Nonlinear System Analysis and Control
Lab 1: Blending Control

Date: 1 December 2020

1 Introduction

This lab's objective is to learn to apply specific tools to analyze and control a nonlinear system, specifically to a blender system.

2 System modeling

Consider a blender system illustrated as:

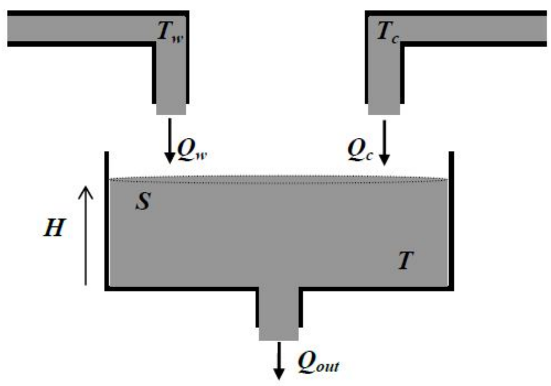

Figure 1: Illustration of the blender system.

The dynamics of the water level and temperature is characterized by the following equations:

$$\begin{cases} S \cdot \frac{d(H(t))}{dt} = Q_w(t) + Q_c(t) - Q_{out}(t) \\ S \cdot \frac{d(H(t) \cdot T(t))}{dt} = T_w \cdot Q_w(t) + T_c \cdot Q_c(t) - T(t) \cdot Q_{out}(t) \end{cases} \tag{1}$$

where the first equation corresponds to the inflow-outflow volume balance and the second one corresponds to the inflow-outflow entropy balance. The outward flow rate is directly proportional to the square root of the current water level, i.e., $Q_{out}(t) = a \cdot \sqrt{H(t)} = a \cdot \sqrt{H_0 + h(t)}$. If we expand the left-hand side of the second equation into $\frac{d(H(t) \cdot T(t))}{dt} = T(t) \cdot \frac{d(H(t)}{dt} + H(t) \cdot \frac{d(T(t))}{dt}$ and substitute $\frac{d(H(t))}{dt}$ from the first one, the system simplifies itself to the following nonlinear system:

$$\Sigma_n : \begin{cases} \dot{H}(t) = -\frac{a}{S} \cdot \sqrt{H(t)} + \frac{1}{S} \cdot (Q_w(t) + Q_c(t)) \\ \dot{T}(t) = \frac{T_w - T(t)}{S \cdot H(t)} \cdot Q_w(t) + \frac{T_c - T(t)}{S \cdot H(t)} \cdot Q_c(t) \end{cases} \tag{2}$$

Denoting $h(t) = H(t) - H_0$, $q_w(t) = Q_w(t) - Q_0$, $q_c(t) = Q_c(t) - Q_0$, and $\theta(t) = T(t) - T_0$, and noting that T_0 is the equilibrium temperature, i.e., $T_w + T_c - 2 \cdot T_0 = 0$, we have:

$$\Sigma_n : \begin{cases} \dot{h}(t) = \frac{1}{S} \cdot (-a \cdot \sqrt{H_0 + h(t)} + 2 \cdot Q_0) + \frac{1}{S} \cdot (q_w + q_c) \\ \dot{\theta}(t) = -\frac{2 \cdot Q_0}{S \cdot (H_0 + h(t))} \cdot \theta(t) + \frac{T_w - T_0 - \theta(t)}{S \cdot (H_0 + h(t))} \cdot q_w + \frac{T_c - T_0 - \theta(t)}{S \cdot (H_0 + h(t))} \cdot q_c \end{cases} \tag{3}$$

We have the equations governing the evolution of h and θ under q_w and q_c.
The values of the parameters are: $k_v = 1.2 cm^3/s/\%$, $a = 24 cm^{5/2}/s$, $T_w = 70°C$, $T_c = 20°C$, $S = \pi \times 5^2 cm^2$, $H_0 = 16cm$, $T_0 = 45°C$, and $Q_0 = 48 cm^3/s$.

3 Linear control

3.1 Linearized system

Linearization around the equilibrium using the Jacobian method gives:

$$\Sigma_l : \begin{bmatrix} \dot{h}(t) \\ \dot{\theta}(t) \end{bmatrix} = \begin{bmatrix} -\frac{a}{2 \cdot S \cdot \sqrt{H_0}} & 0 \\ 0 & -\frac{2 \cdot Q_0}{S \cdot H_0} \end{bmatrix} \cdot \begin{bmatrix} h(t) \\ \theta(t) \end{bmatrix} + \begin{bmatrix} \frac{1}{S} & \frac{1}{S} \\ \frac{T_w - T_0}{S \cdot H_0} & \frac{T_w - T_0}{S \cdot H_0} \end{bmatrix} \cdot \begin{bmatrix} q_w \\ q_c \end{bmatrix} \tag{4}$$

Introducing the two new variables $v_h = \frac{q_w + q_c}{S}$ and $v_\theta = \frac{(T_w - T_0)\cdot q_w + (T_c - T_0)\cdot q_c}{S \cdot H_0}$, we have the following linear system:

$$\Sigma_l : \begin{bmatrix} \dot{h}(t) \\ \dot{\theta}(t) \end{bmatrix} = \begin{bmatrix} -\frac{a}{2\cdot S\cdot\sqrt{H_0}} & 0 \\ 0 & -\frac{2\cdot Q_0}{S\cdot H_0} \end{bmatrix} \cdot \begin{bmatrix} h(t) \\ \theta(t) \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} v_h \\ v_\theta \end{bmatrix} =: \begin{bmatrix} A_h & 0 \\ 0 & A_\theta \end{bmatrix} \cdot \begin{bmatrix} h(t) \\ \theta(t) \end{bmatrix} + \begin{bmatrix} B_h & 0 \\ 0 & B_\theta \end{bmatrix} \cdot \begin{bmatrix} v_h \\ v_\theta \end{bmatrix} \tag{5}$$

which corresponds to two separate first-order linear systems. We see that each of h and θ can be directly controlled by v_h and v_θ.

3.2 Controller design

We would like to track a reference with a desired settling time. To do so, the controllers are:

$$\begin{cases} v_h = -k_h \cdot h + g_h \cdot r_h \\ v_\theta = -k_\theta \cdot \theta + g_\theta \cdot r_\theta \end{cases} \tag{6}$$

where r_h and r_θ are the respective references. The gains k_h, g_h, k_θ, and g_θ are to be determined so as to satisfy the specifications. If we want to speed up the system, i.e., decrease the settling time by moving the poles further from the origin, by four times for example, then $k_h = -3 \cdot \frac{a}{2\cdot S\cdot\sqrt{H_0}}$, and vice versa, and similar for the control of θ. To track a reference, we need to have static unit gains, which means:

$$\begin{cases} g_h = [1 \cdot (-A_h + B_h \cdot k_h)^{-1} \cdot B_h]^{-1} \\ g_\theta = [1 \cdot (-A_\theta + B_\theta \cdot k_\theta)^{-1} \cdot B_\theta]^{-1} \end{cases} \tag{7}$$

When we simulate the controlled linear systems uisng MATLAB, we obtain coherent results. We see that the controlled system has become four times faster/slower and its tracking performance is perfect (we track a unit reference).

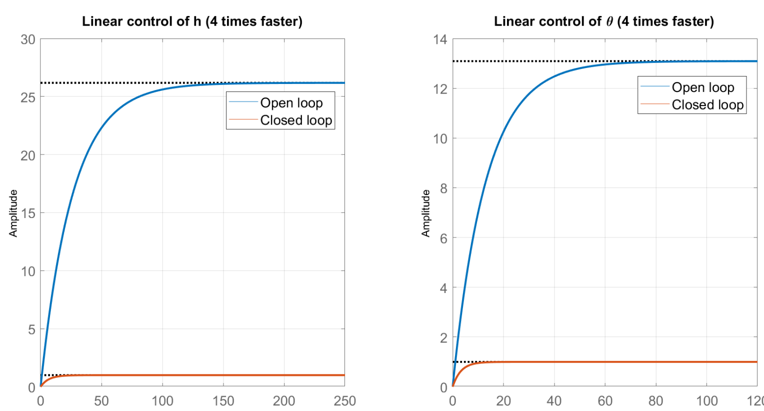

Figure 2: Linear controllers of h and θ to make them 4 times faster.

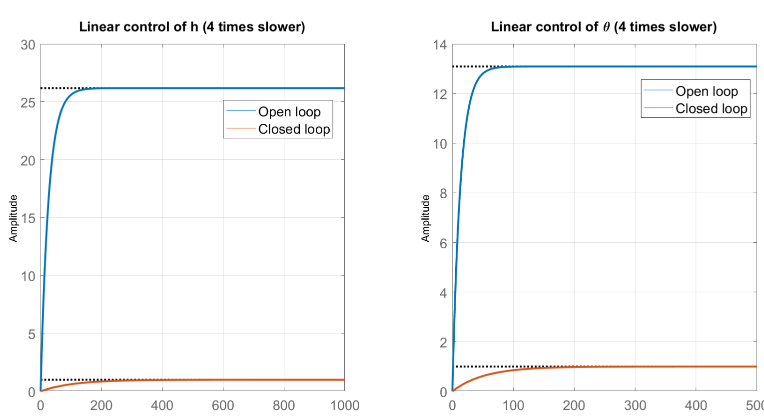

Figure 3: Linear controllers of h and θ to make them 4 times slower.

3.3 Simulation results

We build the following Simulink model to simulate the system. There is a green block to switch from v_h and v_θ to q_w and q_c, which are the systems' input. We want to compare the Jacobian linearized model (yellow) and the nonlinear one (red), so the same control inputs were given to both systems.

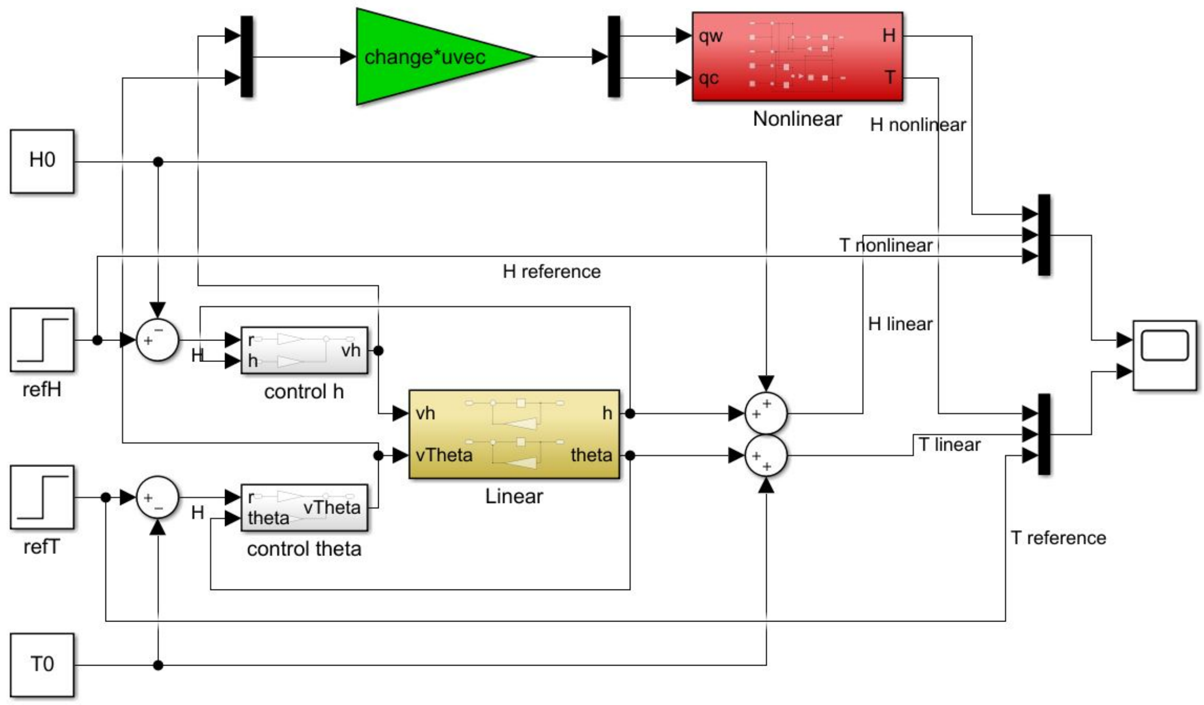

Figure 4: Simulation of the linear control approach.

Now we speed up the system first. When we have a small reference, both systems have quite similar performances. However, only the linear system can track the reference because the linear controller is designed for it. For the nonlinear system, it is slower and has a non-zero steady-state error.

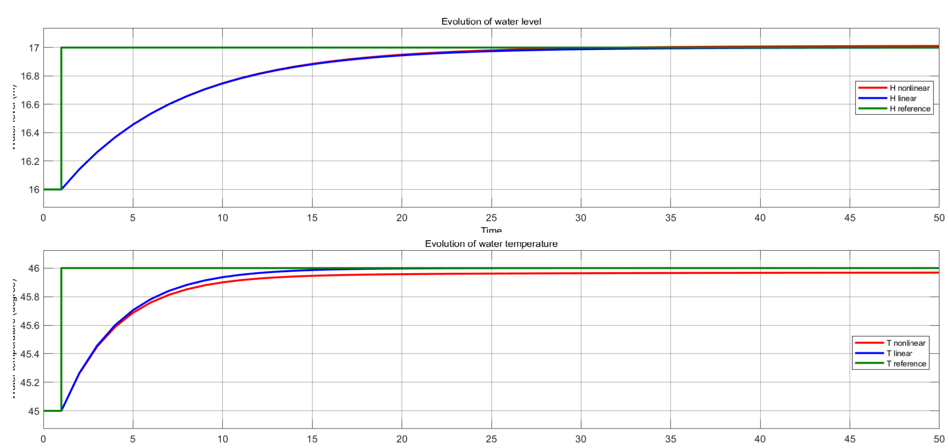

Figure 5: Linear control to fasten four times both linear and nonlinear systems with SMALL references.

When we increase the reference, we see that the error has become even worse as there are large steady-state errors. The linear controller is designed for the linear system, and when we move far away from the equilibrium point, the linearized one does not match with the nonlinear one anymore, i.e., the controller is no longer compatible with the controlled system.

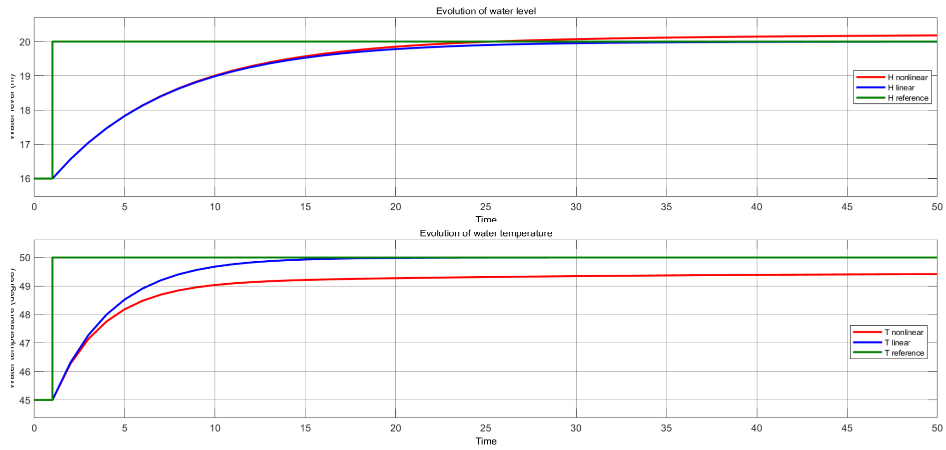

Figure 6: Linear control to fasten 4 times both linear and nonlinear systems with LARGE references.

The same happens to the case where we slow down the system four times, the error is small if the reference is small.

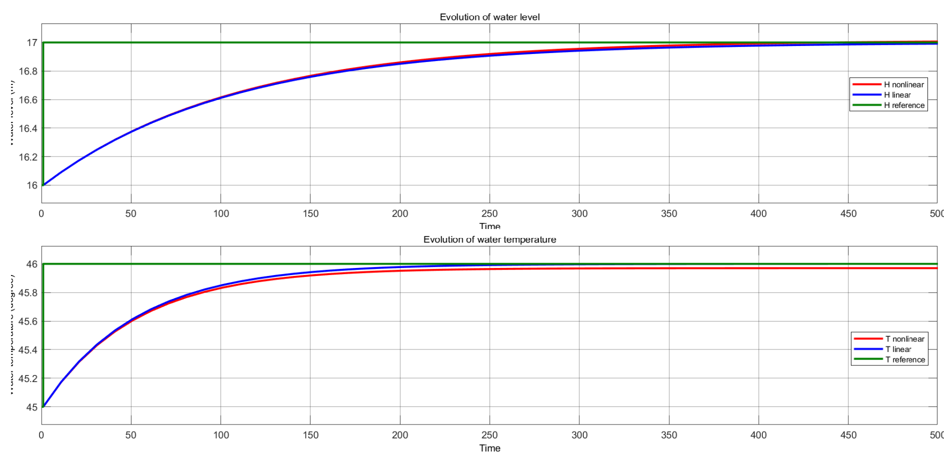

Figure 7: Linear control to slow down 4 times both linear and nonlinear systems with SMALL references.

And the performance gets worse as the reference increases.

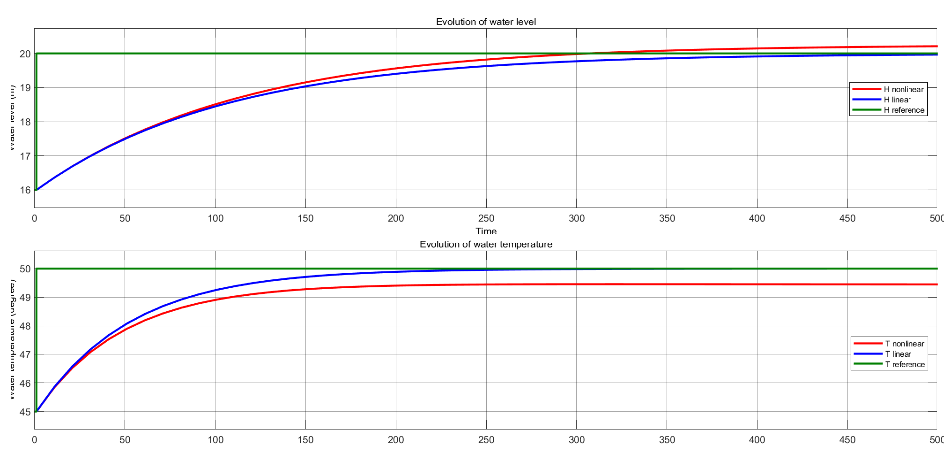

Figure 8: Linear control to slow down 4 times both linear and nonlinear systems with LARGE references.

The settling time for the temperature is around 100s for the open-loop system. If we speed it up four times the result will be 25s and if we slow it down 4 times we will have 400s, which is coherent. After these experiments, we see that applying a linear controller designed according to a Jacobian linearized system on the nonlinear system may lead to bad performances. We need nonlinear approaches to solve these problems.

4 Nonlinear control

4.1 Controller design

With $v_h = \frac{q_w+q_c}{S}$ and $v_\theta = \frac{(T_w-T_0-\theta)\cdot q_w+(T_c-T_0-\theta)\cdot q_c}{S\cdot H_0}$, we re-write the nonlinear system as:

$$\Sigma_n : \begin{cases} \dot{h}(t) = \frac{1}{S}\cdot(-a\cdot\sqrt{H_0+h(t)}+2\cdot Q_0)+v_h \\ \dot{\theta}(t) = -\frac{2\cdot Q_0}{S\cdot(H_0+h(t))}\cdot\theta(t)+v_\theta \end{cases} \tag{8}$$

We treat the nonlinearity by applying two-part control inputs:

$$\begin{cases} v_h = \overline{v}_h + v_{h,linear} = \overline{v}_h - k_h\cdot h + g_h\cdot r_h \\ v_\theta = \overline{v}_\theta + v_{\theta,linear} = \overline{v}_\theta - k_\theta\cdot\theta + g_\theta\cdot r_\theta \end{cases} \tag{9}$$

where the first part is to compensate for the nonlinearity, i.e., $\overline{v}_h = -\frac{1}{S}\cdot(-a\cdot\sqrt{H_0+h(t)}+2\cdot Q_0)$ and $\overline{v}_\theta = \frac{2\cdot Q_0}{S\cdot(H_0+h(t))}\cdot\theta(t)$. The system is reduced to:

$$\Sigma_n : \begin{cases} \dot{h}(t) = -k_h\cdot h + g_h\cdot r_h \\ \dot{\theta}(t) = -k_\theta\cdot\theta + g_\theta\cdot r_\theta \end{cases} \tag{10}$$

and our job is to the design the linear controllers for these linear systems. This is not Jacobian linearized, but instead we use direct compensation to linearize the system. Note that the matrices A, B of the systems have changed, i.e., now $A_h = -k_h$, etc...

4.2 Simulation results

We build the following Simulink model to simulate the system. The yellow block to switch from v_h and v_θ to q_w and q_c which are the input of the systems has been modified. In the last case, as everything was constant, that block was simply a matrix inversion. However, now we need to implement a nonlinear function to solve back the real control inputs:

$$\begin{cases} q_w = \frac{(T(t)-T_c)\cdot S\cdot v_h + H(t)\cdot S\cdot v_\theta}{T_w-T_c} \\ q_c = S\cdot v_h - q_w \end{cases} \tag{11}$$

The controllers are in two parts: linearization term (green block) and control term (blue block).

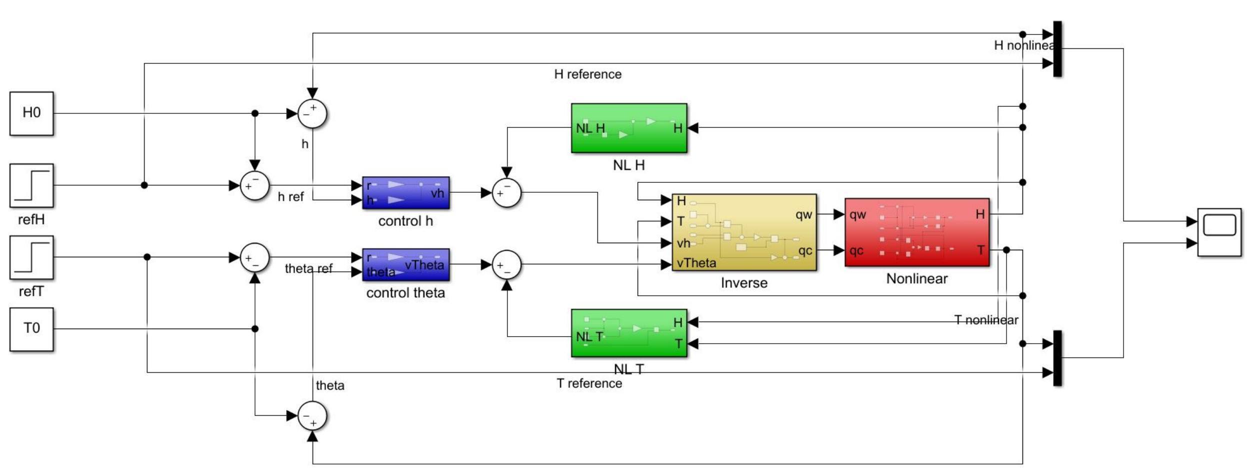

Figure 9: Simulation of the nonlinear control approach.

The results are much better compared to above. Here we speed up the system by four times.

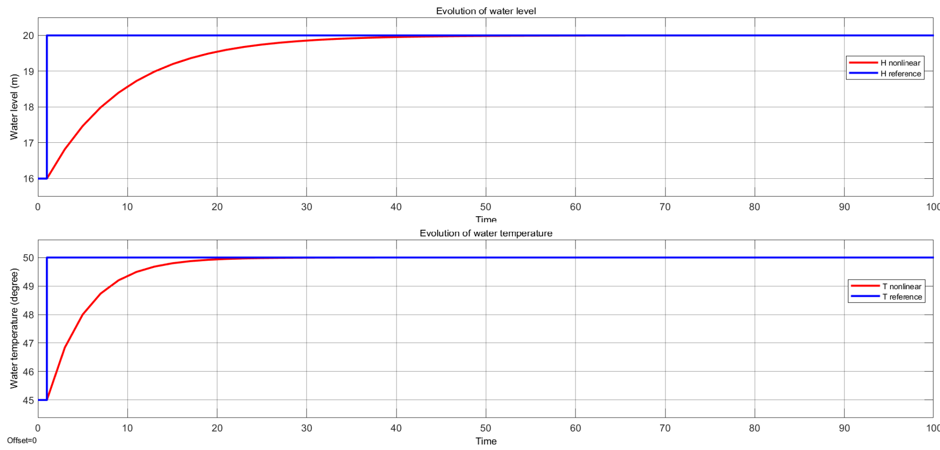

Figure 10: Nonlinear control simulation results.

For a nonlinear system, we need to have nonlinear control so as to guarantee the performance.

5 Extensions

We have seen that the nonlinear approach gives better performance compared to the linear one. We propose some extensions to this.

5.1 Robustness analysis

In this part, we would like to test the robustness of the nonlinear approach w.r.t system modeling uncertainty, as the compensation part depends heavily on the knowledge of the system parameters and states. In reality, the parameters we use to compute the control inputs cannot be the same as those of the real system. Therefore, we implement some error in system modeling. For example, when the real $Q_0 = 48cm^3/s$ but the value we use when we compensate the nonlinearities in $H(t)$ and $T(t)$ is $49cm^3/s$, we have the following results:

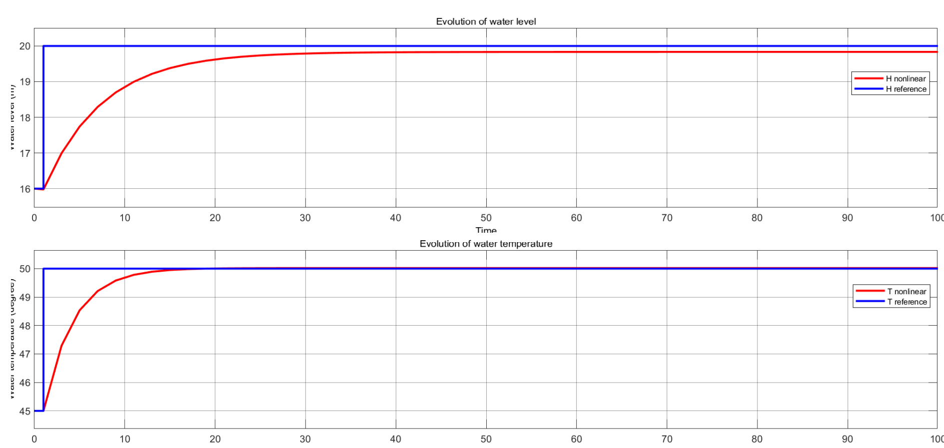

Figure 11: Robustness analysis of the nonlinear control approach.

For both outputs, we have a non-zero steady-state error (the one for $T(t)$ is significantly smaller) because we are not fully compensating for the system's nonlinearity. We control the linear system perfectly, but the nonlinearity is still there.

5.2 Disturbance rejection

In this part, we would like to add the ability to reject disturbances to the controlled system through the use of integral action. Let us denote the integral parts:

$$\begin{cases} z_h(t) = \int_0^t (r_h(\tau) - h(\tau))\, d\tau \\ z_\theta(t) = \int_0^t (r_\theta(\tau) - \theta(\tau))\, d\tau \end{cases} \tag{12}$$

The extended systems are:

$$\begin{cases} \Sigma_h : \begin{cases} \begin{bmatrix} \dot{h} \\ \dot{z}_h \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ -1 & 0 \end{bmatrix} \cdot \begin{bmatrix} h \\ z_h \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \cdot v_{h,linear} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \cdot r_h + \begin{bmatrix} E_h \\ 0 \end{bmatrix} \cdot d_h \\ y_h = \begin{bmatrix} 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} h \\ z_h \end{bmatrix} \end{cases} \\ \Sigma_\theta : \begin{cases} \begin{bmatrix} \dot{\theta} \\ \dot{z}_\theta \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ -1 & 0 \end{bmatrix} \cdot \begin{bmatrix} \theta \\ z_\theta \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \cdot v_{\theta,linear} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \cdot r_\theta + \begin{bmatrix} E_\theta \\ 0 \end{bmatrix} \cdot d_\theta \\ y_\theta = \begin{bmatrix} 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} \theta \\ z_\theta \end{bmatrix} \end{cases} \end{cases} \tag{13}$$

where E_h and E_θ are the gains of the disturbances d_h and d_θ, respectively. The linear–quadratic regulator (LQR) method is used to obtain the gains of the controllers, where equal weights were given to the state and the integral part, as we consider them to be equally important. The simulation results are as follows, where we have a step input disturbance on q_w at 5s and another one on q_c at 10s:

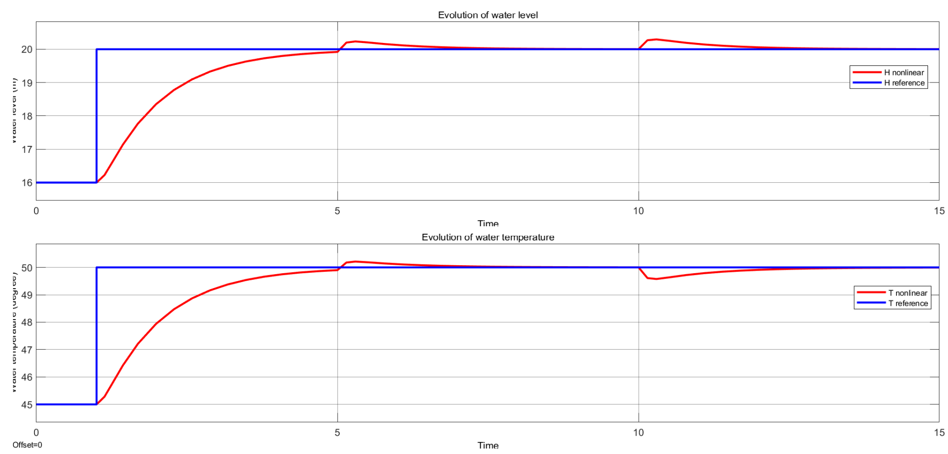

Figure 12: Integral nonlinear control approach.

We see that the disturbances are both rejected thanks to integral action.
We also remark that this integral controller is more robust w.r.t system modeling uncertainty, as we have tried with a different value of Q_0 like in the last part, and the results are almost exactly the same:

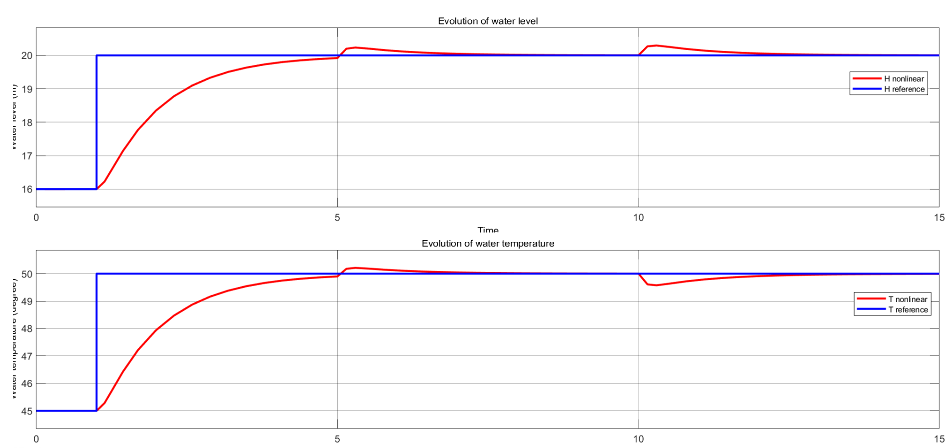

Figure 13: Integral nonlinear control approach robustness analysis.

6 Conclusion

After this lab, the students understand better the nonlinear control method that involves compensating the nonlinearity in the system. We see clearly that for nonlinear systems, we need nonlinear control approaches to achieve good performance. Also, the robustness of the controlled system can depend a lot on system modeling, i.e., the parameters we use to calculate the compensation part. After the system has been linearized thanks to the compensation, we can apply linear control methods, e.g., pole placement, LQR, and integral.