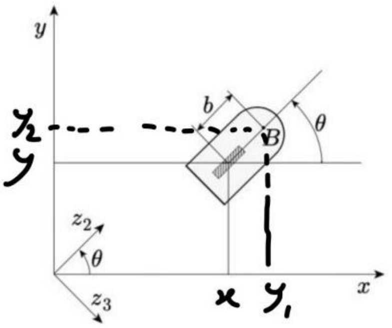
Implement via software an input/output linearization control approach to control the unicycle's position and bring it from q_i=[0 0 0] to the point q_f=[4 1 $\frac{\pi}{2}$]. Adjust the trajectory accordingly to fit the desired coordinates of the reference point B along the sagittal axis.

Feedback linearization is a strategy employed to control nonlinear systems. Feedback linearization techniques may be applied to nonlinear control systems of the form

$$\dot{x} = f(x) + g(x)u$$
$$y = h(x)$$

Where x(t)∈ $\mathbb{R}^n$ is the state vector, y∈ $\mathbb{R}$ is the output vector and u∈ $\mathbb{R}$ is the output vector.

The approach involves transforming a nonlinear control system into an equivalent linear control system through a change of variables and a suitable control input. In our case:



B is a point with distance b=0.04 from the center of the wheel along the sagittal axis.

(x,y) are the coordinates of the center of the wheel.

(y_1, y_2) are the coordinates of the point B.

B has coordinates:

$$y_1 = x + b\cos\theta$$
$$y_2 = y + b\sin\theta$$

The time derivatives of these outputs are (substituting the kinematic model of the unicycle). Notice that det(T(θ))$\neq 0 \leftrightarrow b \neq 0$.

$$\begin{bmatrix} \dot{y}_1 \\ \dot{y}_2 \end{bmatrix} = \underbrace{\begin{bmatrix} \cos\theta & -b\sin\theta \\ \sin\theta & b\cos\theta \end{bmatrix}}_{T(\theta)} \begin{bmatrix} v \\ \omega \end{bmatrix} = T(\theta) \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (1)$$

It is possible to design

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = T(\theta)^{-1} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad (2)$$

u_1, u_2 are the 2 virtual control input.

Substituting (2) into (1), we get

$$\begin{cases} \dot{y}_1 = u_1 \\ \dot{y}_2 = u_2 \end{cases}$$

The following simple controller can be designed(with k_1, k_2>0). This controller guarantees exponential convergence to the desired $y_{1,d}$ and $y_{2,d}$. For this implementation I chose $k_1 = k_2$ =2

$$u_1 = \dot{y}_{1,d} + k_1(y_{1,d} - y_1)$$
$$u_2 = \dot{y}_{2,d} + k_2(y_{2,d} - y_2)$$

Unfortunately, this approach controls the position of the point B only, leaving the orientation uncontrolled.

The following implementation is based on the trajectory calculated in the previous exercise, with the appropriate modifications to fit the desired coordinates of the reference point B along the sagittal axis.

The scheme Simulink for this implementation is saved as input output linearization.slx

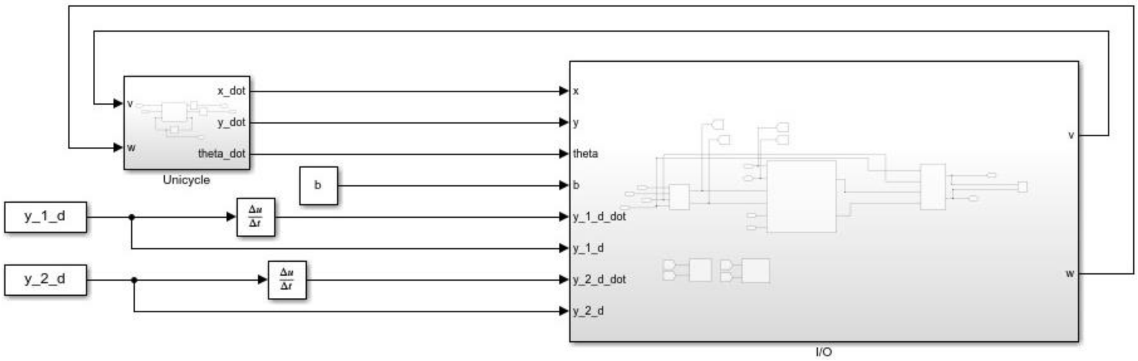


Figure 1:Scheme of the third exercise

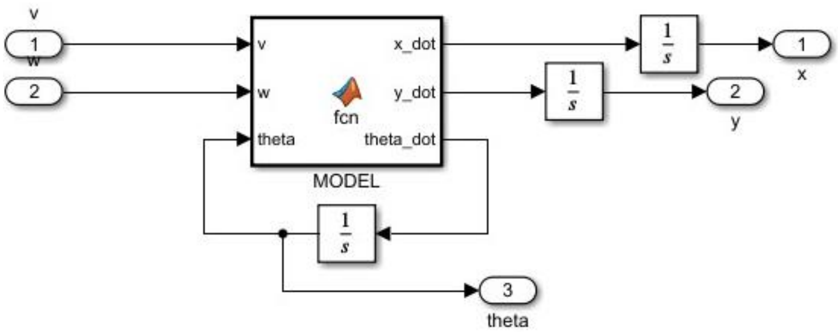


Figure 2:unyicicle

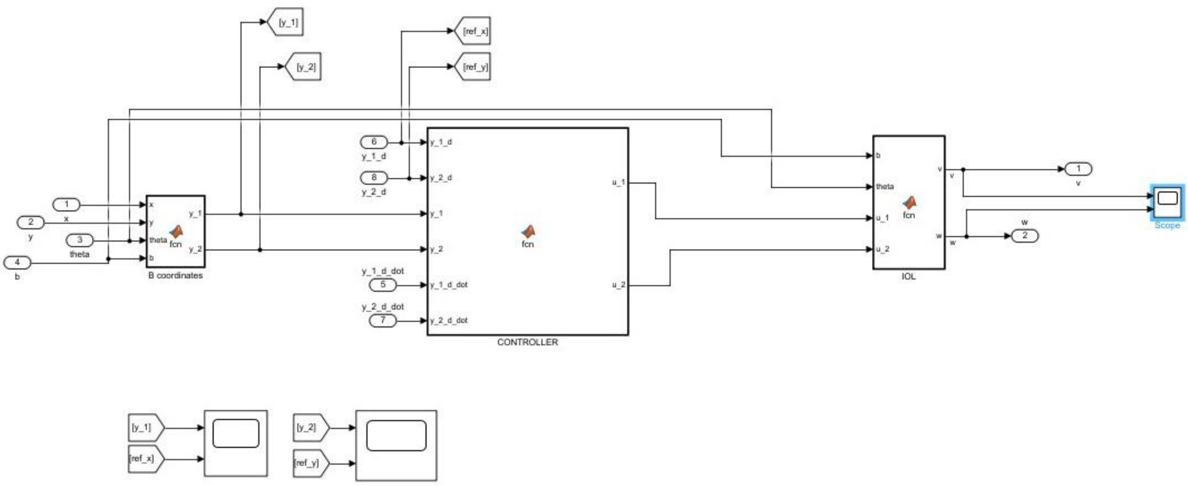


Figure 3:Subsystem third exercise

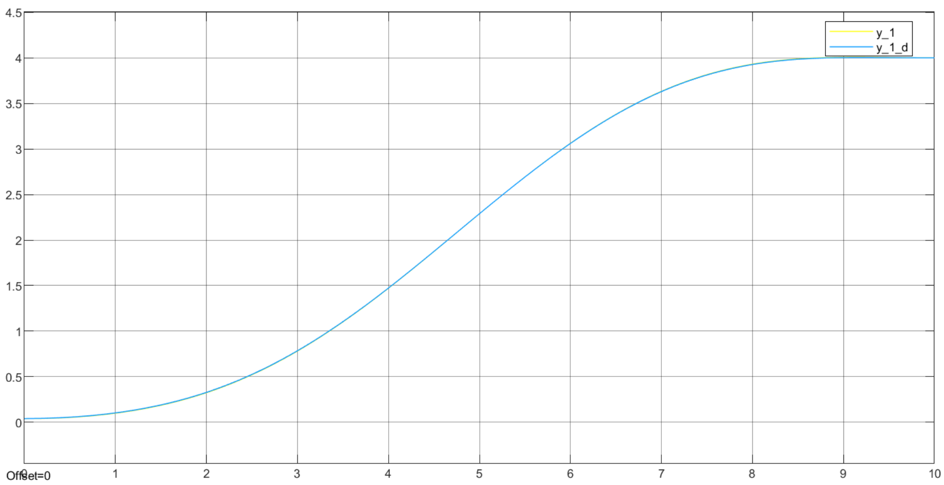


Figure 4:y_1_desired and the real y_1

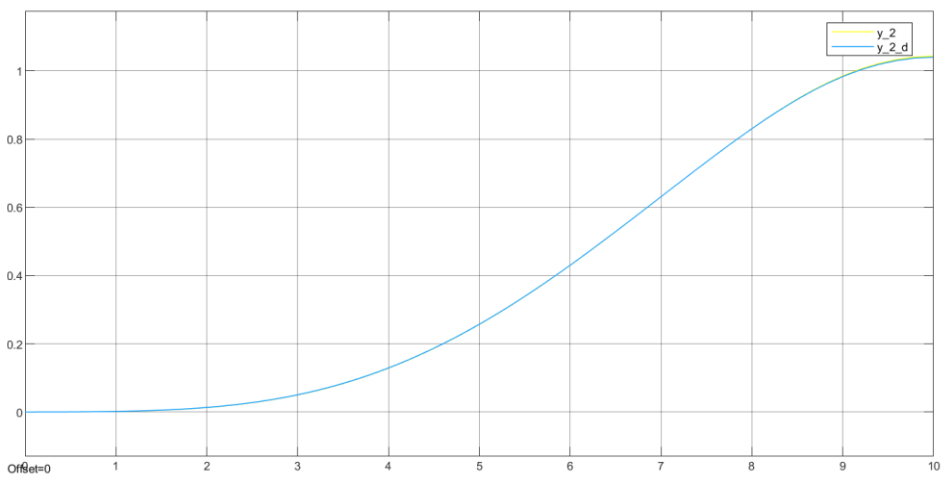


Figure 5:y_2_desired and the real y_2

As it is possible to note, the real and desired trajectories are very similar. The point B does not converge perfectly in the point (4,1), because the reference point remains the center of the unicycle, and therefore B maintains a distance b from the center.

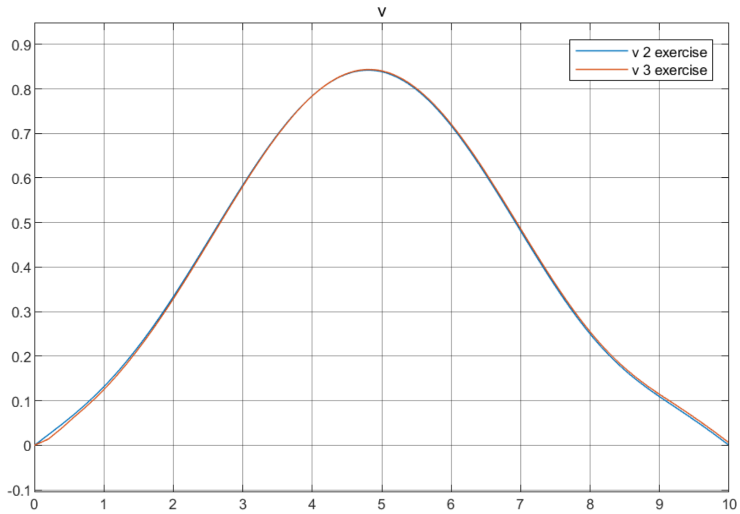


Figure 6:values of v in second and third exercise

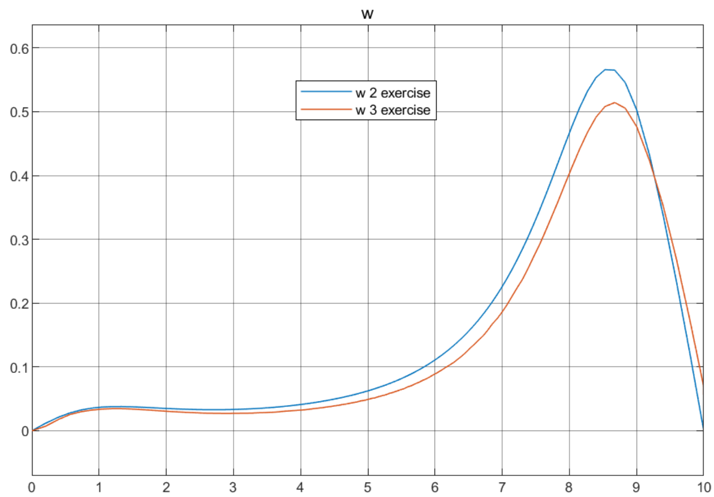


Figure 7:value of w in second and third exercise

The trends of v between the second and third exercise are very similar, while as regards ω it can be seen that in the second exercise the function takes on a greater value. So we can say that v and ω stay under the limit of 1 in module.

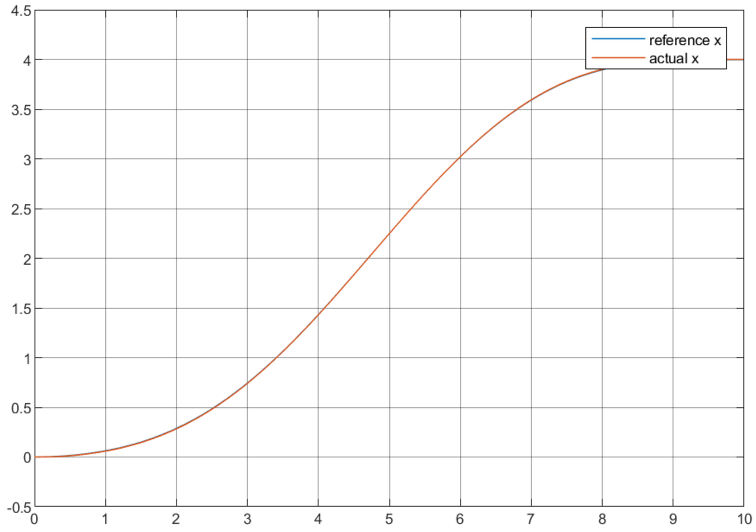


Figure 8:the actual signal of x and the reference signal x form the previous exercise

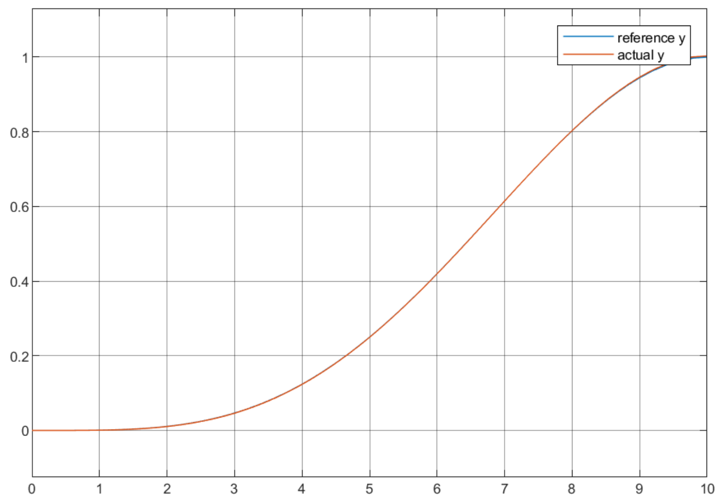


Figure 9:the actual signal of x and the reference signal x form the previous exercise

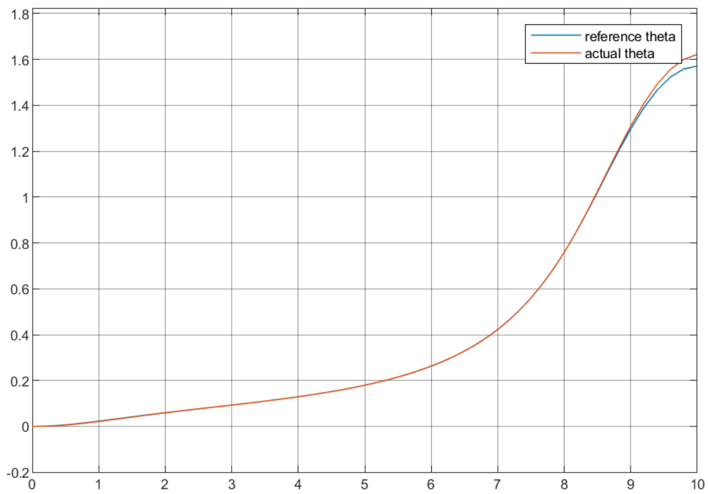


Figure 10:the actual signal of theta and the reference signal theta form the previous exercise

We can see that the signals x y θ are well tracked, and even if θ is uncontrollable, it follows the reference well due to the path and the chosen earnings.