MATHEMATICAL AND COMPUTER MODELING OF NONLINEAR BIOSYSTEMS I

COMPUTER LABORATORY XII: Basic gene expression models, part of p53 signaling pathway

Ph. D. Programme 2013/2014

PhD STUDIES

KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY

Project co-financed by European Union within the framework of European Social Fund

Gene expression from a mathematical point of view

$$\begin{tikzcd}
DNA \arrow[r, "k_r"] & mRNA \arrow[r, "\gamma_r"] \arrow[d, "k_p"'] & \emptyset \\
 & P \arrow[r, "\gamma_p"] & \emptyset
\end{tikzcd}$$

Reaction–rate equations — high molecules counts

$$\begin{cases} \dot{r}(t) = k_r - \gamma_r r(t)\,, \\ \dot{p}(t) = k_p r(t) - \gamma_p p(t)\,. \end{cases}$$

Birth–death process — low molecules counts

$$\begin{aligned} \frac{dP_{r,p}(t)}{dt} &= k_r(P_{r-1,p}(t) - P_{r,p}(t)) + \gamma_r((r+1)P_{r+1,p}(t) - rP_{r,p}(t)) \\ &+ k_p(rP_{r,p-1}(t) - rP_{r,p}(t)) + \gamma_p((p+1)P_{r,p+1}(t) - pP_{r,p}(t))\,. \end{aligned}$$

Stochastic simulation

$$\begin{tikzcd}
DNA \arrow[r, "k_r"] & mRNA \arrow[r, "\gamma_r"] \arrow[d, "k_p"'] & \emptyset \\
 & P \arrow[r, "\gamma_p"] & \emptyset
\end{tikzcd}$$

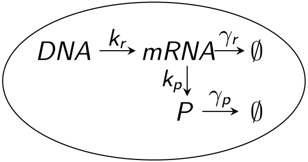
- r - number of mRNA molecules
- p - number of proteins

Reaction intensities:
$Q_1 = k_r,\ Q_2 = \gamma_r r,\ Q_3 = k_p r,\ Q_4 = \gamma_p p.$

Time till next reaction:
$dt \sim Exp(\sum Q_i).$

Reaction selection from discrete distribution $Q_i / \sum Q_i$.

Exercise 1 - stochastic simulation



$$\begin{tikzcd}
DNA \arrow[r, "k_r"] & mRNA \arrow[r, "\gamma_r"] \arrow[d, "k_p"'] & \emptyset \\
 & P \arrow[r, "\gamma_p"] & \emptyset
\end{tikzcd}$$

Implement MATLAB function that returns stochastic trajectory of the above system.

Input arguments:

- time horizon,
- parameters (structure),
- initial condition.

Output as a structure with time mesh and system state.

Write MATLAB script for drawing exemplary trajectory.

Exercise 1 - solution (part 1)

```
function traj = generateTrajectory(  T, par, init )

    function Q = calculateIntensities(r, p)
       Q = zeros(4,1);
       Q(1) = par.kr; Q(2) = par.gammar*r;
       Q(3) = par.kp*r; Q(4) = par.gammap*p;
    end

    function n = selectReaction(Q)
        u = rand(); Q = Q/sum(Q);
        if u<Q(1)
            n = 1;
        elseif u<Q(1)+Q(2)
            n = 2;
        elseif  u<Q(1)+Q(2)+Q(3)
            n = 3;
        else
            n = 4;
        end
    end
```

Exercise 1 - solution (part 2)

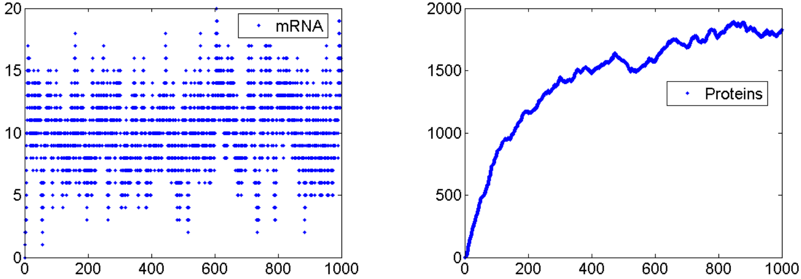
```
    r = init(1); p = init(2);
    traj.t = 0; traj.y = init(:); t = 0;
    while t<T
       Q = calculateIntensities(r, p);
       dt = -log(rand())/sum(Q);
       t = t+dt;
       traj.t = [traj.t t];
       which = selectReaction(Q);
       switch which
           case 1
            r = r+1;
           case 2
            r = r-1;
           case 3
            p = p+1;
           case 4
            p = p-1;
       end
       traj.y = [traj.y [r; p]];
    end
end
```

Exercise 1 - script for drawing exemplary trajectory

```
clear all

par.kr = 2;
par.gammar = 1/5;
par.kp = 1;
par.gammap = 1/180;
init = [0; 0];

traj = generateTrajectory(1000,par,init);

figure(1)
plot(traj.t, traj.y(1,:),’LineStyle’,’none’,’Marker’,’.’)
legend({’mRNA’})
xlim([0 1000])

figure(2)
plot(traj.t, traj.y(2,:),’LineStyle’,’none’,’Marker’,’.’)
legend({’Proteins’})
xlim([0 1000])
```

Result

Exemplary realization of the process



Exercise 2 - deterministic solution

Implement MATLAB function that returns solution to the deterministic version of the system

$$\begin{cases} \dot{r}(t) = k_r - \gamma_r r(t)\,, \\ \dot{p}(t) = k_p r(t) - \gamma_p p(t)\,. \end{cases}$$

Input arguments:

- time horizon,
- parameters (structure),
- initial condition.

Direct solver output.

Exercise 2 - solution

```
function sol = deterministicModel( T, par, init )

    sol = ode45(@model,[0 T], init);

    function y = model(~,x)
       y = zeros(2,1);
       y(1) = par.kr-par.gammar*x(1);
       y(2) = par.kp*x(1)-par.gammap*x(2);
    end

end
```

Exercise 3 - comparizon

Does the average behaviour of the stochastic system correspond to the deterministic solution?

Write MATLAB program that plots the average and standard deviation of N stochastic trajectories (on a given mesh).

Overlay on the plot solution to the deterministic system.

Exercise 3 - step 1

Implement MATLAB function that for a given stochastic trajectory returns state of the system at specified time points.

Input arguments:

- stochastic trajectory (from Ex. 1)
- time mesh

Output: matrix with states of the system at specified time points arranged in rows.

Exercise 3 - step 1 solution

```
function trajMesh = returnAtMesh( tmesh, traj )

    trajMesh = zeros(2,length(tmesh));
    for i = 1:length(tmesh)
        indx = find( traj.t - tmesh(i) <= 0, 1, 'last');
        trajMesh(:,i) = traj.y(:,indx);
    end

end
```

Exercise 3 - step 2

Implement MATLAB script that performs the comparison.

Exercise 3 - step 2 solution

```
clear all;

par.kr = 2;
par.gammar = 1/5;
par.kp = 1;
par.gammap = 1/180;
init = [0; 0];

N = 20;

tmesh = 0:5:100;

trajR = zeros(N,length(tmesh));
trajP = zeros(N,length(tmesh));
```

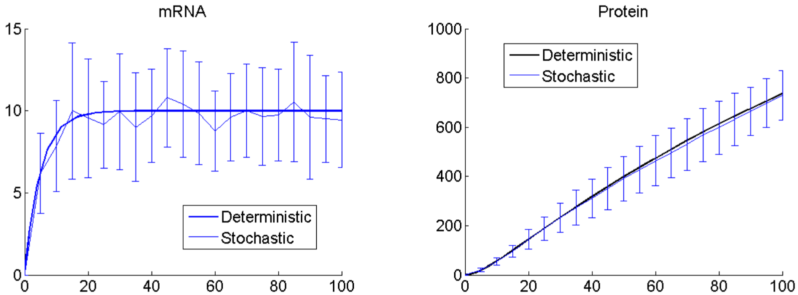
Exercise 3 - step 2 solution

```
for i = 1:N
   display(i)
   tmp = generateTrajectory(tmesh(end),par,init);
   tmp = returnAtMesh(tmesh, tmp);
   trajR(i,:) = tmp(1,:);
   trajP(i,:) = tmp(2,:);
end

determ = deterministicModel(tmesh(end),par,init);
```

Exercise 3 - step 2 solution

```
figure(3)
clf
hold on
plot(determ.x, determ.y(1,:),'LineWidth',2,'Color','b');
errorbar(tmesh, mean(trajR), std(trajR));
hold off
xlim([tmesh(1) tmesh(end)])
title('mRNA')
legend('Deterministic', 'Stochastic')

figure(4)
clf
hold on
plot(determ.x, determ.y(2,:),'LineWidth',2,'Color','k');
errorbar(tmesh, mean(trajP), std(trajP));
hold off
xlim([tmesh(1) tmesh(end)])
title('Protein')
legend('Deterministic', 'Stochastic')
```

Result



Average behaviour of the stochastic system correspond to the deterministic system.

Is it always like that?

L-V reaction system

Consider the following reaction scheme:

$V \longrightarrow 2V$ at rate r

$V + P \longrightarrow 2P$ at rate b

$P \longrightarrow \emptyset$ at rate s

Exercise 4

Compare stochastic and deterministic descriptions of the above system. Does the average of stochastic simulation agree with the deterministic solution?

Modify scripts written in Ex. 1-3.

Exercise 4 - solution

Modify function generating trajectory (step 1):

```
switch which
    case 1
     r = r+1;
    case 2
     r = r-1;
     p = p+1;
    case 3
     p = p-1;
end
```

Exercise 4 - solution

Modify function generating trajectory (step 2):

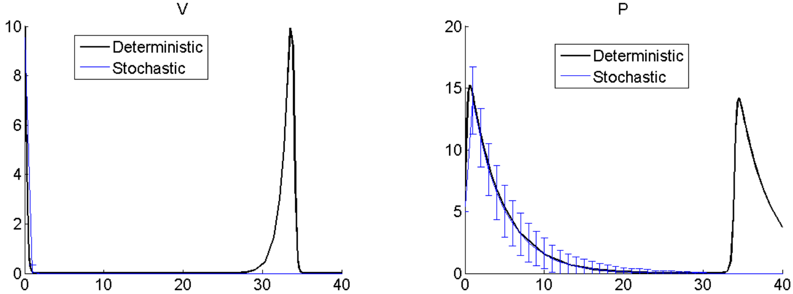
```
function Q = calculateIntensities(r, p)
   Q = zeros(3,1);
   Q(1) = par.r*r;
   Q(2) = par.b*r*p;
   Q(3) = par.s*p;
end

function n = selectReaction(Q)
    u = rand();
    Q = Q/sum(Q);
    if u<Q(1)
        n = 1;
    elseif u<Q(1)+Q(2)
        n = 2;
    else
        n = 3;
    end
end
```

Exercise 4 - solution

Modify function returning deterministic solution:

```
function sol = deterministicModelLV( T, par, init )

    sol = ode45(@model,[0 T], init);

    function y = model(~,x)
       y = zeros(2,1);
       y(1) = par.r*x(1)-par.b*x(1)*x(2);
       y(2) = -par.s*x(2)+par.b*x(1)*x(2);
    end

end
```

Results



Averaged stochastic simulation doesn't agree with the deterministic system.