Bioen 485/585 Lab 5: Stochastic Chemical Reaction Equations

In this lab, we are going to explore the model of dueling positive and negative feedback by Artyomov et. al. (2007). We will solve these stochastic differential equations using the Exact Stochastic Algorithm and the Tau-Leap Algorithm.

Tutorial:

Matlab implementations of these algorithms (DSDEexact and DSDEtauleap) are provided on the following web page https://faculty.washington.edu/wendyt/software.html. There are also two files that solve the Lotka-Volterra model and the SIR model, and plot the results of a single run. Download these and run one of the examples; you should see a plot of a single stochastic run. You may use these programs as a starting point for your own code solving the problems. Note that the SDE solvers are structured like ODE solvers, in that you need to define the equations (reactions) that they will call. Thus, you should not have to revise the solvers, but you will need to write your reaction function for each system, and these must be written in the formalism we learned in class, with matrices S, P, and vector K.

Hints for analyzing stochastic reaction results:

1. You can use tic and toc to measure elapsed time for a calculation. For example:
```
Tic;
[t,y] = DSDEexact(@LotkaRXN,tspan,IC, params);
Toc
```

2. Averaging results of multiple runs is complicated when the integration time step is different on each run. Also, the integration time steps are so small that you probably don't need to save all this data. You can use the **iterp1** command in MATLAB to interpolate the results to get values at output time steps that you identify. For example, to make a vector y2 of the y values at a prescribe set of times, you can use:

```
times=0:0.1:20;
y2 = interp1( t, y, times)
```

Lab assignment.

1. (10 points) Solve the model described by reactions 1 through 6 in the Artyomov paper, and parameters k1 = 1 k2 = 1 k3 = 100 k4 = 1 k5 = 100 kD = 1 with initial conditions A1(0) = 10, A2(0) = 10, all others = 0.
   a. Create a function called TCellRXN(params) that returns the S, P, and K matrix, similar to the example you downloaded for the tutorial above. Simulate the system using the Exact Stochastic Simulation method(DSDEexact). Run the simulations for 20 time units and plot all variables as a function of time. Repeat the simulation several times, and show representative plots.

   b. Create a function for simulating the same system deterministically. *Hint: you can write the equations from the original reactions, or can use the S, P, and K matrices you already wrote to guide you*. Plot the deterministic simulation.

   c. Explain how your results relate to the conclusions of the Artyomov paper.

2. (10 points) Compare solver efficiency and accuracies.
   a. Efficiency. Measure the time elapsed per simulation while you simulate this stochastic system with 4 methods (the Tau Leap solver (DSDEtauleap) with a relative tolerance of 0.1, 0.01, and 0.001, and the exact solver), for A1(0)= A2(0) = 10, 30, 100, 300, 1,000, and the remaining initial values = 0. Make an efficiency plot by plotting the (log of) simulation time for each method as a function of the A1(0). Describe what you learned about the algorithm efficiency, noting specifically at what value of A1(0)= A2(0) the exact algorithm becomes less efficient that the tau leap for each tolerance level you tested. *Hint: If the time elapsed is stochastic, only worry if it is enough to interfere with your interpretation; in that case, what should you do to reduce the level of noise?*
   b. Accuracy. Run the simulation with the parameters of problem 1, and with initial conditions A1(0) = A2(0) = 100, for as many runs as you can afford to do in 5 minutes, using the exact algorithm. Make a histogram of the number protected (A1prot) at the end of the simulation, using just 10 bins, and calculate the fraction of simulations that resulted in "all on" (A1prot = A1(0)), "all off" (A1prot = 0), "partially on" (any intermediate behavior), or "error" (the simulations crashed, resulted in NAN, a negative value, etc). Also calculate the average and standard deviation of A1prot at the end. Now do this same thing (5 minute of run time) for all four algorithms, including the one you just did. Plot the histograms for all 5 runs, and include the statistics in a table or a bar chart for all 5 runs. Which methods are sufficiently accurate and efficient with this 5 minute limitation to draw a meaningful conclusion about the behavior of the model? *Hint: in general, for stochastic simulations requiring multiple runs, you will want to write and trouble shoot your code doing only a few repeats. Once the code works, including the statistical analysis and figures, you can run enough repeats to get the statistical data you need while working on something else. When you do this final step, you will probably want to know the run time data to estimate when it will be done.*

3. (20 points) Figure 3 of the Artyomov paper shows how the stochastic system changes in behavior as the number of molecules in the initial conditions change. Here we will analyze this question by analyzing statistic results of runs using A1(0) = A2(0) = 10, 100, 1000, 10,000, and 100,000. *Hint: Use what you learned about accuracy and efficiency to choose your simulation methods and number of runs for each condition wisely. It's OK to use different methods or numbers in different conditions, but always use 10 bins. Remember that some messy noise in the figures is OK as long as you can draw reliable conclusions. Even so, remember the hint in the previous problem.*
   a. Plot a histogram for the number protected for each condition.
   b. Plot the average and standard deviation of the fraction protected (A1prot/A1(0)) against system size (A1(0)).
   c. Calculate and plot the fraction of simulations that end in "all on" against the system size.
   d. Describe what you conclude from these plots about how the fraction protected changes with system size. While you can describe the specific plots using variable names like "A1prot," draw your final conclusions using conceptual terms like "switched on" and "system size."