

SIMMECHANICS VISUALIZATION OF EXPERIMENTAL MODEL OVERHEAD CRANE, ITS LINEARIZATION AND REFERENCE TRACKING-LQR CONTROL

Thein Moe Win¹ Tim Hesketh² Ray Eaton³

School of Electrical Engineering & Telecommunication
The University of New South Wales, High St, Kensington, NSW 2052, Australia

ABSTRACT

Overhead Crane experimental model using Simmechanic Visualization is presented for the robust antisway LQR control. First, 1D translational motion of overhead crane is designed with exact lab model measurements and features. Second, linear least square system identification with 7 past inputs/outputs is applied on collected simulation data to produce more predicted models. Third, minimize root mean square error and identified the best fit model with lowest RMSE. Finally, Linear Quadratic Regulator (LQR) and Reference tracking with pre-compensator have been implemented to minimize load swing and perform fast track on trolley positioning.

KEYWORDS

Simmechanic Visualization, linear least square system identification, Linear Quadratic Regulator (LQR), Reference tracking with pre-compensator, Overhead Crane.

1. INTRODUCTION

In modern industrial system, shipyard, warehouse, and construction, tower cranes are widely used for the heavy loads transfer. The crane operates hoist up-down motion, trolley forward-backward motion, and flat-top rotation causing unexpected load swing. In those working aspects, transferring load at a shorter time, perfect safety with no load swing or damage, and low operational cost are the main concerns. The cranes use open loop system to control position while anti-swing control is done manually by skilled operators using joysticks with analog control, [6]. Since, skilled operator always needs to focus on the operational trajectory map back and forth to carry the load, it creates fatigue problem and eventually affect the whole operation. Higher running costs, operational delay, and environmental damages have to be taken into account. Therefore, minimizing the load swing with proper control action during operation would be essential to avoid several risks. Even though researchers worldwide have been proposing many forms of anisway approaches, yet using manual joy stick with human operators are inevitable.

2. PROBLEM STATEMENT

Conducting live research on the real operating crane on site is usually impossible and therefore, most of the researches derived mathematically assumed crane model for simulations purpose. Furthermore, to simplify the crane modeling, assumptions were sometimes taken on the factors such as; (frictions, noises and disturbances, actuators, and sensors, etc.). Linearization also plays vital role in getting better accurate model however considering certain range of small load swing angle to be zero would have impact on linear approximation of the model and subsequently the controller design.

Therefore in this research, having a reliable model to represent real crane is the initial focus followed by linearization and suitable controller development. Experimental model visualization of the crane system has been developed based on real crane parts/parameters, which include trolley cart, rail jib, steel cable, load, sensors and actuators, actual mass-moment of inertia-densities, and frictions. Visual appearance of this crane design would not only feature the real crane but also be flexible enough to assemble parts and assign real time parameters. It is the landmark for researchers to do real time research without having trouble in dealing with real cranes on site. The raw data collected from model simulation are used by linear least square approach to maximize better approximation model for further LQR with reference tracking. In order to compare the performances, normal mathematical model derivation is first developed in the following session.

3. FREE BODY DIAGRAM OF 2D X-MOTION

Simple free body diagram, Fig (1), below is developed to represent 2D crane model. A trolley cart is mounted on a jib (rail) along x direction, [9]. Trolley cart mass (M), payload mass (m), and load length (l) are assigned certain values while (θ) is considered load swing angle in XY plane. For 2 dimensional (2D) simple motion, payload length is initially considered unchanged and frictionless between trolley and jib. Since certain amount of applied force F pushes trolley to move along X direction, large swing angle θ appears which needs to be minimized to as small as possible. In order to get mathematical model of the crane, kinetic energy, potential energy, Lagrange equation are derived to form equations of motions (nonlinear).

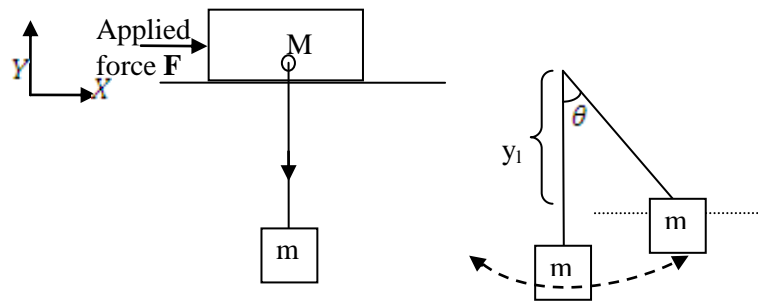


Figure (1): Simple 1D free body diagram

3.1 Mathematical Modeling of Trolley cart and load swing

From free body diagram of the system, nonlinear equations, Equ.(1) and Equ.(2), have been derived as follow;

$$\ddot{x}_t = \frac{1}{(M+m)} [F - ml(\ddot{\theta} \cos \theta - \dot{\theta}^2 \sin \theta) - b\dot{x}] \quad (1)$$

$$\ddot{\theta} = \frac{1}{(l-mI^2)} [ml(\ddot{x}_t \cos \theta + g \sin \theta) - c\dot{\theta}] \quad (2)$$

3.2 Linearization using simple assumption

Several assumptions such as; ($\sin \theta = 0, \cos \theta = 1$ (OR) $\sin \theta = \tan \theta = \theta, \cos \theta = 1$ ect.) have been made from nonlinear model by the researchers in the past to linearize nonlinear equations of crane model. Even though that assumption makes a model simpler, the linearized model could not be close enough to actual crane model, [5]. In this research, at equilibrium p oint of load swing, both θ and $\dot{\theta}$ are considered as small and assumed as $\sin \theta \approx \theta, \cos \theta \approx 1, \text{ and } \dot{\theta} \approx 0$, [12]. Using these approximations, the mathematical model

linearized around the unstable equilibrium point of load swing has been obtained, Equ.(3) and Equ.(4). Final linearized equations and state-space form of the overhead crane free body diagram are;

$$\dot{x}_2 = -\frac{(I - ml^2)b}{J}x_2 - \frac{m^2l^2g}{J}x_3 + \frac{mlc}{J}x_4 + \frac{(I - ml^2)F}{J} \quad (3)$$

$$\dot{x}_4 = -\frac{mlb}{K}x_2 + \frac{(M+m)mlg}{K}x_3 - \frac{(M+m)c}{K}x_4 + \frac{ml}{K}F \quad (4)$$

Linearized State Space model is; $\dot{x} = Ax + Bu, y = Cx + Du$

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{(I - ml^2)b}{J} & -\frac{m^2l^2g}{J} & \frac{mlc}{J} \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{mlb}{K} & \frac{(M+m)mlg}{K} & -\frac{(M+m)c}{K} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{(I - ml^2)F}{J} \\ 0 \\ \frac{mlF}{K} \end{bmatrix} u, y$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_r \\ \dot{x}_r \\ \theta \\ \dot{\theta} \end{bmatrix}$$

4. SIMMECHANIC VISUALIZATION OF OVERHEAD CRANE

This simmechanic-based experimental model is based on Physical Modeling blocks which represents physical components, geometric, and kinematic relationships directly. This is not only more intuitive, it also saves the time and effort to derive the equations of motion. In this research, simple 2D Overhead crane mechanical model is designed using MATLAB Simmechanic visualization. First of all, 5 meter long jib bar frame was mounted above the ground using two fixed revolute joints. Trolley cart with certain measurement has been mounted on top of jib frame,[4],[8]. At this stage, low friction prismatic joint is considered in order to minimize constraints. A steel cable type body with payload was attached to the trolley using revolute joint,[11],[13]. Certain measurements were assigned to appear 2D crane model, Fig (2). Signal builder produces applied force (step input signal) which actuates the trolley cart, Fig (2), for X-directional motion.

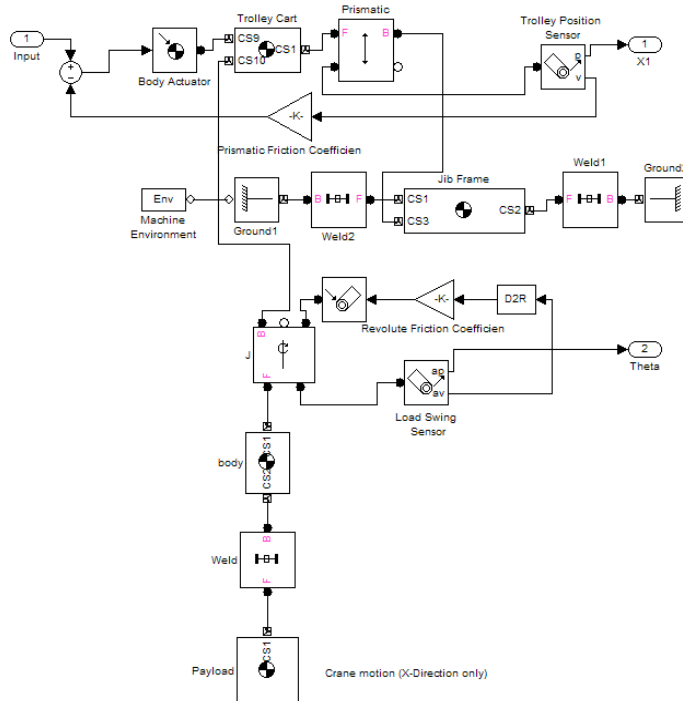


Figure (2,a): Mechanical Visualization of Overhead Crane

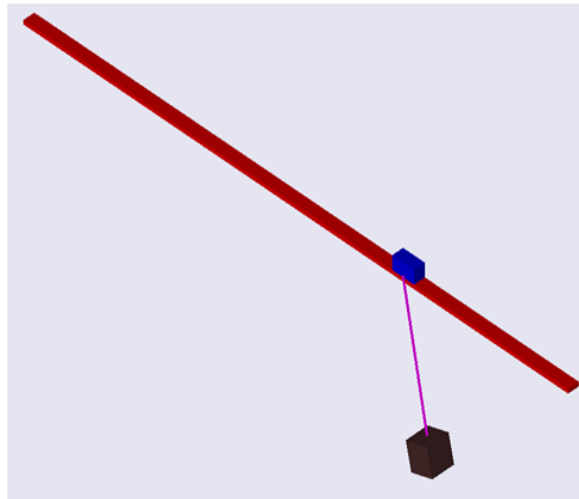


Figure (2,b): Mechanical Visualization of Overhead Crane

5. LINERIZATION OF 2D OVERHEAD CRANE USING LEAST SQUARE APPROACH

To linerize the model, adaptive perturbation approach with “linmod” command is used in Matlab simulink. That command will search equilibrium point (operating point) of the system where the net force is zero. Then, it generates linear time-invariant (LTI) state-space models. However, a simple step of creating model using “linmod” command without the details would make difficulty in analyzing the model. Therefore, instead of direct linearizing by MatLab, the mathematical model would be developed from collected dataset. The following steps have taken; model the input signal, collect the data set, and determine denominator/numerators using Least Square System Identification.

5.1 Modelling Input Signals Design

For x , y , and z motions, the system uses 3 different motors and therefore, different input designs have to be performed. Initially, other higher frequencies were tested but since, the frequency content of the input too high, the system does not have enough time to respond to changes in the input. In results, the response produced averaging effect. In current x -directional motion simulation, pseudo-random binary signal (PRBS) with 10 Hz sampling frequency and [1,-1] upper-lower bounds was designed, Fig (3). By varying the input signal and adjust longer pulse, it will enhance the statistics of the data, optimize the effectiveness of system identification as well as have visual respond to each input state.

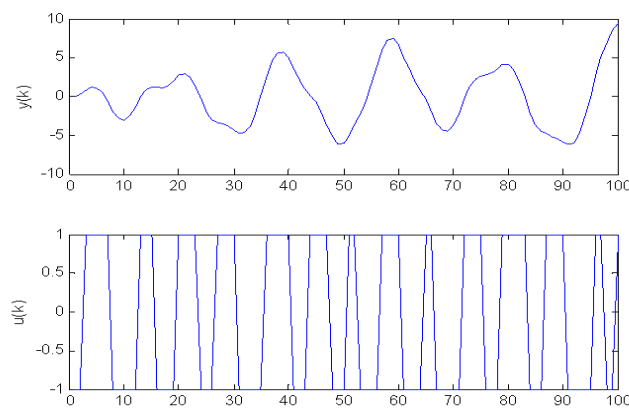


Figure (3): Input Signal at Frequency 10, range -1 to 1

5.2 Linear Least Square System Identification

Once the input/outputs dataset is collected, Least Square System Identification is applied to determine the number of parameters required for the system transfer function and find those parameters (b_1, b_2, \dots, b_m and a_1, a_2, \dots, a_n), Equ.(5), [1].

$$\Delta y(k) = \frac{b_1 z^{-1} + b_2 z^{-2} + \dots + b_m z^{-m}}{1 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_n z^{-n}} \Delta u(k) \tag{5}$$

The output of the system represents the linear combination of past inputs and outputs, where, $\theta_1 \dots \theta_m, \theta_{m+1} \dots \theta_{m+n}$ are the states of the system, Equ.(6), [1].

$$y(k) = -\theta_1 y(k-1) - \dots - \theta_m y(k-m) + \theta_{m+1} u(k-1) + \dots + \theta_{m+n} u(k-n) \tag{6}$$

$$y \begin{bmatrix} y(k) \\ y(k+1) \\ \vdots \\ y(k+N) \end{bmatrix} = \begin{bmatrix} -y(k-1) & \dots & -y(k-m) & u(k-1) & \dots & u(k-n) \\ -y(k) & \dots & -y(k-m+1) & u(k) & \dots & u(k-n+1) \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ -y(k-1+N) & \dots & -y(k-m+N) & u(k-1+N) & \dots & u(k-1+n) \end{bmatrix} \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_m \\ \theta_{m+1} \\ \vdots \\ \theta_{m+n} \end{bmatrix}$$

$$y(k) = [-y(k-1) \quad \dots \quad -y(k-m) \quad u(k-1) \quad \dots \quad u(k-n)]$$

$$y(k) = X\theta \quad (X \text{ is } m \times n \text{ with } m > n) \tag{7}$$

5.3 Denominator and Numerator Coefficients consideration

To compute 2 Denominator and 2 Numerator Coefficients, least square approach uses two simultaneous past inputs/outputs data to form **X** Matrix and therefore the following matrix would be formed, [10].

$$\begin{bmatrix} y(3) \\ y(4) \\ \vdots \\ y(100) \end{bmatrix} = \begin{bmatrix} -y(2) & -y(1) & u(2) & u(1) \\ -y(3) & -y(2) & u(3) & u(2) \\ \vdots & \vdots & \vdots & \vdots \\ -y(99) & -y(98) & u(99) & u(98) \end{bmatrix} \theta$$

5.4 Training and Checking Model

Initially, the collected dataset is divided into two parts, training and checking, Fig(4). The reason is, the developed model from the training part would be tested against checking output in order to verify how well the model matches. In this system, first half training dataset, y_{tr}, u_{tr} and X_{tr} have been developed to calculate Denominator and Numerator Coefficients, θ and produce Model Transfer Function.

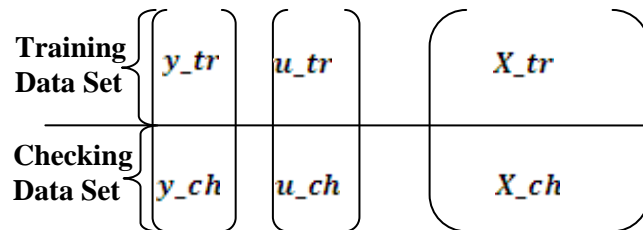


Figure (4): Training and Checking Data separation

Considering 2 Denominator and 2 Numerator Coefficients case, X_{tr} Matrix is identified and θ is computed using Equ.(7). The coefficients are estimated $\theta = x_{est} = [a_1, a_2, b_1, b_2]$ to form the following Model Transfer Function, [2].

$$X_{tr}^T * y_{tr} = (X_{tr}^T * X_{tr}) * x_{est}$$

$$x_{est} = [X_{tr}^T * X_{tr}]^{-1} * X_{tr}^T * y_{tr}$$

$$y_{Model_tr} = \frac{b_1 q^{-1} + b_2 q^{-2}}{1 + a_1 q^{-1} + a_2 q^{-2}}$$

5.5 Root Mean Square Error Calculation

Root-mean-square error (RMSE) is used to measure the differences between estimated values by the developed model, and actual outputs such as; y_{Model_tr} VS y_{tr} and y_{Model_ch} VS y_{ch} . Minimizing RMSE could produce better fit of the model and enhance approximation. RMSE is calculated in the following form, Fig(5);

Least Square General Form, [3]:

$$\|Ax - y\| = \left(\sum_{i=1}^m \left(\sum_{j=1}^n a_{ij} x_j - y_i \right)^2 \right)^{1/2} \tag{8}$$

$$\|X * x_{est} - y_{tr}\| = \|y_{Model_tr} - y_{tr}\| = norm(y_{tr} - y_{Model_tr})$$

$$RMSE = \frac{norm(y_{tr} - y_{Model_tr})}{\sqrt{length\ of\ (y_{tr})}} \tag{9}$$

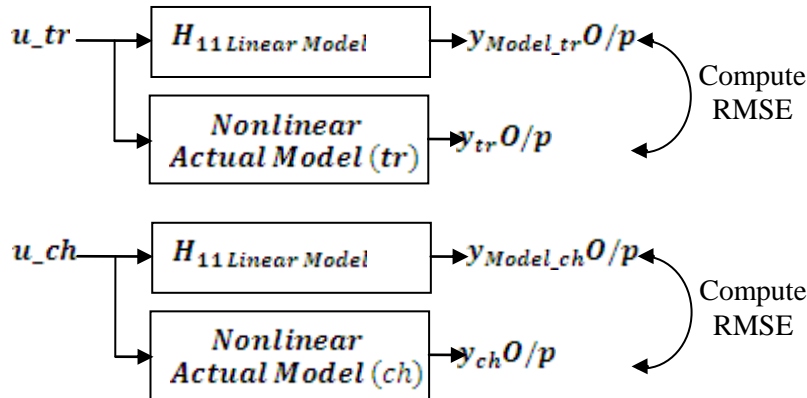


Figure (5): RMSE Comparison between actual and predicted models

Using Least Square System Identification; Den2Num2, Den3Num3, Den4Num4, Den2Num3, Den3Num2, Den4Num3, and Den3Num4 models are computed then followed by their respective RMSE. After comparing all RMSEs, lowest RMSE and its respective model would be picked up as the best linearly approximated model of the system. The following table shows, each model adjustment and calculated RMSE for both Training and Checking parts. Least Square System Identification with past consecutive inputs-outputs, Table (1);

Table(1): Least Square System Identification

Den	Num	Data Start from	Past data start from
2	2	y_3	[-y2 -y1 u2 u1]
3	3	y_4	[-y3 -y2 -y1 u3 u2 u1]
4	4	y_5	[-y4 -y3 -y2 -y1 u4 u3 u2 u1]
2	3	y_4	[-y3 -y2 0 u3 u2 u1]
3	2	y_4	[-y3 -y2 -y1 u3 u2 0]
4	3	y_5	[-y4 -y3 -y2 -y1 u4 u3 u2 0]
3	4	y_5	[-y4 -y3 -y2 0 u4 u3 u2 u1]

Since 1D overhead crane has Trolley cart’s translational motion and load’s swing angle outputs, the Least Square calculation produce two models for Trolley and load swing while each model has training RMSE and checking RMSE. After generating all RMSEs, lowest RMSE would be picked up from Trolley model and load swing model considering the best linear approximation. The following Table (2.a) and Table (3.a) show each computed RMSE(Training), RMSE(Checking), as well as its total RMSE for models (22,33,44,23,32,43,34) while Table (2.b) and Table (3.b) show lowest to highest computed RMSE in which Den 2 Num 2 model appears to have lowest RMSE not only in Trolley model but also in Load Swing model.

TrolleyCartRMSEDenNum

RMSEs(Tr)	RMSEs(Ch)	RMSE	Den	Num	RMSEs(Trolley)	Den	Num
0.12645	0.13602	0.13123	2	2	0.095499	3	3
0.09208	0.098917	0.095499	3	3	0.095686	3	4
0.12587	0.13558	0.13073	2	3	0.096211	4	3
0.12461	0.13458	0.12959	3	2	0.12959	3	2
0.093626	0.098796	0.096211	4	3	0.13073	2	3
0.092217	0.099155	0.095686	3	4	0.13123	2	2
1.574e+008	1.6271e+008	NaN	4	4	NaN	4	4

Table(2.a): RMSE(Training) and RMSE(Checking) for Trolley

Table(2.b): from lowest to highest RMSE for Trolley Model

LoadSwingRMSEDenNum

RMSEs(Tr)	RMSEs(Ch)	RMSE	Den	Num	RMSEs(LS)	Den	Num
0.82772	2.4455	1.6366	2	2	1.6366	3	2
0.8423	2.4453	1.6438	3	3	1.6366	2	3
0.82772	2.4455	1.6366	2	3	1.6366	2	2
0.82771	2.4455	1.6366	3	2	1.6438	3	3
5.4277	4.2815	4.8546	4	3	4.8546	4	3
9.0589	16.653	12.856	3	4	12.856	3	4
1.574e+008	1.6271e+008	NaN	4	4	NaN	4	4

Table(3.a): RMSE(Training) and RMSE (Checking) for Load Swing

Table(3.b): from lowest to highest RMSE for Load Swing

Trolley model, Den3 Num3, with lowest RMSE shows how well the linear approximation matches the actual output graph, Fig (6.a) in which, residuals are less than 0.08 and RMSE is only

0.095499. For the load swing model, Den3 Num2, with lowest RMSE produces residuals which are between (-0.4 to 0.4) while its RMSE is only 1.6366, Fig (6.b).

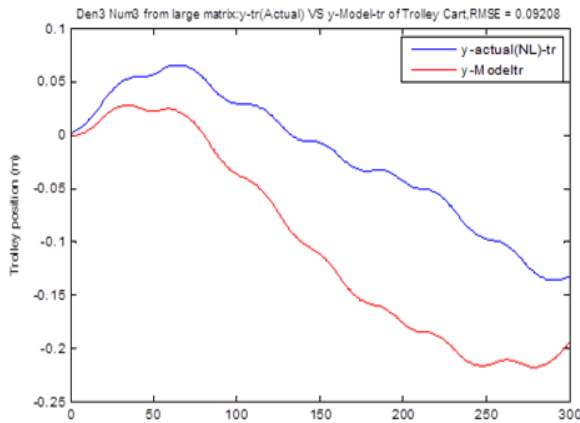


Figure (6.a): Den 2 Num 2 Trolley model

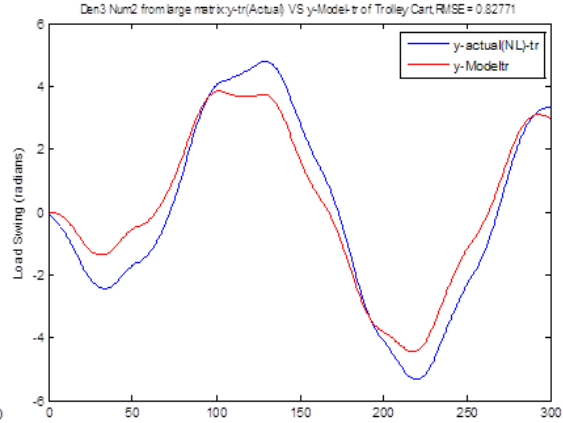


Figure (6.b): Den 2 Num 2 Load Swing model

6. LINEARIZATION OF THE CRANE USING MODIFIED LEAST SQUARE WITH 7 PAST INPUTS/OUTPUTS

To represent the X-translational motion of the load-attached trolley cart, the collected dataset is used to develop linear least square form of linearization. Usually, Linear Least Square approach uses linear combination of the past inputs/outputs dataset, for instant, $[y_3]=[-y_2 \ -y_1 \ u_2 \ u_1]$ and it can achieve only one Root Mean Square Error (RMSE) by comparing actual and estimated outputs. And that would be hard to analyze whether the model provides better approximation from the processed past inputs/outputs 4 dataset. Therefore, up to 7 past inputs/outputs dataset in the matrix X (mentioned below) have been applied in the linear least square which would produce more models, develop many RMSEs and eventually get better approximation of the model. The following X matrix shows how the 7 past inputs/outputs dataset are organized.

$$\begin{bmatrix} y(8) \\ y(9) \\ \vdots \\ y(100) \end{bmatrix} = \begin{bmatrix} 1 & \dots & 7 & 8 & \dots & 14 \\ -y(7) & \dots & -y(1) & u(7) & \dots & u(1) \\ -y(8) & \dots & -y(2) & u(8) & \dots & u(2) \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ -y(99) & \dots & -y(93) & u(99) & \dots & u(93) \end{bmatrix} \theta$$

6.1 Den and Num Coefficients consideration with Least Square modified approach

Selecting every possible combinations of the columns from a large matrix with 7 past inputs and 7 past outputs according to the model needs and forming X matrix to compute estimated states, are some of the essential tasks to look for better approximated model. For instance, to get Den 2 Num 2 model, X matrix with 2 outputs columns $-y(?)$ and 2 inputs columns $u(?)$ are required.

$$\begin{bmatrix} y(8) \\ \vdots \\ y(N) \end{bmatrix} = \begin{bmatrix} -y(?) & -y(?) & u(?) & u(?) \\ \vdots & \vdots & \vdots & \vdots \\ -y(?+N) & -y(?) & u(?+N) & u(?+N) \end{bmatrix} \theta$$

$$y(k) = X\theta$$

From available 7 outputs columns, every possible 2 columns are picked at each time and same as every possible 2 from 7 inputs columns. That is, all possible 21 pairs of outputs/inputs columns can be picked to form 21 possible models. In previous session, $y_3=[-y_2 \ -y_1 \ u_2 \ u_1]$ with only one

model could be formed while in this case there would be 21 models with Den2 Num2 coefficients which may produce better approximation

$$\begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_{21} \end{bmatrix} = \begin{bmatrix} \text{Column} & \text{Column} & \text{Column} & \text{Column} \\ 1 & 2 & 8 & 9 \\ 1 & 3 & 8 & 10 \\ \vdots & \vdots & \vdots & \vdots \\ 6 & 7 & 13 & 14 \end{bmatrix}$$

For each X1, X2,...,X21, estimated states are calculated to form estimated Models which would be compared against actual output. From, X1 matrix;

$$X_1 = \begin{bmatrix} \text{Column} & \text{Column} & \text{Column} & \text{Column} \\ 1 & 2 & 8 & 9 \\ -y(7) & -y(6) & u(7) & u(6) \\ \vdots & \vdots & \vdots & \vdots \\ -y(7+N) & -y(6+N) & u(7+N) & u(6+N) \end{bmatrix},$$

$$\theta_1 = X_1^T * Y_1 ==> RMSE_1$$

Overall steps from selecting pairs, calculating estimated states and RMSEs for each case to computing lowest one from 21 RMSEs are shown in the following algorithm figure, Fig (7). For the Den2Num2 Model, the Algorithm could generate 21 RMSEs and respective models. After comparing all RMSEs, the lowest one and its related model would be picked up for further analysis. Figure, Fig(8), shows all highest to lowest RMSEs and their columns combinations.

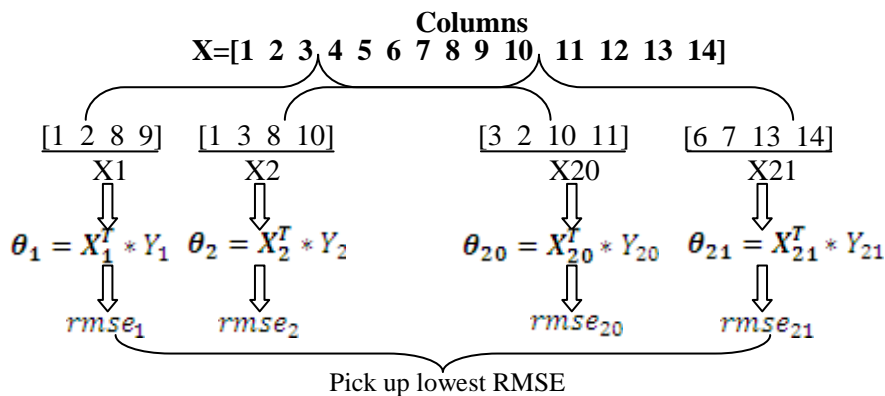


Figure (7): Algorithm to compute lowest RMSE

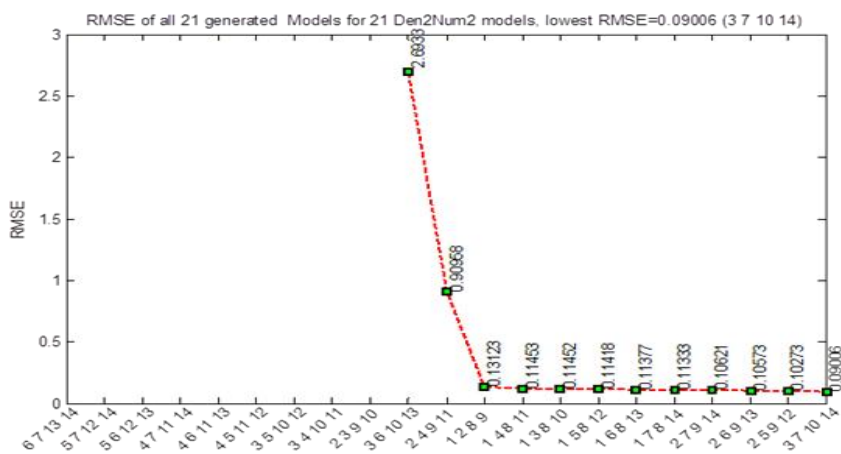


Figure (8): RMSEs comparison for all 21 generated models

For the other modeling cases such as; (Den 3 Num 3, Den 2 Num 3, Den 3 Num 2, Den 4 Num 3, Den 3 Num 4, Den 4 Num 4), the above-mentioned Algorithm is applied in which, all possible combination of columns pairs are first established then calculate estimated sates, generate Model Transfer Functions and compute RMSEs for all to pick up lowest RMSE with best approximation. The following Table (4,a) and Table (5,a) provide details of RMSE (Training part), RMSE (Checking part), and total RMSE each model for both Trolley and Load Swing simulations. Table (4,b) and Table (5,b) show the lowest RMSEs such as; RMSE=0.014485 for Trolley model Den 4 Num 4, and RMSE=1.616 for Load Swing Model Den 3 Num 4.

TrolleyCartRMSEDenNum					RMSEs (Trolley)		
RMSEs (Tr)	RMSEs (Ch)	RMSE	Den	Num	Den	Num	RMSE
0.062355	0.11777	0.09006	2	2	0.014485	4	4
0.062837	0.16689	0.036442	3	3	0.036442	3	3
0.063118	0.11684	0.089978	2	3	0.079812	4	2
0.058021	0.1539	0.10596	3	2	0.080342	4	3
0.055614	0.10507	0.080342	4	3	0.08938	2	4
0.092217	0.099155	0.095686	3	4	0.089978	2	3
0.046732	0.13316	0.014485	4	4	0.09006	2	2
0.06305	0.11571	0.08938	2	4	0.095686	3	4
0.042649	0.11697	0.079812	4	2	0.10596	3	2

LoadSwingRMSEDenNum					RMSEs (LS)		
RMSEs (Tr)	RMSEs (Ch)	RMSE	Den	Num	Den	Num	RMSE
0.82772	2.4455	1.6366	2	2	1.616	3	4
0.82523	2.444	1.6346	3	3	1.6281	4	3
0.82772	2.4455	1.6366	2	3	1.6345	4	4
0.82771	2.4455	1.6366	3	2	1.6346	3	3
0.81693	2.4393	1.6281	4	3	1.6364	4	2
0.8021	2.43	1.616	3	4	1.6366	3	2
0.82509	2.444	1.6345	4	4	1.6366	2	4
0.82772	2.4455	1.6366	2	4	1.6366	2	3
0.82751	2.4454	1.6364	4	2	1.6366	2	2

Table(5.a): RMSE(Tr) and RMSE (Ch) for Load Swing

Table(5.b): from lowest to highest RMSE for Load Swing

Using modified Linear Least Square with 7 past inputs/outputs data, Den 4 Num 4 trolley model provides lowest RMSE 0.046732, Fig (9,a) and while load swing model Den3 Num4; appeared to have better approximation with RMSE 0.08021, Fig (9,b).

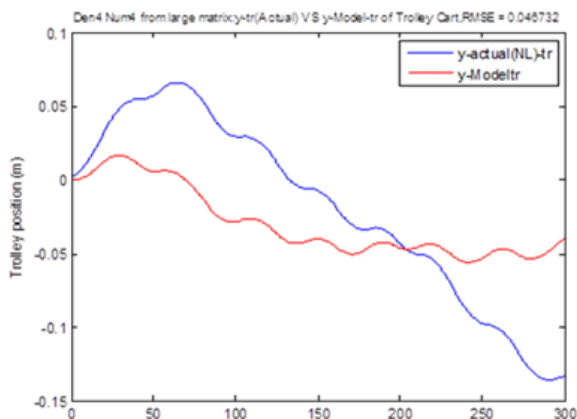


Figure (9,a): Den 4 Num 4 Trolley model

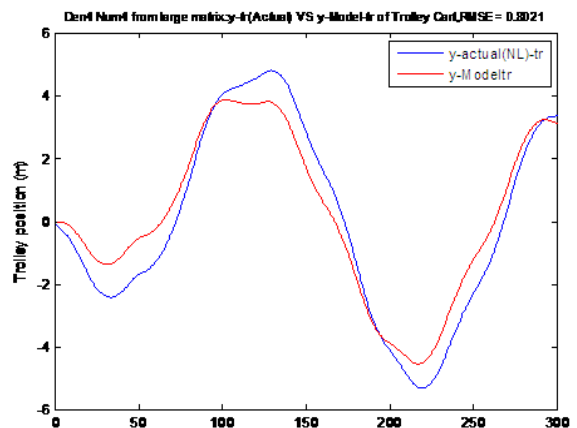


Figure (9,b): Den 3 Num 4 Load Swing model

6.2 Model Structure and RMSE comparison

From a simple Linear Least Square approach, trolley model structure with 3 Denominator 3 Numerator Coefficients appeared to have lowest RMSE=0.095495 while 3 Denominator 2 Numerator Coefficients load swing model has RMSE= 1.6366 respectively. Likewise in Least Square using 7 past inputs/outputs data approach, model structures with 4 Denominator 4 Numerator Coefficients appeared to have lowest RMSE (0.014485) while 3 Den 4 Num Load Swing model has lowest RMSE (1.616), as shown in Table (6). Those developed models from both approaches would be transformed into continuous state-space for control and performance comparison purposes.

According to the above simulation results, Linear Least Square with 7 past inputs/outputs data, produced lower RMSE than using simple linear least square approach which uses immediate past inputs/outputs data. Those selected models from both approaches would be implemented in controller designs and analyzed their responses.

	Den	Num	Model		RMSE
Least Square	3	3	Trolley	$\frac{4.924e - 005q^{-1} + 4.964e - 010q^{-2} - 4.924e - 005q^{-3}}{1 - 3q^{-1} + 3q^{-2} - q^{-3}}$	0.095499
Least Square	3	2	Load Swing	$\frac{-0.002723q^{-1} - 0.002722q^{-2}}{1 - 1.999q^{-1} + q^{-2} - 7.993e - 005q^{-3}}$	1.6366
Least Square with 7 past IOs	4	4	Trolley	$\frac{8.07e - 005q^{-1} + 1.321e - 005q^{-2} + 3.682e - 005q^{-3} - 2.993e - 005q^{-4}}{1 - 1.703q^{-1} + 1.026q^{-2} - 0.8826q^{-3} + 0.5598q^{-4}}$	0.014485
Least Square with 7 past IOs	3	4	Load Swing	$\frac{-0.002723q^{-1} - 0.002721q^{-2} + 2.736e - 006q^{-3} + 1.992e - 007q^{-4}}{1 - 2q^{-1} + 1.001q^{-2} - 0.0002781q^{-3}}$	1.616

Table(6): Model Structure and RMSE comparison

7. LQR CONTROLLER IMPLEMENTATION AND REFERENCE TRACKING

Linear quadratic regulation method is implemented for this overhead crane to determine the state-feedback control gain matrix K. LQR needs two parameters, Q and R weighting matrices which will balance the relative importance of the control effort (u) and error (deviation from 0), respectively, in the cost function J. Initially, $Q = \rho C^T C$ with $\rho = 1$ and $R = 1$ were assumed,[2]. The cost function corresponding to this and Q and R places equal importance on the control and the state variables outputs (cart's position x , and the pendulum's angle θ).

7.1 Q, R weighting matrices and Cost function J

Q and R matrices are considered as diagonal, $q \times q$ Matrix is positive-definite ($x^T Q x > 0$ for every nonzero vector, x) and were adjusted by hit and trial method to obtain the desired responses. The element in the (2, 2) position of Q represents the weight on the cart's position and the element in the (5,5) position represents the weight on the pendulum's angle. The input weighting R will remain at 0.001. In order to reach faster stabilization, it has been put more weights on the states as; $x = 30$, and $\theta = 20$ while R=0.001 to produce good controller gain K matrix. The following cost function J, Eq.(10), is considered to define the tradeoff between regulation performance and control effort of x-directional motion with the states $x, \dot{x}, \theta, \dot{\theta}$:

$$J = \int_0^{\infty} [x^T Q_x x + \theta_x^T Q_{\theta} \theta_x + u^T R u] dt \tag{10}$$

$$J = \int_0^{\infty} [x^T(t) \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 30 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 20 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} x(t) + u^T(t) * 0.001 * u(t)] dt$$

The plant state equations can be written as follow;

$$\dot{x}(t) = Ax(t) + B_u u(t), \text{ and } y = Cx(t) + Du(t)$$

7.2 Linear Quadratic Regulator (LQR) Design

After Q, R weighting matrices are established and controller gain K values are computed, LQR Controller design, Eq.(11), with step response would run for the system performance, Fig (10). Initially, the system use Q (2,2)=Q (5,5)=1 and R=1 and it showed the plot was not satisfactory. Both cart and pendulum responses overshoot. To improve their settling times and reduce rise time, Q (2,2)= 30, Q (5,5)=20 and R=0.001 are selected after several trials.

$$\begin{aligned} \dot{X} &= Ax + B(-Kx), \text{ and } Y = Cx + D(-Kx), \\ u &= -R^{-1}B_u^T P(t)x(t) = -Kx \end{aligned} \tag{11}$$

$$\dot{X} = (A - BK)x, \quad Y = (C - DK)x$$

Whereby, the optimal feedback gain is, $K = -R^{-1}B_u^T P(t)$.

In the following Riccati Equation in which P is the steady state solution that yields a unique optimal control to minimize the cost function, J, [2].

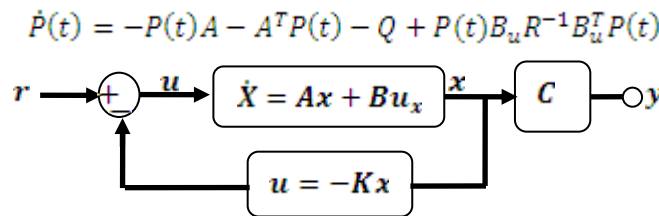


Figure (10): LQR Controller Design

In this overhead crane system, LQR performance of Trolley cart models, Den3 Num3 and Den4 Num4, are compared. The following figure, Fig(11) shows, Den4 Num4 model of modified Linear Least Square using 7past inputs/outputs dataset appeared to have shorter rise time and reach stability before 3 seconds while Den3 Num3 model of simple Least Square approach still fluctuating with longer rise time though the same Q and R weighting matrices have been applied.

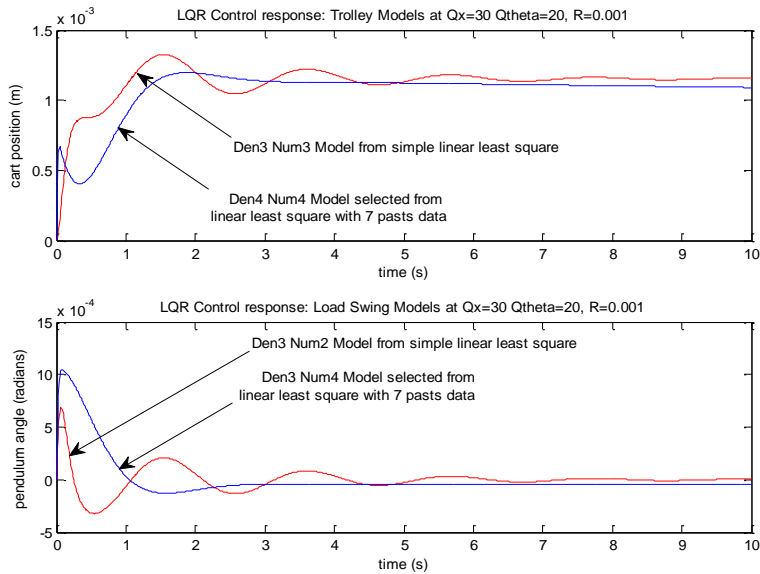


Figure (11): LQR-Controlled models response comparison

For Load swing performance, again Den3 Num4 from Linear Least Square using 7past inputs/outputs data appeared to reach zero swing angle in less than 3 seconds with less rise time and shorter settling time compared to Den3 Num2 model from simple least square. Besides, Linear Least Square using 7past inputs/outputs data produce more models, in this case up to 35 models have been analyzed before selecting the best fit model for control purpose. In this specific case of up to 600 data points, Den4 Num4 and Den3 Num4 models with lowest RMSEs have been picked for further reference tracking performance analysis. Though the controller response is better, steady-state errors still appear and therefore reference input tracking has been implemented to achieve desired inputs.

7.3 Reference Point tracking with full state feedback control

Adding the reference input to the system, $u = -Kx + r$ can lead to steady state errors. Pre-multiply r by carefully chosen matrix N . In order to create pre-compensation \bar{N} , there are two possible ways such as; full-state feedback, and full-state feedback with full-order observer, [7]. In this case, full-state feedback reference input tracking would be used. The following equations, Equ(12) and Equ(13), and figures, Fig (12,a) and Fig(12,b) explain the pre-compensation \bar{N} calculation. Full-state feedback controller form is;

$$u = -Kx + \bar{N}r \text{ where } \bar{N} = N_u + KN_x \tag{12}$$

And therefore, full-state regulating input is; $u = -Kx + (N_u + KN_x)r$

$$u = N_u r - K(x - N_x r) \tag{13}$$

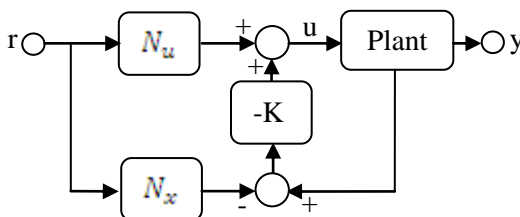


Figure (12.a): Full-state Feedback reference tracking

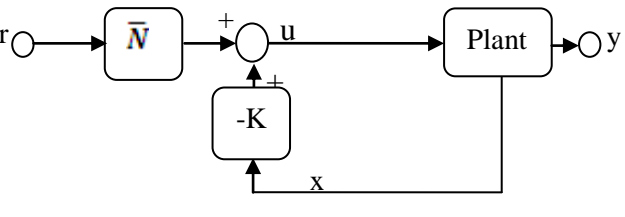


Figure (12.b): LQR Controller Design with pre-compensator

Full-state feedback would regulate steady-state output y_{ss} to desired steady-state value r_{ss} . In this trolley cart X directional motion, to track a constant desired position, x_{ss} (ss=steady-state) with control, u_{ss} , the control equation is;

$$\begin{aligned} 0 &= Ax_{ss} + Bu_{ss} \{for\ steady\ state,\ \dot{x} = 0\} \\ y_{ss} &= Cx_{ss} + Du_{ss} \end{aligned}$$

By substituting

$$\begin{aligned} x_{ss} &= N_x r_{ss}, \text{ and } u_{ss} = N_u r_{ss}, \\ 0 &= AN_x r_{ss} + BN_u r_{ss} \\ y_{ss} &= CN_x r_{ss} + DN_u r_{ss} \end{aligned}$$

When $y_{ss} = r_{ss}$ in the steady-state

$$\begin{aligned} 0 &= AN_x r_{ss} + BN_u r_{ss}, \\ r_{ss} &= CN_x r_{ss} + DN_u r_{ss} \end{aligned}$$

The above equations can be formed as state-space;

$$\begin{bmatrix} 0 \\ 1 \end{bmatrix} r_{ss} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} N_x \\ N_u \end{bmatrix} r_{ss} \text{ (or) } \begin{bmatrix} N_x \\ N_u \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

However, when computing the large matrices inverse, $\begin{bmatrix} A & B \\ C & D \end{bmatrix}^{-1}$, it gives error. To be able to solve for larger matrices, this research applies the following way, Equ (14).

$$\begin{aligned} F &= \begin{bmatrix} A & B \\ C & D \end{bmatrix}, Z = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ N &= inv(F^T * F) * (F^T * Z^T) \\ N &= [N_x \quad N_{u1} \quad N_{u1} \quad \dots \quad N_{un}]^T \\ N_u &= N_{u1} + N_{u2} + \dots + N_{un} \\ \bar{N} &= N_u + KN_x, \text{ where K is the controller gain.} \end{aligned} \tag{14}$$

The established full-state feedback pre-compensators N_u and N_x are then applied in LQR-controlled, Fig (13), to achieve the desired input. The robustness of reference tracking produces perfect control stability on both trolley motion and pendulum load swing. Each Trolley desired position has been achieved in less than 3 seconds while load swing could be suppressed in 2 seconds time compared to nonlinear high fluctuating pendulum output, Fig (14). It shows that, the developed trolley and load swing linear models using modified linear least square with 7 past inputs/outputs are more reliable and LQR with pre-compensator just make the whole system perfect.

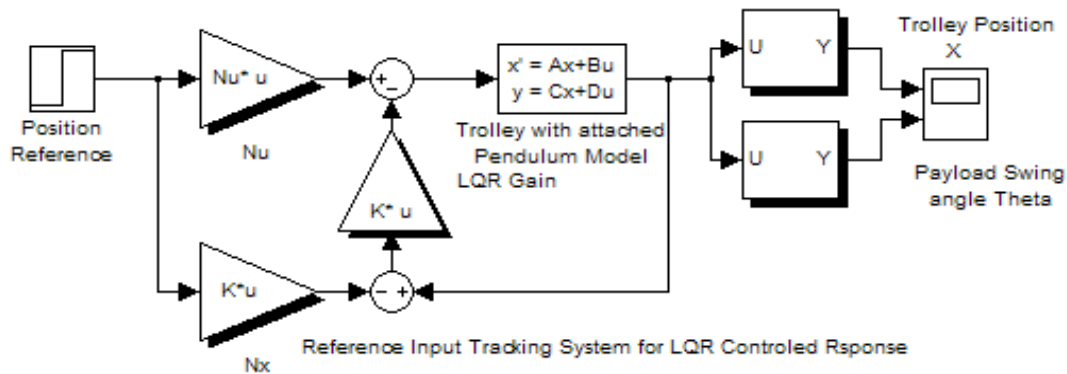


Figure (13): Reference Input Tracking with Full-state feedback

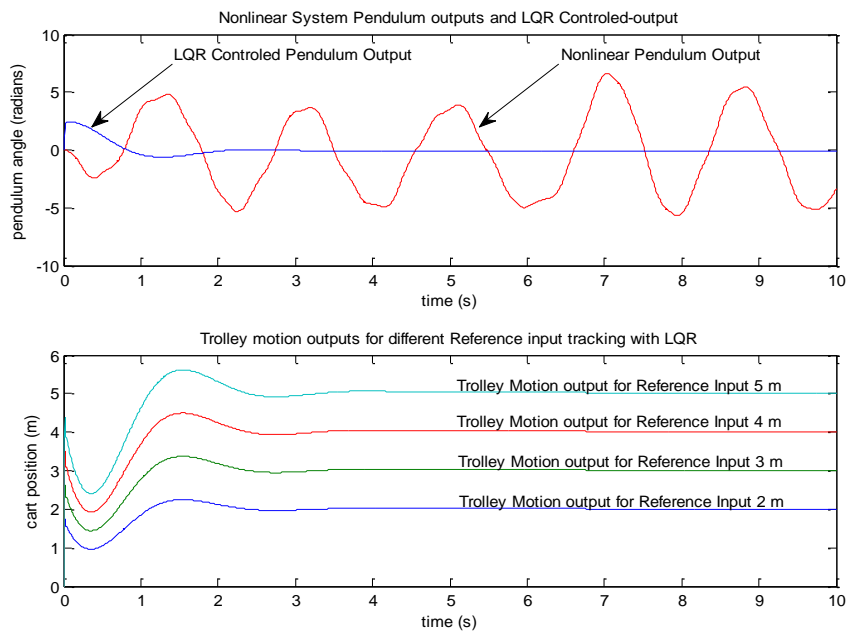


Figure (14): LQR Controlled response and Non-linear output

8. CONCLUSION

The main objective of this work is to design Overhead Crane experimental model using Simmechanic Visualization aiming not only to achieve real crane-like model feature but also to implement robust, fast and practical controller. Furthermore, it would also be a major milestone to deliver better crane modeling and control for the real operation instead of relying on lab-scaled model work. This development would close the gap between pure mathematical sketch with real time operation.

Throughout this work, trolley translational motion with attached pendulum is designed which represents exact lab model measurements and features. Practical considerations, such as joints actuation, moment of inertia, and the gravity, are taken into account. In addition, friction effects are included in the design using a friction-compensation technique.

To accomplish the objective, collected simulation data are then used in linear least square system identification to produce predicted models. To have better estimated model, 7 past outputs and 7 past inputs dataset matrix Algorithm was designed to produce more models. Root Mean Square

Errors checking has then been implemented to compare predicted models against the actual output.

Once the best fit linearized models have been established, those were implemented in reference input tracking-LQR controller design. The simulation result shows proposed control scheme guarantees both rapid damping of load swing and accurate control of crane position.

Since the major milestone of physical modeling using simmechanic visualization was achieved, the future step is to design 3D Tower Crane physical model which would have Trolley translation, Jib rotation, and Hoist up/down motion. Furthermore, jib oscillation would be thoroughly analyzed to suppress payload swing during the operation.

9. REFERENCES

- [1] Bei Na Wei, (2008). "An Innovative Approach to Overhead Crane Control". The University of New, South Wales, Sydney, Australia.
- [2] Hesketh T. Linear quadratic control. Elec9733 Real Time Computing and Control Lecture Notes, Apr 2008.
- [3] Ilse C. F. Ipsen. (2009). "Numerical Matrix Analysis: Linear Systems and Least Squares". North Carolina State University, Raleigh, North Carolina.
- [4] Inteco (2010). "Tower Crane Control User Manual", Inteco Ltd. (Inteco User Manual, Version 1.4.)
- [5] Lee H.H. Modeling and control of a three-dimensional overhead crane. *Journal of Dynamic Systems, Measurement and Control*, 120(4):471–476, Dec 1998.
- [6] Omar, H. M. (2003). "Control of Gantry and Tower Cranes", Virginia Polytechnic Institute and State University, Blacksburg, Virginia, USA
- [7] Oishi Meeko. (2012). "State Space Reference Tracking and other control problems". (Lecture Notes:EECE360). Electrical and Computer Engineering, The University of British Columbia.
- [8] Quanser (2010). "3 Degrees of Freedom Crane User Manual", Quanser Ltd. (Quanser User Manual)
- [9] Raul G. Longoria.(2002) "Modeling of Mechanical System for Mechatronics Applications". The University of Texas, Austin, USA.
- [10] R. S. Caprari (Robert S.). (1992). "Least squares orthogonal polynomial approximation in several independent variables". Flinders University of South Australia.
- [11] Sahan S. Hiniduma Udugama Gamage and Patrick R. Palmer. "Hybrid System Modeling, simulation and visualization; A crane system". *Proceedings of SPIE Vol. 5097* (2003). Department of Engineering, University of Cambridge,UK
- [12] Shebel Asad, Maazouz Salahat,(2011). "Design of fuzzy PD-controlled overhead crane system with anti-swing compensation". *Journal of Engineering and Computer Innovations Vol. 2(3)*, pp. 51-58, March 2011. Academic Journals
- [13] Schlotter Michael (2003). "Multi Body System Simulation with SimMechanics. Dynamic Seminar. Darmstadt University of Technology