Advanced Control Systems Workshop 5

Jake Langton [833857]

May

1 Mathematical Modelling

Equations of motion can be determined using Lagrange or Newton method can be used to model the active spring damping system.

$$M_s\ddot{x_2} = -k_s(x_2 - x_1) - B_s(\dot{x2} - \dot{x1}) + F_c$$
$$M_{us}\ddot{x_1} = k_s(x_2 - x_1) - B_s(\dot{x_2} - \dot{x_1}) - K_{us}(x_1 - z_r) - B_{us}(\dot{x_1} - \dot{z_r}) + F_c$$

Let $B_{us} = 0$ and

$$\begin{bmatrix} x1 & x2 & x3 & x4 \end{bmatrix} = \begin{bmatrix} x_1 & x_2 & \dot{x_1} & \dot{x_2} \end{bmatrix}$$

Rearrange and organise into state space matrices given the measurements

$$\begin{bmatrix} y1 & y2 & y3 & y4 \end{bmatrix}' = \begin{bmatrix} x_2 - x_1 & \dot{x_2} & x_1 - Z_r & \dot{x_1} \end{bmatrix}'$$

and the inputs F_c, Z_r $\dot{Z}_r$

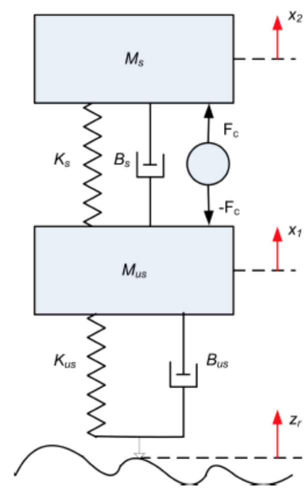


Figure 1: Diagram Of Model

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -(K_s + K_{us}) & K_s/M_{us} & B_s/M_{us} & -B_s/M_{us} \\ -K_s/M_s & K_s/M_s & -B_s/M_s & B_s/M_s \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1/M_{us} & K_{us}/M_{us} & 0 \\ 1/M_s & 0 & 0 \end{bmatrix} \qquad C = \begin{bmatrix} -1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \qquad D = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Discretisation of the plant model using forward Euler and sampling rate T yields $A_q = (I - AT)$ and $B_q = BT$.

2 Parameter Identification

Given that the system is presented as a black box the parameters are unknown. These parameters can be determined using least squares or recursive least squares. Given that MPC controllers require accurate models of the plant, through extensive experimentation it was determined that large quantities of datapoints were required to achieve the required accuracy. As a result of the numerical constraints of least squares approximation, recursive least squares was chosen as the method of parameter identification. This section will analyse the various factors which impact the accuracy of recursive least squares.

2.1 Recursive Least Squares

$$r_k = \begin{bmatrix} x_1(k) \\ x_2(k) \\ \dot{x_1}(k) \\ \dot{x_2}(k) \\ F_c(k) \\ Z_r(k) \\ \dot{Z}_r(k) \end{bmatrix}' \quad y_k = \begin{bmatrix} x_1(k) \\ x_2(k) \\ \dot{x_1}(k) \\ \dot{x_2}(k) \end{bmatrix}' \quad \Psi_N = \begin{bmatrix} r_k \\ r_k + 1 \\ . \\ . \\ . \\ r_k + N \end{bmatrix}' \quad Y_N = \begin{bmatrix} y_k \\ y_k + 1 \\ . \\ . \\ . \\ y_k + N \end{bmatrix}'$$

$$\Theta_{LS} = (\Psi^T \Psi)^- 1 \Psi^T Y$$

$$\Theta_{RLS_{k+1}} = \Theta_{RLS_k} + L_{k+1}[y_{k+1} - r_{k+1}\Theta_{RLS_k}]$$

Recursive least squares Θ_{RLS} is a computationally efficient mechanism for updating parameters Theta. L_{k+1} is computed each iteration from previous state.

2.2 Hyper - Parameters

Hyper-parameters are the methods which the experimental set-up can be tuned to deliver optimal results. We will be discussing the following hyper-paramters.

- Sampling time (ts).
- Period of experiment
- Gaussian Noise, added to the measured outputs.
- Low Pass Filter, filters out the frequency components of the output signals less than 100 Hz.
- Parameters of the system, and their relative values can influence the efficacy of results. Due to the fact that K_{us} is orders of magnitude larger than the other paramteres oin the plant we will be experimenting with reducsing this number.

2.2.1 Experimental Method

- All hyper-paramters are defined
- A sequence of F_c and Z_r is chosen as the input and the output is used to calculate recursive least squares (RLS).
- Parameters are solved using simultaneous equations using the results of RLS.
- A model is created using these parameters and then simulated and evaluated using a different input.
- Results are evaluated by measuring the difference between the actual and model output for the first 10 seconds (sim error) and the average percentage difference between the estimated and true parameters (param error).

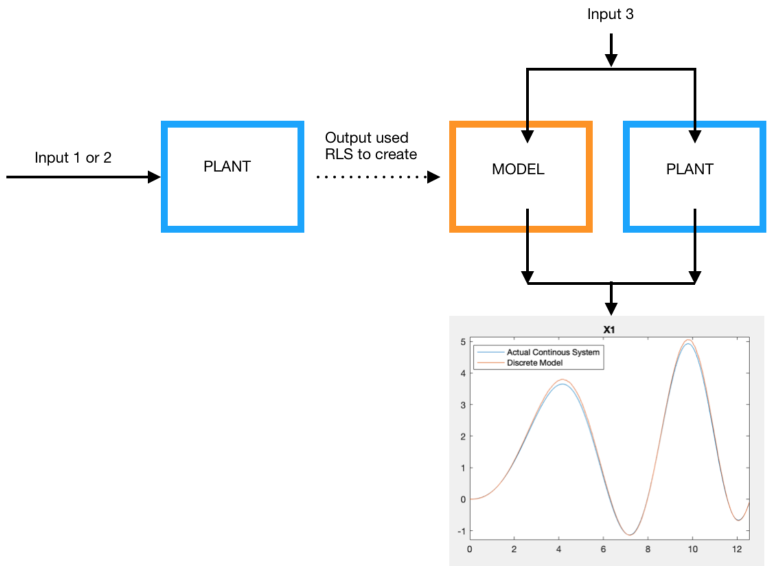


Figure 2: Experimental Method

2.2.2 Results

Experiment #	ts	Noise	Filter	K_{us}	Sim Time	Sim Error	Param Error
1	0.005	0	None	50050	50	0.3	57%
2	0.05	0	None	50050	500	2.8	288%
3	0.005	0	None	500.5	50	0.2	4.3%
4	0.005	0.05	None	50050	50	124	190%
5	0.0005	0.05	None	50050	50	92	314%
6	0.05	0.05	None	50050	50	67	97%
7	0.05	0.05	None	50050	500	23.1	110%
8	0.0005	0.05	YES	50050	50	15	99%
9	0.05	0.05	YES	50050	500	52	109%

Table 1: Experimental Results, Underlined variables indicate change since experiment 1

2.2.3 Discussion

In both experiments 1 and 2, 10,000 data points were collected. Although both improved performance, increasing the sampling time was far more effective. This is likely because the higher order dynamics of the system are captured. Sample rate significantly greater than 1/11 are required to capture the dynamics of the lower spring.

$$NaturalFrequency(\omega_{NK_{us}}) = \sqrt{k/m} = \sqrt{\frac{50050}{325+65}} = 11$$

Experiment 3 indicates that simply reducing K_{us} to a value less extreme relative to the rest of the system improves accuracy. This is likely because a high value of K_{us} results in a high natural frequency. Lowering K_{us} allows the dynamics to be properly captured with a lower frequency input and slow sampling rate.

$$NewNaturalFrequency(\omega *_{N\ K_{us}}) = \sqrt{\frac{500.5}{325+65}} = 1.1$$

Experiment 4 illustrates how noise drastically reduces accuracy, in some cases to the point where the model system is unstable.
Experiment 5 and 6 indicates that in the presence of noise a faster sampling time does not necessarily give better results. In fact it seems lower sampling rates are beneficial. This is likely due to the fact that we are sampling from a perfect linear system that behaves well during intervals. Thus for longer samples the noise of the measurement is less significant to the change in output.
Comparing experiments 5 and 8 as well as 7 and 9 indicates that a filter can improve accuracy when faster sampling rates are required so long as delays are not introduced. The same is not true for lower sampling rates.
It is important to note that only experiments 1 2 and 3 resulted in simulated responses that followed the true response beyond the 10 second mark.

2.2.4 Input Experimentation

The input chosen is very important in RLS identification. As a minimum it needs to be constantly exciting. For the first set of experiments a chirp signal of oscillating amplitude was used. The chirp signals were told to start and frequencies of 0.01Hz and sweep to 1Hz and 10Hz for F_c and Z_r respectively. This was used to ensure a wide range of relevant frequencies and amplitudes excited the system. This provided good results when the parameters derived were fed into a simulation with the same inputs. When the model was evaluated with respect to another input the results were extremely poor. It is hypothesised that this is because that there was not a wide enough variety of inputs. To test this a second signal with a wider variety of inputs was used and evaluated with respect to a third unique signal.

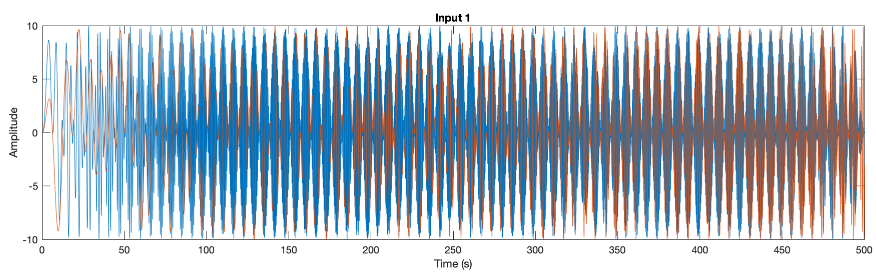


Figure 3: Chirp signal up to 10Hz and 1Hz for F_c and Z_r respectively

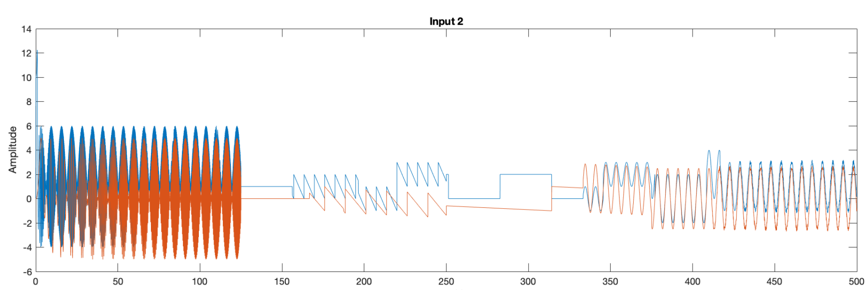


Figure 4: Second Chirp Signal Input, Chirp signal up to 20Hz and 5Hz for F_c and Z_r respectively, and additional sawtooth signal and steps

ts	Noise	Filter	K_{us}	Sim Time
0.05	0.05	NO	50050	500

Exp #	Input	Evaluate	Sim Error	Param Error
1	Input 1	Input 1	3.4	98%
2	Input 1	Input 3	23.1	110%
3	Input 2	Input 3	3.83	91.3%

As the results indicate,the second signal provided significant, tangible improvements to the first. It was confirmed through additional experimentation that the increased frequency of the chirp signal was found to increase accuracy but the effect was limited by the sampling rate. Furthermore, the addition of steps and sawtooth signals also improved performance of the parameter identification.

2.2.5 Summary

Given that MPC is feedback control the longer the model remains close to the actual plant the better. This allows us to increase the time horizon without inducing large amounts of error which could cause a poor optimisation and a poor choice of input.
In retrospect a 3rd error parameter would have been useful relating to the time which it takes for the simulation to deviate significantly from the true plant.

2.3 MPC

2.3.1 Model Reformulation

The model must be reformulated such that the state variables contain information of Z_r.
Two Methods can be used to approach this. Method 1 keeps $X = (x_1, x_2, \dot{x}_1, \dot{x}_2)'$ and uses IMP to estimate Z_r. This assumes you have a measurement of Z_r. Method 2 uses $X = (x_2 - x_1, \dot{x}_2, x1 - z_r, \dot{x}_1)'$ which is the form of the measurement but not the form of the model developed. This formulation provides more convenient forms of the cost functions and state variable constraints and does not require an explicit measurement of F_c. If an explicit measurement of F_c is possible, IMP can also be applied.

Method 1:
Let IMP be a state space representation of the input.

$$A_{aug} = \begin{bmatrix} A & 0 \\ 0 & IMP \end{bmatrix} B_{Aug} = \begin{bmatrix} 0 \\ 0 \\ 1/M_{us} \\ 1/M_s \\ 0 \\ . \\ . \end{bmatrix}$$

Method 2:
Where C is a change of basis from $(x_1, x_2, \dot{x}_1, \dot{x}_2) \rightarrow (x_2 - x_1, \dot{x}_2, x1 - z_r, \dot{x}_1)$

$$A_{aug} = C * A * C^{-1}$$

$$A_{Aug} = \begin{bmatrix} 1 & 0 & 0 & -1 \\ -ks/M_s & -B_s/M_s & 0 & B_s/M_s \\ 0 & 0 & 0 & 1 \\ K_s/M_{us} & B_s/M_{us} & k_{us}/M_{us} & -(B_{us} + B_s)/M_{us} \end{bmatrix}$$

$$B_{Aug} = \begin{bmatrix} 0 \\ 1/M_s \\ 0 \\ 1/M_{us} \end{bmatrix}$$

2.3.2 Cost Formulation

- Comfort: Minimise $\dot{x}_2$
- Suspension Travel: Minimise $(x_2 - x_1)$
- Road Handling: Maximise friction, Minimise $(x_1 - z_r)$

Costs take take the form:

$$\sum_{i=1}^{N}(x_{k+i}^T Q x_{k+i} + u_{k+i}^T R u_{k+i}) + x_{k+N}^T P x_{k+N}$$

Let the feedback be in the form:

$$x_k = (x_1, x_2, \dot{x}_1, \dot{x}_2, Z_r)$$

Cost Q determined by the co-coefficients of x, U is determined experimentally. Although there is no explicit cost with regard to U, a value needs to be assigned to ensure that the controller does output an unrealistic controller action.

$$\begin{aligned} f(x) = b\dot{x}_2^2 + a(x_2 - x_1)^2 + c(x_1 - Z_r)^2 &= b\dot{x}_2^2 + ax_2^2 - ax_2x_1 + ax_1^2 + cx_1^2 - cx_1Z_r + cZ_r^2 \\ P &= Q \\ R &= 1 \end{aligned}$$

Formulations of the Q matrix depend on which method a model is developed.

Method 1:

$$Q = \begin{bmatrix} a+c & 0 & 0 & 0 & . & 0 \\ -a & a & 0 & 0 & . & 0 \\ 0 & 0 & 0 & 0 & . & 0 \\ 0 & 0 & 0 & b & . & 0 \\ . & . & . & . & . & . \\ -c & 0 & 0 & 0 & . & c \end{bmatrix}$$

Please note that the dotted area of the Q matrix corresponds to variables related to the IMP which are not Z_r which is depicted in the bottom row.

Method 2:

$$Q = \begin{bmatrix} a & 0 & 0 & 0 \\ 0 & b & 0 & 0 \\ 0 & 0 & c & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

2.3.3 Problem Formulation

Measured Disturbances: Z_r
Initial Conditions: $(x_1, x_2, \dot{x}_1, \dot{x}_2, Z_r) = (0, 0, 0, 0)$

$$\Psi = diag\{R, R, R...\}$$

$$\Omega = diag\{Q, Q....P\}$$

$$\Phi = \begin{bmatrix} A \\ A^2 \\ . \\ . \\ A^N \end{bmatrix} \Gamma = \begin{bmatrix} B & 0 & .. & 0 \\ AB & B & .. & 0 \\ . & . & . & . \\ . & . & . & . \\ A^{N-1}B & ... & AB & B \end{bmatrix}$$

$$F = 2\Gamma'\Omega\Phi$$

$$G = 2(\Psi + \Gamma'\Omega\Gamma)$$

2.3.4 Constraints

for $X = (x1, x2, x3, x4) = (x_2 - x_1, \dot{x}_2, x_1 - Z_r, \dot{x}_1)$

$$\begin{aligned} |F_c| \leq 2500N &\rightarrow -2500 \leq F_c \leq 2500 && \rightarrow F_c \leq 2500, -F_c \leq 2500 \\ |x_2 - x1| \leq 0.2m &\rightarrow -0.2 \leq x_2 - x1 \leq 0.2 && \rightarrow x1 \leq 0.2, -x1 \leq 0.2 \\ |\dot{x}_2| \leq 0.4m/s &\rightarrow -0.4 \leq \dot{x}_2 \leq 0.4 && \rightarrow x2 \leq 0.4, -x2 \leq 0.4 \end{aligned}$$

Need constraints in the form: $M_i x_i + E_i u_i \leq fi$ and $G x_N \leq h$, Let $G = M_i$ and $h = 0$

$$M_i = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix} E_i = \begin{bmatrix} 1 \\ -1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} f_i = \begin{bmatrix} 2500 \\ 2500 \\ 0.2 \\ 0.2 \\ 0.4 \\ 0.4 \end{bmatrix}$$

$$\begin{bmatrix} M_0 \\ 0 \\ . \\ . \\ 0 \end{bmatrix} X_0 + \begin{bmatrix} 0 & . & . & 0 \\ M1 & . & . & 0 \\ . & . & . & . \\ . & . & . & . \\ 0 & . & . & G \end{bmatrix} \begin{bmatrix} X_0 \\ X_1 \\ . \\ . \\ X_N \end{bmatrix} + \begin{bmatrix} E_0 & ... & 0 & \\ . & . & . & . \\ . & . & . & . \\ 0 & ... & E_{N-1} & \\ 0 & ... & 0 & \end{bmatrix} \begin{bmatrix} U_0 \\ . \\ . \\ U_{N-1} \\ U_N \end{bmatrix} \leq \begin{bmatrix} f_0 \\ . \\ . \\ f_{N-1} \\ h \end{bmatrix}$$

Such that $DX_0 = MX + \epsilon U \leq c$
Organise into form for Quadprog MATLAB function such that $J = M\Gamma + \epsilon$ and $W = -M\Phi - D$
In Quadprog function constraints are defined by $A = J$, $b = c + Wx_0$, $H = G$, $f = Fx_0$

2.3.5 Results

All simulations are ran for 10 second. Error a, b, and c count the accumulated values of $x2 - x1$, x_2, $x1 - Z_r$ respectively. Future input Z_r not known

Effect Of Controller:

Controller makes a significant improvement compared to now control with the blue, red and yellow lines being minimised significantly. However it is a fairly poor controller and cannot be improved by increasing gain significantly.

Control Result

Freq	Control	N	err a	err b	err c
1/5Hz	N0	5	66.3	101	1

With Constrained Feedback

Freq	Control	N	err a	err b	err c
1/5Hz	YES	5	8.1	23	0

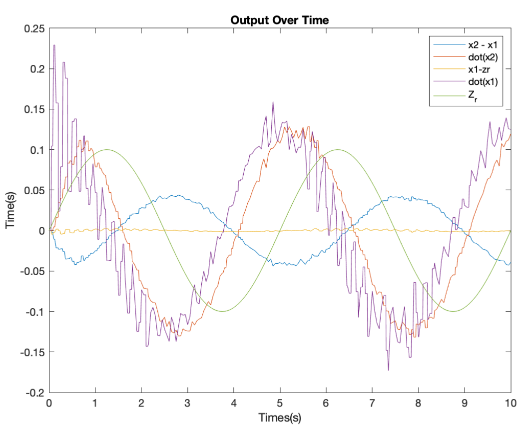


Figure 5: Diagram Of Model

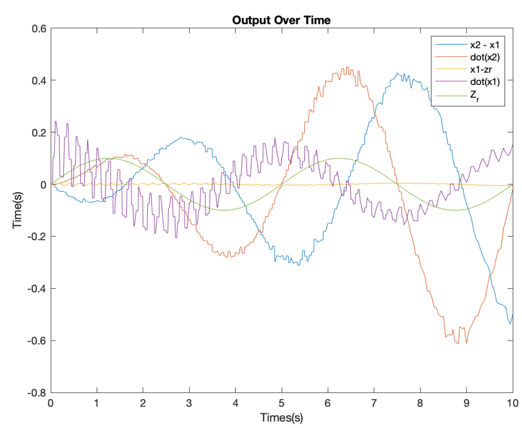


Figure 6: Diagram Of Model

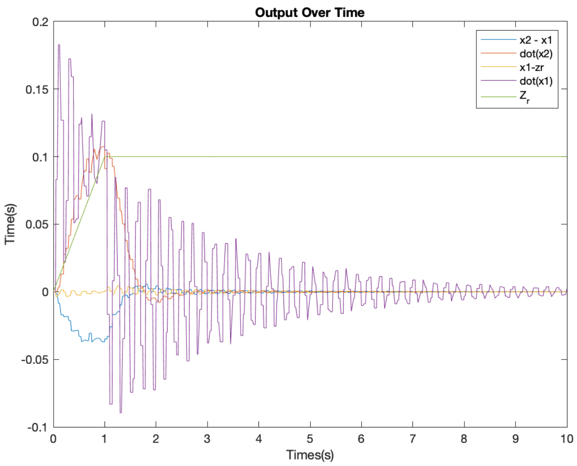
Changing Q

Freq	Control	N	a	b	c	error a	error b	error c
1/5Hz	YES	N	10	100	50	8.1	23	0
1/5Hz	YES	N	500000	0	0	8.7	24	0
1/5Hz	YES	N	0	0	0	8.0	23	0

Q has virtually no influence on output performance. This may be because the MPC does not have a direct measurement of $\dot{Z}_r$ to update the state variable $\dot{Z}_r$ - $\dot{x}_1$.

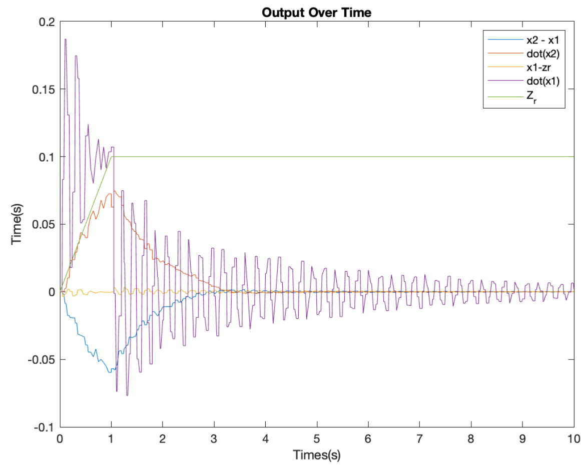
Q and R !=0

Control	N	error a	error b	error c
Yes	5	2.4	7	1



Q and R ==0

Control	N	error a	error b	error c
YES	5	4.4	6	0



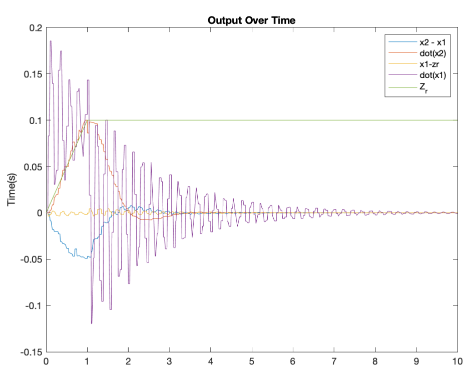
As soon as R!=0 results are comparable to all other responses. Changing Q once again does not influence the output. The controller works decently enough by only relying on the constraints. It may be that the terminal set constraint is forcing the states to converge over time.

Look ahead Time

Increasing look-ahead time does not result in improvements. In fact it causes further issues, likely because the attempt made to estimate $x_1 - Z_r$ is very poor. Future states, are estimated using the A matrix. In the A_{Aug} matrix $\dot{x_1} - \dot{z_r} = \dot{x_1}$. As Z_r is not measured directly and not information about the signal are known the prediction is poor. An augmented IMP matrix is required.

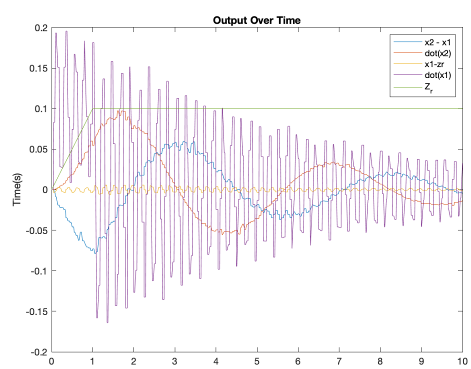
N=10

Control	N	error a	error b	error c
YES	10	3.2	6	0

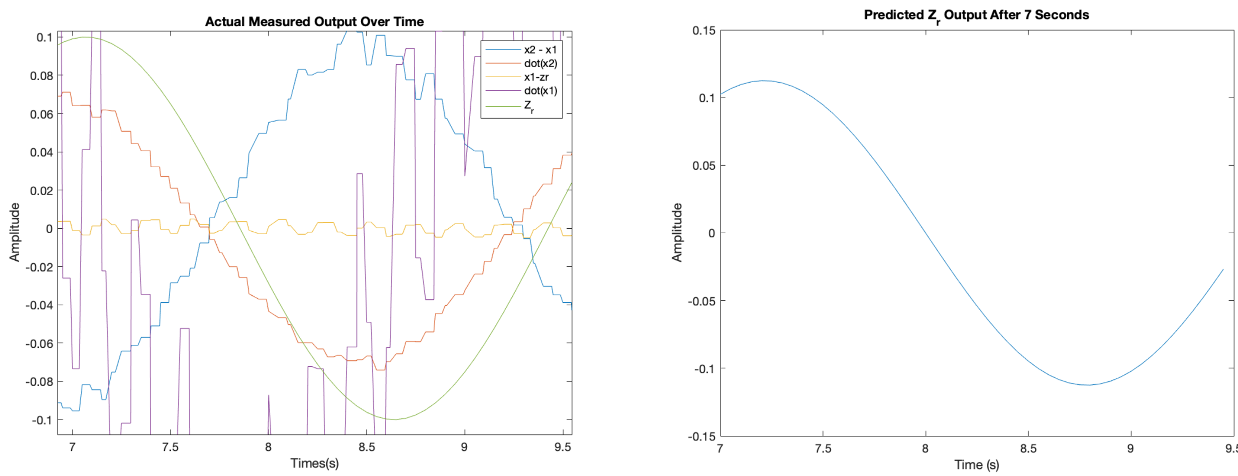


N=100

Control	N	error a	error b	error c
YES	100	18	24	1



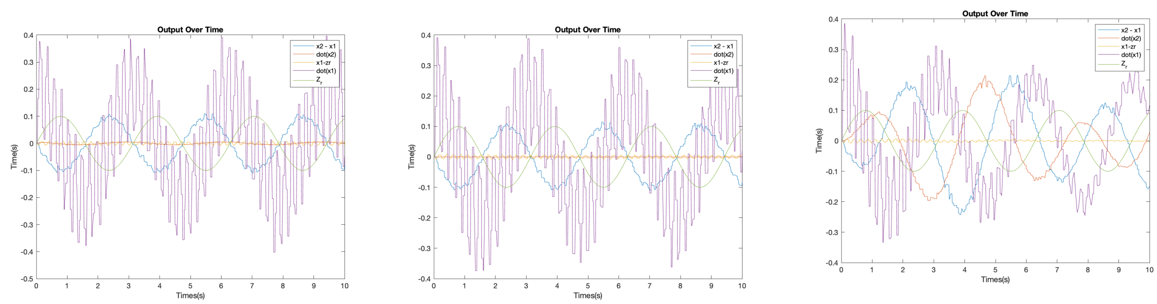
Internal Model Principle



(a)

(b)

Figure 7: Notice how the green Z_r in (a) matches closely with the predicted Z_r (b) at the 7 second mark. For this experiment N=50



(a) Constraints Applied: Total Error for $\dot{x}_2$ = 2

(b) No Constraints Applied: Total Error for $\dot{x}_2$ = 0

(c) Constraints Applied, Equal cost (a=b=c) Total Error for $\dot{x}_2$ = 48

Figure 8: For these experiments costs were only applied to the velocity of the sprung mass in order to maximise comfort. This resulted in an extremely stable $\dot{x}_2$ (a) especially when there were no constraints. N=50. When costs are applied to a, b and c they have conflicting goals. To reduce $\dot{x}_2$ you will need to increase $x2 - x1$ while driving over a bump. This results in poor performance as illustrated in figure c.

This is achieved by creating an augmented matrix of A and B.

$$\begin{bmatrix} \dot{x}_2 - \dot{x}_1 \\ \ddot{x}_2 \\ \dot{x}_1 - \dot{Z}_r \\ \ddot{x}_1 \\ \dot{Z}_r \\ \ddot{Z}_r \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & -1 & 0 & 0 \\ -ks/M_s & -B_s/M_s & 0 & B_s/M_s & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & -1 \\ K_s/M_{us} & B_s/M_{us} & k_{us}/M_{us} & -(B_{us}+B_s)/M_{us} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \omega \\ 0 & 0 & 0 & 0 & -\omega & 0 \end{bmatrix} \begin{bmatrix} x_2 - x_1 \\ \dot{x}_2 \\ x_1 - Z_r \\ \dot{x}_1 \\ Z_r \\ \dot{Z}_r \end{bmatrix} + BU$$

This results in impeccable results when constraints are removed and no cost is assigned to action.

Frequencies

For this section N=50 and we will be applying costs to maximise comfort (ie $a = 50e^4, b = c = 0, R = 0.1$). The amplitude of the input will be 0.1. Errors were maintained to a minimal level until at a frequency of 10 the controls required to maintain control were too large and the F_c constraint was violated.

Freq	error x_2
2	2
3	3
5	5
10	9
20	Infeasible

Feasibility
Recursive feasibility cannot be guaranteed in an absolute sense. If the Quadprog algorithm cannot find a solution N steps ahead that is feasible it will crash. In an industrial setting a crash or previous trigger point would cause an emergency stop to ensure safety. The Quadprog algorithm does not have the flexibility off implementing slack variables. Choosing $h = f_i/2$ or similar and setting $G = M_i$ you can implement a terminal set constraint such that $GX_N <= h$. Implementing this terminal set constraint results in the Quadprog program crashing before a single loop. In this state, barring major errors in the model, once it has commenced the control loop is guaranteed to be recursively feasible.

Stability
Stability requires Q and R to be positive definite. Hence why a very small but positive value of R was chosen in the frequency experiment.
Furthermore $A + BK$ must be Hurwitz and P must be chosen such that a stability ricatti equation is satisfied. In order to completely guarantee stability these equations need to be solved. It is important to note that the plant itself is stable and only a positive feedback loop could cause instability. Given that the controller is recursively feasible it will converge towards a terminal set. This is combination with the fact that their are constraints on the actuator and a robust model will ensure stability.

3 Summary

After much experimentation a result was finally reached that ensured recursive feasibility and excellent ride-comfort, or road-handling or minimum suspension travel. The effects of the controller are most obvious when only one goal is maximised. This is because there are physical limitations of the system that mean ride-comfort and road-handling cannot both be maximised.

4 MATLAB Instructions:

Get Model: Run "Get_Model_Script"
Run MPC with IMP (Best): Run "Run_MPC_Controller_IMP"
Run MPC without IMP (Poor Performance): Run "Run_MPC_Controller"