MATHEMATICAL AND COMPUTER MODELING OF NONLINEAR BIOSYSTEMS I

COMPUTER LABORATORY XIII: Logistic model with vaccinations, L-V with harvesting

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

Modified LV model with harvesting

Consider predator-prey system described by the following set of ordinary differential equations:

$$\begin{cases} \dot{V} = rV\left(1 - \frac{V}{K}\right) - aVP\,, \\ \dot{P} = -sP + bVP\,. \end{cases}$$

Assume that every *Nth* day some fraction (always the same) of preys V is harvested (instantaneous removal).

What is the optimal fraction of removed preys that maximises the total harvest in a period of one year?

Step 1

Implement MATLAB function that returns solution to the system

$$\begin{cases} \dot{V} = rV\left(1 - \frac{V}{K}\right) - aVP\,, \\ \dot{P} = -sP + bVP\,. \end{cases}$$

in which every *Nth* day some fraction (always the same) of preys V is harvested (instantaneous removal).

Input arguments

- time horizon
- harvesting moments and fractions in a structure
- systems parameters
- initial condition

Output arguments

- structure with system solution
- total amount of harvested preys

Draw exemplary solution.

Step 1 - solution (part 1)

```
function [sol, harvested] = solveLVharvesting(T, harvest, par, init )
    tmesh = [0 harvest.t(harvest.t<T) T];
    harvested = 0;

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

Step 1 - solution (part 2)

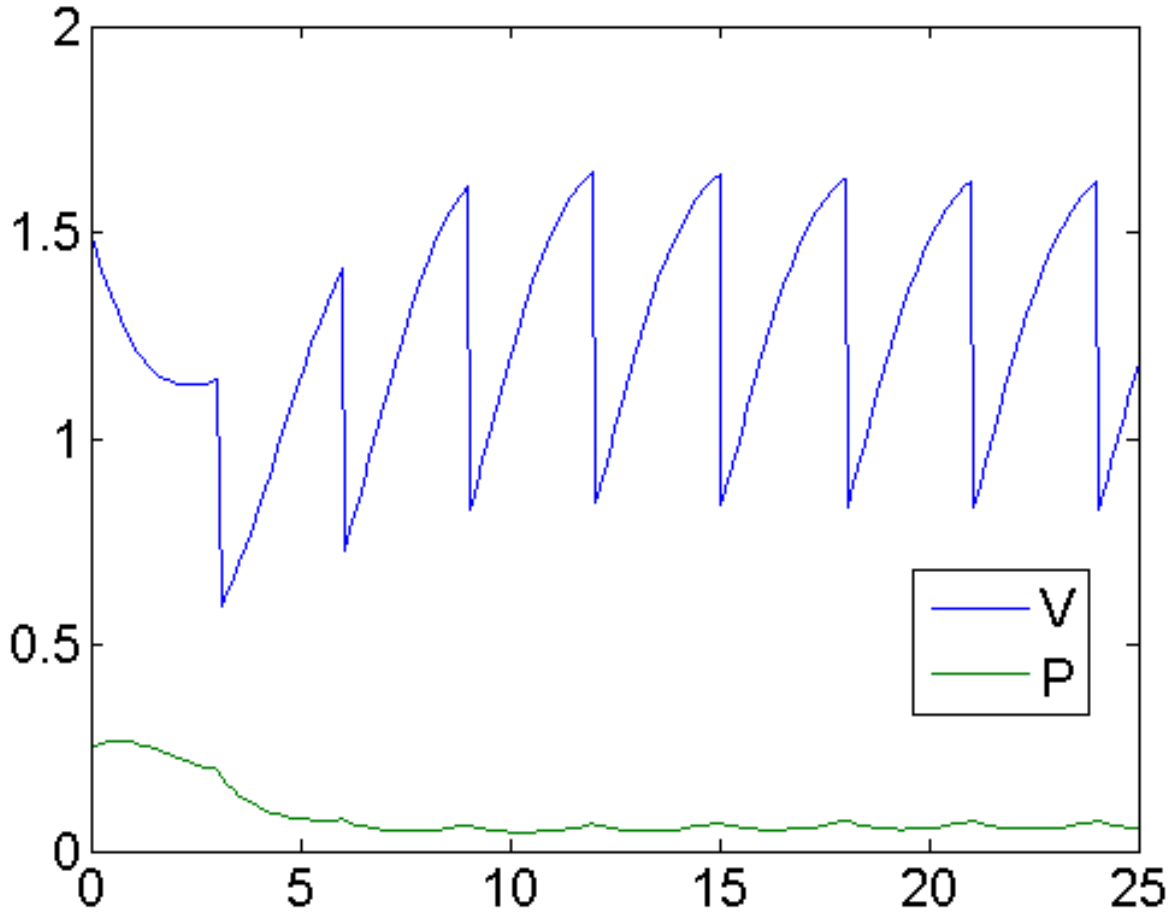
```
    sol.x = 0;
    sol.y = init;
    for i = 2:length(tmesh);
        opt = odeset('RelTol',1e-8,'AbsTol',1e-8);
        solTmp = ode45(@model,[tmesh(i-1) tmesh(i)],init,opt);
        sol.x = [sol.x solTmp.x(2:end)];
        sol.y = [sol.y solTmp.y(:,2:end)];

        init = solTmp.y(:,end);
        init(1) = init(1)*(1-harvest.frac(i-1));

        harvested = harvested+sol.y(1,end)*harvest.frac(i-1);
    end
end
```

Drawing exemplary solution

```
clear all;

params.r = 1;
params.a = 2;
params.b = 1;
params.s = 1.3;
params.K = 2;

harvest.t = 3:3:25;
harvest.frac = 0.5*ones(size(harvest.t)+1);
T = 25;

init = [1.5; 0.25];

    [sol, harv] = solveLVharvesting(T, harvest, params, init );

figure(1)
clf
plot(sol.x, sol.y)
legend('V','P')
```

Result

Exemplary solution



Step 2

Implement MATLAB function that plots the total amount of harvested preys for different harvest fractions.

Input arguments

- time horizon
- interval between each harvest
- systems parameters
- initial condition
- density of plot mesh

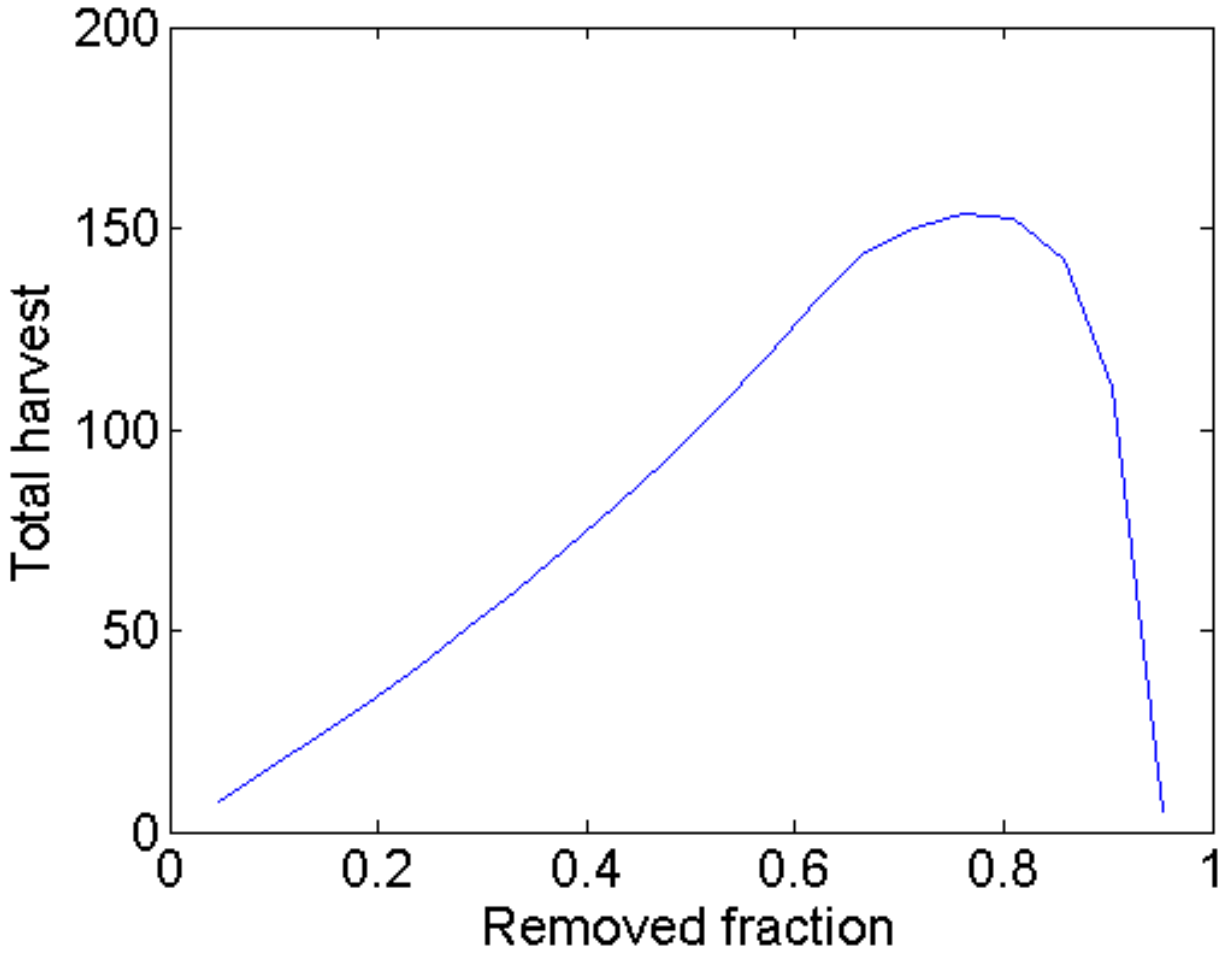
Step 2 - solution (part 1)

```
function plotHarvest(T, sep, params, init, n )

    harvest.t = sep:sep:T;

    frac = linspace(0,1,n+2);
    out = zeros(1,n);

    for i = 2:length(frac)-1
        harvest.frac = frac(i)*ones(size(harvest.t)+1);
        [~, harv] = solveLVharvesting(T, harvest, params, init );
        out(i-1)=harv;
    end

    figure(1)
    plot(frac(2:end-1), out)
    xlabel('Removed fraction')
    ylabel('Total harvest')
end
```

Step 2 - solution (part 2)

```
clear all;

params.r = 1;
params.a = 2;
params.b = 1;
params.s = 1.3;
params.K = 2;

init = [1.5; 0.25];

n = 20;

plotHarvest(365, 3, params, init, n )
```

Result



Step 3

Implement MATLAB function that finds numerically the optimal harvest fraction, that is the fraction for which the total amount of harvested preys is maximal.

Input arguments

- time horizon
- interval between each harvest
- systems parameters
- initial condition

Output arguments

- optimal harvest fraction
- total amount of harvested preys for optimal fraction

Step 3 - solution

```
function [frOpt, harvTot] = optimalHarvest(T, sep, params, init )

    harvest.t = sep:sep:T;

    opt = optimset('Display','iter');
    [frOpt, harvTot] = fminsearch(@F, 0.5, opt);
    harvTot = -harvTot;

    function y = F(x)
        harvest.frac = x*ones(size(harvest.t)+1);
        [~, y] = solveLVharvesting(T, harvest, params, init );
        y = -y;
    end

end
```

Final result

Optimal fraction: 0.7769
Total harvest: 153.813