BioE 135/235 Homework 2

Due Date: ~~Thursday, 3/4/2021, 5:00pm PST~~ **Saturday, 3/6/2021, 11:59pm PST** to Gradescope
Late work will be deducted 20% for each day it is late.

1. Negative Feedback Regulation

Consider the following mechanism of negative feedback of protein production, where X is a repressor protein:

$$\begin{aligned} Promoter &\rightarrow Promoter + X \\ X &\rightarrow \emptyset \\ Promoter + X &\leftrightarrow Promoter : X \end{aligned}$$

It can be shown that the rate of protein production can be represented in the following equation:

$$\begin{aligned} \frac{d}{dt}[X] &= \frac{k_{tr}k_{tx}}{k_{dm}}[Promoter]_0 \frac{K_d}{K_d + [X]} - k_{dp}[X] \\ &= k_p \frac{K_d}{K_d + [X]} - k_{dp}[X] \end{aligned}$$

For an abstracted production rate $k_p := \frac{k_{tr}k_{tx}}{k_{dm}}[Promoter]_0$.

(a) Suppose the repressor X can bind to a small molecule S_X, otherwise known as an inducer, to form a complex XS_X. Only free X can bind to the promoter and repress production. When bound to S_X, the repressor is inactive, but can still be degraded at the same rate k_{dp}. Assuming (1) equilibrium binding dynamics between S_X and X (with dissociation constant $K_X := \frac{[X][S_X]}{[XS_X]}$), and (2) a total amount of $X = X_T$ (on the timescale of binding), give an expression for $\frac{d}{dt}X_T$ as a function of X_T, S_X, and others constants. Note that the dependent variable is now X_T and **not** X itself.

To begin, we can look at the new reactions that this small-molecule inhibitor introduces:

$$\begin{aligned} X + S_X &\leftrightarrow XS_X \\ XS_X &\rightarrow \emptyset \end{aligned}$$

By assumption, we have that the first reaction is at equilibrium, with $K_X = \frac{S_X \cdot X}{XS_X}$. The second reaction has flux $\nu_1 = k_{dp}XS_X$, so that we have $\frac{d}{dt}XS_X = -\nu_1 = -k_{dp}XS_X$.

Now, we define $X_T = X + XS_X$, so that

$$\frac{d}{dt}X_T = \frac{d}{dt}X + \frac{d}{dt}XS_X$$

We already know from the prompt what $\frac{d}{dt}X$ is; we also already determined what $\frac{d}{dt}XS_X = -k_{dp}XS_X$. Thus in total, we have

$$\frac{d}{dt}X_T = k_p \frac{K_d}{K_d + X} - k_{dp}X - k_{dp}XS_X$$

The prompt specified that we want to remove all instances of X from our equation, and replace them with $X_T = X + XS_X$. First, notice the two subtraction terms can be factored, such that we have

$$\begin{aligned}\frac{d}{dt}X_T &= k_p\frac{K_d}{K_d + X} - k_{dp}(X + XS_X)\\ &= k_p\frac{K_d}{K_d + X} - k_{dp}X_T\end{aligned}$$

To remove the final X from this, we must turn to our equilibrium assumption:

$$\begin{aligned}K_X &= \frac{S_X \cdot X}{XS_X}\\ \Rightarrow XS_X &= \frac{S_X \cdot X}{K_X}\\ X_T &= X + XS_X\\ \Rightarrow X_T &= X + X\frac{S_X}{K_X}\\ \Rightarrow X &= \frac{1}{1 + \frac{S_X}{K_X}}X_T\end{aligned}$$

Which ultimately leaves us with the final expression:

$$\frac{d}{dt}X_T = k_p\frac{K_d}{K_d + \dfrac{X_T}{1 + \frac{S_X}{K_X}}} - k_{dp}X_T$$

(b) Oftentimes, repressors are multimeric proteins, wherein each monomer binds to a small molecule. Now, let's suppose that X can be bound to n molecules of S_X to form the complex nS_XX. Assuming (1) a simple treatment where we consider only unbound X or the fully-formed complex nS_XX, (2) equilibrium binding dynamics in which all n molecules of S_X bind to X at the same time (with dissociation constant K_X – note that this dissociation constant may not be the same as it was before), and (3) a total amount of $X = X_T$ (on the timescale of binding), give an expression for $\frac{d}{dt}X_T$ as a function of X_T, S_X, n, and others constants.

This analysis, as it turns out, is almost identical to the previous case; the only difference now is in the first reversible reaction mentioned above, and the fact that XS_X is now technically XnS_X:

$$X + nS_X \leftrightarrow XnS_X$$

So that now, our definitions of K_X and X_T are slightly different, but not fundamentally affected:

$$\begin{aligned}K_X &= \frac{S_X^n \cdot X}{XnS_X}\\ \Rightarrow XnS_X &= \frac{S_X^n \cdot X}{K_X}\\ X_T &= X + XnS_X\\ \Rightarrow X_T &= X\left(1 + \frac{S_X^n}{K_X}\right)\\ \Rightarrow X &= \frac{1}{1 + \frac{S_X^n}{K_X}}X_T\end{aligned}$$

Thus we have:

$$\begin{aligned}\frac{d}{dt}X_T &= \frac{d}{dt}X + \frac{d}{dt}XS_X \\ &= k_p\frac{K_d}{K_d+X} - k_{dp}X_T \\ &= k_p\frac{K_d}{K_d+\frac{X_T}{1+\frac{S_X^n}{K_X}}} - k_{dp}X_T\end{aligned}$$

(c) Think back to Question 3 from Homework 1; there, you applied QSSA on mRNA dynamics to determine an expression for the unregulated case of protein production. Now, using the `odeint` function from before, plot out each of the following cases:

(i) $X(t)$ when protein production occurs without regulation;

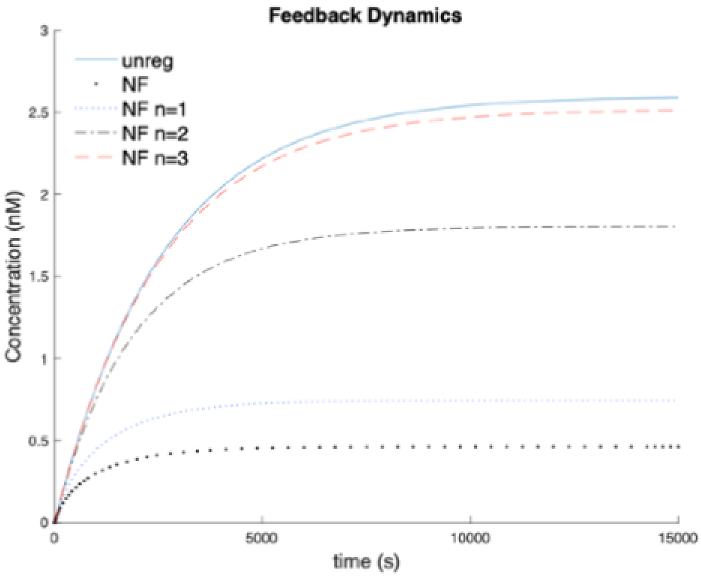
(ii) $X(t)$ when protein production occurs with simple negative feedback; and

(iii) $X_T(t)$ when protein production occurs with a small molecule inducer, with 3 different Hill coefficients $n = 1, 2, 3$.

Assume that $k_{dp} = 3.85 \cdot 10^{-4}$ s^{-1}, $k_p = 0.001$ nM/s, $K_D = 10^{-10}$ M, $K_X = 10$ nM, and $S_X = 2K_X$. Compare and contrast the steady-state values between each of these. Are they different? Does the Hill Coefficient n affect the steady-state concentration?

Note: `odeint` is useful and recommended, but if you are more comfortable with another numerical solver (say, something in MATLAB), feel free; the emphasis is mostly on the plots.

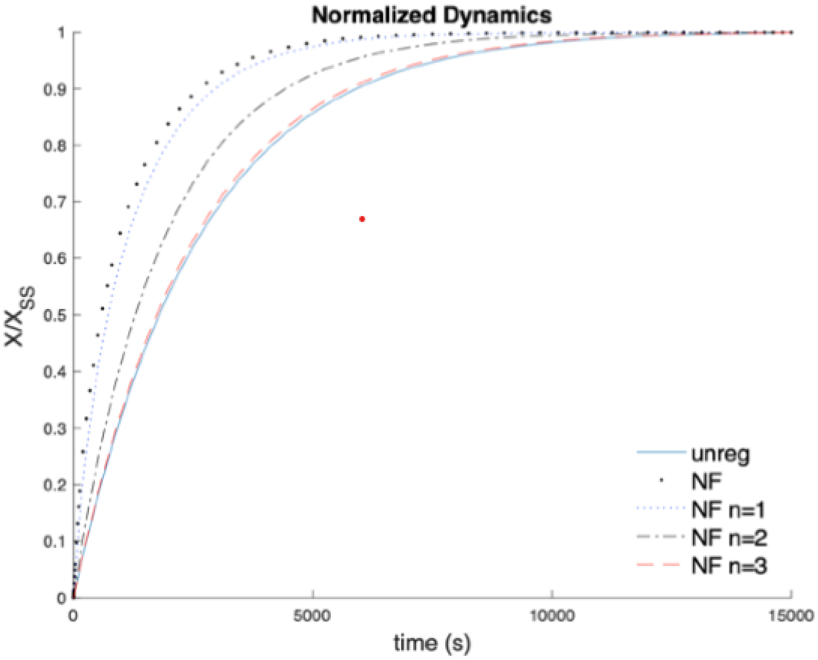
Numerically solving and plotting the system yields:



The steady-state values should be different between the negative feedback and unregulated feedback mechanisms. Qualitatively, this makes sense — the negative feedback of the repressor shuts off transcription from the promoter after it's translated. Quantitatively, we can solve for the steady states and see that the steady state for the unregulated mechanism is $X_{ss,unreg} = \frac{k_p}{k_{dp}}$, while the steady state for the negative feedback case is $X_{ss,nfb} = \frac{K_D}{2} + \sqrt{\left(\frac{K_D}{2}\right)^2 + X_{ss,unreg}K_D}$. The negative feedback from transcriptional repression reduces the production of the system. Increasing cooperativity increases the steady state ($n = 0$ is standard negative feedback, where $n = 3$ gets pretty close to the unregulated steady state). Intuitively, we cna see that if more of the repressor is bound/inactivated by small molecule binding, then there is less free repressor available to bind to the promoter and repress transcription, making the system look more like the unregulated case.

(d) Lastly, Normalize each of the above plots by their steady-state value ($X_{norm}(t) = \frac{X(t)}{X_{ss}}$, where X_{ss} can be found from the plots in part (c)), and plot them against each other to get a sense of how long each version takes to reach its own steady state. Which mechanism reaches steady state fastest? Slowest? Does this make sense?

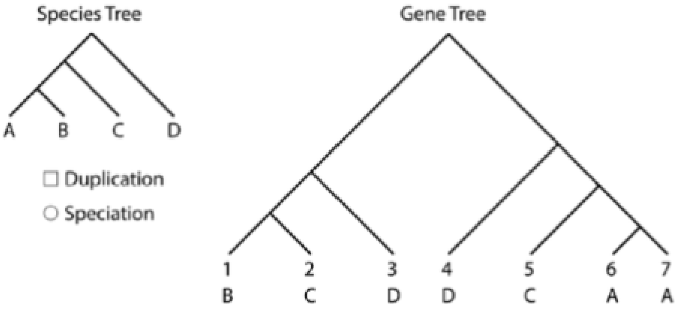
Numerically solving and plotting the system yields:



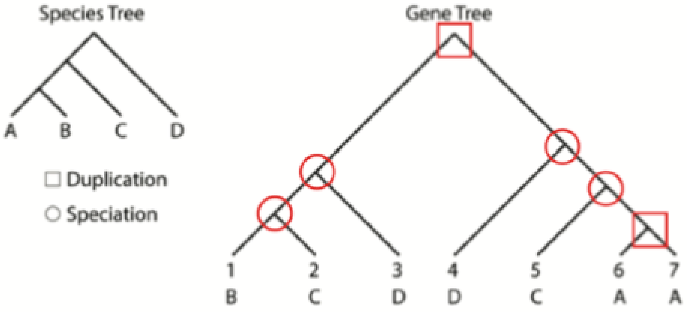
The pure negative feedback without regulation reaches steady state the fastest, while the unregulated cases reaches steady state the slowest. Increasing the Hill Coefficient n increases the time required to get to steady state. In this case, negative feedback reaches steady state the fastest, because its steady state is significantly lower than the steady states of the unregulated and small-molecule inhibitor feedback cases.

2. Phylogeny Practice

(a) On the trees below, A, B, C, and D indicate four different species, and the numbers indicate seven different genes. Assume that no horizontal gene transfer has taken place.



Using the information above in the species tree, reconcile the gene tree, indicating which nodes on the gene tree represent duplication and speciation events. Draw a circle for a speciation event, and a square for a duplication event. What is the evolutionary relationship (ortholog vs. paralog) between the following pairs of genes: 1 and 2; 1 and 3; 6 and 7; & 5 and 7?



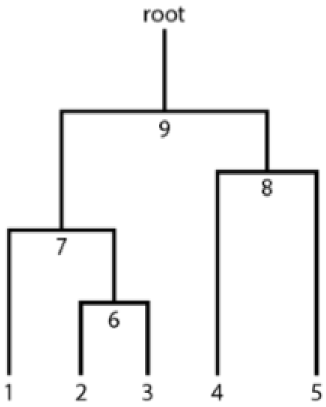
1 and 2: orthologs

1 and 3: orthologs

6 and 7: paralogs

5 and 7: orthologs

(b) An example of a rooted binary tree is shown below:



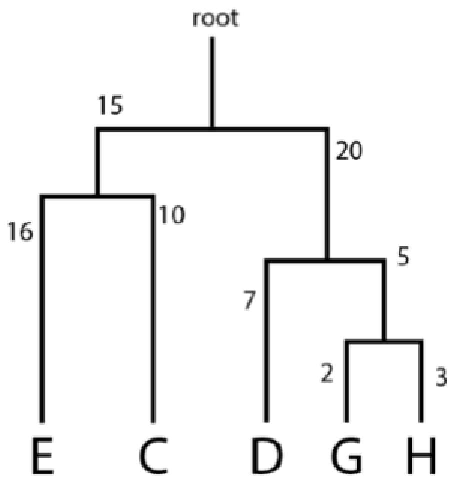
Some terminology: points 1-5 are called leaves, or leaf nodes, and points 6-9 are called inferred nodes, or internal nodes. Node 9 is also referred to as the root node. The branches between nodes are called edges. The whole tree is referred to as a binary tree because, for each event that splits the tree, the tree only splits off in two directions (as opposed to three or more).

Consider a binary tree with n leaf nodes. How many internal nodes does the tree have? How many total nodes are there? Discounting the edge above the root node, how many edges does this n-leaf tree have?

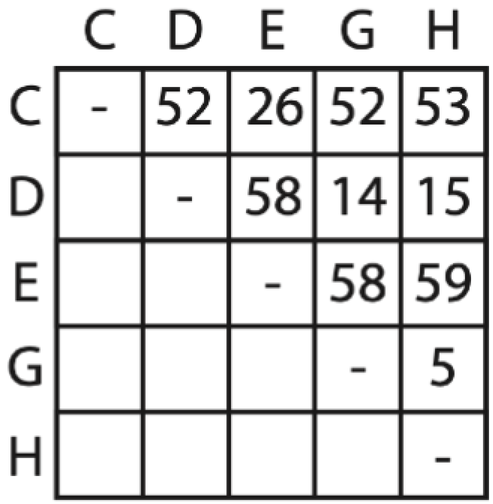
For a rooted binary tree with n leaf nodes, there are an additional $n-1$ internal nodes. Adding up the leaf (n) and internal ($n-1$) nodes there are $2n-1$ total nodes. Given $2n-1$ total nodes and discounting the edge above the root node there are $2n-2$ edges. This can be checked empirically by referencing the provided example. There are a total of 5 leaf nodes (i.e. n=5), 4 internal nodes (i.e. $n-1$=4) and 9 total nodes (i.e. $2n-1$). Finally there are 8 edges (i.e. $2n-2$).

(c) Given a tree, the distances between two nodes can be represented in a distance matrix D, where D_{ij} represents the distance between each pair (i,j) in a given dataset. Fill out the following distance matrix using the edge lengths between nodes provided in the tree below:

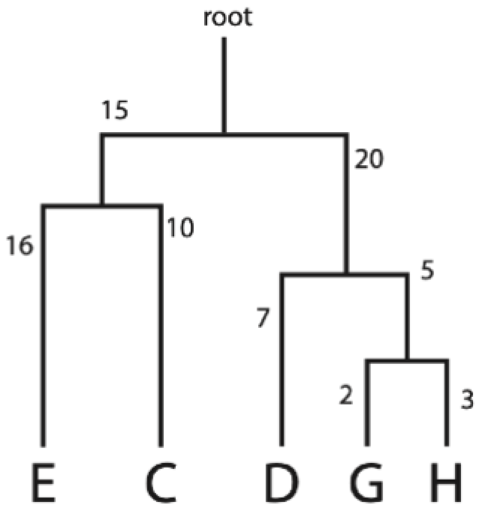
	C	D	E	G	H
C	-				
D		-			
E			-		
G				-	
H					-



Note that $D_{ij} = D_{ji}$; the distance from node i to node j is the same as the distance from node j to node i. Thus this matrix is symmetric, and we will only fill out the top half:



	C	D	E	G	H
C	-	52	26	52	53
D		-	58	14	15
E			-	58	59
G				-	5
H					-

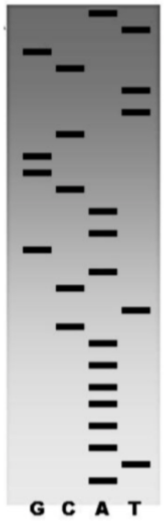


(d) Propose a method for evaluating evolutionary distances between genes. What will you measure?

This question was intended to be open-ended, and as such, there is no strictly correct answer here. One possible proposal could have been to measure codons instead of DNA sequences, to get a clearer image of whether a particular mutation actually added any nontrivial evolutionary distance.

3. Sequencing

(a) Consider the following end product from a Sanger sequencing read:



Assume that the starting wells are at the **bottom** of the gel, and that DNA migrated upwards. Also assume that the labels on the bottom indicate which base was modified within each column (so the

"G" column had a modified Guanine inserted into the pot, the "A" column had a modified Adenine, etc.). What was the original DNA sequence that led to this Sanger read?

In Sanger sequencing, the reactions that terminate early generate smaller fragments. The smaller fragments have less mass, and will therefore travel further along the gel. Thus we can read off the complementary bases that were highlighted by Sanger sequencing as:

3’ ATGCTTCGGCAAG-ACTCAAAAAATA 5’

If we wish to arrive at the original sequence, then, we must take the complement of this sequence:

5’ TACGAAGCCGTTC-TGAGTTTTTTAT 3’

(b) Suppose we want to sequence *E. coli*. There are a wide variety of sequencing technologies available; what is the standard read length for Sanger, Illumina, PacBio, and Oxford Nanopore sequencing? Plot the probability of observing any particular piece of the *E. coli* genome as a function of the number of reads N for each sequencing method.

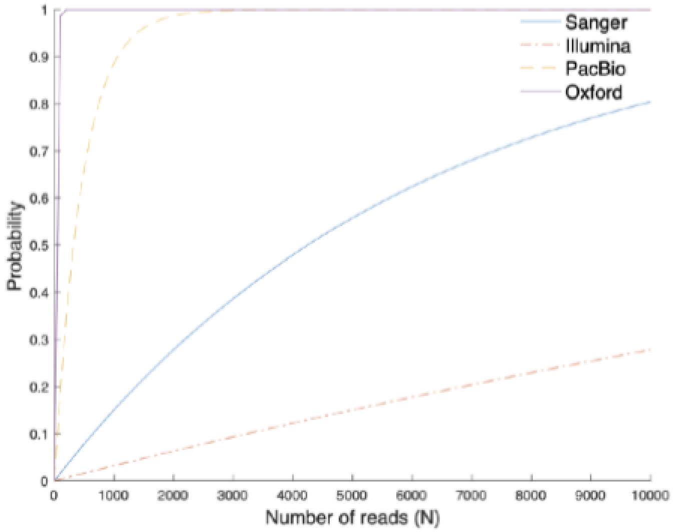
(a) Sanger: 500-1000bp (we’ll use 750bp for our calculations)

(b) Illumina: 50-600bp (we’ll use 150bp for our calculations)

(c) PacBio: 10,000bp

(d) Oxford Nanopore: 1,000,000-2,000,000bp (we’ll use 1,500,000bp for our calculations)

Recall from lecture 5 that, if G is genome size, L is read length, and N is the number of reads we have, then the probability P of observing any particular piece of the genome is $P \approx 1 - e^{-\frac{NL}{G}}$. Plotting this out for $G = 4.6$Mb and L according to each of the above read lengths, we arrive at the following graph:



4. Modeling with COPASI

For this question, you will be using the COPASI software kit to simulate a series of biochemical reactions.

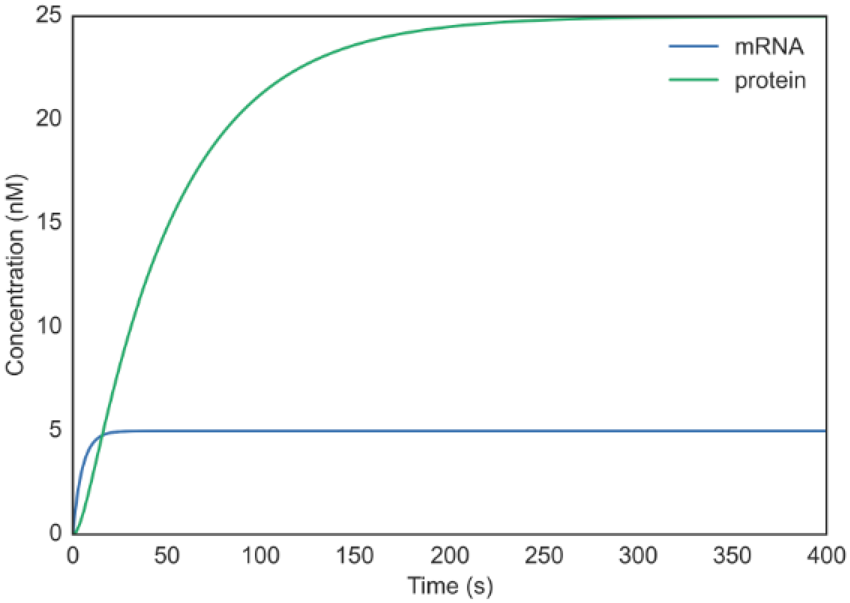
You can download COPASI here, and you can find the COPASI manual here. There are also some tutorial videos with COPASI here. Consider the following system:

$$
\begin{aligned}
&\rightarrow mRNA \\
mRNA &\rightarrow mRNA + Protein \\
mRNA &\rightarrow \emptyset \\
Protein &\rightarrow \emptyset
\end{aligned}
$$

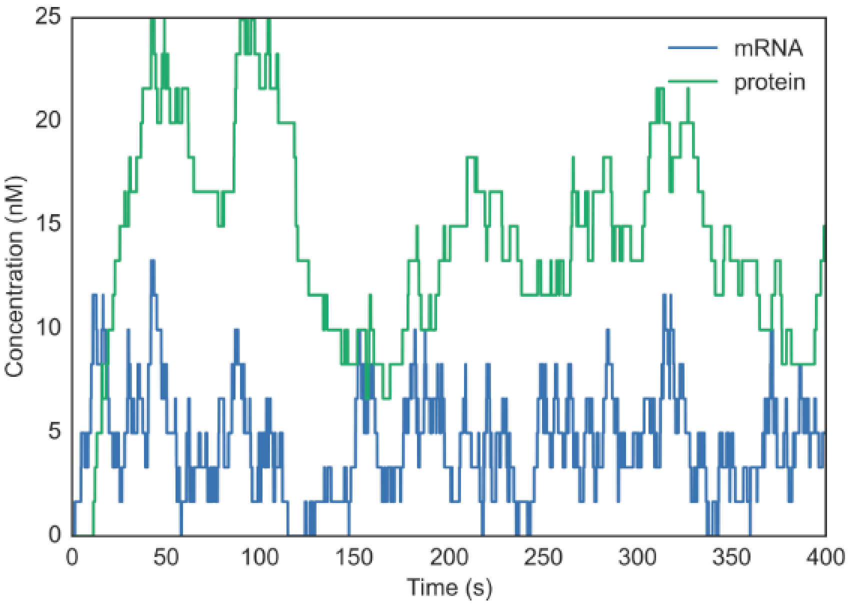
Where $k_{tx} = 1$ nmol/(L·s), $k_{tl} = 0.1$ s^{-1}, $k_{dm} = 0.2$ s^{-1}, and $k_{dp} = 0.02$ s^{-1}.

(a) Model this system in COPASI, and plot the concentration profiles of mRNA and protein using both the Deterministic (LSODA) method and the Stochastic (Direct) method.

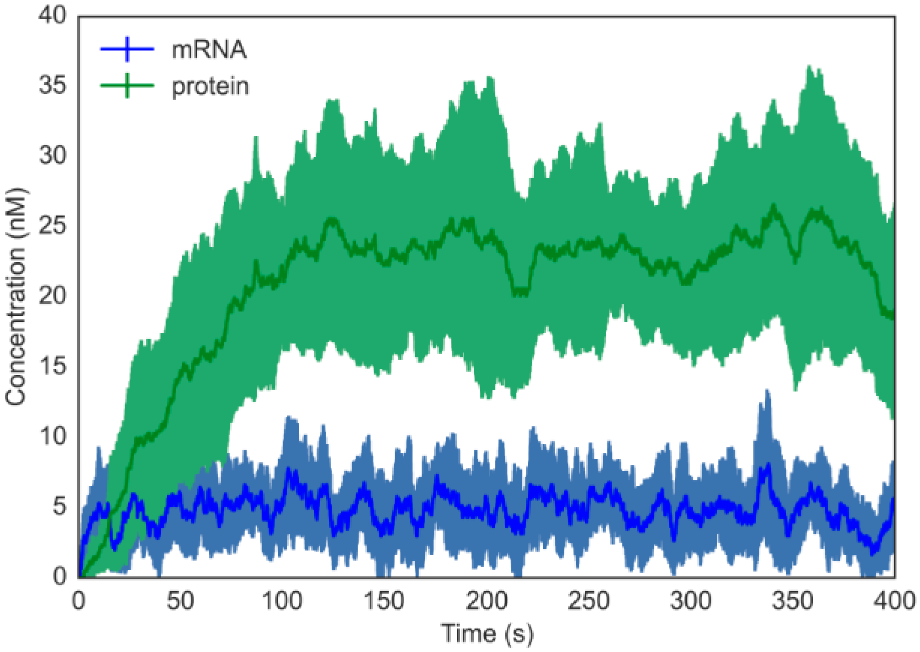
The deterministic simulation should look something like this:



Likewise, the stochastic simulation:



(b) Run the stochastic simulation 10 times and plot the average concentration profile and standard deviation for $mRNA(t)$ and $P(t)$.

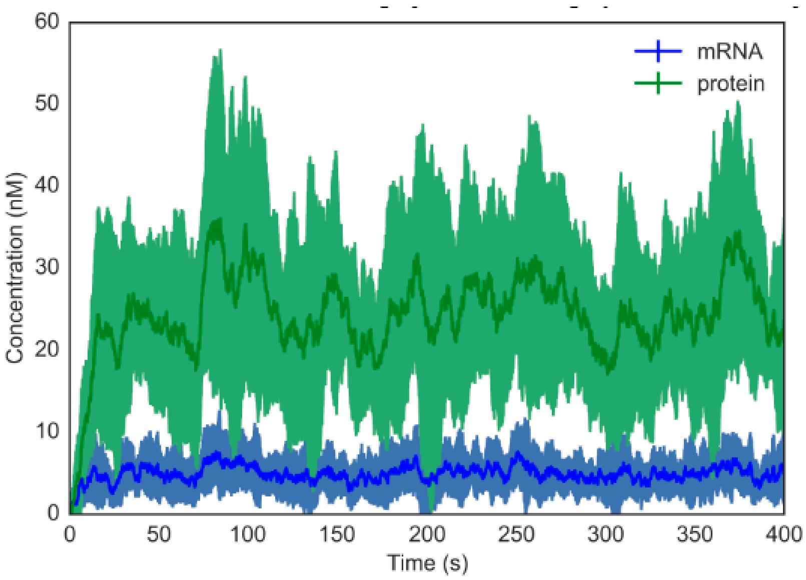


Note: dark lines represent the average concentration, while lighter bars span +/- 1 standard deviations.

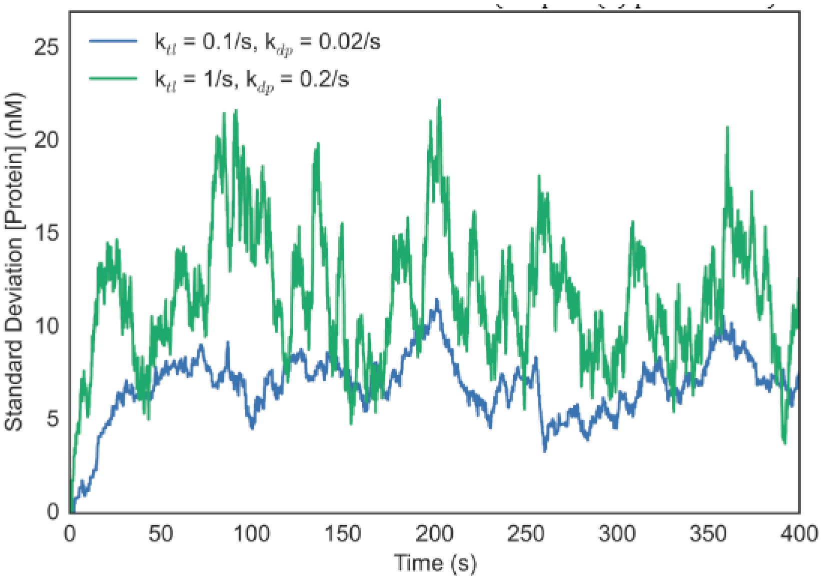
(c) How might you adjust the parameters to keep the average amount of protein roughly constant, and change the standard deviation significantly? Plot an example parameter set and compare to the provided parameters.

Recall that, in the deterministic simulation, we had our steady-state protein concentration at $P_{ss} = \frac{k_{tl} m_{ss}}{k_{dp}} = \frac{k_{tl} k_{tx}}{k_{dp} k_{dm}}$. We wish to keep our average amount of protein roughly the same; to do this, we can simply keep our steady-state protein concentration the same, and modify parameters within that.

To this end, we will try increasing both protein-affecting parameters in the numerator and denominator by the same factor, with the rationale being that, if proteins are both produced and degraded faster, there may be more stochastic variance therein. Below, we increase both k_{tl} and k_{dp} 10-fold, and re-plot the standard deviation plot from part (b):



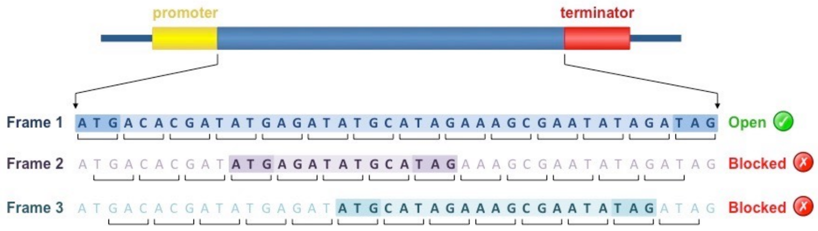
Note that, as we hoped, the average protein concentration is roughly the same as it was in (b). We can also directly compare the standard deviation in protein concentration as a function of time between the two cases (increased rates vs. non-increased rates) to confirm our suspicions:



5. KBase Genome Analysis

For the following question, follow the analysis on Genome Module Part 3: Genome Analysis on KBase, and answer the questions posed therein (Questions 9-13, CQ3).

9: Recall that a codon is interpreted as a set of 3 nucleotides; note that, as we do not have a set frame to work in, there are 3 different possible starting positions that would give rise to different codons, seen below:


promoter
terminator
Frame 1 ATGACACGATATGAGATATGCATAGAAAGCGAATATAGATAG Open
Frame 2 ATGACACGATATGAGATATGCATAGAAAGCGAATATAGATAG Blocked
Frame 3 ATGACACGATATGAGATATGCATAGAAAGCGAATATAGATAG Blocked

The crux here is in remembering that the genome can be transcribed on the positive or negative strand of the DNA. Thus there are two sets of 3 ORFs = 6 rings for "ORFs."

10: %GC content is worth recognizing as an important feature of a genome because of the fact that GC base pairs have 3 Hydrogen Bonds, where AT base pairs have only 2. Thus GC bases are more strongly interactive than AT bases, which amounts to a higher melting temperature in those regions. It also means that processes like transcription or duplication will be harder to initiate in those regions.

11: Searching up "antibiotic" under Module > Browse Features > Search yields approximately 6 results. Specifying this to "antibiotic resistance" yields 2 results.

12: This will vary from answer to answer; any answer that justified why or why not this pathway might be involved in antibiotic resistance received full credit.

13: Once again, this will vary from student to student.

CQ3: One would expect that the presence of an annotated gene/pathway would not guarantee the organism's ability to carry that pathway out; it is possible that this pathway is supplemented by other pathways not immediately connected thereto (i.e., through some post-translational molecular machinery that doesn't reflect in the pathway itself) that may interfere with the pathway's ability to carry out its function.