Imperial College London

5 October 2011

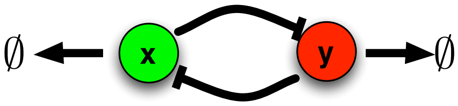
Department of Bioengineering

BE3-HMIB – Modelling in Biology (MiB), Dr Guy-Bart Stan & Dr Aldo Faisal

Training coursework 3

A genetic switch

The system represented below consists of two mutually repressing genes, i.e. the proteins they express act as transcription repressors for the other gene. The produced proteins are also degraded by the cell.



The concentration of proteins x and y can be modelled as

$$\begin{aligned} \frac{\mathrm{d}x}{\mathrm{d}t} &= \frac{\alpha}{1+y^4} - \gamma x, \\ \frac{\mathrm{d}y}{\mathrm{d}t} &= \frac{\beta}{1+x^4} - \gamma y, \end{aligned} \tag{1}$$

with positive parameters α, β and γ.

1. Write Matlab code to solve the system (1) using the solver `ode45`. Choose a positive initial condition and try your code with parameters $\alpha = \beta = 1$ and $\gamma = 0.3$. Plot the time-courses $x(t)$ and $y(t)$.
2. Compute analytically the nullclines of (1) (check your lecture notes!) and superpose both nullclines on a single graph. Identify the equilibria of the system by computing the intersection points between the nullclines; use the Matlab commands `solve` and `fsolve` (check the Matlab help: you will need several initial solution guesses to use `fsolve`). What is the stability of each equilibrium point?
3. Solve the system twice, once for initial condition $x(0) = 1, y(0) = 2$, and then for $x(0) = 2, y(0) = 1$. In each case, plot the *phase plane* trajectories and overlay them with the nullclines. Explain what you observe.
4. Now we want to solve the system for lots of initial conditions to see how trajectories evolve. Use the Matlab command `rand` to generate 20 random initial conditions. Solve the system in each case and plot all the 20 trajectories and the nullclines on the same graph. Explain what you observe.
5. Repeat the previous step, but this time with parameters $\alpha = 0.4, \beta = 1$ and then with $\alpha = 0.8, \beta = 4$. Compare with the behaviour in the previous case.

In this coursework you may need to use the following Matlab commands: `ode45`, `function`, `plot`, `ezplot`, `hold`, `solve`, `fsolve`, `rand`. You can check the Matlab help by using `help COMMAND`.