Ballbot robust stability and disturbance rejection analysis report

Antun Skuric

January 2020

Table of contents

Disturbance support of nonlinear ballbot model **2**
- Disturbance as non-potential force 2
- Additive measurement noise 3

Matlab toolbox **4**
- Simulation 4
- Robustness analysis 4

Robustness analysis **5**
- Linear model stabilisation using LQR controller 5
- Introducing the physical parameter uncertainties 7
  - Physical parameters 7
    - Uncertain parameters 7
  - Uncertainty analysis 8
  - Worst case parameters 9
  - System representation with uncertainties 11
- Disturbance rejection and noise attenuation 12
  - LQR controller disturbance rejection 13
  - H-infinity controller 16
    - System representation and scaling 16
    - Fixed gain H-infinity controller 18
      - LQR and H-infinity comparison on linear models 19
      - Linear vs nonlinear dynamics 22
    - State space H-infinity controller 24
      - LQR and H-infinity comparison on linear models 25

Analysis conclusion **30**
- **Future work** **30**

Literature **31**

Disturbance support of nonlinear ballbot model

In order to support disturbance rejection analysis of the ballbot system we will have to extend previously derived nonlinear model.

Disturbance as non-potential force

The disturbance introduced by a human interacting with the platform can be described as a torque applied to the center of mass of the body of the robot.

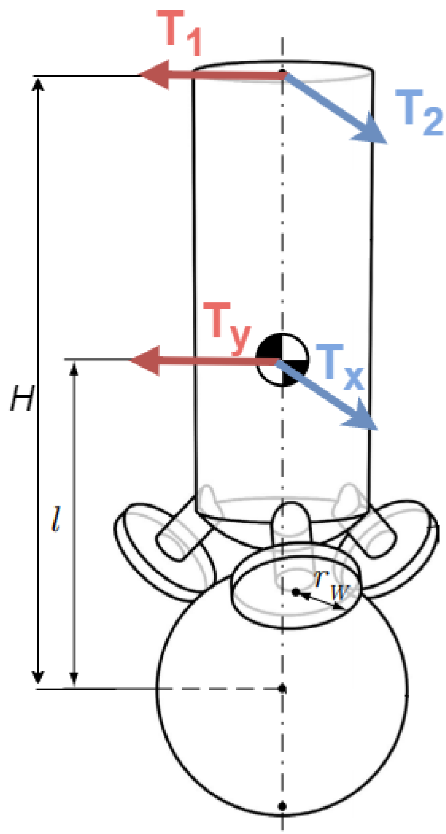

Figure 1.

The disturbance in the form of a push happens in one point on the robot body. The applied torque in any point of the robot body can be scaled to the center of the mass using the transformation:

$$[T_x \ T_y]^T = \frac{l}{H}[T_1 \ T_2]^T$$

But for the sake of this project we will consider as the disturbance torques are always applied to the center of the mass.

Furthermore torques T_x and T_y directly impact the angles second derivations of the angles ϑ_x and ϑ_y. Therefore we can write the non-potential force component of the disturbance as:

$$T_w = J_w[T_x \ T_y]^T = [T_x \ T_y \ 0 \ 0 \ 0]^T$$

And this disturbance non-potential force we add in the final Lagrange equation:

$$\frac{d}{dt}\left(\frac{\partial T}{\partial q'}\right)^T - \left(\frac{\partial T}{\partial q}\right)^T + \left(\frac{\partial V}{\partial q}\right)^T - f_{NP} - T_w = 0$$

And finally we will get the nonlinear model of the ballbot system where the variables are:

$$x = [\vartheta_x \ \vartheta_x' \ \vartheta_y \ \vartheta_y' \ \vartheta_z \ \vartheta_z' \ \phi_x \ \phi_y \ \phi_x' \ \phi_y']^T$$

$$u = [T_1 \ T_2 \ T_3]^T$$

$$w = [T_y \ T_y]^T$$

And finally the full nonlinear system will now include the disturbance input:

$$x' = f(x, u, w)$$

$$y = x$$

As well as the linearised version of the system:

$$x = Ax + B_w w + B_u u$$

$$y = Cx + D_w w + D_u u$$

Additive measurement noise

During the execution of this project we have noticed that measurement noise from the gyros and accelerometers significantly impacts the stability of the ballbot therefore we have included the measurement noise n in the robustness analysis, making the nonlinear system equations:

$$x' = f(x, u, w)$$

$$y = x + D_n n$$

As well as linearized version:

$$x = Ax + B_w w + B_u u$$

$$y = Cx + D_w w + D_n n + D_u u$$

Since accelerometers and gyros measure only signals ϑ_x, ϑ_x', ϑ_y and ϑ_y' the measurement noise n has only 4 components:

$$n = [n_1 \ n_2 \ n_3 \ n_4]^T \quad \text{and} \quad D_n = diag([\sigma_1 \ \sigma_2 \ \sigma_3 \ \sigma_4 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0])$$

Where σ represents the variance of the gaussian noise making the signals

$$|n_i| \leq 1$$

Matlab toolbox

For the purpose of this project the comprehensive simulation toolbox has been developed using Matlab software.

Simulation

The simulation part of the toolbox is mostly concentrated on:

- Parametric Ballbot model derivation
  - Nonlinear model with disturbances
  - Parametric linearization supporting any point $[x_o \ w_o]$
  - File: *model_script_disturbances.m*
- Initial condition response simulation
  - Nonlinear model
  - Linear model
  - File: *initialplot_compare_nlin.m*
- Closed loop simulation for nonlinear model with any controller structure
  - File: *nlsim.m*

Robustness analysis

The analysis part of the toolbox includes:

- LQR controller synthesis
- Introducing the physical parameter uncertainties into the system
- Finding the worst case parameters of the uncertain system
- Analysis of the worst case behavior of the closed loop system
  - Stability and performance
- H-Infinity controller synthesis for the disturbance rejection loopth
- Comparison of the behaviour of different controllers for linear and nonlinear models as well as nominal and worst case parameters
  - Disturbance rejection
  - Bode diagrams
  - Singular value plots

Robustness analysis

As the full nonlinear system proved itself to be overwhelmingly too complex for exact analysis the robustness analysis has been carried out for the linearised system model around the equilibrium at 0. But all the results for the linear systems have been simulated using the nonlinear system model as well and the full comparison has been provided.

Linear model stabilisation using LQR controller

First step in the robustness analysis is linearisation of the system equations around the equilibrium point $x_o = [0\,0\,0\,0\,0\,0\,0\,0\;\;0\,0]^T$ and $w_o = [0\,0]^T$.

$$x' = A(x_o, w_o)x + B_w(x_o, w_o)w + B_u(x_o, w_o)u$$
$$y = Cx + D_n n + D_w w + D_u u$$

Furthermore as a default controller for the closed loop system we use previously developed LQR controller with weight matrices:

$Q = diag([100,\ 50,\ 100,\ 50,\ 40,\ 20,\ 20,\ 10,\ 20,\ 10])$ and $R = diag([100,\ 100,\ 100])$

And control law:

$$u = K_{LQR}x$$

FInally the initial condition response of the linearised and nonlinear system with LQR controller has been carried out. The initial condition is:

$$x_{init} = [\tfrac{\pi}{20}\ 0\ \tfrac{\pi}{20}\ 0\ \tfrac{\pi}{10}\ 0\ 0\ 0\ 0\ 0]^T$$

Figures 2 and 3 show the system responses. The nonlinear system clearly has the *nonlinear* behaviours, especially for time response of the variable $(x6)\ \vartheta_z$, but the other variables perform relatively well for this initial condition. Even the control signals seem to match well.

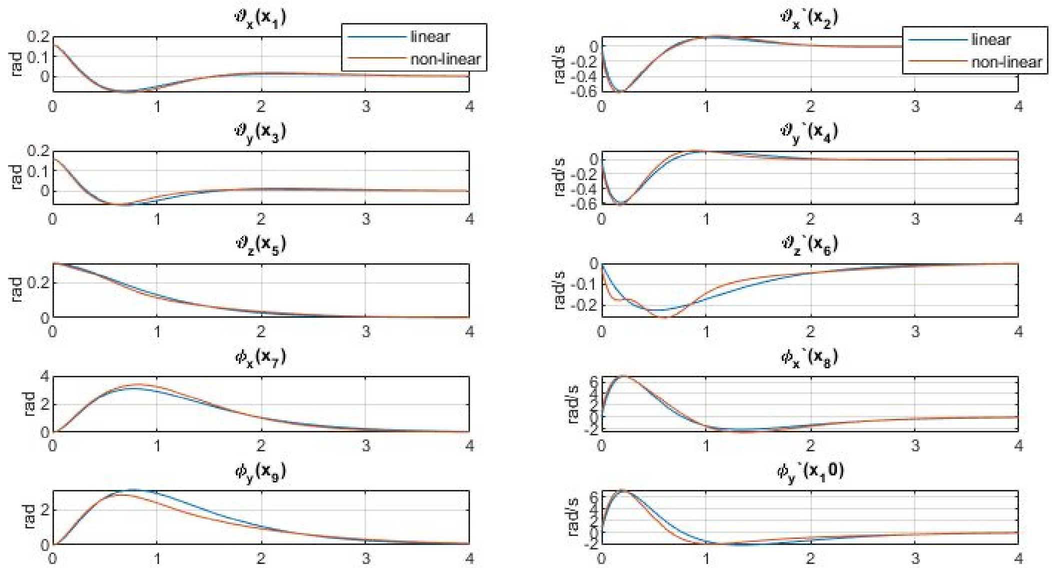

Figure 2. Nonlinear and linear model comparison with LQR controller - states

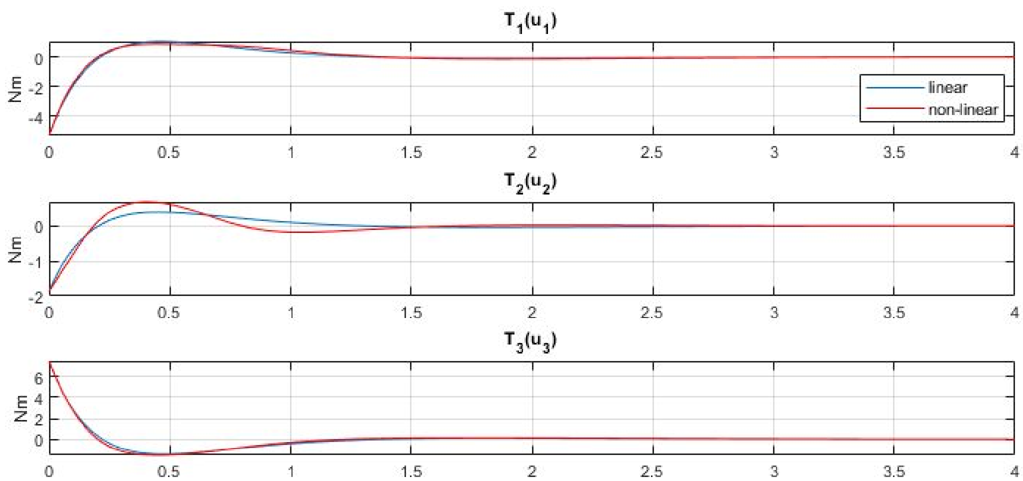

Figure 3. Nonlinear and linear model comparison with LQR controller - control signals

Introducing the physical parameter uncertainties

The uncertainties are introduced into the physical parameters of the system.Since some physical parameters of the model we cannot directly measure it makes sense to investigate the behaviour of the system model based on its uncertainty intervals.

Physical parameters

Physical parameters of the system that are easily measurable are:

- Radius of the ball
  $r_K = 0.120\ m$
- Radius of omniwheel
  $r_W = 0.05\ m$
- Radius of the body
  $r_A = 0.126\ m$
- Mass of the body with omniwheels
  $m_{AW} = 6.71\ kg$
- Mass of the ball
  $m_K = 0.625\ kg$

Uncertain parameters

The uncertainty of the variables is defined as the percentage of possible fluctuation around the nominal value.

- Distance between the centre of ball and centre of gravity of the body
  $l = 0.22634\ m \pm 20\%$
- Inertia of the body and Omni wheels in the body reference frame A
  ${}_A\Theta^x_{AW} = 1.41271413\ kgm^2 \pm 20\%$
  ${}_A\Theta^y_{AW} = 1.41271311\ kgm^2 \pm 20\%$
  ${}_A\Theta^z_{AW} = 0.05359646\ kgm^2 \pm 20\%$
- Inertia of the Ball
  $\Theta_{Ki} = 0.003606375\ kgm^2 \pm 20\%$
- Inertia of the omniwheel
  $\Theta_{Wi} = 0.01504\ kgm^2 \pm 20\%$

Uncertainty analysis

The physical parameters of the system model directly influence its dynamical behaviour. That can be easily seen if we look at the figures 4 and 5. From figure 4 it is clear that based on the different parameter values the initial response of the system can have much higher overshoot and have higher degree of oscillations.

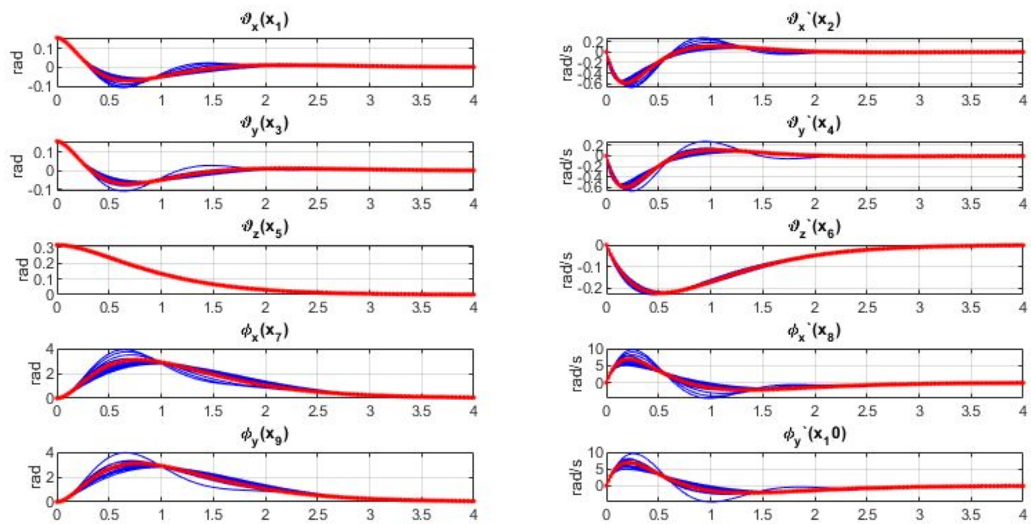

Figure 4. Initial condition response of LQR closed loop linear model with parameter uncertainty

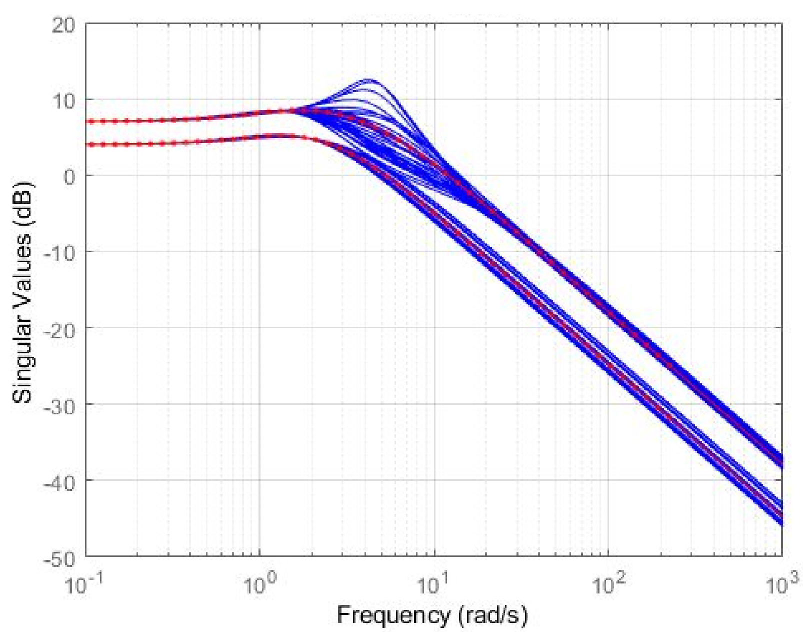

Figure 5. Singular value plot of LQR closed loop linear model with parameter uncertainty

Furthermore from the figure 6 we can see that the singular value plot varies highly as well. For the nominal parameters the singular value peak is at 8dB~6.3 and for some cases of uncertain parameters if rises all the way to 12dB~15 what is more than two times.

Worst case parameters

The worst case parameters of the system are the parameters which cause the lowest stability margin of the system. For this analysis we have used disk margin stability [1] for MIMO systems. What we were searching for was the minimal stability margin of all the input-output transfer functions of the MIMO system. With this in mind we have created a constrained optimisation problem, where optimisation variables were uncertain parameters of the system and the cost function was the stability margin of the system, where we were trying to reach the minimum.
The problem proved itself to have a stable behaviour and converged relatively fast.

The worst case parameters for the earlier described uncertain parameters is:
$l = 0.1811\ m$, ${}_A\Theta^x_{AW} = 1.6953\ kgm^2$, ${}_A\Theta^y_{AW} = 1.2670\ kgm^2$, ${}_A\Theta^z_{AW} = 0.0511\ kgm^2$, $\Theta_{Ki} = 0.0034\ kgm^2$, $\Theta_{Wi} = 0.0180\ kgm^2$ with disk stability margin equal to $DM_{wc} = 0.0331$ and phase margin $PM_{wc} = 1.89^o$.
To compare with the nominal model has the disk margin of $DM_n = 0.1579$ and phase margin of $PM_n = 9.02^o$.

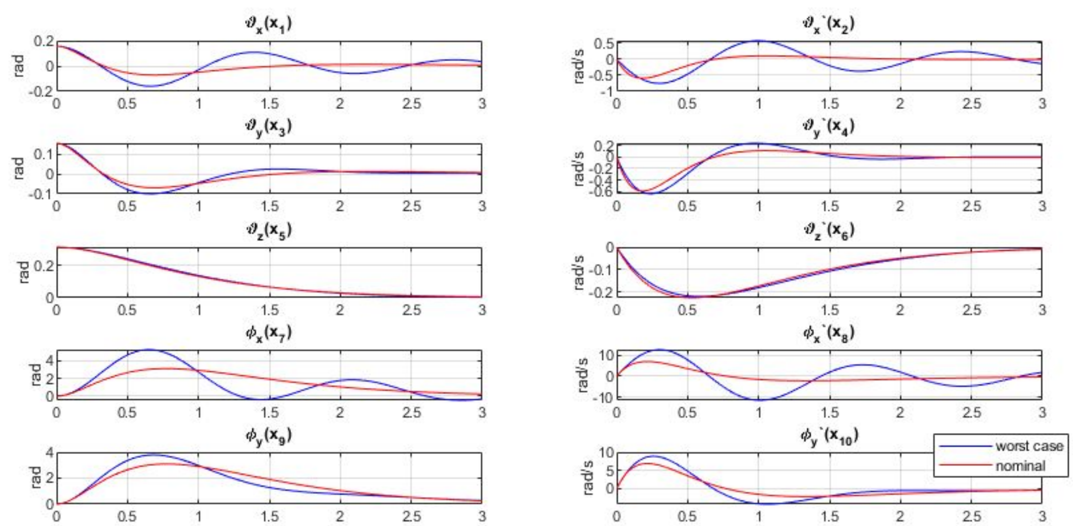

Figure 6. Initial condition response comparison of nominal and worst case parameters

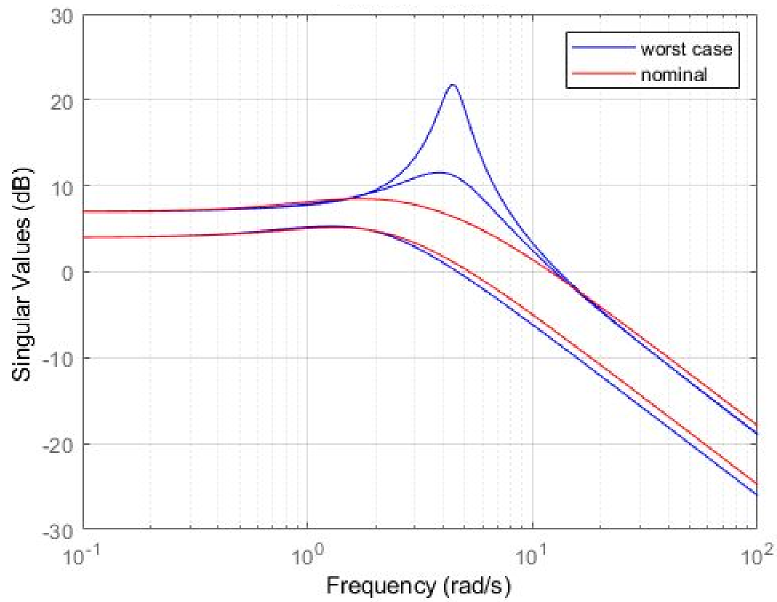

FIgure 7. Singular value plot comparison of nominal and worst case parameters

It is worst noticing that the worst case parameters make sense in physical terms, the worst case center of mass distance is the smallest possible within the limits and the moment of inertia of the motor with the omniwheel is the highest possible within the uncertainty limits.
The lower center of mass distance directly increases the natural frequency of the system making it much faster and on the other hand higher moment of inertia of the motor with omniwheel makes the actuators much slower which. Therefore with faster system and slower actuators the system becomes more unstable, causing additional oscillations of the system, what can be seen on the figure 6. Figure 7 confirms the observations from the figure 5 and shows that the closed loop system has considerably higher peak in singular value plot than nominal parameter version making it much less robustly stable.

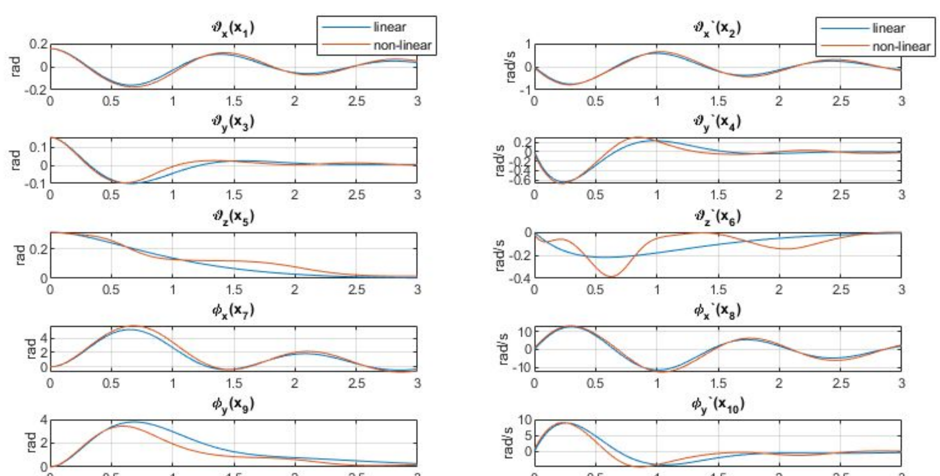

FIgure 8. Comparison of linear and nonlinear model with worst case parameters

Finally the simulation of the simulation of the nonlinear and linear system with worst case parameters has been carried out with still with the default LQR controller closed loop and the same initial condition. It can be seen that the nonlinear system is still stable but shows reasonably higher degree of nonlinearity especially for the time response of the signal $(x6)\ \vartheta_z$.

System representation with uncertainties

Our system equation for this uncertain system now look like:

$$x' = (A + \delta A \Delta)\, x + (B_w + \delta B_w \Delta) w + (B_u + \delta B_u \Delta) u$$
$$\delta A = (A_{WC} - A), \qquad \delta B_w = (B_{WCw} - B_w), \qquad \delta B_u = (B_{WCu} - B_u), \qquad |\Delta| < 1$$

Therefore we can represent our system as:

$$x' = A\, x + [\, B_w \;\; B_u][w\; u]^T + \delta A\, \Delta x + [\delta B_w \;\; \delta B_u]\Delta[w\; u]^T$$

Since we have identified the worst case dynamics of the system based on the physical parameter uncertainty ranges the robust stability can be guaranteed only if the controller stabilises both linear and nonlinear model with both nominal and worst case parameters, if it stabilises the system for any Δ. But to produce a comprehensive result we proceed to the disturbance rejection part of the project.

Disturbance rejection and noise attenuation

In order to test thoroughly and uniformly all the systems the testing signal has been constructed.

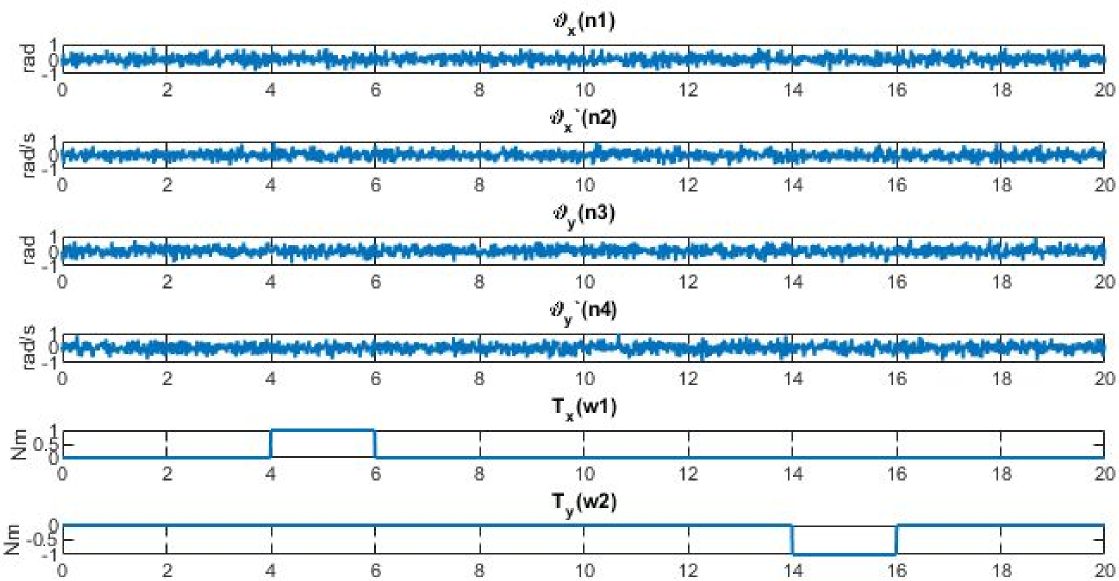

Figure 10. Disturbance test signal

The test signal comprises of the sensory noise signal on all four accelerometer measurements ϑ_x, ϑ_x', ϑ_y, and ϑ_y'. Two disturbance torques T_x and T_y are applied in 4th and 14th second of the simulation with different signs each with a duration of two seconds. The goal of this testing signal is to evaluate the measurement noise attenuation of the controller and of course the performance of disturbance force compensation.

LQR controller disturbance rejection

To test the robustness of the initial LQR controller we have conducted a series of tests and simulations comparing the behaviour of the disturbance rejection loops of the linear and nonlinear system with nominal and worst case parameters.

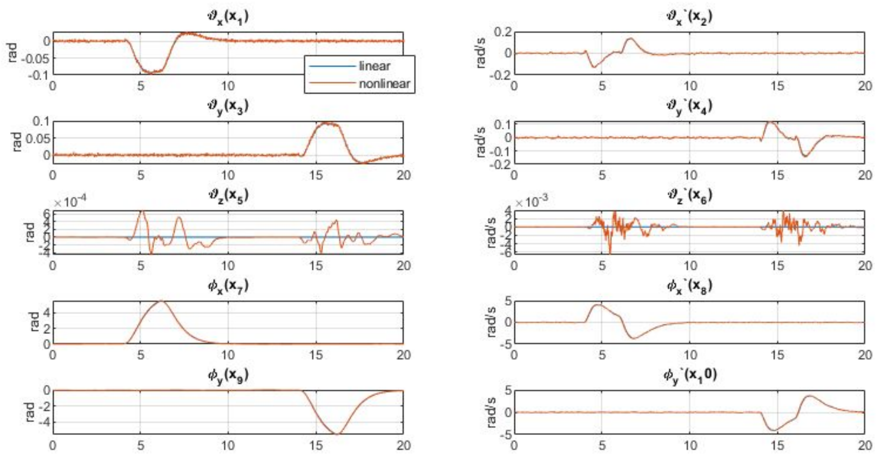

Figure 11. Disturbance rejection linear vs nonlinear response for nominal parameters Outputs $y(t)$

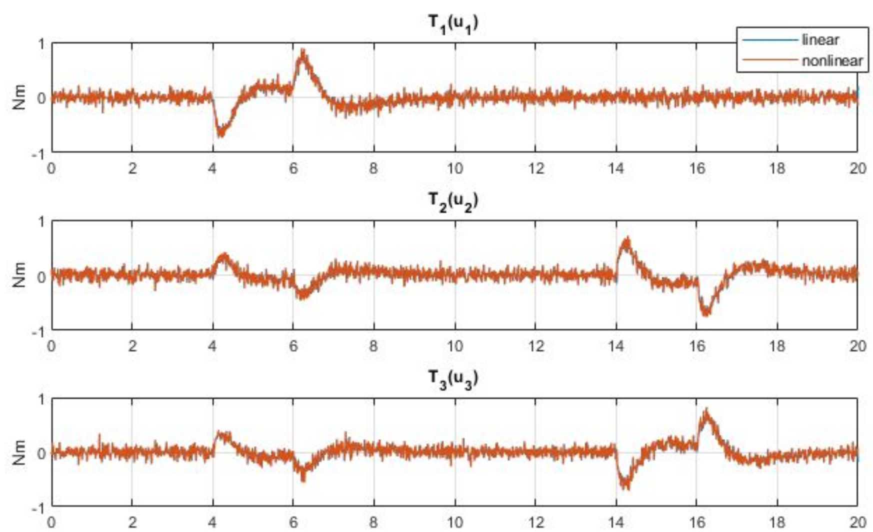

FIgure 12. Disturbance rejection linear vs nonlinear response for nominal parameters control signal $u(t)$

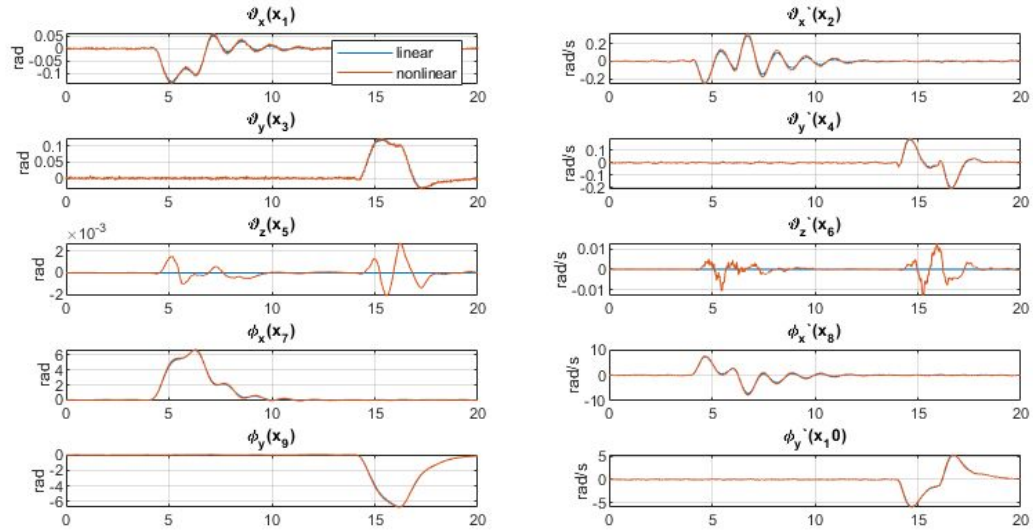

Figure 13. Disturbance rejection linear vs nonlinear response for worst case parameters outputs $y(t)$

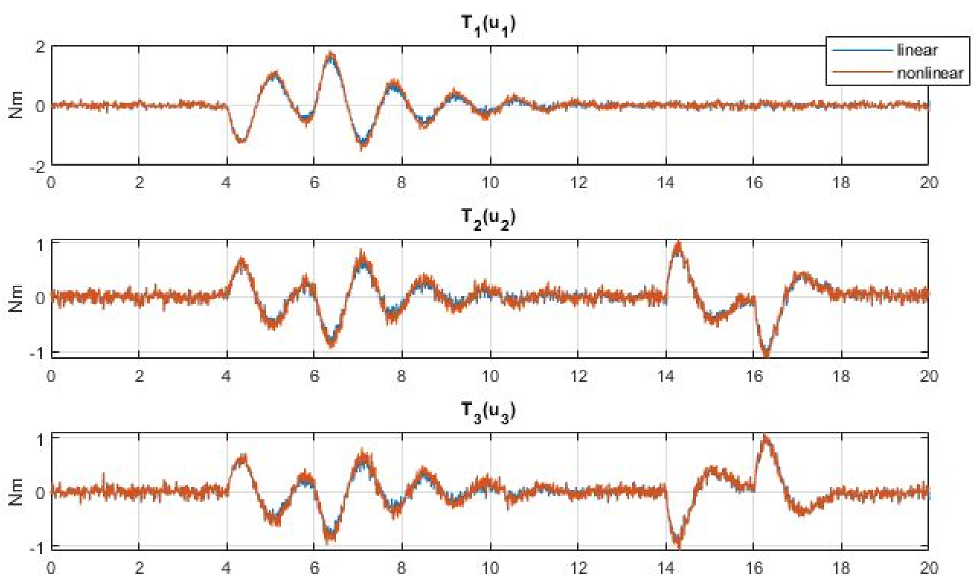

Figure 14. Disturbance rejection linear vs nonlinear response for nominal parameters control signal $u(t)$

The figures 11 to 14 show the comparison of the state variables and input signals generated for both linear and nonlinear models and for both nominal and worst case parameters. The overall conclusion of the analysis is that LQR controller is robustly stable for the previously defined uncertainties and that it performs stable disturbance rejection. Even though it can be seen that

the nonlinear model performance is visibly oscillatory, the system has proved itself to be stable for the testing signal presented above.
Furthermore it can be seen that the worst case performance of both linear and nonlinear model is visibly impared and that the oscillations in the nonlinear model are much higher what is clearly visible for the control input signals on figures 12 and 14.

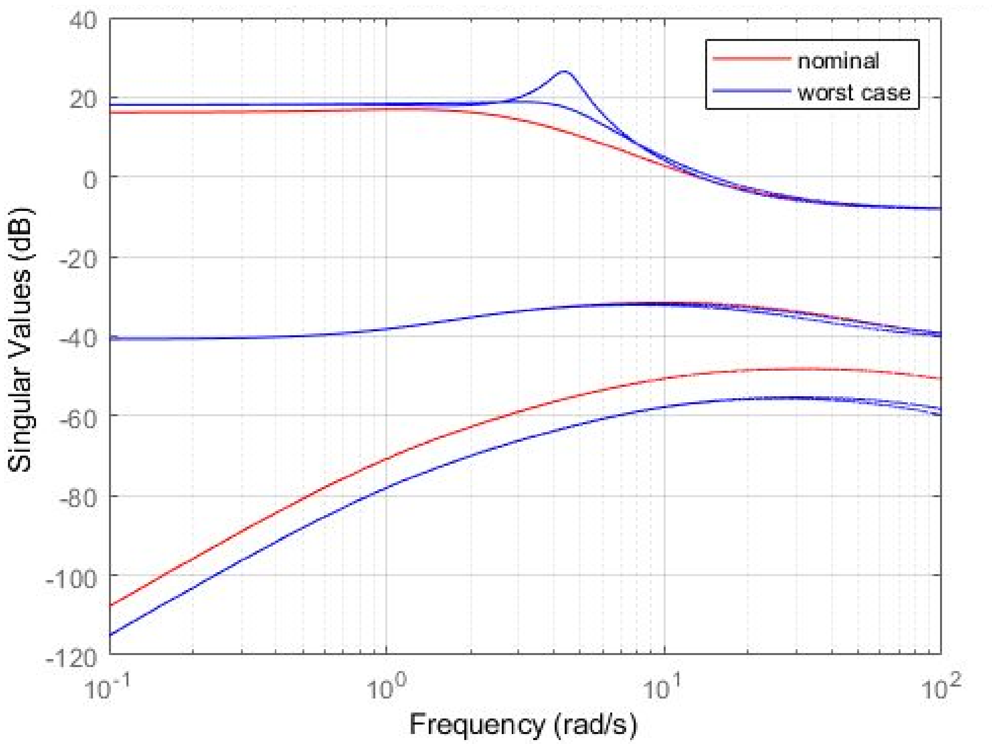

Figure 15. Singular value plot of the disturbance rejection loop based on LQR controller for nominal and worst case parameters

In order to guarantee robust stability [2] of the system based H_∞ norm of the closed loop system should be smaller than 1. But in this case both norms for nominal and worst case linear system with LQR controller are higher than 1.

$$H_\infty(nominal) = \|CL\| = 6.96$$
$$H_\infty(worst\ case) = \|M\Delta_{wc}\| = 20.77$$

This means that the system is not robustly stable according to [2]. And due to the high value of the H_∞ norms we cannot use small gain theorem[2] approach to stabilise this system.

H-infinity controller

Next step in the robustness analysis was to design the robust H-infinity controller for this uncertain system.

System representation and scaling

The disturbance rejection linear system with measurement noise has equations as follows:

$$x' = Ax + B_w w + B_u u$$

$$y = Cx + D_n n + D_w w + D_u u$$

where signal $n \in R^4$, $w \in R^2$ and $u \in R^3$.

Furthermore in order to simplify the analysis our linear system will be transformed intro the linear fractional transformation supportive format:

$$\begin{bmatrix} z \\ y \end{bmatrix} = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix} \begin{bmatrix} \omega \\ u \end{bmatrix}$$

Where y is system measurement signal vector, u is the control signal vector, z is the signal that we are interested in minimising and w is our disturbance vector, in this case containing: $\omega = [n\ w]^T$

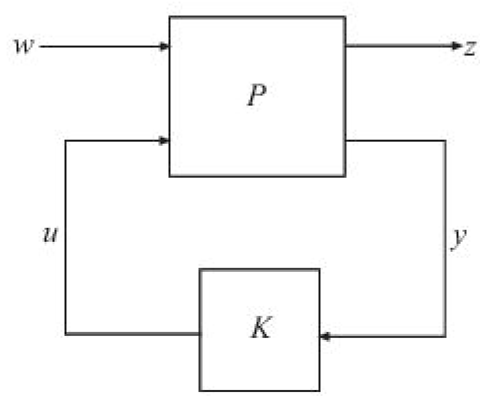

Figure 9. Linear fractional transformation

Finally we can define the full state space model of the system to be:

$$x = Ax + B_w w + B_u u$$

$$z = C^z x + D_n n + D_w w + D_u^z u$$

$$y = Cx + D_n n + D_w w + D_u u$$

$$u = Kx$$

Where z is the evaluated output of the system used to optimise the overall system performance, signal to minimise H_∞ norm.

For MIMO (Multiple input - multiple output) systems since there are multiple transfer functions in between inputs and outputs H_∞ norm tells us only the worst case gain of the system of one transfer function, but does not keep track of the others. Therefore in order to minimise H_∞ norm of all transfer functions at once the system outputs need to be scaled. There are many ways to scale the system values [3], in this project outputs are scaled as follows:

$$z_i = 1/\|y_i\|_\infty, \quad z_i \in [1,\ 100]$$

The outputs are scaled to all have the H_∞ norm equal to 1, this helps to equally distribute the minimisation influence. The reason we have the lower limit (1) of the values of z_i is because we want the H_∞ norm of all signals to be lower than 1. Upper limit of the z_i restricts low gain control loops on optimisation algorithm.

Finally, since we are optimising H_∞ norms of the closed loop transfer function of disturbance rejection the scaling approach can we viewed as Loop-shaping algorithm with constant shape functions with magnitude in between $[10^{-2},\ 1]$.

Additionally, for the purpose of H-infinity controller as a system output z it is necessary to include the control signals u, which will be scaled as well. Therefore our final optimisation system output becomes:

$$z = C^z x + D_n n + D_w w + D_u^z u,$$

Or:

$$z = \begin{bmatrix} W_x C & 0 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ W_u \end{bmatrix} u + D_n n + D_w w$$

Where

$$W_x = diag([9.96 \quad 4.95 \quad 9.96 \quad 4.95 \quad 10 \quad 100 \quad 1 \quad 1 \quad 1 \quad 1]),$$
$$W_u = diag([1 \quad 1 \quad 1]).$$

For the purpose of this project we have designed the fixed (gain) H-infinity controller as well as full state space controller calculated for the worst case parameters linearised system, $\Delta = 1$.

Fixed gain H-infinity controller

To calculate the new H-infinity controller $u = K x,\ K \in R^{3x10}$, we have used Matlab built in function *hinfstruct* which was able to find a stabilising controller for our system with final H_∞ norm for system output z and $\Delta = 1$ equal:

$$||z||_\infty = 6.17$$

To compare with norm for the LQR case:

$$||z||_\infty = 21.42$$

Furthermore it is interesting to see the comparison of the singular value plot :

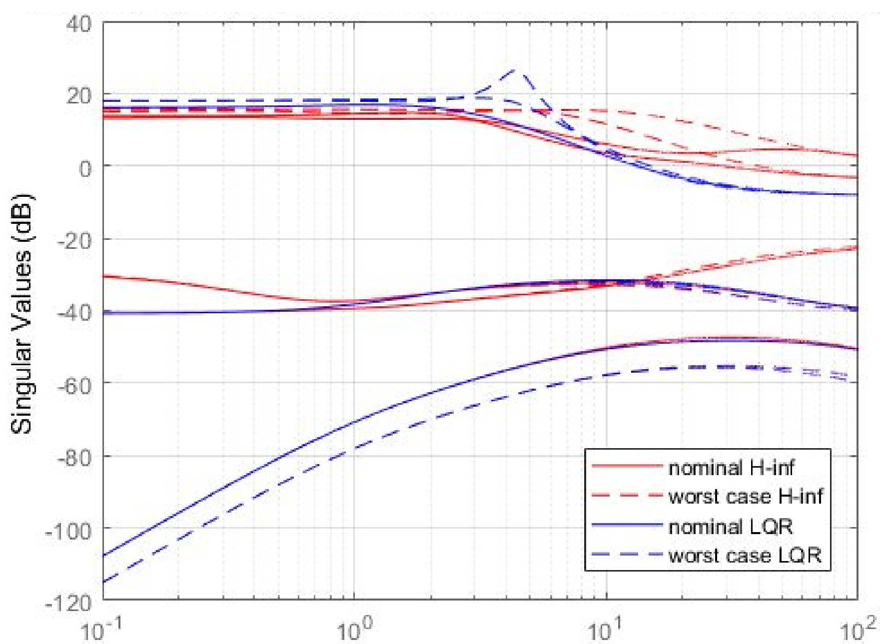

Figure 16. Singular value plot of the LQR and H-infinity controller base for nominal and worst case parameters

Th figure 16 shows that the new H-infinity controller was not able to reduce the H_∞ norm of the system underneath the $0dB$ value, but the reduction in the peak singular value is significant:

$$H_\infty(nominal) = ||CL|| = 5.44$$
$$H_\infty(worst\ case) = ||M\Delta_{wc}|| = 6.09$$

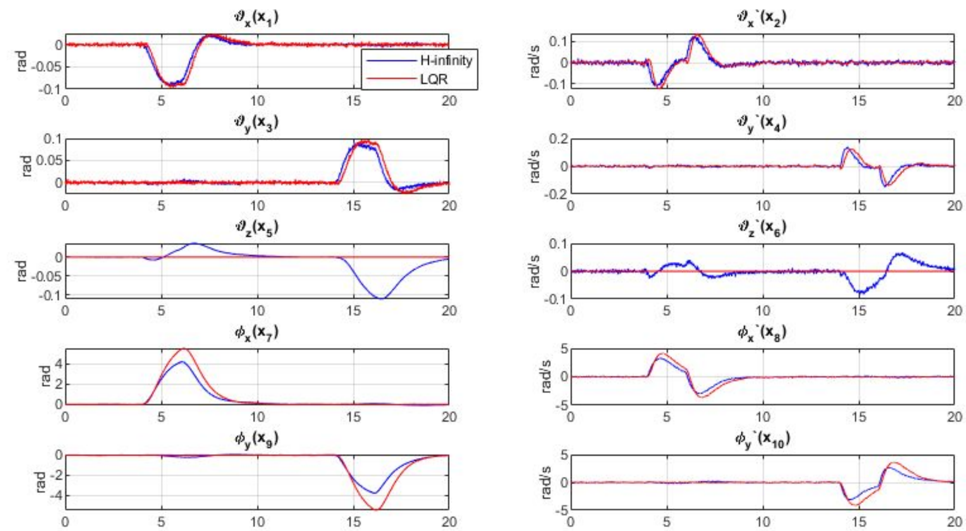

Figure 17. H-infinity and LQR controller comparison for linear nominal models - outputs y

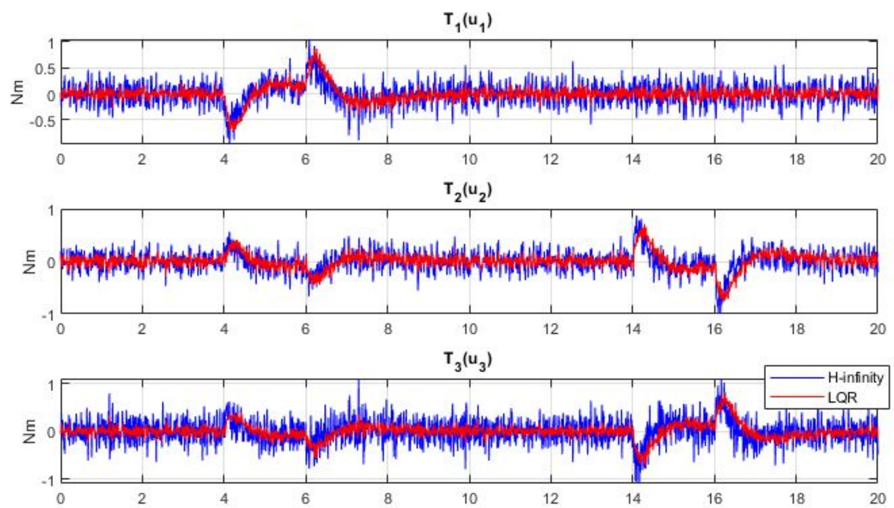

Figure 18. H-infinity and LQR controller comparison for linear nominal models
control signals u

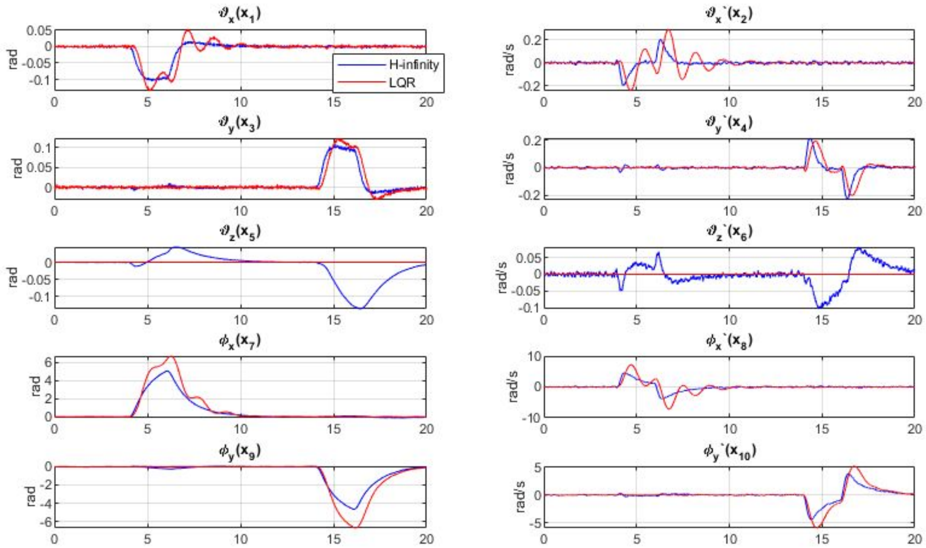

Figure 19. H-infinity and LQR controller comparison for linear worst case models - outputs y

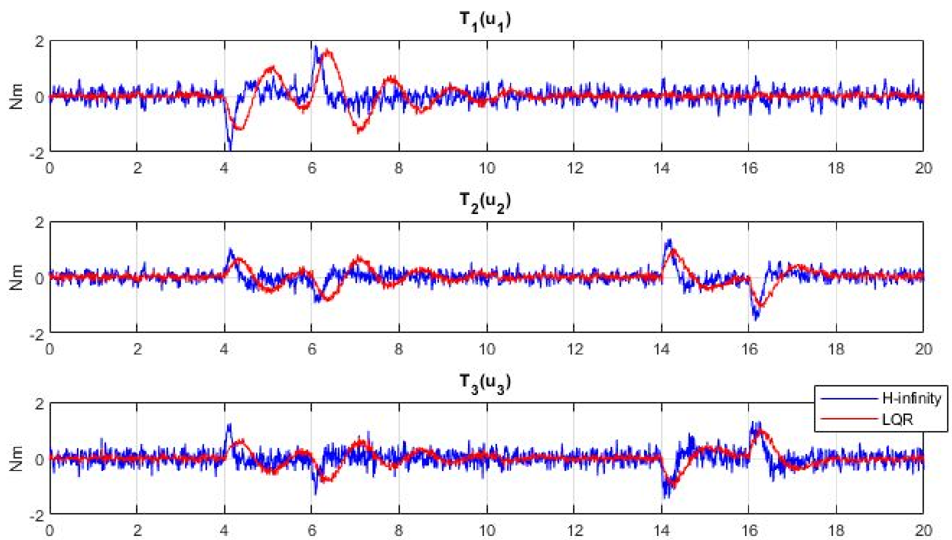

Figure 20. H-infinity and LQR controller comparison for linear worst case models control signals u

Figures 17 to 20 show the comparison of the disturbance rejection control loops based on LQR and H-infinity controller. It can be seen that H-infinity controller is much more consistent for both nominal and worst case parameters not showing any significant changes in the performance. LQR controller has visibly higher overshoots and high oscillations when the disturbance torques T_x and T_y are applied.

Furthermore the disturbance rejection loop of the xy axis rotation of the robot ϑ_z seem to have suffered higher impact when loop is closed with H-infinity controller. This is one of the tradeoffs of the MIMO .systems H-infinity synthesis, the controller is designed to minimise maximal H_∞ norm value, not for all the transfer functions separately. Therefore sometimes some of the control loops can have impareed performance. As for the case of this H-infinity controller, this can be easily seen on the figure 17 and 19.
Finally it can it can be seen that H-infinity controller introduces reasonably higher dynamics into the system, especially for the control signals as shown on figures 18 and 20.

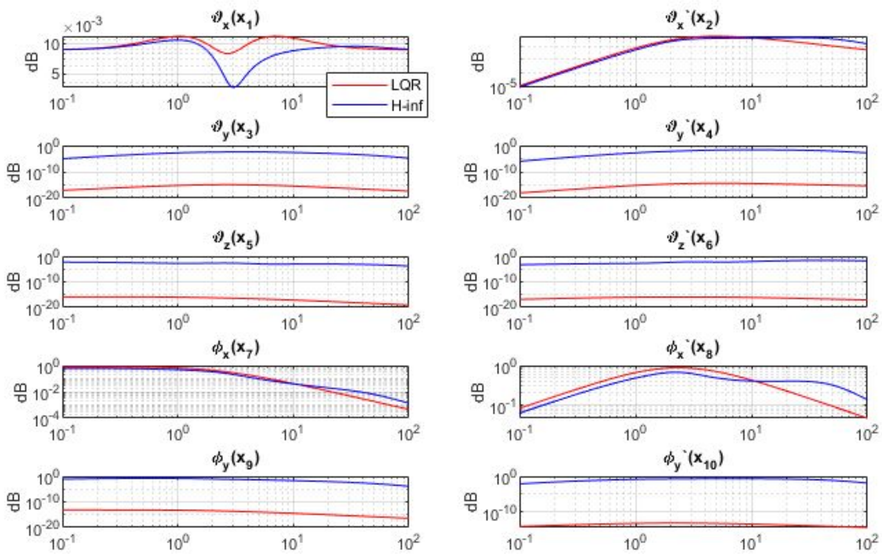

Figure 21. Bode graph of the disturbance rejection closed loop with nominal parameters from the input n_1 to outputs y

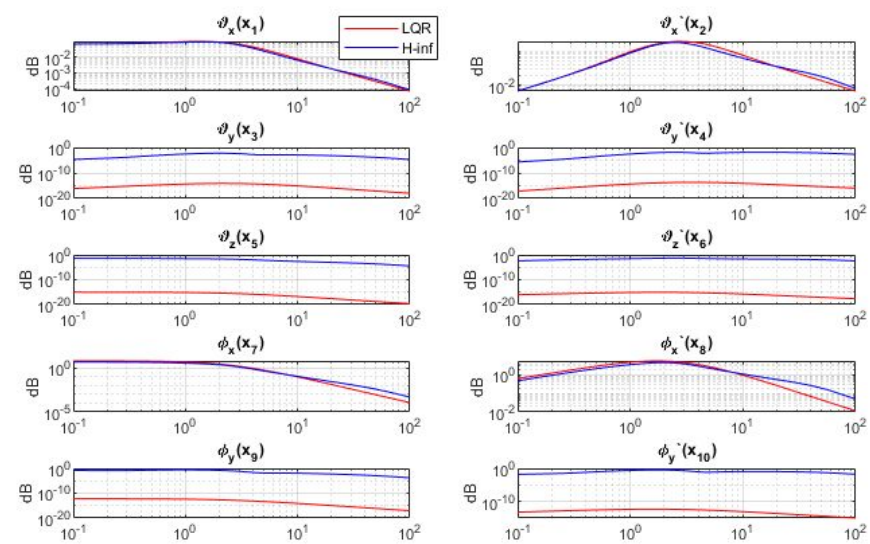

Figure 22. Bode graph of the disturbance rejection closed loop with nominal parameters from the input T_x to outputs y

On the figures 21 and 22 frequency domain comparison of the H-infinity and LQR closed loop algorithms is shown. On these plots we can better see why the disturbance rejection for the signal ϑ_z seems to be worse for the H-infinity controller even though stability margin is better and all the other output variables have better performance. Bode diagram shows that the H-infinity controller raised the gain for this control loop by a large amount. Both for noise attenuation, figure 21, and disturbance rejection figure 22. The reason why this happened is because even though the gain has been significantly raised it still does not influence overall H_∞ norm of the system, it is still very far from the maximal values.

Linear vs nonlinear dynamics

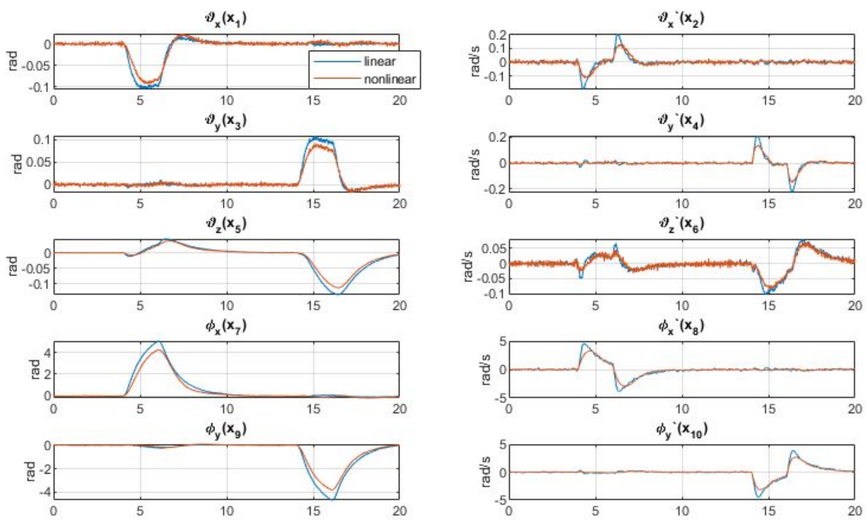

Figure 23. Linear and nonlinear H-infinity worst case models outputs y

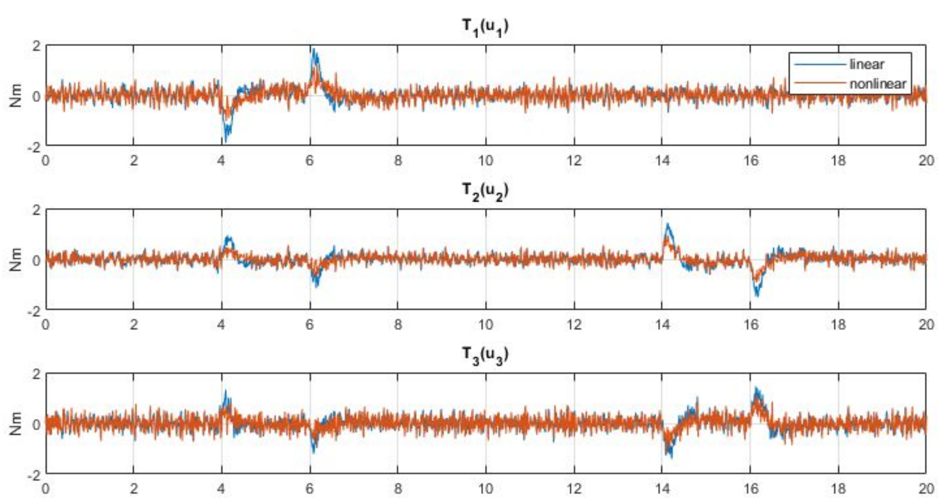

Figure 24. Linear and nonlinear H-infinity worst case models control signals u

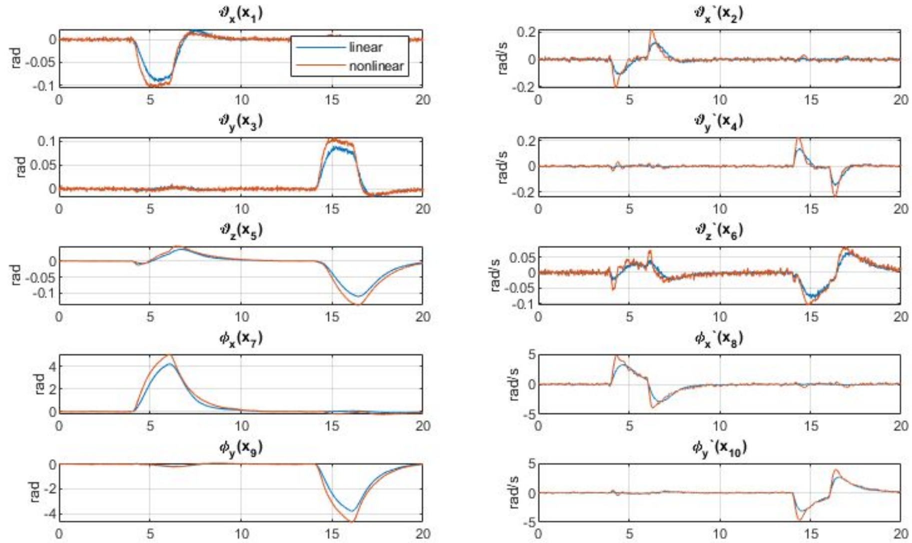

Figure 25. Linear and nonlinear H-infinity nominal models outputs y

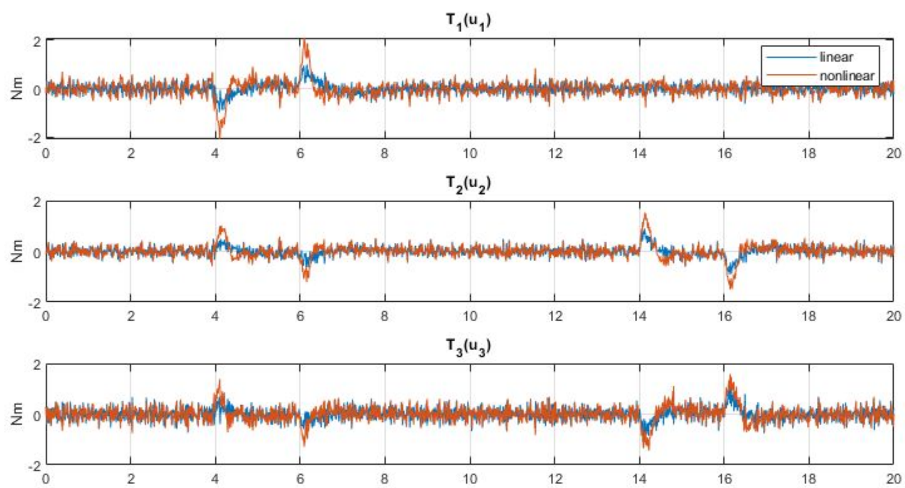

Figure 26. Linear and nonlinear H-infinity nominal models control signals u

Figures 23 to 26 show the comparison of the disturbance rejection control loop using the H-infinity controller for both nominal and worst case parameters. The same conclusion as for the linear models can be drawn in this case. The H-infinity controller proves to be very robust and does not show any significant impairment of performance due to the change of parameters of the system for both the linear and nonlinear case.

State space H-infinity controller

H-infinity analysis gives a powerful tool to design and analyse full state space controller architecture, which is one of it main benefits in comparison to LQR algorithms.
New controller structure is:

$$x_k' = A_k x_k + B_k u_k$$
$$y_k = C_k x_k + D_k u_k,$$

where:

$$u_k = y, \qquad y_k = u.$$

Therefore:

$$A_k \in R^{NxN}, \quad B_k \in R^{Nx10}, \quad C_k \in R^{3xN}, \quad D_k \in R^{3x10}$$

Where N represents the number of states of the controller. In this project we have chosen $N = 3$ due to the.
The values of matrices A_k, B_k, C_k and D_k are optimised using the same Matlab function *hinfstruct* which was able to find a stabilising controller for our system with final H_∞ norm for system output z and $\Delta = 1$ equal:

$$||z||_\infty = 5.32$$

To compare with norm for the LQR case:

$$||z||_\infty = 21.42$$

Furthermore, it is interesting to see the comparison of the singular value plot :

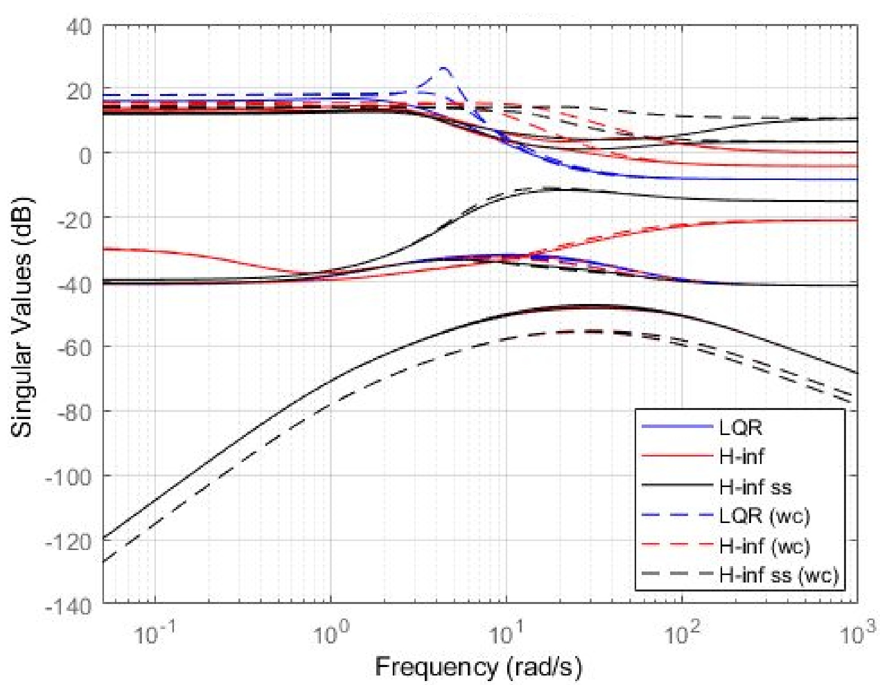

Figure 27. Singular value plot of the LQR and H-infinity controllers for nominal and worst case parameters

The figure 27 shows that the new state space H-infinity controller has further reduced the H_∞ norm of the system making it:

$$H_\infty(nominal) \ = \ \|CL\| \ = \ 4.76$$
$$H_\infty(worst\ case) \ = \|M\Delta_{wc}\| \ = \ 5.25$$

LQR and H-infinity comparison on linear models

Finally in order to compare the dynamics of all three controllers we simulate the test disturbance rejection of the linear system with nominal and worst case parameters.

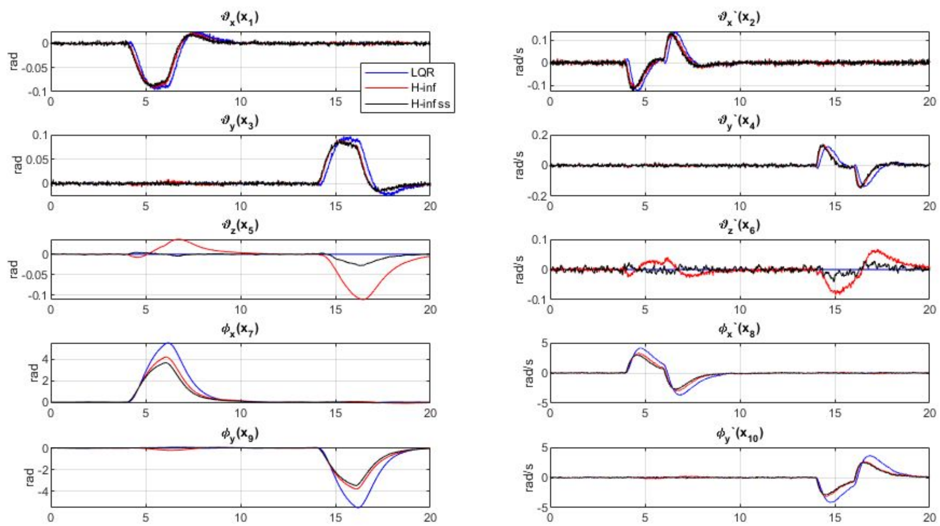

Figure 28. H-infinity and LQR controllers comparison for linear nominal models - outputs y

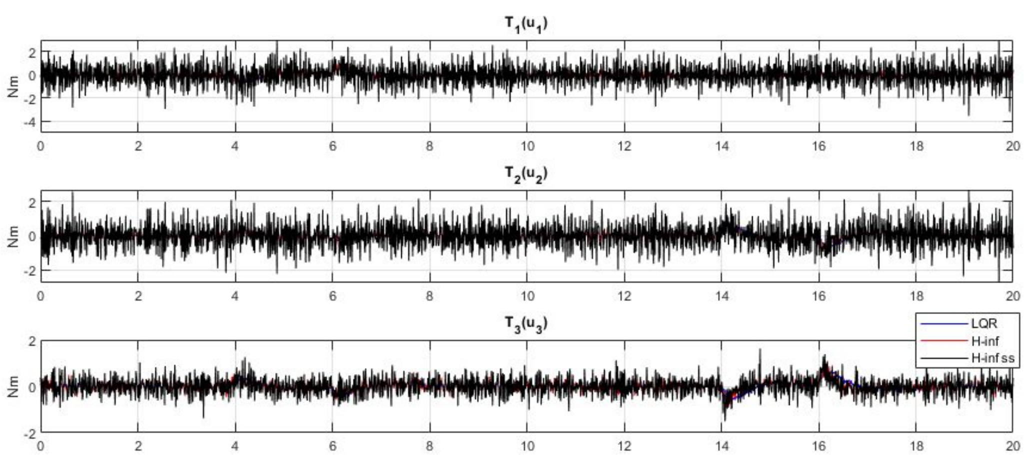

Figure 29. H-infinity and LQR controllers comparison for linearnominal models control signals u

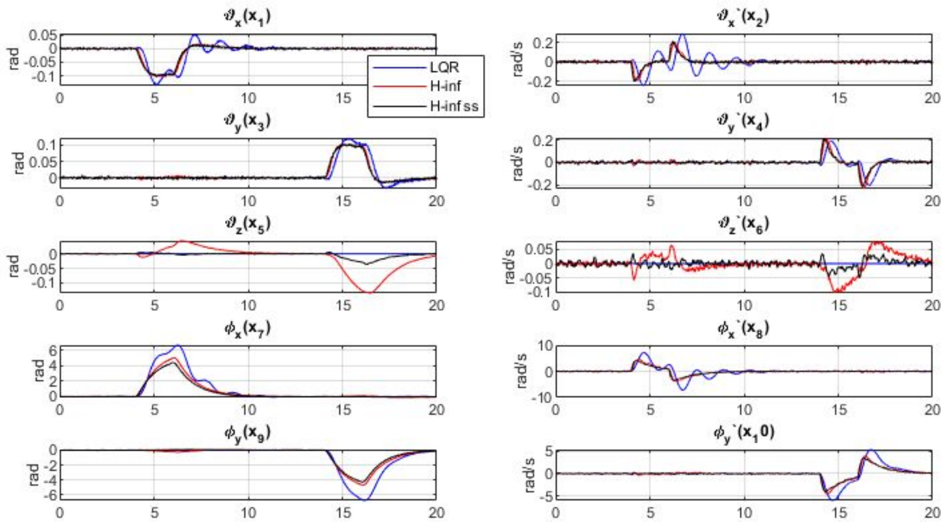

Figure 30. H-infinity and LQR controllers comparison for linear worst case models - outputs y

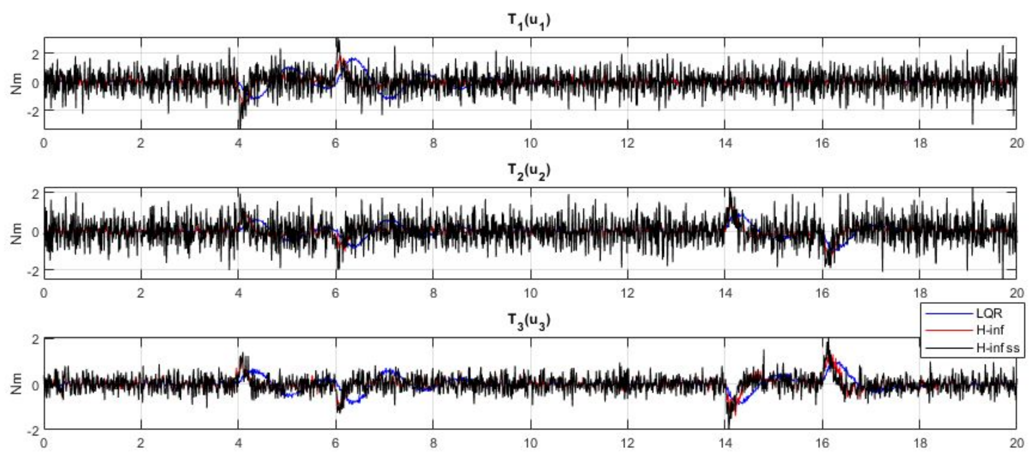

Figure 31. H-infinity and LQR controllers comparison for linear worst case models control signals u

Figures 28 and 31 show the comparison of the disturbance rejection control loops based on LQR and fixed gain and full state space H-infinity controller.

It can be seen that the closed loop system with H-infinity controllers have more consistent dynamical behaviour for nominal and worst case parameters not showing any significant changes in the performance. LQR controller has visibly higher overshoots and high oscillations when the disturbance torques T_x and T_y are applied for the worst case parameters.

Furthermore the disturbance rejection loop of the xy axis rotation of the robot ϑ_z has been greatly reduced for the case of state space H-infinity controller in comparison with fixed gain H-infinity controller.

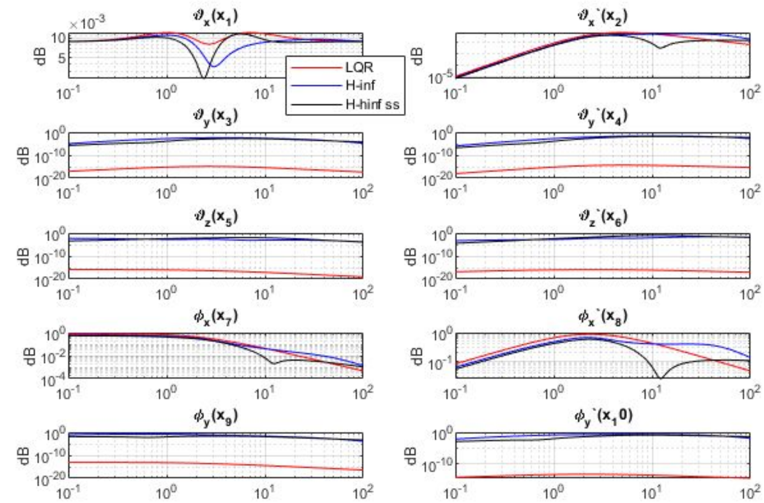

Figure 32. Bode graph of the disturbance rejection closed loop with nominal parameters from the input n_1 to outputs y for all three controllers

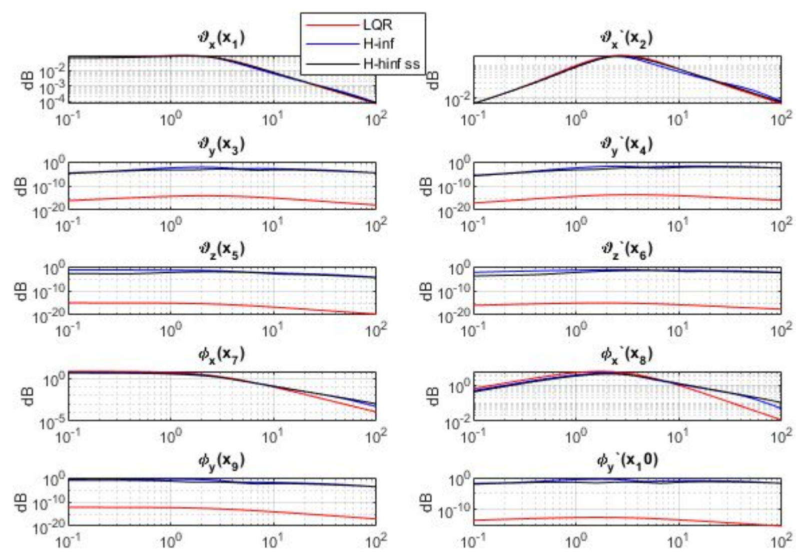

Figure 33. Bode graph of the disturbance rejection closed loop with nominal parameters from the input T_x to outputs y for all three controllers

On the figures 32 and 33 frequency domain comparison of both state space and fixed gain H-infinity and LQR closed loop algorithms is shown. On these plots we can better see why the disturbance rejection for the signal ϑ_z, for the case of state space H-infinity, has lower gain on lower frequencies for disturbance T than the fixed gain version, which explains the better rejection.

Linear vs nonlinear dynamics

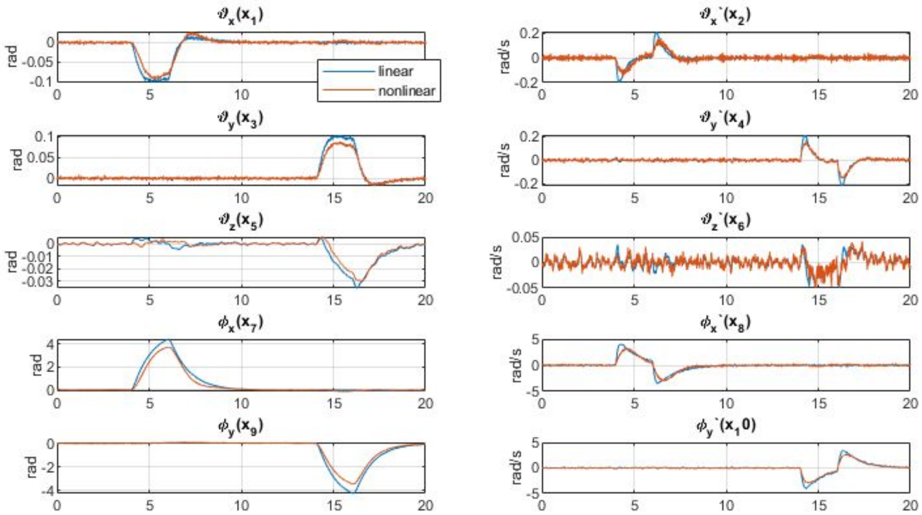

Figure 34. Linear and nonlinear H-infinity worst case models outputs y

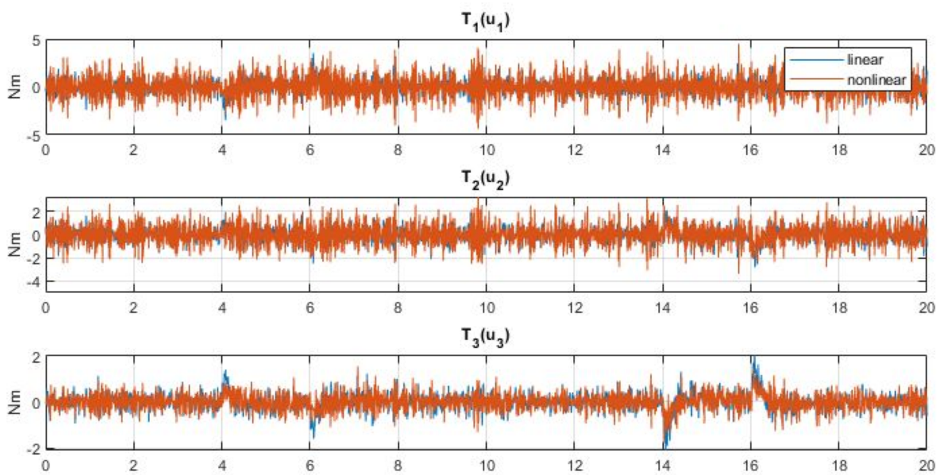

Figure 35. Linear and nonlinear H-infinity worst case models control signals u

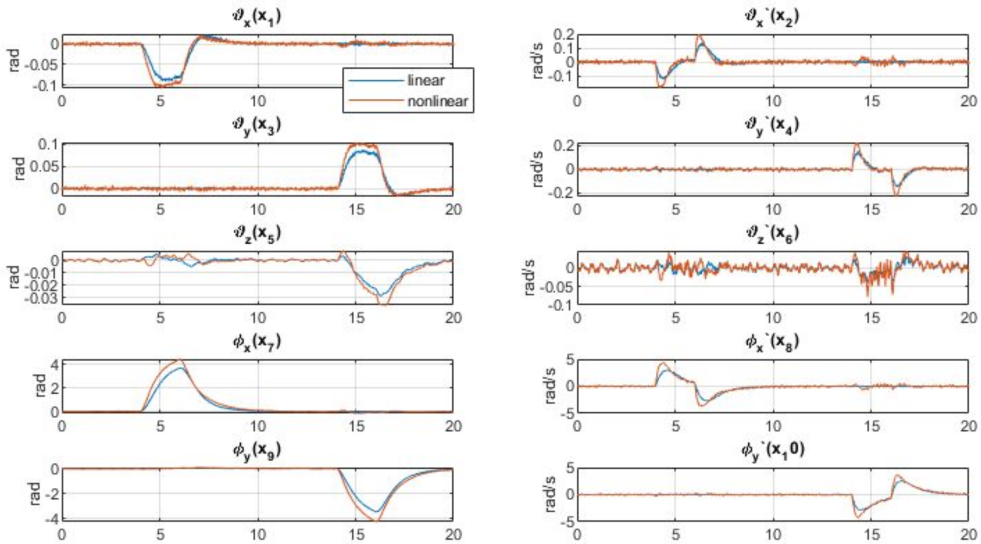

Figure 36. Linear and nonlinear H-infinity nominal models outputs y

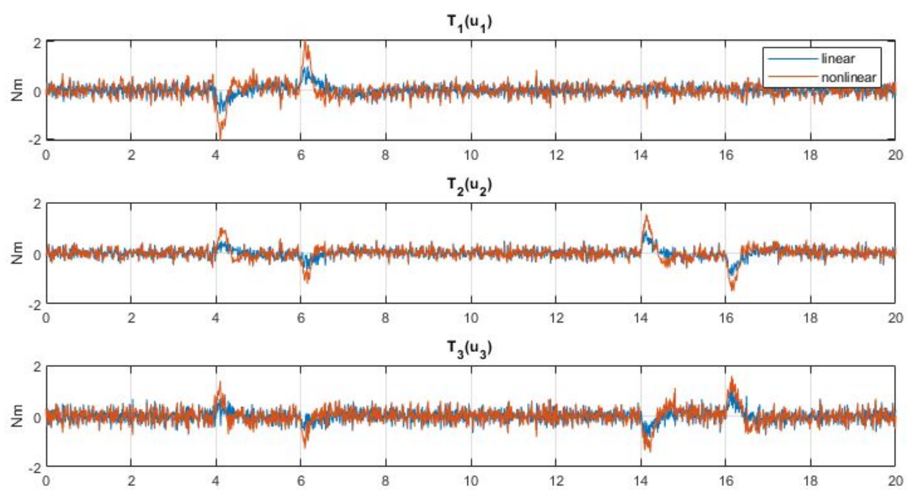

Figure 37. Linear and nonlinear H-infinity nominal models control signals u

Figures 34 to 37 show the comparison of the disturbance rejection control loop using the full state space H-infinity controller for both nominal and worst case parameters. The same conclusion as for the linear models can be drawn in this case. The H-infinity controller proves to be robust and stays in the proximity of the linearization point therefore the dynamical behavior of the system is almost identical in nonlinear and linear case.

The difference in the behaviour can be seen in terms of controller signals, they are much noisier in the case of nonlinear system, especially in the nominal case. What can produce instability in extreme conditions.

Analysis conclusion

In this project robustness analysis was performed for Ballbot nonlinear control system. Three different controllers have been designed and their disturbance rejection and robustness to parameter uncertainty have been analysed:

- Linear Quadratic Regulator (LQR)
- Fixed gain H-infinity controller
- Full state space H-infinity controller

The results have shown that the H-infinity controller handle the uncertainty of system parameters with higher degree of consistency. The analysis has shown that H-infinity controllers handle 20% of parameter uncertainty with much more consistent performance than LQR controller, which comes to the limit of stability in this case. Stability indicators of three systems are shown in table 1.

Furthermore in the analysis it has been shown that full state space H-infinity controller with just 3 states improves significantly the disturbance rejection of the system. The result of analysis implies that the optimisation algorithm searching for the controllers which minimising the H_∞ norm of the system can find more optimal solution with more degrees of freedom, which is a logical finding. But this minimisation of the H-infinity norm using the state space controller has proven to be a problem in the case of this nonlinear system because of high controller gain the system easily jumps out the *linear* state area and becomes unstable.

Table 1. Stability margin comparison for all three controllers - the best values are shown in bold

Stability criteria	Phase margin		Gain Margin		Disk margin		H_∞ norm	
Parameters	Nominal	Worst case	Nominal	Worst case	Nominal	Worst case	Nominal	Worst case
LQR	9.02	1.86	15%	3%	0.158	0.032	6.963	21.09
H-infinity (fixed gain)	**9.67**	**4.11**	**16%**	**7%**	**0.169**	**0.072**	5.56	6.28
H-infinity (state space)	6.08	3.32	11%	6%	0.103	0.052	**5.15**	**5.68**

Finally, whole robustness analysis has been carried out on the linearised system in the case of this project, and at the end the results have been validated on the nonlinear system to guarantee feasibility. The reason why only linearised system was taken in consideration is the system complexity. The nonlinear system model with nominal parameters written as MATLAB function in a file takes 4.3MB or memory. If we consider a nonlinear system with the physical parameters as the parameters of the system, generated MATLAB in that case has over 100MB.

For the same reason MATLAB the Robust Toolbox was not used in its full extent, the system is just too complex.

Future work

The logical extensions of the system is the *gain scheduling* algorithm where the H-infinity controller would not be calculated only for one linearisation point but rather be parameter of the current measured variables: $u = K(y)\,x.$ where $K(y)$ is a nonlinear controller changing its values based on current states of the system. Of course measured outputs y are continuous variables and it is very complicated to find controller $K(y)$ for every possible combination of y, therefore the state of possible y needs to be discretized into a table. For each table entry the nonlinear system is going to be linearized and the optimal H-infinity controller is calculated for it. In this way the change in the controller $K(y)$ compensates for the nonlinearity of the system.

Literature

1. Latchman, H.A., Crisalle, O.D. and Basker, V.R., 1997. The Nyquist robust stability margin—a new metric for the stability of uncertain systems. *International Journal of Robust and Nonlinear Control: IFAC‐Affiliated Journal*, *7*(2), pp.211-226.
2. Scherer, C., 2001. Theory of robust control. *Delft University of Technology*, pp.1-160.
3. R van den Bleek, R.A.P.M., MIMO H∞ control design for the AVIS.