Student names: ... (please update)

Instructions: Update this file (or recreate a similar one, e.g. in Word) to prepare your answers to the questions. Feel free to add text, equations and figures as needed. Hand-written notes, e.g. for the development of equations, can also be included e.g. as pictures (from your cell phone or from a scanner). ***This lab is not graded. However, the lab exercises are meant as a way to familiarise with dynamical systems and to study them using Python to prepare you for the final project.*** *This file does not need to be submitted and is provided for your own benefit. The graded exercises will have a similar format.*

The file `lab#.py` *is provided to run all exercises in Python. Each* `exercise#.py` *can be run to run an exercise individually. The list of exercises and their dependencies are shown in Figure 1. When a file is run, message logs will be printed to indicate information such as what is currently being run and and what is left to be implemented. All warning messages are only present to guide you in the implementation, and can be deleted whenever the corresponding code has been implemented correctly.*

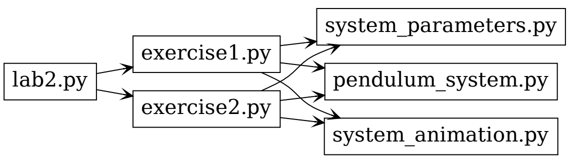

Figure 1: Exercise files dependencies. In this lab, you will be modifying `exercise1.py`, `exercise2.py` *and* `pendulum_system.py`.

In this exercise, you will explore the different modeling techniques that can be used to control a single joint and segment. We initially start by exploring a single joint controlled by a a single simplified pendulum model with damping(friction) (exercise1) and then extend it to pair of spring-dampers muscle models (exercise2). These only represent the passive dynamics observed in a real musculoskeletal system. You are provided with a code that can simulate a pair spring-damp muscle model .

Important note: *Both exercise use a generic class that can handle both a pair of spring damp muscles, or a single spring damp muscle sketched in Figure 2. Each muscle pair contains sping constants and resting angles, and damping coefficients. Simply set all the values of these parameters equal for the two pairs to study the behavior of a single spring-damper instead of a pair (in Exercise 1). Have a look at the specification of parameters of the pendulum system in the class PendulumParameters in* `system_parameters.py`.

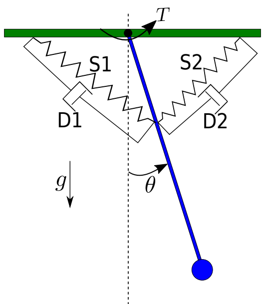

Figure 2: Pendulum model with two springs S1 and S2 and two dampers b1 and b2
T - Positive torque direction.
g - Gravity.
θ - Angle made by the pendulum

Question 1: Pendulum with friction

1.a Find the fixed points of the pendulum with friction (i.e. damping, and analyze their stability using a local linearization under no external input $T_{ext} = 0$) expressed in the following equation (briefly describe the calculation steps).

$$I\ddot{\theta} = -mgLsin(\theta) + T_{ext} - b\dot{\theta} \tag{1}$$

Considering Inertia $I = mL^2$, the equation of the pendulum can be written as,

$$\ddot{\theta} = -g\frac{sin(\theta)}{L} + \frac{T_{ext}}{I} - b\frac{\dot{\theta}}{I} \tag{2}$$

where θ is the angle, g the gravity constant, L the length of the pendulum and b is the damping coefficient.

1.b Implement the the damping equation of the pendulum using equations described above in the function `pendulum_system.py::pendulum_equation`. Numerically solve the differential equations of the pendulum with different initial conditions. Show several time evolutions and phase portraits with different initial conditions that illustrate several aspects of the interesting behavior of the pendulum. Additionally, implement the damping parameter and demonstrate examples of underdampded, critically damped and overdamped behaviors. See `exercise1.py` and `system_parameters.py` and `pendulum_system.py` for help with implementation.

1.c Investigate and describe how the behavior of the pendulum changes if friction is zero (b=0). Show a new phase portrait.

1.d Does the pendulum without friction (b=0) produce stable limit cycles? Discuss, and try to support your statement with some numerical simulations (show figures) and/or analytical arguments.

1.e Investigate how the behavior of the pendulum changes if the viscous friction term is replaced with a dry (Coulomb) friction term. Unlike viscous friction, dry friction does not depend on speed, only the direction of movement. What are the main differences between the two types of pendulum? (discuss and show some examples). And is there anything notable about the numerical integration of the pendulum with dry friction? If yes, what and why?

$$\ddot{\theta} = -\frac{g}{L}\sin\theta - b \cdot sign(\dot{\theta}) \tag{3}$$

Exercise 2 : Pendulum model with passive elements

Mechanical behavior of muscle tissue can be approximated by simple passive elements such as springs and dampers. These elements, when combined properly, allow to study the behavior of muscle under compressive and tensile loads.

Consider the following equation describing the motion of simple pendulum with an external torque T_{ext},

$$I\ddot{\theta} = -mgLsin(\theta) + T_{ext} \tag{4}$$

Consider the system only for the pendulum range $\theta = [-\pi/2, \pi/2]$

Explore the pendulum model with two antagonist spring elements

In this question the goal is to add two antagonist springs to the pendulum model which you are already familiar with from lab 2 exercises. For simplicity we assume the springs directly apply a torsional force on to the pendulum. Use equation 5 to develop the spring model.

***Note** : The springs can only produce force in one-direction like the muscles. That is, they can only apply a pulling force and apply a zero force when compressed. In terms of torsion this translates to, spring S1 can exert only clockwise torque and spring S2 can exert only counter-clockwise torque. You need to accommodate for this condition in the equations shown below.*

The setup for the pendulum with a pair of antagonist springs is as shown in figure 3. Use `exercise2.py`, `pendulum_system.py` and `system_parameters.py` files to complete the exercise.

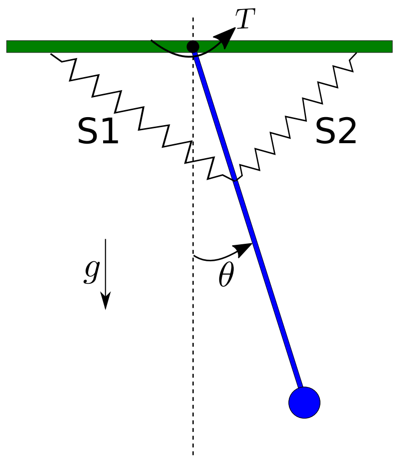

Figure 3: Pendulum model with two springs S1 and S2.
T - Positive torque direction.
g - Gravity.
θ - Angle made by the pendulum

$$T_S = k \cdot (\theta_{ref} - \theta) \tag{5}$$

Where,

- T_S : Torsional Spring force
- k : Spring Constant
- θ_{ref} : Spring reference angle
- θ : pendulum angle

Substituting the above in 2,

$$\ddot{\theta} = -g\frac{sin(\theta)}{L} + \frac{T_{ext}}{I} + \frac{T_S}{I} \tag{6}$$

$$\ddot{\theta} = -g\frac{sin(\theta)}{L} + \frac{T_{ext}}{I} + \frac{k \cdot (\theta_{ref} - \theta)}{I} \tag{7}$$

Use the generalized form of the spring equation described in 7 to extend it to both the antagonist springs S1 and S2 with the necessary conditions to make sure springs do not produce when compressed.

2.a Implement the dynamic equations of the pendulum with springs using equations described above in the function `pendulum_system.py::pendulum_equation`. Does the system have a stable limit cycle behavior? Describe and run an experiment to support your answer. You can use the function `exercise2.py::pendulum_perturbation` to perturb the pendulum either by changing states or applying an external torque. Use the class `system_animation.py::SystemAnimation` to visualize the pendulum. Example code can be found in `exercise2.py::exercise2`

2.b Explore the role of spring constant (k) and spring reference angle (θ_{ref}) in terms of range of motion, amplitude and frequency of pendulum. Keep the constants equal, i.e $k_1 = k_2$ and $\theta_{ref1} = \theta_{ref2}$
Refer to `exercise2.py::exercise1` for an example

2.c Explain the behavior of the model when you have asymmetric spring constants (k) and spring reference angles (θ_{ref}), i.e. $k_1 \neq k_2$ and $\theta_{ref1} \neq \theta_{ref2}$ Support your responses with relevant plots

Explore the pendulum model with two antagonist spring and damper elements

Over time muscles lose energy while doing work. In order to account for this property, let us now add a damper in parallel to the spring model. Use equation 8 to develop the damper model.

***Note** : Like the previous springs, the springs in spring-dampers can only produce a force in one-direction. However, the damper terms do not have this limitation and each damper can exert a force in both directions.*

Again use `exercise2.py`, `pendulum_system.py` and `system_parameters.py` files to complete the exercise. The setup for the pendulum model with a pair of antagonist spring and dampers in parallel is as shown in figure 4.

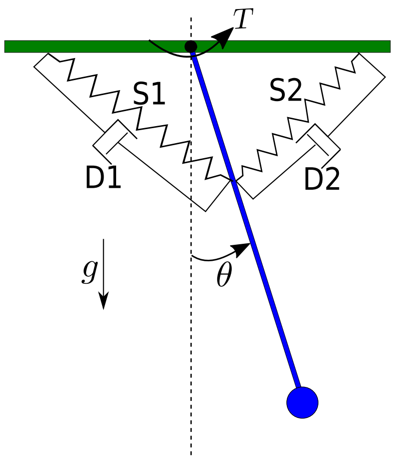

Figure 4: Pendulum model with two springs S1 and S2 and two dampers b1 and b2
T - Positive torque direction.
g - Gravity.
θ - Angle made by the pendulum

$$T_B = b \cdot \dot{\theta} \tag{8}$$

Where,

- T_B : Torsional Damper force

- b : Damping Constant
- $\dot{\theta}$: pendulum angular velocity

The combined spring damper torque is given by,

$$T_S - T_B = k \cdot (\theta_{ref} - \theta) - b \cdot \dot{\theta} \tag{9}$$

The minus for the damper comes from the fact that damper is acting against the work done by the spring.

Substituting the above in 2

$$\ddot{\theta} = -g\frac{sin(\theta)}{L} + \frac{T_{ext}}{I} + \frac{T_S - T_B}{I} \tag{10}$$

$$\ddot{\theta} = -g\frac{sin(\theta)}{L} + \frac{T_{ext}}{I} + (\frac{k \cdot (\theta_{ref} - \theta) - b \cdot \dot{\theta}}{I}) \tag{11}$$

Use the generalized form of the spring equation described in 11 to extend it to both the antagonist spring-damper systems (S1-D1) and (S2-D2).

2.d Implement the dynamics equations of the pendulum to now include the damping using the equations described above. Modify `pendulum_system.py::pendulum_equation`. How does the behavior now change compared to the pendulum without dampers? Briefly explain and support your responses with relevant plots

2.e Can you find a combination of spring constants (k), damping constants (b) and spring reference angles (θ_{ref}) that makes the pendulum rest in a stable equilibrium at ($\theta = \pi/6$) radians? Describe how you arrive at the necessary parameters and support your response with relevant plots.

2.f What is the missing component between a real muscle and the muscle model with passive components that you just explored? What behavior's do you lack because of this missing component?