Repetitive control for Inverted Double Pendulum *

Huangshuai Shi
UCLA MAE Department
shishi2020@g.ucla.edu

Mingzhang Zhu
UCLA MAE Department
normanzmz@g.ucla.edu

Ruoning Ren
UCLA MAE Department
ruoningr@ucla.edu

Jiahui Xi
UCLA MAE Department
jiahui0523@ucla.edu

***Abstract*—This report demonstrates a complete process of how our team design and implements an integration of repetitive and feedforward controller on a spring-connected inverted double pendulum plant. System Identification of both motors and pendulums were performed to access the accurate mathematical model of the plant. The repetitive and feedforward controller were designed through pole placement and LQG. The simulation and experimental results showed that the pendulum followed all desired trajectories (Sine, triangular, square, step)with small error, which indicates the robustness and accuracy of our controller.**

***Index Terms*—component, formatting, style, styling, insert**

I. Introduction

Undoubtedly, the double inverted pendulum system plays a vital role in control research due to its overall ease of construction and testing. As a result, specific estimation models utilized with this system can be confirmed using a tangible system, and these identical estimation models can be implemented with a variety of more intricate mechanical systems.

Many industrial applications require control systems to track or reject periodic exogenous signals, which may be desired reference outputs or unknown disturbances. Examples of such applications include the periodic motion of robot manipulators, repeatable run-out in disc drives, torque ripples in harmonic drives, and periodic force disturbances in metal cutting. The control method that can successfully guide the system to track or reject periodic signals is known as repetitive control.

Two approaches are being used generally for repetitive control design. The first approach is based on the internal modelling principle, where a time delay internal model is placed inside the loop, and a feedback controller is designed to stabilize the plant together with the internal model. However, this approach has some disadvantages such as reduced robustness to measurement noise and unmodelled dynamics, and the computer requiring a lot of memory space to store signal values over the previous period, especially if the signal's period is long. The second approach is the basis function approach or adaptive feedforward cancellation (AFC) approach, where the periodic exogenous signal is modelled as a linear combination of finite or infinite basis functions with unknown coefficients. The AFC approach does not alter the system loop gain and can be designed independently to maintain rejection of disturbances with other frequencies and enhance robustness to unmodelled dynamics. In this paper, we choose the second approach to design our repetitive controller, which also compensates the unmodelled non-linearies well such as damping between the spring and two pendulums, inertia loss due to drilled holes, etc.

The remainder of this paper is organized as follows: Section II describes the how we characterize this problem to a mathematical model, Section III details the approach of system identification for different parameters, Section IV indicates the design of SOFCI and Repetitive Controller and feedforward filter, Section V analyzes the controller performance, and Section VI shows the simulation and experimental results of different trajectories.

II. System Model

A. Rigid Body Modeling

The experimental setup places the spring in between two pendulums shown in Fig 1. Only one pendulum is connected to the drive motor. When the motor is activated, the driving pendulum swings into a vertical inverted pendulum position, the paired pendulum is consequently spun into its vertical inverted position by the connecting spring.

However, the spring may not simply undergo a linear extension. The spring may be subject to torsional influence from the moment provided by the arm and motor torque. For this reason, when modeling the entire system, the spring constant k may not be applicable, since torque instead of force is applied to the spring. Thus, the relationship between torque applied on the spring and the difference in rotation angle of the two pendulums must be derived.

To understand how the spring would react to the torque output from the motor in the experimental configuration, a kinematic model based on the physical system is created and shown in the figure below. Here a new constant k_c is introduced. Similar to k, k_c is a torsional spring constant that relates torque and angular displacement.

B. System Dynamics

The full system dynamics was derived with the following relations. The double pendulum configuration has the following linear dynamic model using parameters in Table 1:

$$\begin{aligned} J_{pend_1}\frac{d\omega_1}{dt} &= K_{m_1}\frac{(-K_{m_1}\omega_1+V_{s_1}u_1)}{R_1} \\ &\quad -(m_1 g l_{c_1} cos\theta_{e_1})\theta_1 - T_{d_1} \end{aligned} \tag{1}$$

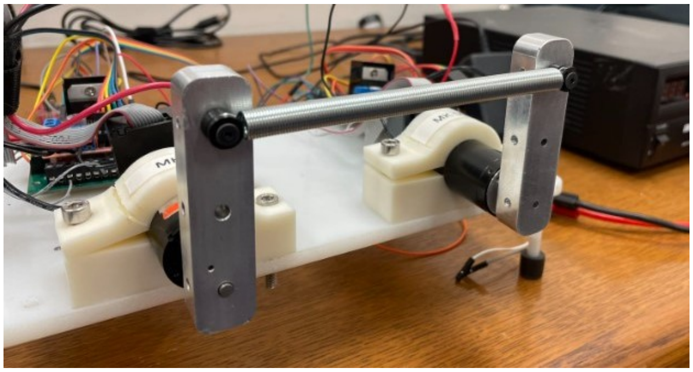

Fig. 1. Spring Connected Inverted Double Pendulum Configuration

$$J_{pend_2}\frac{d\omega_2}{dt} = K_{m_2}\frac{(-K_{m_2}\omega_2 + V_{s_2}u_2)}{R_2} - (m_2 g l_{c_2} cos\theta_{e_2})\theta_2 - T_{d_2} \tag{2}$$

Denoting k_c as the equivalent torsional spring constant, the pendulum's model can be derived by the state equation of the single pendulum:

$$\dot{x_1} = Ax_1 + Bu_1 - B_d T_{d_1} \tag{3}$$

$$\dot{x_2} = Ax_2 + Bu_2 - B_d T_{d_2} \tag{4}$$

$$y_1 = Cx_1 \tag{5}$$

$$y_2 = Cx_2 \tag{6}$$

$$T_{d_1} = (y_1 - y_2)k_c + T_w \tag{7}$$

$$T_{d_2} = (y_2 - y_1)k_c \tag{8}$$

Using equations (3) to (8), the standard state space form is found as follows:

$$\begin{bmatrix}\dot{x_1}\\ \dot{x_2}\end{bmatrix} = \begin{bmatrix}A_1 - k_c B_{d1} C_1 & k_c B_{d1} C_1\\ k_c B_{d2} C_2 & A_2 - k_c B_{d2} C_2\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix} + \begin{bmatrix}0\\ B_2\end{bmatrix}u - \begin{bmatrix}B_{d1}\\ 0\end{bmatrix}T_w \tag{9}$$

$$y = \begin{bmatrix}C & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix} \tag{10}$$

where,

$$\dot{x_1} = \begin{bmatrix}\dot{\theta}_1\\ \dot{\omega}_1\end{bmatrix}, \dot{x_2} = \begin{bmatrix}\dot{\theta}_2\\ \dot{\omega}_2\end{bmatrix} \tag{11}$$

$$A_1 = \begin{bmatrix}0 & 1\\ \frac{-m_1 g l_{c_1} cos\theta_{e_1}}{J_{pend_1}} & \frac{-K_{m_1}^2\omega_1}{J_{pend_1}R_1}\end{bmatrix} \tag{12}$$

$$A_2 = \begin{bmatrix}0 & 1\\ \frac{-m_2 g l_{c_2} cos\theta_{e_2}}{J_{pend_2}} & \frac{-K_{m_2}^2\omega_2}{J_{pend_2}R_2}\end{bmatrix} \tag{13}$$

B_1 is 0 since there is no u_1 because only u_2 corresponds to the driving motor.

$$B_2 = \begin{bmatrix}0\\ \frac{V_{s_2}}{J_{pend_2}R_2}\end{bmatrix} \tag{14}$$

$$B_{d_1} = \begin{bmatrix}0\\ \frac{-1}{J_{pend_1}}\end{bmatrix}, B_{d_2} = \begin{bmatrix}0\\ \frac{-1}{J_{pend_2}}\end{bmatrix} \tag{15}$$

III. System Identification

In order to increase the performance of any controller implemented onto the physical system, a system identification must be performed beforehand to obtain precise numbers for the necessary parameters of the spring-connected double pendulum system. A series of experiments were conducted to determine different variables for the physical system we tested.

The inertia of the pendulum was obtained through its CAD model. The voltage of the motor is determined to be 10.7v using a voltmeter. The original supply voltage is decreased by two transistors of the H-bridge. The viscous damping coefficient, b and motor coil winding inductance, L are neglected due to the reduced order of the system model for the controller design. The mass of the pendulum is proved to be the mass of the nominal model.

A. Gray Box Method

For the parameters that are difficult to physically measure, gray box method was implemented to find motor constant K_m, motor resistance R, and rotor inertia J_{rotor}. Two experiments were accomplished to find the value of κ and τ: a step response with an exponential fit and a chirp response using Matlab toolbox tfest, which is a command specifically for estimating the transfer function model. After cross-checking the performance of the step response of the physical plant, we chose to adapt the parameters we got from tfest method. The detailed steps of how we determined those values are described in our Assignment 1 report.

B. Linear Spring Constant k

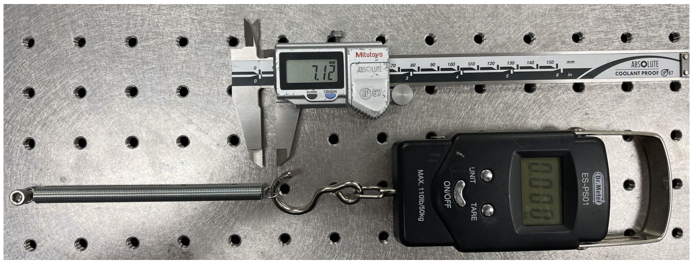

Fig. 2. Experimental Setup to Measure the Spring Constant K

We determined the linear spring constant k using the experimental setup as shown in Fig 3. A fishing scale was used to measure the exerted force and a caliper was used to measure the displacement. We found that this spring has a large 'dead zone', the spring will not generate any displacement until the force reach about 3N. However, based on our observation, the force exerted on the spring during step response will not reach that threshold value. In order to get the value of k_c, we picked the threshold value. When 3N was exerted on the spring, the generated displacement is about 0.75mm. In this case, the estimated spring constant in our case is about 4000 N/m.

C. Torsional Spring Constant k_c

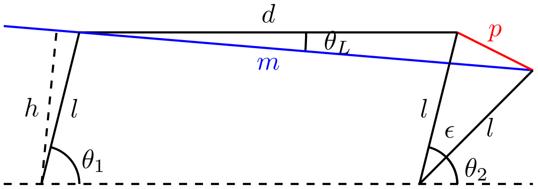

Fig. 3. System Model for k_c Derivation

A small angle model is built to help calculate the approximated k_c value based on k value. First, set ϵ as the difference in angle between the two pendulums. i.e $\epsilon = \theta_1 - \theta_2$. When $\epsilon = 0$, the spring is at its original length, and the active pendulum does not exert any force on the passive pendulum. However, when epsilon is not zero, the passive pendulum will be subjected to a force from the spring, which will be converted into a torque. From the configuration in Fig 3., torque T applied by the spring can be calculated by equation (1):

$$T = hf = hk\Delta x = hk(m - d) \tag{16}$$

where d is the spring length before stretch and m is the stretched length. k is the linear spring constant and h is length of the force arm. The following four equations help us get a function of Torque:

$$p = 2l \cdot sin(\frac{\epsilon}{2}) \tag{17}$$

$$m = \sqrt{p^2 + d^2 - 2pd \cdot cos(\frac{\pi - \epsilon}{2} + \theta_1)} \tag{18}$$

$$\theta_L = arcsin(\frac{p \cdot sin(\frac{\pi-\epsilon}{2} + \theta_1)}{m}) \tag{19}$$

$$h = l \cdot sin(\theta_1 + \theta_L) \tag{20}$$

Thus, equation (16) can be simplified to a function of T by substituting all the constant, the variables are θ_1 and ϵ.

$$T = f(\theta_1, \epsilon) \tag{21}$$

ϵ can be seen as a small angle belongs to [-0.1, 0.1]. Thus, for each θ_1 the relation between ϵ and T is linear, and the slope of this linear function is the approximated value of k_c. Since the planned trajectory signal for the pendulum is between 0 and 0.4, a list of k_c is calculated when θ_1 is between $\pi/2 - 0.4$ and $\pi/2$ and the plot is shown in Fig 4. The mean value of this k_c list is equal to 9.7728, choose k_c = 10 as the approximated constant value.

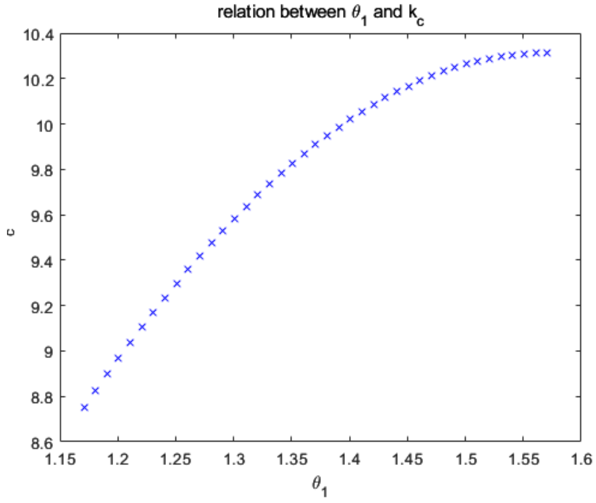

Fig. 4. plot of k_c value when θ_1 in $[\frac{\pi}{2} - 0.4, \frac{\pi}{2}]$

TABLE I
SYSTEM IDENTIFICATION SUMMARY

Parameters	***Unit***	***Nominal Value***	***Motor 1***	***Motor 2***
Damping Coeff. b	kgm^2/s	0	0	0
Resistance R	Ω	3.85	3.85	3.8
Supply Voltage V_s	V	10.7	10.7	10.58
Motor Constant k	$Nm/(radA)$	0.024	0.0258	0.0195
Motor Mass m	kg	0.044	0.044	0.046
Motor Inertia	kgm^2	1.67e-6	5.57e-5	4.56e-5
Pendulum Inertia	kgm^2	/	7.53e-5	8.41e-5
Inductance L	$kgm^2/(s^2A^2)$	2.3e-4	2.3e-4	2.3e-4
Spring Constant k_c	Nm/rad	10	N/A	N/A

IV. CONTROLLER DESIGN

A. State Feedback Observer Control with integral action

Due to the current physical plant's instability, a stable plant must be obtained prior to the design of subsequent repetitive controller and feedforward filter. Therefore, a state feedback observer controller with an integral action was first designed. The coresponding block diagram is shown in Fig 5. The SOFC incorporates an integral term in the model input to achieve zero steady-state error. The controller uses the Linear Quadratic Regulator method to calculate the controller gain and the pole-placement method to calculate the observer gain. The pole locations and gain results are recorded in Table 2.

B. Repetitive Controller and Feedforward Filter Design

With the SOFCI controller, the whole system can now be seen as a stable plant, which indicates that the inverted plant method can be applied. F filter in both repetitive and feedforward path can be computed through this method. Suppose stable plant:

$$P = \frac{z^{-d}B^{+}(z^{-1})B^{-}(z^{-1})}{A(z^{-1})} \tag{22}$$

TABLE II
IDENTIFIED PARAMETERS FOR BOTH PLANTS

Parameters	***Nominal Value***	***Motor 1***	***Motor 2***
0.8694 + 0.4769i	0.8026	0.5395	4.0610
0.8694 - 0.4769i	0.8026	-100.0788	0.0707
0.9564 + 0.0385i	0.8026	-0.2603	3.7281
0.9564 - 0.0385i	0.8026	101.0086	0.1142

TABLE III
POLE LOCATION AND GAIN VALUES OF SOFCI

Poles(A-BK)	***Poles(A-LC)***	***Observer Gain***	***Controller Gain***
0.8694 + 0.4769i	0.8026	0.5395	4.0610
0.8694 - 0.4769i	0.8026	-100.0788	0.0707
0.9564 + 0.0385i	0.8026	-0.2603	3.7281
0.9564 - 0.0385i	0.8026	101.0086	0.1142

Then the inverted plant filter F can be computed as:

$$P = \frac{z^d A(z^{-1})}{B^+(z^{-1})} * \frac{B^-(z)}{\gamma} \tag{23}$$

where $\gamma = B^{-^2}(1)$ Then the outside controller can be constructed by the Fig 6: In this diagram, Q filter is a low pass filter:

$$Q = (\frac{z + 2 + z^{-1}}{4})^{N_2} \tag{24}$$

In this case, N_2 is chosen to be 100 to filter enough high frequency components. N_1 is set to equal to 3 to make F filter casual. And N_p is equal to the number of reference points in 1 period.

V. CONTROLLER ANALYSIS

To understand the performance of independent controllers have on the system plant (here it is assumed the SOFCI block being the plant), several aspects are looked at. First, the loop gains of the stable plant SOFCI and with Repetitive control implementation is compared in Fig 7. In the loop gain bode magnitude plot, the spike is resulted from the system plant. The asymptotic increases in loop gain for the system with repetitive control exist at 6.28 rad/s, 12.6 rad/s 18.8 rad/s and so on, all at frequencies that are integer multiples of 6.28 rad/s which correspond to multiples of 1Hz. This is an indication that the repetitive control is robust against periodic signals.

With repetitive control implementation, a very significant factor that decides whether or not the controller for the system is robust for a wide frequency range is Q filter design. As previously mentioned in section IV. equation (24), the power of N_2 can be selected for filtering high frequency components. In Fig 8. is a sensitivity comparison of Repetitive Control with different Q filter designs. Presented are 3 different Q filters with powers 5, 10 and 50, and because of cascading filter within the system block diagram, the resulting powers of the Q filter is 10, 20 and 100 respectively. While maintaining the sensitivities for lower frequencies as well as the asymptotic dips that produce resistance to periodic inputs that occur at frequencies of integer multiples of 6.28 rad/s, it is clear that as a Q filter with higher power is implemented, the sensitivity of the high frequency range is improved.

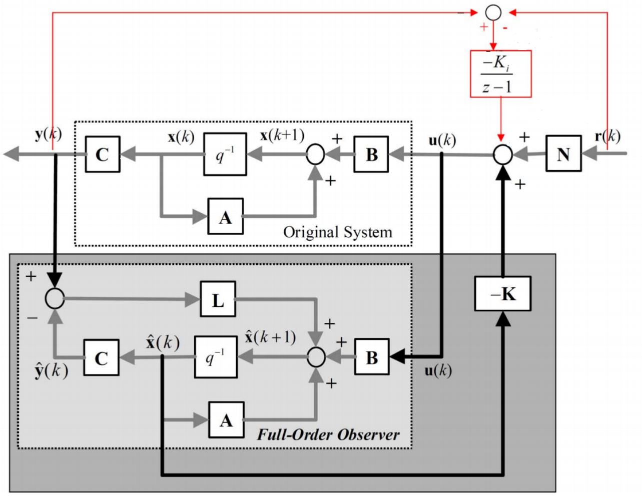

Fig. 5. Block Diagram of SOFCI

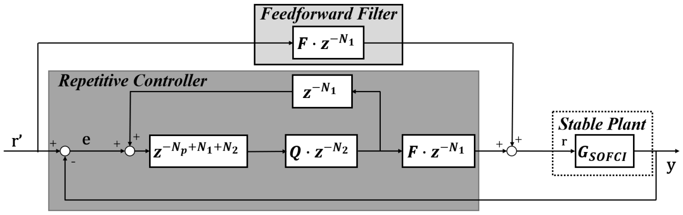

Fig. 6. Block Diagram of Repetitive control

A similar trend is shown in Fig 9. With the addition of the Feed Forward control, the sensitivity of low frequency is greatly improved as the peak sensitivity from 0 to 10 rad/s is well maintained under -10 db. The effect of Q filter design is present since the controller is a combined version of both repetitive and feed forward control.

Furthermore, when a cross comparison between the aforementioned control architecture is generated, the advantage of SOFCI with repetitive and feed forward control is pronounced. In Fig 10. the sensitivity of SOFCI has a peak of around 4.85 db at 100 rad/s. With the repetitive control implementation, although a higher sensitivity magnitude overall of 6.93 db is observed, the system will respond well to periodic signals as indicated by the asymptotic dips that occur at frequencies that are integer multiples of 6.28 rad/s. While with only repetitive control implementation, the system reacts better at periodic signals as compared to the original SOFCI control, the control method with both repetitive and feed forward control implementation shows great promise at maintaining a good low frequency sensitivity as well as an impressive response to periodic signals, preserving the advantage of repetitive control.

The final control architecture for the system settled on the SOFCI with repetitive and feed forward control implementation. The eventual design of Q filter used in the control system

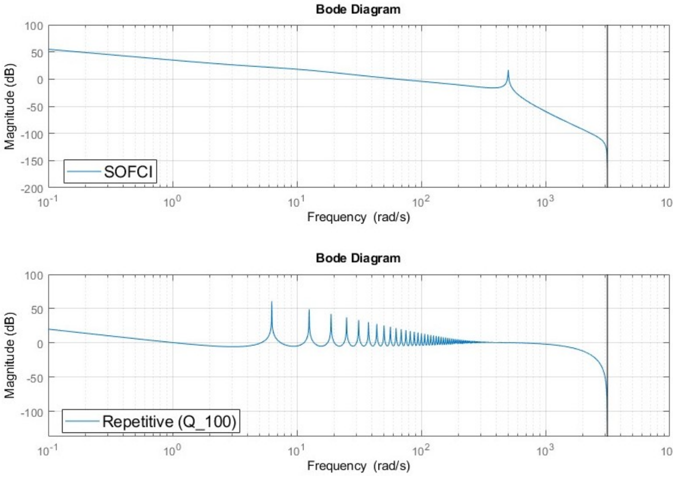

Fig. 7. Loop Gain for SOFCI and SOFCI with Repetitive Control

is with power 50. The simulation and experimental results are discussed in section VI.

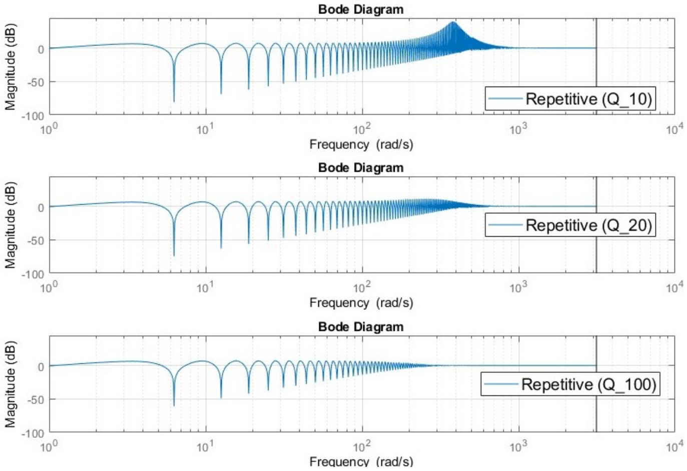

Fig. 8. Sensitivity Comparison of Repetitive Control with Different Powers in Q filter

VI. Simulation and Experimental Results

A. Simulation Results

The simulation result of different wave types (sine, triangle, square) and its error bar is shown in Fig 11. It can be found that the error of sine and triangular waves are steady and small than 0.05, while the square wave has a relatively large overshoot, which due to the trade off of our controller design. Fig 12 shows the comparison of the simulation under different frequencies (0.5Hz, 1Hz, 5Hz). It can be seen that the error at 5Hz is larger than 0.5Hz and 1Hz. We also tested the robustness of our controller as shown in Fig 13. A 5Hz sin wave was added to the sampling 1Hz triangular wave. The result shows that the error only slightly larger than the original one, which indicates the good robustness of our controller.

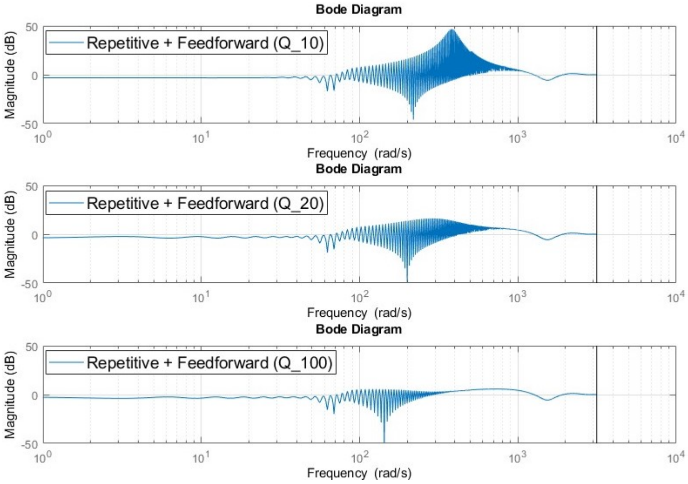

Fig. 9. Sensitivity Comparison of Repetitive and Feed Forward Control with Different Powers in Q filter

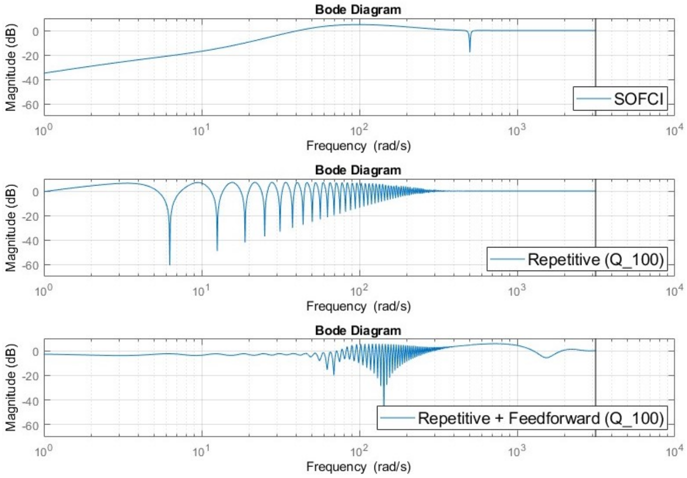

Fig. 10. Sensitivity Comparison between SOFCI, Repetitive Control and Repetitive + Feed Forward Control

B. Experimental Results

To verify the advantage of our repetitive + feedforward controller, we performed tests on four different controllers (SOFCI, repetitive, feedforward, repetitive + feedforward) with the same input waves. As shown in Fig. 14, The repetitive + feedforward and pure repetitive controller have much smaller steady state error than SOFCI and feedforward controller. The better performance is largely due to the repetitive part of the controller. Fig. 15 shows the comparison of the simulation under different frequencies (0.5Hz, 1Hz, 5Hz). It can be seen that the error at 5Hz is larger than 0.5Hz and 1Hz. The simulation result of different wave types (sine, triangle, square) and its error bar is shown in Fig 11. It can be found that the error of sine and triangular waves are much smaller than square wave, which matches the simulation result we got in previous section.

A swing up tasks was also tested on the actual system. The pendulum successfully swing 180 degrees from the bottom

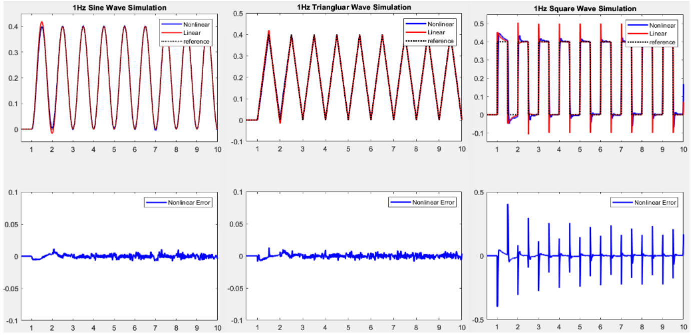

Fig. 11. Comparison of the simulation under different wave type

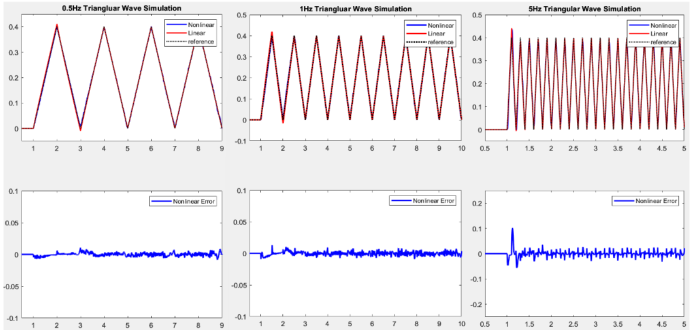

Fig. 12. Comparison of the simulation under frequencies

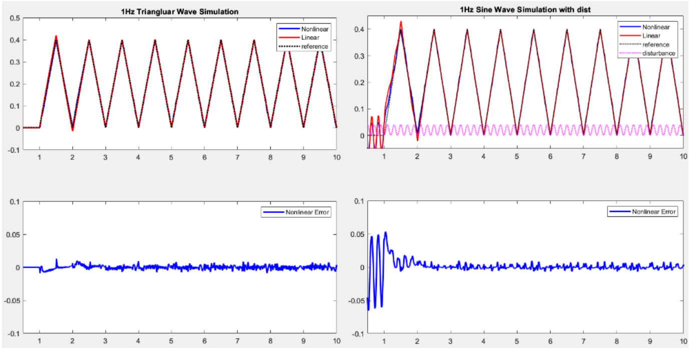

Fig. 13. Comparison of the simulation under 5Hz sin wave disturbance

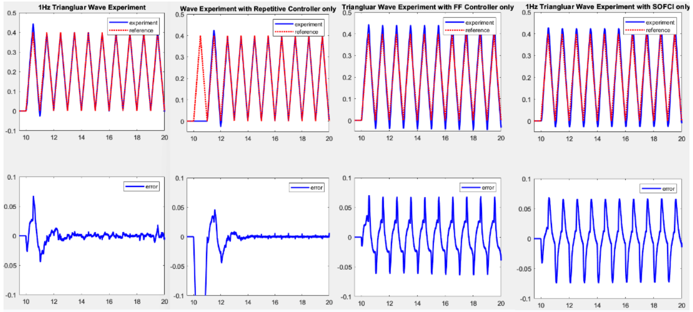

Fig. 14. Comparison of the experimental results under different controllers

to the top in 0.2 seconds, which satisfy the initial requirement. We observed that the swing motion has relatively large overshoot compared with the 0.4 rad step response we tested previously. We believed that this phenomena is mainly due to a large disturbance induced in the singularity point while swinging. If we treated the whole system as a four bar linkage, when the cranker and follower are aligned in the same line, the error between two pendulum is maximized, which means the spring is forced to be extended over the threshold. However, the system still swing to its desired position in a short period of time, indicating the robustness of our repetitive + feedforward controller.

VII. Conclusion

In this report, our team presents a comprehensive overview of the design and implementation process of integrating a repetitive and feedforward controller onto a spring-connected inverted double pendulum plant. We conducted a System Identification analysis of both motors and pendulums to obtain an accurate mathematical model of the plant. The design of the repetitive and feedforward controller involved utilizing pole placement and LQG methods. Our simulation and experimental results indicated that the pendulum successfully followed desired trajectories, such as sine, triangular, square, and step, with minimal error. These findings demonstrate the robustness and accuracy of our controller design.

In future work, the system identification can be detailed to some non-linearities that are not considered in this report to minimize the error between the mathematical model and actual plant. Besides, the controller algorithm of both LQG and pole placement can be optimized to increase the controller performance.

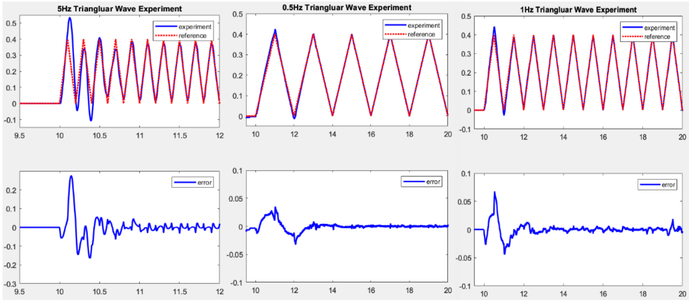

Fig. 15. Comparison of the experimental results under different frequencies

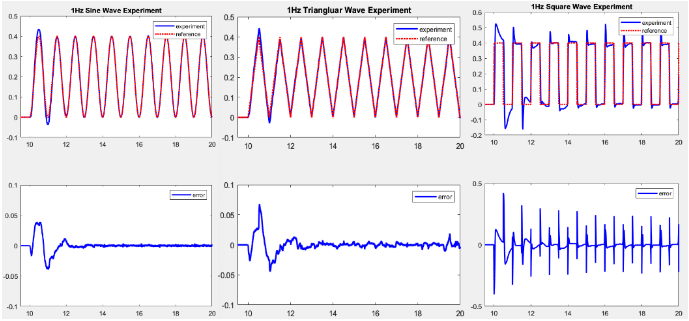

Fig. 16. Comparison of the experimental results under different wave types