Furuta Pendulum Balancing Control and Disturbance Rejection via State-Observer Feedback and Internal Model Design– Linear Systems Final Project

Wei-Hsuan Cheng and Cheng-Wei Chen

Abstract— **This project aims to stabilize the Furuta pendulum system by the design of state-observer feedback controller. An internal model is proposed to reject the sinusoidal process noise in the input signal of maxon motor. The simulation of system dynamics and code implementation of control algorithm are done using Simulink, Simscape, and MATLAB.**

I. Introduction

In linear system theory [1], several important concepts are introduced, such as state-space descriptions, controllability, observability, state feedback control, and state observer,etc. In this project, a Furuta pendulum system is introduced in [2] and our goal is to design a controller to balance the pendulum at the top position so it does not fall down. We'll design the *state-observer feedback controller* to balance the pendulum.

The Furuta pendulum system in real world is shown in Fig. 1.

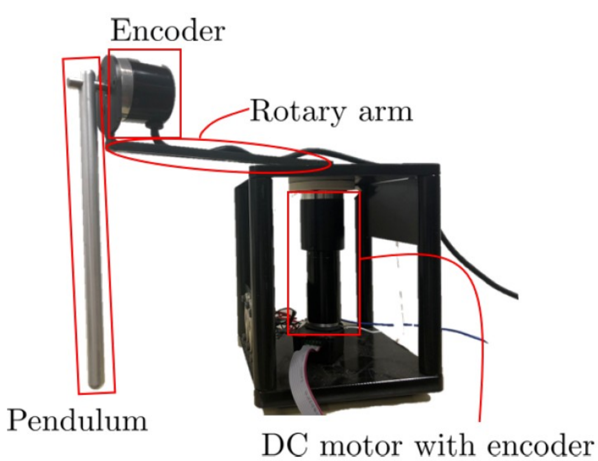

Fig. 1: Furuta Pendulum System in Real World.

II. Dynamic Model of the System

The kinematic model of the Furuta pendulum is given in Fig. 2, with its nonlinear dynamic model shown in Eq. 1.

Thanks to all the efforts from the classmates- B10901042 Wen Perng, B10901002 Chu Rong Chen, B10901061 Gaun Ying Chen, and B09901044 Bing Cheng Jiang. We made this work together.

Wei-Hsuan Cheng is with Biomechatronics Engineering, National Taiwan University, Taipei, Taiwan r11631045@ntu.edu.tw

Cheng-Wei Chen is with Electrical Engineering, National Taiwan University, Taipei, Taiwan cwchenee@ntu.edu.tw

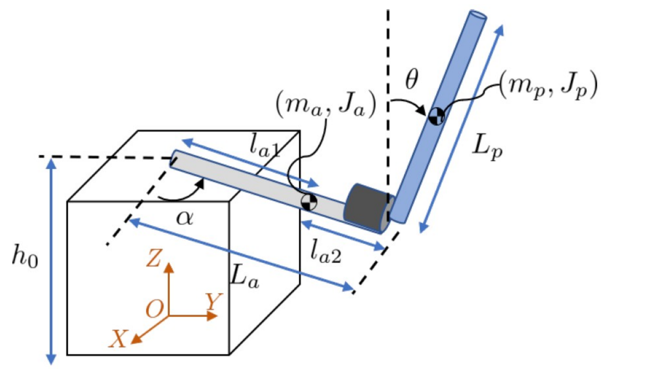

Fig. 2: Kinematic model of Furuta Pendulum.

$$\begin{bmatrix} m_a l_{a1}^2 + J_a + m_p L_a^2 + \frac{1}{4} m_p L_p^2 \sin^2\theta & \frac{L_a L_p}{2} m_p \cos\theta \\ \frac{L_a L_p}{2} m_p \cos\theta & \frac{1}{4} m_p L_p^2 + J_p \end{bmatrix} \begin{bmatrix} \ddot{\alpha} \\ \ddot{\theta} \end{bmatrix}$$

$$+ \begin{bmatrix} \frac{1}{2} m_p L_p^2 \sin\theta\cos\theta\dot{\theta} + c_a & -\frac{L_a L_p}{2} m_p \sin\theta\dot{\theta} \\ -\frac{1}{4} m_p L_p^2 \sin\theta\cos\theta\dot{\alpha} & c_p \end{bmatrix} \begin{bmatrix} \dot{\alpha} \\ \dot{\theta} \end{bmatrix} \tag{1}$$

$$+ \begin{bmatrix} 0 \\ -\frac{L_p}{2} m_p g \sin\theta \end{bmatrix} = \begin{bmatrix} k_g \\ 0 \end{bmatrix} \left(\frac{K_t}{R_a} u - \frac{K_t K_b}{R_a} \dot{\alpha} \right)$$

, where u [V] is the only input of the system, which is the actuated force that drives the motion of the rotary arm. The output is the measured angles of the rotary arm α_m and the pendulum θ_m by the encoder.

Moreover, a voltage disturbance $d(t) = 0.1\sin(2\pi t)$ is added into the input as process noise, and the unknown additive sensor noises $n(t) \in \mathbb{R}^2$ also occur in the output measurements.

Hence, the actual input is $u + d$ and the actual output is $\begin{bmatrix} \alpha_m & \theta_m \end{bmatrix}^T = \begin{bmatrix} \alpha & \theta \end{bmatrix}^T + n$. The Simscape model of the Furuta pendulum system with process and measurement noises is shown in Fig. 3.

A. Linearization and Discretization of the System

The nonlinear model can be converted into a nonlinear state-space form by defining the state vector $x := \begin{bmatrix} \alpha & \theta & \dot{\alpha} & \dot{\theta} \end{bmatrix}^T \in \mathbb{R}^{4\times 1}$, and thus the system becomes

$$\dot{x}(t) = f(x(t), u(t)).$$

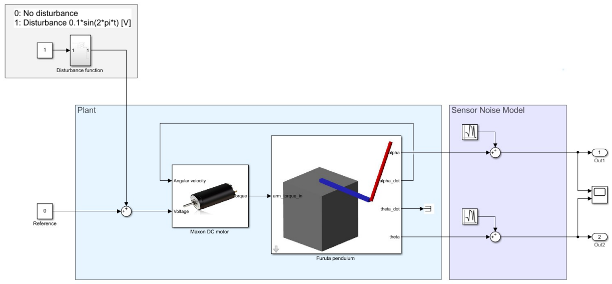

Fig. 3: Furuta pendulum System with Process and Measurement Noises.

For balancing problem, only the dynamics near $x \approx 0$ needs to be considered, so we can further *linearize* the system at the the equilibrium point $x = x_e = 0$. We use MATLAB command `jacobian` to generate the *jacobian matrices* of the linearized system. Then, the linearized system can be expressed as

$$\begin{cases} \dot{x}(t) &= A_c x(t) + B_c u(t) \\ y(t) &= C_c x(t) + D_c u(t) \end{cases} . \tag{2}$$

By MATLAB command `c2d`, Eq. 2 can be further discretized using zero-order hold method (ZOH) with 1 ms sampling period and then becomes Eq. 3

$$\begin{cases} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) + Du(k) \end{cases} . \tag{3}$$

By the substitution of system parameter values given in [2] into the $\{A, B, C, D\}$ matrices, we obtain the numerical forms of the matrices

$$A = \begin{bmatrix} 1.0000 & -0.0000 & 0.0010 & 0.0000 \\ 0 & 1.0001 & 0.0000 & 0.0010 \\ 0 & -0.0123 & 0.9917 & 0.0000 \\ 0 & 0.1165 & 0.0132 & 0.9997 \end{bmatrix} \in \mathbb{R}^{4\times 4}$$

$$, B = \begin{bmatrix} 0.0002 \\ -0.0003 \\ 0.3500 \\ -0.5612 \end{bmatrix} \in \mathbb{R}^{4\times 1}$$

$$, C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \in \mathbb{R}^{2\times 4}$$

$$, D = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \in \mathbb{R}^{2\times 1}.$$

B. Controllability and Observability Analyses of the Linearized System

Before we implement state-observer feedback control, the *controllability* and *observability* of the linearized system needs to be guaranteed so that the states could be controlled and estimated. By the test of the ranks of controllability matrix and observability matrix

$$\mathcal{C} = \begin{bmatrix} B & AB & A^2B & A^3B \end{bmatrix} \text{ (full rank)}$$

$$, \mathcal{O} = \begin{bmatrix} C \\ CA \\ CA^2 \\ CA^3 \end{bmatrix} \text{ (full rank)}$$

, we conclude that the system is complete controllable and observable. Hence, we can then implement state-observer feedback on the system.

C. State Observation of the System

Since the system is complete observable, we first design a state observer to estimate system state and to see that if the state estimation is consistent with the actual states.

The state equation of observer is shown in Eq. 4,

$$\begin{aligned} \hat{x}(k+1) &= A\hat{x}(k) + Bu(k) + L[y(k) - \hat{y}(k)] \\ &= A\hat{x}(k) + Bu(k) + L[Cx(k) - C\hat{x}(k)]. \end{aligned} \tag{4}$$

By defining estimated error state $e(k) := x(k) - \hat{x}(k)$ and by Eq. 3 & 4, we obtain the dynamics of the estimated error state

$$\begin{aligned} e(k+1) &= x(k+1) - \hat{x}(k+1) \\ &= (A - LC)e(k). \end{aligned} \tag{5}$$

This shows that by choosing the value of the observer gain L, we can design the eigenvalues of $A - LC$, which directly controls how fast $\hat{x}$ could converge to x.

We further use the MATLAB command `kalman` to design L via Kalman filter algorithm, which ends up with

$$L = \begin{bmatrix} 0.0207 & -0.0778 \\ -0.0195 & 0.1335 \\ 0.9017 & -5.5347 \\ -1.3954 & 8.9689 \end{bmatrix} \in \mathbb{R}^{4\times 2}$$

and

$$\begin{aligned} \lambda(A - LC) = &0.9229 + i0.0706, 0.9229 - i0.0706 \\ &, 0.9957 + i0.0032, 0.9957 - i0.0032 \end{aligned}$$

, where $\forall |\lambda(A - LC)| < 1$ (stable), and the $\big(Q_o = 1,\ R_o = \text{diag}(0.01, 0.0025)\big)$ in Kalman filter algorithm are obtained by tuning.

The result comparing estimated states to actual states is shown in Fig. 5, and the configuration of observer in Simulink is shown in Fig. 4.

It is clear that the estimated signals are not consistent with actual values, this outcome is not what we want, yet is not surprised as well. The huge difference between the estimated states and actual states stems from the neglecting of the process and sensor noises in Eq. 4 & 5; in other words, the process and sensor noises are not considered in the design of the observer.

From this viewpoint, we may also have another expected outcome that if the input noise is rejected, then the estimation will be more accurate with respect to actual values. The correctness of this statement will be discussed in part IV.

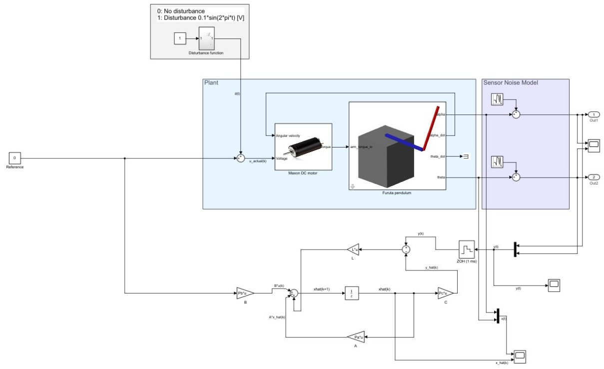

Fig. 4: Configuration of State Observer in Simulink.

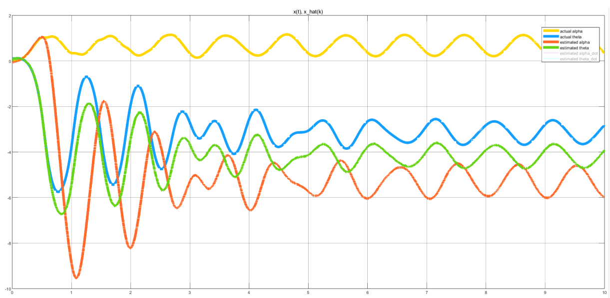

Fig. 5: Estimated States v.s. Actual States *without* control.

III. Design Procedure of Control Algorithm

Our control objective is to balance the pendulum at the top and to free it from the suffering of the input sinusoidal disturbance, which lead to two typical control problems with corresponding control methods:

1) Regulator Problem
   - *State-Observer Feedback Design with* $r = 0$.
2) Sinusoidal Disturbance Rejection
   - *Internal Model Design using Peak Filter.*

By the understanding of linear system theory, it is clear that each of these two problems can be solved and designed separately.

A. Regulator Problem (State-Observer Feedback Design)

After the design of state observer, since the system is complete controllable, we can implement *state observer feedback control law* directly (Regulator Problem, $r = 0$ case)

$$u(k) = -K\hat{x}(k). \tag{6}$$

Then, by the use of the *separation principle* in linear system theory, we can design feedback gain K separately from the observer and stabilize the system by placing the eigenvalues of $A - BK$ inside the unit circle of complex plane.

B. Sinusoidal Disturbance Rejection (Internal Model Design using Peak Filter)

Apart form state-observer feedback, there's still a big issue that the input disturbance should be rejected. This leads to the design of internal model using *peak filter*, which has the capability of rejecting a specific disturbance with sinusoidal characteristics. The discrete-time (DT) state-space description of the peak filter is shown in Eq. 7.

$$\begin{cases} x_m(k+1) & = A_m x_m(k) + B_m[y(k) - r(k)],\ r = 0 \\ u_m(k) & = -K_m x_m(k) \end{cases} \tag{7}$$

, where

$$A_m = \begin{bmatrix} 0 & 1 \\ -1 & 2\cos(\omega_o T)_{\omega_o = 2\pi,\, T = \frac{1}{1000}} \end{bmatrix} \in \mathbb{R}^{2\times 2}$$

$$, B_m = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \in \mathbb{R}^{2\times 2}$$

, and K_m is one of the feedback gain in the *augmented plant* that will be discussed in part III-C.

C. Augmented Plant (Combining State-Observer Feedback and Internal Model)

After we formulates state-observer feedback design and internal model design, we can combine these two together to generate an augmented plant containing both the states of the original plant and the states of the internal model.

The augmented system state $x_{aug}(k)$ is defined as

$$x_{aug}(k) := \begin{bmatrix} x(k) \\ x_m(k) \end{bmatrix}$$

, and the state-space description of the augmented plant is then written as

$$\begin{cases} x_{aug}(k+1) & = \begin{bmatrix} A & 0 \\ B_m C & A_m \end{bmatrix} x_{aug}(k) + \begin{bmatrix} B \\ 0 \end{bmatrix} u(k) \\ & := A_{aug} x_{aug}(k) + B_{aug} u(k) \\ y(k) & = \begin{bmatrix} C & 0 \end{bmatrix} x_{aug}(k) \\ & := C_{aug} x_{aug}(k) \end{cases} . \tag{8}$$

By the test of the rank of the controllability matrix of pair$\{A_{aug}, B_{aug}\}$, it is proved that the augmented plant is also controllable.

Similarly, the extension of the control law in Eq. 6 is Eq. 9

$$\begin{aligned} u(k) &= -K\hat{x}(k) + u_m(k) \\ &= -K\hat{x}(k) - K_m x_m(k) \\ &= -K[x(k) - e(k)] - K_m x_m(k). \end{aligned} \tag{9}$$

By further defining of the feedback gain of the augmented plant $K_{aug} := \begin{bmatrix} K & K_m \end{bmatrix}$ and by Eq. 8 & 9, we can formulate the *closed-loop augmented system*

$$\begin{cases} \begin{bmatrix} x_{aug} \\ e \end{bmatrix}_{k+1} = \begin{bmatrix} A_{aug} - B_{aug}K_{aug} & B_{aug}K \\ 0 & A - LC \end{bmatrix} \begin{bmatrix} x_{aug} \\ e \end{bmatrix}_k \\ y(k) = \begin{bmatrix} C_{aug} & 0 \end{bmatrix} \begin{bmatrix} x_{aug}(k) \\ e(k) \end{bmatrix} \end{cases} . \tag{10}$$

Eq. 10 is the ultimate combination of what we have done in part II-C, III-A, & III-B, and it also displays the insights of separation principle. So, we can directly design the K_{aug} to stabilize the closed-loop augmented system with the design of L remaining the same as it was in part II-C.

By the use of MATLAB command `lqr`, we obtain the feedback gain of the augmented system

$$\begin{aligned} K_{aug} &= \begin{bmatrix} K_{1\times 4} & K_{m,1\times 2} \end{bmatrix} \in \mathbb{R}^{1\times 6} \\ , K &= \begin{bmatrix} -9.5192 & -38.6430 & -3.6387 & -3.3539 \end{bmatrix} \\ , K_m &= \begin{bmatrix} 0.0024 & -0.0024 \end{bmatrix} \end{aligned}$$

and

$$\begin{aligned} \lambda(A_{aug} - B_{aug}K_{aug}) =& 0.4034 + i0.0000, 0.9786 + i0.0000 \\ &, 0.9998 + i0.0063, 0.9998 - i0.0063 \\ &, 0.9960 + i0.0028, 0.9960 - i0.0028 \end{aligned}$$

, where $\forall|\lambda(A_{aug} - B_{aug}K_{aug})| < 1$ (stable), and the $\big(Q_c = \mathrm{diag}(200, 1111, 2, 2, 10^{-9}, 10^{-9}),\ R_c = 1\big)$ in LQR control algorithm are obtained by tuning.

Finally, we finish the design of state-observer feedback controller with internal model. The configuration in Simulink in shown in Fig. 6.

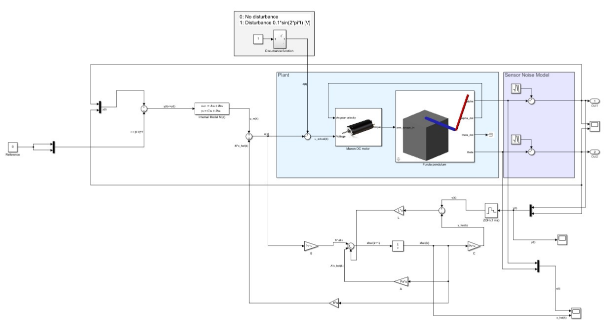

Fig. 6: Configuration of State-Observer Feedback Controller with Internal Model in Simulink.

IV. Results Discussion

A. Separation Principle

Compare the observer error pole $\lambda(A - LC)$ and the closed-loop augmented system poles $\lambda(A_{aug} - B_{aug}K_{aug})$ already designed in part III-C, these two are both stable and unaffected by one another (decoupled). Hence, the observer does not affect the closed-loop system in the state-observer feedback control scheme.

B. Sensor Output v.s. Estimator Output Response (with State-Observer Feedback Control Only)

For the results of the state-observer feedback controller *without* internal model which is shown in Fig. 7, we can see that the estimated pendulum angle $\hat{\theta}$ is still fluttering with comparatively large amplitude; that is, the system is still suffering from the sinusoidal input disturbance.

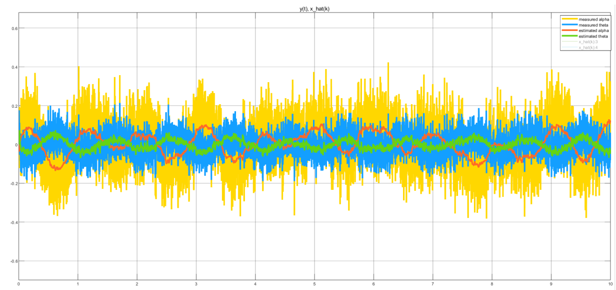

Fig. 7: Sensor Output v.s. Estimator Output Response (*with* State-Observer Feedback Control Only).

C. Sensor Output v.s. Estimator Output Response (with State-Observer Feedback Control and Internal Model)

For the results of state-observer feedback controller *with* internal model which is shown in Fig. 8, we can see that the estimated pendulum angle $\hat{\theta}$ still flutters yet its amplitude decreases overtime; that is, the controller has the ability to reject the sinusoidal input disturbance.

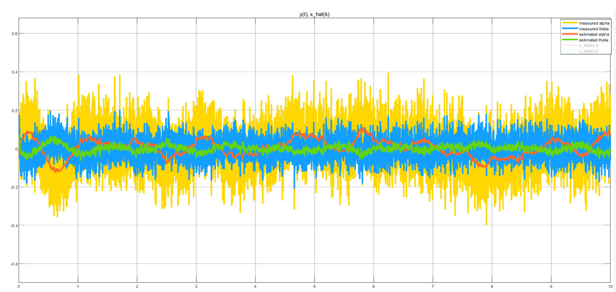

Fig. 8: Sensor Output v.s. Estimator Output Response (*with* State-Observer Feedback Control and Internal Model).

D. Estimator Output v.s. Actual Output Response (with State-Observer Feedback Control and Internal Model)

For the results for state-observer feedback controller *with* internal model which is shown in Fig. 9, we can see that the estimated states successfully tracks the actual states; that is, the state observation becomes more accurate with the design of internal model since it provides the ability to reject the sinusoidal input disturbance which the system would not have if no internal model included. Thus, the statement we mentioned in the end of part II-C is correct.

V. Conclusion

In this project, we successfully implement the state-observer feedback controller with internal model on a Furuta pendulum system. During our work, we showed that the design procedure could be separated into regulator problem and sinusoidal input disturbance rejection, which the two parts could be design respectively and combined together as an augmented system. The significance of the internal model lies not only on the ability for input disturbance rejection but also on its capability of making state observations more accurate with respect to the actual states.

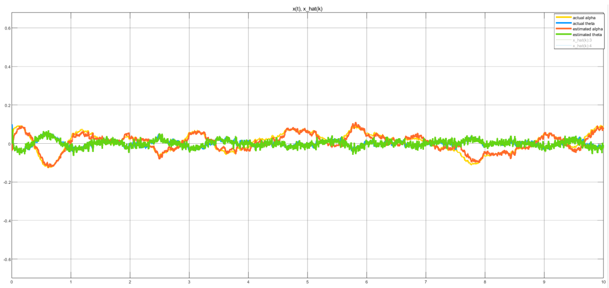

Fig. 9: Estimator Output v.s. Actual Output Response (*with* State-Observer Feedback Control and Internal Model).

REFERENCES

[1] A. N. M. Panos J. Antsaklis, "A linear systems primer," 2007.
[2] H. YI LUN, "Linear systems final project documentation," 2022.