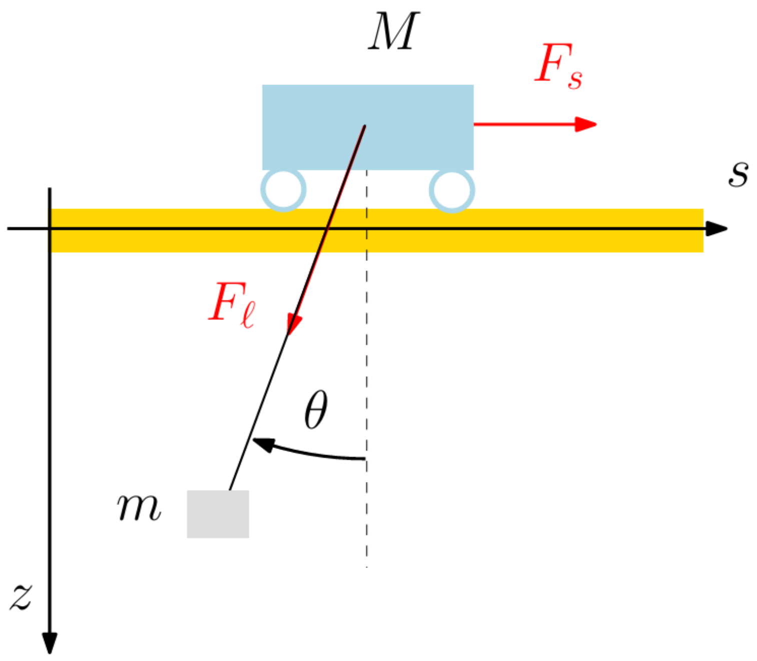
Automatic Control Project 2025

Giacomo Montagna [257773]

26 June 2025



Contents

1 Question 1

To assess the feasibility of state feedback control and state estimation, we examine the system's controllability and observability properties. This is done by constructing the controllability matrix R and the observability matrix O, and verifying that both have full rank.

1.1 Controllability

Since matrix $A \in \mathbb{R}^{6\times 6}$, the controllability matrix is given by

$$\mathcal{R} = \begin{bmatrix} B & AB & A^2B & A^3B & A^4B & A^5B \end{bmatrix}.$$

The system is controllable if $\mathbf{rank}(\mathcal{R}) = 6$, which matches the dimension of A.

```
R = [B A*B A^2*B A^3*B A^4*B A^5*B];
rank(R) == size(A,1)
```

```
ans = logical
   1
```

1.2 Observability

Since $A \in \mathbb{R}^{6\times 6}$, the observability matrix is given by

$$\mathcal{O} = \begin{bmatrix} C \\ CA \\ CA^2 \\ CA^3 \\ CA^4 \\ CA^5 \end{bmatrix}.$$

The system is observable if $\mathbf{rank}(\mathcal{O}) = 6$, matching the dimension of A.

```
O= [ C; C*A; C*A^2; C*A^3; C*A^4; C*A^5];
rank(O) == size(A,1)
```

```
ans = logical
   1
```

1.3 Design a controller

Since the system is both controllable and observable, it meets the key requirements for designing a full state-space control architecture. Controllability ensures that the input can influence all internal states, while observability guarantees that the full state vector can be reconstructed from output measurements.

Under these conditions, it is possible to implement a static state feedback control law of the form $u = K\hat{x}$, where K is a gain matrix designed to assign the eigenvalues of the closed-loop matrix $A + BK$. By placing these eigenvalues in the left half of the complex plane, the linearized system around the equilibrium point can be stabilized, ensuring that all state trajectories converge asymptotically to zero.

2 Question 2

2.1 Observer Design (L)

Our goal is to estimate x with a convergence rate of at least α by designing an observer.
We compute the observer gain L by solving a Linear Matrix Inequality (LMI), ensuring that the estimation error converges rapidly.
The following dual LMI must be satisfied:

$$1.\quad W \succ 0 \tag{1}$$

$$2.\quad (A^\top W - C^\top Y) + (A^\top W - C^\top Y)^\top \preceq -2\alpha W \tag{2}$$

where:

- W is a symmetric positive definite matrix,
- Y is a design variable,
- the feedback observer is computed as $L = W^{-1}Y^\top$.
- the error dynamic is computed as $\dot{e} = (A - LC)e$.

This condition guarantees that the eigenvalues of the observer error dynamics matrix $A - LC$ have real parts strictly less than $-\alpha$, ensuring exponential convergence of the estimation error with a rate of at least α.

```
alpha = 6;
% Number of States, Inputs and Outputs
n = 6;
m = 2;
p = 2;

% YALMIP settings
yalmip('clear');
opts = sdpsettings('solver', 'mosek', 'verbose', 0);
fake_zero = 1e-4;

W_obs = sdpvar(n,n);
Y = sdpvar(p,n);

constr_L = [W_obs >= fake_zero*eye(n);
            (A_num'*W_obs - C_num'*Y) + (A_num'*W_obs - C_num'*Y)' <= -2*alpha*W_obs];


sol_L = solvesdp(constr_L, [], opts);
[primal_res_L, dual_res_L] = check(constr_L);
```

State Feedback Observer fesability

```
all([primal_res_L; dual_res_L] >= -fake_zero)
```

```
ans = logical
   1
```

```
Wv_obs = value(W_obs);
Yv = value(Y);
L = inv(Wv_obs) * Yv';
Gain_L = mat2str(L, 4)
```

```
Gain_L = '[50.83 1.597e-14;-3.569e-14 19.31;-202 -7.885e-14;1181 3.812e-13;-3.003e-13 159.2;288.1 -2.
```

```
Observer_poles = mat2str(real(eig(A_num - L*C_num)), 4) %(A - L*C)
```

```
Observer_poles = '[-17.23;-17.23;-8.256;-8.256;-9.655;-9.655]'
```

2.2 Considerations

To develop a robust and stable control system, we first verified that the system is both controllable and observable. These properties ensure the theoretical feasibility of implementing a full-state feedback controller and a state observer.
Based on this, we proceeded to design:

- A state observer that estimates the state $\hat{x}$ from the output y with fast convergence.
- A state feedback controller of the form $u = K\hat{x}$, which will be designed in the next sections to stabilize the system.

The effectiveness of both the observer and the controller will be evaluated through numerical simulations. The observer was designed using Linear Matrix Inequalities (LMIs) formulated in YALMIP and solved with the MOSEK solver. A minimum convergence rate of $\alpha = 6$ was imposed, ensuring that all eigenvalues of the observer dynamics matrix $(A - LC)$ lie strictly to the left of -6 in the complex plane. The state feedback controller $u = K\hat{x}$ will be addressed in subsequent sections, with various target convergence rates.

3 Question 3

3.1 Controllers Design (K_i)

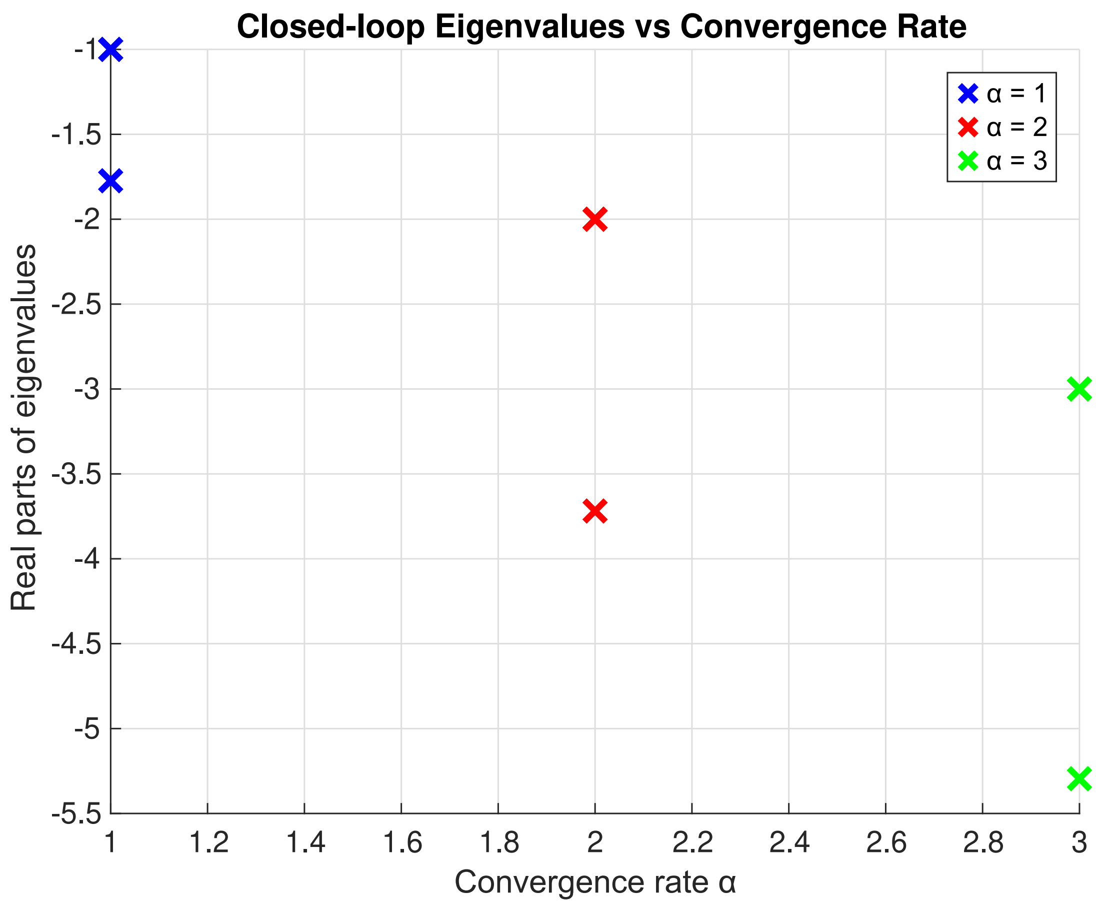
To ensure not only stability but also satisfactory performance, we designed three state feedback controllers K_1, K_2, K_3, each enforcing increasing convergence rates $\alpha = 1, 2, 3$. For each case, we minimized the gain norm $\|K_i\|$ to reduce control effort while satisfying the stability constraints. This was formulated as an LMI optimization problem.
The LMI constraints enforce:

$$1. \quad W \succ 0 \tag{3}$$

$$2. \quad (AW + BX) + (AW + BX)^\top \preceq -2\alpha_i W \tag{4}$$

$$3. \quad \begin{bmatrix} \mu I_n & X^\top \\ X & \mu I_m \end{bmatrix} \succeq 0 \tag{5}$$

```
% Solve LMI with YALMIP

K_list = {}; P_list = {}; mu_list = []; eig_list = [];

for alpha_i = 1:3
    yalmip('clear');
    W = sdpvar(n,n);
    X = sdpvar(m,n);
    mu = sdpvar(1,1);

    % LMI formulation
    LMI_K = [
        W >= fake_zero*eye(n),
        (A_num*W + B_num*X) + (A_num*W + B_num*X)' <= -2*alpha_i*W,
        [mu*eye(n), X'; X, mu*eye(m)] >= fake_zero*eye(n+m)
    ];

    diagnostics = optimize(LMI_K, mu, opts);
    Wv = value(W); Xv = value(X);
    K_i = Xv * inv(Wv);
    mu_i = value(mu);
    eigs_cl = real(eig(A_num + B_num*K_i));

    % Store
    K_list{alpha_i} = K_i;
    mu_list(alpha_i) = mu_i;
    eig_list{alpha_i} = eigs_cl;
    P_list{alpha_i} = inv(Wv);

end
```



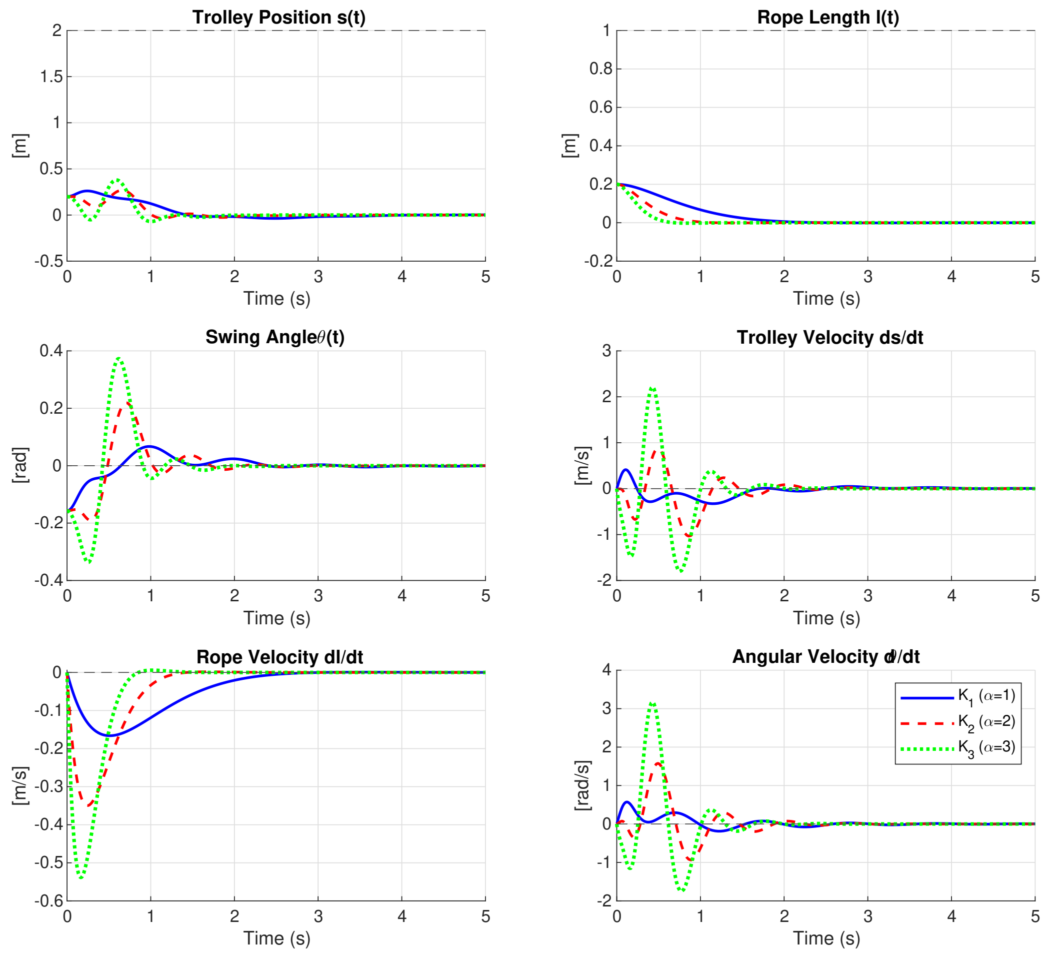
Among the three gains, each satisfies its corresponding convergence rate.
We later compare their performance in simulation of both linear and nonlinear systems.
Simulation results show that increasing the convergence rate leads to stronger damping (faster decay), but also requires larger feedback gains.

The linear dynamics of the system were simulated by numerically integrating a set of differential equations that describe the full physical behavior of the crane. The function `observer_augmented_ode` implements these equations.

- **True state dynamics:** the actual system evolves according to the linear state-space model $\dot{x} = Ax + Bu$, where the control input $u = K\hat{x}$ is computed from the estimated state.
- **Estimated state dynamics:** the observer reconstructs the state using the model $\dot{\hat{x}} = A\hat{x} + Bu + L(y - \hat{y})$, where the correction term $L(y - \hat{y})$ drives the estimation error to zero.

3.2 State Trajectories

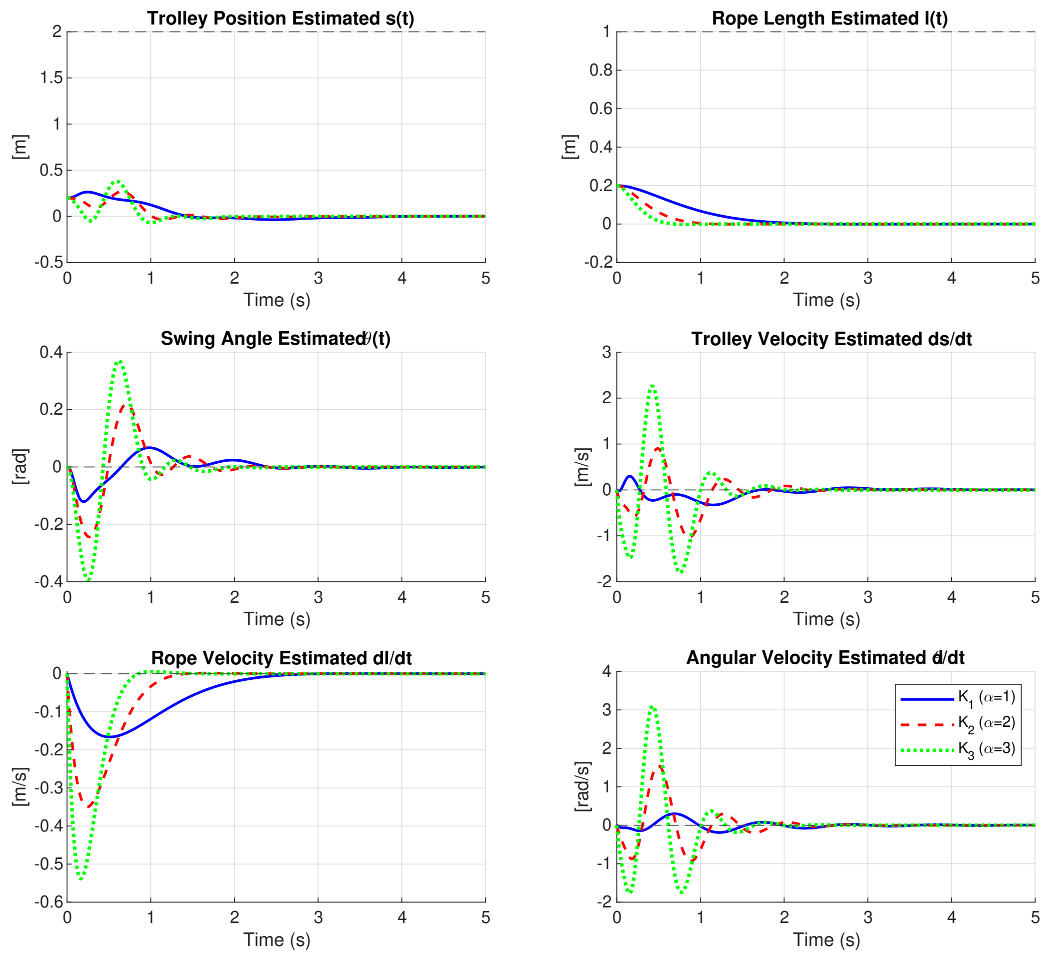
We now compare the linearized dynamics using different K_i.
The initial state is $x_0 = [0.2,\ 0.2,\ -\frac{\pi}{20},\ 0,\ 0,\ 0]^\top$.



Since we are working with the linearized dynamics, the equilibrium point is at the origin ($x = 0$), as is typical for linear systems obtained via local linearization.

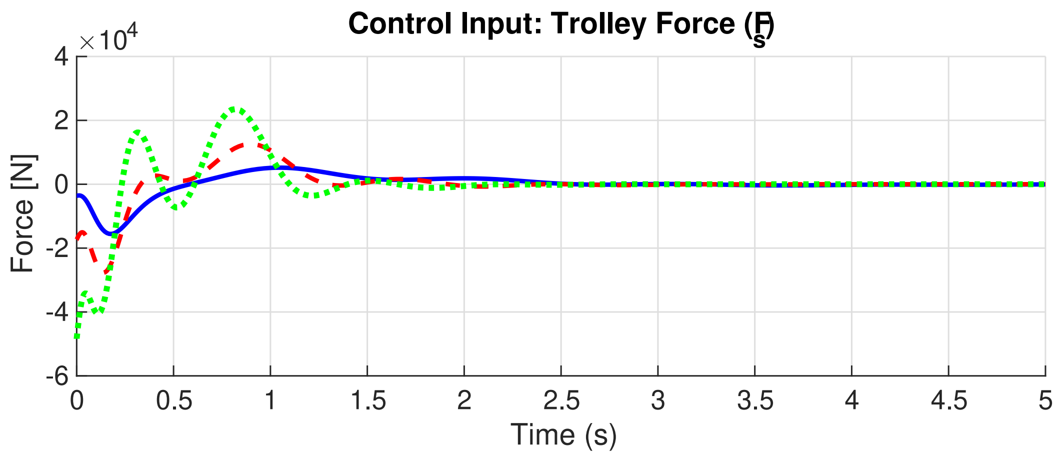
3.3 Estimated State Trajectories

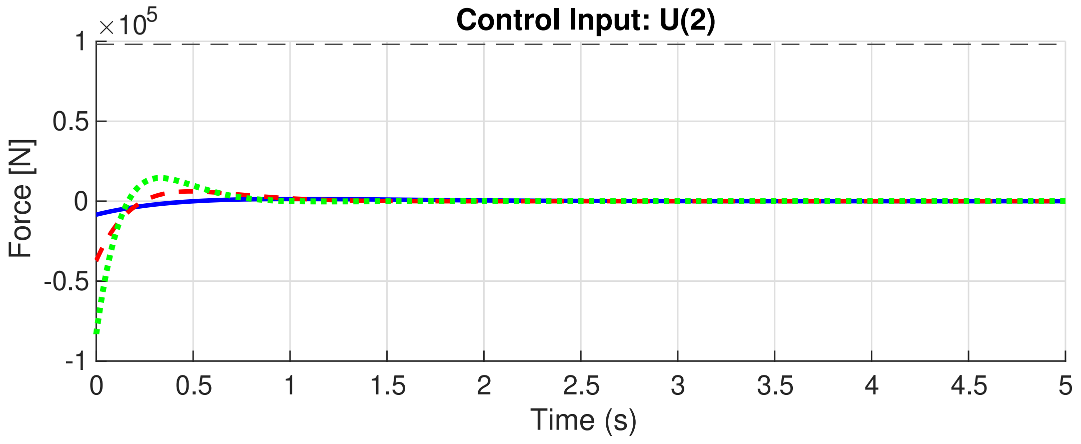
The initial estimated state is $\hat{x}_0 = [0.2,\ 0.2,\ 0,\ 0,\ 0,\ 0]^\top$. The first two components are considered accurate since they are directly measured, while the third (unmeasured) component is initialized to zero arbitrarily.



The estimated dynamics match the true system, except for the incorrectly initialized third state. However, this discrepancy vanishes quickly, as the estimate converges to the true dynamics after a short time, indicating that the observer gain (L) has been correctly implemented.

3.4 Control Input Forces





When computing the control input, the only available information about the state x comes from the output measurements y. Therefore, the control input was computed as $u = K\hat{x}$, using the estimated state provided by the observer.

In the linearized system, the control input at equilibrium is zero, as it represents the deviation from the nominal input required to maintain the system at rest. This is a consequence of linearizing the system around an operating point where all states and inputs are constant.

4 Question 4

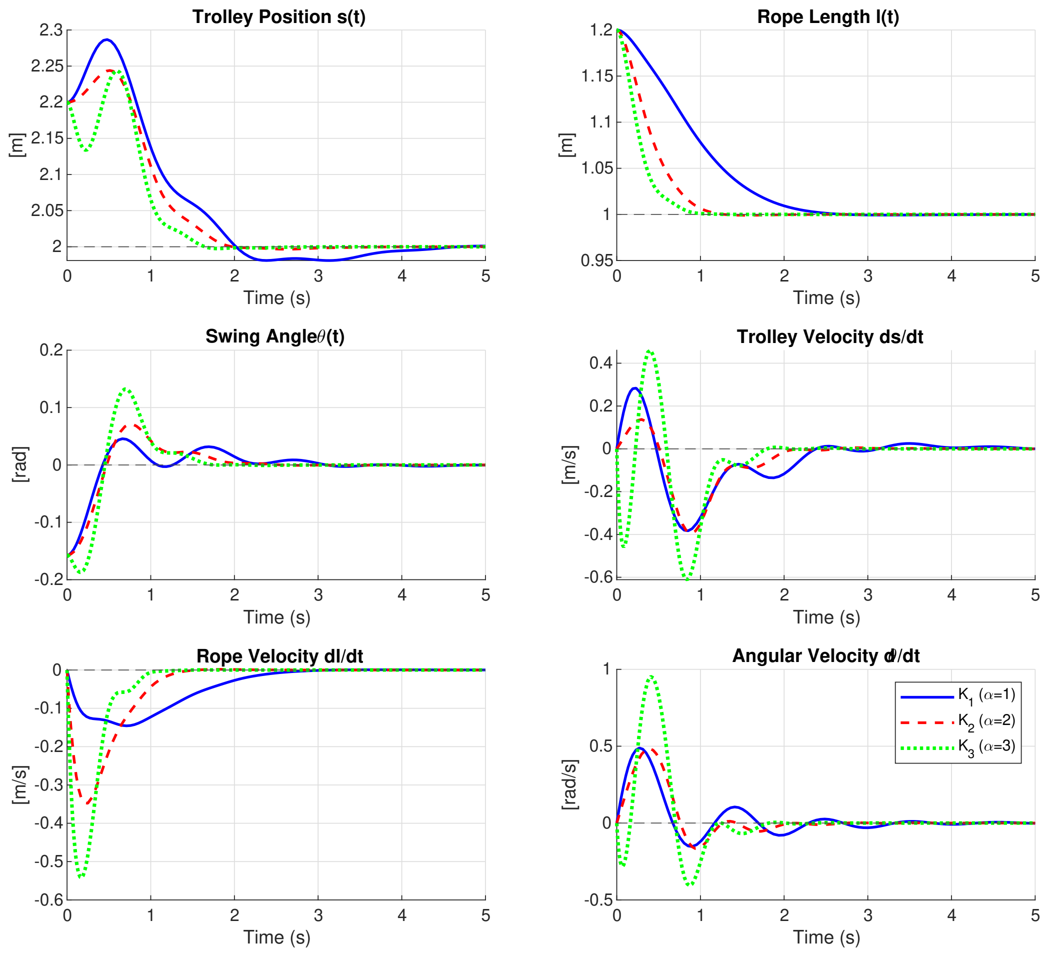
We now simulate the original nonlinear crane model using each controller $K_i \in \{K_1, K_2, K_3\}$. The goal is to evaluate which controller provides the best robustness in practice.

To isolate the performance of the controllers, we assume an ideal observer based on the previous results. So the control input is computed using the true state as

$$u = K_i x.$$

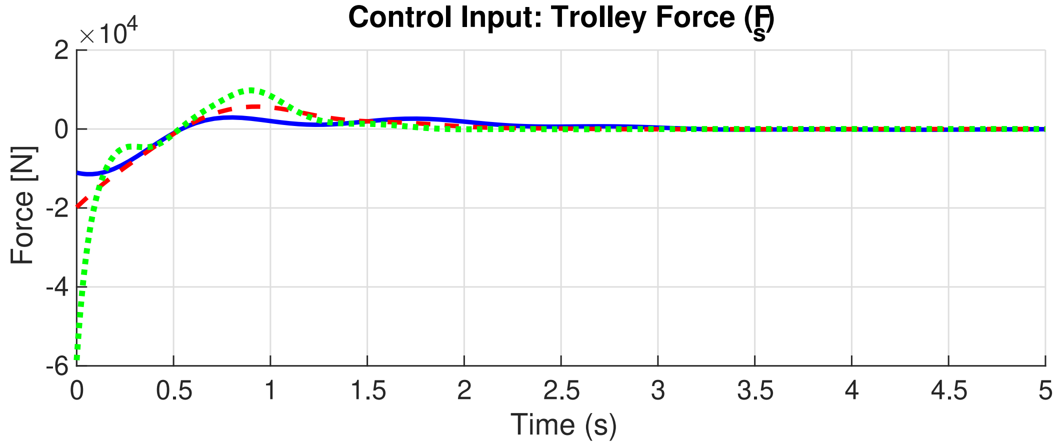
The nonlinear dynamics of the system were simulated by numerically integrating a set of differential equations that describe the full physical behavior of the crane. The function `nonlinear_dynamics` implements these equations based on the system's physical parameters and current state x.
At each time step, the control input u is computed using the estimated state deviation from the equilibrium: $x_{\text{lin}} = x - x_{\text{eq}}$, and the linear controller gain K, as $u = Kx_{\text{lin}}$. The resulting control inputs are then mapped to the physical actuation forces F_s and F_l, which are applied to the nonlinear model.
The state derivatives are computed using the full nonlinear equations of motion.
The simulation was performed using MATLAB's ODE solver (e.g., `ode15s`), with this dynamics function passed as the model to simulate the system's time evolution under closed-loop control.

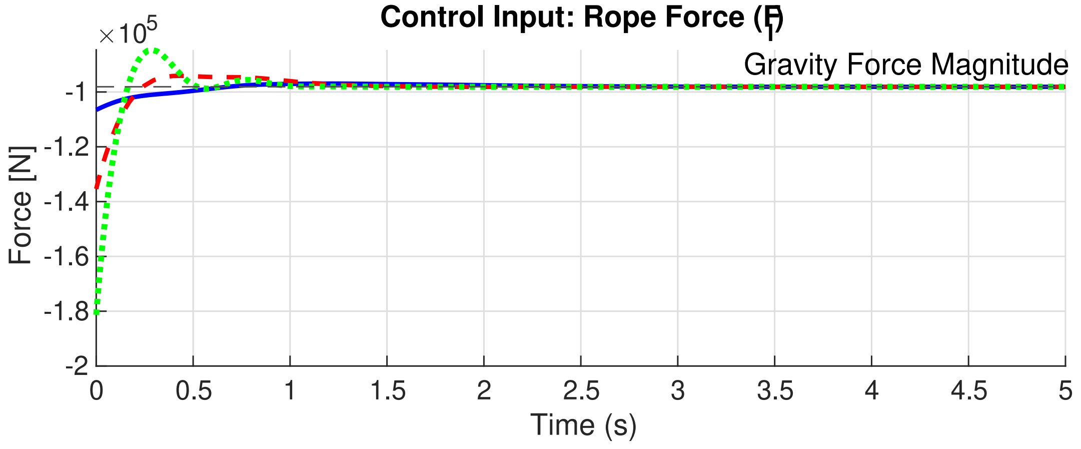
4.1 State Trajectories



Since the nonlinear dynamics are used in the simulation, the system converges to the actual equilibrium point rather than the origin. Specifically, the trolley position converges to approximately 2, and the rope length to around 1, which correspond to the true equilibrium values of the nonlinear system.

4.2 Control Input Forces





From the system model, we observe that the force F_l and the gravitational force act in the same direction, so it is expected that F_l be negative to maintain equilibrium.

4.3 Conclusion

Based on the results of the nonlinear simulations, we observed distinct behaviors for each of the controllers. The controller K_1, designed for a convergence rate of $\alpha = 1$, exhibited very slow convergence. Although it successfully stabilizes the system, the response is too sluggish to be considered effective in a real-world application.
On the other hand, the controller K_3, with a convergence rate of $\alpha = 3$, achieved very fast stabilization. However, this came at the cost of aggressive control actions, resulting in higher oscillations.
The controller K_2, corresponding to $\alpha = 2$, demonstrated a balanced performance. It provided a faster response than K_1, while avoiding the excessive control effort and oscillations observed with K_3.
Therefore, considering both dynamic performance and practical implementation aspects, K_2 emerges as the most suitable controller for the real system.