Translational Oscillator with a Rotating Actuator: A Complete Output-Feedback Control System

Don Kaluarachchi, 10496262

***Abstract*— An output-feedback controller aims to stabilize a system where the current problem proposed is a Translational Oscillator with a Rotating Actuator (TORA). To complete this controller the dynamics of the system were analysed, and the non-linear model was linearised. Equilibrium points for the resulting models were calculated and the system was checked for stability. A state feedback controller was designed using pole placement for the controller gains. Following this a Luenberger observer was designed to estimate states for both non-linear and the linear system. The performance and capabilities of the system were verified and was deemed as a satisfactory controller with a reasonable capability for a real-world implementation.**

Keywords: TORA, Model, Dynamics, Design, State-feedback, Output-feedback, Analysis.

I. Introduction

Control of the Translational Oscillator with a Rotating Actuator system is a classical problem in control theory where it was originally studied as a simplified model of a dual-spin spacecraft [3]. This can be categorized as a non-linear system where unlike a linear actuator, the actuator is involved in the system dynamics.

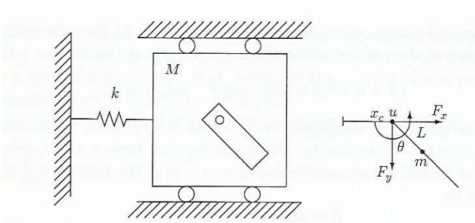

Figure 1: TORA system

As shown in the picture above it consists of a cart which has a mass M connected to a fixed frame of reference by a linear spring with spring constant k. The platform is limited to motion in the horizontal plane along the spring axis. [2] The rotating mass which has a mass of m is actuated by a DC motor which produces a moment of inertia around its centre of mass which is at a distance L from its rotational axis. This input torque is given by u. The relative axis and other measurements are showing in the figure (angle of actuator, Vertical and Horizontal forces).

The following task aims to control fully this non-linear system with a complete controller design such that,

A) The closed-loop system is stable.

B) The closed-loop system exhibits good settling behaviour for a class of initial conditions.

C) The control effort is feasible.

D) Can be regulated to desired point

E) Able to function with disturbances

In addition to the controller, observer design will be used to estimate the state of the system [3].

II. Mathematical Model and Dynamics

The physics describing the equations of motion of the system have been provided [1].

$$F_x = m\frac{d^2}{dt^2}(x_c + Lsin\theta) = \; m\frac{d}{dt^2}\left(\dot{x}_c + L\dot{\theta}cos\theta\right)$$

$$= m(\ddot{x_c} + L\ddot{\theta}cos\theta - L\dot{\theta}^2 sin\theta)$$

$$F_y = m\frac{d^2}{dt^2}(Lcos\theta) = \; m\frac{d}{dt}\left(-L\dot{\theta}sin\theta\right)$$

$$= mL\ddot{\theta}sin\theta - mL\dot{\theta}^2 sin\theta$$

In addition to the above equations the state equations have been provided for the system. Once initial substitutions were carried out the resulting state space model is given by,

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = \frac{[(m+M)u - mLCos(x_1)(mLx_2^2\sin(x_1) - kx_3)]}{(J+mL^2)(m+M) - m^2L^2Cos^2(x_1)}$$

$$\dot{x}_3 = x_4$$

$$\dot{x}_4 = \frac{[-mLuCos(x_1) + (J+mL^2)(mLx_2^2\sin(x_1) - kx_3)]}{(J+mL^2)(m+M) - m^2L^2Cos^2(x_1)}$$

Where each of the state variables were given by,

x_1 : Angle of rotating actuator

x_2 : Angular velocity of rotating actuator

x_3 : Position of Translational oscillator

x_4 : Translational Velocity of Translational oscillator

[1] **Don** is with Faculty of Engineering, Queensland University of Technology (QUT), Australia. This report is in partial fulfilment of EGH445 unit assessment requirements and **submitted on 23/05/2022.** **don.kaluarachchi@connect.qut.edu.au**

The parameters given for the mathematical system are as follows:

Parameter	Value
Mass of Platform (M)	1.3608 kg
Mass of Rotating Actuator (m)	0.096 kg
Inertia (J)	0.0002175 kg m2
Spring Constant (k)	186.3 N/m
Length of Rotating Mass (L)	1 m

Figure 2: Parameters for TORA system

The TORA model as previously established is a non-linear system. It's categorized as such because it does not follow the principle of homogeneity which means that to analyse the system it must be done about its equilibrium points. These equilibrium points are such that the non-linear system behaves much like a linear system allowing for the use of linear mathematical tools to analyse it. These LTI tools can only be used under the condition that the non-linear system is within the region of convergence.

At the equilibrium points the control input will be zero and the state equations are set as,

$$\bar{x} = [\overline{x_1}\ 0\ 0\ 0\ ]$$

For the linearisation the Jacobian matrix is calculated for the 'A' matrix with the intermediary partial derivative step as follows,

$$\begin{bmatrix} \frac{\partial \dot{x}_1}{\partial x_1} & \frac{\partial \dot{x}_1}{\partial x_2} & \frac{\partial \dot{x}_1}{\partial x_3} & \frac{\partial \dot{x}_1}{\partial x_4} \\ \frac{\partial \dot{x}_2}{\partial x_1} & \frac{\partial \dot{x}_2}{\partial x_2} & \frac{\partial \dot{x}_2}{\partial x_3} & \frac{\partial \dot{x}_2}{\partial x_4} \\ \frac{\partial \dot{x}_3}{\partial x_1} & \frac{\partial \dot{x}_3}{\partial x_2} & \frac{\partial \dot{x}_3}{\partial x_3} & \frac{\partial \dot{x}_3}{\partial x_4} \\ \frac{\partial \dot{x}_4}{\partial x_1} & \frac{\partial \dot{x}_4}{\partial x_2} & \frac{\partial \dot{x}_4}{\partial x_3} & \frac{\partial \dot{x}_4}{\partial x_4} \end{bmatrix} = \begin{bmatrix} 0 & 1 & \mathbf{0} & 0 \\ 0 & 0 & \beta & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \alpha & 0 \end{bmatrix}$$

where β and α are given by,

$$\beta = \frac{kLm\cos(\bar{x}_1)}{(J+mL^2)(m+M) - m^2L^2\cos^2(\bar{x}_1)}$$

$$\alpha = \frac{-k(J+mL^2)}{(J+mL^2)(m+M) - m^2L^2\cos^2(\bar{x}_1)}$$

The Jacobian is re-calculated with the input with respect to the states giving the resulting 'B' matrix,

$$\begin{bmatrix} \frac{\partial \dot{x}_1}{\partial u} \\ \frac{\partial \dot{x}_2}{\partial u} \\ \frac{\partial \dot{x}_3}{\partial u} \\ \frac{\partial \dot{x}_4}{\partial u} \end{bmatrix} = \begin{bmatrix} 0 \\ \frac{(m+M)}{(J+mL^2)(m+M) - m^2L^2\cos^2(\bar{x}_1)} \\ 0 \\ \frac{-mL\cos(\bar{x}_1)}{(J+mL^2)(m+M) - m^2L^2\cos^2(\bar{x}_1)} \end{bmatrix}$$

The values for $\overline{x_1}$ are chosen as 0 and 180 for the A and B matrices at the two equilibrium points.

Parameters given for the model are substituted into the matrices and they are calculated resulting in the following set of A and B matrices,

$$A_a = \begin{bmatrix} 0 & 1 & \mathbf{0} & 0 \\ 0 & 0 & \frac{kLm}{(J+mL^2)(m+M) - m^2L^2} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{-k(J+mL^2)}{(J+mL^2)(m+M) - m^2L^2\cos^2(\bar{x}_1)} & 0 \end{bmatrix}$$

$$B_a = \begin{bmatrix} 0 \\ \frac{(m+M)}{(J+mL^2)(m+M) - m^2L^2} \\ 0 \\ \frac{-mL}{(J+mL^2)(m+M) - m^2L^2} \end{bmatrix}$$

$$A_b = \begin{bmatrix} 0 & 1 & \mathbf{0} & 0 \\ 0 & 0 & \frac{-kLm}{(J+mL^2)(m+M) - m^2L^2\cos^2(\bar{x}_1)} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{-k(J+mL^2)}{(J+mL^2)(m+M) - m^2L^2\cos^2(\bar{x}_1)} & 0 \end{bmatrix}$$

$$B_b = \begin{bmatrix} 0 \\ \frac{(m+M)}{(J+mL^2)(m+M) - m^2L^2} \\ 0 \\ \frac{+mL}{(J+mL^2)(m+M) - m^2L^2} \end{bmatrix}$$

Once the equilibrium points have been calculated it is important that these points be checked for stability – this can be done with the use of MATLAB's "eig" function.

As shown below the computed eigenvalues are marginally stable as per Lypunov's indirect method of identifying stability. For both equilibrium points the real value lies on the y-axis.

$$A_a : S_{1,2} = 0$$
$$A_a : S_{3,4} = 0 \pm j11.6997$$

$$A_b : S_{1,2} = 0$$
$$A_b : S_{3,4} = 0 \pm j11.6997$$

These poles aren't ideal as we would like for the system to be completely stable. This can be achieved by moving these poles to the stable left-hand side through the use of a controller.

This TORA system discussed above has been modelled as a Non-Linear system and as a Linear system as shown by the simulink block diagram below.

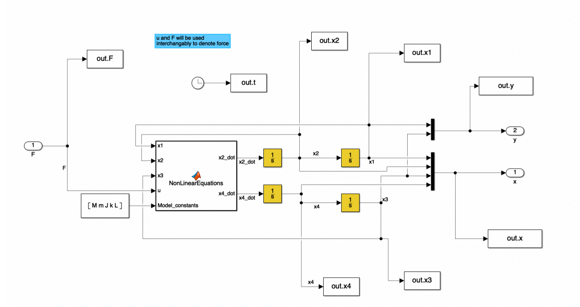

Figure 3: Non-Linear TORA model

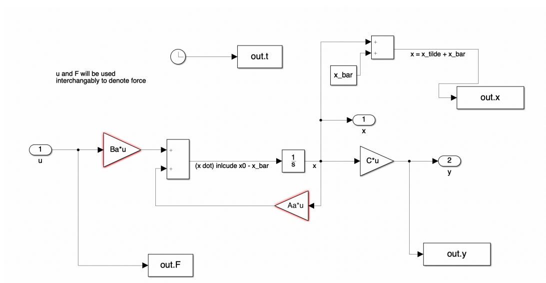

Figure 4: Linear TORA model

These models are simluated using ode4 which is a fixedstep solver with a step size of 0.01 and run for 10 seconds. The Other conditions set are,

$$x_0 = [10\,\text{deg}\ , 0\ , 0.1\ , 0]$$

These will be the baseline conditions for the simulation.

III. State-Feedback Control Design

Pole placement is used to for the design of a feedback controller to stabilise the system. However, before the design of a controller it is important to first find out whether the system state can be transferred from an initial state to another desired state.

The system given by $\dot{x} = Ax + Bu$ can be classified as completely controllable if and only if the matrix known as the controllability matrix is full rank where the controllability is given by,

$$C_{AB} = [B \quad AB \quad A^2B \quad \dots A^{n-1}B\ ]$$

MATLAB functions are used on this controllability matrix. These include checking if the matrix is full rank or for square matrices checking if it has a non-zero determinant.

```
if (Rank_at_Point_a == length(Aa) )
    fprintf("At point A matrix is full
rank, therefore controllable\n")
```

Figure 5: Code Snippet (Controllability)

The result for both equilibrium points concluded that both points are fully controllable. The controllability matrices for the equilibrium points are,

$$C_{ABa} = \begin{bmatrix} 0 & 10.4875 & 0 & -100.1198 \\ 10.4875 & 0 & -100.1198 & 0 \\ 0 & -0.7331 & 0 & 100.3467 \\ -0.7331 & 0 & 100.3467 & 0 \end{bmatrix}$$

$$C_{ABb} = \begin{bmatrix} 0 & 10.4875 & 0 & -100.1198 \\ 10.4875 & 0 & -100.1198 & 0 \\ 0 & 0.7331 & 0 & -100.3467 \\ 0.7331 & 0 & -100.3467 & 0 \end{bmatrix}$$

The only equilibrium point taken will be point "a" as the eignevalues for point "b" are the complex conjugate of eigenvalues at point "a".

Pole placement was carried out with the design characteristics following the standard 2 percent overshoot and settling time of 4 seconds. The slow response poles which are the dominant poles are place on the pole locations,

$$\lambda_{1,2} = -1.0000\ \pm\ 0.8030i$$

The rest of the poles are fast poles and are placed 7 times away from the slow poles as per the dominant pole theorem. This is computed to be at,

$$\lambda_{3,4} = -7.0000\ \pm\ 5.6214i$$

For the design of the state feedback controller given by $u = -Kx$ where K is the gain matrix, u is the input and x is the state the design is done by using it in the desired closed-loop dynamic response. ($\dot{x} = (A - BK)x$)

The controller gain matrix is calculated for the eigenvalues in pole placement via the use of the inbuilt MATLAB command "place" which results in the desired gain matrix that will be implemented in the Non-Linear and Linear controllers. The K matrix for the gain is given by,

$$K = [0.0993\ , 0.1380, -37.7566, -19.8520]$$

The controllers for the non-linear and linear TORA systems are as shown by Figures 6,7. The control input force u is primarily a result of the gain block which multiplies the K by the states and the only difference between the Non-Linear and Linear controllers being that in the Non-Linear there is an input x_bar. The controller can be turned on and off by changing the flag variable in the simulink model.

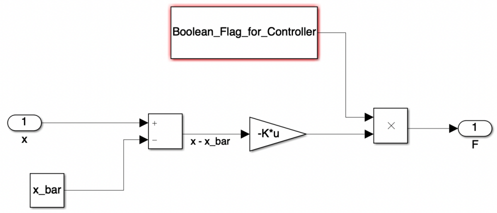

Figure 6: Non-Linear Controller

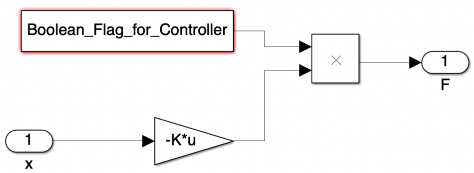

Figure 7: Linear Controller

By simulating the response from the controllers above the following graphs were obtained.

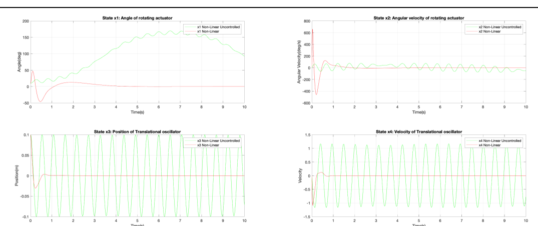

Figure 8: Uncontrolled vs Controlled (Non-Linear)

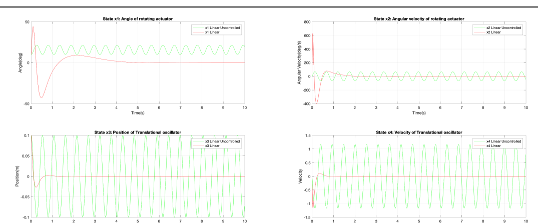

Figure 9 : Uncontrolled vs Controlled (Linear)

As expected for the systems where the controller is enabled they reach their equilibrium points whereas in the uncontrolled model the system continues to oscillate.

IV. Output-Feedback Controller

Much like controllability the system can be classified as observable if the initial state at x(t0) can be reconstructed from it's input and output up to a time after an initial starting point.

This is given by the observability matrix,

$$O_{AC} = \begin{bmatrix} C \\ CA \\ . \\ . \\ CA^{n-1} \end{bmatrix}$$

Where the matrix must have a non-zero determinant or be full rank. For this system all four states were not measured instead only the angle and position given by states x1 and x3 are measured as the remaining states can be derived from the above states. This makes the new C matrix as follows,

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

With this new C matrix that is efficient in observing states we compute its observability and rank,

$$O_{ACa} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 136.5735 & 0 \\ 0 & 0 & -136.8829 & 0 \\ 0 & 0 & 0 & 136.5735 \\ 0 & 0 & 0 & -136.8829 \end{bmatrix}$$

```
Observability_Point_a = obsv(Aa,C)
% Calculate the rank of the system to check if system is observable
Rank2_at_Point_a = rank(Observability_Point_a)
```

Figure 10: Code snippet (Observability)

Once the system has been deemed observable the observer can be designed. The states x1 and x3 are considered where the observer will initially estimate the states with a certain error which will be minimzed by the gain block. This error will be added to the remainder of the observer given by the equation

$$\dot{\hat{x}} = A\hat{x} + Bu + L(y - C\hat{x})$$

The gains for the observer were designed with it being greater by a factor of 10 than the controller gains. Through uninformed iteration of different permutations of the observer gains the following values were considered to give an optimal response.

L = [-65, -65, -165, -165]

Using the above equations the observers were designed for both the non-linear and linear systems as shown below by Figures 11,12. The observers are quite similar in their structure where the only difference is in the linear model it doesnt account for the x_bar as it doesn't take into account the equilibrium point.

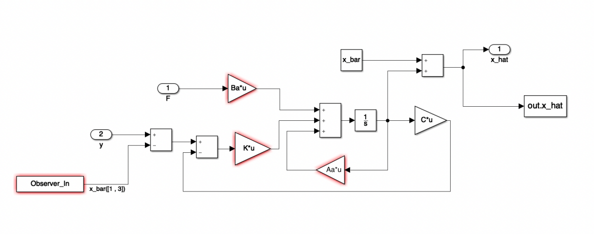

Figure 11: Non-Linear Observer

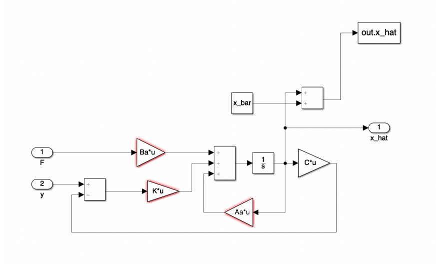

Figure 12: Linear Observer

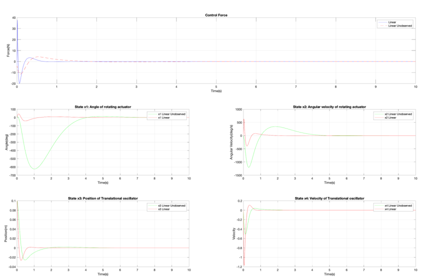

Figure 13: With State Estimates vs Without State Estimates

V. System Synthesis

The final component that completes the system into an output feedback controller is the sub-system block "system synthesis". This concatenates the different states via the use of mux and demux blocks to get the full state vector.

Given by the variable "stateEstimates" is a Boolean flag that enables the system in using the states from the observer. When this is turned off the x matrix is passed through whereas when it is tuned on the estimated states from the observer are fed it into the controller. Shown in Figure 14 is the general structure for the complete linear and non-linear system.

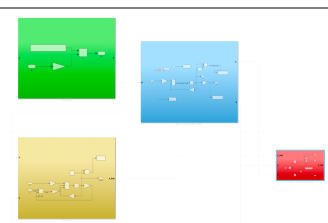

Figure 14: General structure for complete system

VI. Performance Analysis

Given that the system works well under baseline initial conditions further testing was carried out to test the capabilities and limitations of the system. The figure below illustrates the final TORA system that works in initial conditions.

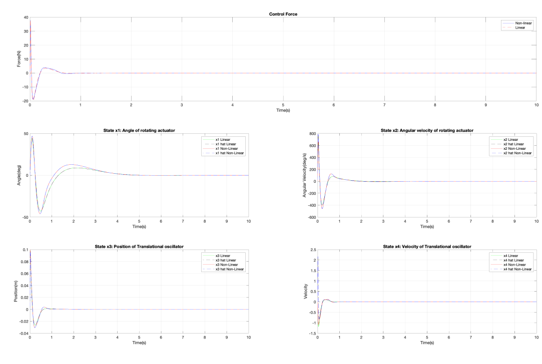

Figure 16: Baseline Complete TORA system

The testing plan was carried out by changing the initial conditions of the system and checking the response. When the starting angle was varied away from the equilibrium points the controller was still able to get the system to equilibrium with the same time of four seconds but required a force around four times the baseline as seen in figure 17.

The next test in the analysis was to the change in initial position of the oscillator and observe the response from the controller. After rigorous testing the range that the controller could handle was ± 0.28m away from the equilibrium point. The equilibrium was reaching within four to five seconds but required a large control force of 50N. Anything beyond this region the system was not controllable. This can be seen by the plots obtained in figures 18 and 19.

The two categories of tests discussed above were testing the limits of the controller and checking the initial conditions such that they fall within the region of convergence. This is the range of values for which the linear controller works for non-linear.

As seen by all the testing done the control force required had a baseline of 5N and in extreme conditions it required 50N which when implemented in a real-life system is a very feasible amount of force to be generated.

In an actual system we cannot ensure that no interference enters and disrupts the controller. The final step in the analysis was to replicate this. To replicate the disturbance, we add a step input that occurs after a certain period of time in the system and observe the system response. It was seen that the system was able to handle small amounts of disturbances and control the system back to its equilibrium, but it required a longer time as seen by the plots in figure 20.

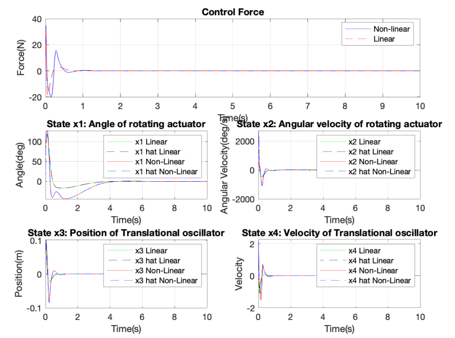

Figure 17: Initial angle changed

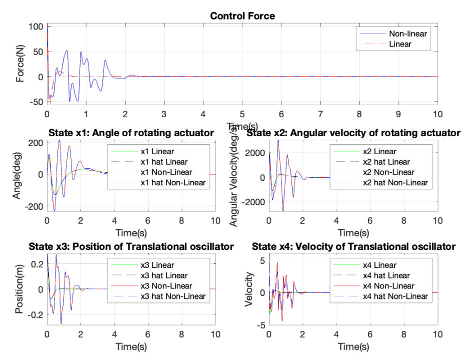

Figure 18: Initial position changed

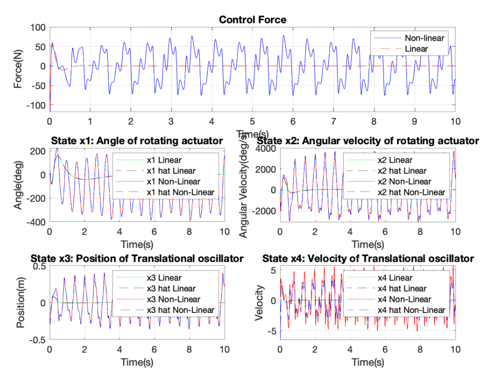

Figure 19: Initial position changed (extreme)

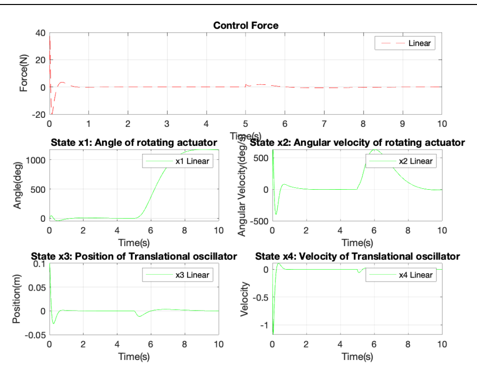

Figure 20: Disturbance added

VII. Future Work

While pole place worked well the poles were selected at random or using an uninformed iterative search, a better way would be through the use of a Linear Quadratic Regulator (LQR) which utilises optimal control. Given below is the equation for the cost function along with matrices Q and R which can be "tuned". The Q matrix penalises the magnitude of the state while the R matrix penalises the magnitude of control.

$$J(x_0, u) = \int_0^{\infty} (x^T Q x + u^T R u)\ dt$$

Furthermore, the system can be made to be more realistic using a damper to regulate the response of the system and to ensure system reaches equilibrium.

VIII. Conclusion

The output-feedback controller aims to stabilize the system. In this paper the entire process of M-D-A (Model Design Analyse) is carried out. Once the system was modelled, controllers and observers were built, and the response was plotted. The complete systems were then put through rigorous testing to observe behaviour along with additional ideas for future work.

We can conclude that the resulting system while contains the mentioned limitations carries out the intended tasks successfully.

Appendix

Approved Extension attached in email.

Acknowledgments

I would like to thank the teaching team for their help in the completion of this project. Notably, Dr D. Quevedo, Dr. A. Mcfayen, G. Silva, A. Razjigaev, H. Shah, M. Edwards. I would also like to thank my peers in the engaging discussions of ideas in class and over slack that helped debug and complete this project.

References

[1] Hassan K. Khalil, "Non-Linear Systems", pp. 347-348,2015.
[2] D. Quevedo, "Report Assignment Instructions", pp. 1-5, 2022.
[3] Mahdi Tavakoli, Hamid D. Taghirad, "Identification and Robust H Control of the Rotational/Translational Actuator System", pp. 3-11, 2005.
[4] A. Moran, "Control of Inverted Pendulum - Wheeled Cart Systems Using Nonlinear Observers", pp. 1-5, 1995.