École Centrale Nantes

EMARO-CORO M1

Nonlinear control

Report of:

LUCIA BERGANTIN

MICHELE TARTARI

Introduction

The aim of this lab was to implement a control law for a nonlinear system, using the proper approach depending on the range of settings given in the four subsequent tasks. This was meant to show us when and why the various theories previously analysed in class are to be preferred one to the other, according to the situation dealt with.

The system given was based on the model of a PVTOL aircraft, as shown below:

$$\begin{aligned} \ddot{x} &= -\sin(\theta)\, u_1 + \epsilon\, cos(\theta) u_2 \\ \ddot{z} &= \cos(\theta)\, u_1 + \epsilon\, sin(\theta) u_2 - 1 \\ \ddot{\theta} &= u_2 \end{aligned}$$

considering the parameter $\varepsilon = 10^{-3}$. As we can see from a first quick analysis, the system has a state vector of dimension 6 (due to the presence of the second derivatives) and two inputs, u_1 and u_2.

The aim of our control was in every case to stabilize to zero the two outputs y_1=x and y_2=z.

Part 1: control through decoupling and linearization, with no perturbations nor uncertainties

In the first task, the system presented no perturbations nor uncertainties. Having two inputs and two outputs, the best approach in this case was to control the nominal system decoupling and linearizing it, using an input-output point-of-view approach. Using this method, each output was influenced by one and only one of the two inputs and not by both, as it would have happened in case we chose not to decouple the system. This, however, wouldn't have been a feasible approach because we wouldn't have been able to see clearly which input was affecting an output and how, making the system way more complicated to manage.

First, we analysed the system's relative degree. Since we had to differentiate the outputs two times before the inputs appeared in their equations, the system had a relative degree of 4, which was lower than the state vector dimension (6). Therefore, the system presented internal dynamics of dimension two on θ and $\dot{\theta}$.

Proceeding with the calculations, we came to the following conclusions:

$$\begin{pmatrix} \mathrm{u}_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} -\sin(\theta) & \cos(\theta) \\ \dfrac{\cos(\theta)}{\varepsilon} & \dfrac{\sin(\theta)}{\varepsilon} \end{pmatrix} \begin{pmatrix} w_1 \\ w_2 \end{pmatrix} + \begin{pmatrix} \cos(\theta) \\ \dfrac{\sin(\theta)}{\varepsilon} \end{pmatrix}$$

with

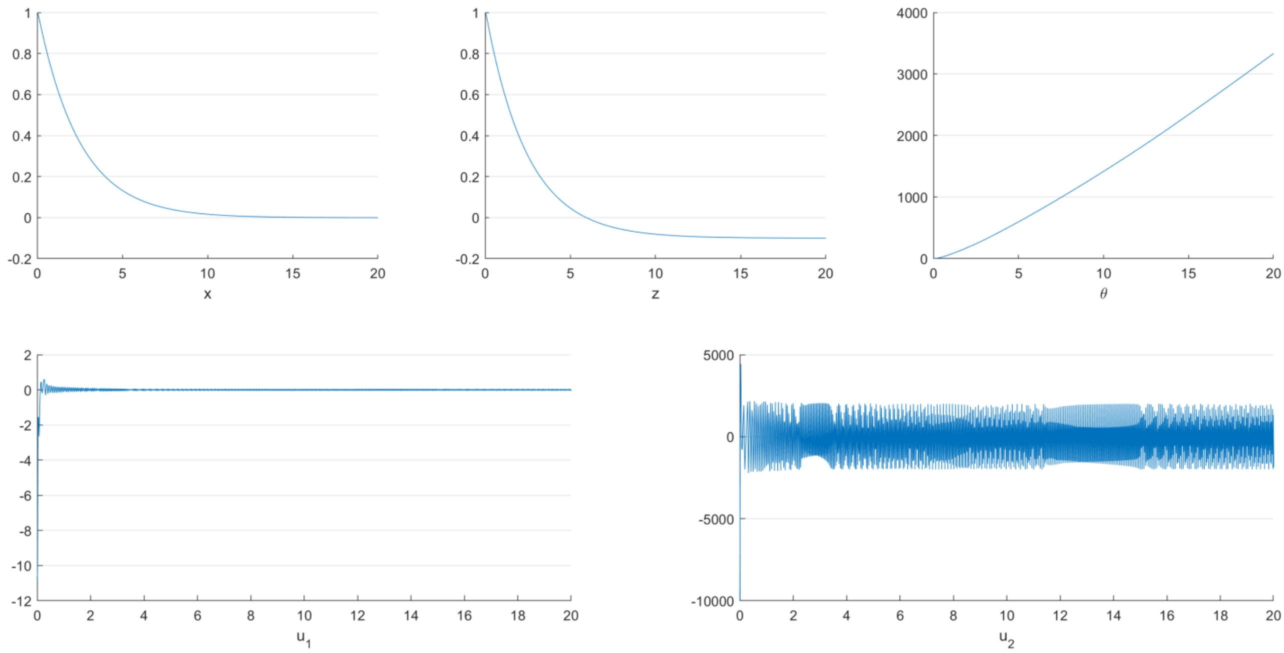
$$\begin{pmatrix} w_1 \\ w_2 \end{pmatrix} = \begin{pmatrix} -k_{11}\dot{x} - k_{12}x \\ -k_{11}\dot{z} - k_{12}z \end{pmatrix} = \begin{pmatrix} -\omega_1^2\dot{x} - 2\xi_1\omega x \\ -\omega_2^2\dot{z} - 2\xi_2\omega z \end{pmatrix}$$

Since the relative degree is not infinite, we can write

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} w_1 \\ w_2 \end{pmatrix}$$

As we can see, we obtained two different control laws, one for each output, depending only on one input each. The equations of w_1 and w_2 were set using the model of a second order system.

We then tuned the parameters, setting finally both responses ω to 5 and the damping coefficients respectively to 1 and 40. With these control laws and these settings, we obtained from the Simulink simulation the following figure:



As we can see, both outputs stabilized to 0 as required. However, the presence of internal dynamics made this approach not entirely effective, being them unstable as shown in the figure above. In fact, we can clearly see that the variable θ diverged to infinity.

To demonstrate mathematically that the internal dynamics were unstable, let us consider the equations of u_1 and u_2. They now read as

$$u_1 = -\sin(\theta)\, w_1 + \epsilon \cos(\theta)\, w_2 + \cos(\theta)$$
$$u_2 = \frac{\cos(\theta)}{\epsilon} w_1 + \frac{\sin(\theta)}{\epsilon} w_2 + \frac{\sin(\theta)}{\epsilon}$$

Assuming the objectives satisfied, both w_1 and w_2 tended to 0 so the equations now tended to

$$u_1 = \cos(\theta)$$
$$u_2 = \frac{\sin(\theta)}{\epsilon}$$

which are not linear functions and imply an oscillation on the variable θ. Thus, the internal dynamics were not stable.

Part 2: dynamical state feedback control

In this second task, we considered $\varepsilon = 0$ and the possible presence of uncertainties on θ. Thus, the system model read as:

$$
\begin{aligned}
\ddot{x} &= \quad -\sin(\theta)\,u_1 \\
\ddot{z} &= \quad \cos(\theta)\,u_1 - 1 \\
\ddot{\theta} &= \quad u_2 + \delta(t)
\end{aligned}
$$

At first, we considered the uncertainties null (δ(t)=0). The system could therefore be written in matrixial form as shown below:

$$
\begin{pmatrix} y_1^{(2)} \\ y_2^{(2)} \end{pmatrix} = \begin{pmatrix} -\sin(\theta) & 0 \\ \cos(\theta) & 0 \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} + \begin{pmatrix} 0 \\ -1 \end{pmatrix} = \beta u + \alpha
$$

With

$$
\beta = \begin{pmatrix} -\sin(\theta) & 0 \\ \cos(\theta) & 0 \end{pmatrix}
$$

$$
\alpha = \begin{pmatrix} 0 \\ -1 \end{pmatrix}
$$

Although, as it can be seen above, matrix β didn't depend on the input u_2. This made decoupling as done in the previous task impossible. It was therefore necessary to use a different approach and we opted for a dynamic state feedback controller, using as inputs $u_1^{(2)}$ and u_2. To implement this new approach, we differentiated the outputs until the new variables appeared. We had to differentiate four times and therefore the relative degree of the system was now increased to 4 (which meant the internal dynamics were still present). The system could then be written in matrixial form as shown below:

$$
\begin{pmatrix} y_1^{(4)} \\ y_2^{(4)} \end{pmatrix} = \begin{pmatrix} -\sin(\theta) & -u_1\cos(\theta) \\ \cos(\theta) & -u_1\sin(\theta) \end{pmatrix} \begin{pmatrix} u_1^{(2)} \\ u_2 \end{pmatrix} + \begin{pmatrix} -2\dot{u}_1\cos(\theta)\,\dot{\theta} + u_1\sin(\theta)\dot{\theta}^2 \\ -2\dot{u}_1\sin(\theta)\,\dot{\theta} - u_1\cos(\theta)\dot{\theta}^2 \end{pmatrix} = \beta_* u_* + \alpha_*
$$

With

$$
\begin{aligned}
u_* &= \begin{pmatrix} u_1^{(2)} \\ u_2 \end{pmatrix} \\
\alpha_* &= \begin{pmatrix} -2\dot{u}_1\cos(\theta)\,\dot{\theta} + u_1\sin(\theta)\dot{\theta}^2 \\ -2\dot{u}_1\sin(\theta)\,\dot{\theta} - u_1\cos(\theta)\dot{\theta}^2 \end{pmatrix} \\
\beta_* &= \begin{pmatrix} -\sin(\theta) & -u_1\cos(\theta) \\ \cos(\theta) & -u_1\sin(\theta) \end{pmatrix}
\end{aligned}
$$

The new laws obtained were now:

$$
\begin{pmatrix} \mathrm{u}_1 \\ u_2 \end{pmatrix} = \beta_*^{-1} \cdot \begin{pmatrix} w_1 \\ w_2 \end{pmatrix} - \alpha_* = \begin{pmatrix} -\sin(\theta) & \cos(\theta) \\ -\dfrac{\cos(\theta)}{\mathrm{u}_1} & -\dfrac{\sin(\theta)}{\mathrm{u}_1} \end{pmatrix} \begin{pmatrix} w_1 \\ w_2 \end{pmatrix} + \begin{pmatrix} 2\dot{u}_1\cos(\theta)\,\dot{\theta} - u_1\sin(\theta)\dot{\theta}^2 \\ 2\dot{u}_1\sin(\theta)\,\dot{\theta} + u_1\cos(\theta)\dot{\theta}^2 \end{pmatrix}
$$

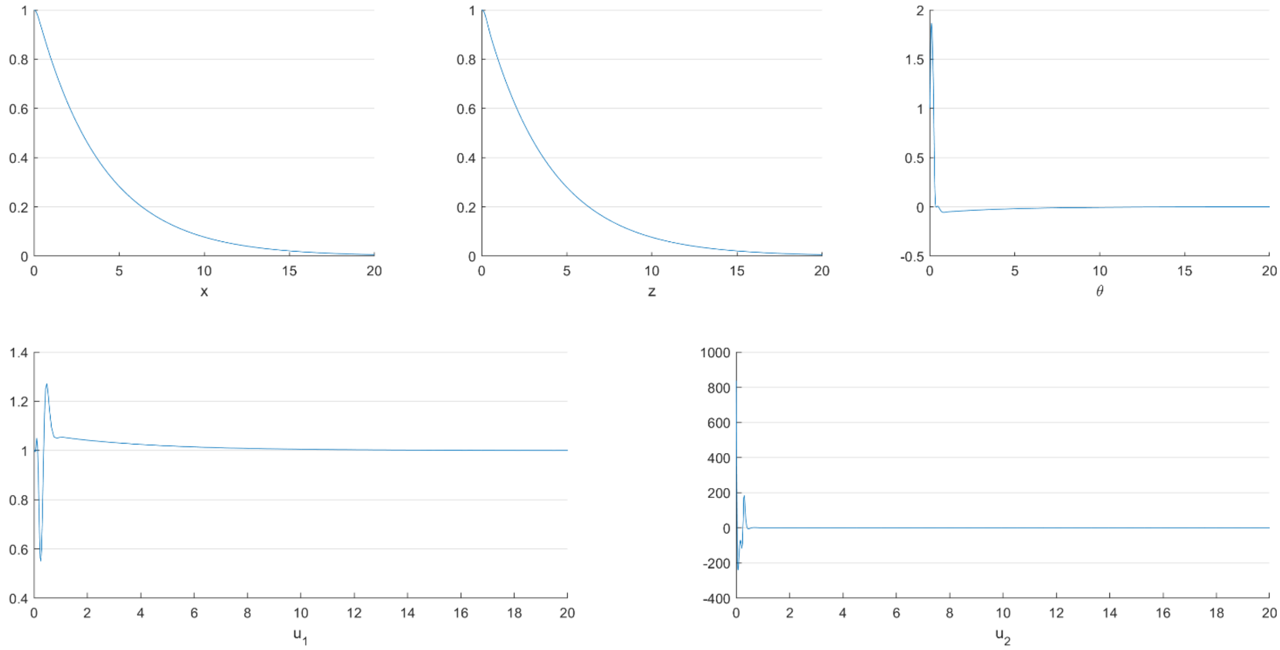
Which gives

$$
\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} w_1 \\ w_2 \end{pmatrix}
$$

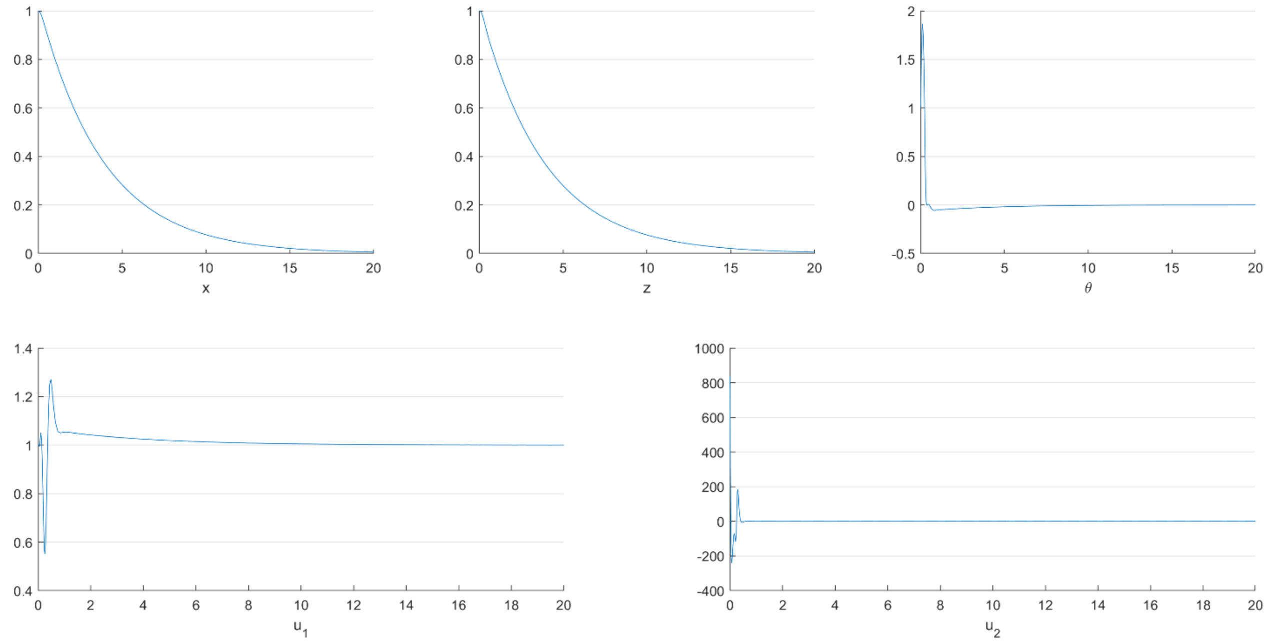
Note that with this new approach the equations of w_1 and w_2 changed drastically compared to the previous task. In this case we had to use Akerman's pole placement technique and their equations read as

$$\begin{pmatrix} w_1 \\ w_2 \end{pmatrix} = \begin{pmatrix} -K_1 x \\ -K_2 z \end{pmatrix}$$

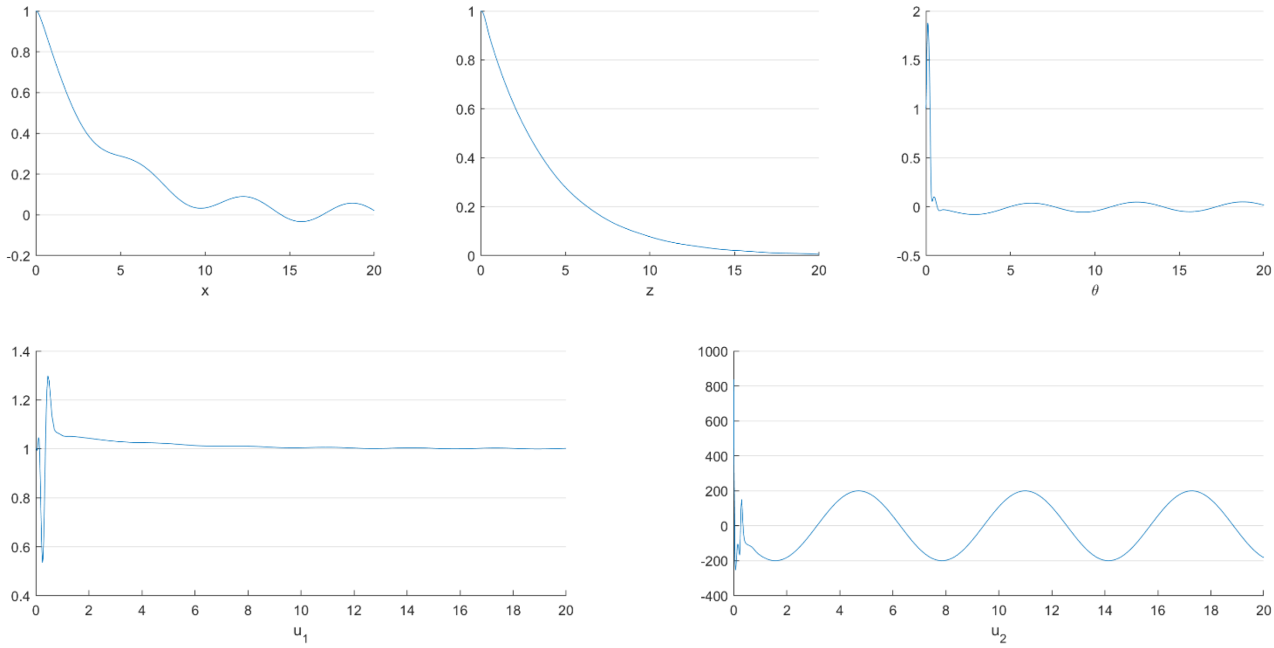
with K_1 and K_2 matrices. This was due to the fact that the system couldn't be approximated anymore with the same precision to a second order system, as in the previous case, imposing a more sophisticated approach. With these control laws and the same settings used in the first task, we obtained from the Simulink simulation the following figure:



To complete the analysis of this second part, we considered the presence of uncertainties. Firstly, we set δ(t)=200, obtaining the following figure:



Then, we set δ(t)=200sin(t), obtaining the following figure:



As we can see, once introduced uncertainties in the system, the stabilization stopped working properly. We could then affirm that the system wasn't robust and that this approach was then not sufficiently efficient to achieve our goal. A possible solution was then to change approach completely and implement the controller by the means of the sliding control theory previously introduced in class.

Part 3: sliding mode control

In this third part, we implemented the controller by the means of the sliding control theory to fix the problems previously analysed in Part 2. First, we had to define the two sliding variables and to do so we considered the two linearized systems

$$\begin{cases} y_1^{(1)} = x^{(1)} \\ y_1^{(2)} = x^{(2)} \\ y_1^{(3)} = x^{(3)} \\ y_1^{(4)} = w_1 \end{cases} \qquad \begin{cases} y_2^{(1)} = z^{(1)} \\ y_2^{(2)} = z^{(2)} \\ y_2^{(3)} = z^{(3)} \\ y_2^{(4)} = w_2 \end{cases}$$

We could then proceed to define the sliding variables:

$$\begin{cases} \dot{s_1} = y_1^{(4)} + \lambda_{11} y_1^{(3)} + \lambda_{12} y_1^{(2)} + \lambda_{13} y_1^{(1)} = w_1 + \lambda_{11} x^{(3)} + \lambda_{12} x^{(2)} + \lambda_{13} x^{(3)} \\ \dot{s_1} = -k \cdot sign(s_1) \end{cases}$$

$$\begin{cases} \dot{s_2} = y_2^{(4)} + \lambda_{21} y_2^{(3)} + \lambda_{22} y_2^{(2)} + \lambda_{23} y_2^{(1)} = w_2 + \lambda_{21} z^{(3)} + \lambda_{22} z^{(2)} + \lambda_{23} z^{(1)} \\ \dot{s_2} = -k \cdot sign(s_2) \end{cases}$$

Putting the first equation of each system equal to zero we obtained:

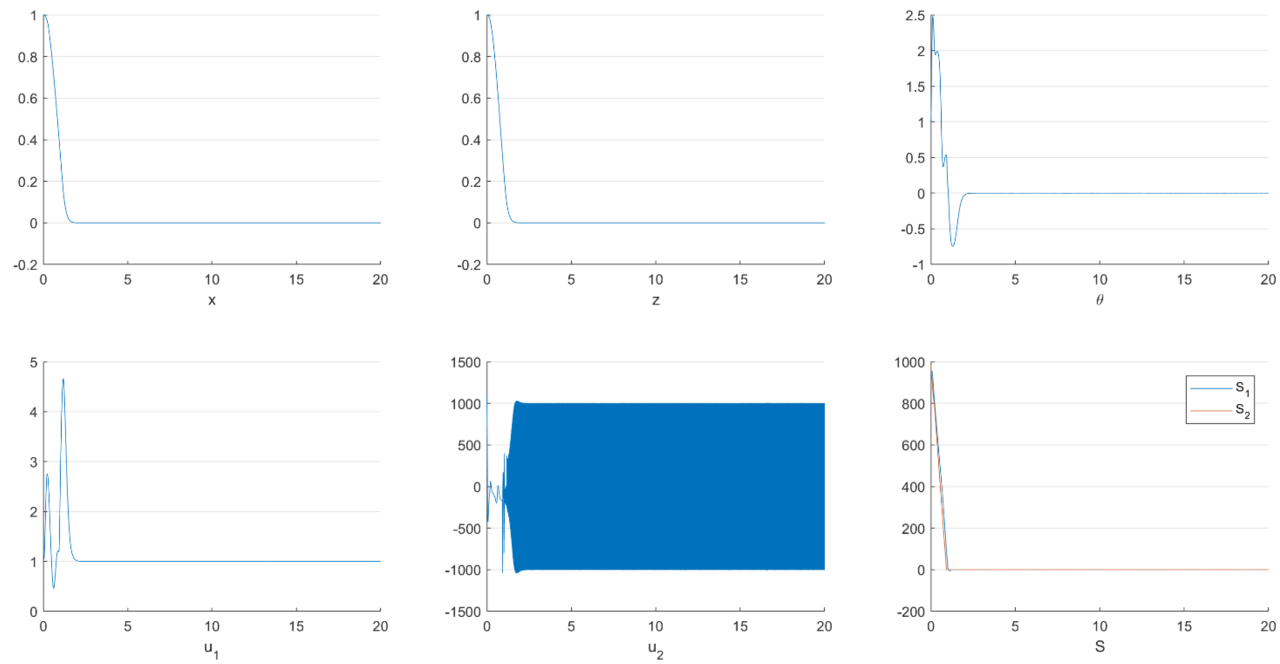
$$w_1 = -\lambda_{11} x^{(3)} - \lambda_{12} x^{(2)} - \lambda_{13} x^{(1)}$$

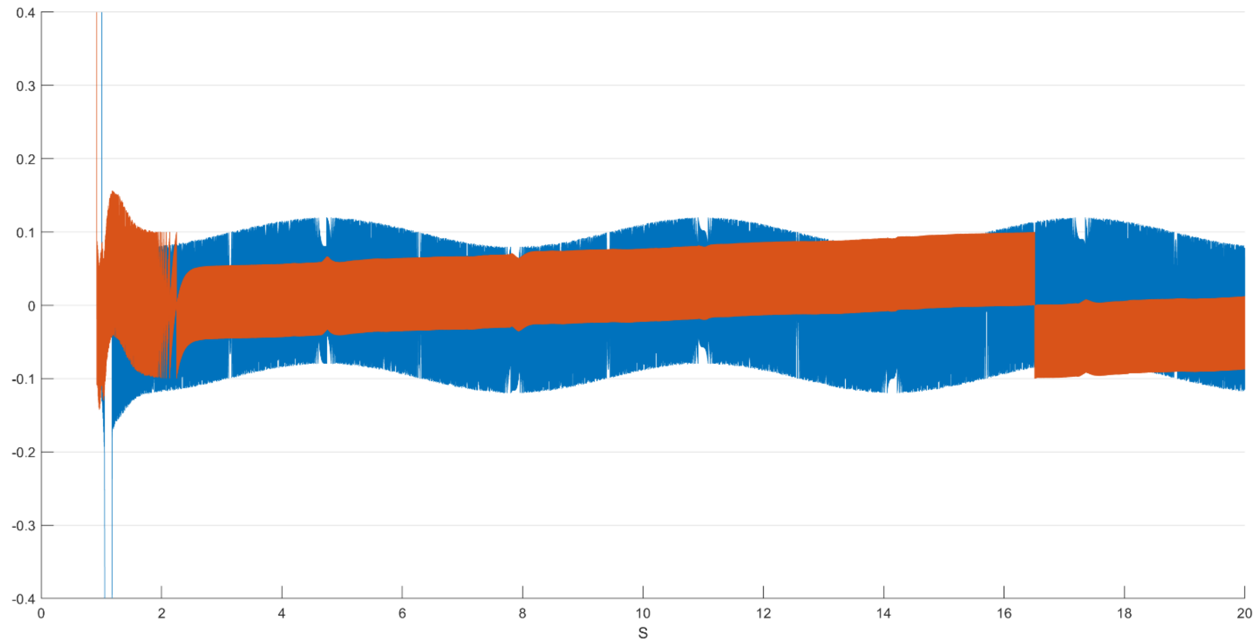
$$w_2 = -\lambda_{21} z^{(3)} - \lambda_{22} z^{(2)} - \lambda_{23} z^{(1)}$$

The linearized systems could now be written as shown below:

$$\begin{cases} \dot{x_1} = x_2 \\ \dot{x_2} = x_3 \\ \dot{x_3} = -k sign(s_1) - \lambda_{11} x_3 - \lambda_{12} x_2 - \lambda_{13} x_1 \end{cases} \qquad \begin{cases} \dot{x_1} = x_2 \\ \dot{x_2} = x_3 \\ \dot{x_3} = -k sign(s_2) - \lambda_{21} x_3 - \lambda_{22} x_2 - \lambda_{23} x_1 \end{cases}$$

We used Akerman's pole placement technique to find the right values of the λ variables and we then inserted them in the sliding variables equations. With these control laws and the found settings, we obtained from the Simulink simulation the following figure:





As we can see, the outputs stabilized to zero as required. Still the control wasn't ideal, since the θ variable presented an oscillation due to the chattering effect, which is a consequence of the fact that the system had trouble reaching the sliding surface exactly and therefore only stabilized partially. In the figure above, we can see the chattering effect on the sliding variables using 200sin(t) as disturbance.

Part 4: adaptive sliding mode control

In the fourth part, we tried to fix the chattering effect problem implementing the controller by the means of the adaptive sliding mode theory. The main difference with the previous method is the fact that the value of the gain wasn't tuned manually, as done so far, but was instead tuned automatically (or "online") by the simulation itself.

Still using the two sliding variables s_1 and s_2 defined in the previous task, we considered

$$w_1 = -K_1 \cdot sign(s_1)$$

$$w_2 = -K_2 \cdot sign(s_2)$$

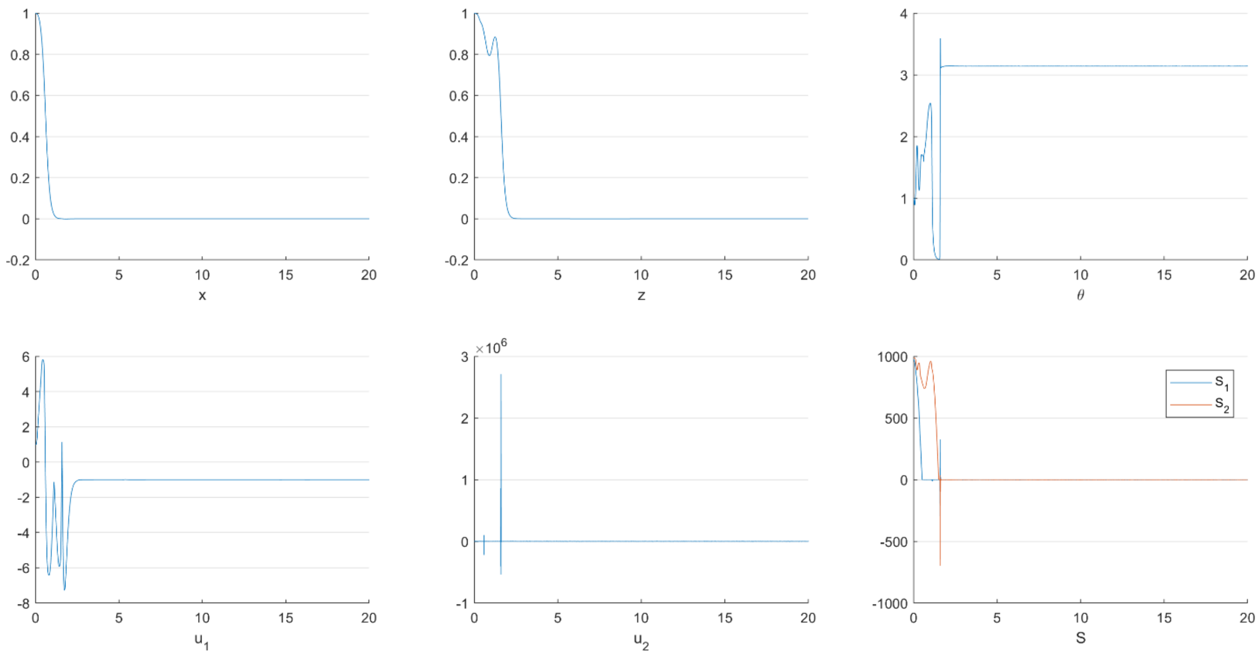
The two gains K_1 and K_2 were tuned through their derivative by the means of the given sliding condition

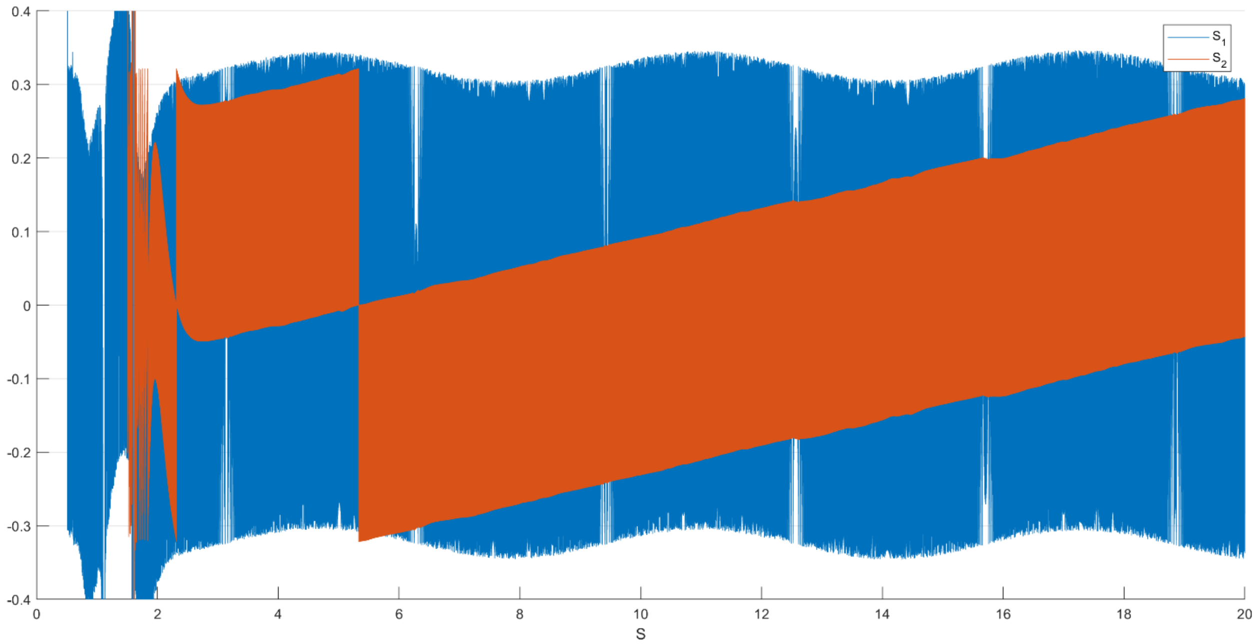
$$\dot{K}_i = \begin{cases} \bar{K} \cdot |\sigma_i| \cdot \mathrm{sign}(|\sigma_i| - \mu_i) & \text{if } K_i > \eta_i \\ \eta_i & \text{if } K_i \leq \eta_i \end{cases}$$

which would not allow their value to go neither below a set value η (in our case equal to 1) nor higher than an upper set boundary, depending on the value of the gain. This limit depended on the sliding variable's module and it could have been regulated by the following parameters:

- μ, which is the tolerance between the sliding variable and 0. Its minimum value is set by the formula $\mu = 2KT = 0.02$. Although, to make the simulation perform better we had to increase the value to 0.023.
- $\overline{K}$ (equal to 10), which acted as fixed gain on the $\dot{K}_\iota$, increasing iteratively the value of the gain only of a certain set value each time.

With this control law and the new settings, we obtained from the Simulink simulation the following figure:





As we can see, in this case not only the outputs stabilized to zero as required. In the figure above, we can see the chattering effect on the sliding variables using 200sin(t) as disturbance. Unfortunately, the phenomenon instead of decreasing when compared to Part 3 (as expected by theory) augments slightly, due to a tuning problem that we were not able to solve.

Conclusions

In this lab we implemented control laws for a nonlinear system, using the more appropriate approach depending on the settings given in each task. This way, we learnt to recognize which method was more efficient depending on the situation dealt with and its complexity. We saw that, even if a more traditional and simplistic approach still worked when applied to the given PVTOL model, the controller wasn't robust and failed to deal with perturbations and uncertainties. We progressed then to a more sophisticated approach applying the concepts previously analysed in class of sliding mode theory. Still, we observed that even this approach had its problems due to the chattering effect on the variable θ. We then modified our method slightly, applying an adaptive approach to the gain's tuning, trying to fix the chattering problem. However, due to a tuning issue we did not manage.