Gantry Crane Project Report

Tarek Al-Bakri
albakritarek@mines.edu
Electrical Engineering Senior

Brian Durland
bdurland@mines.edu
Mechanical Engineering Senior

Eddie Silva
easilva@mines.edu
Electrical Engineering Senior

***Abstract*—This project extends the gantry crane controller from Project 1 by redesigning it within the state-space framework. The goal of this project is to move a payload to a desired horizontal position with bounded overshoot, acceptable settling time, and zero steady-state error, all while respecting the cart velocity limit and actuator saturation. A linear state-space model of the crane is constructed from the previously derived transfer function, and state feedback is used to place the closed-loop poles within reasonable specifications for the gantry system. Through reference scaling and state augmentation with an integrated error state, zero steady-state error to a 2 m step reference is enforced. Both designs use an estimator, as the states of the plant are arbitrary and inaccessible. Simulation results demonstrate that the state-space controllers track the reference with adequate transient behavior and small payload motion, and keep the velocity within reasonable limits. The new system provides a more flexible design than the original transfer-function-based controller.**

I. Introduction

In this project, the team revisited the gantry crane from Project 1 [1], now using a state-space system and stricter performance requirements to optimize the controller. The goal is to move the theoretical payload to a reference with slight overshoot, acceptable settling time, and zero steady-state error, all while respecting the cart's velocity limit. The crane is modeled with a state-space representation, and optimal control methods are used to select state feedback and estimator gains that meet the specifications in the project description [2].

The initial setup included the derived transfer from Project 1,

$$\frac{g/\ell}{s(s^2+g/\ell)}, \tag{1}$$

where g is gravity (9.81 m/s^2), ℓ is the cable length (3.7 m), and s is the Laplace variable. The equivalent state space version is discussed in section II.

Two controller architectures are considered, one being a reference-scaling design (section III), and the other being a state-augmented design that adds an integral error state (section IV). Both systems are implemented using optimal control techniques and utilize a state feedback controller on the estimated plant states.

The remainder of the report outlines the plant modeling, the design of each controller and estimator, and a comparison of their tracking performance, control effort, and robustness.

II. Modeling

To represent the crane in state space, the team converted the linearized transfer function to control canonical form for a system with only poles[1]. Fully expanded, this system becomes the coupled equations,

$$\begin{aligned}\dot{x} &= \begin{bmatrix}0 & 1 & 0\\ 0 & 0 & 1\\ 0 & -g/\ell & 0\end{bmatrix}x + \begin{bmatrix}0\\0\\g/\ell\end{bmatrix}u\\ y &= \begin{bmatrix}1 & 0 & 0\end{bmatrix}x,\end{aligned} \tag{2}$$

where $\dot{x}$ represents the state velocities as a function of the initial position states and the input reference state, and y represents the output position given the x position. Once again, g is the gravitational acceleration and ℓ is the cable length.

While control canonical form is not derived from any natural relation to the system, the meaning of each state can be extracted from the $\dot{x}$ and y components of equation (2). The first state is the payload position, the second state is the payload velocity (the first derivative of the first state), and the third state is the payload acceleration (the second derivative of the first state). When augmented states are introduced, the error state is the difference between the measured and estimated positions of the mass.

III. Design via Reference Scaling

As guiding criteria for the controller design, the team used the suggested specifications outlined below[2][2]. Of the following parameters, assume the following definitions: $\%OS$ represents the amount of acceptable overshoot for the payload to move, $t_{settling}$ is the amount of time for the payload to settle (98%) at a given position, e_{ss} is the amount of steady state error that is acceptable, and $\dot{z}_{max}(t)$ the maximum velocity of the payload.

1) $\%OS_{max} = 20\%$
2) $t_{settling} = 5$ seconds
3) $e_{ss} = 0$
4) $\dot{z}_{max}(t) = 1$ m/s

These specifications present an entirely new challenge compared to Project 1, as system speed is a much higher priority, and overshoot maximum is far more tolerable. Throughout the entire design process, optimal control methods were used to determine pole locations. C^TC and I_1 were taken as the first pass matrices for Q and R, though these were iterated on to achieve desired system performance.

In order to achieve the outlined criteria, MATLAB's `lqr` function was used, with $Q = 20 \times C^TC$, and an unmodified R. This setup produced the resultant state feedback controller,

$$K_{ref} = \begin{bmatrix}4.4721 & 2.7148 & 1.4310\end{bmatrix}. \tag{3}$$

To estimate and verify the behavior of the poles produced by this state feedback vector, the eigenvectors of $A - BK_{opt}$ are found. Using `eig` in MATLAB on this matrix gives eigenvalues, $\lambda = \{-0.9485 \pm 2.3131j, -1.8971\}$, which are equivalent to the poles of the system with this feedback controller. Then, the control response is characterized with `stepinfo`, which gives a maximum overshoot of 10.1723% and settling time of 3.6926 s. The steady state error is 0.9157, though this is expected to be large as the scale factor has not yet been accounted for.

To ensure the implemented estimator did not drive the system response, real pole components 4× or larger than the controller poles were chosen. This required significant weighting of system speed in

[1]Refer to [3] for more information

[2]item 4, the same parameter from Project 1 [4], this time is achieved through actuator saturation rather than control design

the `lqr` function, with a final $Q = 500000 \times BB^T$, and R still $= I_1$. As such, the estimator,

$$L_{ref} = \begin{bmatrix} 24.518 & 300.56 & 1809.8 \end{bmatrix}^T, \tag{4}$$

was selected. The final magnitudes of the estimator poles are 6× faster than the controller poles, but this was not further iterated upon.

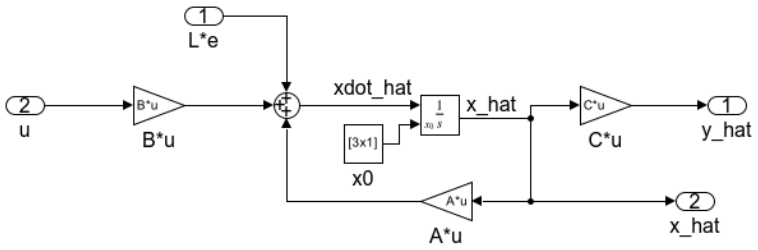

Fig. 1: Estimator Simulink model

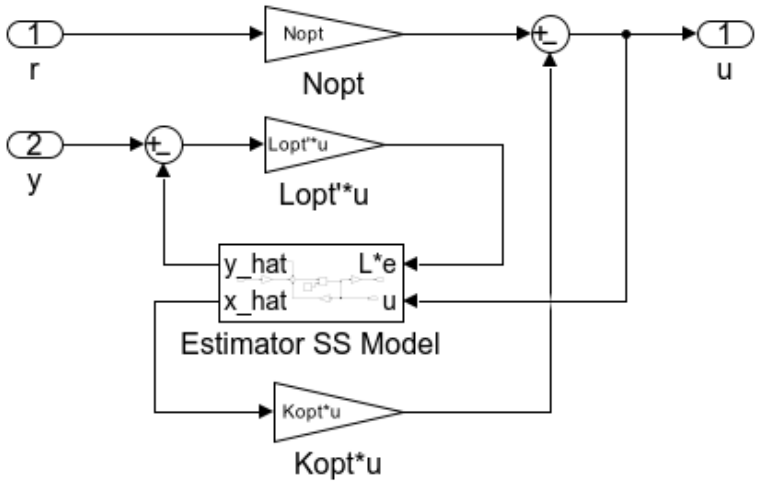

Fig. 2: Reference scaling controller Simulink model

Since this is a state space design, the settled response was not equal to the reference signal. As a result, reference scaling was required to eliminate steady-state error. First, the closed loop dynamics of the full system with estimator and state feedback must be determined. According to the block diagram developed for the system, figure 1 being the estimator block included in figure 2, the system has closed loop dynamics,

$$\begin{aligned} \mathbf{A_{cl}} &= \begin{bmatrix} A & -BK_{ref} \\ L_{ref}C & A - BK_{ref} - L_{ref}C \end{bmatrix} \\ \mathbf{B_{cl}} &= \begin{bmatrix} B \\ B \end{bmatrix} \\ \mathbf{C_{cl}} &= \begin{bmatrix} C & 0_{1\times n} \end{bmatrix}, \end{aligned} \tag{5}$$

which can be used to find the necessary reference scale factor, N. For this system behavior,

$$\mathbf{C_{cl}}(\mathbf{A_{cl}})^{-1}\mathbf{B_{cl}}N = I \tag{6}$$

must be true, and can be rearranged to

$$N = \left(\mathbf{C_{cl}}(\mathbf{A_{cl}})^{-1}\mathbf{B_{cl}}\right)^{-1}, \tag{7}$$

as an explicit definition for N. Solving equation (7) gives a final scale factor of $N = 4.4721$.

All together, this state feedback controller, estimator, and reference scale factor produce a unit step response that has a maximum

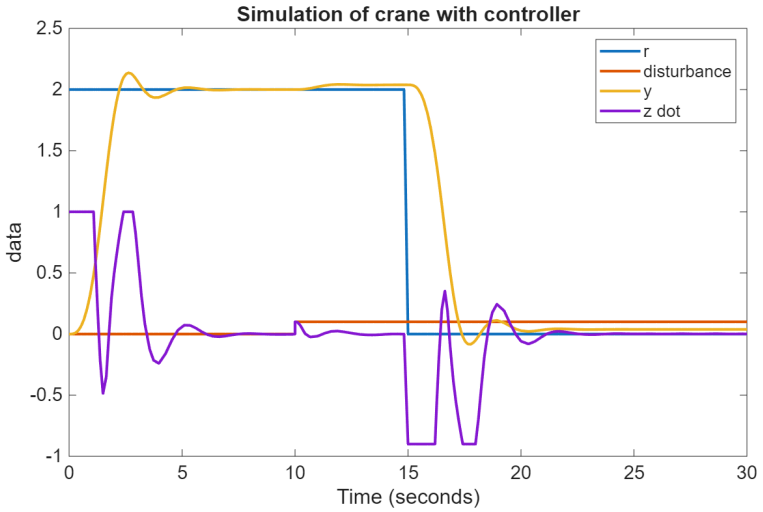

Fig. 3: Reference scaling disturbed pulse response

overshoot of 9.0237%, settling time of 3.9498 s, and negligible ($\sim 10^{-7}$) steady-state error. A saturation block was used to maintain the maximum velocity specification, as defined in [2], though the controller's desired velocity does exceed this bound. These specifications are slightly different than the predicted behavior based on the pole locations, but this is due to the velocity saturation, as the system cannot reach the desired control speed, which damps some of the overshoot, but slows the overall response. This is summarized in figure 3, as the pulse simulates both a step on and off.

IV. Design via State Augmentation

Instead of scaling the reference input, which is not robust to input changes, state augmentation is considered as an alternative design. To do this, the team appended an error state, $e = y - r$, into the state space model. However, using integrator action in this augmentation is desirable, so the integrated error is chosen such that the state and state velocity vectors are described by equation (8). Thus,

$$\begin{aligned} q &= \begin{bmatrix} x \\ \int_0^t e(\tau)\, d\tau \end{bmatrix} \\ \dot{q} &= \begin{bmatrix} \dot{x} \\ e(t) \end{bmatrix} \end{aligned} \tag{8}$$

are modeled in figure 4.

After this augmentation, the system dynamics must be determined before creating a state feedback controller. As a system of state equations, the dynamics can be represented as the following:

$$\begin{aligned} \dot{q} &= \begin{bmatrix} A & 0_{n\times 1} \\ -C & 0 \end{bmatrix} q + \begin{bmatrix} B \\ 0 \end{bmatrix} u + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r \\ y &= \begin{bmatrix} C & 0 \end{bmatrix} q. \end{aligned} \tag{9}$$

To use the optimal control methods, one more modification must be made. Because the additional error state should be weighted differently in the cost function,

$$Q = C^T C + \sigma C_e^T C_e, \tag{10}$$

where $C_e = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}$. Through iteration, the team found $\sigma = 20$, Q weighted by 100, and $R = 0.5 \times I_1$, which lead to the overall augmented state feedback controller,

$$K_{aug} = \begin{bmatrix} 25.573 & 10.350 & 2.7942 & -20 \end{bmatrix}. \tag{11}$$

The controller can be split into K_x, for the first 3 components as the system states, and K_w, the final component as the error state.

Similar to the reference scaling design, the system response can be estimated by finding the eigenvalues (or equivalently, the poles) of $\mathbf{A_q} - \mathbf{B_q}\mathbf{K_q}$. Because of the extra state, there are now four state poles, given by $\lambda = \{-1.5219 \pm 3.2069j, -2.9267, -1.4380\}$. To extract a step response from these, a `zpk` transfer function is made with no zeros, the poles above, and no DC amplification. Then, the system has 0 overshoot, and a settling time of 3.3553 s.

Since the system is measuring the error, the estimator does not need to be adjusted for the additional state. Thus, L_{ref}, equation (4), is kept for this design. Although the final block diagram has a step disturbance noise, the team was unsuccessful in implementing an `lqg` design to mitigate undesired additional signals. If the saturated control signal can be directly measured the noise is inherently rejected, but this is uncertain in physical contexts.

Optimal control methods were used to determine both controllers, and as such, the overall process for achieving desired performance was similar. Since Q prioritizes system response speed, and R prioritizes low control effort, the general methodology taken was to increase the ratio of $Q : R$. While changing the ratio could have been achieved by scaling R down, this method seemed indirect, as the control authority is important even with it being truncated by saturation. The drawback to the arbitrary cost function weighting is the lack of contextual meaning. The `lqr` function produced an optimal result, but that is due to its compliance to specifications, not any objective optimality.

Tuning the cost functions became more challenging with the augmented states, as it was difficult to get the error state to move faster than the system states. The gains that satisfied this requirement are stated above, but the end result only gave an error state pole $2\times$ faster than the slowest system state pole. Use of the augmented states with a saturated actuator also required additional anti-windup that the reference scaling design did not need. The team chose to implement this via back calculation gain, and did not explore a software-based clamping strategy. This gain was simple to tune, as there was a direct correlation between gain and improved settling time, so K_b was chosen as 10. This choice of gain nearly completely damped the windup. Similar to the cost functions, this seems to have limited physical meaning, however, given it is a reasonably large gain, K_b may be susceptible to noise, which is important to note given the inability of the compensator to reject noise.

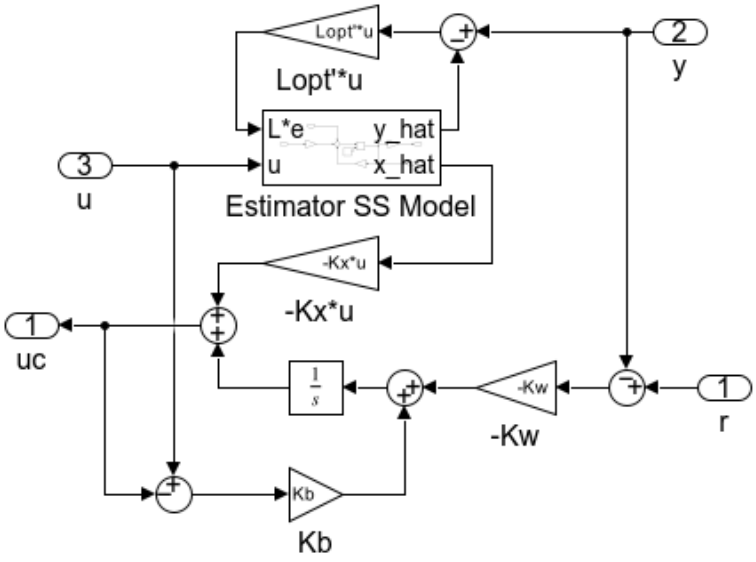

Fig. 4: Augmented states controller Simulink model

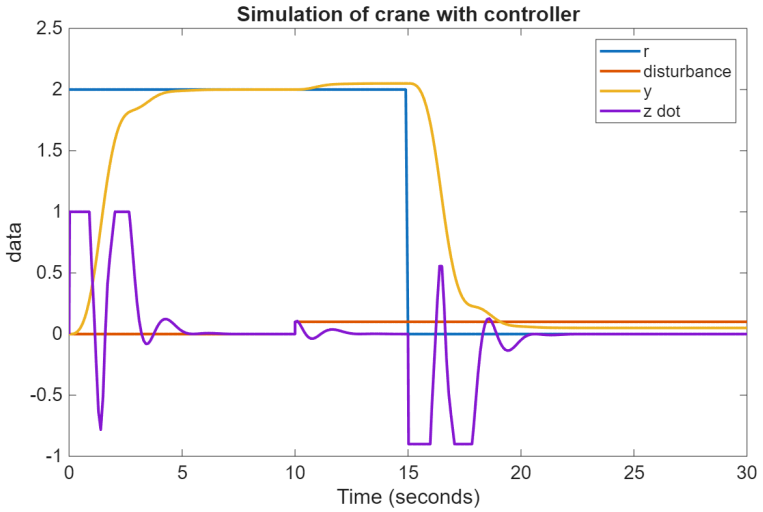

Fig. 5: Augmented state disturbed pulse response

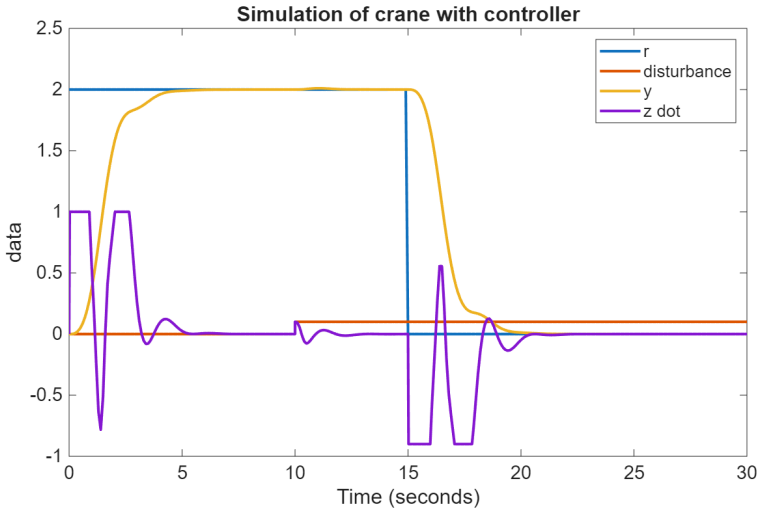

Fig. 6: Augmented state pulse response with direct actuator observability

With all of this implemented, the performance of this controller design can be quantified, as seen in figure 5. In response to a unit step, the controller has a negligible overshoot (on the order of 10^{-14}), a settling time of 3.4465 s, and a negligible steady-state error (on the order of 10^{-16}). As with the reference scaling design, the velocity specification is achieved through actuator saturation, and the implications of this are addressed in section III, and above with the back calculator.

Compared to the reference scaling design, the augmented state design performs better to steps and pulses, as it has a similar settling time without any overshoot. However, the cost of this improved performance is worse margins, which will be discussed in section V. While it is marginal, the augmented state design is also more susceptible to the step disturbance at 10 s. The disturbance is slightly unexpected, as the error state should be mitigating it more actively, but as discussed previously, this is likely due to the lack of an effective `lqg` design. Again, accessing the purely saturated control signal achieves this disturbance rejection, as shown in figure 6, but that assumes the noise exists at the plant input, not the controller output, which may or may not be a realistic assumption.

V. Analysis

In order to determine the margins on both designs, approximations are made without the saturation, as it is a non-linear element that cannot be analyzed with conventional methods. For the reference scaling design,

$$\begin{aligned} \dot{\hat{x}} &= (A - L_{ref}C - BK_{ref})\hat{x} + L_{ref}y \\ u &= -NK\hat{x}, \end{aligned} \tag{12}$$

which is multiplied by G, the original plant model in equation (1) to get the loop gain. From MATLAB, the gain margin is 3.18 dB at $\omega_{co} = 9.8488$ rad/s, and the phase margin is 36.8925° at $\omega_{co} = 3.4293$ rad/s, as seen in figures 11 and 12. These values were verified by adding an additional switched gain into the reference signal, shown in figure 10. Then, $K = 20\log_{10}(GM) = 10.0487$ was directly referenced into the Simulink block diagram to check the exact gain at which the system loses stability, which can be seen in figure 7. The phase margin was not validated, but could be easily implemented with a transport delay of 0.1878 seconds from the plant output back into the controller input, found from

$$36.8925^\circ \times \frac{\pi \text{ rad}}{180^\circ} \times \frac{1 \text{ sec}}{3.4293 \text{ rad}} = 0.1878 \text{ seconds}. \tag{13}$$

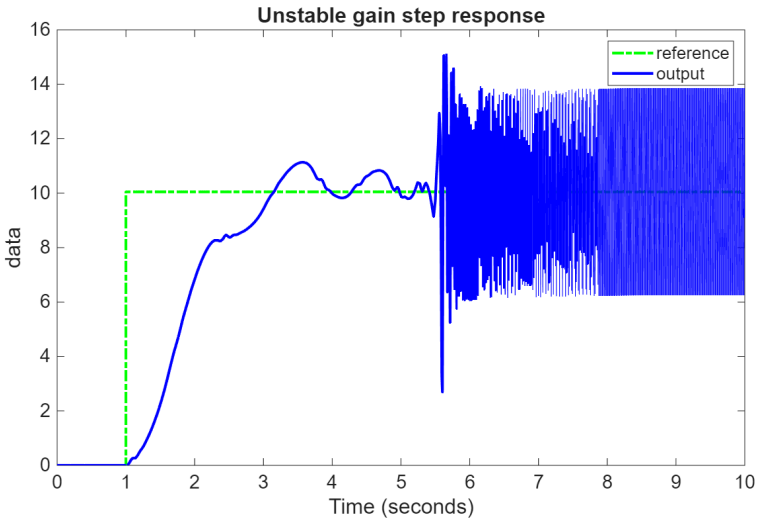

Fig. 7: Step response outside gain margin

A similar approach is taken to find the margins for the augmented state design, but the system is slightly more complicated. The method is still an approximation, with the non-linear saturation block, but the back calculation must be included. This gives

$$\dot{x_c} = \begin{bmatrix} A - L_{ref}C & 0 \\ K_bK_x & -K_b \end{bmatrix} x_c + \begin{bmatrix} 0 & L_{ref} & B \\ -K_w & K_w & K_b \end{bmatrix} \begin{bmatrix} r \\ y \\ u \end{bmatrix} \tag{14}$$

$$u_c = \begin{bmatrix} -K_x & I \end{bmatrix} \text{[5]},$$

which can be used to find the loop gain, and thus, the system margins.

To simulate the back calculator, this state space model is fed into positive unity feedback with gain of 1. The output of this loop can then be multiplied by the plant, G, as earlier, to find the loop gain. This approximated loop gain gives a gain margin of 2.0215 dB at $\omega_{co} = 10.7396$ rad/s and a phase margin of 24.1966° at $\omega_{co} = 5.0921$ rad/s when put into the `margin` function in MATLAB. Since this approach to margin analysis was tested for the reference scaling, it was not checked for the augmented state design. If this controller were being used in a real crane application, these margins should be validated before physical implementation.

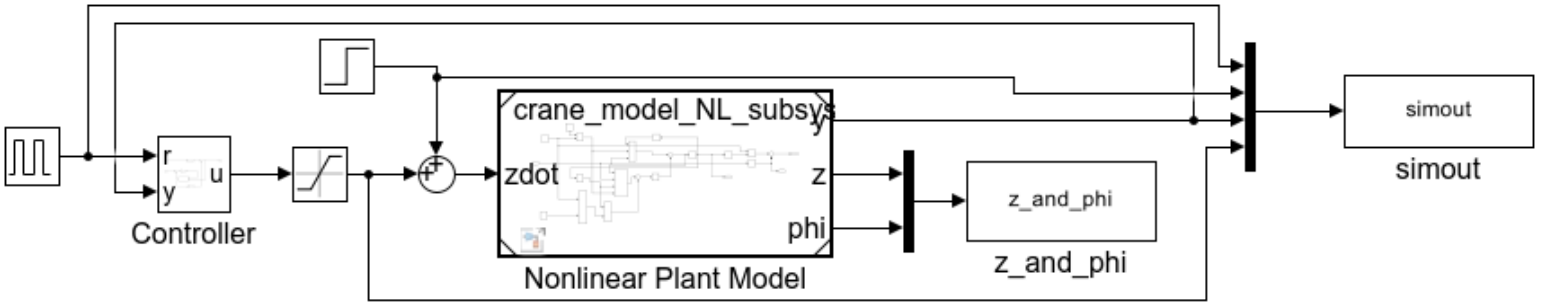

Fig. 8: Reference scaling outer Simulink model

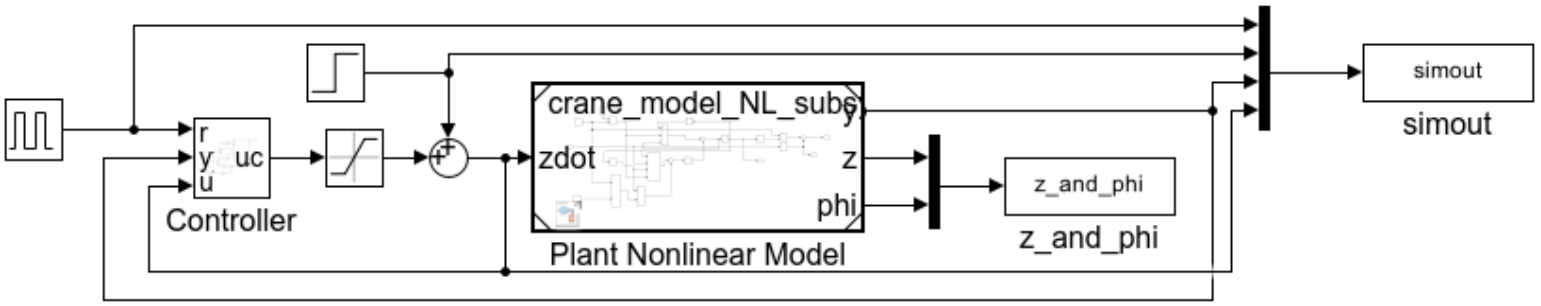

Fig. 9: Augmented states outer Simulink model

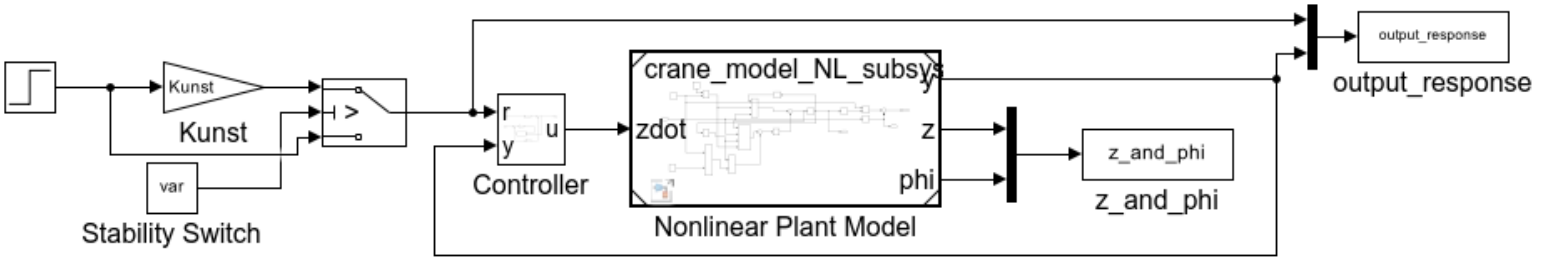

Fig. 10: Gain margin analysis Simulink model

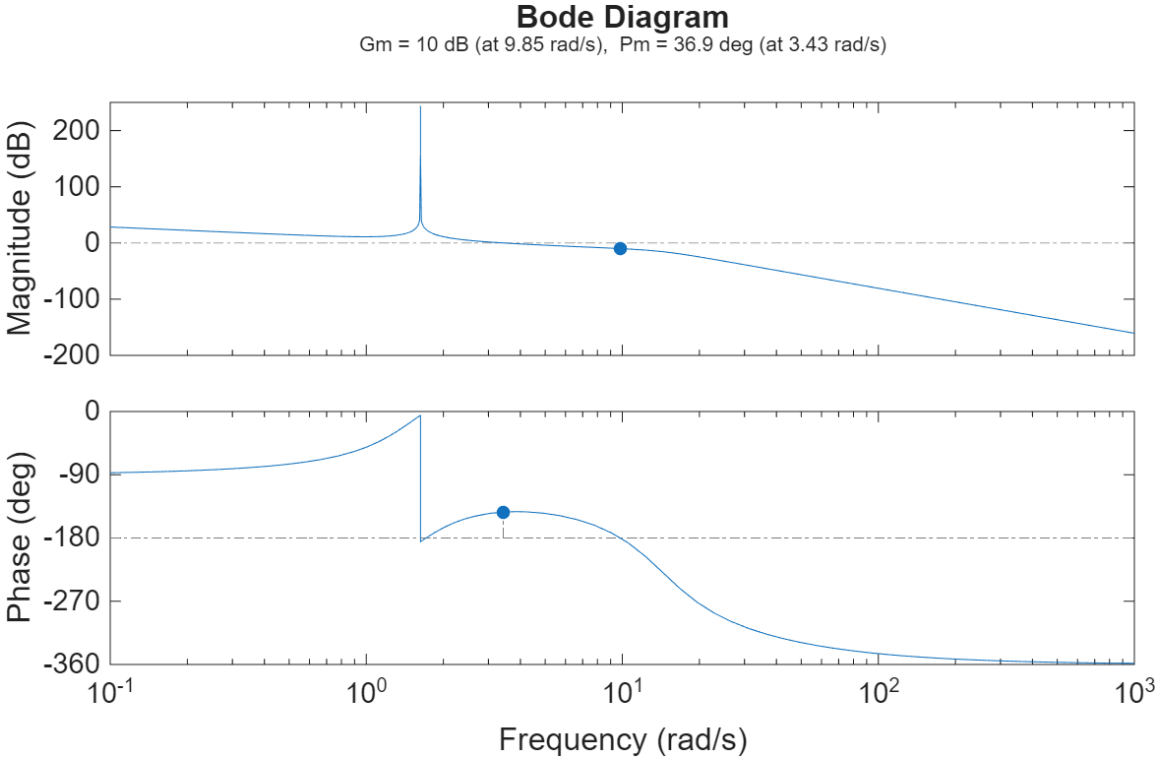

Fig. 11: Reference scaling loop gain Bode plot

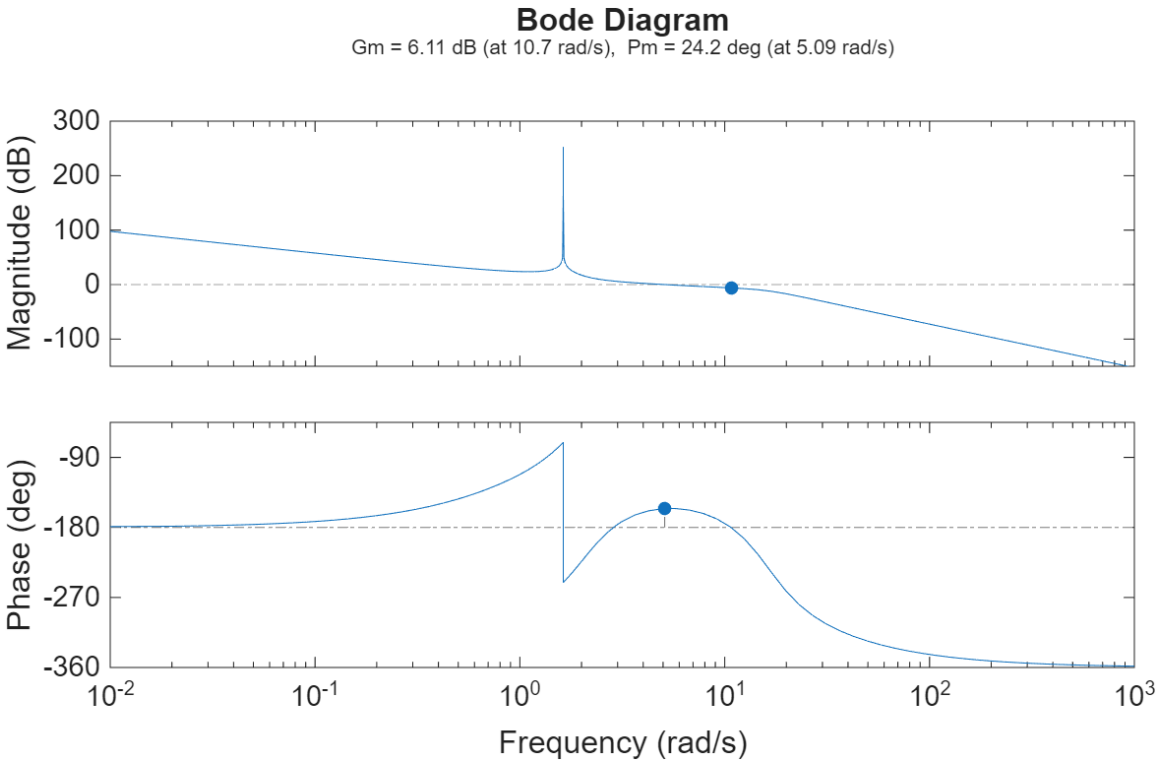

Fig. 12: Augmented state loop gain Bode plots

References

[1] T. Vincent, *Eeng417: Modern control design, project 1*, 2025.

[2] T. Vincent, *Eeng417: Modern control design, project 2*, 2025.

[3] T. Vincent, *Canonical forms and conversion from transfer function to state space, lecture 17 eeng 417*, 2025.

[4] E. S. Tarek Al-Bakri Brian Durland, *Eeng417: Modern control design, project 1*, 2025.

[5] T. Vincent, *Actuator saturation, lecture 23 eeng 417*, 2025.