Modeling with SimBiology

Quantitative Methods Boot Camp

SimBiology is a MATLAB package that automates and simplifies the process of modeling biological systems. It provides a graphical, intuitive interface for setting up models that otherwise would require a lot of expertise in differential equations and patience in debugging.

How to use this document

Section 1 (**Some background on biochemical reaction modeling**) provides a brief overview on why we model at all and what the main idea behind reaction modeling is.

Sections 2 and 3 (**A very simple model** and **Model of ligand-receptor binding**) provide a step-by-step guide on how to model in SimBiology.

After that, choose what you want to do next according to your interests: You can combine your knowledge from the first few sections to create a **Combined drug delivery, clearance and receptor binding** model. You can instead choose to have a look at how to model **Enzymatic reactions**. You can apply your modeling skills to understanding the mechanism of **hemoglobin and carbon monoxide poisoning**. Or you can learn about how (and why!) to **use existing models** from a public repository.

In this document, **boldface text** will represent names of objects you see on screen, and red text will represent input that you need to type or perform.

Table of Contents

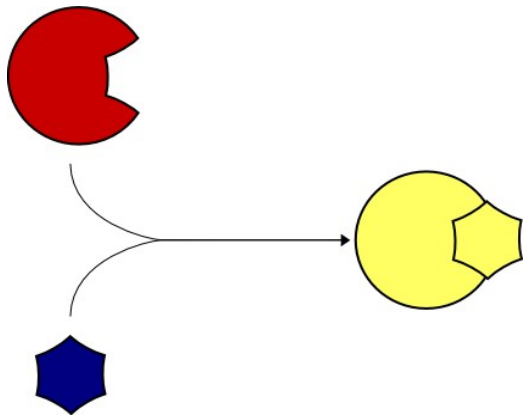
Some background on biochemical reaction modeling.....1
A very simple model: drug uptake and elimination in SimBiology.....4
Model of ligand-receptor binding.....13
Combined drug delivery, clearance, and receptor binding model.....14
Enzymatic reactions.....15
Hemoglobin and CO poisoning.....17
Importing and analyzing existing models.....18

Some background on biochemical reaction modeling

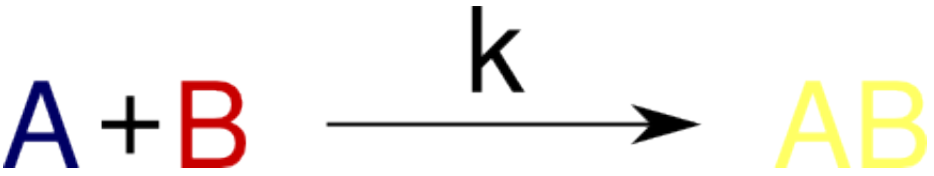
Modeling a system allows us to

- Think about the system in depth and make our assumptions transparent
- Perturb the system to predict its behavior under specific conditions
- Design and plan experiments
- Build intuition about the system
- Uncover non-intuitive emergent behaviors

Assume we want to model a biochemical reaction in which molecule A (shown in blue) binds to molecule B (shown in red) to form complex AB (shown in yellow). This is shown in the following cartoon:



Another way to represent the same system is through a chemical equation:



Now, say we have a solution containing fixed concentrations of A, B, and AB. We are interested in how this system changes as time progresses, and how quickly those changes happen. In the above example, for instance, the rate of formation of the complex AB depends on the reaction rate k, but also on the concentrations of the reactants A and B (the higher the concentration, the less time it takes for one molecule of A to find one molecule of B and form a complex).

Mathematically, we can express these rates of change as a system of differential equations. (Other modeling frameworks exist, but we will focus on this class of models for the purpose of this tutorial).

$$\frac{d[AB]}{dt} = k*[A]*[B]$$
$$\frac{d[A]}{dt} = -k*[A]*[B]$$
$$\frac{d[B]}{dt} = -k*[A]*[B]$$

(Here, square brackets denote concentrations.)

A reaction system with more participants would just correspond to a larger number of (more complex) differential equations. What simulation software such as SimBiology does is find numerical solutions to those systems of differential equations. The result is (most often) a time course showing how the concentrations of all molecules involved change over time.

We don't even have to enter the differential equations themselves, but can specify our system of reactions in a way that is very similar to the chemical equation or even the diagram shown above.

In order to do this, we have to tell SimBiology a few things about the system we want to model:

- List of components
- How components interact (reactions/rules)
- Constraints
- Initial conditions (e.g. concentrations at the outset of the simulation).
- Rates and other parameters
- Simulation variables (dependent, independent variables, time scales, ...)

A very simple model: drug uptake and elimination in SimBiology

A good starting point for learning SimBiology is to set up a simple model in it. We will use as an example the simplest possible pharmacological, constant intravenous perfusion of a drug, and its elimination from the blood stream. The model contains a single biochemical species or component, the drug itself, which can participate in either of two reactions. One is the delivery of the drug to the blood stream which will be modeled as molecules of the drug being added to the system at a constant rate. The second reaction, elimination, results in a molecule of the drug disappearing at some rate per time. A key feature of the model is that the rate of drug addition or delivery is constant (i.e. it doesn't depend on how many molecules of the drug there are) whereas the elimination rate depends on the amount of drug—the more drug is in the bloodstream, the more gets eliminated. We will come back to these conceptual points throughout the tutorial.

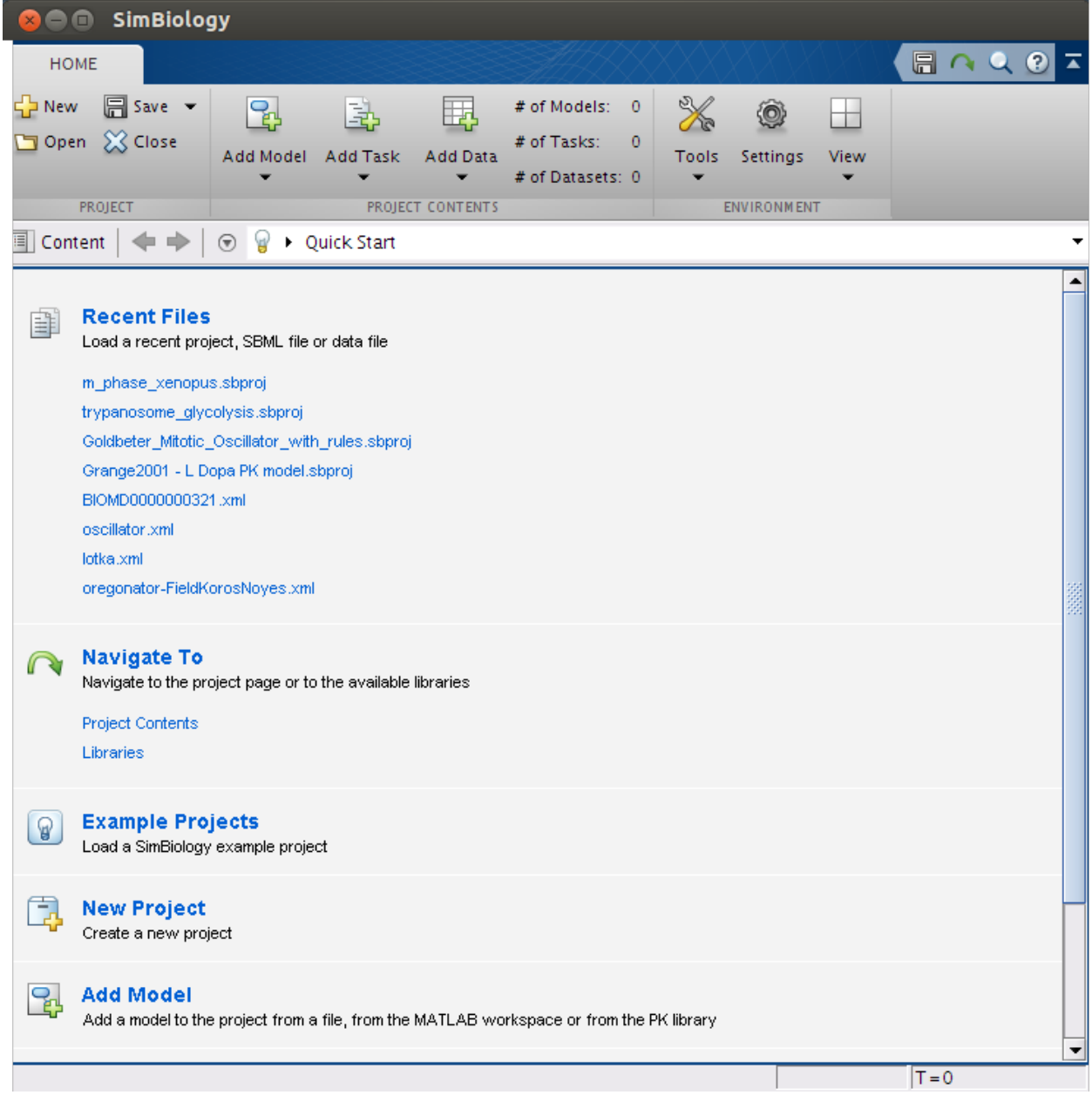
Launch SimBiology

To start SimBiology, type

```
>> simbiology
```

at the the MATLAB command prompt. After a few seconds, a window will open with the SimBiology home screen.

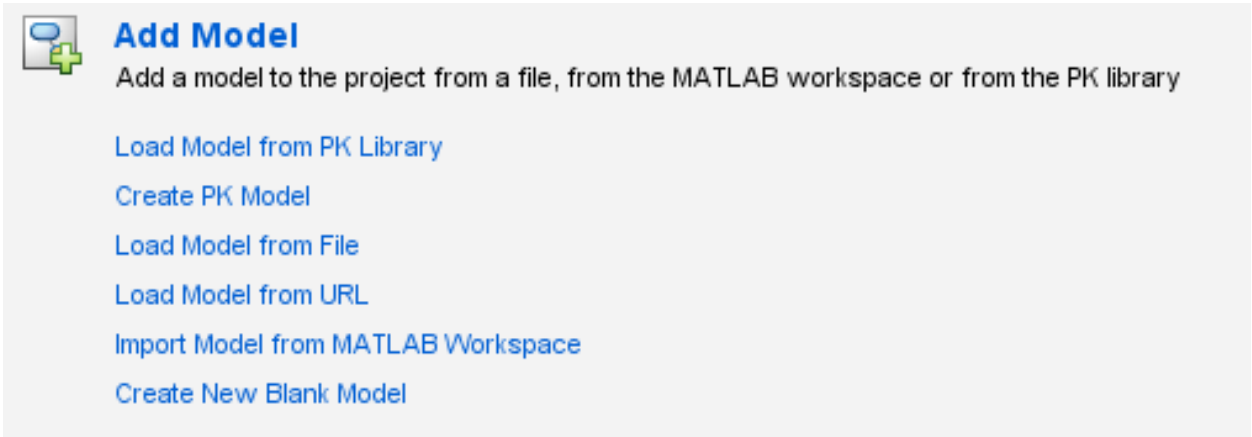
The layout will vary a bit depending on your version of MATLAB, but the most important screen you should see is the **Work Area**, which will display the details of the model you're working on. If you cannot see your Work Area, you can click on View → Work Area in the tool bar, or hit Ctrl + Shift + 0 to see it.



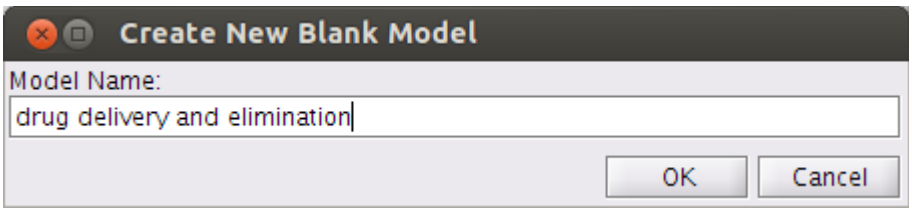
SimBiology is very powerful, but it may seem to have a lot of buttons and windows at first. Try to resist the temptation to memorize sequences of button-clicks and instead focus on why we set up models and reactions in certain ways.

Create a new model

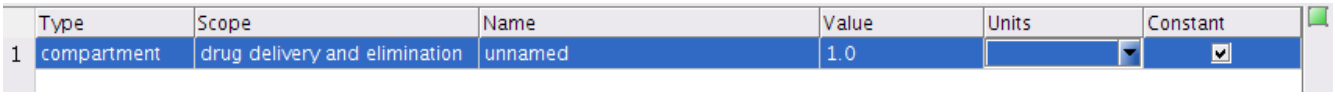
In the Work Area, and find and click Add Model → Create a blank model.



When the prompt asks you for the name of the model, type "drug delivery and elimination".



Now you should see the Work Area change to display the contents of your model. Since you haven't done anything, this should be mostly blank. Notice, however, that the bottom half of the workspace now has a table with one row whose **Type** is "**compartment**."

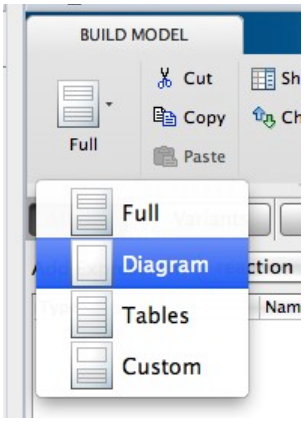


	Type	Scope	Name	Value	Units	Constant
1	compartment	drug delivery and elimination	unnamed	1.0		☑

This is the default compartment in which your reactions will take place. In general, any biochemical reaction (in the broadest sense) has to take place in a compartment, and you can define additional compartments to represent separate organs, cells, organelles, the nucleus, etc.

At the moment, your model is in "**Full view**". This view lists all the reactions and building blocks of the model (chemical species, compartments, etc.) in text form, along with their details.

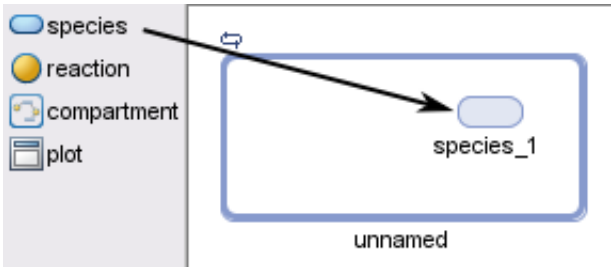
Sometimes you may find a graphical view of the model to be more intuitive. Let's switch the view by clicking the icon that says "**Full**" and choosing "**Diagram**" from the menu.



You should see your work area turn into a blank white space, with a rectangle in the middle labeled "**unnamed**" at the bottom. This represents your default compartment.

Add species and reactions

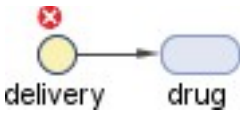
Let's add a species to the model, so we can start to build reactions. To do this, look to the **Block Library Browser** to the lower left. Click and drag "**species**" to the compartment in the Work Area.



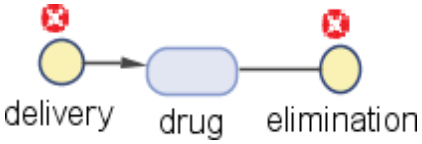
You should see a small oval that is labeled "**species_1**." Double-click on the text to edit it, and change it to say "**drug**". This oval will represent molecules of your drug in your model.

Now that you have a species, let's add some reactions that will affect the amount of the species. Drag a "**reaction**" from the Block Library Browser to the compartment. Put it to the left of your drug and name the reaction "**delivery**".

We need to tell MATLAB that this reaction acts on the drug. Hold down the Control (Windows) or Option (Mac) key and click and drag from the synthesis reaction to the drug. You should see an arrow appear.



Now add a second reaction called "**elimination**". This time, draw an arrow *from* the drug *to* the elimination reaction.



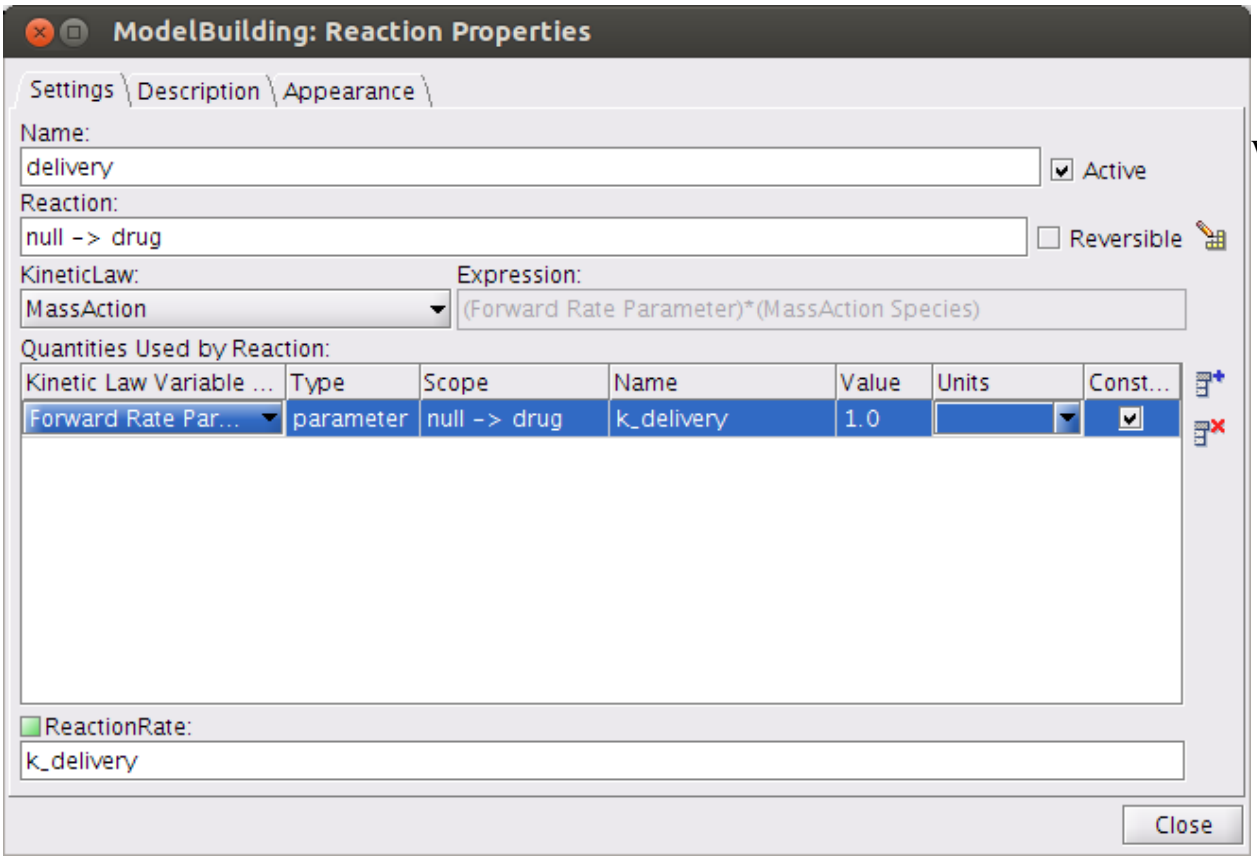
Configure the kinetics of the reactions

What does our model represent so far? We have a compartment (i.e. the blood) to which a drug is added a steady rate and from which it is eliminated. Like biochemical reactions, these processes will have characteristic *rate constants* that describe how fast they happen. We need to tell SimBiology the *kinetics* of our model, i.e. what the rates of the processes are.

To change the kinetics of the delivery reaction, double-click on the circle that corresponds to it. You should get a window titled "ModelBuilding: Reaction Properties." Click the box labeled "**KineticLaw**"

and choose "**MassAction**" from the list that drops down.

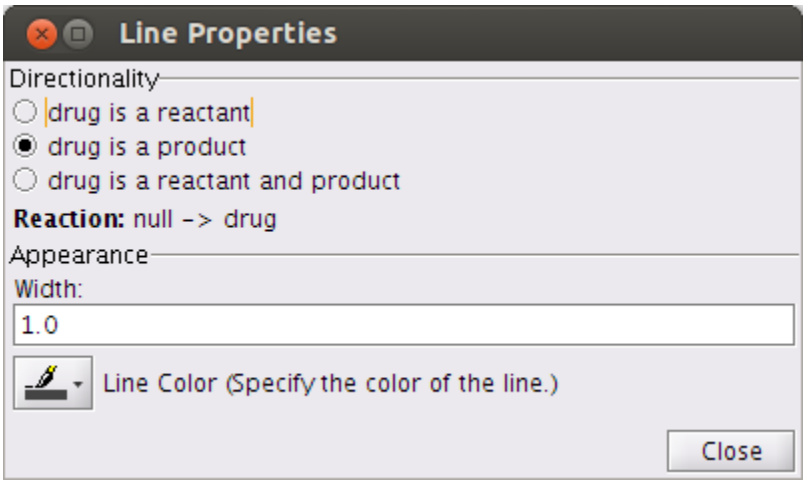
A new row of settings should show up in the box labeled "**Quantities Used by Reaction**". This row corresponds to the (forward) rate constant for this reaction. Double-click the row under the heading "**Parameter Name**" and type in "**k_delivery**". You can leave the "**Value**" at its default value of 1.



When you're done, click **Close** on the box.

Using the same procedure, change the kinetics of the elimination reaction to "**MassAction**" and name the rate parameter "k_elimination". Leave the default value of 1.

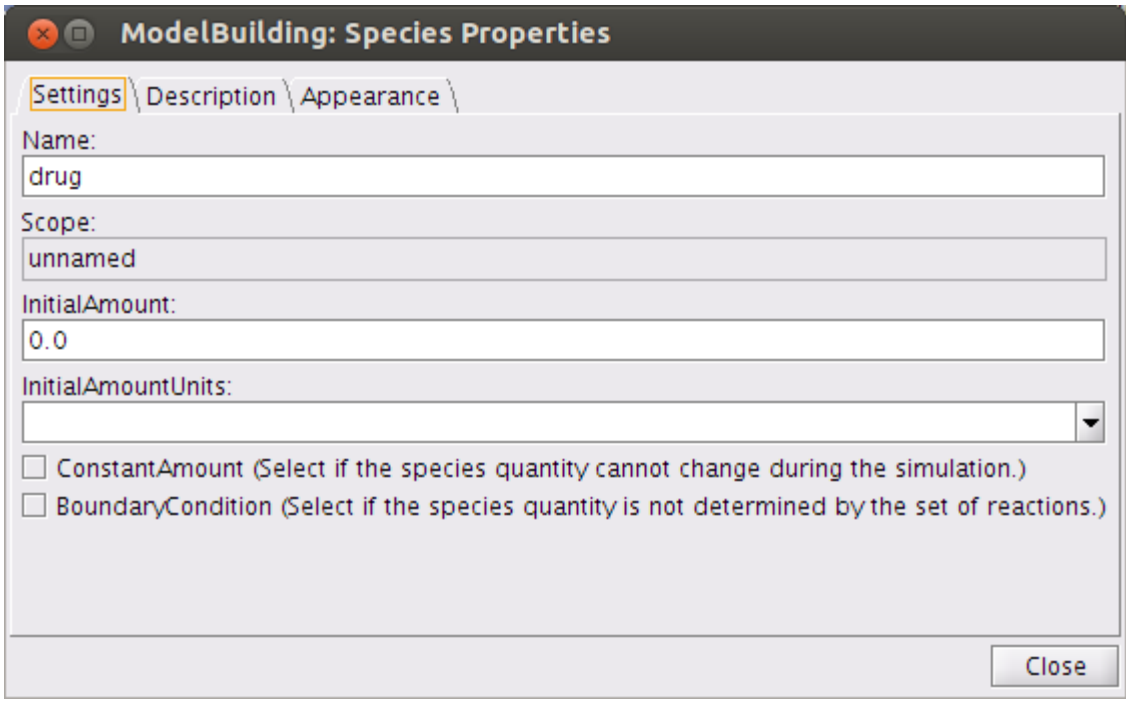
You can change the directionality of reactions by clicking the arrows on them. Try double-clicking the arrow from the delivery reaction to the drug. A box will show up and tell you that "**drug is a product**" of the reaction. This is what we want, so just click **Close**.



Similarly, clicking the arrow from the drug icon to the elimination reaction will show you that "**drug is a reactant**."

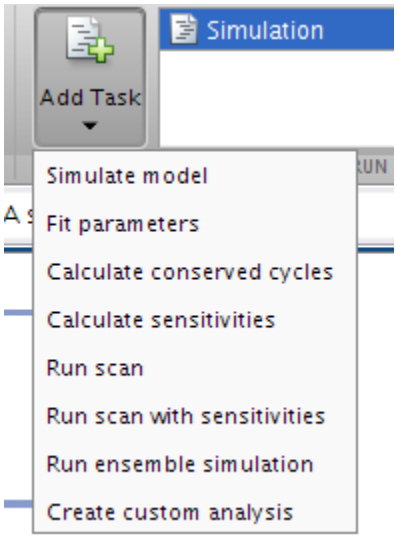
Set initial conditions

We are almost ready to run the model and see how it changes over time. One last thing we need to do is specify the *initial conditions*, or in this case, how much of the drug is present at the beginning of a run. In general, any species you create will by default have an initial concentration of 0. To see this, double-click the oval for drug and notice that "**InitialAmount**" is set to 0. This is what we want, so just click **Close**.

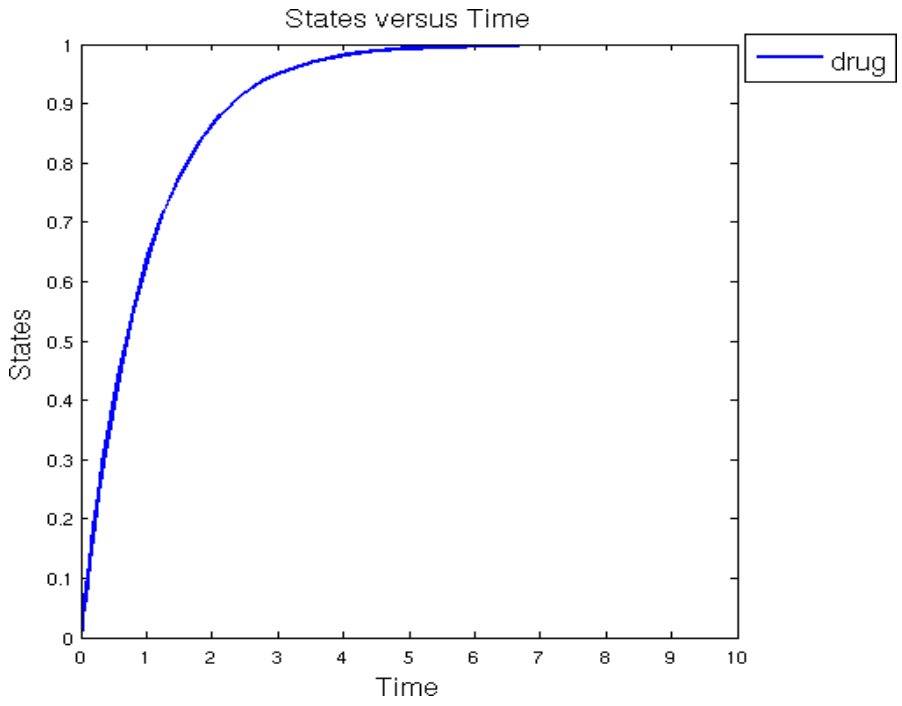


Simulate a time-course of the model

Let's see how the concentration of the drug changes over time in our model. In the top row, find the **Add Task** icon and click on **Simulate Model** in the drop-down menu.



Double-clicking on "**Simulation**" will make your **Work Area** change to display settings associated with your model. There's no need to change any of the defaults for now. Simply click the "**Run**" button in the top of the **Work Area**, and wait for the simulation to complete. A plot should pop up displaying the results of your simulation.



What shape is the graph? Is this what you expected?

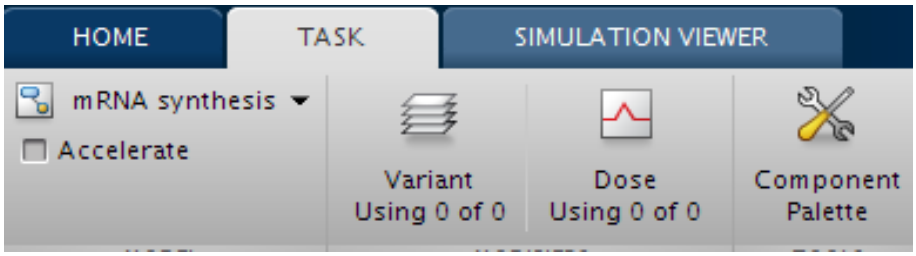
Scan a range of parameter values

In modeling we're not always interested in any particular set of parameter values. Often we want to see

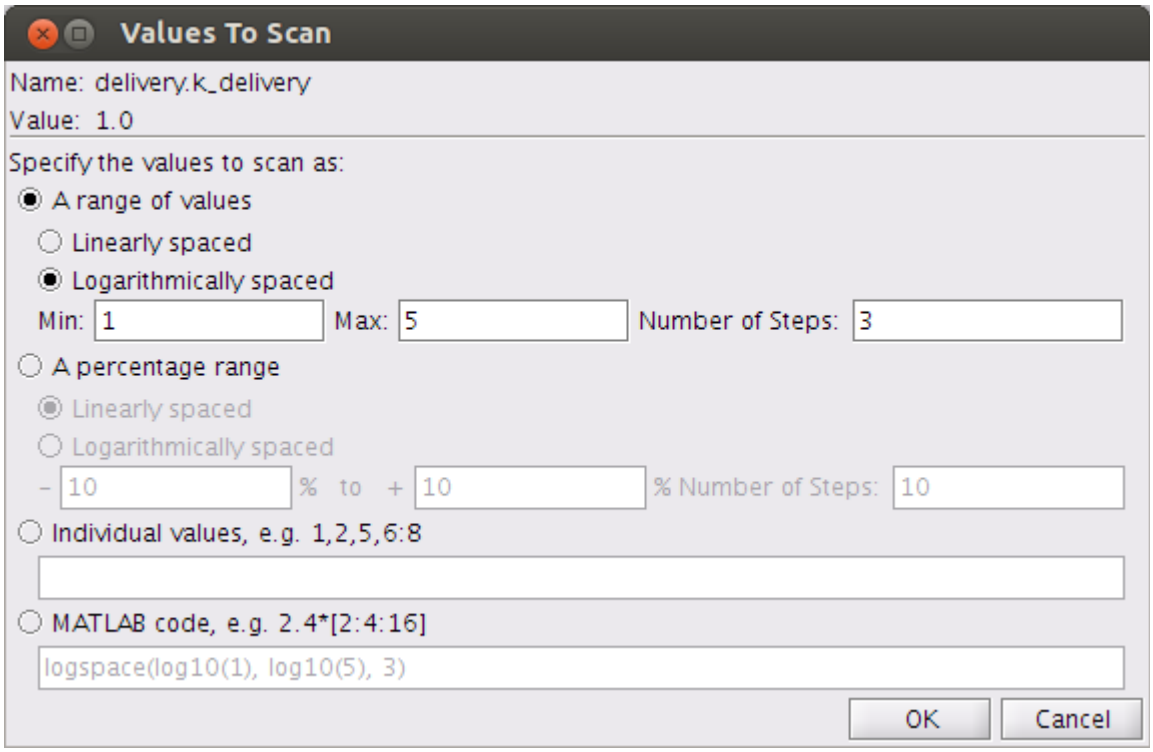
how the behavior of the model changes when we change one or more of the parameters over a wide range of values. Let's see what happens when we change the drug delivery rate in our model.

Add another task (see previous step) but this time choose "**Run Scan**" from the menu. (If you cannot find the icon, make sure you are back to the "**HOME**" tab) This will add a task called "**Scan**", and bring up some options in the **Work Area**.

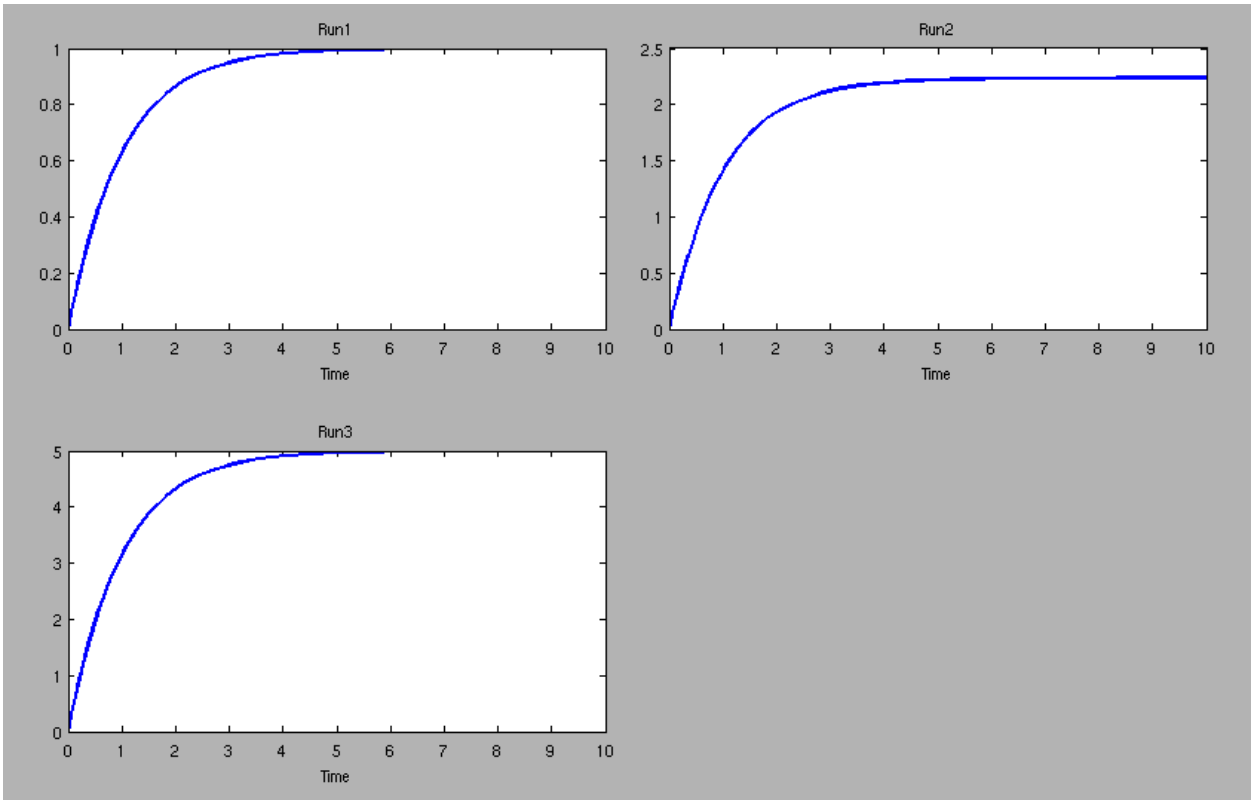
To tell SimBiology we want to scan values of the synthesis rate, go to the **Component Palette** and drag the parameter called "k_delivery" (recall that this is our delivery rate constant) to the **Work Area**. We haven't mentioned the **Component Palette** yet—if you cannot see it, you can bring it up by clicking the icon in the top menu.



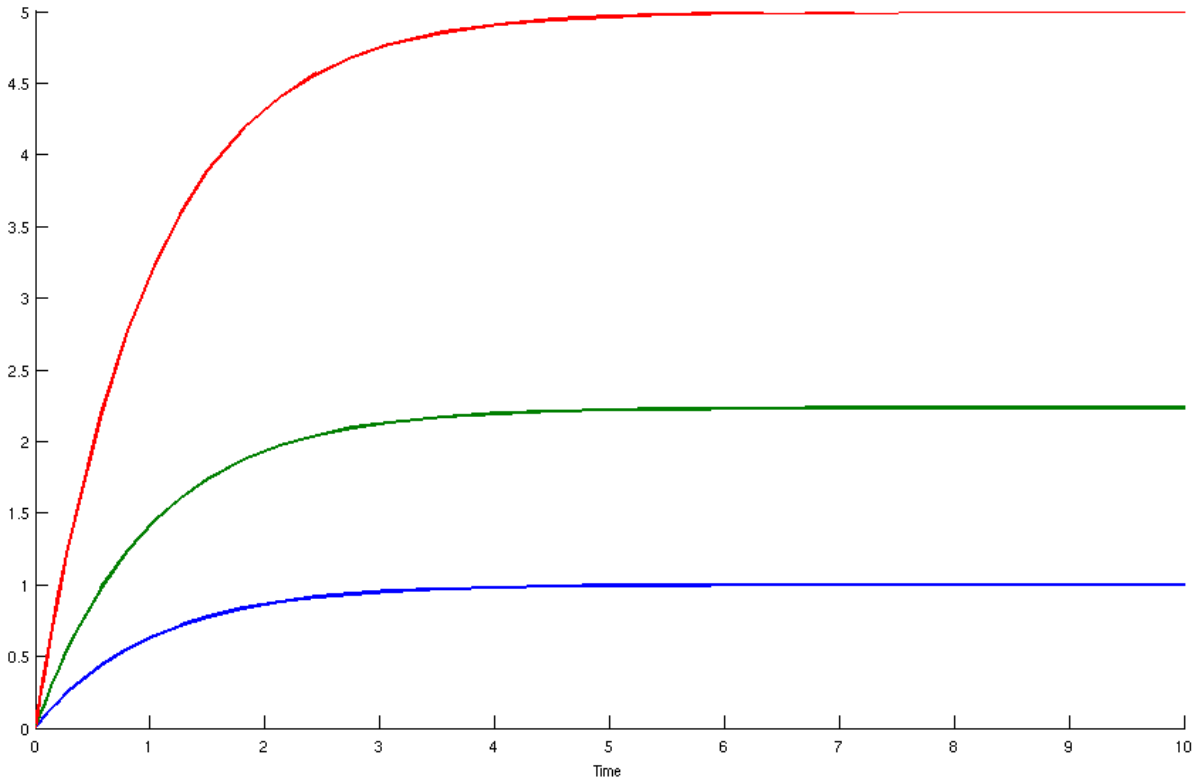
Now let's decide which values to scan. In your Work Area, click the row that corresponds to k_delivery under the header "Values to Scan." In the box that pops up, choose "**A range of values**" and make them "**Logarithmically spaced**" with a **Min** of 1, **Max** of 5, and a **Number of Steps** of 3. Click **OK** when done. This will run three simulations, each of which has a different delivery rate (1, 2, and 5).



Now click the **Run** button. After a few seconds, you will see a set of plots show up for each parameter value you selected.



Click the tab at the top of the plots for "**Time – Figure 2**." This will put the plots on the same axis.



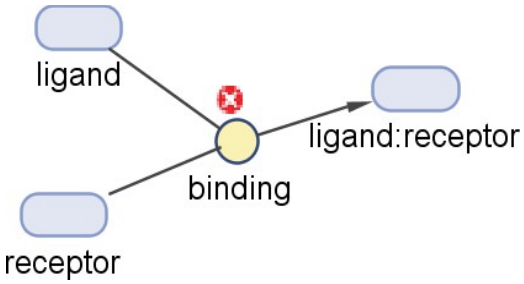
What effect does varying the delivery rate have on drug concentration over time? Try the scan now for the elmination rate. What effect does varying the elimination rate have? Is this what you expected?

Model of ligand-receptor binding

In this next example, we will look at a ligand binding to a receptor with one ligand binding site. For this purpose, we create a new model. We will again at the time course of ligand binding to the receptor, and we will also investigate how the steady-state level of the ligand-receptor complex depends upon the initial concentration of free ligand.

Model setup

Create a new blank model. We now have three molecular species (free ligand, free receptor, and the ligand-receptor complex). Combination of ligand and receptor to form the complex is the only reaction. Both ligand and receptor are reactants, and the ligand:receptor complex is the product of the reaction. Create the three species and the reaction in the **Diagram** view.



Note, however, that this reaction is reversible (the ligand:receptor complex can dissociate again into its component parts.) Bring up the **Reaction Properties window** by double-clicking on the binding reaction. Under KineticLaw, select **MassAction.** In addition, select the box on the right to make the reaction reversible. Because the reaction is reversible, you are asked to specify both a forward rate parameter and a reverse rate parameter. Name your forward rate parameter (e.g. k_on) and set its value to 0.5. Name the reverse rate parameter (k_off) and set its value to 0.1. Note that by specifying both the forward and the reverse rate parameter, you have implicitly specified the K_d of your reaction ($K_d = k_{off}/k_{on}$). After closing the window, the reaction icon will have two little arrows drawn above it to indicate that the reaction is indeed reversible.

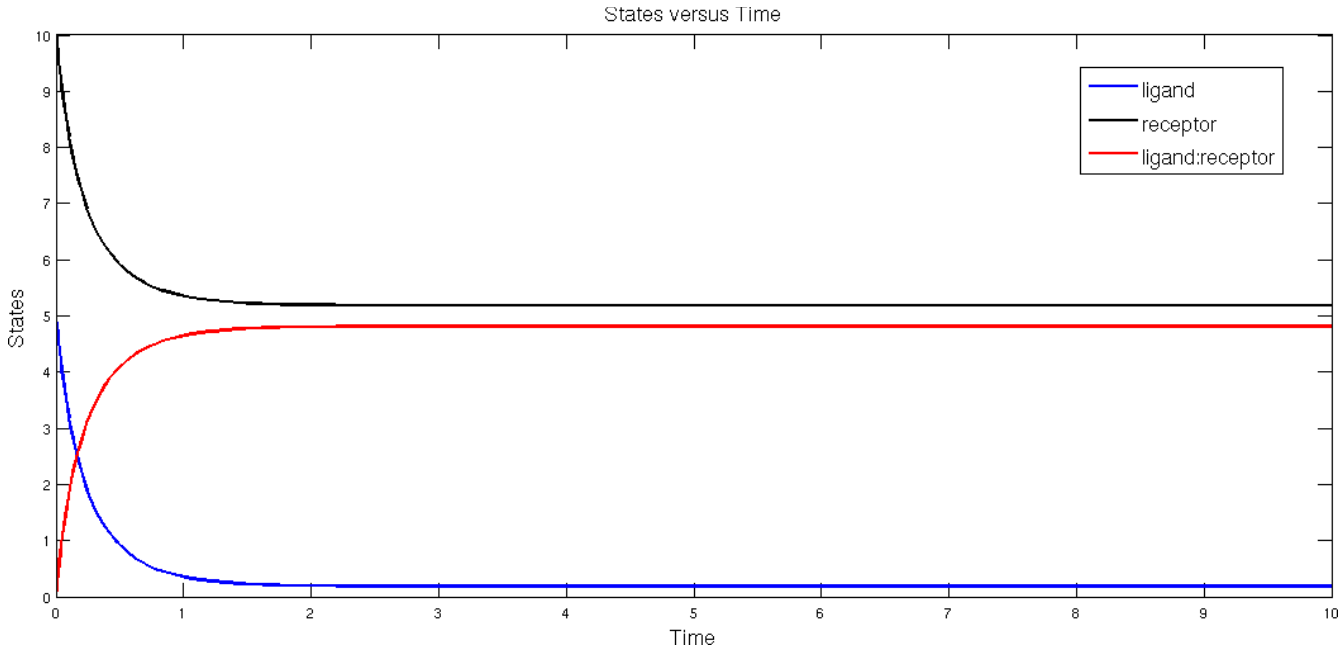
Now, we need to set our initial conditions. By double-clicking on each species, set the initial amount of receptor to 10, and of ligand to 5. Since we want to observe formation of the ligand:receptor complex, we leave its initial amount at 0.

Time course simulation

Using **Add Task**, run a simulation of your model.

The resulting plot shows amounts of ligand, receptor and the ligand:receptor complex as a function of time. As expected, the concentrations of free ligand and free receptor drop as the concentration of the ligand:receptor complex goes up. The total amount of receptor (free receptor plus ligand-bound

receptor) will be constant, as is the total amount of ligand (free plus receptor-bound).



Combined drug delivery, clearance, and receptor binding model

In the first example, we saw how to model the creation and destruction of a species of interest (in our case, a drug being delivered to the blood, and eliminated from it). The second example looked at a bimolecular reaction: the binding of a ligand to a receptor to create a complex.

Build a model of the following system: A drug is intravenously delivered to the blood (at a constant rate). In the blood, the drug binds to a receptor to form a complex (with 1:1 stochiometry). In addition, free molecules of the drug are cleared from the blood at a constant rate.

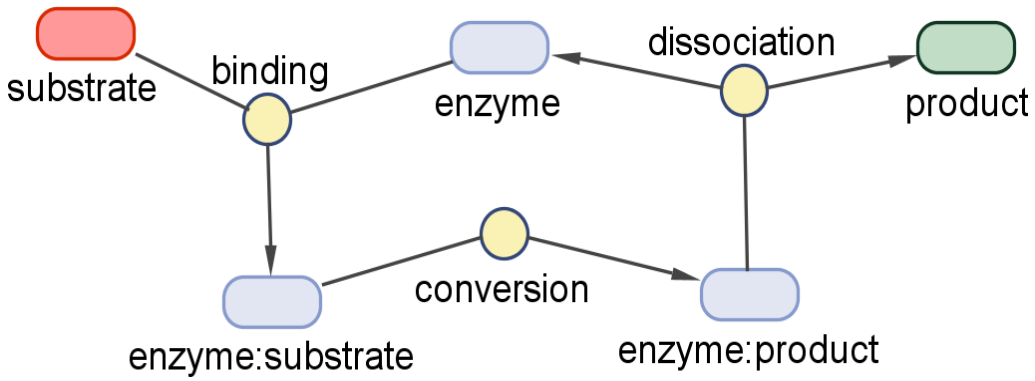
Use your model to answer the following questions:

1. How does the concentration of receptor in the blood affect the steady-state level of unbound drug in the blood?
2. How does the concentration of receptor affect the time it takes to reach steady-state?

Enzymatic reactions

So far, we have learned how to model creation and destruction of a chemical species, as well as binding of two species to form a complex. We have also seen that this allows us to model a wide range of systems. One question we have not touched upon so far is how to model enzymatic reactions. There are several ways of doing this.

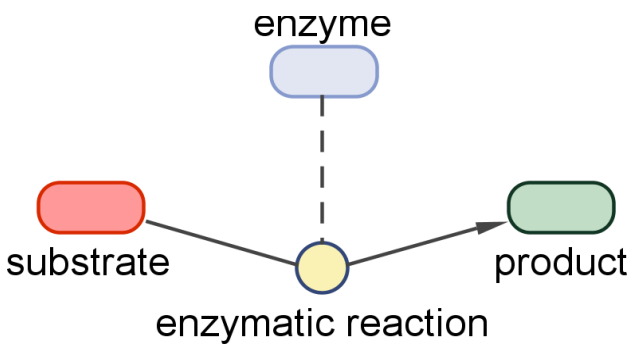
In principle, an enzymatic reaction can be broken down into its component steps and modeled as a series of mass action reactions: Binding of enzyme and substrate, conversion of the enzyme-substrate complex into an enzyme-product complex, and dissociation of that complex into enzyme and product:



However, this level of detail is not always necessary (or possible given the parameters accessible to measurement). Another way is to model the system as a one-step reaction of the form

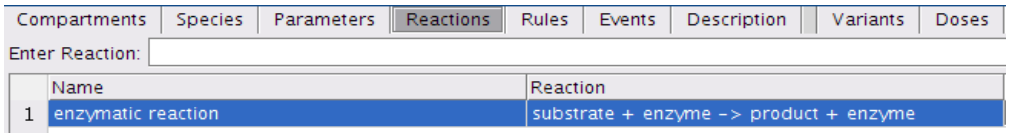
enzyme + substrate → enzyme + product

In SimBiology it is possible to declare a species as both a reactant and a product in a given reaction. For this purpose, draw an arrow from the species to the reaction (to declare it as a reactant), and then another arrow from the reaction to the species (to declare it as a product). This “double arrow” will show up in your SimBiology diagram as a dotted line:

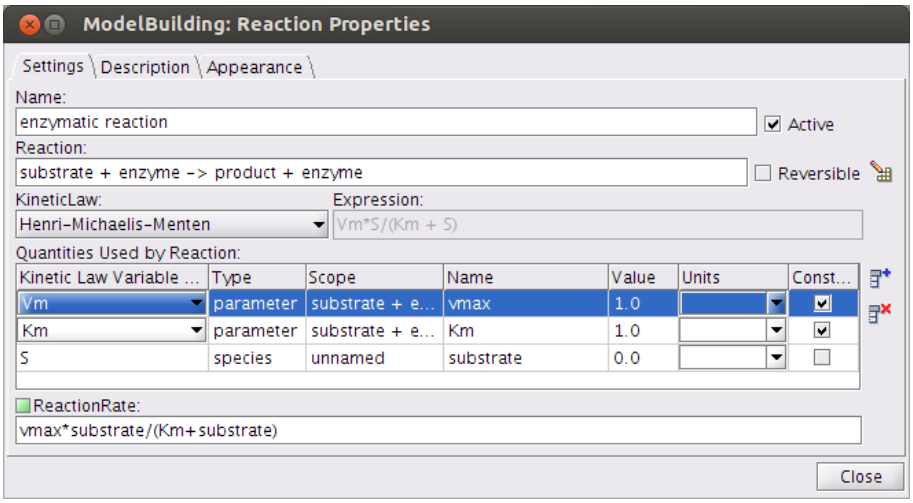


To confirm that this reaction really does what you want it to do, you can switch the view from

Diagram to **Tables**. The **Reactions table** will show you that enzyme is indeed both a substrate and a product to this reaction.



The tricky part here is what kinetic law to use for the reaction. This will depend on the nature of the reaction. If formation of the enzyme-substrate complex is the rate-limiting step, and everything else is fast compared to that, then the reaction can be approximated as having mass action kinetics. It is also possible to model the reaction using a Henri-Michaelis-Menten reaction (double-click on the reaction to bring up the **Reaction Rroperties window**, and select Henri-Michaelis-Menten from the drop down menu. You will be asked to specify v_{max} and K_m).



Note that the concentration of enzyme does not explicitly enter into the Henri-Michaelis-Menten equation. It should only be used in a model where the concentration of active enzyme is sufficiently high to be considered constant. In this case, the enzyme can be left out of the model altogether and the system represented as

substrate → product

More information about the different ways of representing enzymatic reaction can be found here: http://www.mathworks.com/help/simbio/ug/defining-reaction-rates-with-enzyme-kinetics.html

Pick your favorite enzymatic reaction and model it in SimBiology. What implementation do you choose? What behaviors can you observe? How do you know what parameters (reaction rates, initial concentrations, etc. to use?)

Hemoglobin and CO poisoning

Hemoglobin is a protein present in red blood cells that carries oxygen from the lungs to the rest of the body. It is an allosteric molecule that binds four molecules of oxygen cooperatively. This means that the more oxygen is already bound, the higher the affinity for an additional oxygen molecule.

How could you model such a system?

Carbon monoxide (CO) is usually not present in the body in large quantities. Exposure to CO, however, can have severe effects: A recent study reports that CO poisoning accounts for over 50 000 Emergency Department visits a year in the US (Hampson and Weaver, *Undersea Hyperb Med*, 2007).

The toxic effects arise from CO binding to hemoglobin: CO has a 230 times stronger affinity for hemoglobin than oxygen. This has two effects: The first is that CO outcompetes oxygen for hemoglobin binding, and therefore less oxygen can be transported to the organs. Worse still, CO binding to one site increases hemoglobin's affinity for oxygen on the other four binding site. Oxygen thus binds more easily – but that also means that it is more difficult to release.

Add CO to your hemoglobin-oxygen system. What effects do you see?

The hemoglobin model can be extended in various directions. For instance:

- The lung and other organs can be modeled as separate compartments whose micro-environments affect hemoglobin's affinity for oxygen (for instance, low pH favors oxygen release, which facilitates transport of oxygen to the organs most in need of it).

- A number of diseases are characterized by hemoglobin malfunctions, including increased affintiy for oxygen or decreased stability (=increased decay rate) of hemoglobin.

- 2,3-Bisphosphoglyceric acid (2,3-BPG) binds to hemoglobin and promotes oxygen release. High altitudes promote the production of 2,3-BPG in order to allow more oxygen to be released into tissues.

- Fetal hemoglobin has a higher affinity for oxygen than maternal hemoglobin, so that oxygen from the mother's blood can be taken up by the fetus.

Importing and analyzing existing models

In recent years, more and more modelers have started making their models publicly available. This has a range of benefits:

- Other scientists can check the accuracy of a model during and after peer review

- A model does not “die” when its creator moves on to a new project. Other scientists can use an existing model and alter or extend it, instead of having to “reinvent the wheel”

- A model that is used by other people has more impact in the scientific community

- Models of different parts of a system can be combined into comprehensive models of the whole

- Model sharing has led to the development of new standards that ensure quality and compatibility with a range of modeling software packages

The most common standard for model sharing is the Systems Biology Markup Language (SBML). SBML files can be read and interpreted by a large number of modeling and simulation platforms including MATLAB/Simbiology. Simbiology also allows users to export models to SBML. An SBML file can be deposited in a public repository of models, such as BioModels database (http://www.ebi.ac.uk/biomodels-main/) A number of journals now ask for deposition of a model in a public repository when an article describing the model is submitted (the same way that protein structures or DNA sequences are deposited into public databases).

Go to http://www.ebi.ac.uk/biomodels-main/ If there is a biochemical system you are particularly interested in, see if you can find models about it.

If not, check out the Model of the Month feature: Every month, a member of the BioModels team selects a model and writes a short article describing the model and its impact.

Choose a model, download the SBML and import it in SimBiology. Examine the structure of the model. Do you understand what the paths are and how they interact? Can you simulate a time course? Can you think of a way to alter the model (for instance, to account for specific mutations, different environmental conditions, drug treatment etc.?)

Created 2/5/14 by MIS for BCMP 309 (drug delivery and clearance example based on SimBiology Intro file by JW)
Modified and extended for Neuro 306qc, MIS 8/21/14