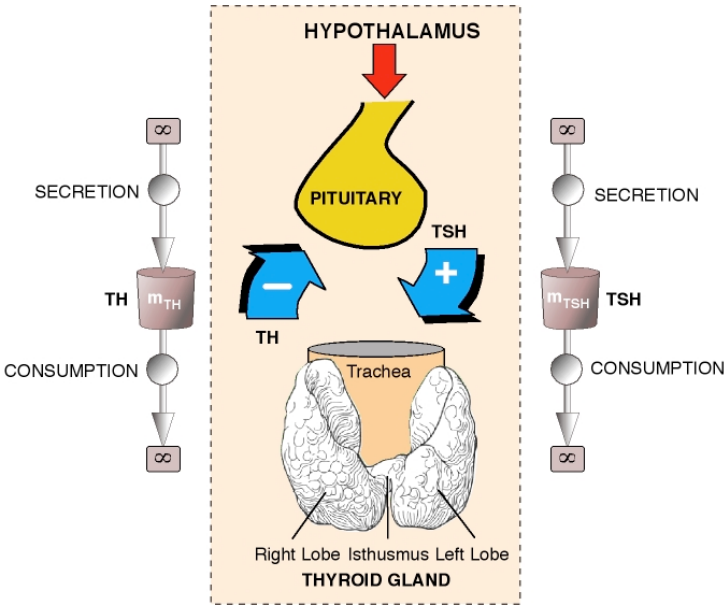


Figure 7. Plot of reaction velocity (dP/dt) vs. initial substrate concentration (INIT S) using the Parameter Plot dialog.

[1] See, for example: Stryer, L. (1995). *Biochemistry*. New York, W. H. Freeman..

Modeling systems with MM kinetics: Feedback regulation in thyroid-pituitary secretion

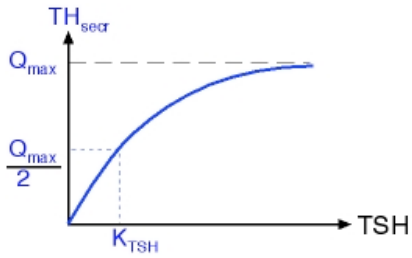
We illustrate the use of MM reactions to simulate the regulation of the thyroid gland secretion. This system also illustrates the role of thyroxine feedback on TSH secreting cells of the anterior pituitary gland. To illustrate the principles involves we will only consider two hormones, thyroxine (**TH**) and thyroid stimulating hormone (**TSH**). Thus we will neglect most of the intricacies of iodine metabolism.



Stimulation of thyroid secretion by TSH

We model the secretion of TSH by a MM function:

Equation 6 $$TH_{SECR} = \frac{Q_{\max}[TSH]}{K_{TSH} + [TSH]}$$



where Q_{max} = maximal rate of TH secretion, and K_{TSH} = concentration of TSH required for 50% maximal TH secretion.

Removal of TH

Metabolic and/or renal clearance is assumed to be proportional to [TH], where [TH] denotes the concentration of free TH *and* the concentration of TH bound to plasma protein. Since [TH] is proportional to the total mass of TH, m_{TH}, we can write

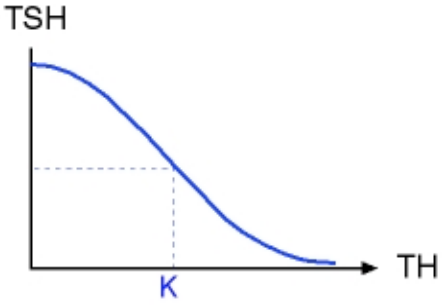
Equation 7 $$TH_{removal} = R_{TH}\, m_{TH}$$

where R_{TH} is a rate constant i.e. 1/time constant [day^{-1}].

Inhibition of TSH secretion by TH:

Equation 8 $$TSH_{SECR} = \frac{S_{\max} K_{TH}^{n}}{K_{TH}^{n} + [TH]^{n}}$$

Equation 9 $$TSH_{REMOVAL} = R_{TSH} \cdot TSH$$



where S_{max} = the rate of TSH secretion in the absence of TH, and K_{TH} = the concentration of TH required for 50% maximal TSH inhibition.

Parameters

Because the bound TH and the free TH are proportional to each other, assuming a steady state is not a serious compromise. For normal humans in a steady state:

TH_{secr} = 80 µg/day	TSH_{secr} = 110 µg/day
$[TH]_{ss}$ = 80 µg/liter	$[TSH]_{ss}$ = _____
$[TH]_{half\ life}$ = 6.5 days	$[TSH]_{half\ life}$ = 1 hour
Volume of distribution of thyroid hormone = 10 liters. (This includes the TH bound to plasma protein—which soaks up the hormone as if there were an additional volume available.)	
Volume of distribution for TSH = 3 liters. (This is the volume of plasma—a guess—TSH is a protein, not bound to other plasma proteins)	

Exercise: Estimate the remaining parameters for this simulation (see the **Appendix**)

- R_{TH}: compute the value from $[TH]_{half\ life}$
- R_{TSH}: compute the value from $[TSH]_{half\ life}$
- $[TSH]_{ss}$: Use the known value of TSH_{Secr} (see Table) and compute the steady state value, $[TSH]_{SS}$, using the steady state condition (Secr = Removal)
- S_{max} and K_{TH}: assume K_{TH} is the steady state value of [TH] and compute S_{max}
- Q_{max} and K_{TSH}: assume K_{TSH} = steady state value of [TSH] and compute Q_{max}
- **Assume:** n = 3

Exercise: Run the simulation for 50 days using initial values of [TH] = 30 µg/liter, and [TSH] = 1 µg/liter. Plot both [TH] and [TSH] and determine their "normal" steady state values. Are they consistent with the data? Note how fast the feedback system operates. Compare with different values of n. [Hint: $[TH]_{ss}$ = 80 µg/l; if you don't get this, check your parameter values carefully.

Exercise: A patient suspected of chronic hypothyroidism has blood samples taken every few days and the averaged measured levels of [TH] = 36.7 µg/L and [TSH] = 4.6 confirm the original suspicion. [TH] level is low, but because [TSH] is too high, the thyroid deficiency may be due to a reduced affinity of the TSH receptor (increased K_{TSH}), or a deficiency of the total number of TSH receptors, or of TH secretion (a decreased Q_{max}). Show that the latter assumption (defective Q_{max}) will account for the data.

Exercise: A physician wants to compensate for low levels of TH (in the patient described above) by administering daily doses of TH. What dose should he use? (Simulate the daily dose with the pulse function).

Handling long chains is easy using arrays

If a chain of reservoirs is more than 4 or 5 long, the Flowchart becomes unwieldy. If all of the transitions are identical then there is an easier way to enter a long list of identical differential equations: use Berkeley Madonna's array notation. Consider the following chain of irreversible reactions:

$$\mathbf{A1 \rightarrow A2 \rightarrow \cdots \rightarrow Ai \rightarrow \ldots \rightarrow An}$$

Instead of defining n compartments as A1, A2,…An, we can define an array A[1..n] that contains n entries. We can access any one entry by using the notation A[i]; for example, the 5th entry in the array is A[5]. Assume that at time zero we introduce 100 units of A1 into the first compartment of the system. Then the equation for the first compartment is:

d/dt (A[1]) = k[1]*A[1], INIT A[1] = 100.

Notice that we have also introduced an array of rate constants: k[1..n] for the transitions between compartments. The equations governing all of the compartments from 2 up to n-1 can be written simply as

$$d/dt\ (A[2..n]) = \underbrace{k[i-1]*A[i-1]}_{\substack{\textit{Flow}\text{ into compartment i}\\ \text{from compartment i - 1}}} - \underbrace{k[i]*A[i]}_{\substack{\text{Flow out of compartment i}\\ \text{to compartment i + 1}}}$$

The initial conditions for all 2,…n compartments can also be set in one step: INIT A[2..n] = 0. Finally, the last compartment has only an inflow:

d/dt (A[n]) = k[n-1]*A[n-1]

Thus the equation window for this system looks like that in Figure 8.

To plot the contents of all the compartments, choose the variable A[] from the Graph dialog window. To plot the contents of any compartment, define a variable Ax = A[m]. Then you can make a slider with m as a variable so scan through the graphs of each compartment. Another slider for the number of compartments, n, lets you see how the system behaves as the number of compartments is varied. Note that for large n, the graph of the last compartment, A[n], is sigmoidal, indicating the time delay between introducing substance into the first compartment and when it shows up in the last compartment.

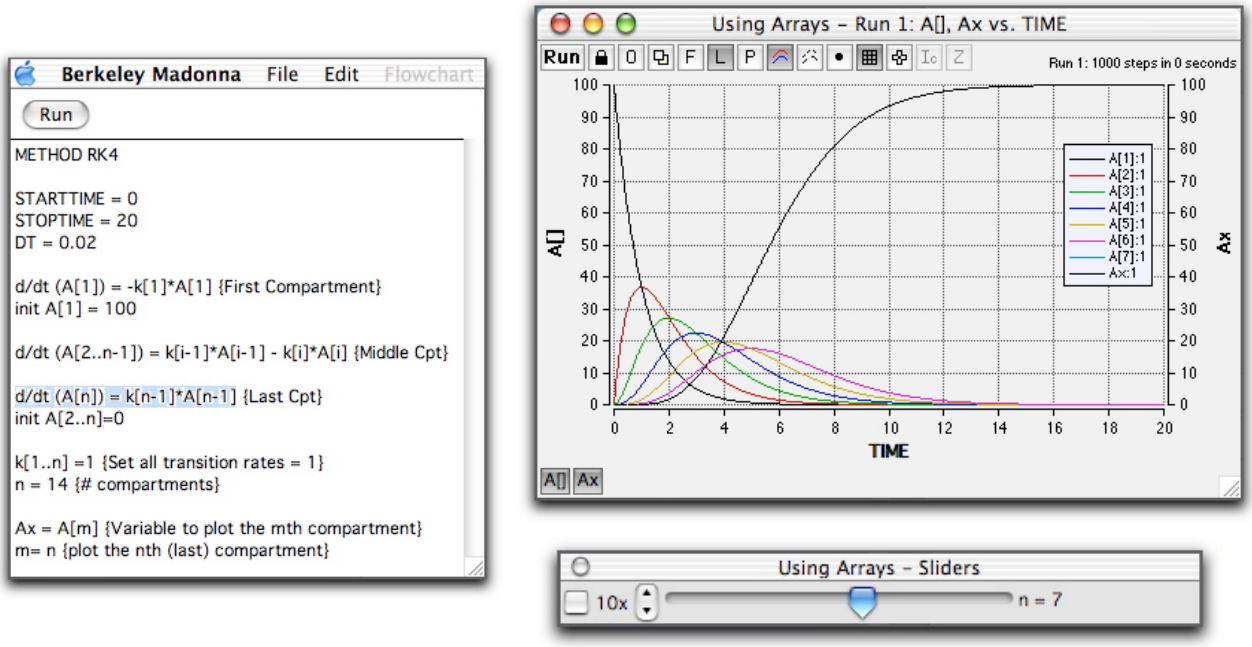


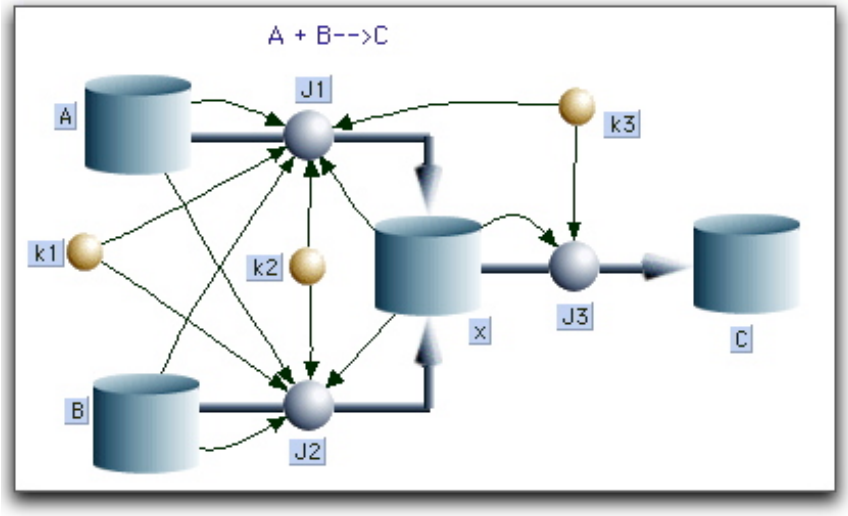
Figure 8. Equation and Graph windows for the linear array of compartments.

Exercise. Change the model in Figure 8 so that all transitions except the last are reversible: **A1 ↔ A2 ↔⋯↔ Ai ↔ ... ↔ An.** That is, connect each pair of neighboring compartments by forward and reverse flows. Make sliders to view the contents of the first and last compartments and one to vary the reverse rate constants. Compare the solutions for reversible and irreversible reactions.

Appendices

The Flowchart is not convenient for representing chemical reactions

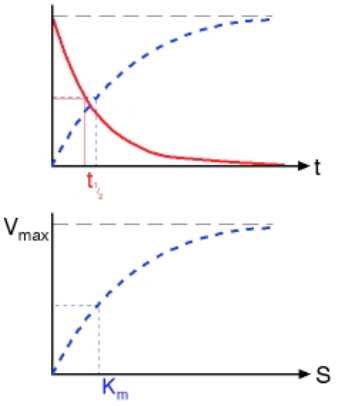
The Flowchart and equations for the reaction A + B ↔ C is shown in Figure 9. In order to make the Flowchart resemble the reaction equation we have had to introduce a separate reservoir for the 'complex' formed by A and B before they combine into the product, C: x = A·B. Moreover, the flow into reservoir C has to be halved because conservation of mole numbers (Equation 11) would require that a mole efflux from reservoirs A and B would add to make two moles entering reservoir x, and thence to reservoir C: J3 = k3*x/2 (see the equations in Figure 9). Thus the flowchart must be altered in ways that appear somewhat artificial. This reflects one of the limitations of representing chemical systems using the Flowchart interface. The Flowchart becomes even less intuitive when the stoichiometric coefficients are not unity. Therefore, for chemical reactions, the Chemical Reaction module is a much more efficient way of setting up model equations than the Flowchart.




```
{Reservoirs}
   d/dt (A) = - J1
   INIT A = 100
   d/dt (B) = - J2
   INIT B = 50
   d/dt (x) = + J2 + J1 - J3
   INIT x = 0
   d/dt (C) = + J3
   INIT C = 0
{Flows}
   J1 = k1*A*B-(k2+k3)*x
   J2 = k1*A*B-k2*x
   J3 = k3*x/2
{Functions}
   k1 = 0.002
   k2 = 0.001
   k3 = 0.1
```

Figure 9. Flowchart for the reaction A + B ↔ C. Note the halving of the flow, J3, to compensate for the lack of stoichiometry in the flows out of A and B into C.

'Guesstimating' parameters from data



1. Assume the process is 'first order' (i.e. exponential rise or decay):

   $$\text{Rate constant } k \sim 0.69/t_{1/2}$$

2. Assume a steady state ($dx/dt \equiv 0$) and use the resulting algebraic equation to eliminate one parameter.
3. If S is a 'regulated' quantity, then assume that at steady state $\overline{S} \sim K_m$ (this 'rule of thumb' is expected to be accurate only to an order of magnitude).

Stoichiometry

Consider a chemical reaction wherein ν_1 moles of reactant N_1 combines with ν_2 moles of reactant N_2 to produce ν_3 moles of product, N_3. (Or, conversely, R_3 dissociates into ν_1 moles of reactant N_1 and ν_2 moles of reactant N_2.):

Equation 10 $$\nu_1 N_1 + \nu_2 N_2 \leftrightarrow \nu_3 N_3$$

For example, in the reaction $2H_2 + O_2 \leftrightarrow 2H_2O$ the stoichiometric coefficients, ν_i, are (3, 1, 2); they give the proportion of reactants and products, e.g. 3 moles of O_2 per 2 moles of H_2O.

Although there are three reactants, there is only one reaction, and the reactants are constrained to appear and disappear in accordance with the stoichiometry, a reflection that the reaction conserves the total mass of reactants. Therefore, we can define a variable that measures how far the reaction has advanced, or the extent of reaction, which we denote by *x*, the *advancement* (Katchalsky and Curran 1965). Whatever the time course of the individual reactants during the reaction, the rate reactants disappear and products appear are constrained by the stoichiometry:

Equation 11 $$\frac{1}{\nu_i}\frac{dN_i}{dt} = \frac{1}{\nu_j}\frac{dN_j}{dt} = \frac{dx}{dt}$$

The rate at which a reaction proceeds in a well-stirred volume *V* can be described by a set of differential equations in the molar concentrations, $c_i = N_i/V$

Equation 12 $$dc_i/dt = \Re_i(\nu_1 c_1, \nu_2 c_2, \nu_3 c_3, T,p,\ldots), i = 1,2,3$$

where T and p are temperature and pressure, which are generally considered constant in biological systems, and ν_i are the stoichiometric coefficients. The rate functions, $\Re_i(\cdot)$, are empirical or theoretical functions that depend on the model. The simplest and most widely used model describes the reaction rate according to the *law of mass action*, although it is not a 'law', but simply a useful approximation. For unimolecular reactions (e.g. $A \leftrightarrow B$) the equations are linear:

Equation 13 $$\frac{dx}{dt} = k \cdot x = \frac{da}{dt} = \frac{db}{dt}$$

where *k* is a rate constant. For bimolecular reactions (e.g. $A + B \leftrightarrow C$), the equations are quadratic:

Equation 14 $$\frac{dc}{dt} = k_+ a \cdot b - k\_c$$

where lower case letters are the concentrations of the reactants.

Cooperative reactions

The MM curve is hyperbolic, characteristic of simple enzymatic reactions. One frequently encounters reaction velocity curves that are sigmoidal, indicating some sort of cooperativity. The simplest example of this is and enzyme that has two conformations, denoted (C_1, C_2), depending on whether it has bound one or two substrate molecules. Thus the enzyme has three states: empty, one substrate bound, and two substrates bound:

Equation 15

$$E + S \underset{k_{-1}}{\overset{k_1}{\rightleftarrows}} C_1 \xrightarrow{k_2} E + P$$

$$C_1 + S \underset{k_{-3}}{\overset{k_3}{\rightleftarrows}} C_2 \xrightarrow{k_4} C_1 + P$$

The equations governing these reactions are written exactly as for the MM kinetics. The conservation equation for the enzyme is $E + C_1 + C_2 = E_0$, so that we can eliminate one equation. Morever, the equation for the product can be solved independently of the rest of the kinetics once E and C_1 are known. Therefore, we need only write equations for S, C_1, and C_2.

Exercise. (a) Write the model equations for the reaction scheme in Equation 15 in terms of S, C_1, and C_2, taking into account the conservation equation for the enzyme. (b) Use the same approximation as in the MM model for the complexes: $dC_1/dt = dC_2/dt = 0$ to solve for C_1 and C_2. From this, write the equation for the reaction velocity $V = k_2C_1 + k_4C_2$ (Keener and Sneyd 1998, p. 12):

Equation 16

$$\boxed{v = \frac{(k_2K_2 + k_4S)E_0S}{K_1K_2 + K_2S + S^2}}$$, where $K_1 = (k_{-1} + k_{-2})/k_1$, $K_2 = (k_4 + k_{-3})/k_3$

Cooperativity means that binding of one substrate increases the binding rate of the second substrate. Use Berkeley Madonna to plot Equation 16 when $K_1 = 10^3$, $K_2 = 10^{-3}$, $E_0 = 1$, $k_2 = 1$, $k_4 = 2$. Use sliders to vary K_1 and K_2 till the sites are independent: $K_1 = 0.5$, $K_2 = 2$.

Plotting indeterminate functions with Berkeley Madonna

Berkeley Madonna is not a powerful plotting program so you must be careful when graphing functions that are 'indeterminate', i.e. have terms like $0\cdot\infty$ or 0/0, etc. For example, the formula for the free energy of mixing two solutions is

$$\frac{\Delta G}{RT} = x\ln(x) + (1-x)\ln(1-x) + \chi\, x(1-x)$$

where $0 \le x \le 1$ is the mole fraction, RT is the gas constant times the absolute temperature, and χ measures the attraction between molecules of the two substances (Dill and S. Bromberg 2003, chapters 15 & 25). This function is not determinate at x = 0, 1, and Berkeley Madonna does not know how to compute L'Hospital's rule to determine the values there. So you must set the plotting interval to avoid such points, as shown in Figure 10. A more powerful plotting program, like *Mathematica*™ or *Maple*™ knows how to handle such functions.

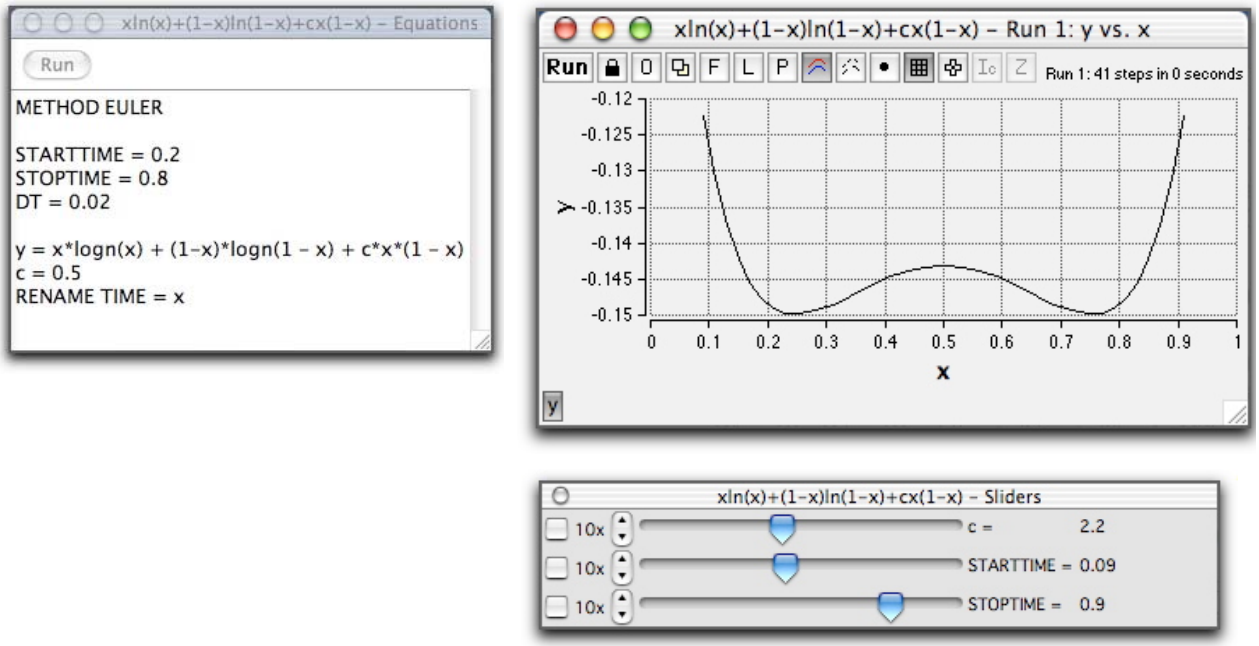


Figure 10. Graph of the function $\Delta G \propto x \cdot \ln(x) + (1 - x) \cdot \ln(1 - x) + \chi \cdot x(1 - x)$ that is indeterminate at x = 0, 1. Berkeley Madonna will not compute L'Hospital's rule to determine the values at these points, so you must set the plotting interval to avoid them.

References

Dill, K. and S. Bromberg (2003). *Molecular Driving Forces: Statistical Thermodynamics in Chemistry and Biology*. New York, Garland.

Katchalsky, A. and P. Curran (1965). *Nonequilibrium Thermodynamics in Biophysics*. Cambridge, MA, Harvard University Press.

Keener, J. P. and J. Sneyd (1998). *Mathematical physiology*. New York, Springer.

Northrop, D. (1998). On the meaning of K_m and V/K in enzyme kinetics. J. Chem. Edu. **75**(9): 1153-1157.

Northrop, D. (1999). So What Exactly is V/K Anyway? *Enzymatic Mechanisms*. P. F. a. D. B. Northrop. Washington, DC, IOS Press**:** 250-263.

Press, W. H. (1997). *Numerical Recipes in C : The Art of Scientific Computing*. Cambridge Cambridgeshire ; New York, Cambridge University Press.

Stryer, L. (1995). *Biochemistry*. New York, W. H. Freeman.

Web resources

- UBC Online calculus course:
  http://www.ugrad.math.ubc.ca/coursedoc/math100/notes/index.html - mordifeqs
- SOS Mathematics! An online tutorial for math:
  http://www.sosmath.com/index.html
  Euler's Method: http://www.sosmath.com/diffeq/first/numerical/numerical.html