DEPLOYING STATE SPACE CONTROL TO REGULATE AN INVERTED ROTARY PENDULUM WITH NON-ZERO INPUT

Bui, Hoang Dung

*Department of Mechanical Engineering and Material Science, Faculty of International Training,
Thai Nguyen University of Technology, 666, 3-2 Street, 251812 Thai Nguyen City, Vietnam*

---***---

Abstract - *State space control has a wide application for multiple inputs and multiple outputs systems. In several researches, it is used to control Inverted Rotary Pendulum (IRP) whose reference inputs are zero. In this paper, this control algorithm is deployed to regulate an IRP working when the arm is at non-zero position. The controlling challenge is the IRP state vector has to be extended to deal with the arising error. As consequently, the IPR's parameter matrices are also expanded to handle with the new vector. Base on that, a state space control law is built to implement the regulation by LQR method. The calculated control law is tested by simulation and experiments. The results present the availability of the LQR approach in controlling the IRP with nonzero input.*

Key Words: Non-zero Input, State Space Control, LQR, Inverted Rotary Pendulum

1.INTRODUCTION

Inverted Rotary Pendulum (IRP) is a nonlinear under-actuated mechanical system which is well-suited to verify and practice the ideas of control theory [1][2]. There were several control methods applied on IRP models and their responses to hold the pendulum at upright position were good. [1]-[5].

In some researches, real IRP models were built and controlled by some control methods such as State Space Control, PID, Fuzzy Control [3]-[5]. Those control algorithms regulated the IRP and stabilized it when the pendulum is at the upright position. However, all the researches have only focused on controlling the IRP as its arm stays at the zero position. It means the arm can work only in one position in the whole operating range (0° ÷ 360°).

This paper is to find a way to regulate the IRP to work with non-zero input for the arm position. In order to do that, State Space Control method is used to determine a control law. To handle with the error causing by the nonzero position of the arm, one more state is added to the IRP state vector, increasing the considered states into five. This will make the controller more complicated.

In the following sections, we will set up the dynamic equation systems of the IRP, establish an appropriate control law, build a model and simulate it, and finally make experiments by applied the control law in a real IRP.

2. IRP's DYNAMIC EQUATION SYSTEM

IRP consists of two rotary axes whose positions are the two system's outputs. Its structure is shown in the Fig - 1 with an arm with length ρ and a pendulum bar l rotating around the Z-axis and Y-axis. The rotation of the arm and bar are called angles α and ϕ, respectively. In steady state, the mass m stays at the upright position (ϕ = 0) and the arm's position α stands at a desired position.

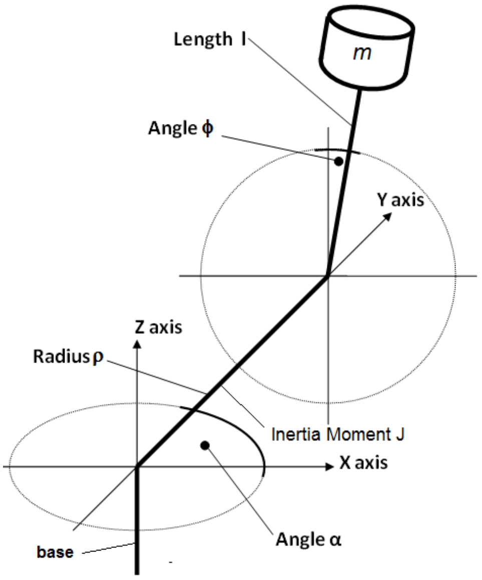

Fig - 1: An IRP structure

The pendulum is driven by a DC motor via a gearbox and a belt transmission. As angles α and φ change, the differences between the values and the desired inputs are used as the input for the controller. With the parameters above, the IRP dynamic equations have built by author's master thesis [6], and the results are presented here:

$$\begin{cases}\left(J+m\rho^2+ml^2\sin^2\phi\right)\ddot{\alpha}-(m\rho l\cos\phi)\ddot{\phi}+(m\rho l\sin\phi)\dot{\phi}^2+ \\ \qquad +(2ml^2\sin\phi\cos\phi)\dot{\alpha}\dot{\phi}=M_{in} \\ ml^2\ddot{\phi}-(m\rho l\cos\phi)\ddot{\alpha}-(ml^2\sin\phi\cos\phi)\dot{\alpha}^2-mgl\sin\phi=0\end{cases} \quad (1)$$

MM_{in} is the torque driven from DC motor.

The IRP only works around the upright position with small deviation ϕ and α varies slowly around desired position. Therefore an approximated model is built by linearization around the working point. Before doing that, a state vector is established by introducing some new variables (α_1, ϕ_1, α_2, ϕ_2):

$$\alpha_1 = \alpha; \quad \phi_1 = \phi; \quad \alpha_2 = \dot{\alpha}; \quad \phi_2 = \dot{\phi}$$

$$\rightarrow \dot{\alpha}_1 = \alpha_2; \quad \dot{\phi}_1 = \phi_2; \quad \dot{\alpha}_2 = \ddot{\alpha}; \quad \dot{\phi}_2 = \ddot{\phi}$$

Introduce state vector $\underline{x}$:

$$\underline{x} = [\alpha_1 \quad \phi_1 \quad \alpha_2 \quad \phi_2]^T \tag{2}$$

Linearizing the equation system (1) around the working point ($\phi \approx 0$; $\alpha \approx 0$), $\sin\phi \approx 0$; $\sin\alpha \approx 0$, $\cos\phi \approx 1$; $\cos\alpha \approx 1$, results in:

$$\begin{bmatrix} \dot{\alpha}_1 \\ \dot{\phi}_1 \\ \dot{\alpha}_2 \\ \dot{\phi}_2 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \dfrac{2m\rho g}{J} & 0 & 0 \\ 0 & \dfrac{\left(J + m\rho^2\right)g}{Jl} & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} \alpha_1 \\ \phi_1 \\ \alpha_2 \\ \phi_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \dfrac{1}{J} \\ \dfrac{\rho}{Jl} \end{bmatrix} \cdot M_{in} \tag{3}$$

Combination with DC motor and gearbox: The IRP is driven by a DC motor with gearbox and belt transmission which are shown in Fig - 2:

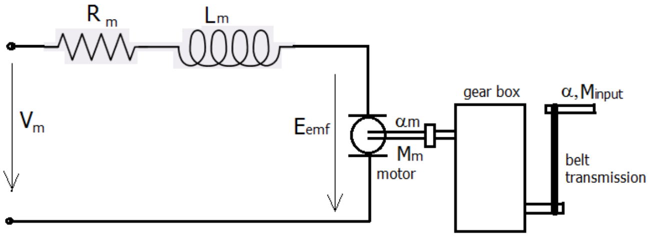

Fig - 2: DC motor circuit with gearbox - belt transmission

From the DC motor electrical circuit and the transmission via gear box and the belt, the relationship between the input voltage, arm angle α_1 and M$_{in}$ is:

$$M_{in} = \frac{\eta K_1}{R_m} V_m - \frac{\eta K_2 K_m}{R_m} \dot{\alpha}_1 \tag{4}$$

where $\eta = \eta_g \eta_m \eta_b$, $K_1 = K_g K_b K_t$; $K_2 = K_g^2 K_b^2 K_t$.

(R_m: resistance of the motor, V_m,: supplied voltage, M_{in},: output torque of the belt transmission, η_m, η_b, η_g: efficiencies of DC motor, belt transmission, gearbox, K_m, K_t: motor & torque constant, K_b, K_g transmission ratios of the belt and gearbox)

Substitute Equation (4) into Equation (3):

$$\begin{bmatrix} \dot{\alpha}_1 \\ \dot{\phi}_1 \\ \dot{\alpha}_2 \\ \dot{\phi}_2 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & P_1 & -P_3 & 0 \\ 0 & P_2 & -P_4 & 0 \end{bmatrix} \cdot \begin{bmatrix} \alpha_1 \\ \phi_1 \\ \alpha_2 \\ \phi_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ P_5 \\ P_6 \end{bmatrix} \cdot V_m \tag{5}$$

where $P_1 = \dfrac{2m\rho g}{J}$; $P_2 = \dfrac{\left(J + m\rho^2\right)g}{Jl}$; $P_3 = \dfrac{\eta K_2 K_m}{JR_m}$;

$$P_4 = \frac{\rho\eta K_2 K_m}{JlR_m}; \quad P_5 = \frac{\eta K_1}{JR_m}; \quad P_6 = \frac{\rho\eta K_1}{JlR_m}$$

As the dynamic equations get done, the features of the IRP can be verified.

3. CONTROL LAW

Before finding out an appropriate control law, the IRP is inspected for *controllability* and *observability*.

Introducing matrices $\underline{\underline{A}}$, $\underline{\underline{B}}$,and $\underline{\underline{C}}$:

$$\underline{\underline{A}} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & P_1 & -P_3 & 0 \\ 0 & P_2 & -P_4 & 0 \end{bmatrix}; \quad \underline{\underline{B}} = \begin{bmatrix} 0 \\ 0 \\ P_5 \\ P_6 \end{bmatrix}; \underline{\underline{C}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix};$$

3.1 Controllability

It is a property, which enables one to steer the dynamic system to a desired trajectory [8]

The IRP system is controllable if and only if the matrix

$$\underline{\underline{Q}}_c = \left[\underline{\underline{B}} \quad \underline{\underline{AB}} \quad \underline{\underline{A}}^2\underline{\underline{B}} \quad \underline{\underline{A}}^3\underline{\underline{B}}\right]$$

is non-singular [8].

The controllability of IRP is checked at section 4 when all parameters are substituted in the formula.

3.2 Observability

It is a property, with which the system state trajectory can be deduced from the input and output trajectories [8].

The IRP is observable if and only if the matrix

Where $\underline{\underline{Q}}_o = \left[\underline{\underline{C}} \quad \underline{\underline{C}}\,\underline{\underline{A}} \quad \underline{\underline{C}}\,\underline{\underline{A}}^2 \quad \underline{\underline{C}}\,\underline{\underline{A}}^3\right]^T$

is non-singular [8].

Multiplying two matrices $\underline{\underline{C}}$ and $\underline{\underline{A}}$ with the form above, lead to:

$$\underline{\underline{Q}}_o = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & P_1 & -P_3 & 0 \\ 0 & P_2 & -P_4 & 0 \\ 0 & -P_1P_3 & P_3^2 & P_1 \\ 0 & -P_1P_4 & P_3P_4 & P_2 \end{bmatrix}$$

Rank($\underline{\underline{Q}}_o$) = 4, so $\underline{\underline{Q}}_o$ is non-singular and the IRP system is observable.

3.3 Establishing an appropriate control law

The control parameters $\underline{K}$ is calculated by LQR approach with Matlab function *lqr()*. The function's inputs are four matrices: $\underline{\underline{A}}$, $\underline{\underline{B}}$, $\underline{\underline{Q}}$ and $\underline{\underline{R}}$, the last two matrices are used to balance the effort and responses of the system.

To control the IRP at a non-zero position of the arm, there is a new state added to the State Vector (2).

Introduce a new variable *e*:

$$e = \alpha_1 - r \tag{6}$$

where *r* is a non-zero desired position of the arm.

Differentiating Equation (6):

$$\dot{e} = \dot{\alpha}_1 = \alpha_2 \tag{7}$$

Introduce another new variable x_I with:

$$\dot{x}_I = e \tag{8}$$

A new state vector $\underline{x}_N$ is created by combining the State Vector (2) and the new one x_I:

$$\underline{x}_N = [\alpha_1 \quad \phi_1 \quad \alpha_2 \quad \phi_2 \quad x_I]^T$$

Because of changing the state vector, the IRP model can now be expressed as

$$\dot{\underline{x}}_N = \underline{\underline{A}}_N \underline{x}_N + \underline{\underline{B}}_N u_r$$

And their new parameters:

$$\underline{\underline{A}}_N = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & P1 & -P3 & 0 & 0 \\ 0 & P2 & -P4 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}; \quad \underline{\underline{B}}_N = \begin{bmatrix} 0 \\ 0 \\ P5 \\ P6 \\ 0 \end{bmatrix}$$

with five states and one input, the matrices $\underline{\underline{Q}}$ and $\underline{\underline{R}}$ have dimension *5x5* and *1x1*. The control law is adjusted by the educated trail-and-error-repetition technique until the performance of the controlled system is stable within 3 seconds. The adjustment gets done by changing the matrices $\underline{\underline{Q}}$ and $\underline{\underline{R}}$. After check and error, the most appropriate values of two matrices $\underline{\underline{Q}}$ and $\underline{\underline{R}}$ are:

$$\underline{\underline{Q}} = diag([0.5,2,0.2,0.5,1]); \qquad R = 0.5;$$

Applying *lqr()* function with the parameters above to find the control law.

$$\underline{K} = lqr(A_N, B_N, Q, R)$$

4. MODELING AND SIMULATION

Based on the dynamic equation and control law, the model is built on MatLab-Simulink.

4.1 Modeling

To ease the understanding, components of the IRP equations are grouped into a subsystem called PLANT. The others two parts are the integral one and feedback one. The structure is shown in the Fig - 3.

The states are multiplied with their own control parameters *K1, K3, K2, K4* and *K5*. The first four ones belong to Feedback part whose inputs are four states (positions and angular velocity of pendulum and arm) and its output is a part of feedback signal.

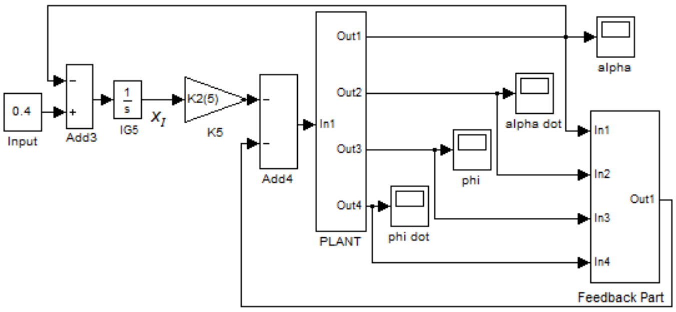

Fig – 3: Closed Loop Model

4.2 Simulation

Based on a real IRP, the parameters for the Simulink - model are:

K_t = 0.02 Nm/A, l = 0.3m, m = 0.03Kg, η_b = 0.98, J = 0.002 Kgm^2, ⍰ = 0.13m, η_g = 0.98, η_m = 0.51, g = 9.8 m/s^2, R_m = 3Ohm, K_g = 18, K_m = 0.02 Vs/rad, K_b = 4.

Substituting the system parameters to calculate the coefficients of the matrices $\underline{\underline{A}}$ and $\underline{\underline{B}}$:

$$P_1 \approx 36.4 \, {}^{1}\!/_{s^2}\, ; \; P_2 \approx 40.6 \, {}^{1}\!/_{s^2}\, ; \; P_3 \approx 161.2 \frac{1}{rad \cdot s}$$

$$P_4 \approx 69.9 \frac{1}{s \cdot rad}; \; P_5 \approx 111.9 \, {}^{1}\!/_{Vs^2}\, ; \; P_6 \approx 48.5 \, {}^{1}\!/_{Vs^2}$$

The next step is to check the IRP's controllability. Using the *det()* function from MatLab:

$$\det\left(\underline{\underline{Q}}_c\right) = -2.36e + 8 \neq 0\,;$$

Hence, rank($\underline{\underline{Q}}_c$) = 4 and the IRP is controllable.

The IRP is controllable and observable; therefore it fulfills the necessary conditions to design a control law.

Using *lqr()* function to get the control law:

$$\underline{K} = lqr(A_N, B_N, Q, R)$$

$$\underline{K} = [-2.92 \quad 50.5 \quad -4{,}1 \quad 10{,}1 \quad -1{,}41]^T$$

With these parameters, the IRP's stability needs to be verified. The system is stable if and only if all the poles and zeros are negative [8]. The poles and zeros are determined by function *eig()*:

$$ZP = eig\left(\underline{\underline{A}}_N - \underline{\underline{B}}_N * \underline{K}\right)$$

$$ZP=[-182.7\ -0.7+0.6i\ \ -0.7-0.6i\ \ -4.9+0.6i\ \ -4.9-0.6i]^T$$

The found result meet the mentioned condition, therefore the controlled IRP is stable.

To simulate the model, there are several prerequisites. Firstly, the controlled IRP only works in a small range *(-0.16 ÷ 0.16) rad* around the upright position and the mass M's velocity is small *(≤ 0.4m/s)*. The initial conditions for the model are: The desired inputs: $\alpha = 0.4$ *rad;* $\phi = 0$ *rad,* the initial positions of angles $\alpha_0 = 0.15$ *rad;* $\phi_0 = 0.1$ *rad* and two initial angular velocities $\dot{\alpha}_0 = 0m/s,\ \dot{\phi}_0 = 0.1m/s$.

The IRP model's responses are shown in Fig-4 and 5. The horizontal and vertical axes represent the simulation time (second) and the angle values (rad), respectively. Fig - 4 presents the behavior of the arm. From initial position $\alpha_0 = 0.15$ *rad*, it moves rapidly to the negative position to hold the mass M at the upright position. When the mass M is stable, the pendulum's arm reaches its desired position $\alpha_0 = 0.4$ *rad* after 2.8 seconds.

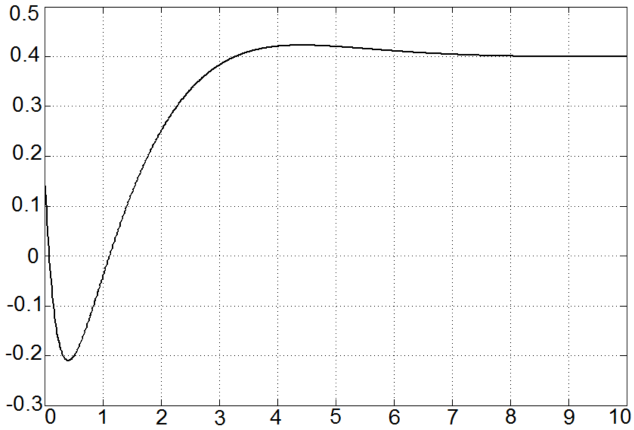

Fig – 4: Angle α response

Fig-5 shows the behaviors of the mass M. From initial position $\phi_0 = 0.1$ *rad,* it moved to the upright position and stayed there after 0.8 seconds.

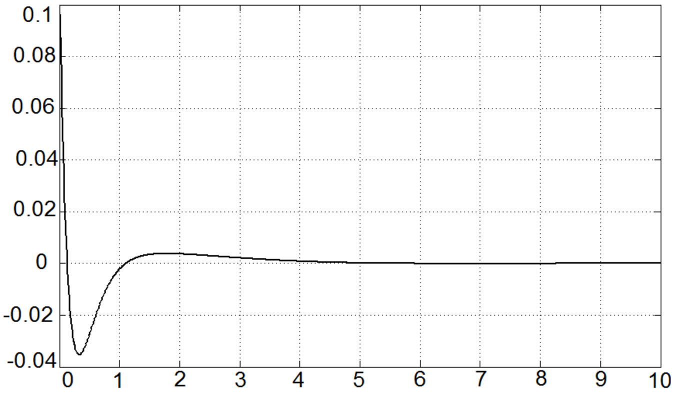

Fig – 5: Angle ϕ response

5. EXPERIMENT AND RESULTS

After the simulation, the control law is applied to a real IRP (Fig. 7) whose parameters are mentioned on previous section. The machine is controlled by four microcontrollers (*uC*) AT90CAN128. The arm and pendulum angles are measured by two optical encoders which have 2000 values/revolution. The communication among these microcontrollers is done by TTCAN protocol.

The positions of the two angles during the IRP operation are transmitted to a computer via *uC*'s *USART* port and a *Hterm* device. These data are plotted graphically by software *Gnuplot*.

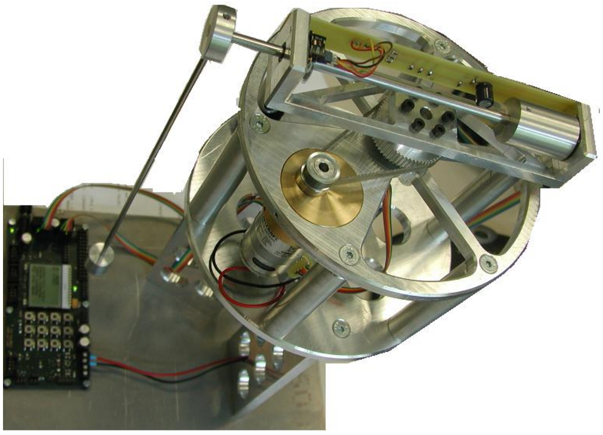

Fig – 6: A real model of IRP

At non-operation state, the pendulum is at right-down position and its angle is set to "0". At stable working condition, the pendulum should stay at upright range ($171^o \div 189^o$) or ($-189^o \div 171^o$) and the values of the pendulum's encoder should be in (910÷1090) or (-1090÷ -910). To move the pendulum into the upright range, the DC motor supplies kinetic energy to the pendulum and make it oscillating with increasing magnitude. Since the pendulum reaches the Upright range, the IRP controller switches to the *Upright* phase, and use the State Space control law (which is established in the previous section) to regulate the pendulum stay in that range.

The Fig-7 and 8 present graphically positions of the pendulum and the arm from the experiment. The data were collected from the encoders and processed by a computer. The time is counted by *uC*'s counters and presented by the horizontal axes. Each counter value is equal to 2 msec.

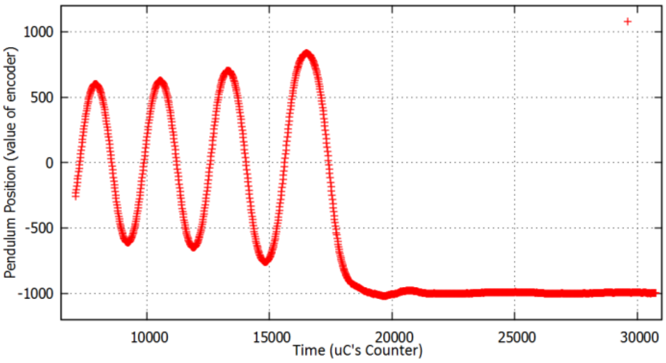

Fig – 7: The Pendulum is in Upright range after oscillating with increasing amplitude

The Fig - 7 shows the pendulum position from oscillating with increasing amplitude to reach the Upright range then stays in that range. The vertical axis presents the position of the pendulum (from the pendulum's encoder with the value range from -1999 to +1999). The time to get the data starts at the value 6000. Starting from the zero position, the DC motor had driven the pendulum to swing up. After several oscillations, the pendulum had enough energy to go to the Upright range. In that range, the IRP switches to Upright phase and *uCs* use the calculated control law to regulate the pendulum stays at that range steady. The pendulum position still stays stable in this range even though the arm position varies.

The Fig - 8 presents the arm position with nonzero reference input. When the pendulum was at the upright position, the new position of the arm was set to -21^0, which was equal the encoder's value -117. As the new reference input was applied, the arm moved slowly to the new position. The movement was not smooth but oscillating then reached the desired position at the counter value 5000. After that, the arm still oscillated but around the new position steady with decreasing amplitude. From the time of 25000, the amplitude of arm oscillation is stable about 2.7^o (the encoder values are from 111 to 126).

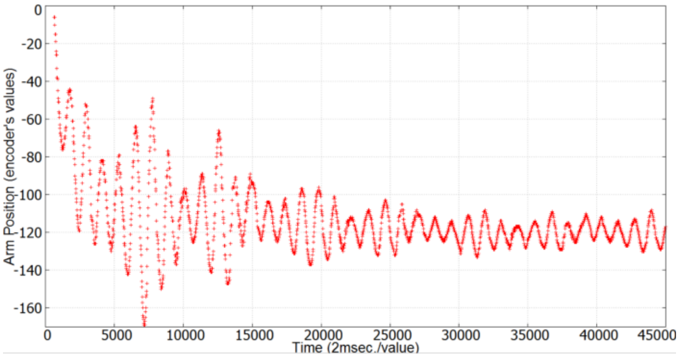

Fig - 8: The Arm position in Upright phase with nonzero reference input

There are two reasons for that phenomenon. The first one is the gearbox's backlash (around 0.5^0), which makes the arm loss amount of motion as the DC motor reverse the movement's direction. That caused the arm slipped out the predefined amplitude and oscillated longer to reach the upright point. The second reason is the parallel misalignment between the shaft of the gearbox and the arm rotation axis, which makes the inconstant transmitted torque between the gearbox and arm rotated axis. It also creates the vibration of the arm axis.

The Fig - 9 showed the IRP in the Upright State when the arm position was set to -21^0

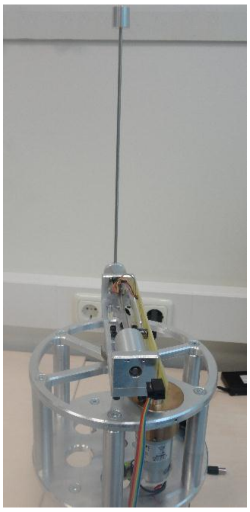

Fig – 9: The IRP at upright position

6. CONCLUSIONS

In this paper, State Space Control method is deployed to control the IRP work when the arm can stay at non-zero position. By adding one more state to the system state vector, the state space controller can handle with the error causing by non-zero position of the IRP arm. Despite of several mechanical problems such as the backlash of the gearbox, the parallel misalignment between the arm rotated axis and the gearbox's output shaft, the controlled algorithm still can regulate the pendulum's operation to meet the requirements. The result proves the robustness of the control method in the SIMO system.

REFERENCES

[1] Jadlovska S., Jan Sarnovsky J., *A Complex Overview of Modeling and Control of the Rotary Single Inverted Pendulum System* - POWER ENGINEERING AND ELECTRICAL ENGINEERING, VOLUME 11-2, 2013, pp. 73-85

[2] Sirisha V., Junghare A. S. – *A comparative study of controllers for stabilizing a Rotary Inverted Pendulum* - International Journal of Chaos, Control, Modeling and Simulation (IJCCMS), Vol.3, No.1/2, DOI : 10.5121/ijccms.2014.3201 , June 2014

[3] Jose A., Augustine C., Malola S. M., Chacko K., - *Performance Study of PID Controller and LQR Technique for Inverted Pendulum* - World Journal of Engineering and Technology, 2015, 3, pp. 76-81

[4] Babu J., Varghese E., *Stabilization of Rotary Arm Inverted Pendulum using State Feedback Techniques* - International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181, Vol. 4 Issue 07, 2015, pp.563-567

[5] Hassan M., Kadri M., Amin I., *Proficiency of Fuzzy Logic Controller for Stabilization of Rotary Inverted Pendulum based on LQR Mapping* – Fuzzy Logic and Applications: 10th International Workshop, WILF 2013, Italy, pp. 201-211

[6] Bui, D. H. - *Development and Realization of a distributed control algorithm for an inverted rotary pendulum using FreeRTOS with TTCAN communication*, Embedded system Institute, University of Siegen, 2014

[7] Franklin G. F. J., Powel D., Emami-Naeini A.– *Feedback control of Dynamic System* 5th Pearson Education, Inc, 2006, New Jersey, - ISBN 0-13-149930-0, pp 438-563

[8] Polderman J. W., Willens J. C., *Introduction to the Mathematical Systems Theory: A Behavioral Approach*, Spinger Verlag 1998, ISBN 978-1-4757-2953-5, New York, pp, 151-190