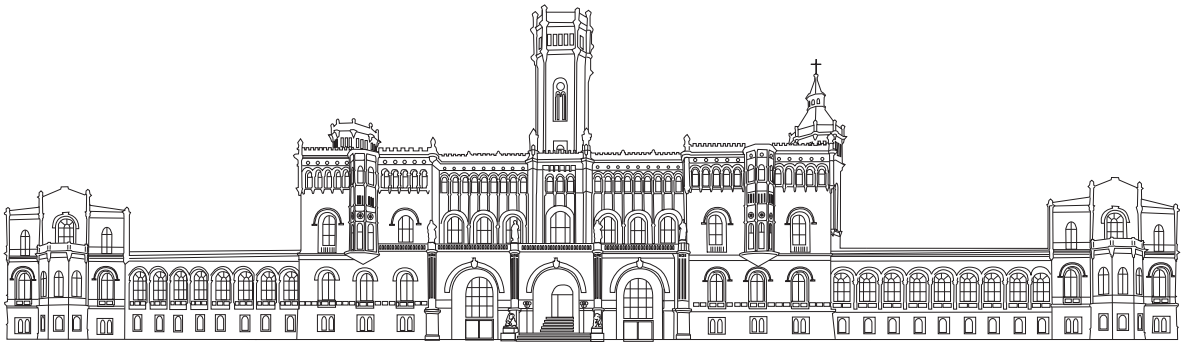
LEIBNIZ UNIVERSITÄT HANNOVER

INSTITUT OF AUTOMATIC CONTROL

STUDY ACHIEVEMENT WINTER TERM 25/26:
DATA AND LEARNING-BASED CONTROL



REPORT BY EMMANUEL JREIJE
M.SC. IN MECHATRONICS & ROBOTICS
REGISTRATION NR. 10040381

28.01.2026

Data- and Learning-Based Control – Study Achievement Winter term 25/26

Number of problems: 3, Maximal number of points: 36

Consultation Hours:
January 8, 2026, 1:00 pm – 2:30 pm
January 15, 2026, 1:00 pm – 2:30 pm
January 22, 2026, 1:00 pm – 2:30 pm
January 29, 2026, 1:00 pm – 2:30 pm
IRT seminar room, A247, building 3404

Submission Deadline: February 5, 2025, 6pm
Please upload a short report (in one pdf document) which contains this *filled* cover sheet, all the required figures, the answer to all questions, and your Matlab code (in a single .zip file) to the Stud.IP group **Labor: Data- and Learning-Based Control** in the folder `Final submissions`. Name the files in the following way `matriculation_number-surname-first_name` and choose the option *Selbst verfasste Werke*.

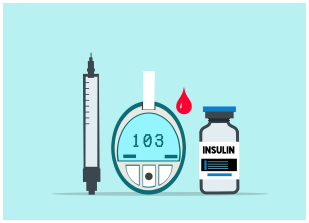
Name: Emmanuel Jreije

Given Name:

Matr.-No.: 10040381

Cousre of Study: Master of Science in Mechatronics & Robotics Engineeing

Problems



The human body is equipped with several feedback loops, such as, e.g., the pituitary-thyroid feedback loop and the glucose-insulin feedback loop. In the latter, insulin is released whenever the glucose levels rise. This leads to a higher conversion rate of glucose into glycogen and ultimately into decreased glucose levels. If the glucose levels decrease, more glycogen (which acts as a reservoir for glucose) is released and converted to glucose. Around 6 million people in Germany are affected by diabetes type I and II. Type I diabetes is particularly dangerous as an autoimmune response destroys insulin-producing cells meaning that the pancreas is unable to produce insulin. Symptoms include regular urination, fatigue, and weight loss. To prevent an insulin shortage in the human body, a common therapy is to inject insulin externally using specific pumps, which allows to regulate blood sugar levels.

In this study achievement, we consider a simplified linear system representing the glucose-insulin feedback loop. The objective is to control the blood glucose level of a patient with type I diabetes. The control inputs of the model correspond to the insulin injection and the intake of carbohydrates. The state is given by

$$x^\top = \begin{bmatrix} g & I_1 & I_2 & D_1 & D_2 \end{bmatrix} \tag{1}$$

where g denotes the blood glucose concentration, I_1, I_2 the insulin concentrations in two compartments and D_1, D_2 the carbohydrate concentrations in two compartments. The system has one output which corresponds to the blood glucose level. The system is sampled with a sampling time of $T = 10$ min.

Problem 1 (6 points)

The objective of this task is to apply Willems' fundamental lemma to simulate a trajectory of the controllable system. The data needed for this problem is saved in the file `data_problem1.mat`. In this file, you have three

Contents

List of Figures

List of Tables

1 Problem 1

1.1 a

Based on the plotted control input sequences presented in Figure 1.1, $u^{(1)}_{\text{off},1}$ and $u^{(2)}_{\text{off},1}$ stay constant at the value 1. The control input sequences $u^{(1)}_{\text{off},2}$ and $u^{(2)}_{\text{off},2}$ are periodic. The control input sequences $u^{(1)}_{\text{off},3}$ and $u^{(2)}_{\text{off},3}$ show random signal. To form a full row rank Hankel matrix, the input signal must have enough independent variations to excite the system in all required modes. A simple control input sequence leads to a rank-deficient Hankel matrix. Considering this, the first two sequences, $u^{(1)}_{\text{off},1}$ and $u^{(2)}_{\text{off},1}$, are too simple and contain no variation, so they are not persistently exciting. The third and fourth sequences, $u^{(1)}_{\text{off},2}$ and $u^{(2)}_{\text{off},2}$, are sine and cosine signals, but they also fail to be persistently exciting because they use only one fixed frequency during the measurement time. The last two sequences, $u^{(1)}_{\text{off},3}$ and $u^{(2)}_{\text{off},3}$, show rich variations and are expected to be persistently exciting of a sufficiently high order.

1.2 b

To check whether an input signal is persistently exciting, we build a block Hankel matrix of order L from the input data. For an m-dimensional input, this matrix has mL rows. The input is persistently exciting of order L if and only if the Hankel matrix has full row rank. The function `is_persistently_exciting` transposes the input ($m \times N$) into the format required by the Hankel builder ($N \times m$), constructs the Hankel matrix, computes its rank, and checks whether this rank equals the number of rows. Using this test, only $u_{\text{off},3}$ is persistently exciting of order $L + n = 77$. Therefore, only the data pair $(u_{\text{off},3}, y_{\text{off},3})$ can be used in Problem 1(c).

1.3 c

Since $u_{\text{off},3}$ is persistently exciting of order $L + n = 77$, the offline data pair $(u_{\text{off},3},\ y_{\text{off},3})$ contains enough information to generate new trajectories of length L. In this step we use these data to simulate the output corresponding to the online input u_{sim}, without using a system model.

First, block Hankel matrices of depth $T_{\text{ini}} + L$ are constructed from the offline inputs and outputs. These matrices are then split into a past part of length $T_{\text{ini}} = n$ and a future part of

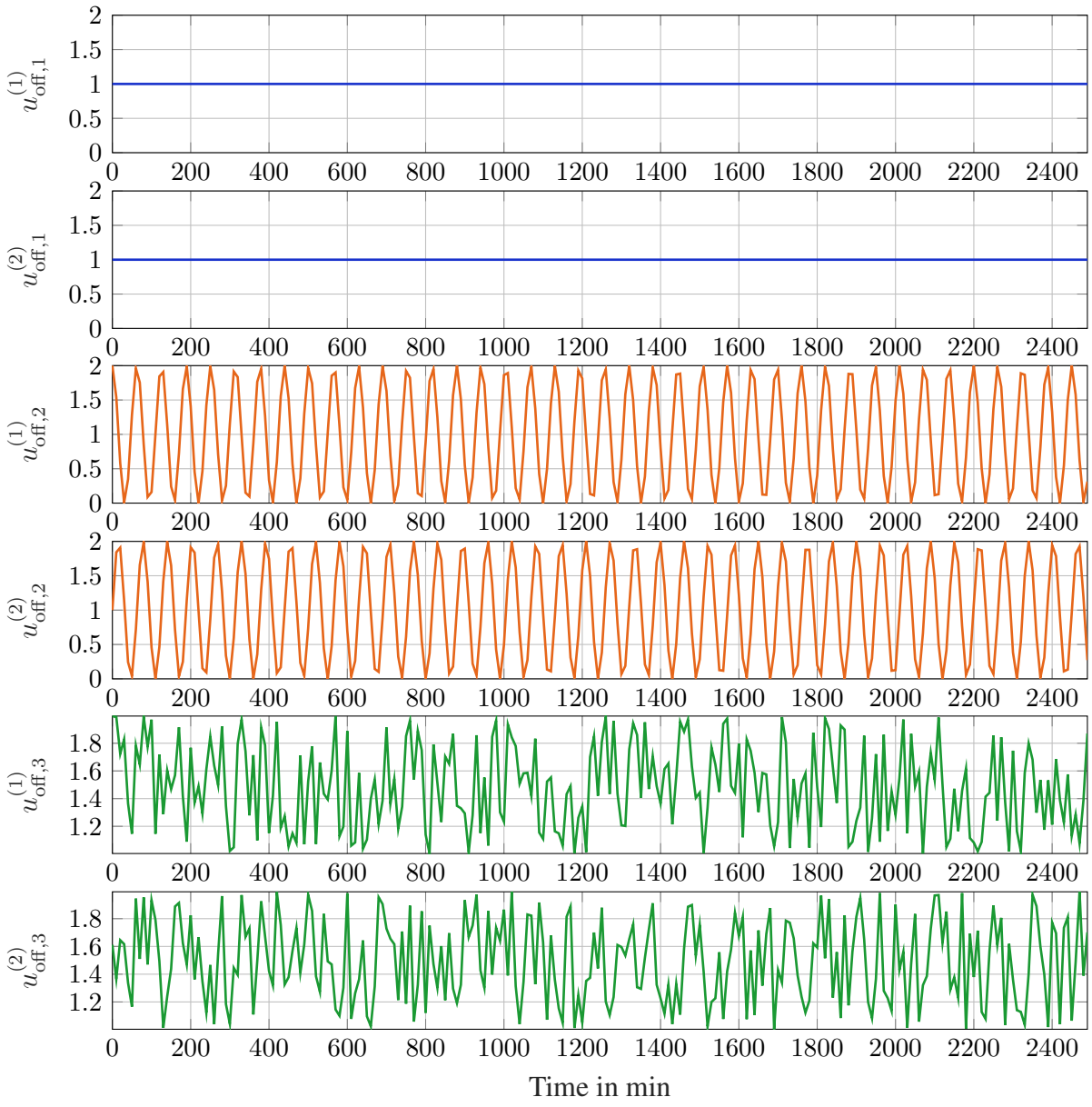


Figure 1.1: Offline control input sequences.

length L. We impose an initial trajectory with zero inputs and outputs and enforce the future inputs to match the first L samples of u_{sim}. This leads to the linear equation

$$\begin{bmatrix} U_p \\ Y_p \\ U_f \end{bmatrix} g = \begin{bmatrix} u_{\text{ini}} \\ y_{\text{ini}} \\ u_f \end{bmatrix},$$

which is solved for the coefficient vector g. The predicted future outputs follow directly from the offline output Hankel matrix as $\hat{y} = Y_f g$. Finally, the obtained trajectory $\hat{y}$ is compared with the given reference y_{sim} for the same input u_{sim}.

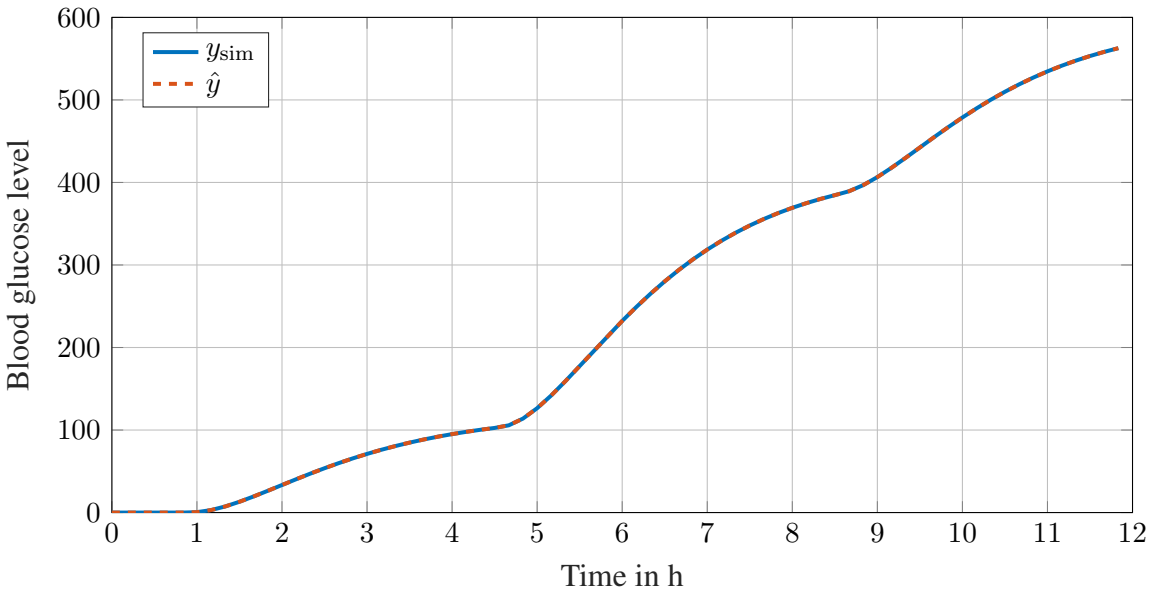


Figure 1.2: Simulation of the blood glucose levels based on the given trajectory using Willems' lemma

The similarity between the reconstructed trajectory $\hat{y}$ and the reference trajectory y_{sim} is expected in this setting. The offline input $u_{\text{off},3}$ is persistently exciting of order 77, and the underlying system used to generate all data is linear. Under these conditions, Willems' fundamental lemma guarantees that any trajectory of length $T_{\text{ini}} + L$ produced by the system can be exactly reconstructed from the offline data by a suitable linear combination of Hankel matrix columns. Since the assumptions are perfectly satisfied here, the reconstructed output $\hat{y}$ matches the true output y_{sim} almost exactly.

2 Problem 2

2.1 a

The matrices $\boldsymbol{A}_{\text{id}}$ and $\boldsymbol{B}_{\text{id}}$ shown above are the identified system matrices obtained from Dataset 2 (and identically from Dataset 3). Since these datasets are informative for system identification, the equation

$$\boldsymbol{X}_+ = [\boldsymbol{A}\ \boldsymbol{B}]\,[\boldsymbol{X}_-; \boldsymbol{U}_-]$$

admits a unique solution. The resulting matrices $\boldsymbol{A}_{\text{id}}$ and $\boldsymbol{B}_{\text{id}}$ therefore represent the best data-driven estimate of the true system dynamics

$$x_{k+1} = \boldsymbol{A}x_k + \boldsymbol{B}u_k.$$

Their structure closely matches the expected linear glucose–insulin model.

$$\boldsymbol{A}_{\text{id}} = \begin{bmatrix} 1.0000 & -18.4542 & -3.3219 & 2.2083 & 0.1722 \\ 0 & 0.6807 & 0.2618 & 0 & 0 \\ 0 & 0 & 0.6807 & 0 & 0 \\ 0 & 0 & 0 & 0.8519 & 0.1365 \\ 0 & 0 & 0 & 0 & 0.8519 \end{bmatrix}, \qquad \boldsymbol{B}_{\text{id}} = \begin{bmatrix} -0.4539 & 0.0094 \\ 0.0575 & 0 \\ 0.3193 & 0 \\ 0 & 0.0115 \\ 0 & 0.1481 \end{bmatrix}.$$

The matrix $A_{\text{err},2}$ represents the difference between the identified matrix $A_{\text{id},2}$ and the true system matrix $\boldsymbol{A}$. All entries are on the order of 10^{-4}, which confirms that Dataset 2 provides an accurate reconstruction of the state transition matrix. The small magnitude of the error indicates that the identification is numerically very close to the true dynamics.

$$\boldsymbol{A}_{\text{err},2} = \boldsymbol{A}_{\text{id},2} - \boldsymbol{A} = 10^{-4} \begin{bmatrix} 0 & 0.1520 & -0.4819 & 0.2657 & 0.2506 \\ 0 & 0.1240 & 0.1246 & 0 & 0 \\ 0 & 0 & 0.1240 & 0 & 0 \\ 0 & 0 & 0 & 0.2532 & 0.2649 \\ 0 & 0 & 0 & 0 & 0.2532 \end{bmatrix},$$

The matrix $\boldsymbol{B}_{\text{err},2}$ contains the identification error for the input matrix $\boldsymbol{B}$ when using Dataset 2.

As with $\boldsymbol{A}_{\text{err},2}$, the error levels remain in the range of 10^{-4}, showing that the estimated influence of the inputs on the state evolution matches the true model very well. This confirms that Dataset 2 is not only informative but also yields a highly accurate estimate of the input dynamics.

$$\boldsymbol{B}_{\text{err},2} = \boldsymbol{B}_{\text{id},2} - \boldsymbol{B} = 10^{-4} \begin{bmatrix} 0.3299 & 0.4837 \\ -0.2486 & 0 \\ -0.1240 & 0 \\ 0 & -0.0181 \\ 0 & -0.2532 \end{bmatrix}.$$

The matrix $A_{\text{err},3}$ shows the identification error obtained from Dataset 3. The error values are again on the order of 10^{-4} and essentially match those obtained from Dataset 2. This indicates that Dataset 3 is equally informative and provides an accurate reconstruction of the state transition matrix $\boldsymbol{A}$.

$$\boldsymbol{A}_{\text{err},3} = 10^{-4} \begin{bmatrix} 0 & 0.1520 & -0.4819 & 0.2657 & 0.2506 \\ 0 & 0.1240 & 0.1246 & 0 & 0 \\ 0 & 0 & 0.1240 & 0 & 0 \\ 0 & 0 & 0 & 0.2532 & 0.2649 \\ 0 & 0 & 0 & 0 & 0.2532 \end{bmatrix},$$

The matrix $\boldsymbol{B}_{\text{err},3}$ shows the deviation between the identified input matrix $\boldsymbol{B}_{\text{id},3}$ and the true input matrix $\boldsymbol{B}$. As with the previous results, the deviations remain extremely small (10^{-4}), confirming that Dataset 3 yields a reliable estimate of the input-to-state dynamics. Both informative datasets therefore lead to almost identical identification quality.

$$\boldsymbol{B}_{\text{err},3} = 10^{-4} \begin{bmatrix} 0.3299 & 0.4837 \\ -0.2486 & 0 \\ -0.1240 & 0 \\ 0 & -0.0181 \\ 0 & -0.2532 \end{bmatrix}.$$

In Problem 2(a) the goal was to determine whether the available datasets are informative for system identification and, if so, to estimate the system matrices $\boldsymbol{A}$ and $\boldsymbol{B}$. For each dataset, we constructed the data matrices

$$\boldsymbol{X}_- = [x_0, \ldots, x_{N-1}], \qquad \boldsymbol{U}_- = [u_0, \ldots, u_{N-1}], \qquad \boldsymbol{X}_+ = [x_1, \ldots, x_N]$$

and tested the rank condition

$$\operatorname{rank} \begin{bmatrix} \boldsymbol{X}_- \\ \boldsymbol{U}_- \end{bmatrix} = n + m.$$

Datasets 2 and 3 satisfy this condition and are therefore informative. For these datasets, we computed the unique least-squares solution

$$\begin{bmatrix} \boldsymbol{A}_{\text{id}} & \boldsymbol{B}_{\text{id}} \end{bmatrix} = \boldsymbol{X}_+ \begin{bmatrix} \boldsymbol{X}_- \\ \boldsymbol{U}_- \end{bmatrix}^\dagger$$

and compared the result with the true system matrices. The corresponding error matrices are on the order of 10^{-4}, showing that the identified model matches the true dynamics very well.

2.2 b

Table 2.1: Informativity results for the three datasets

Dataset	System Identification	Controllability	Stabilizability
Dataset 1	No	Yes	Yes
Dataset 2	Yes	Yes	Yes
Dataset 3	Yes	Yes	Yes

The Table 2.1 summarizes the results of questions a and b, dataset 1 is informative for controllability and stabilizability but not for system identification. Datasets 2 and 3 are informative for all three properties.

2.3 c

Using the LMI-based procedure within the data-informativity framework, the stabilizing feedback gain $\boldsymbol{K}_{\text{stab}}$ obtained from the stabilizable dataset is given by

$$\boldsymbol{K}_{\text{stab}} = \begin{bmatrix} 0.7195 & -17.5961 & -4.6834 & 2.1799 & 0.1292 \\ -0.0569 & 1.4937 & 0.2156 & -1.5700 & -5.8595 \end{bmatrix}$$

2.4 d

The closed-loop system is simulated using a sampling time of $T_s = \frac{1}{6}$ and a horizon of 21 time steps. The initial condition is chosen as

$$x_0 = \begin{bmatrix} 15 & 0 & 0 & 0 & 0 \end{bmatrix}^\top .$$

The resulting control inputs are presented in Figure 2.1.

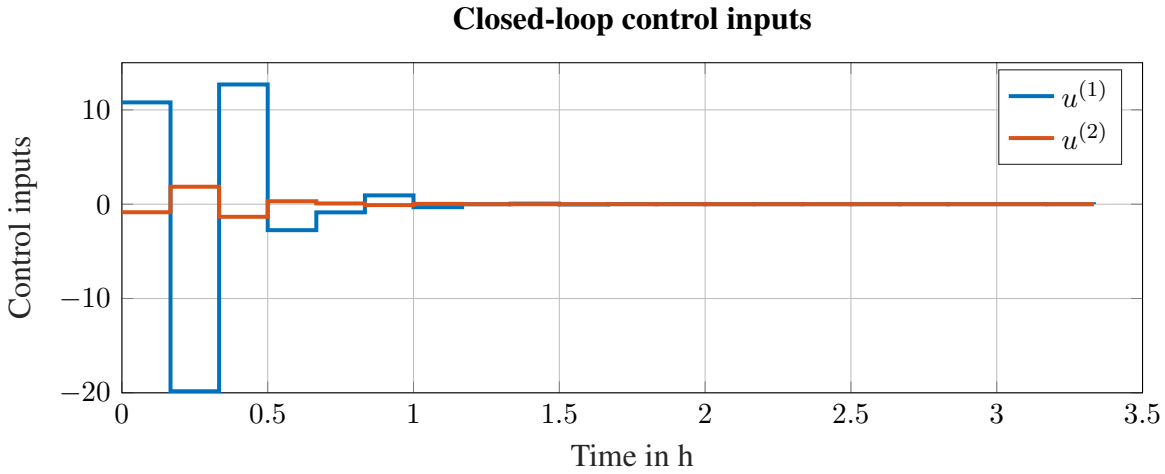


Figure 2.1: Simulation of the control inputs using the closed-loop system.

The resulting blood glucose levels are presented in Figure 2.2.

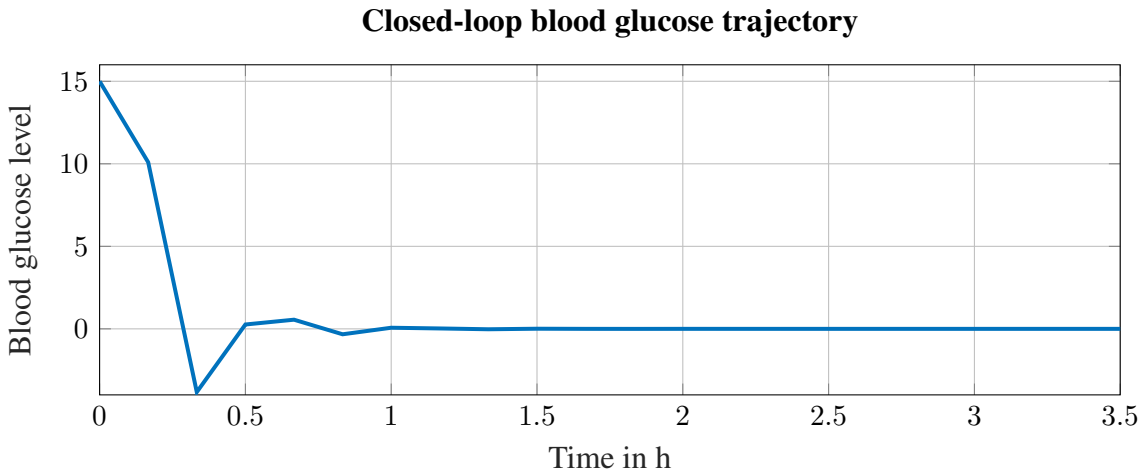


Figure 2.2: Simulation of the blood glucose levels using the closed-loop system.

2.5 e

In the presence of noisy offline state measurements, a stabilizing feedback gain $\boldsymbol{K}_{\text{stabn}}$ is computed using the data-informativity framework, treating the measured data as noise-free. The resulting gain is given by

$$\boldsymbol{K}_{\text{stabn}} = \begin{bmatrix} -0.1108 & -6.4624 & -3.3539 & -0.3046 & -0.4859 \\ -0.4352 & -2.4552 & -1.1037 & -9.3808 & -6.4130 \end{bmatrix}.$$

2.6 f

The closed-loop system is simulated with $T_s = \frac{1}{6}$ for 21 time steps, starting from x_0. The blood glucose levels and control inputs are reported below. It is noticeable that treating noisy data as noise-free leads to an increase in the control effort over time.

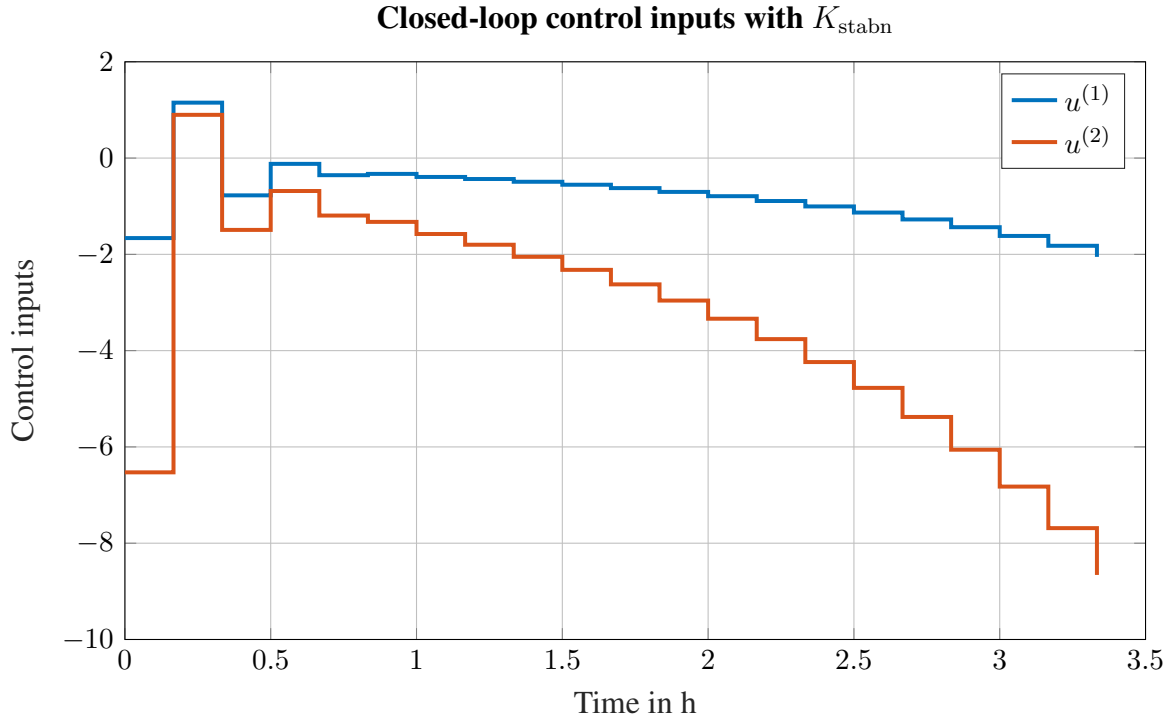


Figure 2.3: Simulation of the control inputs using the closed-loop system.

The continuous increase in the control effort causes the blood glucose level to become unstable, preventing convergence to a steady value, as observed in Figure 2.4.

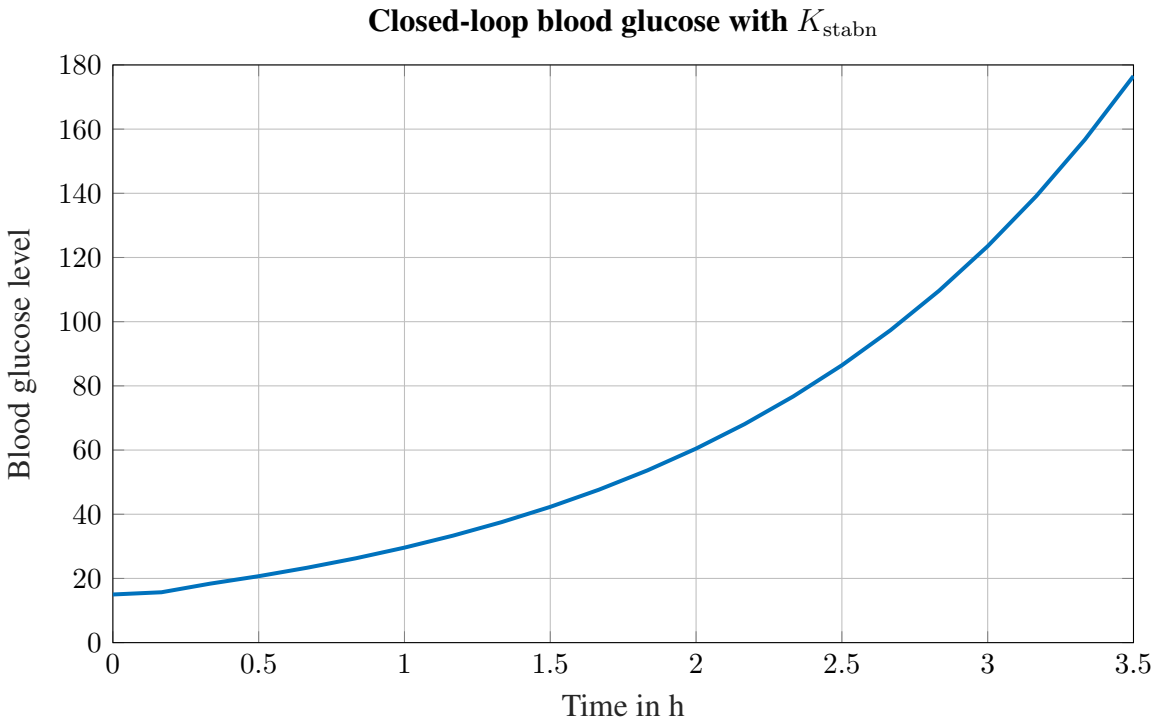


Figure 2.4: Simulation of the blood glucose levels using the closed-loop system.

2.7 g

Using the third dataset, a data-based LQR controller $\boldsymbol{K}_{\text{LQR}}$ is computed within the data informativity framework based on the noise-free offline data with weighting matrices $\boldsymbol{Q} = 2\boldsymbol{I}_{\text{n}}$ and $\boldsymbol{R} = 500\boldsymbol{I}_{\text{m}}$.

$$\boldsymbol{K}_{\text{LQR}} = \begin{bmatrix} 0.0473 & -2.1227 & -1.3040 & 0.3406 & 0.1078 \\ -0.0011 & 0.0535 & 0.0401 & -0.0110 & -0.0081 \end{bmatrix}$$

2.8 h

Figures 2.5 and 2.6 present the closed-loop simulation of the control inputs and the blood glucose level over 144 time steps.

2.9 i

Using dataset 3, the identified matrices are identical to the true system matrices. Therefore, it is expected that the model-based LQR and the data-based LQR yield the same feedback gain.

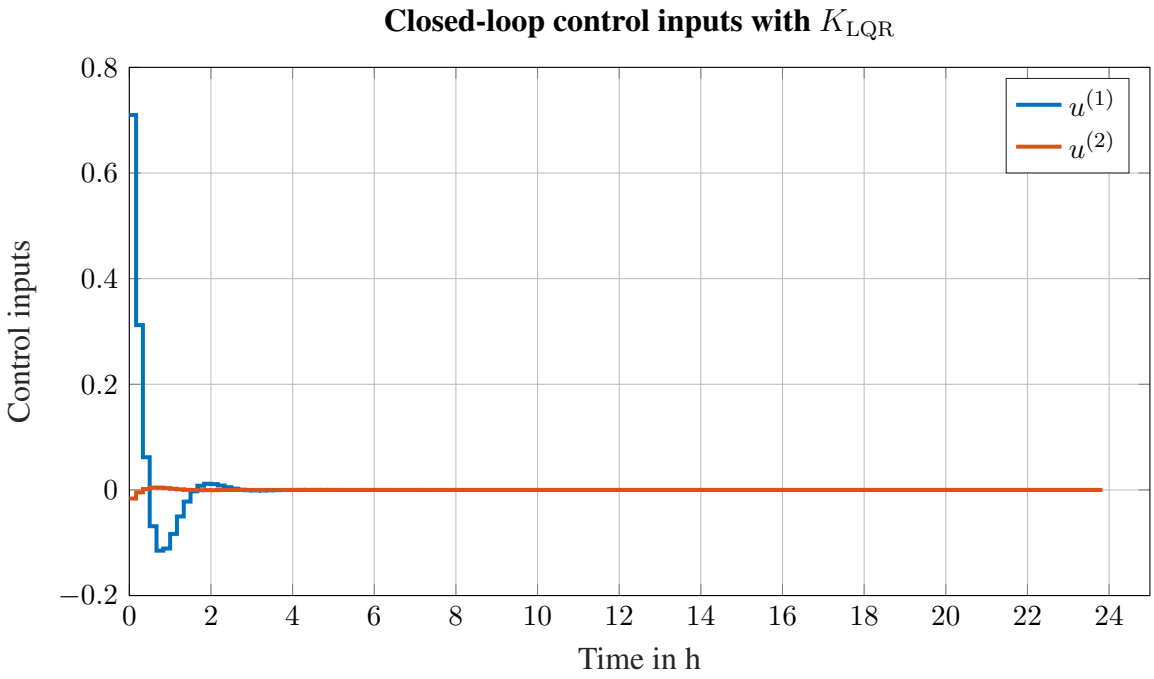


Figure 2.5: Closed-loop simulation of control inputs over 144 time steps using $\boldsymbol{K}_{\mathrm{LQR}}$

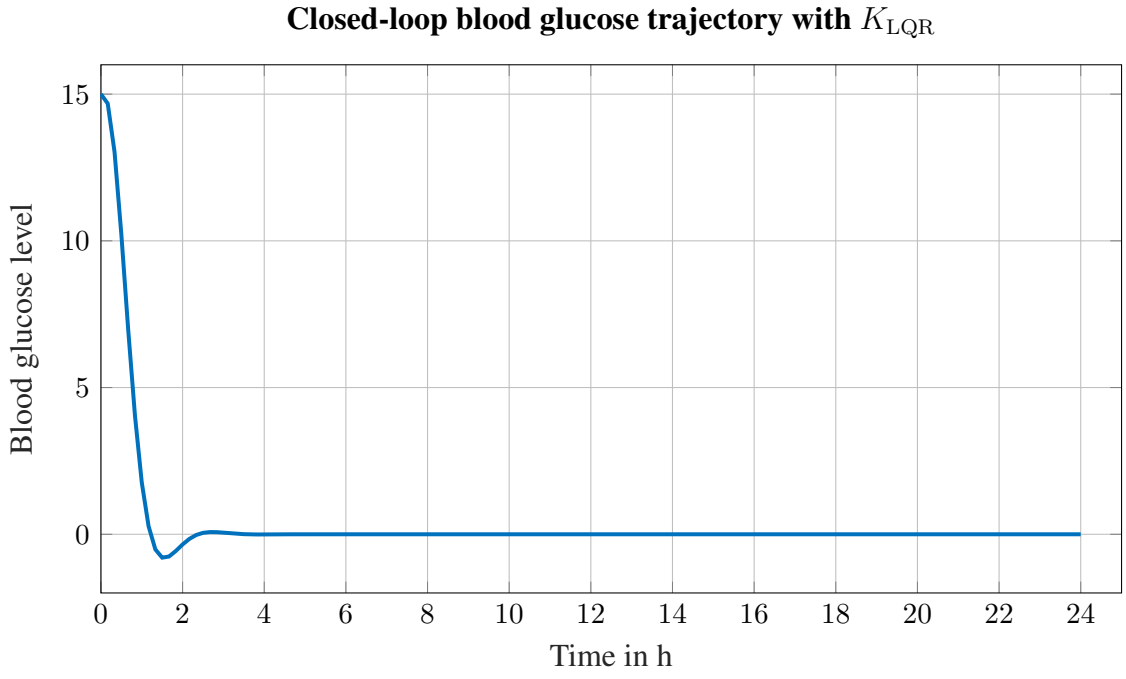


Figure 2.6: Closed-loop simulation of blood glucose levels over 144 time steps using $\boldsymbol{K}_{\mathrm{LQR}}$

Based on this the performance of the model-based and data-based controller is also identical as observed in Figure 2.7 and 2.8.

$$K_{\text{LQR}_{\text{mb}}} = \begin{bmatrix} 0.0473 & -2.1227 & -1.3041 & 0.3406 & 0.1078 \\ -0.0011 & 0.0535 & 0.0401 & -0.0110 & -0.0081 \end{bmatrix}$$

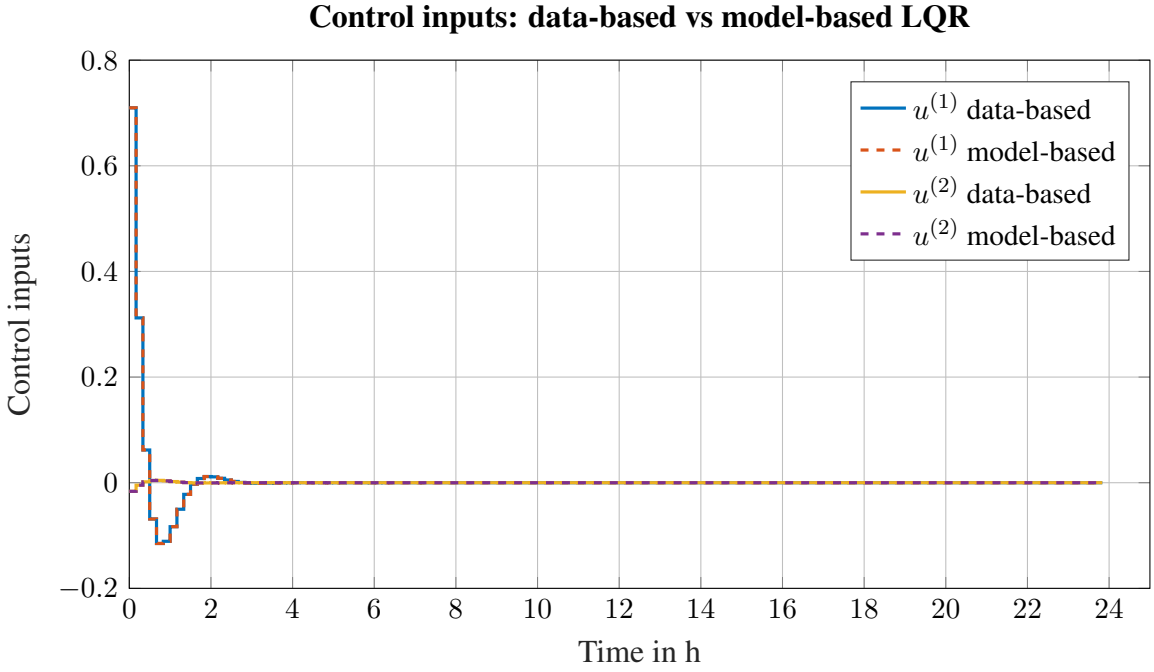


Figure 2.7: A comparison of control inputs over 144 time steps, using data-based versus model-based LQR in a closed-loop simulation.

2.10 j

For the noisy-data LQR gain K_{LQRn}, the closed-loop matrix $\boldsymbol{A}_{\text{true}} + \boldsymbol{B}_{\text{true}}\boldsymbol{K}_{\text{LQRn}}$ has a spectral radius slightly larger than one:

$$\boldsymbol{K}_{\text{LQRn}} = 10^{-6} \begin{bmatrix} -0.1235 & 0.0163 & -0.1674 & 0.0666 & 0.0036 \\ -0.0016 & -0.0036 & -0.0020 & -0.0145 & -0.0790 \end{bmatrix}$$

$$\max_i |\lambda_i| = 1.000002740632.$$

Since discrete-time asymptotic stability requires $|\lambda_i| < 1$ for all eigenvalues, the closed loop is not asymptotically stable. Therefore, the closed-loop simulation with $\boldsymbol{K}_{\text{LQRn}}$ can exhibit a

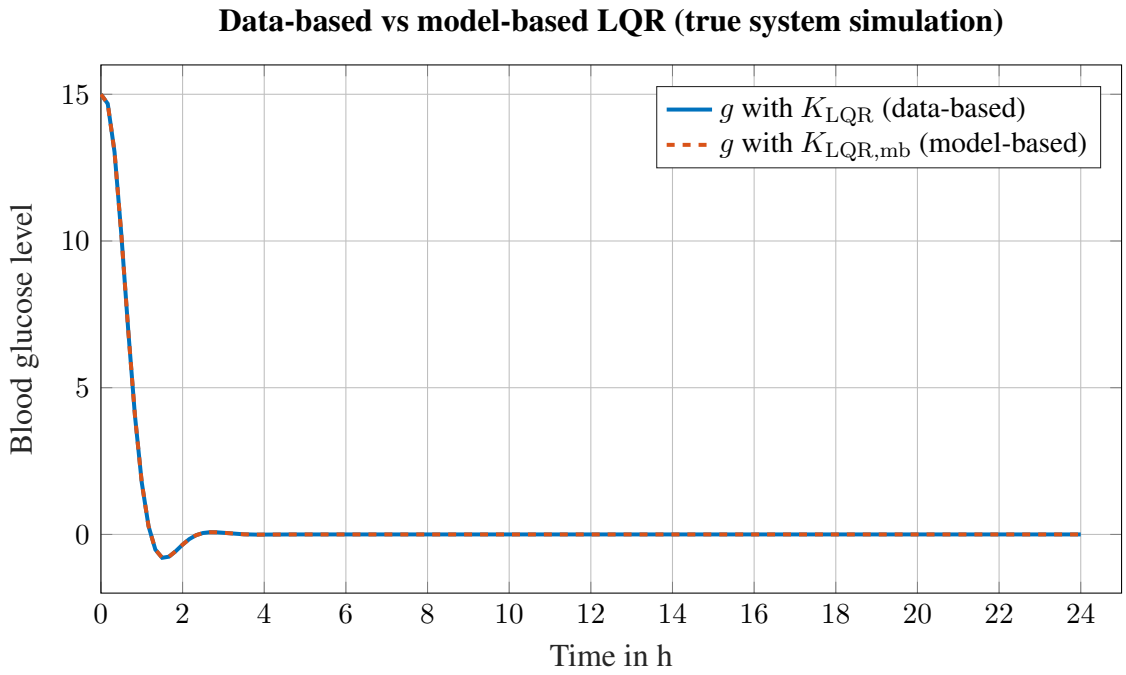


Figure 2.8: A comparison of blood glucose levels over 144 time steps, using data-based versus model-based LQR in a closed-loop simulation.

slow drift over the 144-step horizon, as observed in Figure 2.10.

2.11 k

The closed-loop simulation of the true system using the controller $\boldsymbol{K}_{\mathrm{LQRn}}$ shows that the system is no longer stable. Although the closed-loop response does not diverge rapidly, the blood glucose level does not converge to a steady value and instead exhibits a slow drift over the 144 time steps. This behaviour is consistent with the closed-loop eigenvalue analysis, which indicates a spectral radius slightly larger than one. Therefore, the controller $\boldsymbol{K}_{\mathrm{LQRn}}$ does not stabilise the true system when computed from noisy state data treated as noise-free.

2.12 l

In the last subtasks, the control inputs obtained from noisy data show undesirable behavior. For the noisy stabilizing controller $\boldsymbol{K}_{\mathrm{stabn}}$, the control inputs become relatively large and lead to an unstable glucose trajectory. For the noisy-data LQR controller $\boldsymbol{K}_{\mathrm{LQRn}}$, the control inputs are almost zero, which makes the controller ineffective and results in a slow drift of the blood

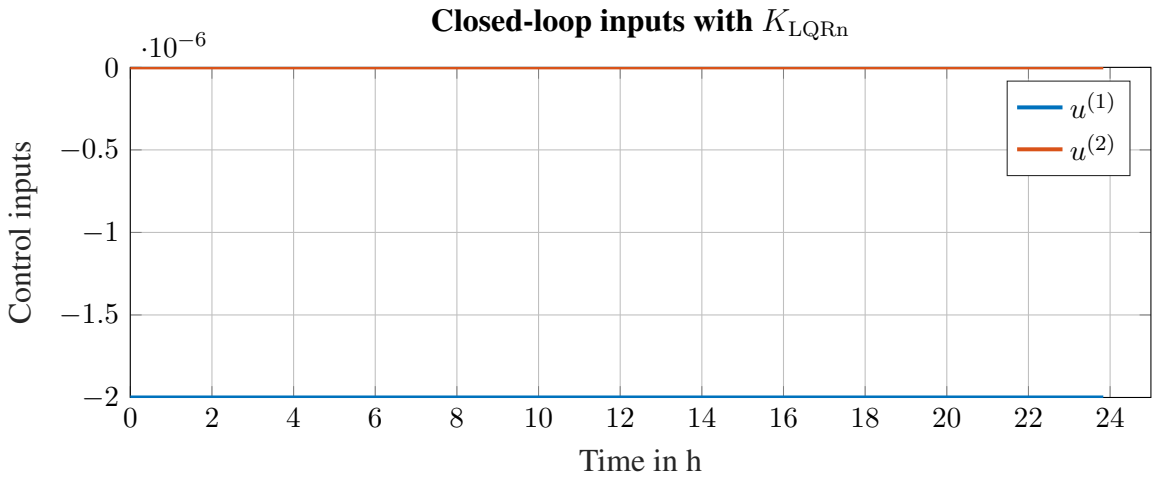


Figure 2.9: Closed-loop simulation of control inputs over 144 time steps using $\boldsymbol{K}_{\mathrm{LQRn}}$

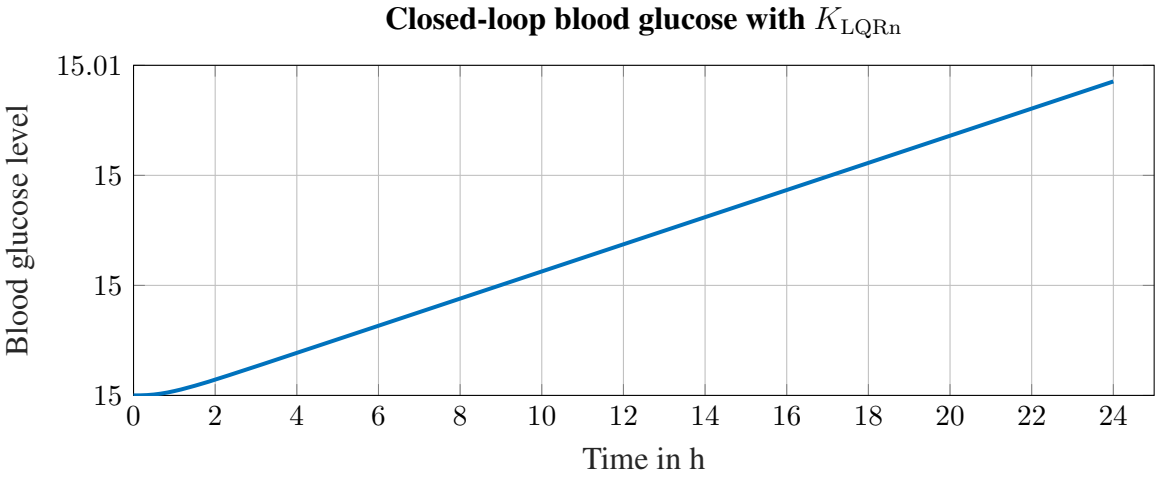


Figure 2.10: Closed-loop simulation of the blood glucose levels over 144 time steps using $\boldsymbol{K}_{\mathrm{LQRn}}$

glucose level.

In both cases, the control inputs are not physically plausible, since real actuators are subject to magnitude and rate limitations and must also produce a meaningful control action. To obtain physically plausible control inputs, input constraints should be explicitly enforced. This can be achieved using constrained control techniques such as Model Predictive Control.

Controller	$g(1)$	$g(\text{end})$	$g(\text{end}) - g(1)$
K_{stab}	15	3.26×10^{-7}	-15
K_{stabn}	15	176.546	161.546
K_{LQR}	15	1.46×10^{-13}	-15
K_{LQRn}	15	15.0057	0.0057075

Table 2.2: Cost function evolution

Controller	$\max \lvert u_1 \rvert$	$\max \lvert \Delta u_1 \rvert$	$\max \lvert u_2 \rvert$	$\max \lvert \Delta u_2 \rvert$
K_{stab}	19.836	32.5288	1.8518	3.18822
K_{stabn}	2.05246	2.81312	8.65909	7.42628
K_{LQR}	0.709773	0.397743	0.0163007	0.0116391
K_{LQRn}	1.85×10^{-6}	5.08×10^{-12}	2.33×10^{-8}	6.40×10^{-14}

Table 2.3: Maximum control inputs and increments

2.13 m

An insulin-only controller was designed by setting the second row of the noise-free stabilizing gain $\boldsymbol{K}_{\text{stab}}$ to zero and denoting the resulting gain by $\boldsymbol{K}_{\text{stab2}}$. In this way, only the insulin input is generated by feedback, while the carbohydrate intake is prescribed by the second component of u_{sim} from Problem 1, which is periodically repeated to cover 144 time steps.

$$\boldsymbol{K}_{\text{stab2}} = \begin{bmatrix} 0.7195 & -17.5960 & -4.6834 & 2.1799 & 0.1292 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

The eigenvalues of the closed-loop matrix $\boldsymbol{A}_{\text{true}} + \boldsymbol{B}_{\text{true}}\boldsymbol{K}_{\text{stab2}}$ are

$$\lambda = \begin{bmatrix} -0.2516 \pm 0.3408\,\mathrm{i},\ 0.0308,\ 0.8519,\ 0.8519 \end{bmatrix}.$$

Hence, $\max_i |\lambda_i| = 0.8519 < 1$, which indicates internal stability of the homogeneous closed-loop system.

In the closed-loop simulation with regular eating, the blood glucose level remains bounded and decreases over the simulation horizon. However, stability is not guaranteed in this setting, since the original data-driven stability certificate applies only to the two-input feedback law $u = \boldsymbol{K}_{\text{stab}}x$ and does not apply after modifying the controller and introducing an external eating signal.

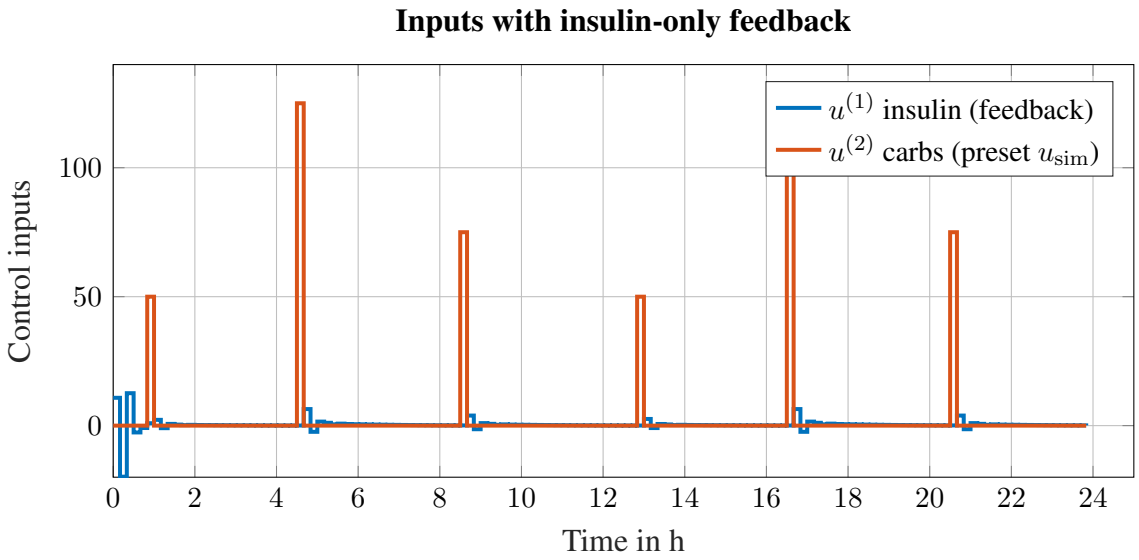


Figure 2.11: Closed-loop simulation of the control inputs over 144 time steps using insulin-only feedback.

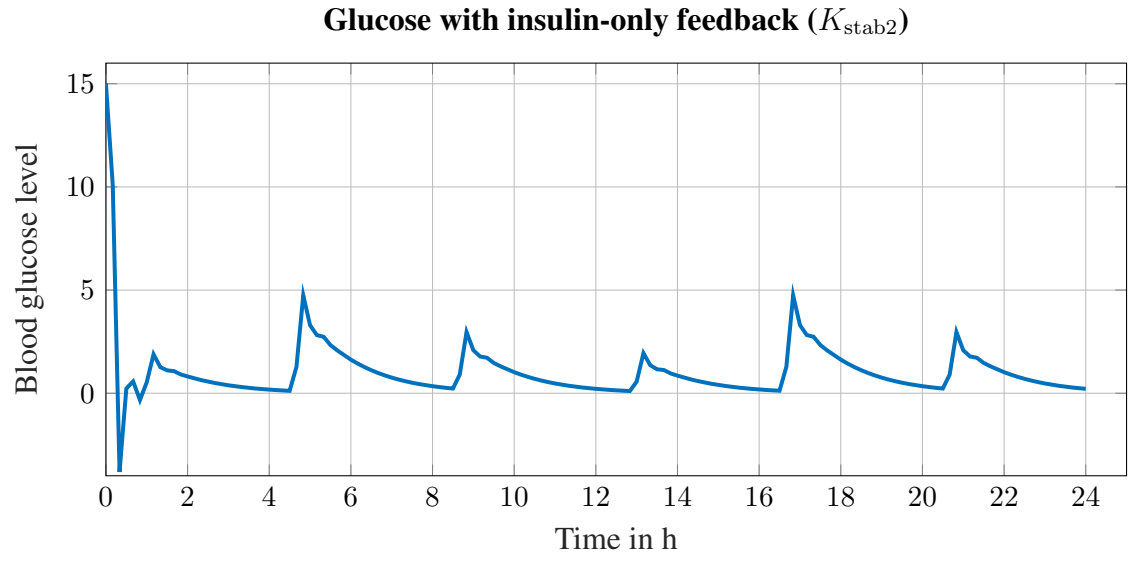


Figure 2.12: Closed-loop simulation of the blood glucose levels over 144 time steps with insulin-only feedback.

3 Problem 3

3.1 a

We consider the nonlinear pendulum system

$$\dot{x}_1(t) = x_2(t), \tag{3.1}$$

$$\dot{x}_2(t) = -\frac{g}{l}\sin(x_1(t)) + u(t), \tag{3.2}$$

The control objective is to stabilize the equilibrium $\boldsymbol{x}_{eq} = [0 \;\; 0]^\top$. Full model knowledge is assumed, and the reference dynamics are given by $\dot{x}_r = 0$.
We choose a linear state feedback gain

$$K = [k_1 \; k_2]$$

such that the eigenvalues of the closed-loop matrix are located at $\lambda_1 = \lambda_2 = -6$.

$$\begin{bmatrix} 0 & 1 \\ -36 & -12 \end{bmatrix}$$

The resulting control law is

$$u(x) = \frac{g}{l}\sin(x_1) - k_1 x_1 - k_2 x_2, \tag{3.3}$$

which exactly cancels the nonlinear term and yields linear closed-loop dynamics.
The results show convergence of the state to the equilibrium, which confirms the correctness of the controller design.
The closed-loop response of the pendulum is simulated using the designed state feedback controller. Figure 3.1 shows the corresponding control input applied to the system. Figure 3.2 shows the evolution of the pendulum angle and angular velocity over time. The pendulum is stabilized at the equilibrium, assuming full model knowledge. The nonlinear gravity term is exactly cancelled, and a linear state feedback controller is applied such that the closed-loop poles are placed at -6. As a result, the angle and angular velocity converge rapidly to zero. The control input exhibits a short transient to compensate the initial deviation and converges to zero once the equilibrium is reached.

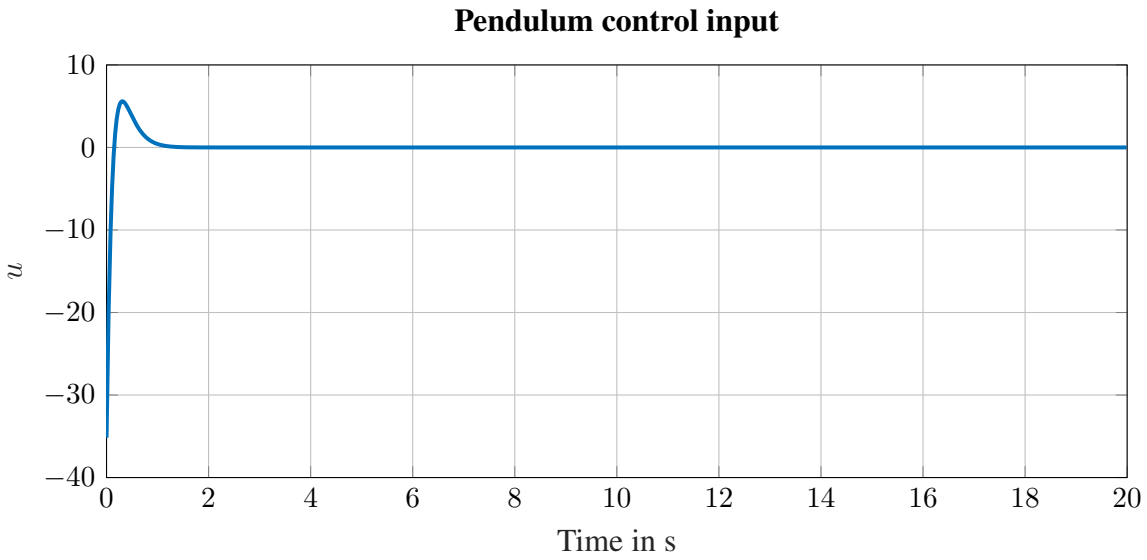


Figure 3.1: The control input applied to the closed-loop response of the pendulum using a state feedback controller

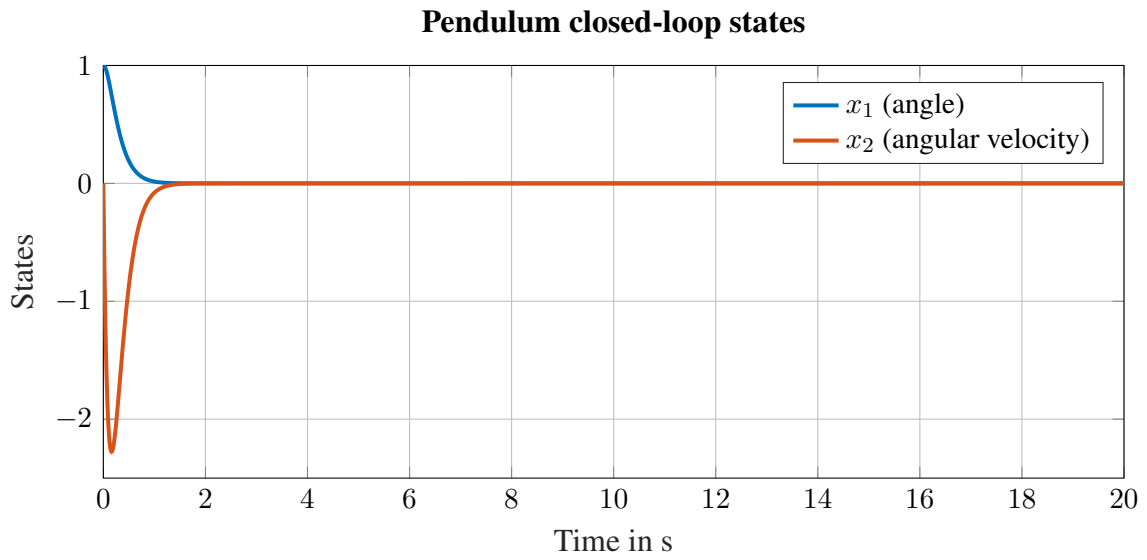


Figure 3.2: The pendulum angle and angular velocity from the simulated closed-loop response of the pendulum using a state feedback controller

3.2 b

The pendulum is controlled to track a reference trajectory generated by a Van der Pol oscillator. Figures 3.3, 3.4, and 3.5 show the tracking performance and the corresponding control input.

After a short transient at the beginning of the simulation, the tracking error remains small.

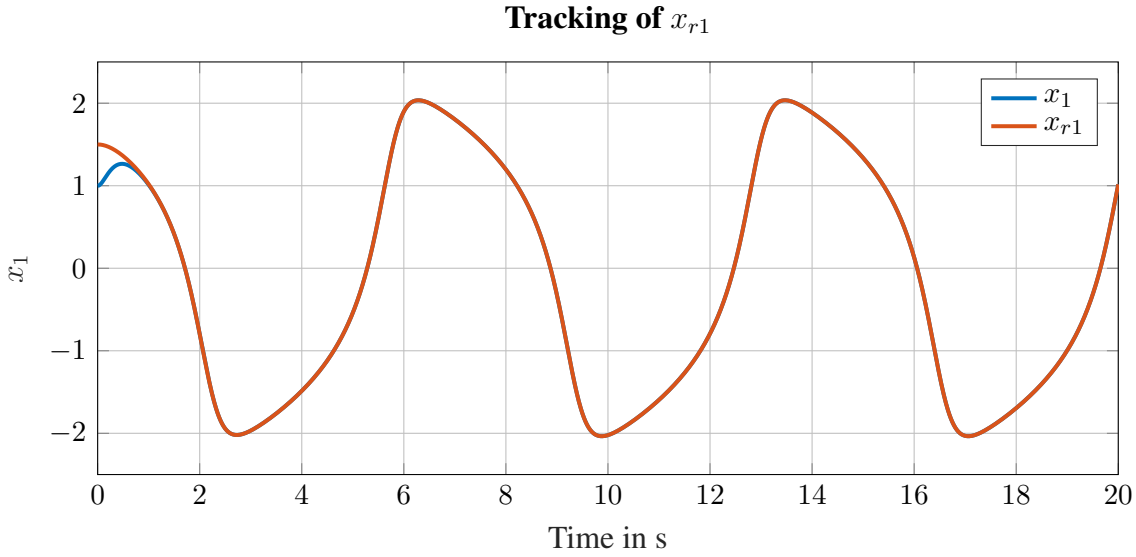


Figure 3.3: Pendulum angle x_1 and reference angle x_{r1} over time.

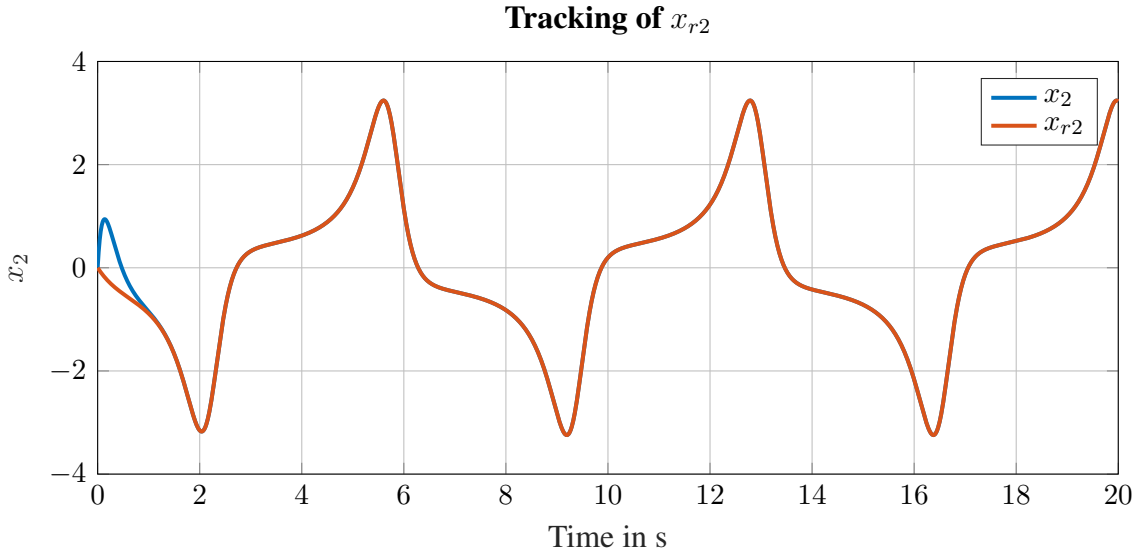


Figure 3.4: Pendulum angular velocity x_2 and reference angular velocity x_{r2} over time.

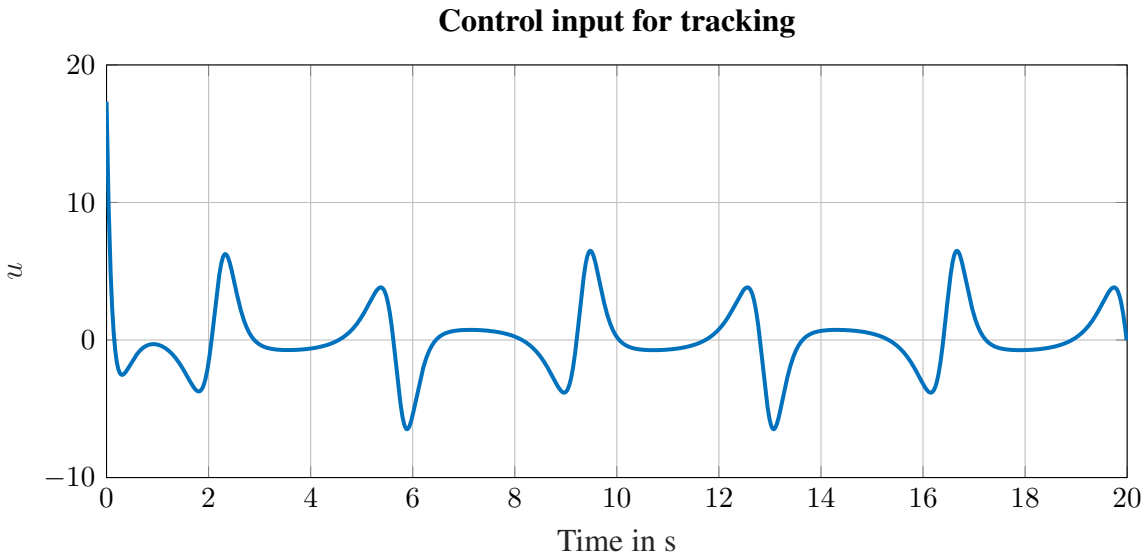


Figure 3.5: Control input u applied to the pendulum during reference tracking.

3.3 c

The pendulum is controlled to track a constant reference setpoint $x_{r1} = 1.5$ using a GP-MRAC scheme with a wrong model. Figures 3.6, 3.7, and 3.8 show the tracking result, the applied control input, and the learned model mismatch.

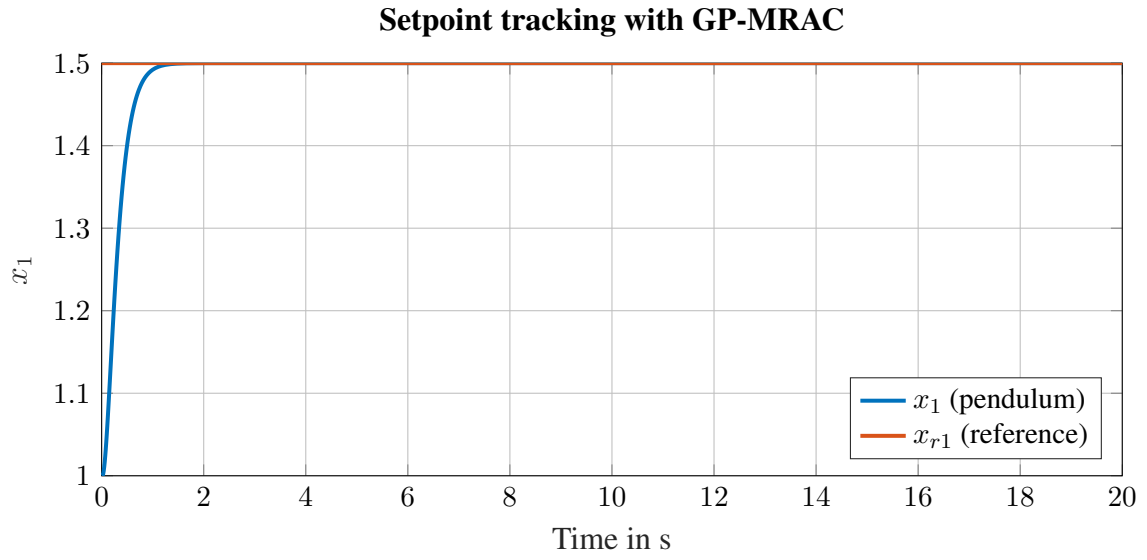


Figure 3.6: Pendulum angle x_1 and reference angle x_{r1} over time.

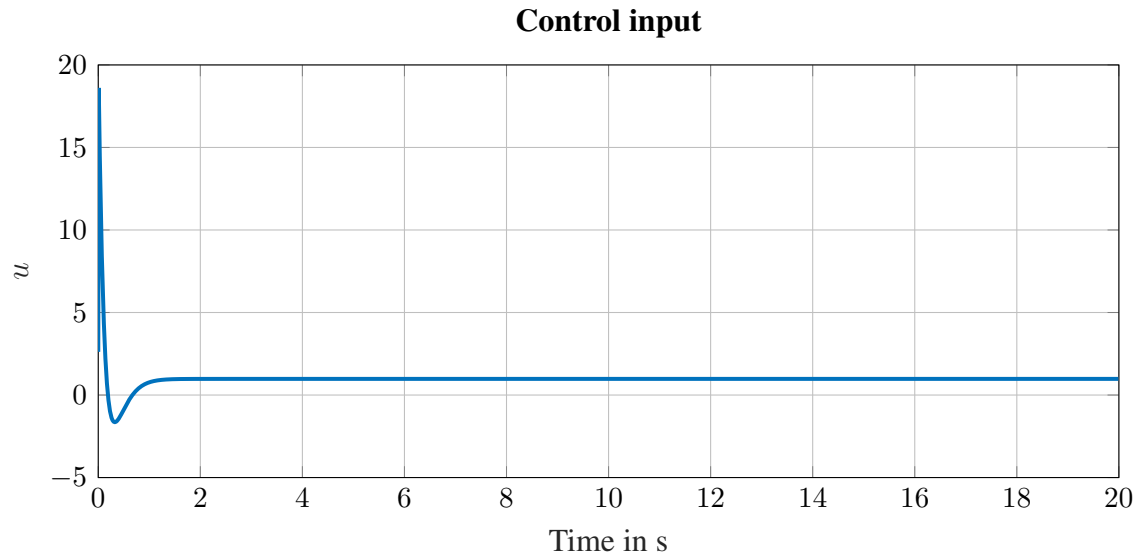


Figure 3.7: Control input u applied to the pendulum over time.

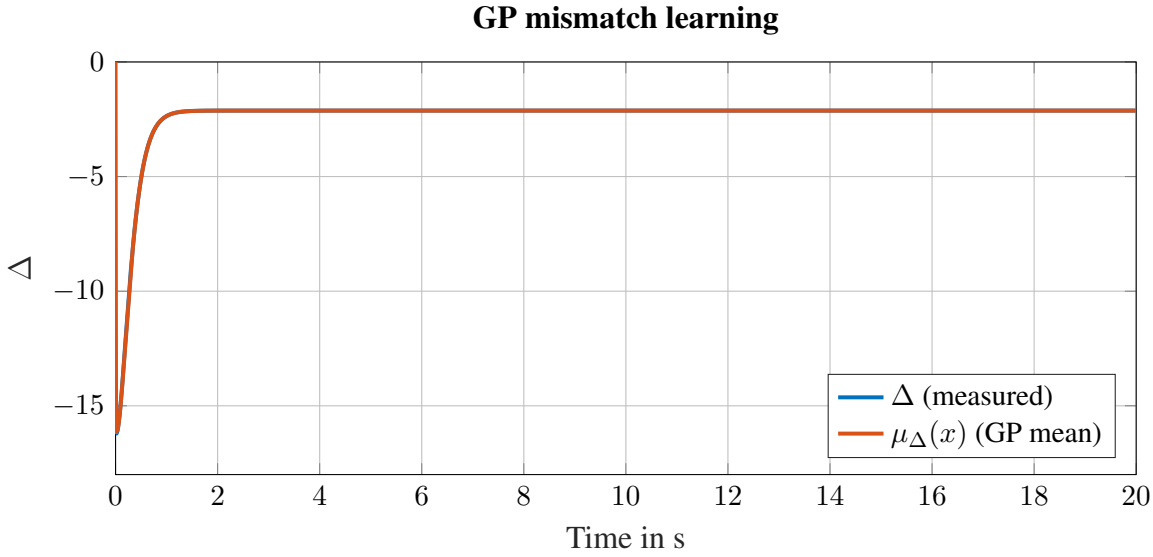


Figure 3.8: Measured mismatch Δ and GP mean prediction $\mu_\Delta(x)$ over time.

The pendulum angle x_1 reaches the reference value quickly and then stays close to it. The control input shows a short transient at the beginning and then settles to an almost constant value. The mismatch plot shows that the GP estimate $\mu_\Delta(x)$ rapidly adapts to the measured mismatch Δ, which reduces the tracking error.

3.4 d

The pendulum is controlled to track a time-varying reference trajectory generated by a Van der Pol oscillator using GP-MRAC. Figures 3.9, 3.10, 3.11, and 3.12 show the tracking results, the applied control input, and the learned model mismatch.

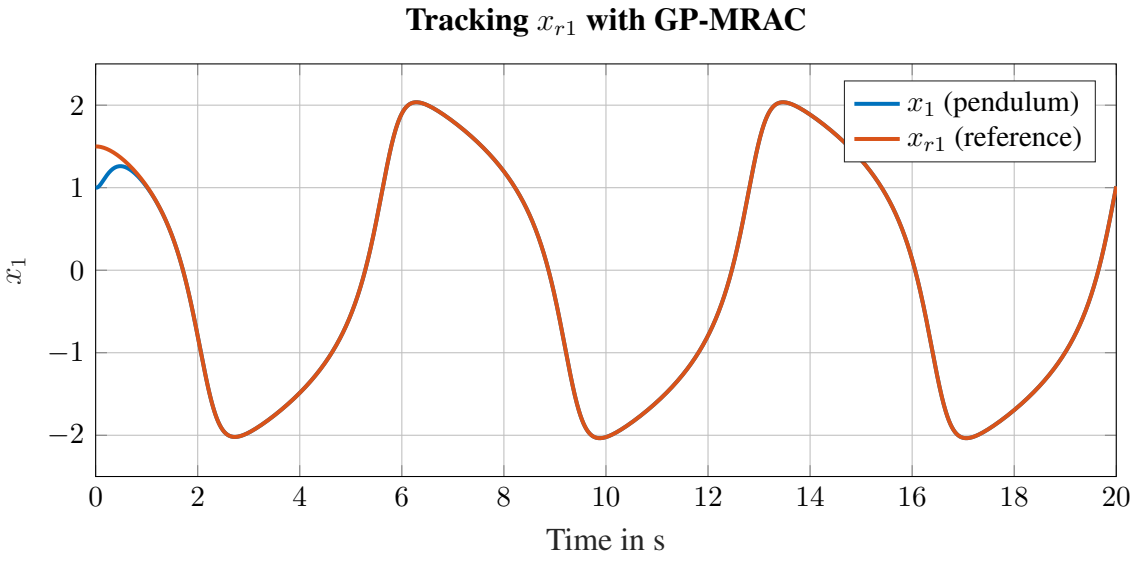


Figure 3.9: Pendulum angle x_1 and reference angle x_{r1} over time.

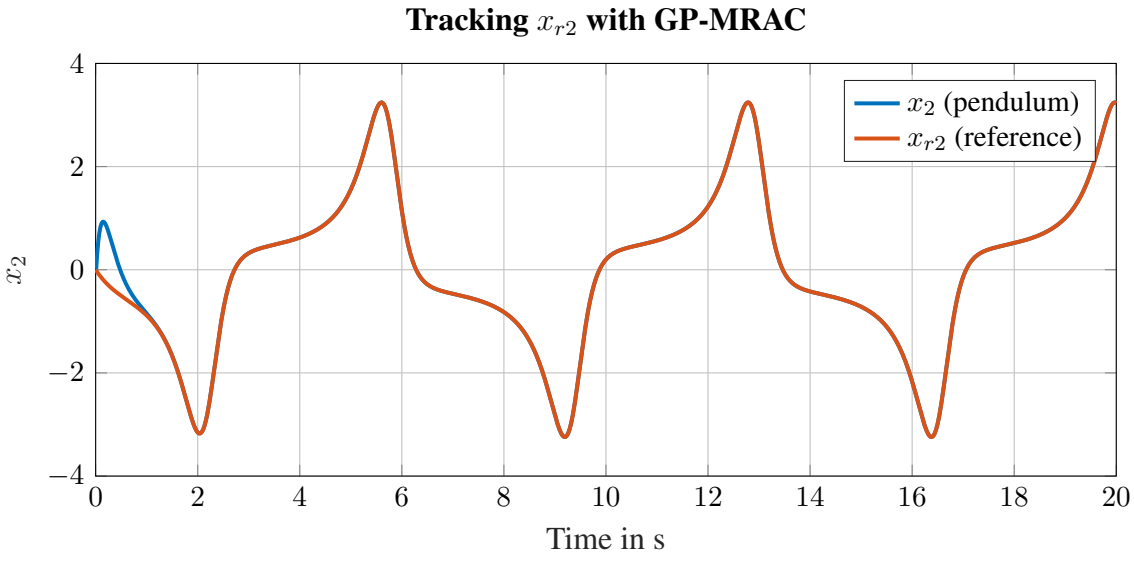


Figure 3.10: Pendulum angular velocity x_2 and reference angular velocity x_{r2} over time.

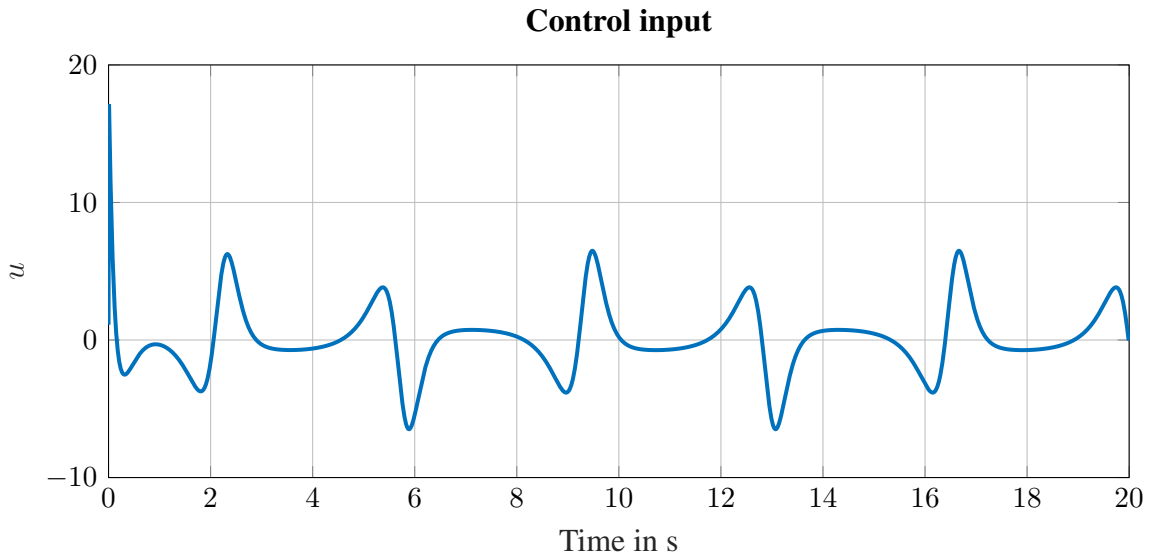


Figure 3.11: Control input u applied to the pendulum during reference tracking.

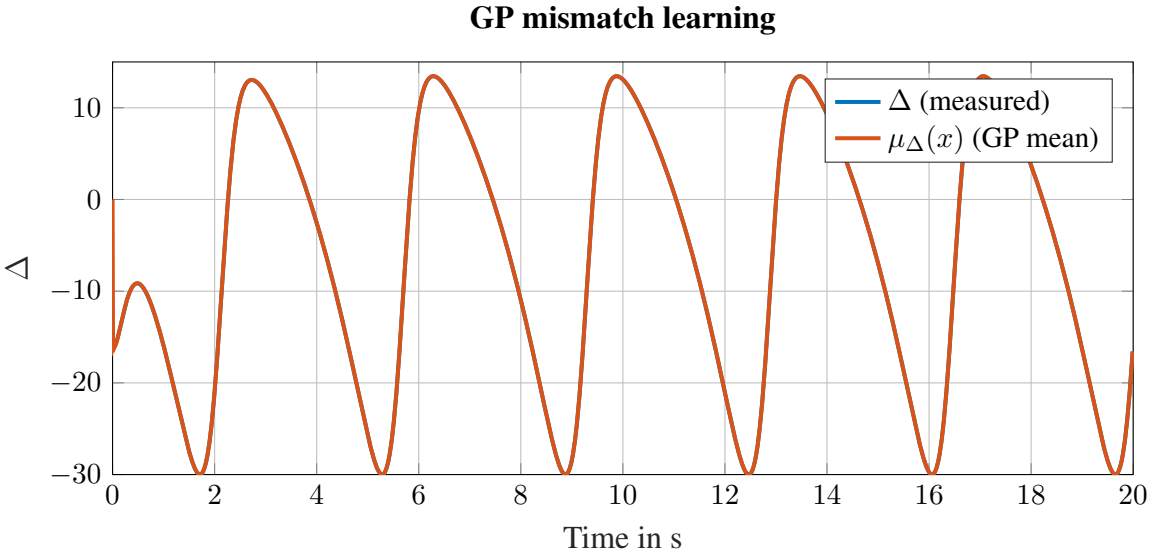


Figure 3.12: Measured mismatch Δ and GP mean prediction $\mu_\Delta(x)$ over time.

The pendulum angle and angular velocity follow the reference trajectories over time. The control input changes periodically to track the reference motion and stays bounded. The mismatch plot shows that the GP mean prediction $\mu_\Delta(x)$ follows the measured mismatch Δ and updates continuously during the motion.