Hanze University of Applied Sciences
Institute of Engineering
Smart Systems Laboratory

Lab Session 1 - Creating a Virtual Model of a Simple Mechanical System

1 | Aim

The aim of this lab is to develop a virtual model of a pendulum and motor system using Python.

2 | Preparation

For this experiment, as material, you only need a laptop with Python installed with the necessary dependencies. You can find them listed in the `requirements.txt` file. (You may use `pip install -r requirements.txt`).
The following two files are necessary for today (place them in the same folder):

- `Digital_twin.py`: important file, used throughout all the labs; and
- `Lab_1_simulation.py`.

Before getting started, watch the following video: http://tinyurl.com/525buknk.

3 | Procedure

3.1. Today, the goal is to create a model that represents the angular acceleration of the pendulum. For the first part, you should implement the following function. This function should return the angular acceleration ($\frac{d^2\theta}{dt^2} = \ddot{\theta}$), also denoted here as `theta_double_dot`. There is already a function implemented for the motor response model in

```
def update_motor_accelerations(self, direction, duration)
```

3.2. *Bonus*: Improve this model to better align with the actual motor response.

3.3. Keep in mind that the sample rate of the virtual model should match the sample rate of the sensor (Lab 2).

3.4. Once you have implemented the function `get_theta_double_dot(self, theta, theta_dot)`, you can run the `Lab_1_simulation.py` file.

3.5. Try to swing up the pendulum using the keys on your keyboard and create a sequence of 10 actions that successfully swings up the pendulum.

Algorithm 1: Function to calculate angular acceleration.

```
def get_theta_double_dot(self, theta, theta_dot):
    """
    Lab 1: Model the angular acceleration (theta_double_dot)
    as a function of theta, theta_dot and the self.currentmotor_acceleration.
    You should include the following constants as well: c_air, c_c, a_m, l,
        and g.
    """
    # Implement your model here.
```

4 | Tasks

4.1. Model the Pendulum:

- Include air friction, mechanical friction, pendulum length, and gravitational effects in the model;
- State space parameters should encompass the angle and angular velocity of the pendulum. Try to set the constants such that the behavior is similar to the real pendulum.

4.2. Model the Motor:

- Explain how the motor acceleration relates to the pendulum model. Read the paper [1] and represent the motor in terms of its acceleration effect on the pendulum;
- Ensure the model can simulate the dynamic interaction between the motor and pendulum.

5 | References

[1] V. H. Pinto, J. Gonçalves, and P. Costa, “Modeling and Control of a DC Motor Coupled to a Non-Rigid Joint,” *Applied System Innovation*, vol. 3, no. 2, p. 24, 2020. [Online]. Available: https://www.mdpi.com/2571-5577/3/2/24