Course Work

Stochastic gene expression: Gillespie Simulation

The aim of this coursework is to guide you through implementing the Gillespie algorithm, an algorithm which allows the stochastic simulation of any biological network that can be written as chemical reactions as discussed in the lecture.

We are going to look at the simple model of gene expression discussed in the lectures with transcription (rate k_m), translation (rate k_p), and degradation (rates d_m and d_p) all modelled as first-order processes. Protein and mRNA are denoted by p , m respectively and DNA = 1. We saw that steady-state mRNA is $m_{ss} = k_m/d_m$ and steady-state protein is $p_{ss} = k_m k_p/(d_m d_p)$.

By solving the master equation for this model of gene expression one can show that protein noise (coefficient of variation) η_p follows:

$$\eta_p^2 = \frac{1}{p_{ss}} + \frac{d_p}{d_m + d_p}\frac{1}{m_{ss}} \tag{1}$$

The Gillespie algorithm

Our model of gene expression consists of just 4 reactions. For example, the probability that translation occurs in a small time interval δt is determined partly by the number of mRNA molecules and partly by the kinetic rate of the reaction, i.e., by the product $k_m m \delta t$. In order to simulate chemical reactions, we need to ask which reaction will occur next and when will that reaction occur. The probability of each reaction must be calculated for the point in time that the simulation has reached, and the next reaction chosen according to those probabilities. The time interval that passes before this reaction occurs is another random variable that must be sampled. Repeatedly sampling to find which reaction occurs and then when it occurs is the essence of the Gillespie method. The algorithm is:

- Step 0. Initialization. Define the kinetic rates $k_m = 0.01 s^{-1}$, $d_p = log(2)/180 s^{-1}$, $k_p = 0.04^{-1}$ and $d_p = log(2)/3600 s^{-1}$ and initial numbers of molecules. Let all chemical species have zero concentration initially except the DNA of the gene, which has a copy number of unity: D = 1.

  Set time t = 0.

- Step 1. Calculate the propensity, a_i , for each reaction. The propensity is directly related to the probability of the reaction and equals the kinetic rate for the reaction times the number of reactant molecules available to react. For example, if translation is the second reaction (i.e. labelled with a 2 subscript) $a_2 = k_p m$. Also calculate a_0 , the sum of all the a_i.

- Step 2. Generate two random numbers, r_1 and r_2 (using the rand command). We need to find which reaction will occur, and must relate r_1 to the rolling of a 4 sided die, where each side of the die corresponds to a particular reaction and is weighted by the propensity of the reaction. Mathematically, if a_i is defined as the propensity associated with reaction i, then μ, the number corresponding to the next reaction, satisfies
$$\sum_{i=1}^{\mu-1} < a_0 r_1 < \sum_{i=1}^{\mu} \tag{2}$$
for this value of r_1 . For example, if a_1 is greater than $a_0 r_1$, then the r_1 chosen corresponds to the first reaction. While if a_1 and $a_1 + a_2$ are less than $r_1 a_0$ but $a_1 + a_2 + a_3$, $a_1 + a_2 + a_3 + a_4$, etc., are greater, then the third reaction is chosen. The time interval, τ, between reactions obeys an exponential distribution. Choose τ by
$$\tau = -\frac{log(r_2)}{a_0}$$
which arises as $1 - a_0 \delta t$ is the probability that no reactions occur in time δt.

- Step 3. Update the time variable $t = t + \tau$ and execute reaction μ. For example, if $\mu = 2$, then the number of proteins increases by 1, but the number of mRNAs remain the same (as translation of an mRNA does not degrade it.

- Step 4. Return to Step 1 and repeat until t surpasses the end of the time interval of interest, using a while loop, for example.

Implement this algorithm following the steps described above.

Finding moments of the probability distributions

Run the simulation for say 20 hours of simulated time, and then plot protein, mRNA and DNA as a function of time. Is DNA conserved? Why do the fluctuations of mRNA occur much more frequently than those of protein? Verify Eqs (1). Is it best to use the routines std and mean to calculate the noise? Note that the time intervals between reactions is not uniform. Use the diff command to calculate the mean and variance of protein. Also calculate the mRNA noise and comment if it matches the formula we derived for mRNA noise in the lecture.

Remember that Eqs. (1) is only valid at steady-state. If k_m is changed to 0.001 s^{-1}, would the time needed to accurately measure noise increase or decrease? Why?