FACULTY OF ENGINEERING SCIENCE
DEPARTMENT OF MECHANICAL ENGINEERING

KU LEUVEN

Control Theory: Assignment 4: Control of Pendulum

Maxime Brion (r0891121)
Mauro Van Tichelen (r0902198)

Professor: Prof. Swevers Jan

2024 - 2025

Contents

List of Figures

1 Modelling the System formed by the Pendulum

1a Derivation of State-Space Equations for Nonlinear Models

The nonlinear dynamics of the pendulum-cart system are represented by a state-space model where the state vector is defined as:

$$\xi = \begin{bmatrix} x \\ \theta \\ L\dot{\theta} + \dot{x}\cos(\theta) \end{bmatrix} \tag{1}$$

Here, x represents the cart's horizontal position, θ is the angular displacement of the pendulum from the vertical, and $L\dot{\theta} + \dot{x}\cos(\theta)$ combines the pendulum's angular velocity with the cart's horizontal velocity projected along the pendulum rod. This choice of states captures both the translational and rotational dynamics of the system.

The nonlinear dynamics are derived from the physical laws governing the system. The complete nonlinear state-space model for the dynamics is:

$$\dot{\xi} = f(\xi, v) = \begin{bmatrix} v \\ \frac{\xi_3}{L} - \frac{v\cos(\xi_2)}{L} \\ -g\sin(\xi_2) - \left(v\sin(\xi_2) + \frac{c}{m}\right) \cdot \left(\frac{\xi_3}{L} - \frac{v\cos(\xi_2)}{L}\right) \end{bmatrix} \tag{2}$$

The measurement equation relates the observable outputs, the cart position x and pendulum angle θ, directly to the states. These outputs are defined as:

$$y = \begin{bmatrix} \xi_1 \\ \xi_2 \end{bmatrix} \tag{3}$$

or equivalently in matrix form:

$$y = C\xi, \quad C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \tag{4}$$

2 Comparing Discretization and Linearization for Discrete Linear Models

2a Discretizing Continuous Nonlinear Models

To discretize the continuous-time nonlinear model, the Forward-Euler method approximates the time derivative using a first-order finite difference:

$$\dot{\xi}(kT_s) \approx \frac{\xi(k+1) - \xi(k)}{T_s} \tag{5}$$

where $T_s = 0.01s$ is the sampling time and k is the current time step. Rearranging gives the update rule:

$$\xi(k+1) = \xi(k) + T_s \cdot \dot{\xi}(k) \tag{6}$$

The continuous-time model is visible in Equation 2. Applying Forward-Euler to each state gives the discrete-time model:

$$\xi(k+1) = \xi(k) + T_s \cdot \begin{bmatrix} v(k) \\ \frac{\xi_3(k)}{L} - \frac{v(k)\cos(\xi_2(k))}{L} \\ -g\sin(\xi_2(k)) - \left(v(k)\sin(\xi_2(k)) + \frac{c}{m}\right)\left(\frac{\xi_3(k)}{L} - \frac{v(k)\cos(\xi_2(k))}{L}\right) \end{bmatrix} \tag{7}$$

This provides a discrete-time approximation of the nonlinear dynamics using Forward-Euler, where the state $\xi(k)$ is updated with the input $v(k)$ and sampling time T_s.

Furthermore, to discretize the measurement equations, we start with the continuous-time measurement model:

$$y(t) = h(\xi(t), v(t)) \tag{8}$$

where $y(t)$ is the output, $\xi(t)$ is the state vector, and $v(t)$ is the control input. By evaluating this model at discrete time steps $t = kT_s$, where T_s is the sampling time, we obtain:

$$y(k) = h(\xi(k), v(k)) \tag{9}$$

If the measurements are direct observations of the states, such as the cart position x and pendulum angle θ, the continuous-time measurement model simplifies to:

$$y(t) = \begin{bmatrix} \xi_1(t) \\ \xi_2(t) \end{bmatrix} \tag{10}$$

Discretizing this by sampling at $t = kT_s$ gives:

$$y(k) = \begin{bmatrix} \xi_1(k) \\ \xi_2(k) \end{bmatrix} \tag{11}$$

In matrix form, the discrete-time measurement equation becomes:

$$y(k) = C \cdot \xi(k) \tag{12}$$

where:

$$C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \tag{13}$$

Here, $y(k)$ represents the cart position $\xi_1(k)$ and pendulum angle $\xi_2(k)$, which are directly measured components of the state vector $\xi(k)$. This transformation provides a discrete-time version of the continuous-time measurement model, sampled at each time step k.

2b Linearizing the Discrete-Time Model Around a State ξ

The state ξ is linearized around the equilibrium point:

$$\xi^* = \begin{bmatrix} x^* \\ \theta^* \\ \xi_3^* \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{14}$$

where x^* represents the cart's position, θ^* the pendulum angle, and ξ_3^* the combined term $L\dot{\theta}+\dot{x}\cos(\theta)$.

This equilibrium corresponds to the upright pendulum ($\theta^* = 0$) and cart at rest ($x^* = 0$, $\xi_3^* = 0$). It is chosen as the operating point because it simplifies the system dynamics, where $\sin(\theta)$ and $\cos(\theta)$ reduce to $\sin(0) = 0$ and $\cos(0) = 1$. The resulting linearized model accurately approximates the system's behavior for small deviations near this point.

This linearization ensures that the Extended Kalman Filter (EKF) operates effectively by using the simplified dynamics to track disturbances and stabilize the pendulum around the upright position. It aligns with standard practices for nonlinear systems and filtering applications.

Furthermore, to derive the Jacobian of the discrete-time state and measurement equations and obtain a linearized model, we begin with the nonlinear discrete-time system:

$$\xi(k+1) = \xi(k) + T_s \cdot f(\xi(k), v(k)) \tag{15}$$

where $f(\xi, v)$ is the same as equation 2. The measurement equation is:

$$y(k) = h(\xi(k)) = C\xi(k), \quad C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \tag{16}$$

The system is linearized around $\xi^* = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$, $v^* = 0$, representing the upright pendulum and stationary cart. The Jacobians of $f(\xi, v)$ with respect to ξ and v are computed as follows:

- Jacobian $J_\xi = \frac{\partial f}{\partial \xi}$:

$$J_\xi = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & \frac{1}{L} \\ 0 & -g & -\frac{c}{mL} \end{bmatrix} \tag{17}$$

- Jacobian $J_v = \frac{\partial f}{\partial v}$:

$$J_v = \begin{bmatrix} 1 \\ -\frac{1}{L} \\ -\frac{1}{L} \cdot \frac{c}{m} \end{bmatrix} \tag{18}$$

The measurement equation is already linear, with the Jacobian:

$$J_y = C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \tag{19}$$

The linearized discrete-time state-space model is:

- State Dynamics:

$$\xi(k+1) = A\xi(k) + Bv(k) \tag{20}$$

where:

$$A = I + T_s J_\xi = I + T_s \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & \frac{1}{L} \\ 0 & -g & -\frac{c}{mL} \end{bmatrix}, \quad B = T_s J_v = T_s \begin{bmatrix} 1 \\ -\frac{1}{L} \\ -\frac{1}{L} \cdot \frac{c}{m} \end{bmatrix} \tag{21}$$

- Measurement Equation:

$$y(k) = C\xi(k), \quad C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \tag{22}$$

This linearized model is valid for small deviations around the equilibrium point, where nonlinear terms are negligible. It is particularly suitable for estimation and control applications such as the Extended Kalman Filter (EKF).

2c Linearize and Discretize with Zero-Order Hold

A good understanding of system dynamics is crucial for effective modeling and control design. To linearize the system around the equilibrium point $\xi^* = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$ and $v^* = 0$, we compute the Jacobians of $f(\xi, v)$ with respect to ξ and v.

The Jacobian of $f(\xi, v)$ with respect to ξ is given by:

$$J_\xi = \frac{\partial f}{\partial \xi} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & \frac{1}{L} \\ 0 & -g & -\frac{c}{mL} \end{bmatrix} \tag{23}$$

Similarly, the Jacobian with respect to v is:

$$J_v = \frac{\partial f}{\partial v} = \begin{bmatrix} 1 \\ -\frac{1}{L} \\ -\frac{1}{L} \cdot \frac{c}{m} \end{bmatrix} \tag{24}$$

Using these Jacobians, the linearized continuous-time state-space model is expressed as:

$$\dot{\xi} = A\xi + Bv, \tag{25}$$

where the matrices A and B are:

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & \frac{1}{L} \\ 0 & -g & -\frac{c}{mL} \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ -\frac{1}{L} \\ -\frac{1}{L} \cdot \frac{c}{m} \end{bmatrix} \tag{26}$$

To discretize the linearized continuous-time model, the zero-order hold (ZOH) approach is applied. The continuous-time state-space model is described as:

$$\dot{\xi} = A\xi + Bv, \quad y = C\xi \tag{27}$$

where the matrices A, B, and C are defined as follows:

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & \frac{1}{L} \\ 0 & -g & -\frac{c}{mL} \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ -\frac{1}{L} \\ -\frac{1}{L} \cdot \frac{c}{m} \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \tag{28}$$

The model represents the linearized dynamics around the equilibrium point. Using a given sampling time T_s, the continuous-time model is converted into its discrete-time equivalent by applying the ZOH method. The resulting discrete-time state-space model is expressed as:

$$\xi(k+1) = A_d\xi(k) + B_dv(k), \quad y(k) = C_d\xi(k) + D_dv(k) \tag{29}$$

where the discrete-time matrices A_d, A_d, C_d, and D_d are derived from the continuous-time matrices. Specifically, A_d and B_d are computed as:

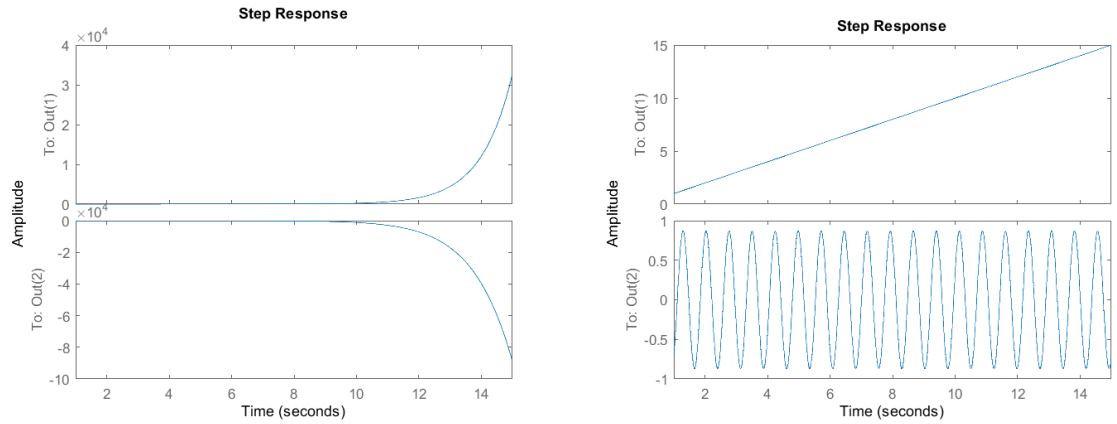
$$A_d = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0.9964 & 0.07398 \\ 0 & -0.09798 & 0.9964 \end{bmatrix}, \quad B_d = \begin{bmatrix} 0.01 \\ -0.07398 \\ 0.003631 \end{bmatrix} \tag{30}$$

Additionally, $C_d = C$ and $D_d = 0$, ensuring that the relationship between states and outputs remains consistent with the continuous-time model.

The discretization process was implemented in MATLAB using the 'ss' command to define the continuous-time state-space system and the 'c2d' command to perform the conversion. This computational approach leverages MATLAB's built-in functions to ensure accuracy in the discretization.

2d Stability Analysis and Comparison of Linear Discrete-Time System

While examining the step responses of two discrete-time systems derived using Forward-Euler and Zero-Order Hold (ZOH) methods, distinct differences emerge. The Forward-Euler method shows divergence in its step response as numerical errors accumulate with larger sampling periods, causing instability (Figure 1a). Conversely, the ZOH method yields a stable and accurate step response that closely mirrors the continuous-time dynamics, even for larger sampling periods (Figure 1b). This highlights the robustness and precision of the ZOH method in preserving system behavior.



(a) FE (b) ZOH

Figure 1: Step Response FE and ZOH

3 Linear Kalman Filter Design

3a Linear vs. Extended Kalman Filter

The Linear Kalman Filter (LKF) assumes linear dynamics and measurements, using fixed matrices A and C, while the Extended Kalman Filter (EKF) handles nonlinear systems by linearizing the process and measurement models at each step using Jacobians A_k and C_k. This enables the EKF to approximate nonlinear behavior but introduces potential errors from inaccurate linearization.

Both filters assume Gaussian noise (Q and R), but the EKF's noise propagation depends on local linearization, which can lead to more variability in uncertainty. The LKF is computationally efficient and consistent, as it applies noise directly to linear models, whereas the EKF's performance

may degrade in highly nonlinear systems.

The LKF is ideal for linear systems due to its simplicity and robustness, while the EKF is suited for moderately nonlinear systems where linearization remains accurate. The choice depends on system characteristics, with the EKF better handling nonlinearity at the cost of complexity and sensitivity to model assumptions.

3b Incorporating Measurement and Process Noise in the Mode

In the cart-pendulum setup, process noise (Q) arises from uncertainties in system dynamics, such as unmodeled effects (e.g., damping or friction), external disturbances (e.g., uneven surfaces or air resistance), and imperfections in the control signal, including delays and motor saturation. Linearization and discretization errors, such as those from Forward Euler, also contribute to process noise.

Measurement noise (R) originates from sensor inaccuracies. Encoders measuring the cart's position and pendulum angle are affected by quantization, limited resolution, drift, and bias. External factors like electrical interference and sensor nonlinearities further exacerbate these inaccuracies.

Modeling approximations amplify both types of noise. Linearization errors inflate process noise when the system deviates from the operating point, while assumptions like neglecting actuator imperfections increase uncertainty. For measurement noise, idealized assumptions, such as Gaussian zero-mean noise and diagonal covariance matrices, may overlook real-world factors like drift, bounded errors, and sensor correlations.

3c Linear Kalman Filter Design

The process noise covariance Q represents uncertainty in system dynamics, accounting for unmodeled effects, disturbances, and inaccuracies. Larger Q values reduce reliance on the model, enabling faster adaptation but noisier estimates, while smaller Q assumes accurate dynamics, resulting in smoother but slower responses.

The measurement noise covariance R reflects sensor inaccuracies. Larger R values prioritize predictions, producing smoother estimates but slower adaptation, while smaller R increases reliance on measurements for quicker adaptation, at the cost of amplifying noise effects.

Tuning Q and R balances these influences. A high Q/R ratio emphasizes measurements for faster response but noisier results, while a low Q/R ratio favors predictions for smoother but slower estimates. Typically, R is first estimated based on sensor noise under stable conditions, and Q is adjusted iteratively to account for disturbances, achieving a balance between responsiveness and stability.

Furthermore, fo the initial state estimate covariance $\hat{P}_{0|0}$ quantifies uncertainties in the system's states: cart position (x), pendulum angle (θ), and the combined velocity term (ξ_3). It is defined as:

$$\hat{P}_{0|0} = \begin{bmatrix} \sigma_x^2 & 0 & 0 \\ 0 & \sigma_\theta^2 & 0 \\ 0 & 0 & \sigma_{\xi_3}^2 \end{bmatrix} \tag{31}$$

where $\sigma_x^2 = 1 \times 10^{-4}\,\mathrm{m}^2$, representing a cart position uncertainty of approximately 1 cm. For the pendulum angle, $\sigma_\theta^2 = 1 \times 10^{-6}\,\mathrm{rad}^2$ corresponds to an encoder precision of 0.1°. The combined velocity term has $\sigma_{\xi_3}^2 = 1 \times 10^{-6}\,(\mathrm{m/s})^2$, reflecting reasonable uncertainty in initial velocities.

Determining cross-covariances between states is complex, so for simplicity, they are assumed to be zero across all matrices.

Consequently, the Kalman filter's performance under varying Q and R (Figures 2, 3, and 4) demonstrates key trade-offs. To scale Q and R, a scale factor of 10000 was used. Larger Q enhances responsiveness, enabling the filter to closely track dynamics, but amplifies noise-induced oscillations. Larger R smooths estimates by prioritizing predictions, at the cost of lag during rapid changes. Conversely, smaller Q overly smooths estimates, increasing lag due to excessive reliance on model predictions.

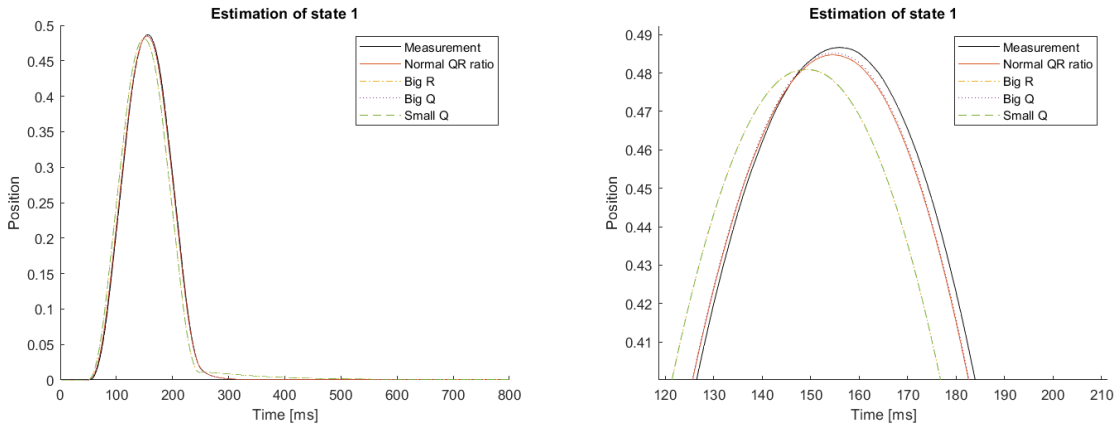
The measurement noise covariance R is derived from sensor characteristics. For x, the spread of measurements between motors A and B in Q-RoboticsCenter indicates a variance of $R_{xx} = 10^{-4}\,\mathrm{m}^2$. For θ, perturbations in the pendulum's equilibrium yield a variance of $R_{\theta\theta} = 10^{-6}\,\mathrm{rad}^2$, resulting in the measurement noise matrix:

$$R = \mathrm{diag}(10^{-4}, 10^{-6})$$

The process noise covariance Q is tuned iteratively, with:

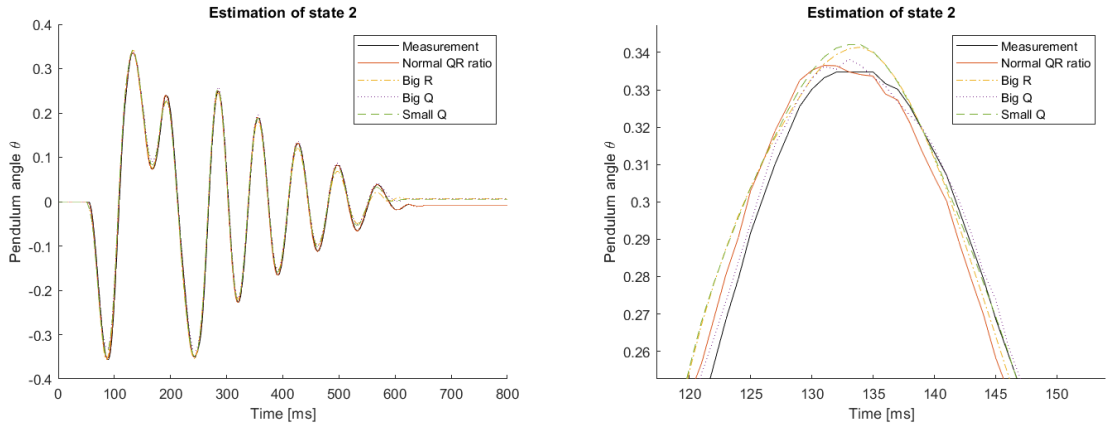
$$Q = \mathrm{diag}(10^{-6}, 10^{-4}, 10^{-2})$$

representing process noise for position ($10^{-6}\,\mathrm{m}^2$), angle ($10^{-4}\,\mathrm{rad}^2$), and velocity ($10^{-2}\,(\mathrm{m/s})^2$). These values ensure a balance between smoothness and responsiveness.



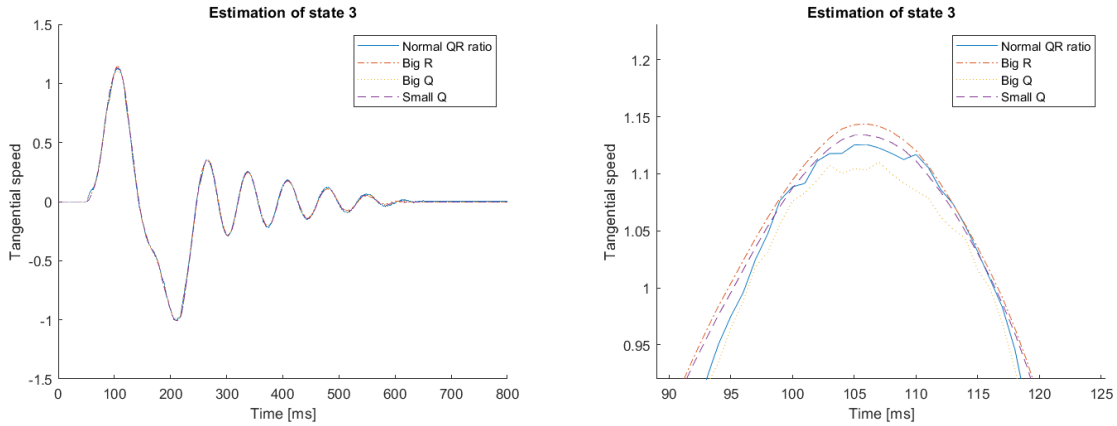
(a) Overview (b) Zoomed

Figure 2: Evolution of State 1



(a) Overview (b) Zoomed

Figure 3: Evolution of State 2



(a) Overview (b) Zoomed

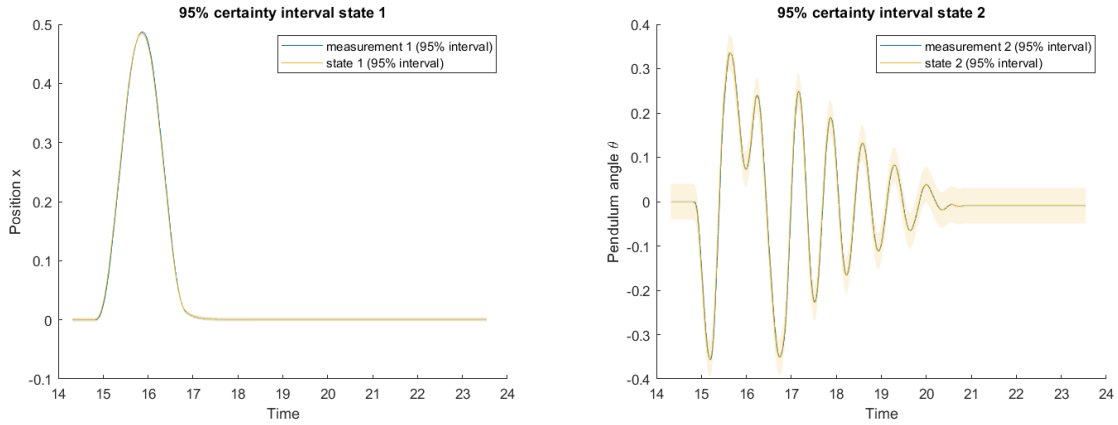
Figure 4: Evolution of State 3

3d Effect of Wheel Encoder on Kalman Filter State Estimation and Uncertainty

Figures 5 present the cart's position (x) and pendulum angle (θ) along with their 95% confidence intervals and corresponding measurements. For the cart position (Figure 5a), the estimator closely tracks the measurements, and the narrow confidence interval indicates high certainty in the estimate.

In contrast, the pendulum angle (θ) (Figure 5b) shows a larger confidence interval, reflecting higher uncertainty due to larger measurement noise (R_θ) or less direct observability of θ. Despite this, the estimator effectively tracks the measurements, and the confidence intervals stabilize over time as the filter processes more data.

This aligns with Kalman filter theory: blending measurements with model predictions reduces uncertainty. Differences in confidence interval sizes arise from variations in initial uncertainty ($\hat{P}_{0|0}$), process noise (Q), and measurement noise (R), with the cart position benefiting from higher measurement accuracy.



(a) State 1 (b) State 2

Figure 5: Measurements of x and θ with 95 % confidence interval on top of the corresponding state estimates with their 95% confidence interval

When the wheel encoder is omitted, the linearized measurement equation becomes:

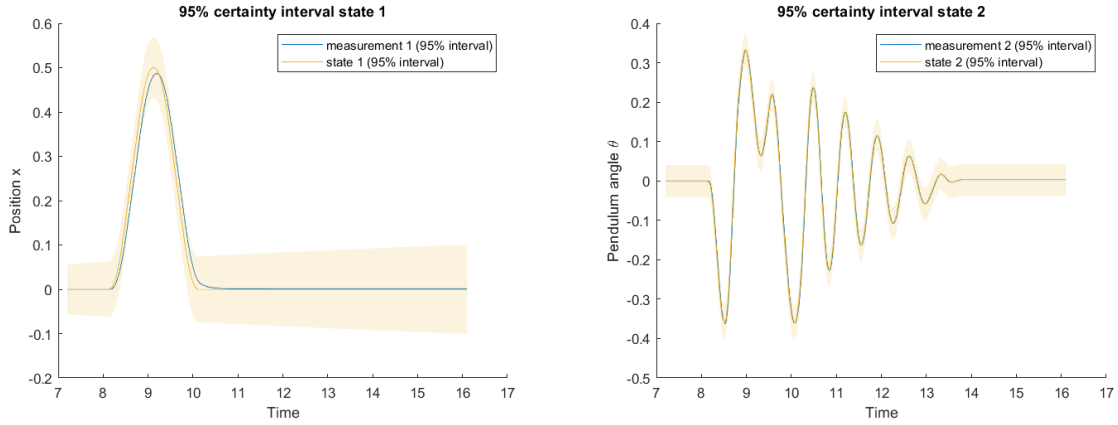
$$y[k+1] = C\xi[k] + Du[k] \tag{32}$$

where:

$$C = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}, \quad D = 0 \tag{33}$$

This modification removes the cart position (x) from the measurements, leaving only the pendulum angle (θ) as the observable state. As a result, the Kalman filter is unable to correct the prediction for x, leading to increasing uncertainty and an expanding confidence interval over time, as shown in Figure 6a. In contrast, the pendulum angle (θ) remains well-estimated, with stable uncertainty, since it is still directly measured (Figure 6b).

The wheel encoder plays a critical role in maintaining accurate estimation of the cart position by bounding its uncertainty and reducing dependence on model predictions. Without it, the confidence interval for x grows unbounded, significantly compromising the system's performance.



(a) State 1 without Encoder (b) State 2 without Encoder

Figure 6: Measurements of x and θ with 95 % confidence interval on top of the corresponding state estimates with their 95% confidence interval without Encoder

4 LQR State Feedback Design and Implementation

4a Feedforward Gain for Zero Steady-State Error

The block diagram is depicted in Figure 7. Since D is equal to zero, that branch is left out. The feedforward controller N_p is clearly illustrated in the diagram.

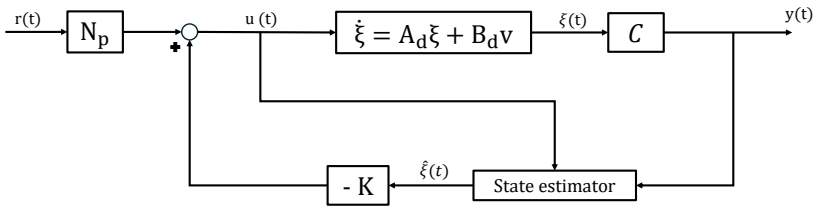


Figure 7: Block diagram

To ensure zero steady-state error for a step reference in the desired pendulum position, the feedforward gain N_p is derived using the system's transfer function. This approach is grounded in a state-space representation and control configuration based on the block diagram framework.

The system's transfer function, $H(z)$, includes components for the cart position and for the pendulum angle. It is expressed as:

$$H(z) = C \cdot (zI - A + BK)^{-1} \cdot B \tag{34}$$

The matrices defining the system dynamics are:

$$A_d = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0.9964 & 0.07398 \\ 0 & -0.09798 & 0.9964 \end{bmatrix}, \quad B_d = \begin{bmatrix} 0.01 \\ -0.07398 \\ 0.003631 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \tag{35}$$

For control design, the Linear Quadratic Regulator (LQR) framework is employed. The weighting matrices are defined as $Q = C^T C \cdot \rho$ and $R = 1$, where ρ is a tunable parameter. Specifically:

$$Q = \begin{bmatrix} \rho & 0 & 0 \\ 0 & \rho & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{36}$$

Using the MATLAB function `dlqr`(A, B, Q, R), the optimal feedback gain K is computed. For different values of ρ, the gains are:

$$K_{\rho=0.1} = \begin{bmatrix} 0.3120 \\ -0.3129 \\ 0.0015 \end{bmatrix} \tag{37}$$

$$K_{\rho=0.5} = \begin{bmatrix} 0.6864 \\ -0.6868 \\ 0.0226 \end{bmatrix} \tag{38}$$

$$K_{\rho=1} = \begin{bmatrix} 0.9590 \\ -0.9577 \\ 0.0570 \end{bmatrix} \tag{39}$$

$$K_{\rho=5} = \begin{bmatrix} 2.0395 \\ -2.0111 \\ 0.2999 \end{bmatrix} \tag{40}$$

At steady state, the system stabilizes with the pendulum at rest ($\theta = 0°$), and the pendulum's position coincides with the cart's position. To eliminate steady-state error under a step input, the transfer function $H(z)$ is evaluated at $z = 1$. The feedforward gain N_p is computed as:

$$N_p = \frac{1}{H(1)} \tag{41}$$

The numerical results for N_p demonstrate how the tuning parameter ρ influences the gain. For different values of ρ, the feedforward gains are:

$$N_{p,\rho=0.1} = 0.312, \quad N_{p,\rho=0.5} = 0.6864, \quad N_{p,\rho=1} = 0.9590, \quad N_{p,\rho=5} = 2.0395 \tag{42}$$

These results show that N_p increases with ρ, emphasizing the trade-off between control effort and steady-state tracking accuracy.

4b Controller Implementation and Q/R Trade-off Analysis

In the LQR design, Q and R play a critical role in balancing performance and control effort. The cost function minimized by the LQR is a trade-off between penalizing deviations in system states, weighted by Q, and penalizing the control effort, weighted by R.

Here, Q is structured as $Q = \rho C_z^T C_z$, where C_z extracts the relevant states—such as the pendulum mass position and velocity. This results in a matrix:

$$Q = \begin{bmatrix} \rho & 0 & 0 \\ 0 & \rho & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{43}$$

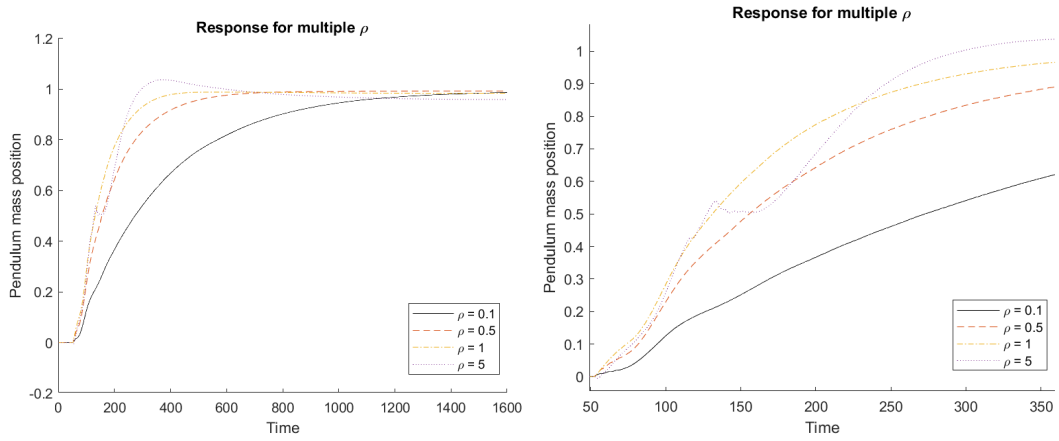
with dimensions matching the number of system states. The tunable parameter ρ determines how heavily deviations in these states are penalized. Larger values of ρ prioritize accurate state regulation, particularly for the pendulum's position and velocity.

The control effort penalty R is set as a scalar $R = 1$, which simplifies the tuning process and ensures consistent weighting of the control input. Its dimension matches the number of control inputs, ensuring compatibility with the system model.

In addition, the units of each row in the Q and R matrices are defined as the square of the units associated with their respective states or measurements.

In Figure 8, the step response of the pendulum mass position is plotted for the different values for ρ. It is clear that a higher value for ρ ensures a faster convergence to 1m. However, when ρ is too high (see $\rho = 5$), the system behaves rather choppy. The pendulum mass position also first surpasses the destination, after which it returns too a little under 1m. $\rho = 1$ seems the best option due to its fast convergence to almost exactly 1m.

The actuator signal, being the desired cart velocity, is shown in Figure 9. The previous reasoning of the choppy behavior for large values for ρ is clearly visible here. A high value of ρ requires high acceleration, which is not always possible because of the voltage constriction of 12 volts.



(a) General (b) Zoomed

Figure 8: Step response of pendulum mass position for multiple ρ

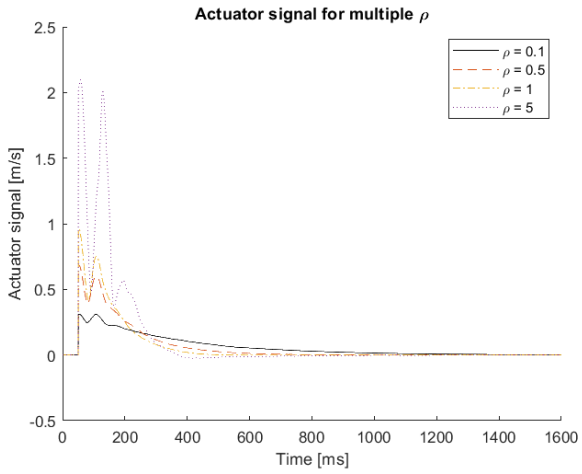


Figure 9: Actuator signal for multiple ρ