BEBI5009 Homework5

Due 05/24/2018(Thursday) midnight

1. Noise-induced oscillations. (7.8.27)

Stochastic systems can exhibit a range of oscillatory behaviors, ranging from near-perfect periodicity to erratic cycles. To explore this behavior, consider a stochastic relaxation oscillator studied by Jos´e Vilar and colleagues (Vilar et al., 2002). The system involves an activator and a repressor. The activator enhances expression of both proteins. The repressor acts by binding the activator, forming an inert complex. A simple model of the system is

R_1 : (activator synthesis)	$\longrightarrow b_A A$	propensity: $\frac{\gamma_A}{b_A}\frac{\alpha_0+N_A/K_A}{1+N_A/K_A}$
R_2 : (repressor synthesis)	$\longrightarrow b_R R$	propensity: $\frac{\gamma_R}{b_R}\frac{N_A/K_R}{1+N_A/K_R}$
R_3 : (activator decay)	$A \longrightarrow$	propensity: $\delta_A N_A$
R_4 : (repressor decay)	$R \longrightarrow$	propensity: $\delta_R N_R$
R_5 : (association)	$A + R \longrightarrow C$	propensity: $k_C N_A N_R$
R_6 : (dissociation and decay)	$C \longrightarrow R$	propensity: $\delta_A N_C$

Here N_A, N_R and N_C are the molecular counts for the activator, repressor, and activator-repressor complex. The parameter b_A and b_R characterize the expression burst size. The Hill-type propensities for the synthesis reactions are not well-justified at the molecular level, but these expressions nevertheless provide a simple formulation of a stochastic relaxation oscillator.

(a) Take parameter values $\gamma_A = 250$, $b_A = 5$, $K_A = 0.5$, $\alpha_0 = 0.1$, $\delta_A = 1$, $\gamma_R = 50$, $b_R = 10$, $K_R = 1$, $k_C = 200$, and $\delta_R = 0.1$. Run simulations of this model and verify its quasi-periodic behavior.

Please simulate the stochastic model using (1) Gillespie algorithm (SSA; 7.3.6) (2) First reaction method (7.8.25; please see the note.). Please also get ensembles and averages of sample paths for both methods.

(b) The deterministic version of this model is

$$\begin{aligned}
\frac{d}{dt}a(t) &= \gamma_A \frac{\alpha_0 + a(t)/K_A}{1 + a(t)/K_A} - k_C a(t) r(t) - \delta_A a(t) \\
\frac{d}{dt}r(t) &= \gamma_R \frac{a(t)/K_R}{1 + a(t)/K_R} - k_C a(t) r(t) + \delta_A c(t) - \delta_R r(t) \\
\frac{d}{dt}c(t) &= k_C a(t) r(t) - \delta_A c(t),
\end{aligned}$$

where a, r, and c are the concentrations of activator, repressor, and complex. Run a

simulation with the same parameter values as in part (a). Does the system exhibit oscillations? How is the behavior different if you set $\delta_R = 0.2$?

(c) The contrast between the behavior of the models in parts (a) and (b), for $\delta_R = 0.1$, can be explained by the excitability of this relaxation oscillator. Run two simulations of the deterministic model ($\delta_R = 0.1$), one from initial conditions (a, r, c) = (0, 10, 35) and another from initial conditions (a, r, c) = (5, 10, 35). Verify that in the first case, the activator is quenched by the repressor, and the system remains at a low-activator steady state, whereas in the second case, this small quantity of activator is able to break free from the repressor and invoke a (single) spike in expression. Explain how noise in the activator abundance could cause repeated excitations by allowing the activator abundance to regularly cross this threshold. This is referred to as noise- induced oscillation.

Note: First-reaction method (7.8.25) involves stepping from reaction event to reaction event. However, rather than sample the next reaction and the waiting time separately (as in the SSA), the first-reaction algorithm samples a waiting time for each reaction in the network, and then selects the shortest of this collection of times. This selection specifies the identity of the next reaction and the elapsed time.

1. Initialise the starting point of the simulation with $t := 0$, rate constants $c = (c_1, \ldots, c_v)$ and initial state $x = (x_1, \ldots, x_u)$.
2. Calculate the reaction hazards $h_i(x, c_i),\ i = 1, 2, \ldots, v$.
3. Simulate a putative time to the next type i reaction, $t_i \sim Exp(h_i(x, c_i)),\ i = 1, 2, \ldots, v$.
4. Let j be the index of the smallest t_i.
5. Put $t := t + t_j$.
6. Update the state x according to the reaction with index j. That is, set $x := x + S^{(j)}$.
7. Output t and x.
8. If $t < T_{max}$ return to step 2.