BIO/AMS 332: Computational Modeling of Physiological Systems — Project 1

Instructions

This project is to be completed using MATLAB. MATLAB is a powerful environment for mathematical modeling — allowing quite sophisticated models to be implemented without requiring a lot of knowledge about programming. In this project, you will learn the basics of how to use MATLAB, using as examples some of the models that we have discussed in class. MATLAB is available on most SINC site computers, and for installation on your own computer via SoftWeb. Please read through the "Introduction to Matlab" appendix in the class notes for a basic introduction to MATLAB commands.

The solutions to the problems should be presented as a computer-formatted document with answers given in full sentences. All figures should be integrated into the document, and should include a brief descriptive caption. All plots should include a title and axes labels (with units), and a legend where appropriate. An appendix including all Matlab scripts should be included. A final copy of the report should be submitted electronically as a single PDF document through BlackBoard.

All MATLAB work presented in your answers must be consistent with your submitted M-files. Separate M-files should be used for each part of the project; for the various questions within a part, you may either create a single M-file that solves all the sub-problems or split your work among multiple M-files. Regardless, a single "zip" file containing all Matlab scripts (M-file) should be submitted at the same time as the project PDF document. The zip file should contain the executable M-files (as unformatted text); this is NOT a substitute for including the code as an appendix to your report.

You are encouraged to work in groups, both in discussing how to implement the problems in MATLAB and in the interpretation of your results, and you may freely use online resources. However, **every student must write their own code, generate their own figures, and write their own project report. Directly copying the work of a fellow student or any material found online is forbidden; this applies to both the written report, the Matlab code, and any generated figures.** All projects must include a statement that explains all resources that were used in the completion of all aspects of the project and lists all collaborators involved. A few examples are given below:

- *I worked on implementing the Matlab code with Tom Hanks and Brad Pitt; Tom really knew how to solve the problems, and Brad and I largely followed his guidance. However, I wrote the program myself, based on Tom's help. Tom, Brad and I also discussed the answers to the questions together, using Wikipedia for guidance on question 2, but we all made equal contributions. The written work is all my own. Signed: Sam Jackson*

- *I did the project completely on my own; I did not discuss any of the problems with anyone in the class, and did not share any of my work with others. However, I did make extensive use of both Wikipedia and MatlabCentral. Signed: Katy Perry*

Part A: Understanding the Hill equation.

We have discussed the use of the Hill equation a general functional form that can be used to describe *cooperativity* in the rate of an enzyme or in the activity of a transcription factor:

$$\nu = \frac{V_{\max}[S]^h}{K_{1/2}^h + [S]^h}$$

When $h = 1$, the Hill equation becomes identical to the Michaelis–Menten equation.

1. (a) *On a single graph*, plot the Hill equation with $h = 1$, $h = 2$ and $h = 10$. Use a range of substrate concentrations, $[S]$, from 0 to 100 mM and assume, $V_{\max} = 5.0$ mM/s and $K_{1/2} = 20.0$ mM.

   (b) Using $h = 2$, $V_{\max} = 5.0$ mM/s, and the same range of substrate concentrations as in 1(a), create a single plot showing the Hill equation with $K_{1/2} = 10.0$ mM, 20.0 mM and 40.0 mM.

   (c) Using $h = 2$, $K_{1/2} = 20$ mM, and the same range of substrate concentrations as in 1(a), create a single plot showing the Hill equation with $V_{\max} = 2.0$ mM/s, 5.0 mM/s and 10.0 mM/s.

   (d) Describe the similarities and differences you see in the curves, and provide an explanation of how the variables in the Hill equation affect the form of curve.

2. (a) Using your results from question 1, *predict* what you expect a plot of the Hill equation to look like if $h = 4$, $V_{max} = 20$ mM/s and $K_{1/2} = 50$ mM. While your prediction should include a sketch of the expected curve, you must also explain the thought process underlying your prediction.

   (b) Test your prediction by plotting the Hill equation with these parameters, using MATLAB, and discuss any differences that you observe.

3. (a) The dynamics of an enzyme-catalyzed reaction, $E + S \rightarrow E + P$, can be modeled with the Hill equation. Using the the Forward Euler algorithm, with a time step of 0.01 s, integrate the following differential equation to obtain $[S]$ as a function of time for 10 s from an initial concentration of 100 mM:

   $$\frac{d[S]}{dt} = -\nu = -\frac{V_{\max}[S]^h}{K_{1/2}^h + [S]^h}$$

   Use the same parameters as in question 2 ($h = 4$, $V_{max} = 20$ mM/s and $K_{1/2} = 50$ mM), and plot your results as $[S]$ vs t.

   (b) Compare your graphs from 2(b) and 3(a), and discuss why they are different even though we used the Hill equation with the same parameters in both cases.

Note: In the above I have provided units, with concentrations in mM and time in seconds; while it is important to get in the habit of always keeping track of units, for this excerise the units do not play a significant role.

Part B: An autoregulatory gene.

We also discussed in class a model for a simple autoregulatory gene; that is, a gene that encodes a protein whose function (or part of whose function) is to activate the transcription of itself. Recall that the system of differential equations for this is:

$$\frac{d[X_{rna}]}{dt} = \frac{\mu[X_{prot}]^2}{K_{1/2}^2 + [X_{prot}]^2} - \chi_{rna}[X_{rna}]$$

$$\frac{d[X_{prot}]}{dt} = \omega[X_{rna}] - \chi_{prot}[X_{prot}]$$

For all the questions below, use the parameters $\mu = \omega = \chi_{prot} = \chi_{rna} = 1\ s^{-1}$ and $K_{1/2} = 0.33\ mM$.

1. (a) Implement the Forward Euler approach to integrating this system to determine the concentration of both RNA and protein over time. Use a total time of 20 s and a time step of 0.01 s, and an initial concentration of 0.5 mM for *both* RNA and protein. Plot the concentrations as a function of time.

   (b) Rerun the simulation from (a), but using an initial protein concentration of 0.2 mM and an initial RNA concentration of 0 mM.

   (c) Rerun the simulation from (a), but using an initial protein concentration of 0.5 mM and an initial RNA concentration of 0 mM.

   (d) Rerun the simulation from (a), but using an initial protein concentration of 0 mM and an initial RNA concentration of 0.2 mM.

   (e) Rerun the simulation from (a), but using an initial protein concentration of 0 mM and an initial RNA concentration of 0.5 mM.

   (f) Discuss your observations.

2. (a) Modify your code from question 1 so that the the simulation will be repeated 64 times, each with a different set of starting concentrations; both protein and RNA initial concentrations should be varied from 0.0 to 1.4 mM in intervals of 0.2 mM, and *all combinations* of these should be considered.

   (b) Plot the results from all simulations in 2(a) on a *single plot* of $[X_{rna}]$ vs $[X_{prot}]$. *Note that we are* **not** *plotting concentrations versus time.*

   (c) Explain what each line on the graph in 2(b) represents, with a particular focus on the significance of the beginning and the end of each line. Is there something interesting you notice?

3. Write a paragraph discussing your results in the context of the biology of this system. Your answer should include a discussion of any stable (or equilibrium) states of the system and what those states would represent in a biological context. Also include any questions you have that your results leave unanswered.

Helpful hints.

Plotting multiple graphs on one plot: The `hold on` command in MATLAB makes all subsequent plots appear on the same graph, rather than overwriting the previous one. The `hold off` command switches the behavior back to the default mode; if you forget to use `hold off` after creating a plot with multiple graphs, you may be surprised by future plots also being added to it!

Forward Euler: Recall that Forward Euler is an iterative algorithm for integrating differential equations that begins with a given starting condition and then generates values for successive time steps using an update rule of:

$$x(t + \Delta t) = x(t) + \left(\frac{dx(t)}{dt}\right) \Delta t$$

and that the derivative must be recalculated at each step.

Nested `for` loops: To repeat a simulation over many initial conditions, as in Part B question 3, you can use multiple `for` loops. Create one loop for each condition that is being varied, and place all your simulation code inside this loop structure.

Your code may look somethink like this:

- `for (rna0 = 0:0.2:1.4)`
- `  for (prot0 = 0:0.2:1.4)`
- `     Fill in this section with your simulation code.`
- `  end`
- `end`