September 12, 2011

System Identification Via Linear Least Squares

A Report On The Characterisation Of
A Simple Cart System
Using Matlab

Wim Looman (*92692734*)
wgl18@uclive.ac.nz

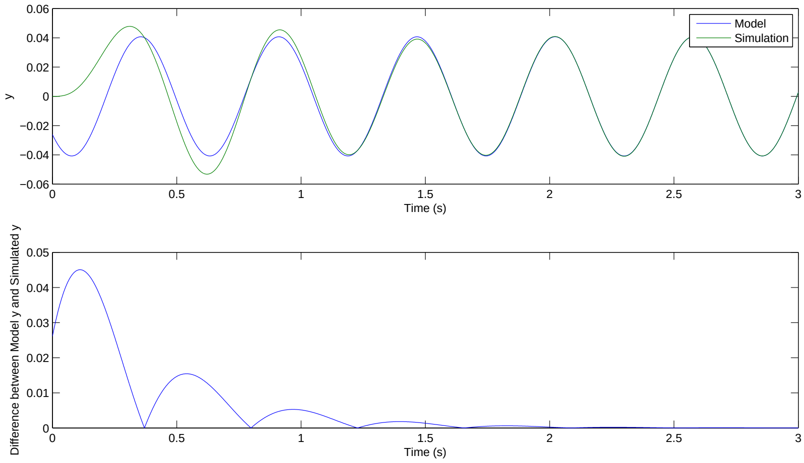

Figure 1: Model vs Matlabs simulated values

Introduction

Wherein the Aims and Concepts of the Report are Introduced

Blah blah blah

Section One and Two

Wherein a Function To Fake the Model Data is Written and Tested

Before developing a function to identify the system a test function was developed to later use on verifying the identification function. The first step in the development of this test function was creating a Matlab function that takes in the C and β parameters that define the system along with a frequency to evaluate it at (ω) and a time to evaluate it for (t_{end}). The function was simply Equation **??** evaulated at the specified C, β and ω with t ranging from 0–t_{end} at a frequency of 1 kHz.

As a verification of this function an additional function utilising Matlab's inbuilt `lsim` function was produced. This function was based around the transfer function from Equation **??** along with the input from Equation **??** to calculate the output.

Both models created were then plotted together with input values $C = 5$, $\beta = 1$, $K_p = 60$, $f = 1.8$ and $t_{\text{end}} = 3$. This can be seen in Fig. 1. Looking at this it can be seen that the steady state response of the two systems is tending to be the same. The major difference is in the initial response, this is because the model used in the first equation is purely a steady state model whereas the transfer function based model does base its output off the initial conditions which are assumed to be zero.

Noise Level	Median (C)	90% CI (C)	Median (b)	90% CI (b)
0.00	5.00000	0.00000	1.00000	0.00000
0.05	4.99905	0.00164	0.99985	0.00018
0.10	5.00216	0.00388	0.99993	0.00030
0.15	4.99518	0.00512	0.99969	0.00062
0.20	5.00194	0.00802	0.99885	0.00075
0.25	4.99897	0.00869	0.99937	0.00090
0.30	4.99082	0.01174	1.00069	0.00115
0.35	4.99939	0.01295	0.99886	0.00125
0.40	5.00423	0.01503	0.99959	0.00162

Table 1: Testing the Identification Function

Section Three (a)

Wherein A Function To
Identify the System is Written

After writing this test function the actual identification function could be developed. This involved simply creating a function that matched equation **??**, the function takes in the y vector, t vector and ω frequency and produces the corresponding A_1 and A_2. From these two values the b and C values were able to be derived as equation **??** shows.

Section Three (b)

Wherein The Identification
Function is Tested

To check the identification function a series of “noisy” sample data streams were created and run through the function. These were based off the test function developed earlier with each y datum multiplied by a normally distributed value. The value used had a mean of 1 and a standard deviation varied between 0 and 0.4 in steps of 0.05. Each of these standard deviations was simulated 100 times and the median and confidence intervals of the series was calculated.

Table 1 and Figure 2 show the median and confidence interval calculated using the test function with $C = 5$, $\beta = 1$, $K_p = 60$, $f = 1.8$, $t_{\text{end}} = 1$ and a varying standard deviation. From these it was seen that the identification of the C and β values are quite accurate, the 90% confidence interval is less than 0.3% of the found value and the difference from the correct value is less than 0.2% in the worst case.

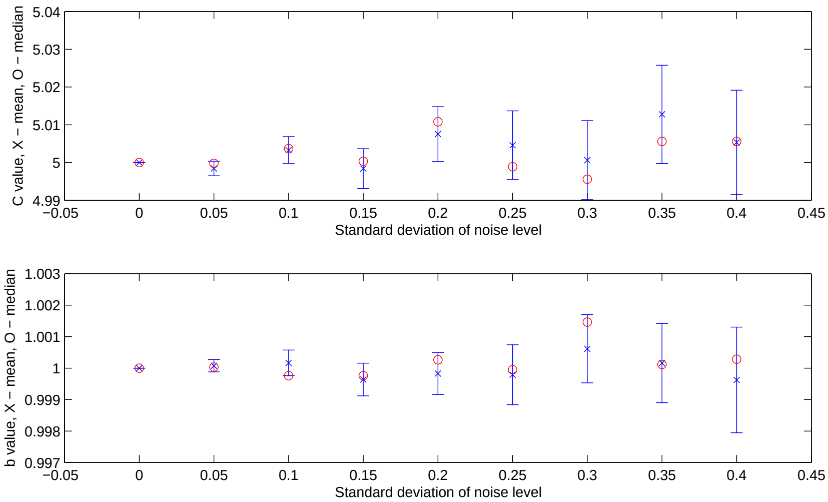

Figure 2: Testing the Identification Function

Section Four (a)

Wherein the Cart System is Identified

Now that the function to identify the system had been developed, tested and found to be accurate the actual characterisation of the cart could begin. The characterisation simply involved taking in the provided datasets and running them through the identification function. This resulted in a C and b value for each frequency at which the system was tested. As instructed the values derived from the seventh data set with a frequency of 1.8 Hz was used as the general model. The results of this identification are shown in Equation (1).

$$\begin{aligned} C &= 5.83523 \\ \beta &= 1.93596 \end{aligned} \tag{1}$$

Section Four (b)

Wherein the Identification is Ridiculed

To check the results of the identification the model was used to generate data at all 12 frequencies tested and this generated data was compared to the measurements. The comparison was performed by finding the absolute difference between the generated data and the measured data for each frequency, scaling this difference by the mean of the absolute measured data for that frequency and finally finding the median and 90% confidence interval of the relative error.

Figure 3 and Table 2 shows these medians and the 90% confidence interval found. Obviously the model is most accurate at the frequency it was derived from, the relative error is down to 0–6% showing the very good accuracy. The worst case is at $\omega = 16.3$ with a 90% CI of the error in the range 0–63%, this is

Frequency	Relative Median error	90% CI
0.6	0.26936	0.26175
0.8	0.30116	0.25065
1.0	0.32473	0.24677
1.2	0.31176	0.23621
1.4	0.24266	0.20618
1.6	0.11872	0.12534
1.8	0.02376	0.03305
2.0	0.11852	0.13654
2.2	0.18552	0.21159
2.4	0.22813	0.21008
2.6	0.28859	0.32453
2.8	0.28994	0.28366

Table 2: Error in the derived model

obviously an unacceptable level of error so the model is really only valid for the one frequency it was derived for, and maybe the ones either side depending on the use case.

The relative errors from each frequency were then concatenated and the median and 90% CI of the overall error was calculated. This came out to a median error of 0.21840 and a 90% CI of 0–49%. Again a most likely case of 22% error and a worst case of around 49% error is likely unacceptable for the majority of use cases of this model.

Figure 4 shows a histogram of the error values. Looking at this the bell shape of a Gaussian distribution can be easily seen.

Section Five (a)

Wherein the Model and Measured Are Compared

Section Five (b)

Wherein All the Models Are Compared

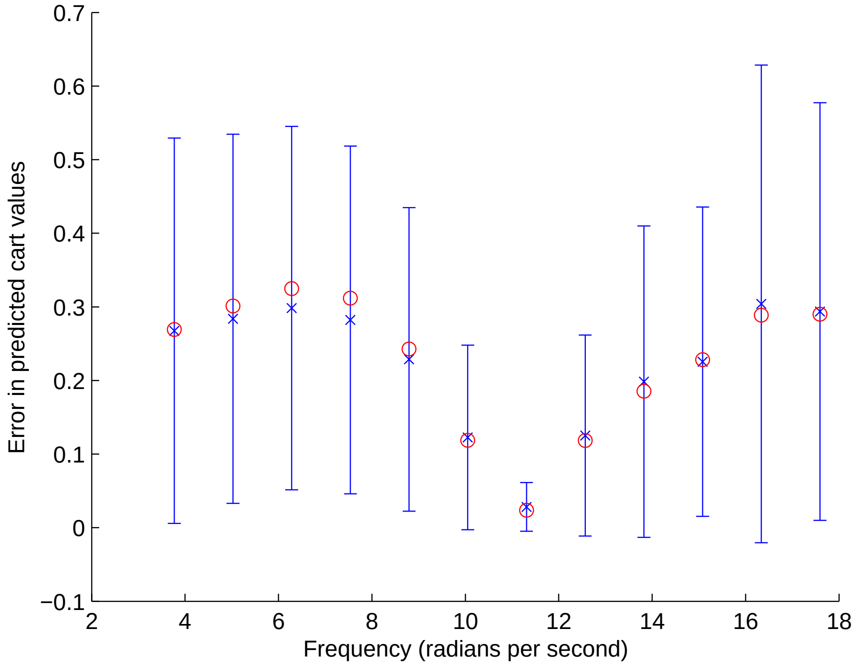

Figure 3: Error in the derived model

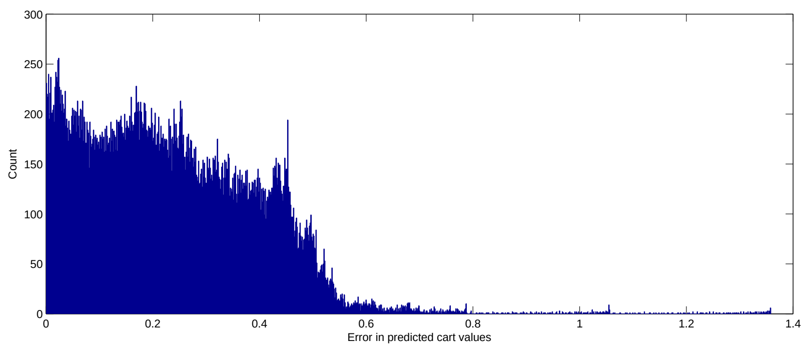

Figure 4: