A Feeling for the Numbers in Biology

1 A Feeling for the Numbers in Biology: Round two
Now that you have written your first estimate vignette in the style of Cell Biology by the Numbers, it’s time to get ready for the second estimate! You will present this second estimate in a 5-minute presentation at the end of the semester. You’re welcome to work in groups.

Write a short paragraph describing the estimate you’re interested in presenting about. Note that the objective at this point is not for you to have a finished estimate, but to have an outline of the calculation you plan to do so that we can give you feedback. Send this paragraph as an email to Hernan, Gabriella and Liz by 4/11 (when Homework 9 will be due).

Spreading the Butter and Cytoskeletal Polymerization

2 Cytoskeletal polymerization
In this problem, we explore how the same master equation approach presented in class to describe the distribution of mRNA molecules can be used to understand how biological polymers such as actin or tubulin are assembled. Specifically, we think of a polymer that can incorporate or lose monomers at one end, as shown in Figure 1. The rate of incorporation is $r$ and the rate with which monomers are lost is $\gamma$. Note that, unlike the mRNA case, the number of monomers that leave the filament in a time interval $\Delta t$ is given by $\gamma \Delta t$ and is independent of the filament length.
(a) The master equation of the probability of a filament having length $n$ at time $t$, $p(n,t)$, is given by

$$p(n, t + \Delta t) = p(n, t) + r\Delta t p(n-1, t) - r\Delta t p(n, t) - \gamma\Delta t p(n, t) + \gamma\Delta t p(n + 1, t).$$  

(1)

Justify why this is the master equation describing the length of a filament by carefully explaining the significance of each term in the master equation.

(b) Using Matlab, solve the master equation assuming $r = 1/s$ and $\gamma = 1.1/s$. Assume that at time $t = 0$ there are no filaments by setting $p(n = 0, t = 0) = 1$ and $p(n > 0, t = 0) = 0$. Be careful about the boundary conditions! In particular, remember that you cannot have filaments with a negative number of monomers. Further, you cannot have Matlab calculate the probability of an arbitrarily long polymer. Hence, you will have to choose a maximum length of the polymer for your calculation. In doing so, make sure that the probability of having $n$ near your chosen maximum polymer length is very close to zero.

(c) Plot the length distribution in steady state.

(d) Explore the master equation for the specific cases of $n = 0$, $n = 1$, $n = 2$ in steady state and show that the distribution can be written as $p(n) = p(0) \left( \frac{r}{\gamma} \right)^n$, which is a geometric distribution.

(e) If you do the math, you can infer that $p(0) = 1 - r/\gamma$ (you don’t need to show this). Now that you have the full distribution, plot it together with your simulation results to show that both approaches are consistent. If you want to compare your results to real data (and figure out more realistic values of $r/\gamma$, you can look at Figure 15.23 of PBoC2. Note that an actin monomer is about 2.4 nm in length.

![Figure 1: Simple model of filament polymerization. Monomers can only be added at one end of the filament.](image-url)
Statistical Mechanics of Binding and Transcriptional Regulation

Note to class: You can complement these problems by reading the paper “A First Exposure to Statistical Mechanics for Life Scientists: Applications to Binding” on the course website.

3 Ion channels and statistical mechanics
In this problem, we will derive a mathematical description of the current passing through a voltage-gated ion channel. To model this channel, we assume that it can exist in an open or closed configuration as shown in Figure 2(A). The thermal fluctuations in the cell result in the channel switching between these states over time as presented in Figure 2(B). Figure 2(C) shows how these fluctuations in channel state can be directly read out from the current flowing through the channel.

(a) Use the statistical mechanics protocol (i.e. calculating the states and weights of the system) to calculate the probability of the channel being in the open state, $p_{\text{open}}$. Assume that the open state has an energy $\varepsilon_{\text{open}}$, and that the energy of the closed state is $\varepsilon_{\text{closed}}$.

(b) Plot $p_{\text{open}}$ as a function of $\Delta \varepsilon = \varepsilon_{\text{open}} - \varepsilon_{\text{closed}}$. Explain what happens in the limits $\varepsilon_{\text{open}} \ll \varepsilon_{\text{closed}}$ and $\varepsilon_{\text{open}} \gg \varepsilon_{\text{closed}}$. What significance does $\Delta \varepsilon = 0$ have for $p_{\text{open}}$?

In a simple model of a voltage-gated ion channel, $\Delta \varepsilon = q(V^* - V)$. Here, $V$ is the voltage applied to the membrane and $q$ is the effective gating charge, which describes the movement of charges along the membrane as the channel configuration changes. You can learn more about this model in section 17.3.1 of PBoC2.

(c) What is the significance of $V^*$? Namely, what happens to the probability of being open when $V = V^*$.

(d) On the website, you will find measurements of $p_{\text{open}}$ vs. $V$ for a sodium-gated ion channel. Write your expression for $p_{\text{open}}$ as a function of $V$ instead of as a function of $\Delta \varepsilon$. Estimate $V^*$ from the data using what you learned in (c). Now that you have $V^*$, to estimate $q$, make a plot where you overlay the data and the model prediction for three different values of $q$ corresponding to -1, -3 or -5 electron charges.

4 Simple repression
In class, we derived a mathematical model of how simple repression dictates gene expression levels. In particular, we showed that the fold-change in gene expression is given by

$$\text{fold-change} = \frac{1}{1 + \frac{R}{N_{NS}} e^{-\beta \Delta \varepsilon}},$$

(2)

where $R$ is the number of repressors, $N_{NS}$ is the number of non-specific sites, and $\Delta \varepsilon$ is the repressor binding energy. Experimentally, we can create bacterial strains that express a
YFP reporter under the control of the repressor. If the YFP fluorescence signal is $F_{\text{reporter}}$, then the fold-change is measured by calculating

$$\text{fold-change} = \frac{F_{\text{reporter}}(R)}{F_{\text{reporter}}(R = 0)}.$$  \hspace{1cm} (3)

However, there is an extra subtlety that has to be taken into account when obtaining such fluorescence measurements. In particular, because of the intrinsic fluorescence of the cells themselves, there is a spurious contribution to the total fluorescence we measure, namely, $F_{\text{total}}$. This quantity is given by

$$F_{\text{total}} = F_{\text{reporter}} + F_{\text{cell}},$$  \hspace{1cm} (4)

where $F_{\text{cell}}$ is the autofluorescence of the cell. As a result, we need to be able to subtract the cells' average autofluorescence if the want to report only on $F_{\text{reporter}}$. Thus, the fold-change is obtained using

$$\text{fold-change} = \frac{F_{\text{total}}(R) - F_{\text{cell}}}{F_{\text{total}}(R = 0) - F_{\text{cell}}}. \hspace{1cm} (5)$$

We already learned how to extract fluorescence levels from microscopy images of bacteria using Matlab. In this problem, you will use the code you wrote in order to test the prediction made by Equation (5). You can get our final version of that code as well as the full bacterial gene expression data set from website. Note that you can also learn more about this protocol from “Computational Exploration: Extracting Level of Gene Expression from Microscopy Images” in chapter 19 of PBoC2.

(a) You can find several images of bacteria under “Full bacterial gene expression data set” in the materials corresponding to the March 19th lecture. This data set corresponds to bacteria containing varying Lac repressor copy numbers, and a reporter containing a three Lac repressor binding sites (called O1, O2 and O3) that control YFP expression. The repressor copy numbers for each strain are given in the following table.
<table>
<thead>
<tr>
<th>Strain</th>
<th>Repressor number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delta</td>
<td>0</td>
</tr>
<tr>
<td>R22</td>
<td>22 ± 4</td>
</tr>
<tr>
<td>R60</td>
<td>60 ± 20</td>
</tr>
<tr>
<td>R124</td>
<td>124 ± 30</td>
</tr>
<tr>
<td>R260</td>
<td>260 ± 40</td>
</tr>
<tr>
<td>R1220</td>
<td>1220 ± 160</td>
</tr>
<tr>
<td>R1740</td>
<td>1740 ± 340</td>
</tr>
</tbody>
</table>

If you want to learn more about the measurements of the repressor copy number, you can refer to Garcia2011c, which is provided on the website. In addition, a strain called “Auto”, lacking a YFP reporter, is provided with the data set. This strain can be used to measure \( F_{\text{cell}} \). Use your code to calculate the fold-change in gene expression as function of repressor copy number by performing the various fluorescence measurements prescribed by Equation 5. Plot your results on a log-log plot.

To get a list of all the images with a given binding site and repressor number, you can do

```matlab
ImageFolder='lacI_titration/'; %Folder where we'll get the images from
BacterialStrain='O2_R22_';
DPhase = dir(fullfile(ImageFolder,BacterialStrain,'Phase*.tif'));
DFluo = dir(fullfile(ImageFolder,BacterialStrain,'YFP*.tif'));
```

Here, we’re getting all phase contrast and fluorescence images corresponding to the O2 binding site and 22 repressors per cell. Note that we’re assuming that your working folder (where your code is saved) contains a folder called lacI_titration where all the data resides. You’re going to run your code for all images in DPhase and DFluo. For each image, you want to save the fluorescence and area values for each cell present. As you analyze new images, you want to save your results in the vectors which we initialize below.

```matlab
CellFluo=[];
CellArea=[];
```

Finally, you want to write a for-loop that goes through each image in the folder and extracts the cell fluorescence and area (saved into NewFluoValues and NewAreaValues, respectively) by concatenating the information into the CellFluo and CellArea vectors by doing

```matlab
CellFluo=[CellFluo,NewFluoValues];
CellArea=[CellArea,NewAreaValues];
```

Once you’re done doing this processing, you can calculate the mean fluorescence per cell and the mean area per cell, and combine these two means to obtain the mean fluorescence per pixel for each bacterial strain.

(b) Estimate the binding energy of O1, O2 and O3 by fitting Equation 5 “by eye” by trying a reasonable range of parameters for \( \Delta \varepsilon \).