Out: January 13, 2020
Due: January 26, 2020

NTNU – Trondheim
Norwegian University of Science and Technology

TTK4130 Modeling and Simulation
Assignment 1

Introduction

The conditions for the existence of a solution to an ODE, as well as the linearization and study of the stability of the dynamic system that this ODE represents, constitute fundamental knowledge and techniques for modeling and simulation.

Regarding modeling, it is mostly performed by using physical principles, such as Newton's second law, the conservation of mass, momentum, energy, etc. However, one can also model dynamic systems using statistical relations or heuristic assumptions.

An example of the later approach are the classical Lokta-Volterra equations:

$$\begin{cases} \dot{x} &= \alpha x - \beta x y \\ \dot{y} &= -\gamma y + \delta x y \,, \end{cases} \tag{1}$$

which model the dynamics of two isolated animal populations: The prey and the predators.

Here, x is the number of prey (e.g. tapirs) and y is the number predators (e.g. jaguars). In this model, $\alpha > 0$ is the birth-rate of the prey and $\gamma > 0$ is the death-rate of the predators. The interactions between these populations are modeled by the product xy, which is a measure of the number of encounters. Some encounters result in the death of prey, and enough eaten prey enable the predators to have offspring. How much the populations vary due to this interactions is described by the parameters $\beta > 0$ and $\delta > 0$.

Problem 1 (Linearization and stability)
Consider the following 3 dynamic systems:

1. $$\begin{cases} \dot{x}_1 &= a u_1 - b\sqrt{x_1} \\ \dot{x}_2 &= \frac{a}{x_1}\left(u_1(u_2 - x_2) + c(u_3 - x_2)\right) \,, \end{cases} \tag{2}$$

   where $u_1, u_2, u_3 > 0$ are inputs and $a, b, c > 0$ are parameters.

2. $$\ddot{x} + c\dot{x} + g\left(1 - \left(\frac{x_d}{x}\right)^{\kappa}\right) = 0 \,, \tag{3}$$

   where $c, g, x_d > 0$ and $\kappa > 1$ are parameters, and $x > 0$.

3. $$\dot{x} = \begin{cases} y - \frac{x}{\ln\sqrt{x^2+y^2}} & , [x,y] \neq [0,0] \\ 0 & , [x,y] = [0,0] \end{cases} \qquad \dot{y} = \begin{cases} -x - \frac{y}{\ln\sqrt{x^2+y^2}} & , [x,y] \neq [0,0] \\ 0 & , [x,y] = [0,0]. \end{cases} \tag{4}$$

(a) Write the systems in state-space form if there are not in this form already.

Solution: Only system 2 is not in state-space form.
As it is standard, we define $x_1 = x$ and $x_2 = \dot{x}$ as the state-variables. Hence,

$$\begin{cases} \dot{x}_1 &= x_2 \,, \\ \dot{x}_2 &= -c x_2 - g\left(1 - \left(\frac{x_d}{x_1}\right)^{\kappa}\right) . \end{cases}$$

(b) Find the equilibrium points of each system.
For system 1, find the equilibrium points given a constant input $\mathbf{u} = [u_1,\, u_2,\, u_3]^T$.
Hint: For system 3, $\dot{x} = 0$ *and* $\dot{y} = 0$ *implies* $y\dot{x} - x\dot{y} = 0$.

Solution: We solve the equation $0 = \boldsymbol{f}(\mathbf{x}^*, \mathbf{u}^*)$ for each system. This gives:

1. The only equilibrium point is

$$\mathbf{x}^* = \begin{bmatrix} \frac{a^2 u_1^2}{b^2} \\ \frac{u_1 u_2 + c u_3}{u_1 + c} \end{bmatrix} \text{ with } \mathbf{u}^* = \mathbf{u}.$$

2. The only equilibrium point is $\mathbf{x}^* = [x_d, 0]^T$.

3. $0 = y\dot{x} - x\dot{y} = x^2 + y^2$. Hence, $\mathbf{x}^* = [0, 0]^T$ is the only equilibrium point.

(c) Linearize each system around each of its equilibrium points.
Hint: For system 3, one can avoid cumbersome calculations using that $\frac{r}{\log|r|} = o(r)$.

Solution: The linearized systems have the form $\dot{\delta x} = \mathbf{A}\delta x + \mathbf{B}\delta u$, where $\mathbf{x} = \mathbf{x}^* + \delta\mathbf{x}$, $\mathbf{u} = \mathbf{u}^* + \delta\mathbf{u}$ (if input present) and

$$\mathbf{A} = \left.\frac{\partial(\boldsymbol{f}(\mathbf{x}, \mathbf{u})}{\partial \mathbf{x}}\right|_{\mathbf{x}=\mathbf{x}^*, \mathbf{u}=\mathbf{u}^*}$$

$$\mathbf{B} = \left.\frac{\partial(\boldsymbol{f}(\mathbf{x}, \mathbf{u})}{\partial \mathbf{u}}\right|_{\mathbf{x}=\mathbf{x}^*, \mathbf{u}=\mathbf{u}^*} \quad \text{(if input present).}$$

1. The Jacobians are

$$\frac{\partial \boldsymbol{f}(\mathbf{x}, \mathbf{u})}{\partial \mathbf{x}} = \begin{bmatrix} -\frac{b}{2\sqrt{x_1}} & 0 \\ -\frac{a}{x_1^2}\left(u_1(u_2 - x_2) + c(u_3 - x_2)\right) & -\frac{a}{x_1}(u_1 + c) \end{bmatrix}$$

$$\frac{\partial \boldsymbol{f}(\mathbf{x}, \mathbf{u})}{\partial \mathbf{u}} = \begin{bmatrix} a & 0 & 0 \\ \frac{a}{x_1}(u_2 - x_2) & \frac{a u_1}{x_1} & \frac{ac}{x_1} \end{bmatrix}.$$

Hence,

$$\mathbf{A} = \begin{bmatrix} -\frac{b^2}{2 a u_1} & 0 \\ 0 & -\frac{b^2}{a u_1^2}(u_1 + c) \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} a & 0 & 0 \\ \frac{b^2 c}{a u_1^2} \frac{u_2 - u_3}{u_1 + c} & \frac{b^2}{a u_1} & \frac{b^2 c}{a u_1^2} \end{bmatrix}.$$

2. The Jacobian is

$$\frac{\partial \boldsymbol{f}(\mathbf{x})}{\partial \mathbf{x}} = \begin{bmatrix} 0 & 1 \\ -\kappa g \frac{x_d^\kappa}{x^{\kappa+1}} & -c \end{bmatrix}.$$

Hence,

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -\frac{\kappa g}{x_d} & -c \end{bmatrix}.$$

3. The limit of the partial derivatives

$$
\begin{aligned}
\frac{\partial}{\partial x}\left(\frac{x}{\ln\sqrt{x^2+y^2}}\right) &= \frac{1}{\ln\sqrt{x^2+y^2}} - \frac{1}{\ln\sqrt{x^2+y^2}}\left(\frac{x^2}{x^2+y^2}\right)\\
\frac{\partial}{\partial y}\left(\frac{x}{\ln\sqrt{x^2+y^2}}\right) &= -\frac{1}{\ln\sqrt{x^2+y^2}}\left(\frac{xy}{x^2+y^2}\right)\\
\frac{\partial}{\partial y}\left(\frac{y}{\ln\sqrt{x^2+y^2}}\right) &= \frac{1}{\ln\sqrt{x^2+y^2}} - \frac{1}{\ln\sqrt{x^2+y^2}}\left(\frac{y^2}{x^2+y^2}\right)\\
\frac{\partial}{\partial x}\left(\frac{y}{\ln\sqrt{x^2+y^2}}\right) &= -\frac{1}{\ln\sqrt{x^2+y^2}}\left(\frac{xy}{x^2+y^2}\right)
\end{aligned}
$$

when $\mathbf{x} \to 0$ is 0. This can be seen as a consequence of $\frac{r}{\log|r|} = o(r)$, where $r = \sqrt{x^2+y^2}$. In other words, the terms with the logarithm go to zero faster than any linear term.

Therefore, the system matrix of the linearized system at $[0,0]^T$ is

$$
A = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}.
$$

(d) Are the linearized systems stable, asymptotically stable or unstable? Justify your answers.

Solution: We study the eigenvalues of the system matrix $\mathbf{A}$ in each case:

1. The eigenvalues of $\mathbf{A}$ can be found on its diagonal. Since they are negative real numbers, the system is asymptotically stable. In particular, the linearized system is stable.
2. The matrix $\mathbf{A}$ is 2-by-2, has negative trace and positive determinant. Hence, the eigenvalues of $\mathbf{A}$ have negative real parts, and the system is asymptotically stable. In particular, the linearized system is stable.
3. The eigenvalues of $\mathbf{A}$ are $\pm i$. Hence, the system is only stable.

Problem 2 (Existence of solution)

Consider the following ODEs:

$$
\dot{x} = x^2, \quad x(0) = 1 \tag{5}
$$

$$
\dot{x} = \sqrt{|x|}, \quad x(0) = 0. \tag{6}
$$

For both ODEs, do the following:

(a) Simulate using the Matlab function `ode45` for a final time $t = 5$. What do you observe?

Solution: The simulation of the first ODE gives an error message. The step size has been reduced below the default tolerance.

The simulation of the second ODE gives a solution that is identically zero.

(b) Provide a formal explanation of the obtained results.

Solution: The first ODE is $\dot{x} = f(x)$ with $f(x) = x^2$. $f(x)$ is continuously differentiable everywhere with $f'(x) = 2x$. Hence, $f(x)$ is locally Lipschitz. However, $f(x)$ is not globally Lipschitz because for all $C > 0$ there are $x, y \in \mathbb{R}$ such that $|x^2 - y^2| > C|x - y|$ (e.g. $x = 2C$ and $y = 0$).

Therefore, the theory can only ensure that the ODE has an unique local solution given an initial condition, but not necessarily a global solution. In other words, the solution is defined on some interval around the initial time, but not necessarily at all time because $f(x)$ is not globally Lipschitz.
The solutions can be found using separation of variables:

$$\begin{aligned}
&\frac{dx}{dt} = x^2\\
\Rightarrow &\int_{x(t_0)}^{x(t)} \frac{dx}{x^2} = \int_{t_0}^{t} dt\\
\Rightarrow &-\frac{1}{x(t)} + \frac{1}{x(t_0)} = t - t_0\\
\Rightarrow &x(t) = \frac{x(t_0)}{1 - x(t_0)(t - t_0)}.
\end{aligned}$$

Note that these solutions are not defined for all time unless $x(t_0) = 0$, in which case $x(t) = 0$. This is in accordance with the theory since $f(x)$ is not globally Lipschitz.
In our case, the solution is $x(t) = \frac{1}{1-t}$. Hence, it only exists for $t \in [0, 1)$ and $\lim_{t \to 1^-} x(t) = +\infty$. As we will study, the `ode45` is an adaptive solver. In particular, this solver increases or decreases the size of the time step in order to keep a consistent approximation error. Since the actual solution diverges at $t = 1$, the solver will in principle make the time step arbitrarily small near $t = 1$. Therefore, this simulation is bound to fail for a final time larger or equal to 1.

For the second ODE, we have that $\dot{x} = f(x)$ with $f(x) = \sqrt{|x|}$. Since $f'(x) = \frac{1}{2\sqrt{|x|}}\text{sgn}(x)$ is well-defined for $x \neq 0$, we can only guarantee that this ODE has an unique local solution if the trajectory never passes through $x = 0$. In such case, separation of variables can be used to find the solution, which is

$$x(t) = \left(\frac{\text{sgn}(x_0)}{2}(t - t_0) + \sqrt{|x_0|}\right)^2 \text{sgn}(x_0).$$

As seen in the lectures, for a zero initial condition, there are infinite many solutions. One trivial solution is the identically zero, i.e. $x(t) = 0$, while the other solutions are a combination of the trivial and a parabola (as shown above):

$$x(t) = \begin{cases} 0, & t \in [t_0, t_1) \\ \frac{1}{4}(t - t_1)^2, & t \in [t_1, \infty). \end{cases}$$

The simulation using `ode45` gives the trivial solution since $f(x = 0) = 0$.

Problem 3 (Zombie apocalypse)

The doomsday is upon us!

Zombies have begun to rise from the dead, and violently and indiscriminately kill and infest the living. In order to save humanity, you have to model and simulate the zombie infestation using the little information available on these abominations.

There are 4 well-defined and non-overlapping populations:

- The healthy (H): Healthy people.
- The infected (I): People that have survived a zombie encounter, but have been infected.

- The zombies (Z).
- The dead (D): Recently deceased people and neutralized zombies.

The dynamics and interactions between these populations can be modeled like the classical predator-prey equations, i.e. the Lokta-Volterra equations, and they are given by the following laws:

- The birth-rate of the healthy is given by the parameter $b > 0$. In addition, the birth-rate is damped by a quadratic term with parameter b_d.
- The healthy can either become dead by "natural" causes (non-zombie interaction) with death-rate $d > 0$, or can become infected due to interactions with zombies with parameter i.
- The healthy that become infected, remain infected for some time and then become zombies. This is model as a first order system with rate $a > 0$.
- Infected individuals can still die by "natural" causes before becoming a zombie with the same death-rate as healthy individuals. In such case, they become dead; otherwise, they become a zombie.
- Zombies rise from the dead with rate $r > 0$.
- Some zombies that interact with healthy individuals are neutralized with parameter $n > 0$. This zombies become dead, and may rise again in the future.

(a) Model the dynamics of the populations H, I, Z and D.

Hint: The expressions for $\dot{H}$, $\dot{I}$, $\dot{Z}$ and $\dot{D}$ are given by polynomials in H, I, Z and D. Furthermore, $\dot{H} + \dot{I} + \dot{Z} + \dot{D}$ is equal to the total birth-rate of the healthy, $bH - b_d H^2$.

Solution:

$$\begin{aligned}
\dot{H} &= (b - d)H - b_d H^2 - iHZ \\
\dot{I} &= -(a + d)I + iHZ \\
\dot{Z} &= aI + rD - nHZ \\
\dot{D} &= d(H + I) - rD + nHZ
\end{aligned}$$

and $H, I, Z, D \geq 0$.

The situation looks grim: All simulations confirm that the outbreak of zombies will lead to $H \to 0$, i.e. the collapse of civilization.

In order to contain the outbreak, the authorities ask you to model the effects of a partial quarantine of zombies and infected individuals. Quarantined individuals are removed from their original populations and are placed on an special area, where they no longer can infect healthy individuals.

Consider the new population of quarantined individuals (Q), and add the following changes to the previous model:

- Infect individuals and zombies are quarantined with rates q_i and q_z, respectively.
- Some quarantined individuals die from "natural" causes, or are killed when they try to escape. In such case, they are moved to the dead population. This happens with rate d_q.
- The quarantined individuals that are moved to the dead population, may still rise as "free roaming" zombies.

(b) Model the dynamics of the populations H, I, Z, D and Q.

Solution:

$$\begin{aligned}
\dot{H} &= (b-d)H - b_d H^2 - iHZ \\
\dot{I} &= -(a+d+q_i)I + iHZ \\
\dot{Z} &= aI + rD - nHZ - q_z Z \\
\dot{D} &= d(H+I) - rD + nHZ + d_q Q \\
\dot{Q} &= q_i I + q_z Z - d_q Q
\end{aligned}$$

and $H, I, Z, D, Q \geq 0$.

(c) Simulate the models with and without quarantine for 100 days. Use the Matlab function `ode45` or another in-built solver of your choosing. Moreover, use the following parameter values and initial conditions:

Parameter	Value [s^{-1}]	Population	Initial value
a	$1.4 \cdot 10^{-6}$	H	$\frac{b-d}{b_d}$
b	$3.1 \cdot 10^{-8}$	I	0
b_d	$5.6 \cdot 10^{-16}$	Z	0
d	$2.8 \cdot 10^{-8}$	D	0
i	$2.6 \cdot 10^{-6}$	Q	0
n	$1.4 \cdot 10^{-6}$		
r	$2.8 \cdot 10^{-7}$		
q_i	$2.7 \cdot 10^{-6}$		
q_z	$2.7 \cdot 10^{-6}$		
d_q	$2.8 \cdot 10^{-5}$		

Table 1: Parameter and initial values.

Add a plot with the obtained results for all populations to your answer.
Comment on the results.
Hint: The time frame of these simulations and the state values are relatively large. Therefore, using the solver directly will not be feasible due to large computational time and lack of memory. Read the documentation of `ode45` and `odeset` to find a way to circumvent these issues.

Solution: Since the populations are in the millions, it is reasonable to increase the default absolute tolerance (e.g. to 1). Similarly, we increase the default initial time step (e.g. to 1 day). Finally, in order to avoid running out of memory, we can ask the solver for the values at specific points (e.g. 1000 points along the time frame).

Example code:

```
% parameters
a = 1.4e-6;
b = 3.1e-8;
b_d = 5.6e-16;
d = 2.8e-8;
i = 2.6e-6;
n = 1.4e-6;
r = 2.8e-7;
q_i = 2.7e-6;
q_z = 2.7e-6;
```

```
d_q = 2.8e-5;
H_0 = (b-d)/b_d;
% simulate
hour = 60*60;
day = 24*hour;
T = linspace(0,100*day,1000);
options = odeset('RelTol',1e-6,'AbsTol',1,'InitialStep',day);
[t_1,x_1] = ode45(@(t,x) f1(t,x,a,b,b_d,d,i,n,r), T, [H_0,0,0,0]',
    options);
[t_2,x_2] = ode45(@(t,x) f2(t,x,a,b,b_d,d,i,n,r,q_i,q_z,d_q), T, [H_0
    ,0,0,0,0]',options);

%...

%vector fields
function dxdt = f1(t,x,a,b,b_d,d,i,n,r)
    % x = [H;I;Z;D]
    dxdt = [(b-d)*x(1) - b_d*x(1)^2 - i*x(1)*x(3);
        -(a+d)*x(2) + i*x(1)*x(3);
        a*x(2) + r*x(4) - n*x(1)*x(3);
        d*(x(1)+x(2)) - r*x(4) + n*x(1)*x(3)];
end
function dxdt = f2(t,x,a,b,b_d,d,i,n,r,q_i,q_z,d_q)
    % x = [H;I;Z;D;Q]
    dxdt = [(b-d)*x(1) - b_d*x(1)^2 - i*x(1)*x(3);
        -(a+d+q_i)*x(2) + i*x(1)*x(3);
        a*x(2) + r*x(4) - n*x(1)*x(3) - q_z*x(3);
        d*(x(1)+x(2)) - r*x(4) + n*x(1)*x(3) + d_q*x(5);
        q_i*x(2) + q_z*x(3) - d_q*x(5)];
end
```

Results:

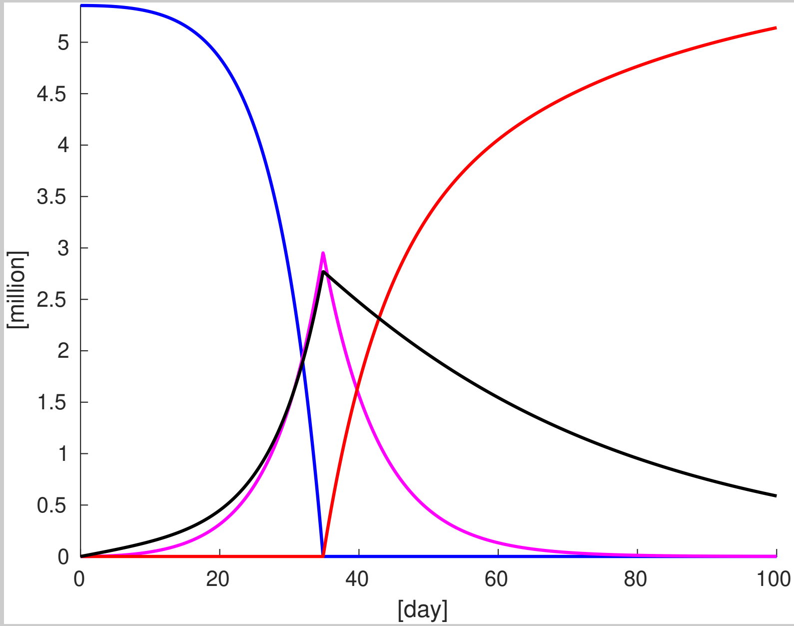

Figure 1: Simulation without quarantine. H (blue), I (magenta), Z (red), D (black).

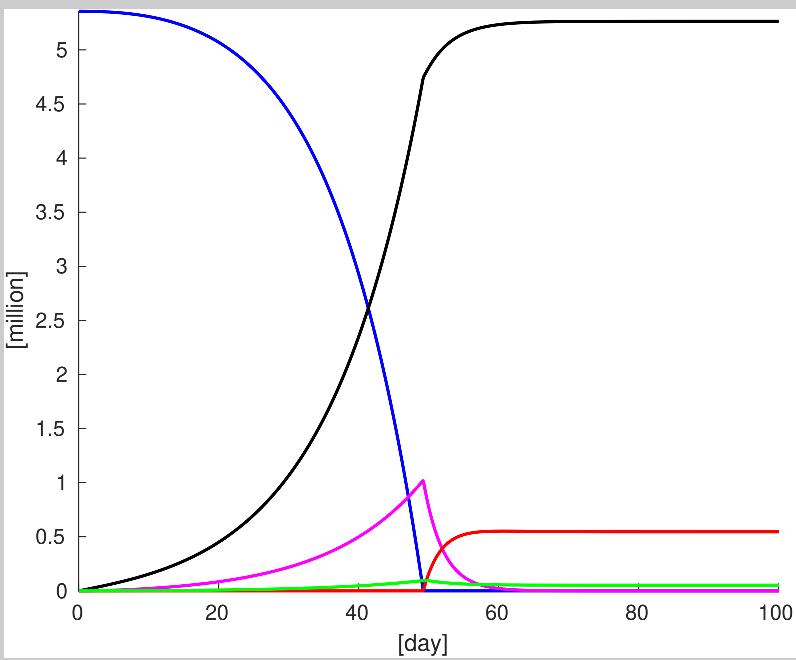

Figure 2: Simulation with quarantine. H (blue), I (magenta), Z (red), D (black), Q (green).

The quarantined approach only seems to delay the inevitable. We are doomed!