ISC3, Fall 2020 (A22)
Computer works report 004

Pierre Lague

October 14, 2022

1 Exercice 1 : cell proliferation model

This exercise aims to picture how cells proliferate under a certain initial state through time. We consider the following system :

$$\frac{dn(t)}{dt} = \beta n(t)(n^{\infty} - n(t))$$

With a given initial state, solve the differential equation problem with the built-in function in Scilab.

```
n_0 = 10
b = 10^(-6)
n_inf = 10^6
function xdot=f1(t, x_t)
    xdot = b*x_t*(n_inf - x_t)
endfunction
t0 = 0;
td = linspace(t0, 20, 200);
num_res = ode(n_0, t0, td, f1)
plot(log(num_res))
```

This outputs the following graph.

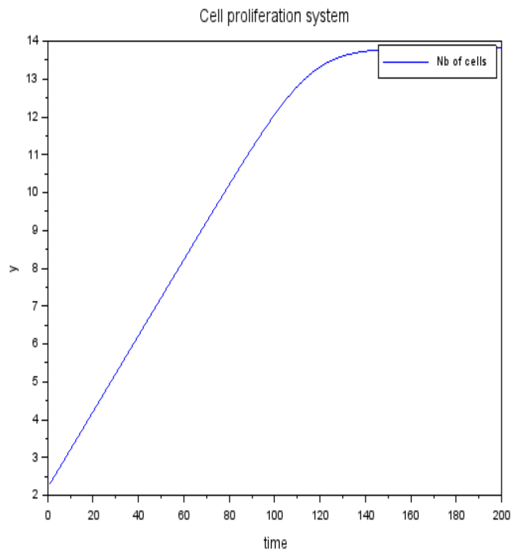

Figure 1: Cell proliferation system. This system is convergent and seems to be constant at $t >= 140$

Then it is important to verify the numeric output with the real analytic output (when the system is not too complex and is analytically solvable). The exact analytic solution is the following :

$$n(t) = \frac{n_0 n^{\infty} \exp(\beta n^{\infty} t)}{n\infty - n_0 + n_0 \exp(\beta n^{\infty} t)} \forall t >= 0$$

The implementation is the following :

```
x_t = (n_0*(n_inf*exp(b*n_inf*td)))./(n_inf - n_0 + (n_0*exp(b*n_inf*td)))

plot(log(x_t), '-r')
title('Cell proliferation system (analytic solution)')
xlabel('time')
ylabel('y')
legend(['Nb of cells'])
```

And outputs the following plot :

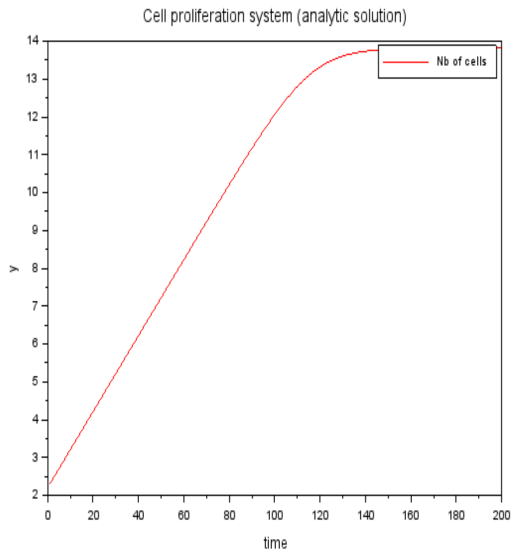

Figure 2: The analytical solution is obviously identical to the numeric solution.

2 Exercise 2 : Duffing equation

The Duffing differential equation

$$\ddot{x} + \beta\dot{x} - x + \gamma x^3 = F\cos(\omega t)$$

can be written as a system of first-order differential equations as

$$\begin{aligned}\dot{x} &= v,\\ \dot{v} &= -\beta v + x - \gamma x^3 + F\cos(\omega t).\end{aligned}$$

Consider the function $\boldsymbol{F}(t, \boldsymbol{x}), \boldsymbol{x} = (x_1, x_2)$ defined by

$$\boldsymbol{F}(t, \boldsymbol{x}) = \begin{pmatrix} x_2 \\ -\beta x_2 + x_1 - \gamma x_1^3 + F\cos(\omega t) \end{pmatrix}.$$

Write a function that implements $\boldsymbol{F}$. Then solve the Duffing equation using the built-in function of Scilab. Initial conditions are $x^0 = (0, 0)^T, \beta = 0.05, y = 1.0, \omega = 1.0, F = 6.0$ and the time $t : 0 : 0.05 : 200$.

The implementation is as follow :

```
//Exercice 2 - Duffing Equation
x_o = [0;0]
bta = 0.05
y = 1
w = 1
F_cons = 6
td = 0:0.05:200

function f=F(t, x)
    f(1) = x(2,:)
    f(2) = -bta*x(2,:)+x(1,:) - y*x(1,:)^3 + F_cons*cos(w*t)
endfunction

t0 = 0;
num_res = ode(x_o, t0, td, F)

subplot(1, 2, 1)
plot(num_res(1,:), num_res(2,:), 'g')
title('state space Duffing Equation')
xlabel('F(t, x)')
ylabel('td')

subplot(1, 2, 2)
plot(td, num_res(1,:), 'r', td, num_res(2,:), 'b')
xlabel('F(t, x)')
ylabel('td')
title('Time series representation of the two components')
legend(['x1(t)', 'x2(t)'])
```

And outputs the following plot :

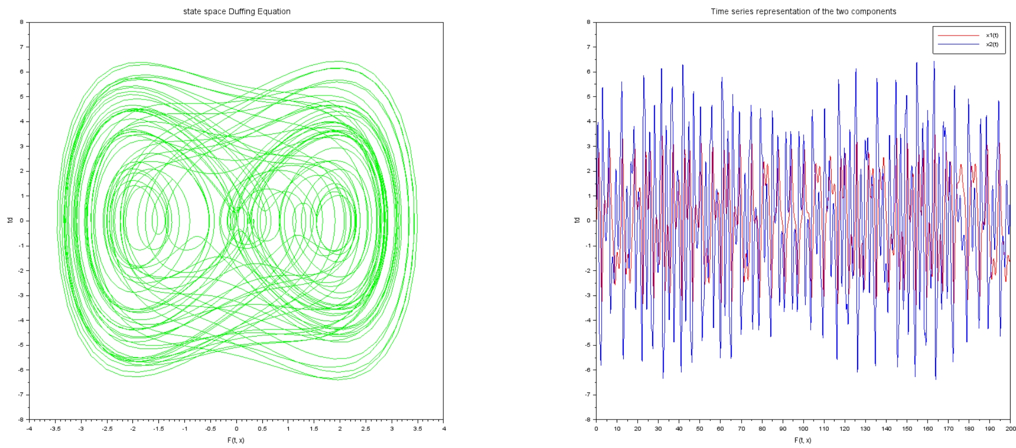

Figure 3: The Duffing Equation System is a highly unstable system. On the time series representation, the two components oscillate around 0. On the state space, we could imagine a tore-like shape in 3D.

3 Exercise 3 : SEIRS Covid-19 differential model

Let's consider the following SEIRS virus outbreak model :
$\dot{S}(t) = -\frac{\beta SI}{N} + \xi R(1 - R/N)$
$\dot{E}(t) = \frac{\beta SI}{N} - \sigma E$
$\dot{I}(t) = \sigma E - \gamma I$
$\dot{R}(t) = \gamma I - \xi R(1 - R/N)$

Where $N = S + E + I + R$.
With given initial conditions, resolve the system with the built-in functions.

```
function Xdot = SEIR(t,X)
betas = 0.5;
sigma = 1/5;
gamar = 1/6;
epsilon = 1/90;

S = X(1); E=X(2); I=X(3); R=X(4);
N = S + E + I + R;
Xdot(1) = -betas*S*I/N + epsilon*R*(1 - R/N);
Xdot(2) = betas*S*I/N -sigma*E;
```

```
Xdot(3) = sigma*E - gamar*I;
Xdot(4) = gamar*I - epsilon*R*(1 - R/N);
endfunction

// initial conditions
X0 = [70*10^6; 2000; 100; 0];
t0 = 0;
t = 0:1:700;
Xsol = ode(X0, t0, t, SEIR)//solving the differential equations
S = Xsol(1,:);
E = Xsol(2,:);
I = Xsol(3,:);
R = Xsol(4,:);
clf;

//plotting the system and evolution of the SEIR through time
plot(t, S, '-b', t, E, '-y', t, I, '-g', ...
t, R, '-r', 'LineWidth', 2); xgrid
title('SEIRS Covid-19 differential model')
xlabel('y -> days')
ylabel('x -> number of people')
legend(['Susceptible(t)';'Exposed(t)';'Infectious(t)';'Recovered(t)']);
```

This script outputs the following plot :

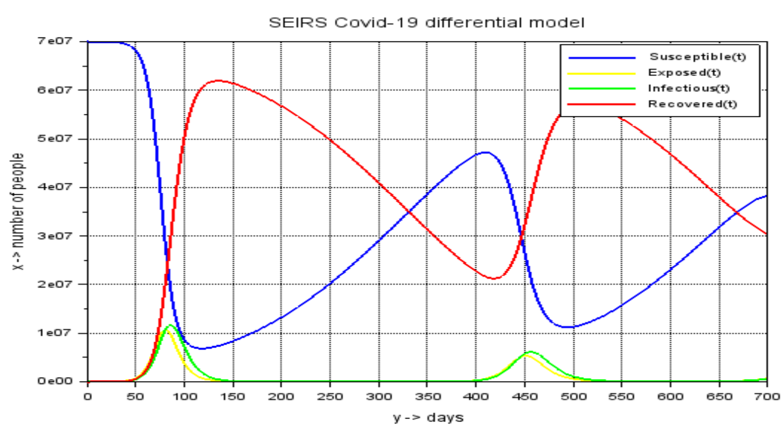

Figure 4: SEIR Covid-19 differential model t = 700

One can think that the number of s(t) and r(t) will converge to a stable state. However, taking a duration of 2000 days, we notice that s(t) and r(t) will converge to 2 independent limits and that e(t) and i(t) will converge to the same limit. We also notice a cyclic behaviour with less and less amplitude.

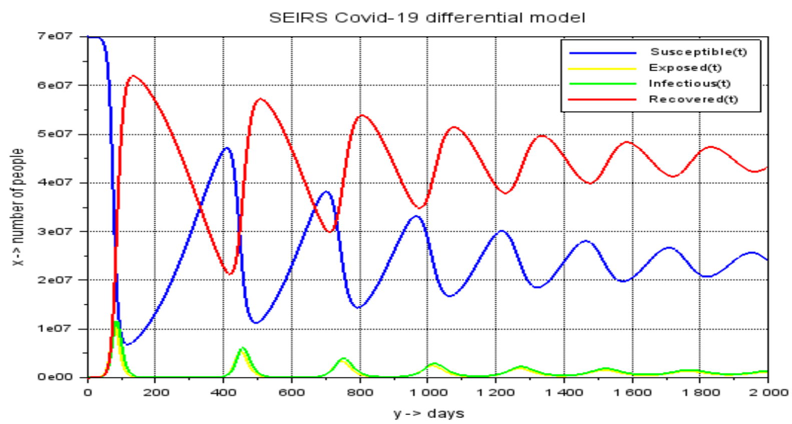

Figure 5: SEIR Covid-19 differential model t = 2000