Controllers design for a double pendulum crane

Daniel Castro*, Gerald Elizondo* y Sebastián Gamboa*

*Escuela de Ingeniería Electrónica, Instituto Tecnológico de Costa Rica (ITCR), 30101 Cartago, Costa Rica,
{daescastros, gerie.ge24, segamboachacon}@gmail.com

***Abstract*—In this paper there are presented two different controllers: LQR and REI. Designed for the plant called Double Pendulum Crane. Controllers are carried out with Matlab tools such as Simulink and Sisotool and finally the results obtained in the real plant are shown. The controller that shows the best behavior for the plant is REI, followed by LQR that presents acceptable results.**

***Keywords*—Controller, Crane, LQR, REI, Pendulum.**

I. INTRODUCTION

In the academy, a large number of controllers are taught, which can be very useful for some applications and not so much for others. As is known, the automatic control area is fundamental in electronic engineering and its application in the industry is of highly importance, therefore, it is required that in addition to knowing how to perform, have criteria when designing them with the objective of carrying out a controller that best meets the requirements.
For this laboratory a port crane was made, which must be controlled by 2 different controllers at the choice of the designers but that comply the design requirements, which in this case are 0 % steady-state error , settling time less than $6s$, a percentage overshoot less than 3 %, and an angle less than 10 degrees or 0.17 radians.

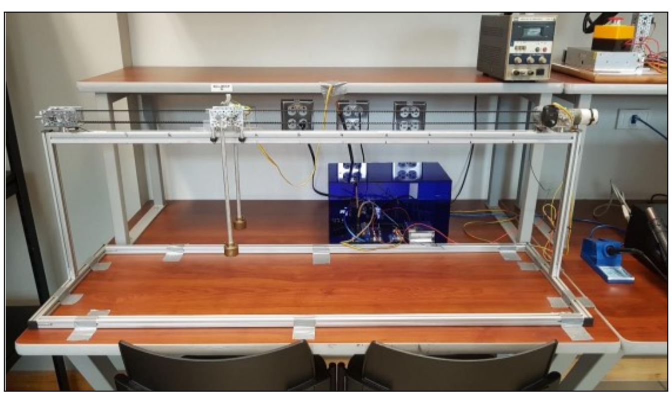

Figure 1. Double Pendulum Crane plant photo

II. IDENTIFICATION OF THE PLANT MODEL

The plant without controller, the angle shows a very underdamped behavior, as can be seen in figure 2 and the position shows a step behavior as can be seen in figure 3. Both curves are made with experimental data of short square signal response. To obtain a transfer function for modeling the plant, it is used the *systemIdentification* tool (or *ident*) from Matlab.

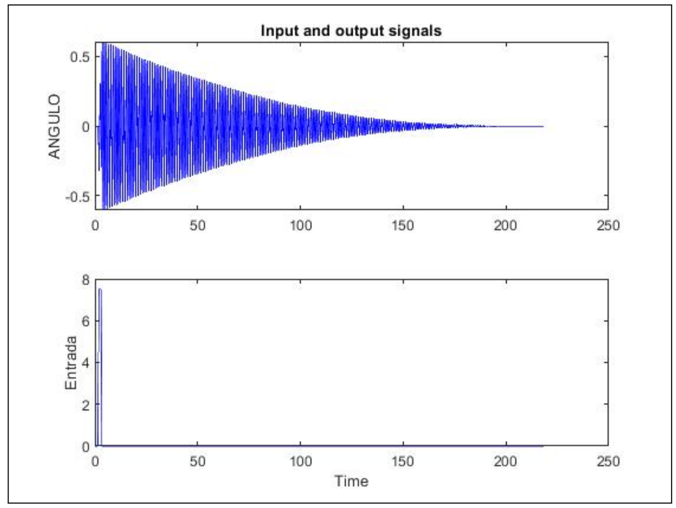

Figure 2. Angle data of the plant behavior without controller.

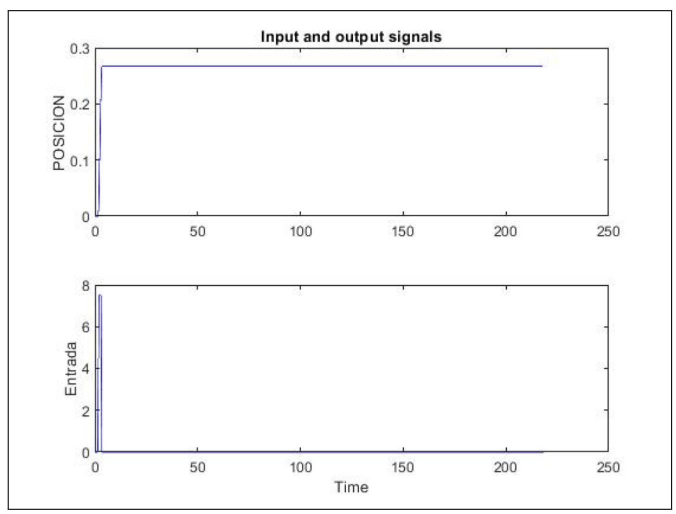

Figure 3. Position data of the plant behavior without controller.

Using the raw data with this tool the approximation percentage of the transfer function is around 98%. With all the data was obtained a more suitable model that a model obtained from a pre-procesing data. The angle transfer-function output and the data comparision resulting is shown in figure **??**. It was obtained a 84.44% similarity.

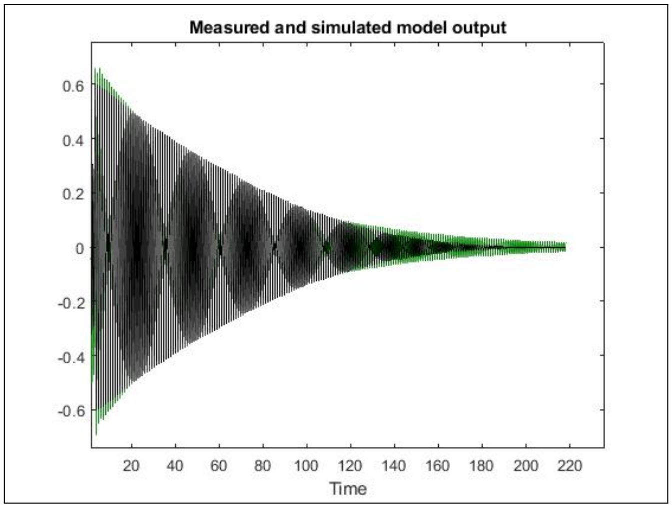

Figure 4. Comparison of angle transfer function model and experimental data

The position transfer function obtained around 99.5% similarity with the data.

The SISO models that describe the plant are:

$$Angulo(s) = \frac{-0.3808(s+2.694)}{(s^2+0.0342s+35.32)} \quad (1)$$

$$Posicion(s) = \frac{0.4342}{s(s+18.05)} \quad (2)$$

It is needed to describe the plant as a SIMO model using space-state equations. In order to re-arrange as SIMO, it is needed to combine the two SISO models obtained previously, by converting the position model to a FCC and expanding the arrays. Finally, the arrays that describe the plant as SIMO are:

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -18.05 & 0 \\ 0 & -35.32 & 0 & -0.0342 \end{bmatrix} \quad (3)$$

$$B = \begin{bmatrix} 0 \\ -0.3808 \\ 0.4342 \\ -1.013 \end{bmatrix} \quad (4)$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \qquad D = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (5)$$

To verify the models in simulink, it is used the C matrix like C = [1 0 0 0]

III. Design of Controllers

A. Integral State Feedback by Poles Location:

For this controller, is required to use the matrixes that were previously obtained for the base model of the plant. Just is necessary to expand the matrix A and B following the next structure:

$$A_t = \begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix} \quad (6)$$

Where A and C are the state space matrices. It results in the following matrix:

$$A_T = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & -18.05 & 0 & 0 \\ 0 & -35.32 & 0 & -0.0342 & 0 \\ -1 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (7)$$

Similarly, there is a new matrix B_t compound by the state space B matrix, giving the shown result:

$$B_t = \begin{bmatrix} B \\ 0 \end{bmatrix} \quad (8)$$

$$B_T = \begin{bmatrix} 0 \\ -0.3808 \\ 0.4342 \\ -1.0130 \\ 0 \end{bmatrix} \quad (9)$$

The following poles have been chosen for the controller according with the design requirements that were, less than 3% of percentage overshoot and less than 6 seconds of settling time, and also according with the suggestion of having 2 dominants conjugated complex poles and 3 non dominants real poles:

$$P = \begin{bmatrix} -0.7300+0.4088i & -0.7300-0.4088i \\ -3.6296 & -7.0297 \\ -11.4297 & \end{bmatrix} \quad (10)$$

These poles locations were obtained using the Mat-Lab's tool, Sisotool, were was possible to adjust condition and place the poles at the allow places. In figure 5 the Sisotool rlocus is shown:

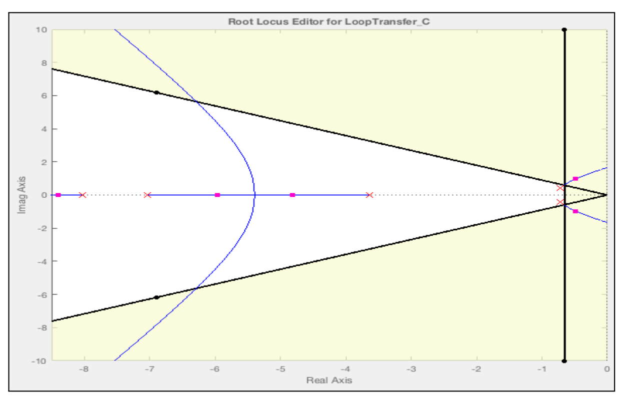

Figure 5. Rlocus - Sisotool for Poles Location

Using $place(A_t, B_t, P)$ (ackerman) command, the K constants can be obtained as:

$$K = \begin{bmatrix} 34.4760 & -15.5235 & -4.3229 & -1.4120 \end{bmatrix}$$

$$Ki = \begin{bmatrix} 13.3110 \end{bmatrix}$$

Finally, following the structure for the Integral State Feedback, a controlled plant simulation was design using the Matlab's tool Simulink, as shown in figure 6:

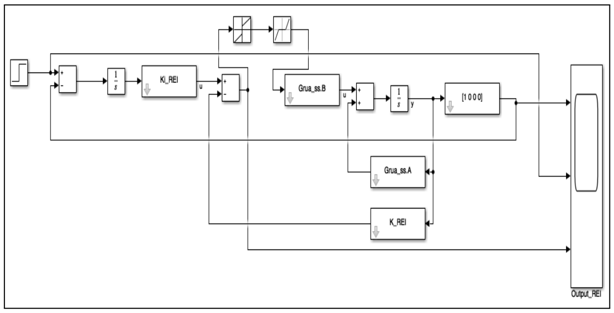

Figure 6. Block diagram for Crane controlled by Poles Location

And for this simulation, the figure 7 was obtained:

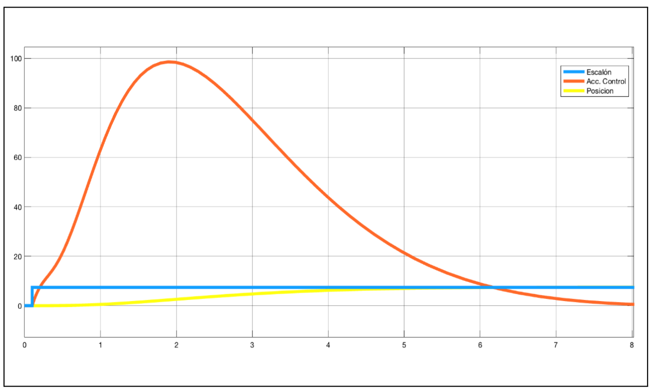

Figure 7. Simulation of crane position with poles location controller

B. LQR:

For LQR controller, the following extended matrixes, made from the system space-state, are made for simulating the plant position. For simulating, only the position system is used in order to simplify the design of the controller. The angle model should be used if the experimental data does not comply with the requirements.

$$A_{LQR} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & -18.05 & 0 & 0 \\ 0 & -35.32 & 0 & -0.0342 & 0 \\ -1 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (11)$$

$$B_{LQR} = \begin{bmatrix} 0 \\ -0.3808 \\ 0.4342 \\ -1.0130 \\ 0 \end{bmatrix} \quad (12)$$

$$C_{LQR} = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \qquad D_{LQR} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (13)$$

It is needed to choose the matrix Q that better fits the system, where it is given weight to the states that are most important. In this case the position and the integral values are increased to 400. Aditionally, matrix R is made with a weight of 1, since the output seems to comply without the need to change this value.

$$Q = \begin{bmatrix} 400 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 400 \end{bmatrix} \quad (14)$$

$$R = \begin{bmatrix} 1 \end{bmatrix} \quad (15)$$

Using the matlab function lqr, we generate the next K and Ki constants. It is also important to note that the K values should not be bigger than around 55, and Ki should not exceed 35. This is because the real plant will not work properly with too high values.

Realizando la descomposición de las constantes en forma de un PID, tenemos:

- $$K = \begin{bmatrix} 49.0240 & -5.9160 & 2.6524 & -0.0811 \end{bmatrix} \quad (16)$$
- $$Ki = 20 \quad (17)$$

Using these values and simulink, we study the feasibility of the crane's LQR controller, as shown in figure 8:

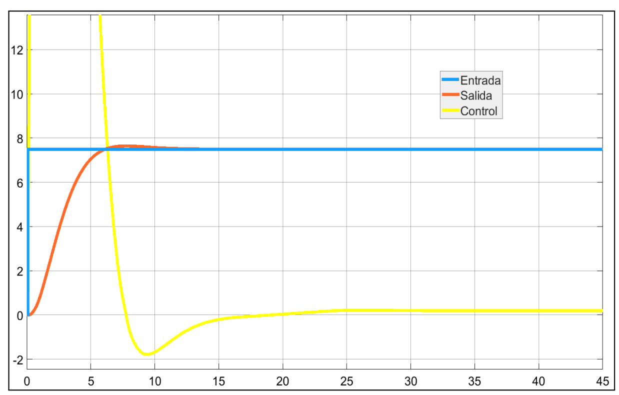

Figure 8. Simulation of crane position with LQR controller

IV. Results Analysis

A. LQR:

For LQR, the experimental results can be seen in figures 9 where it shows the position of the crane, and 10, that shows the angle. On both, it can be seen the output and the controller correction.

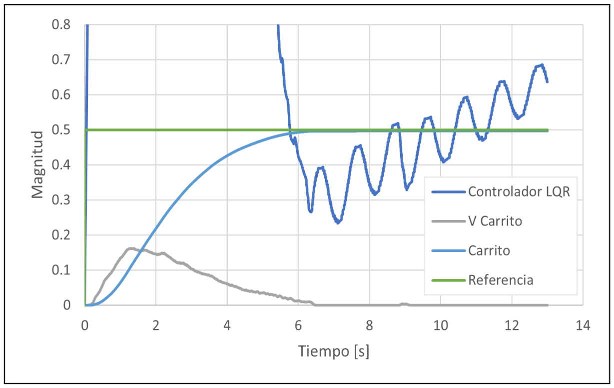

Figure 9. Graph of experimental data of crane position with LQR controller

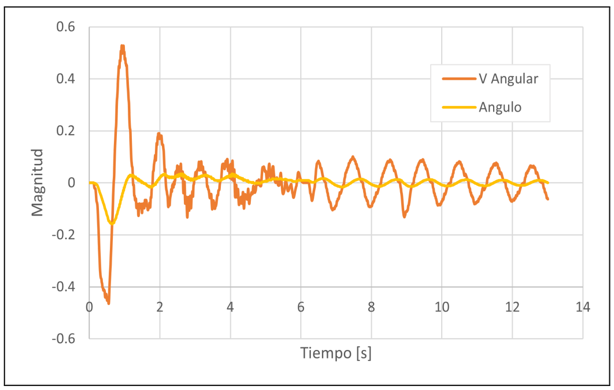

Figure 10. Graph of experimental data of crane angle with LQR controller

The actual data confirms that the controller behaves between limits, but almost reaching them:

- Stationary State Error: 0%
- Overshoot percent: approximately 0%
- Stabilization time: $5.74s$
- Max. angle: $0.159rad = 9.11°$

B. Poles Location:

For the Poles Location, the experimental results can be seen in figure 11.
Analyzing the figure, can be seen that orange curve that represents position of the crane, actuated as expected, in a soft way, and also the blue curve shows the Angle, where is possible to appreciate that it almost doesn't has oscillations. A summary of the main data is shown below:

- Stationary State Error: 0%
- Overshoot percent: approximately 0%
- Stabilization time: $5.565s$
- Max. angle: $0.094rad = 5.39°$

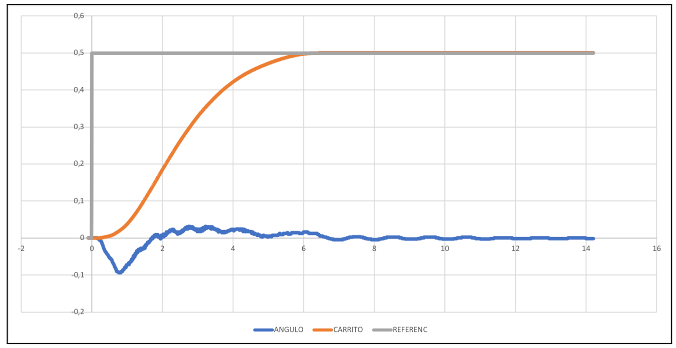

Figure 11. Experimental data for Crane with Poles Location controller.

So, with the results showed is possible to affirm that the Poles Location technique is a very good way for controlling a crane because can be adjust with the design requirements.

V. Recommendations:

- Avoid using too big values for controller constants, since this could cause a misbehavior of the plant.
- When designing a controller for a specific application, make research to find the best controller that suits similar appliances. This with the objective of avoiding designing a controller that could not fulfill the task because of it's nature.
- It would be recommended to perform a detailed calibration to verify if Poles Location is always better than LQR for this plant.

VI. Conclusions

There are many controllers that can be used to control a plant, but not everyone can fit to the expected behavior depending on the specific applications. In these case scenario, while both Poles Location and LQR controllers comply with the stated requirements, Poles Location is the one with the best behavior found. It is important to note that the way of obtaining the values for each controller can wrongly show a bad behavior of a controller that should work correctly for a application.

Also, for the LQR controller, the matrix Q has more weight in the position and steady-state error values.

Access to Github Matlab code: GitHub

References

[1] E. Interiano. Control de velocidad angular del motor CD hps5130. ITCR, Cartago, Costa Rica, 2021
[2] A. Ruiz. Instructivo Grúa. ITCR, Cartago, Costa Rica, 2021
[3] E. Interiano. Presentación - Control por LQR y LQG. ITCR, Cartago, Costa Rica, 2021
[4] E. Interiano. Presentación - Realimentación de estado integral. ITCR, Cartago, Costa Rica, 2021