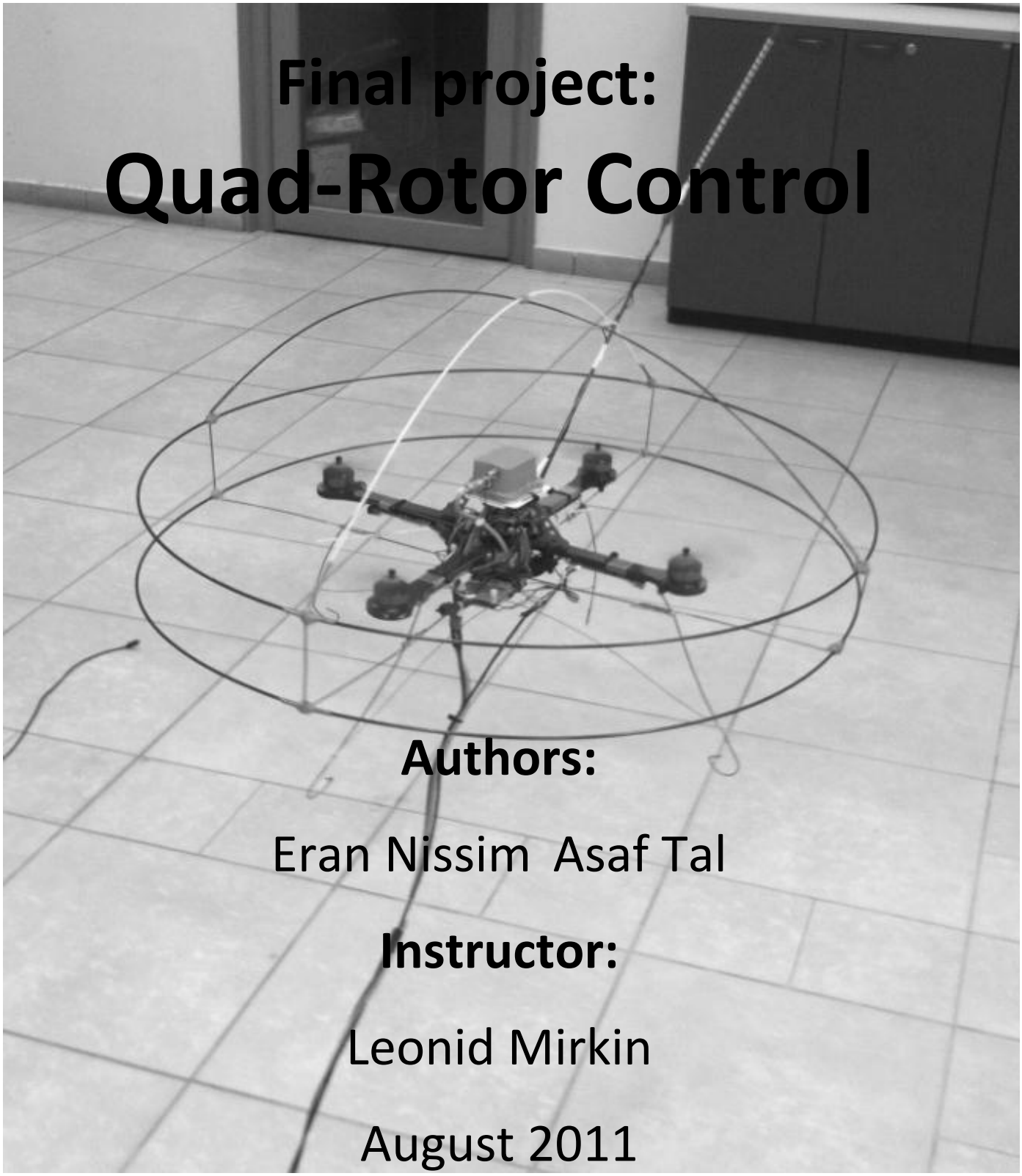
הטכניון - מכון טכנולוגי לישראל
TECHNION - ISRAEL INSTITUTE OF TECHNOLOGY
THE DEPARTMENT OF MECHANICAL ENGINEERING



Final project:
Quad-Rotor Control

Authors:

Eran Nissim Asaf Tal

Instructor:

Leonid Mirkin

August 2011

Table of Contents

1 Preface

The presented report documents the result of the project devoted to dynamic modeling and control design development for an autonomous micro quadrotor.

The quadrotor is of the type of X-3D-BL Research-Pilot.

Our objectives in this project are to identify open loop transfer function, considering "black box" controller added for stabilizing, in all 4 channels: pitch roll yaw and throttle, from experiments on a pre-stabilized quadrotor and design a controller for each channel and inquire coupling between channels.

In this report we present:

- Given hardware and interfaces for real time control.
- System frequencies prediction and sampling times of the sensor.
- Various ways of identifying open-loop transfer function from the close loop measurements for each channel as siso linear models.
- Controller design for each channel as linear siso systems
- Identifying the coupling between channels to asses mimo transfer function.

2 Introduction

The quadrotor is a MAV- miniature aerial vehicle type. The development of the MAV is accelerating these days in order to use this device in various fields, such as precision payload delivery, aid in rescue missions,aerial surveillance
In complex or cluttered environments like office buildings and commercial centers, there is a need to acquire intelligence in hostile or dangerous environments such as caves, forests, or urban areas, rather than risking human life.

This project involves several engineering fields in order to achieve the maximum performances, the quadrotor in the aspect of mechanics have to be light, but still strong in order to endure harsh weather and strong impacts.
In addition electronics have to be on board such as sensors, controllers and in several quadrotors also camera.

In our project, we received the quadrotor from the ISL lab in computer eng , and in addition we received computer code that is communicating with an electrical box that sends and receives information in 50 Hz, an IMU sensor from XSens and a remote control that has option to control the quadrotor manually or automatically.

3 System description

3.1 System components

3.1.1 X-CSM

The X-CSM is the mechanical frame of the X-3DBL UFO. The booms, which are made of a rigid carbon fiber-balsa wood sandwich material, can bereplaced individually. The central unit of the frame called the ”X-CSM Core” is made of light weight laser-cut magnesium parts. Being built out of these state-of-the-art materials the X-CSM is a very robust high-tech basis for your quadrotor aircraft.

Figure3.1: X-CSM

3.1.2 X-Base

The X-Base is the central control unit which isconnected to and communicates with all activeelements of the X-3D-BL. Next to the battery, themotor controllers, the X-3D gyro and the receiveryou can also connect several LEDs to the X-Base togive your X-3D-BL a unique fancy look.

3.1.3 X-3D

The X-3D is the sensor unit of the X-3D-BL. Withthree piezo-gyros and highly optimized controlloops it does the actual flight/attitude control. Allparameters influencing the in-flight behavior can betuned by connecting the X-3D to a PC using the USBadapter that came with your X-3D-BL. Once you areon the field for flying you can select four differentparameter sets using two jumpers.

3.1.4 Motors

The X-BL-52s motors by HACKER Motors Germanyare custom-built for the X-3D-BL. The motors areperfectly suited for the application in this vehicle.

Figure3.2: the X-BL 52s motor

3.1.5 X-BLDC brushless motor controllers

Every motor is controlled by an independent XBLDCbrushless motor controller. The controllers arehighly optimized for the X-BL-52s motors and thusensure the highest efficiency possible. Please note thatfor this reason the controllers might not work with adifferent motor type.

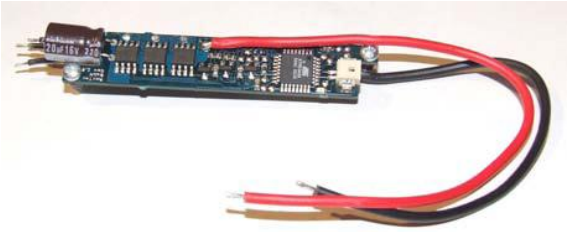


Figure3.3: XBLDC brushless

3.1.6 System remote control

Our system has 2 configuration of control. Manual control, that is completely manual in all 4 channels, and automatic control.

Switch between manual and auto

Figure3.4 remote control

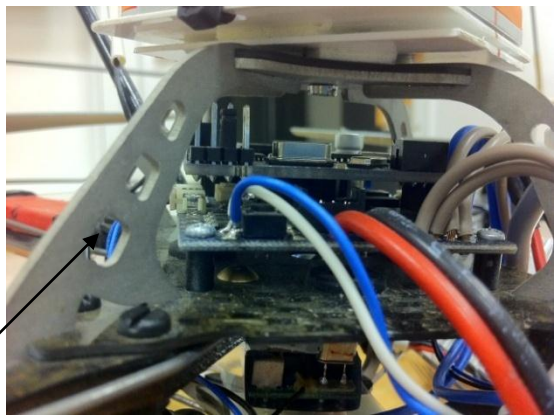
Figure3.5: switches

In the automatic control mode we have the option of choosing between complete automatic control in all 4 channels or manual in few channels and automatic in the others.

This feature of semi-automatic gives us the ability to examine our automatic controller will manually stabilizing the other channels.

Switch between semi automatic and manual control in the automatic mod

In addition we have 3 activation mods for the quadrotor that we can switch between them by moving an on-board jumper. The 3 mods are Beginner, Advanced, and Expert. These mods relate to the quadrotor's inner controller and change its band-width according to the mod. The change of the quadrotor's mod effects on the dynamics of the system and makes the helicopter react faster or slower to the given control signal.



Mode change jumper

Figure 3.6: jumper

3.2 Basic control

The quadrotor generates upward force using four rotors whose angular velocity can be controlledusing the remote control.As a result of rotating rotors, liftforces are generated. As the rotors rotate they are subject to drag caused by the air beingmoved. The drag results in a reactive moment on the rotors known as the induced moment. Theinduced moment acts on the rotor in the opposite direction of the rotors direction.

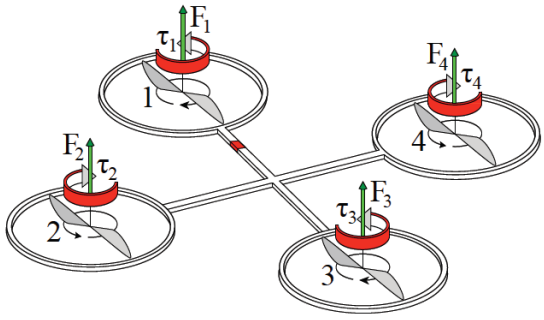


Figure3.7

As a result of counter rotation between two pairs of rotors, counter moments has induced and results balance in the yaw channel

3.2.1 Pitch channel

Pitch is when the quadrotor's performs a rotation around the y-body axis. This is achieved by changingthe angular velocity of the rotors on the x-body axis that is rotor 1 and 3. If the desired rotation is positive the pitch is performedby increasing the angular velocity of rotor 1 and decreasing the angular velocity of rotor 3. Thisresults in a positive pitch or rotation around the y-body axis.

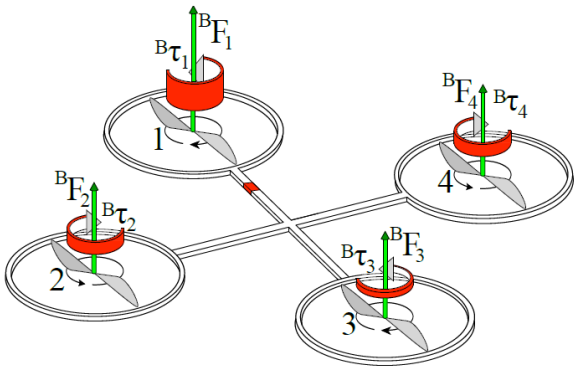


Figure3.8: pitch rotation

3.2.2 Roll channel

Roll is when the quadrotor's performs a rotation around the x-body axis. This is achieved by changingthe angular velocity of the rotors on the x-body axis that is rotor 2 and 4. If the desired rotation is positive the roll is performedby increasing the angular velocity of rotor 2 and decreasing the angular velocity of rotor 4. Thisresults in a positive roll or rotation around the x-body axis.

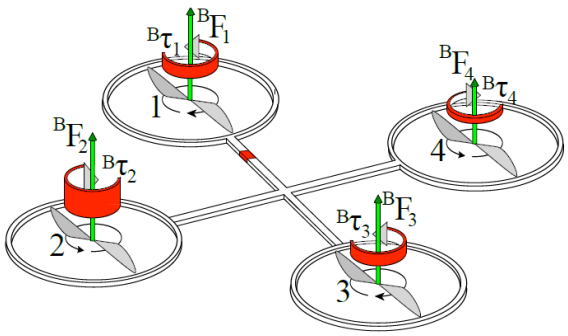


Figure3.9: roll rotation

3.2.3 Yaw channel

Yaw is when the quadrotor preforms a rotation around the z-axis. This rotation is achieved by changing simultaneously the angular rotation of one pair of rotors. Positive rotation around the z-axis is results from increasing angular rotation in motors 1 and 3 and as a result , the torque increases in those motors and results moment around the z-axis which makes the quadrotor spin .

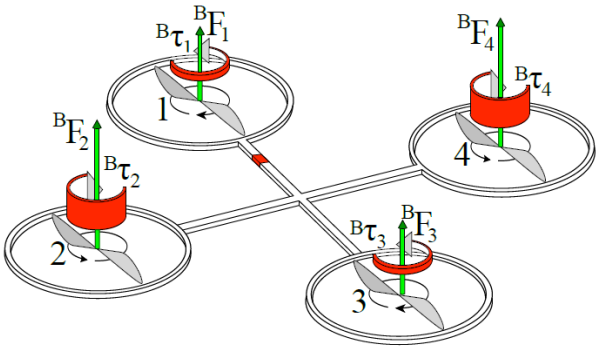


Figure3.10: yaw rotation

3.2.4 Throttle channel

The throttle along z-axis is achieved by increasing the angular rotation equally in all 4 motors, which results an increase in the lift force which makes the quadrotor gain altitude.

3.3 Transmitter

The transmitter we used is a custom made transmitter that is connected to computer through com port.

This transmitter gives us the ability to design a reference input to the system and a controller in matlab environment, compile it with a computer code and transmit it to the helicopter. Communication to and from the transmitter is in 100 Hz

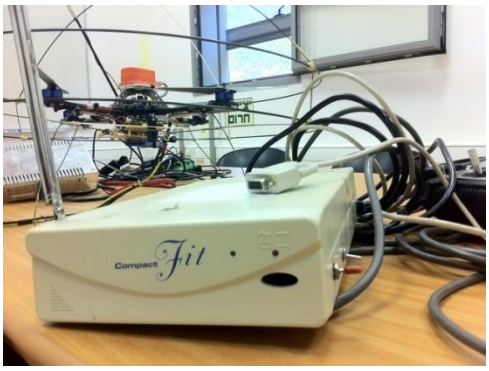


Figure3.11: transmitter

3.4 Sensors

The QuadRotor's platform was further equipped with an IMU system designed by
the German company XSens, the MTi-G. The MTi-G is a GPS aidedMEMS based IMU and static pressure sensor. The MTi-G has an onboard Attitude and Heading Reference System (AHRS) andNavigation processor. This low-power Digital Signal Processor runs a real-time Kalmanfilter providing GPS-enhanced, 3D orientation data.

Figure3.12: MTI-G sensor

3.5 Blocks diagram

After analyzing the system architecture we understand that our helicopter dynamics is affected by an on board given controller that supposed to stabilize the system and we have no ability to change this controller or know its effect on the system.

Therefor we cannot predict the system behavior with the understanding of the dynamic equations of the helicopter only.

In the next figure we can see the block diagram of the open-loop transfer function and the effects of the inner controller.

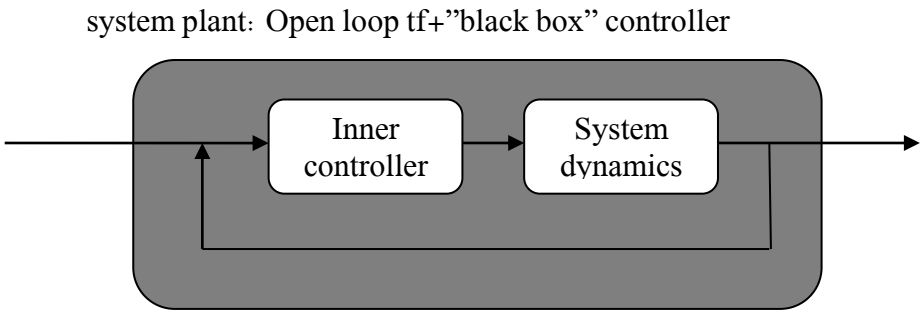


Figure3.13: Process

3.6 Experiment system

Our system contains the helicopter with it sensors, the computer and the remote control.

The sensor operates with the frequency sampling of 512 Hz, while the remote control is transmits the data to the helicopter with the sampling frequency of 50 Hz.

In fact in order to activate the system, we required to synchronize all those system the: acquisition data system and transmitter system.

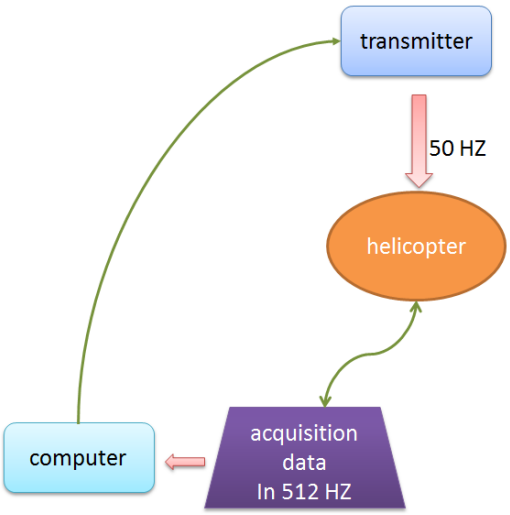


Figure 3.14: system scheme

4 Frequency analysis

4.1 Introduction

In order to receive better measurements and being able to sort inaccuracies between the measurement signal and the simulation signal to disturbances in the experiment or noise from the sensor we must understand our sensors.

Our sensor system is an IMU (inertial measurement unit) from XSens [3.4]

Our unit takes measurements of angular rate and linear acceleration in all 3 axis. Using The MT-Manager software that was given with the sensor, gave us the ability to choose sampling rate of the sensor.

The IMU has the ability to estimate the Euler angles using an onboard Kalman filter, but this option restricts the sampling rate to 100-120 Hz.

We choose to work with sampling rate of 100 Hz and control the Euler angles.

4.2 Tests and problems encountered

First tests

Our first measurements took place on a swing. The experiment took place while the quadrotor is mounted to a wire as shown in the picture.

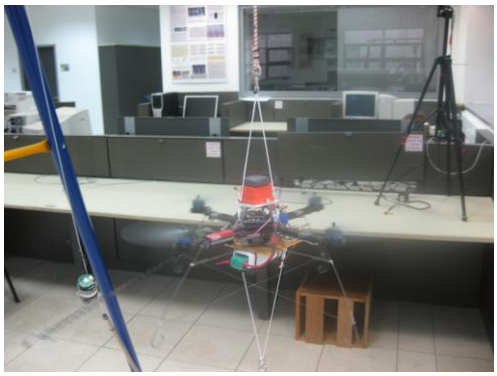


Figure4.1: quarotor on awing

In the first tests, in order to test the helicopter dynamics we need a stable controller that makes our helicopter hovering.

We choose in the begging a proportional control, but because this controller is sensitive to interruption we switch the controller to PID, the tuning set by trial and error.

The helicopter react good to the controller for several seconds and afterwards start swinging.

We could see from the IMU output that the quadrotor indeed swings and we saw that the sensor measures large angles. We guessed that the instability of the helicopter results from large disturbances from the rope and from the restriction of the movement of the helicopter.

Second set of tests

After sensing that we could encounter problems with the rope we decided to disconnect it and try to preform test without any constrains on the system.

As we made those tests after few seconds the helicopter started to divert from his desire angels. It was obvious that the behavior of the helicopter on the swing wasn't according as a result of the wire as we suspected.

Therefore we decided to take measurements from the sensor while the helicopter is in manual mode while hovering.

4.3 Manual record conclusion

After the manual record we notice that the IMU starts to measure unrealistic angels, which don't reflect the actual condition of the helicopter, after 30 second.
In addition some other student that work with this sensor before us, mentioned that we can't rely on the IMU results of measuring angles, because this sensor has inner observer which isn't suitable for helicopter behavior.

Our first assumption is that we have probably some uncertainty in the helicopters spectrum.
That is why we decided to check the helicopter spectrum.

This is the result we got from the acceleration and the gyro in the three different axes.
The Fourier Transform made by the function **"fouriertrans"**{appendix A}.

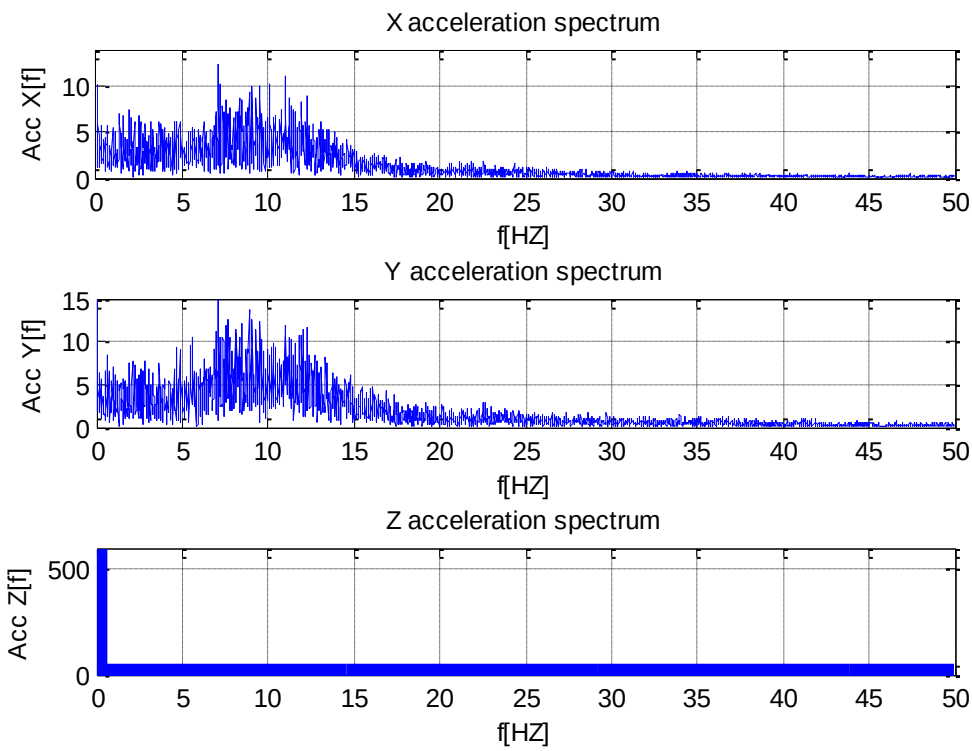


Figure 4.2: acceleration spectrum with sampling of 100 Hz

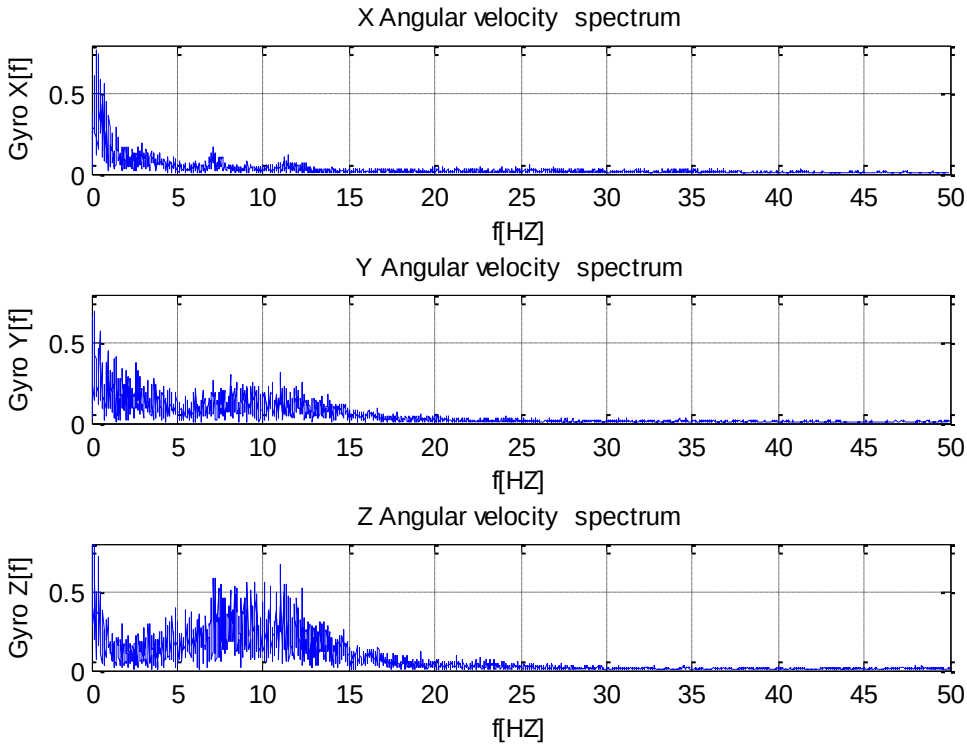


Figure 4.3: angular velocity spectrum with sampling of 100 Hz

We can see that we received spectrum of signals within the range of 0-50 Hz. This result makes clear that when we want to observe the Euler angle with the onboard kalman filter, all signals that has the spectrum larger than 50 Hz (nyquist frequency) will damage our observation of the angle (folding effect).

Quadrotor signals spectrum prediction

We realize that we must understand the spectrum of the signals in the system in order to determine the correct sampling rate of the helicopter.

Our prediction was:
Assuming speed rotor of $n = 3000 \sim 4000[rpm]$ and because we have two blades to each rotor.

$$f = \frac{3000}{60} \cdot 2 \sim \frac{4000}{60} \cdot 2 = 100 \sim 133[HZ]$$

From this assumption we can see that wrong sampling rate has chosen an there for we decided to increase the sampling rate to 512 Hz

4.4 Aliasing effect

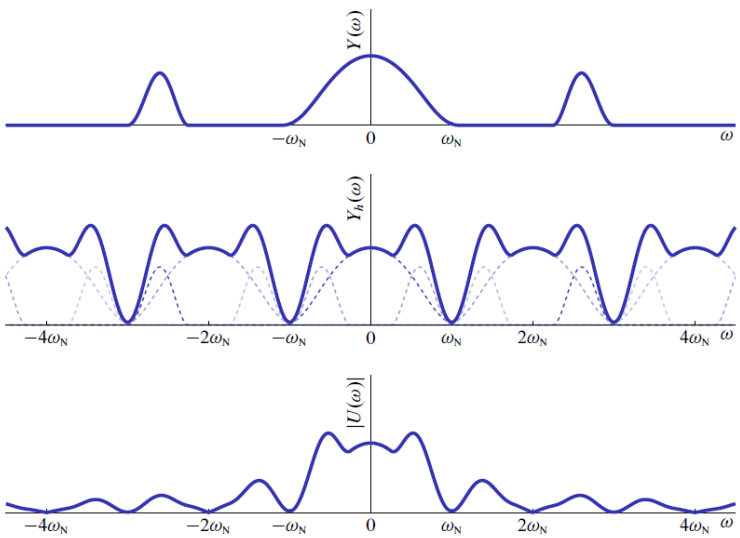
One of our concerns is that the spectrum that we received from the Fourier Transform as shown above, is not the actual spectrum of the system, we sense this because no high spectrum signal has appeared in the range.

Therefore, we can understand that we have aliasing affect.

4.4.1 The aliasing problem

In theory, a Nyquist frequency just larger than the signal bandwidth is sufficient to allow perfect reconstruction of the signal from the samples; however, this reconstruction requires an ideal filter that passes some frequencies unchanged while suppressing all others completely. In practice, perfect reconstruction is unattainable. Some amount of aliasing is unavoidable.

Signal frequencies higher than the Nyquist frequency will encounter a "folding" about the Nyquist frequency, back into lower frequencies. For example, if the sample rate is 20 kHz, the Nyquist frequency is 10 kHz, and an 11 kHz signal will fold, or alias, to 9 kHz. However, a 9 kHz signal can also fold up to 11 kHz in that case if the reconstruction filter is not adequate. Both types of aliasing can be important.



Another example to aliasing effect we can see in the next figure.

We can see that sampling the signal at 2Wn causes the spectrum that is after the sampled frequency to fold.

Figure 4.4: example of folding (from control theory notes by Leonid Mirkin)

4.4.2 Anti-aliasing

When the condition $\frac{fs}{2} > f$ is met for the highest frequency component of the original signal, then it is met for all the frequency components, a condition known as the Nyquist criterion. That is typically approximated by filtering the original signal to attenuate high frequency components before it is sampled. They still generate low-frequency aliases, but at very low amplitude levels, so as not to cause a problem. A filter chosen in anticipation of a certain sample frequency is called an anti-aliasing filter. The filtered signal can subsequently be reconstructed without significant additional distortion.

4.4.3 Aliasing in our signals

As we can see from the result of the test that we measured with higher sampling rate, the system indeed has high frequency signals.

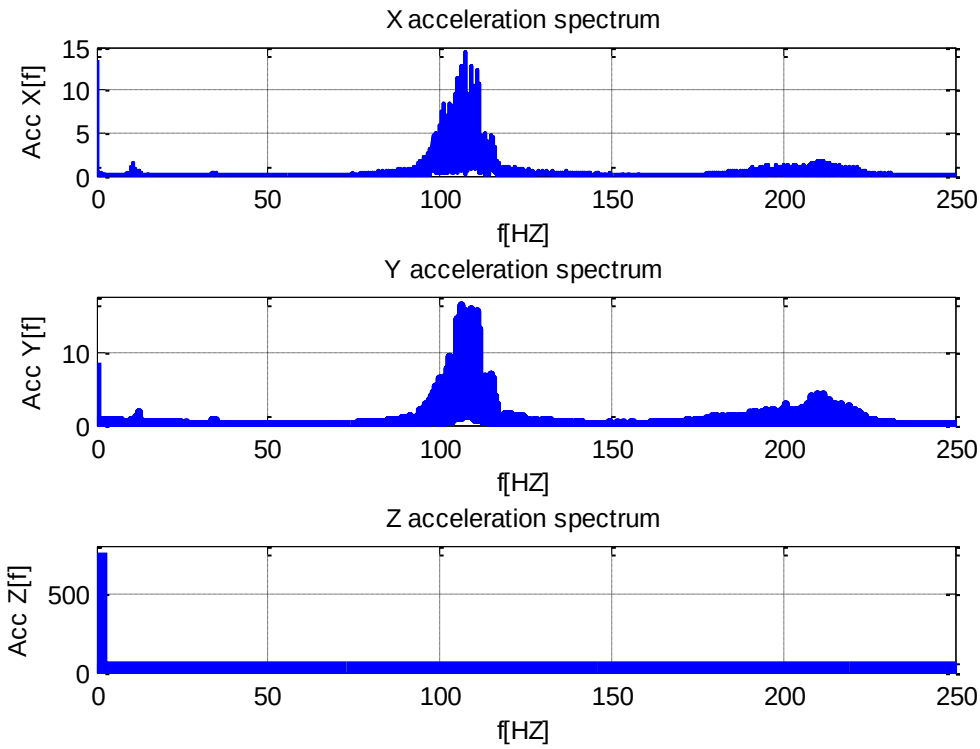


Figure 4.5: acceleration spectrum with sampling of 512 Hz

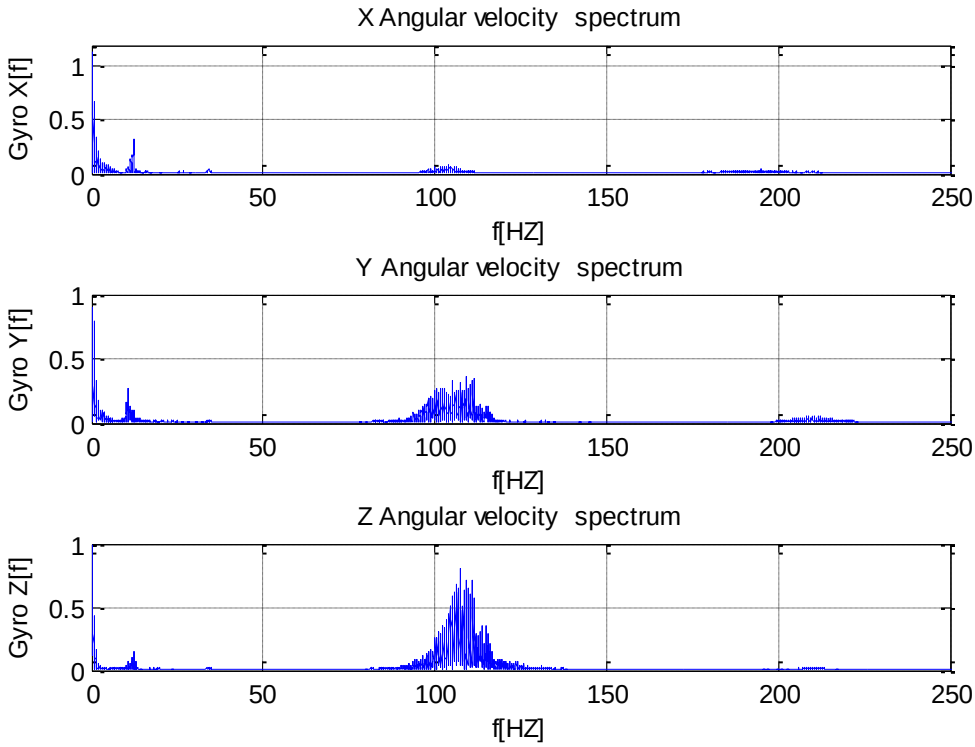


Figure 4.6: angular velocity spectrum with sampling of 512 Hz

We can see from the plots that high frequencies indeed exists in the system ($f = 90 - 130[HZ]$) and as a result of measuring in low sampling rate we received the aliasing effect.

4.5 Filtering signal

In order to inquire the helicopter spectrum and to verify our conclusion, we decided to filter our signal through antialiasing Filter, and then to perform a down sampling, we assume that after we filtered the signal, we won't suffer from the aliasing effect, because all high frequencies signals will be filtered .So we perform the test according to next figure.

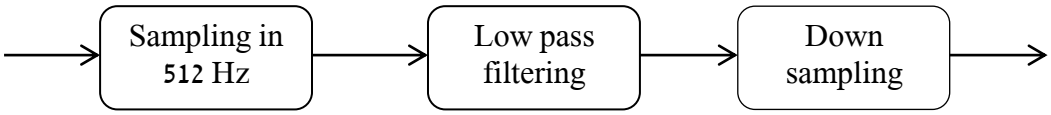


Figure 4.7: The process of filtering the signal

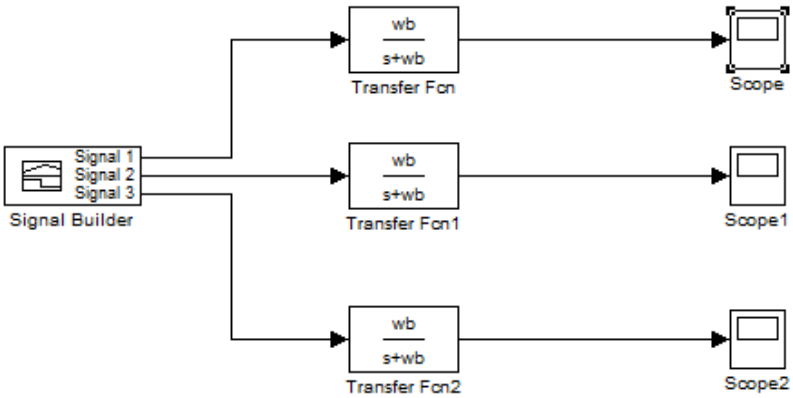


Figure 4.8: scheme of the signal filtering

Our "antialiasing Filter": is: $L.P.F = \dfrac{w_b}{s + w_b}$; $\qquad w_b = 50[HZ]$

The signal with sample time of $f = 512[HZ]$ after L.P.F:

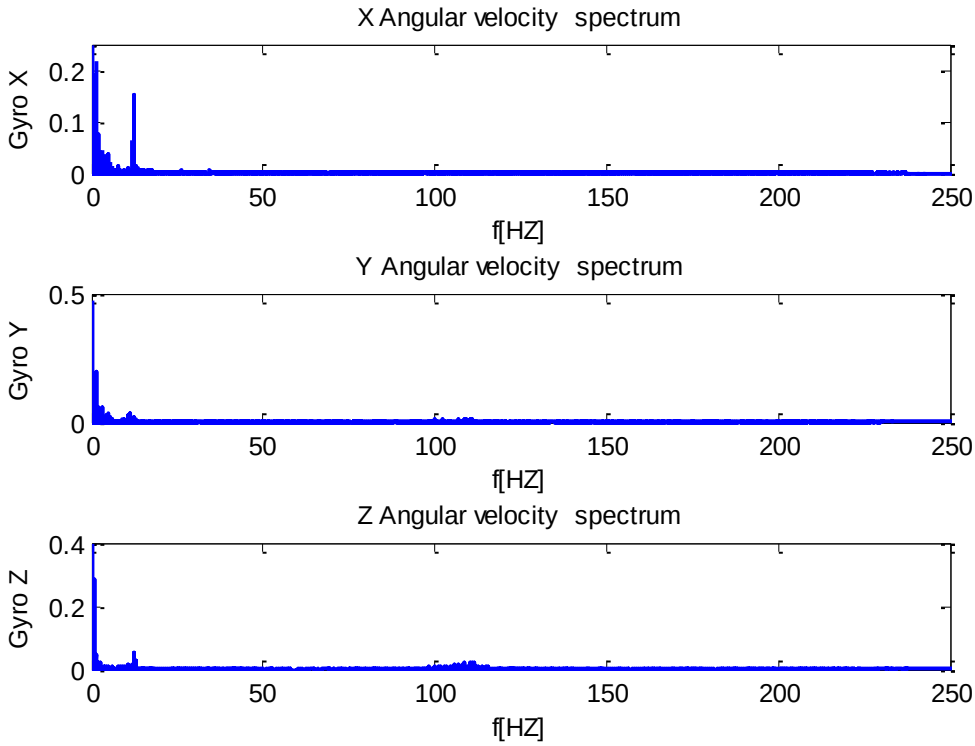


Figure 4.9: spectrum of sampling at 512 Hz after filtering

As we can see form this figure we indeed decrease the system spectrum frequency.

The signal with sample time of $f = \frac{512}{5}[HZ]$ after "antialiasing Filter":

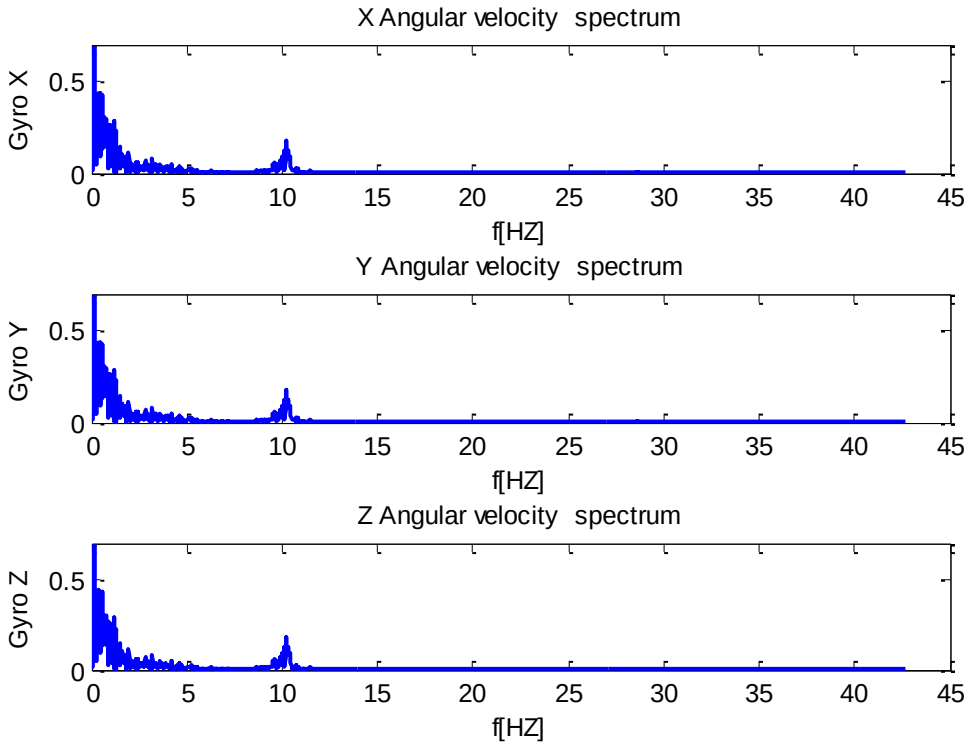


Figure 4.10: spectrum after filtering and down sampling

As we see from this figure a lot of our previous frequencies now disappeared, because we don't have folding frequencies in the system.

Now we had performed a test which makes an averaging process to the signal.

$$average\,Filter = \frac{1 - se^{-5h}}{5 \cdot s}$$

Passing the signal through averaging process it's actually similar to passes the signal through "antialiasing Filter", because it passes only the lowest frequencies.

We compered those result to the signal that we received from the "antialiasing Filter" and to the signal with rate sampling of 512 Hz.

We know that the signal with higher sample rate will be without an aliasing, so we want to exam which filtering is causing to the signal to be similar to the original signal (without aliasing).

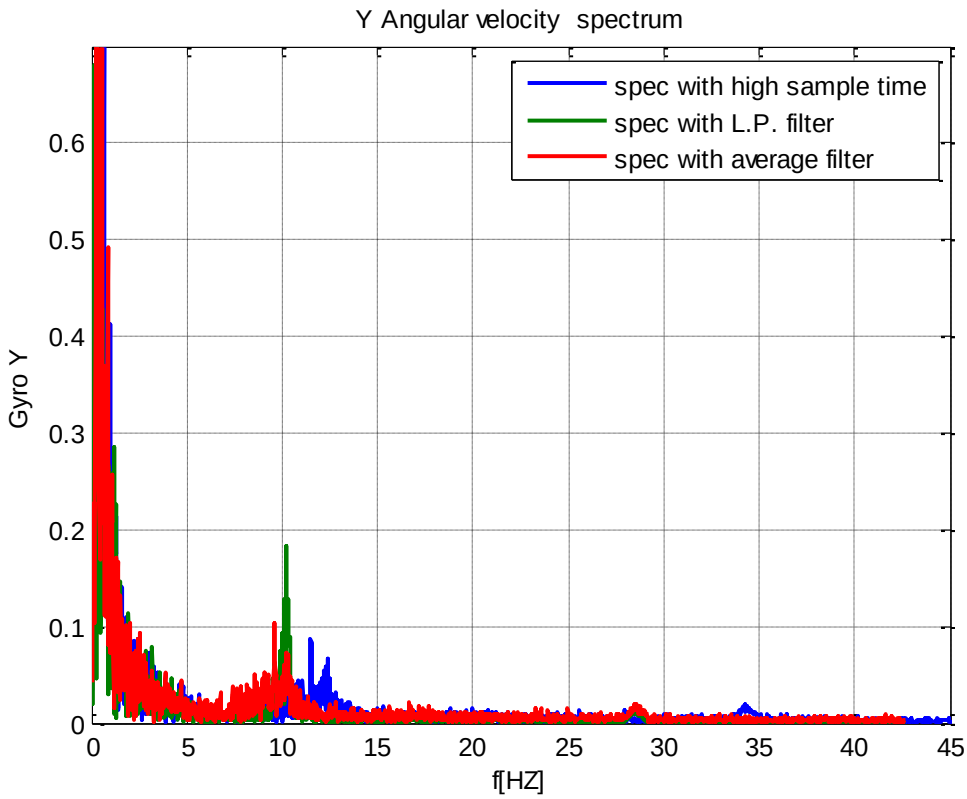


Figure 4.11: comparison between L.P and average filters.

As we can see from this figure the blue line is the real spectrum because it sampled at high rate, the red line is the spectrum in sample rate of 100 Hz but after averaging filter, the green line is same to the red ones, but with L.P.F.

From this result we can see that there is a range of frequencies around 10 Hz that the L.P.F is not filtering well.

At a range frequencies around 12 Hz, the averaging isn't filtering well.

Our conclusion from this comparison is that antialiasing filter is a suitable solution for us in case we need to work at sampling rate that is smaller than 512 Hz.
we need to remember that our antialiasing filter it a first order tf and if we increase the order we will receive a better filtering.

The reason that we can filtered the signal it because, the highest frequencies of the spectrum are a result of noise measurement and we don't need them for analysis the signal.

4.6 Summary

- Errors such as instability of a system can occur as a result of bad measurements.
- When sampling a system we must first estimate the system's spectrum in order to implement a correct sampling rate
- As a result of low sampling rate relative to the actual frequencies of the quadrotor aliasing effect had occurred.
- The helicopter orientation can be measured on a low sample time only if the sensor signals filter with some kind of antialiasing filter, it recommend the filter will be with higher order then one.
- We can filter the signal because higher frequencies of the spectrum are result of measurements noises.
- Higher sample time allow us to measure the helicopter acceleration and orientation without any filtering.

5 Model identification

5.1 Theory

The first principles modeling are to have a predicted model, that we could compare it to our selected model.

5.1.1 Linearization of the model (deviation model)

Our control signal is the pulse that is sent to the helicopter.

Using a scope we successfully identified the range of the pulse.

The pulse range in all 4 channels is:

	Min[pulse]	Max[pulse]	Control name
Pitch channel	684	1350	U1
Roll channel	710	1355	U2
Throttle channel	782	1238	U3
Yaw channel	715	1356	U4

Table 5.1: input value of all 4 channels

In order to control the system with LTI control system theories we shell linearize the signal and control the system around the equilibrium point.

The new control signals now are:

$$\bar{u}_1 = u_1 - 995, \quad \bar{u}_2 = u_2 - 990, \quad \bar{u}_3 = u_3 - 1123, \quad \bar{u}_4 = u_4 - 985$$

5.1.2 Theoretic transfer function model

We know that the control signal is passing through inner controller that has a gyro, we consider the inner controller as a “black box” and we cannot assume its effects on our dynamic model. There for theoretic transfer function in not relevant and we cannot use dynamic equations to conclude about the model transfer function.

But we consider in the future removing the inner control and for that we add our theoretic model to the appendix B.

5.2 Identification methods

As we want to receive our parameter from the tests, we had to determine our modeling method.

In this project we study several methods for modeling systems, but not all of them are practical in order to find our parameters.

5.2.1 Step response

In this method we set the reference signal to be a step response, and then we examine our output signal and by that we can determine the optimal parameters that make match of our simulations to the real output signal.

There are several points that have to take in consider when identifying using step reference:

- The reference has to be big enough so we could identify the parts of the output that relates to the system response and the parts that relates to the noisy measurement taken by the sensor.
- The response has to reach steady state in order to examine parameters such as settling time.

The main problem with step reference in roll/pitch channels is that our system is a dynamic system, because of that, when we insert a big step to the system the quadrotor tilts in the reference angle and accelerating.

Because of the room boundaries we cannot insert to the system reference bigger then 3-4 degrees and still wait for steady state. There for reference angle of 3 degrees has been chosen, but the problem is that we cannot distinguish between the system behavior and the measurement noise in this case.

There for this method of identifying is not acceptable for pitch and roll channels; only for yaw channel.

5.2.2 Sine response

In this method we set the reference signal to be $r = A\sin(wt)$.We know from the finite value theorem that if the system is stable, our output will be $y = B\sin(wt + \phi)$.

So by that we can find the amplitude and the phase of the system in any frequency.

From testing the system in different frequencies we could map the bode figure and recover the system's model.

Because of the system dynamics our method of inserting sine functions to the system is by constraining the quadrotor to a rod, from this configuration we constrain the system to rotate only in one axis.

Figure 5.2: the helicopter constraining to a rod

From this test we build bode diagram for measurement amplitude as function of the frequency and by that we can calculate models parameters.
we can see the bode diagram of real and simulate system in figure 5.3.

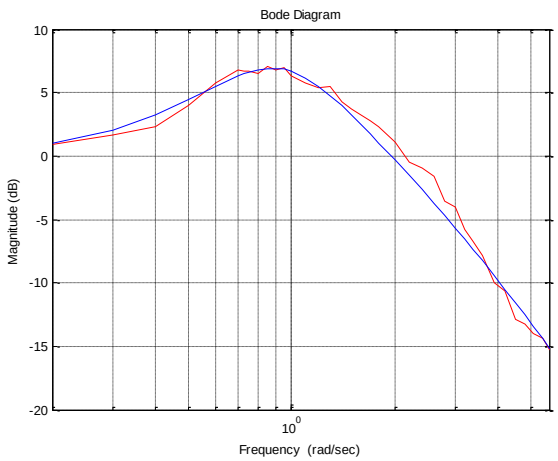


Figure 5.3: Usage of sine method

5.2.2.1 Lissajou figures

In order to measure the amplitude in every measurement point we studied a method which his called lissajou figure.

This method is define: $x = r(t) = A\sin(wt)$, $y = y(t) = B\sin(wt + \phi)$

as we need for bode diagram the variable : $B/A(w) = amplitue(w), \quad \phi(w)$.

By the plot of the graph of y as function of x we receive ellipse as shown in figure 5.4.

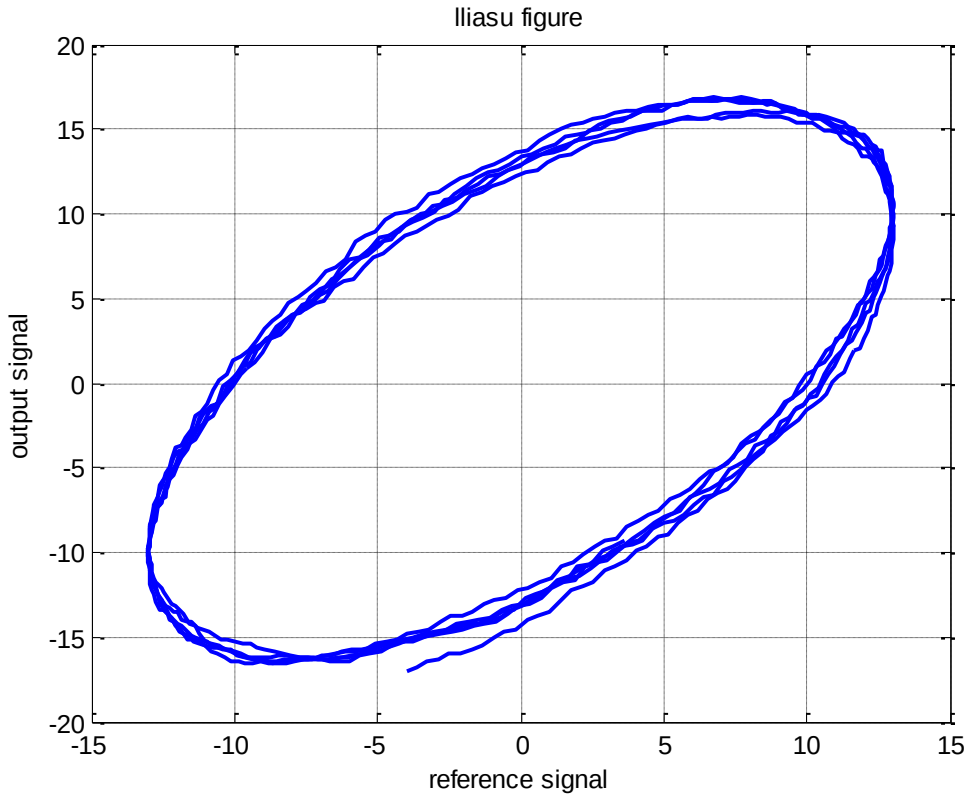


Figure 5.4: example of lissajou figures from our tests

From this figure we can measure the variable: $\frac{B}{A}, \phi$, by measure two specific point.

Point A and point B, with those points we receive the ratio $\boxed{\frac{B}{A} = amplitude}$, that's our amplitude.

After we identify points A, B we can draw two auxiliary lines (red color), as shown in figure 5.5.

Because the lines are in the vectors $(A, B), (B, -A)$ direction, then we can measure point C and D, those points are the most distant from the origin.

By knowing point C and D we receive: $\boxed{\phi = 2 \cdot tg^{-1}\left(\frac{D}{C}\right)}$.

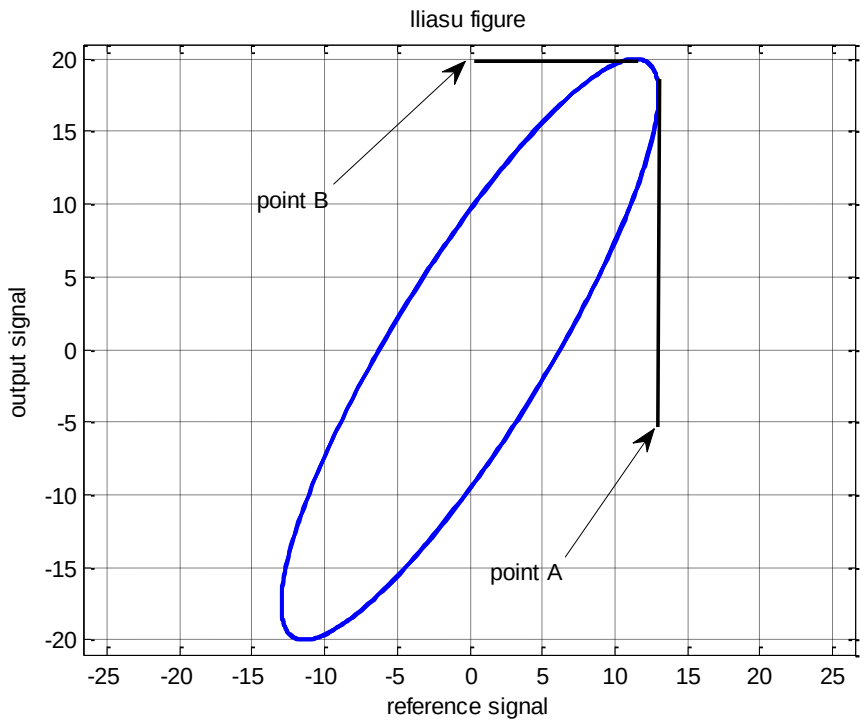


Figure 5.5: finding point A and B

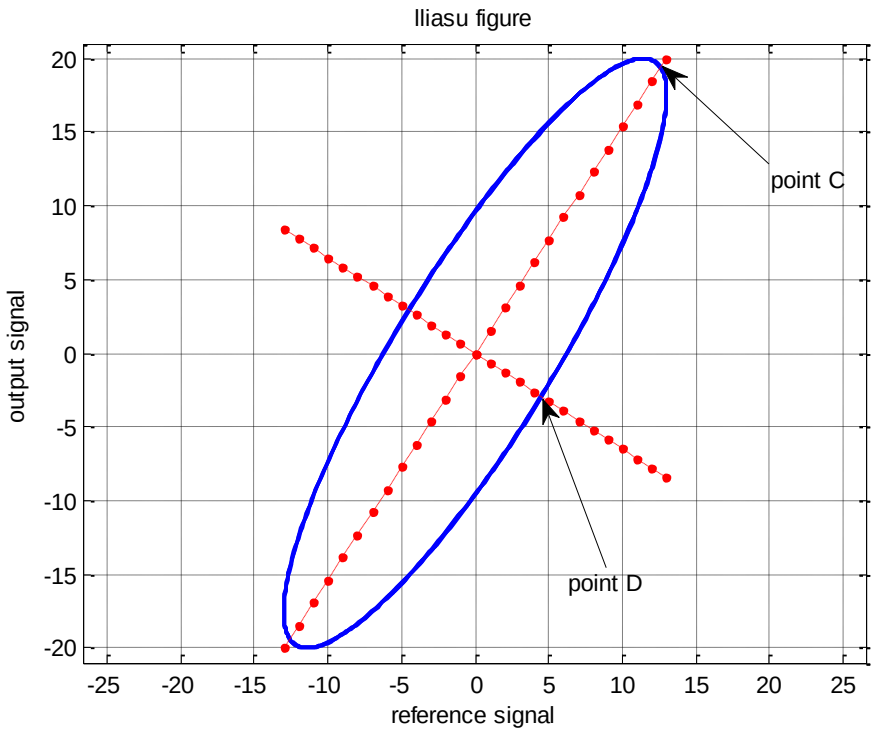


Figure 5.6: description of point C and D

Application of lissajou figures:

- ✓ Model with a noise measurement.
- ✓ For uncertainty models.
- ✓ for non-linearity models

5.2.3 Chirp function

In this method we set reference signal to be with changing frequency every second, so by that our spectrum will contain all the frequencies that we order to him to contain and those frequencies are crucial in order to identify the model of the system.

After we had received our output signal, we analyzed the spectrum and by divide the Fourier transform of the output and the input, we receive the amplitude of our transfer function on range of frequencies (only the vital ones).

Example of input and output thatoutcomeby chirp function in the reference signal.

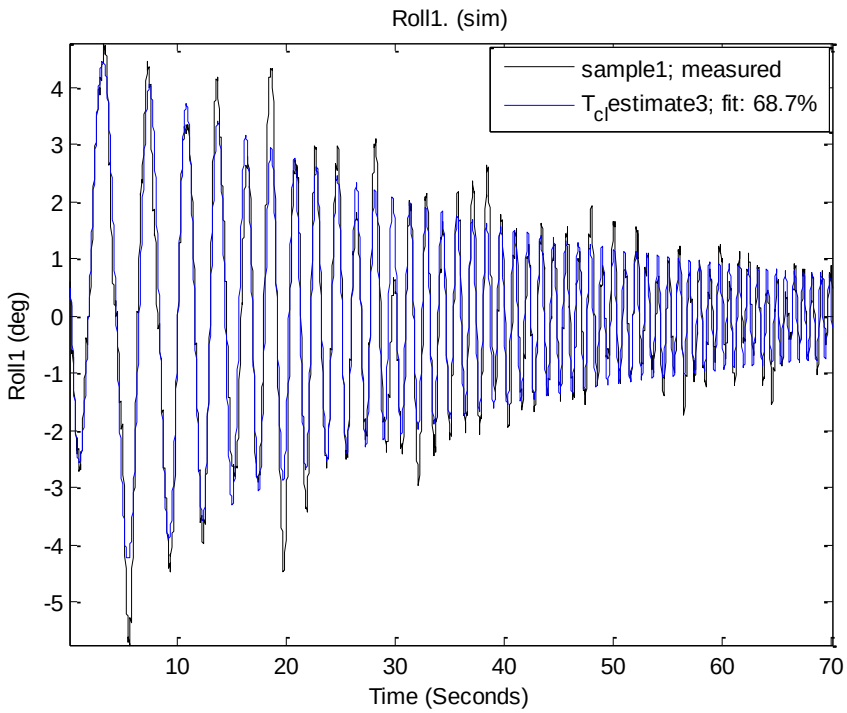


Figure 5.7: example of chirp function

The main advantage in this method is that we can insert large spectrum of signals in one test, there for, random disturbances effect evenly in all frequencies in contrast of inserting one frequency at the time to the system and analyze it with lissajou figures or any different way

The disadvantage in this method is that first, we have a band limit for the frequencies that we can insert to the system. In our dynamic system, low frequency input can cause the system to crash the boundaries of the room .Second, the function spread the input frequencies evenly when we should have higher resolution in low frequencies and lower in low frequencies.

5.2.4 Pulse response

In order to stimulate the system in a limited space, we use a reference control in a form of pulse as shown at figure 5.8.

This is a very effective method because we can insert reference input large enough that makes the measurement noise become relatively small and still maintain the helicopter in one position and by that we avoid the limited space limitation.

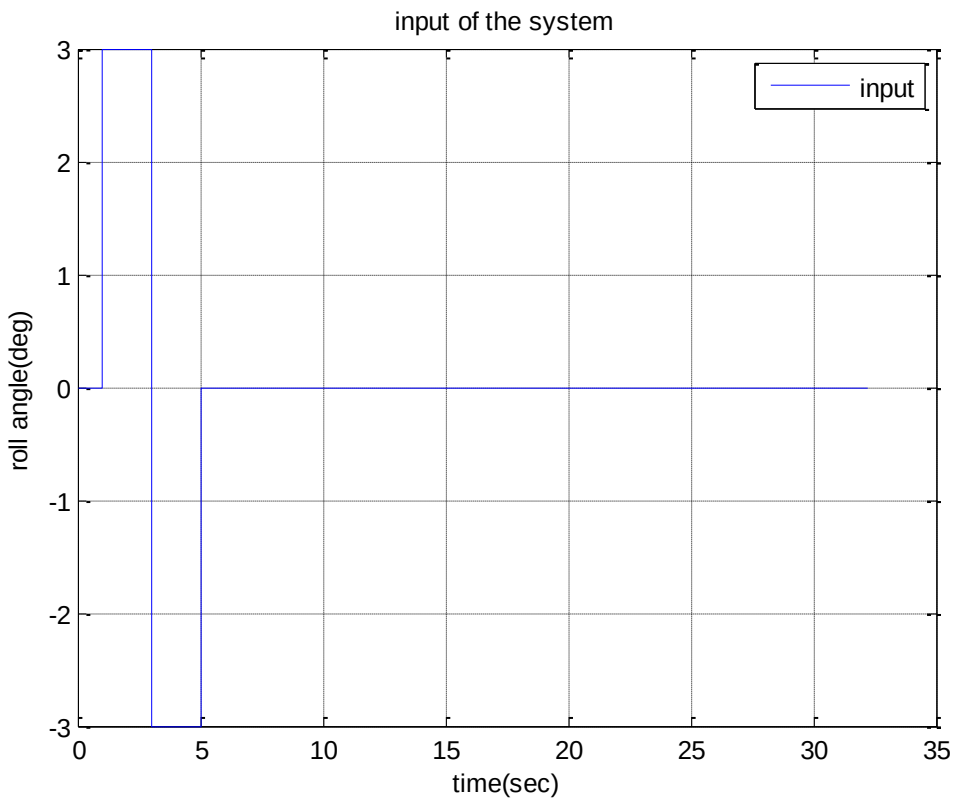


Figure 5.8: example of pulse function

5.3 Limitation in the process

In our tries to model the system we had quite a few problems; those problems caused us to receive a non-accurate model at the beginning.

Our main problems/limitation was:

- We preformed simulations on the helicopter in the computer science lab, this lab doesn't have un- limited space and every time that the helicopter received an angles that bigger then 5-7 degrees he went to the wall and we had to stop our test and to start over.
- The sensor has ±1° measurement error. Because of that, any step that is too small will be non-accurate; we can't distinguish between the real response and the response that came from our sensor.

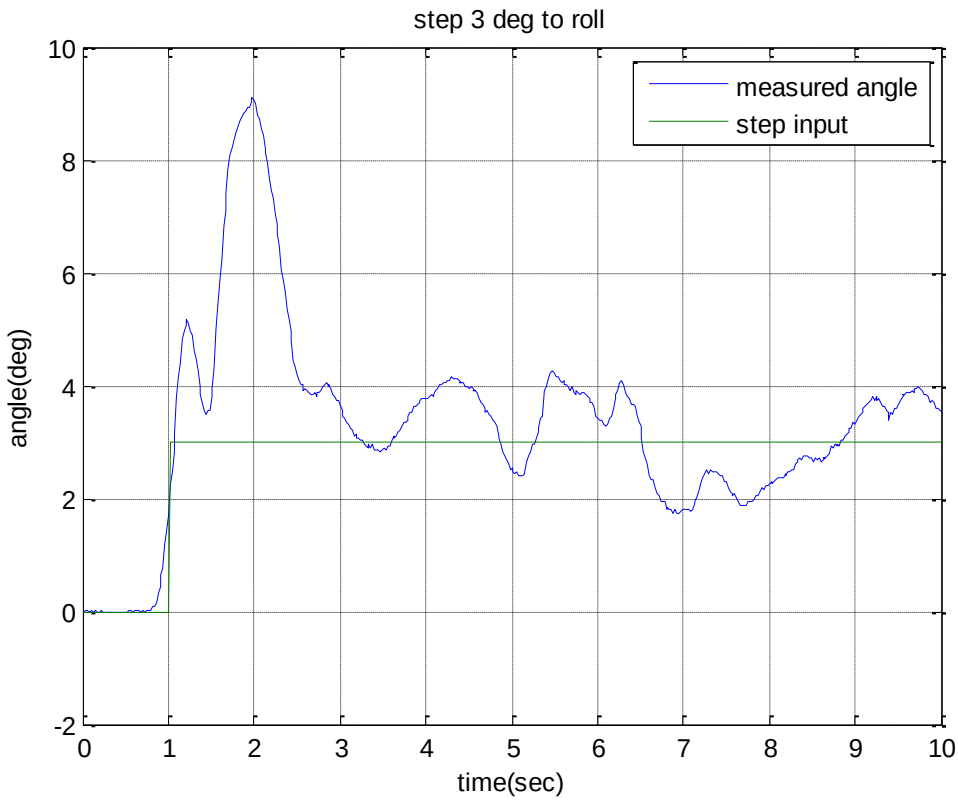


Figure 5.9: example of bed step response

- Our helicopter was connected to the computer with wires, which gave rise to applied forces on the helicopter. Of course those forces changed our model in a very random way.
  This is the reason that we receive a few models every test that we carried out. Fortunately those models were very close to each other.

5.4 Controller effect on modeling

In order to model the system we first needed a stable controller that caused the system to be stable, there are infinite controllers that stabilize the system, but not all of them are suitable for us.

It is common to think that a good controller for modeling a strong controller but it isn't like this at all.
Our modeling process is first to find a matching closed loop transfer function and then to reconstruct the open loop transfer function.

We know for sure that we have uncertainty in the closed loop tf, the question here is which controller causes less uncertainty in the reconstruction of the open loop tf.

We made some test in order to examine this question.

We examine the effect of two different controllers , were one, is a proportional controller and the other one is a pid controller. We examine those controllers on a random transfer function witch we think will be similar to the real one that we will identify.

$C_1(s) = k = 10$ $\qquad$ $C_2(s) = 10 + \frac{0.1}{s} + 0.5 \cdot s$

We assume process of: $P(s) = \frac{2.3}{s(s+3.7)}$

For frequencies range of we $w \in [0.3, 2.5]$, we see in figure 5.10 that for receiving a radios of 0.1 in the open loop we need less uncertainty in T2, it means he needs to be more accurate than the first closed loop tf.
So by this we notice that for two equal closed loop uncertainties we shall receive less uncertainty if we use c1.

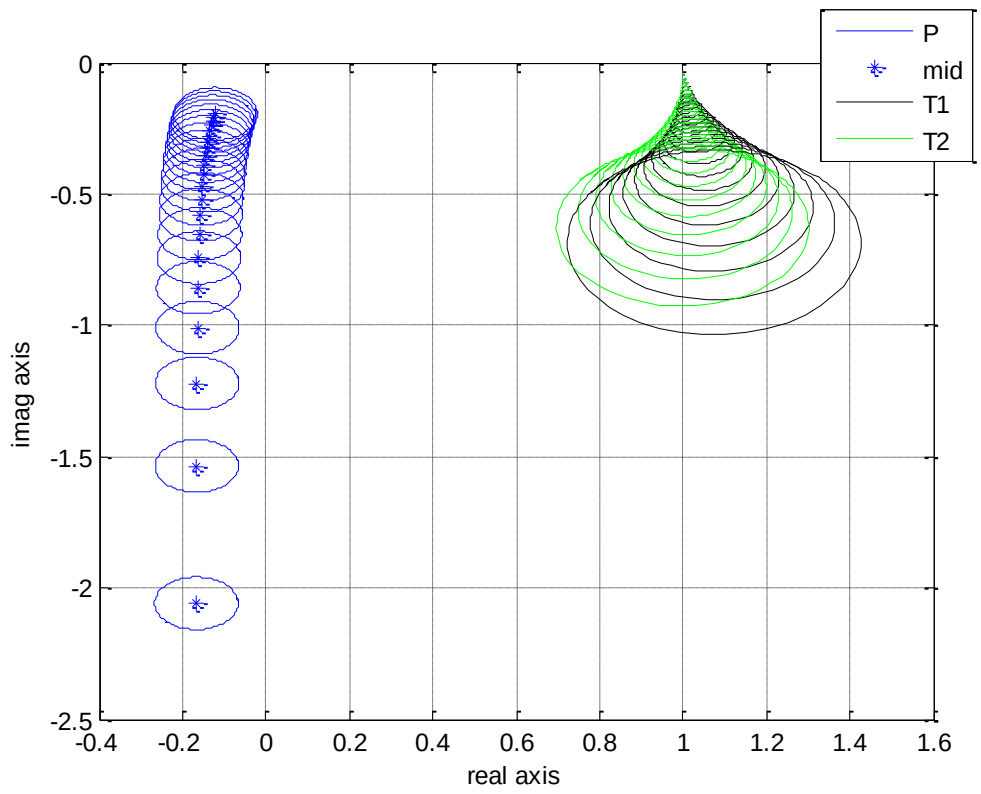


Figure 5.10: uncertainty circle's at lower frequencies

For much higher frequencies we see a significant difference between the controller as shows at figure 5.11, it mean that if we want to modeling the system for higher frequencies we have to choose c1.

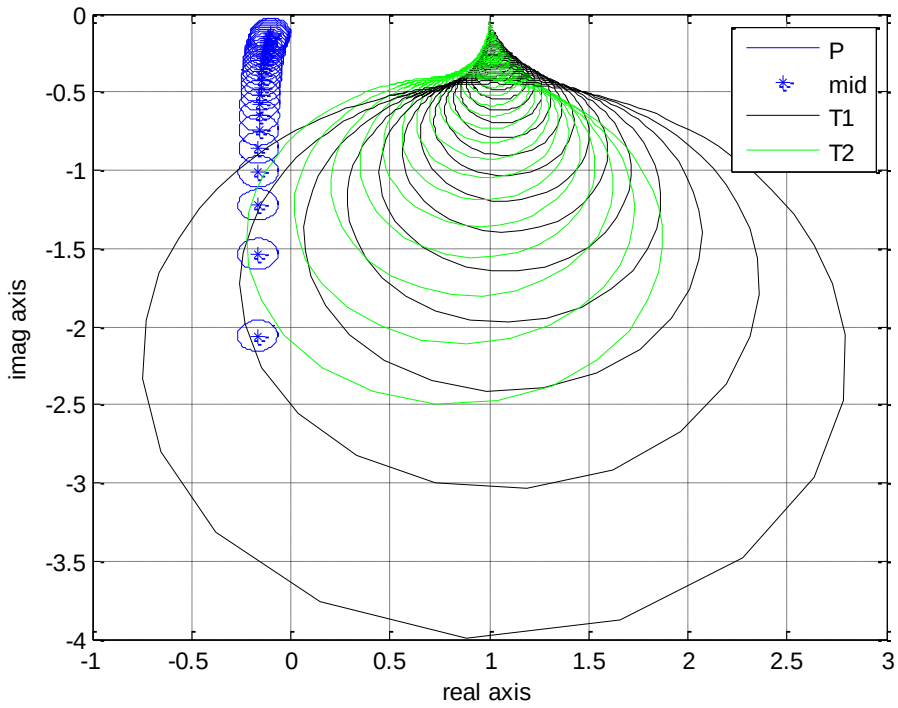


Figure 5.11: uncertainty circle's for f=0.3-3 HZ

5.5 Roll and pitch model estimation

After viewing several ways to examine the behavior of the system, in order to estimate the open loop transfer function, we decided to choose the pulse method.

Repeated tests has been made in order to check the model parameters and the followings

- The existence of integrator in our system.
- The model's order.

5.5.1 Model order

First steps of the estimation of the model has made with 4th order model. The idea is to find the minimal order of the open-loop transfer function using the close-loop measurements.

Because the open loop tf is reconstructed from the closed loop we choose to use the simplest controller that we can use and that is still stabilizing the system.

We used parametric search where the goodness of fit is the norm of the difference between the measured data and the simulation data. The next step, after evaluating the open loop transfer function parameters is to use the 'balreal' function from **matlab**that gives us the information about the weights of the mods in the model. Repeated test's results gave us the minimal model to be from 2nd order.

We can see from figure 5.12 the difference between 2nd order and 4th order. Differences are neglected.

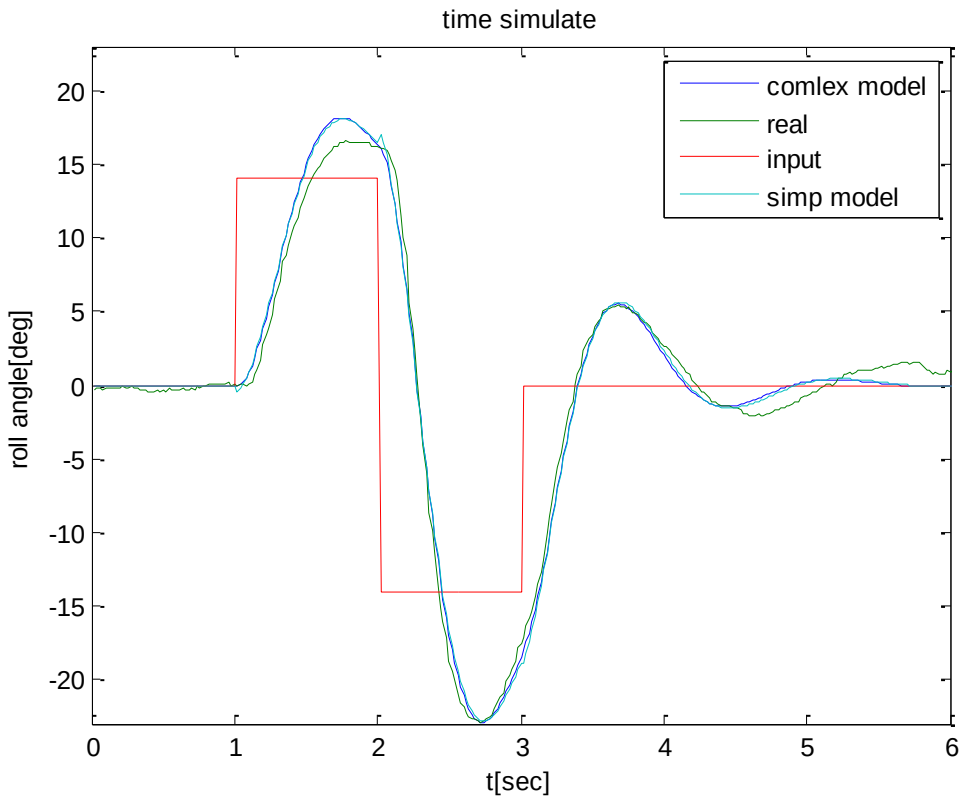


Figure 5.12: comparison between second and fourth order model response

From the Root Locus diagram figure 5.13 we verify that the fourth order model has two pole non relevant to the model estimation.

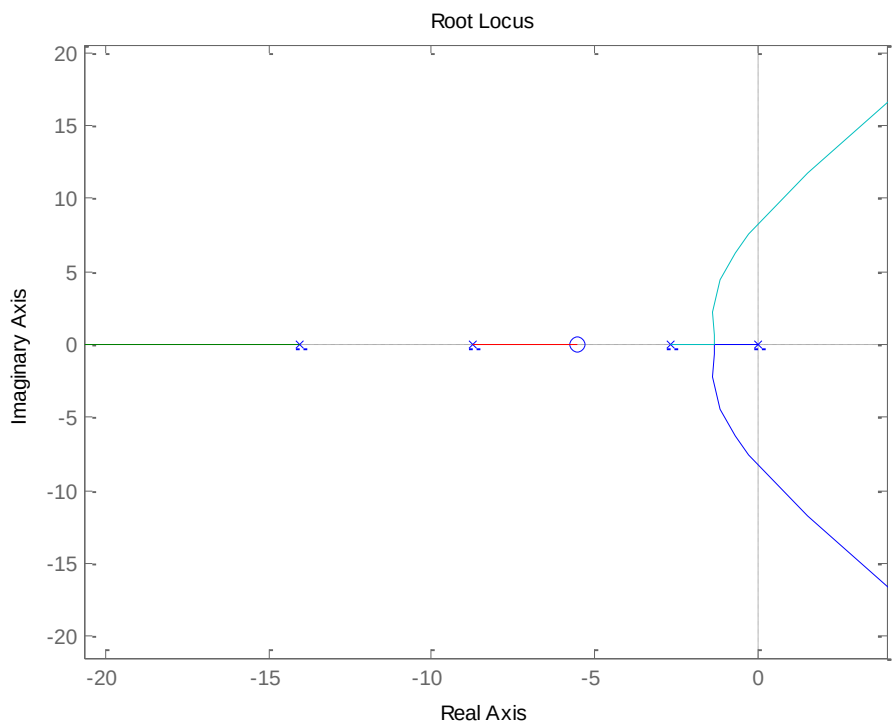


Figure 5.13: root locus of 4 order model

5.5.2 Integrator verification in the model

In order to check the existence of an integrator in the transfer function, we determined the transfer function to be from 2nd order with 2 poles.

$$P = \frac{1}{(s+\alpha)(s+\beta)}$$

The same method of parameter estimation was used here in order to identify the two parameters.

We can see the two poles in figure 5.14.

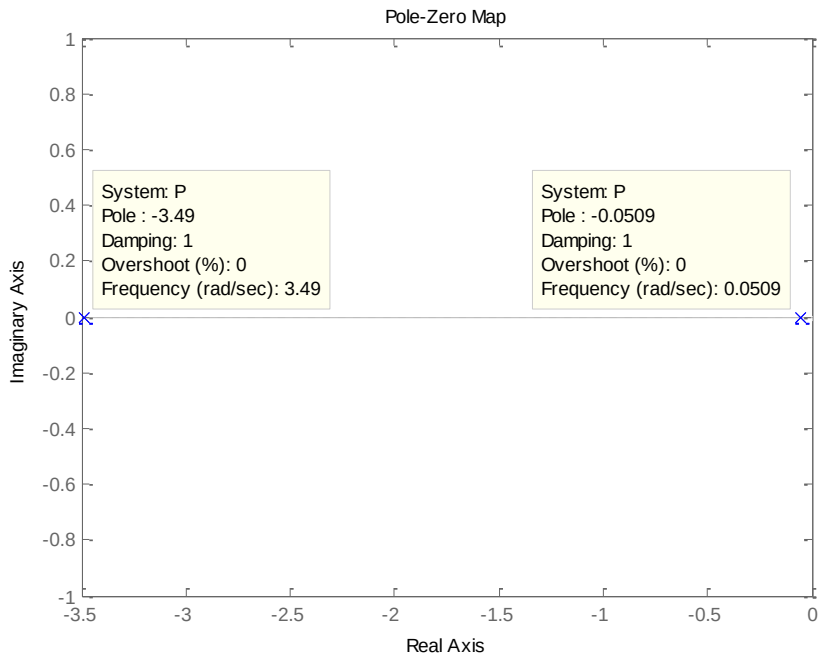


Figure 5.14: pole-zero map of second order model

Clearly we can see the one pole is 70 times bigger than the other, there for we can determine that there is an integrator in the model.

Our ratio between the poles is $\frac{p_1}{p_2} = \frac{3.49}{0.05} \approx 70$

The result that there is an integrator in the model is not surprising , because from the model physics we can estimate an integrator in the open loop, and another hint for the existence of integrator in the systems is the fact that the open loop transfer function is a non-stable transfer function.

5.5.3 Zeros in the model

At last before we determine the model shape, we need to decide how many zeros we have in the system, that why we run a test that allow to the model to be with two zeros. as we seen from the poles zeros map figure 5.15 the system has non dominant two zeros because they far away from the imagine axis.
the closest one is locate at -34 , we can't say for certain that these zeros do not exist in the model, but these zeros can been ignored because he isn't relevant for our purpose in modeling the system, we don't intent to control the system in such Bandwidth.

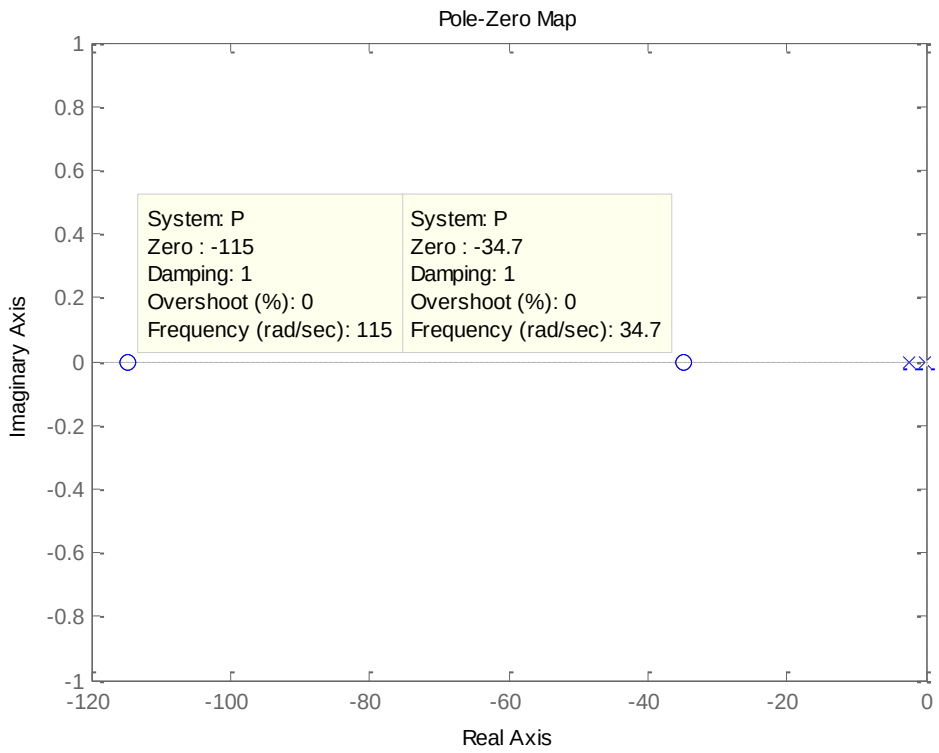


Figure 5.15: pole-zeros map of second order model with 2 zeros

Model shape

Finally we can determine the model form; we decided that the model is from the form of:

$$\boxed{P(s) = \frac{K}{s(s+\beta)}}$$

Again, the same algorithm of parameter search has made in order to estimate the

K, β Parameters

5.5.4 Model identification parameters

Test 1

Graphs:

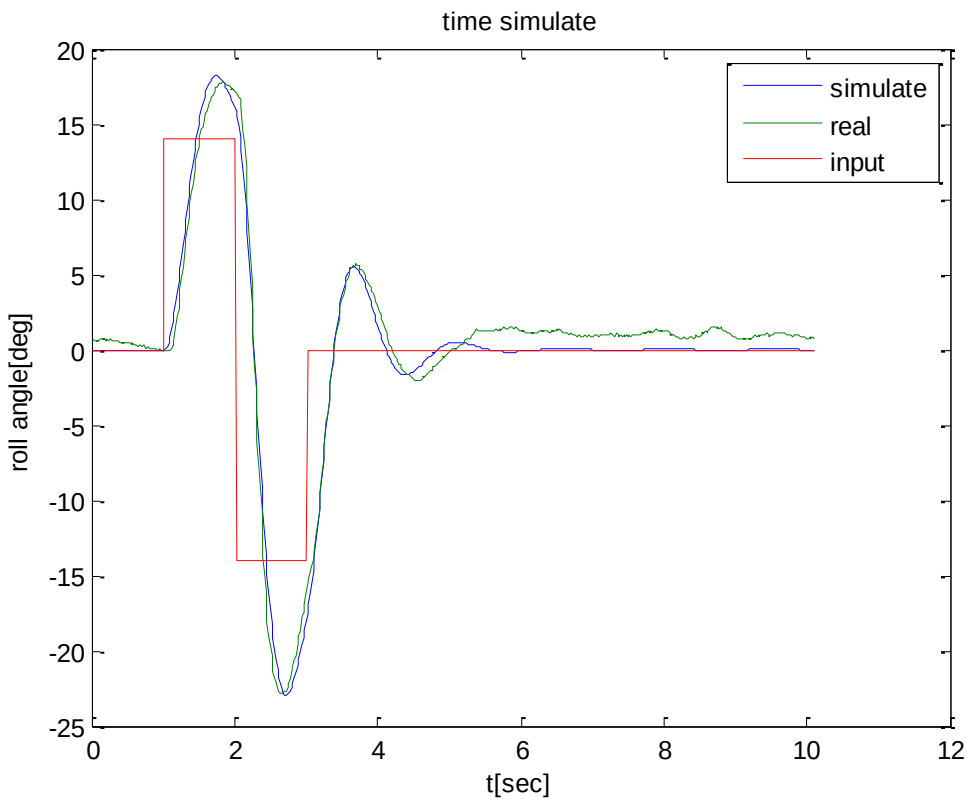


Figure 5.16: pulse response

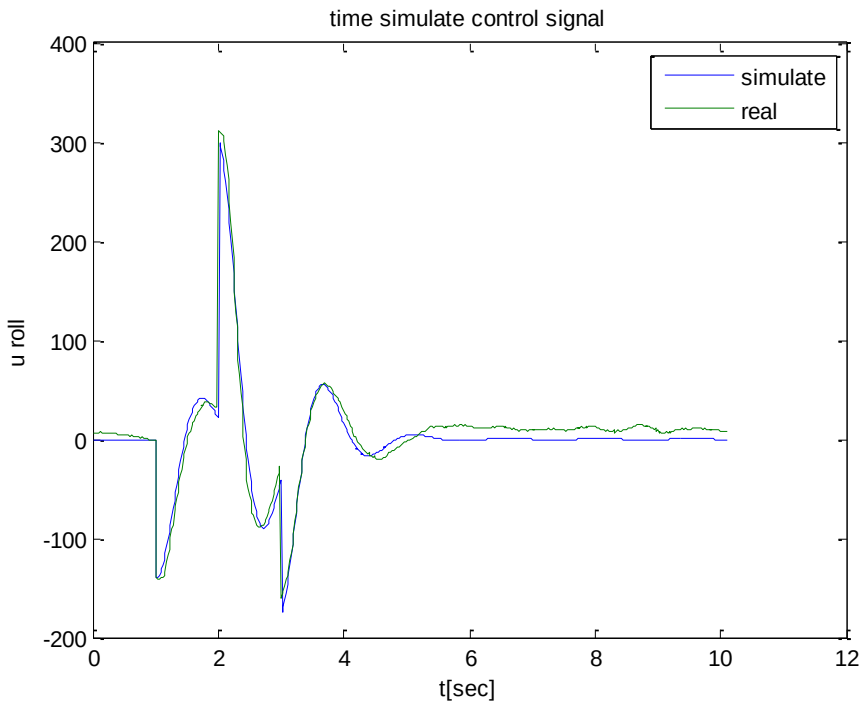


Figure 5.17: control signal to pulse response

Test 2

Graphs:

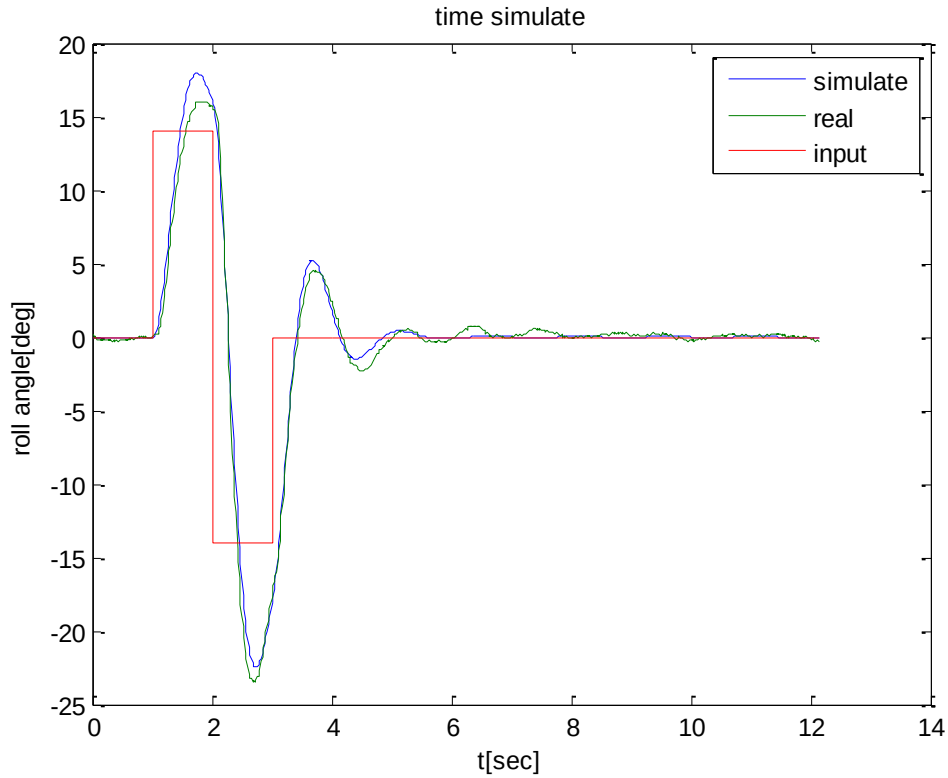


Figure 5.18: pulse response

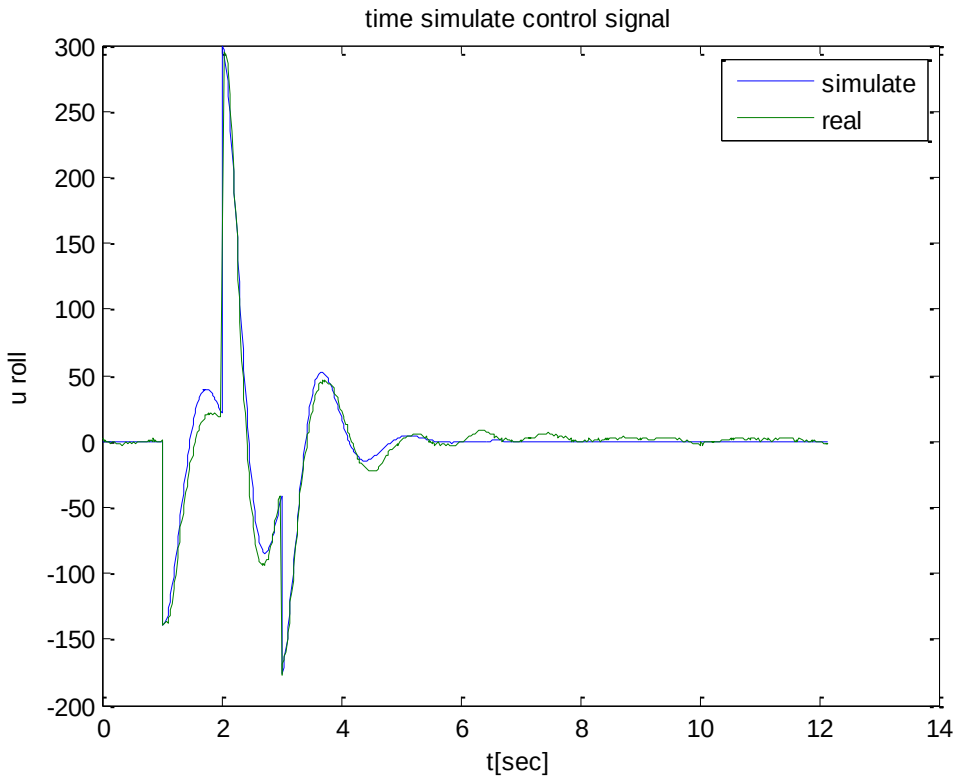


Figure 5.19: control signal to pulse response

Test 3

Graphs:

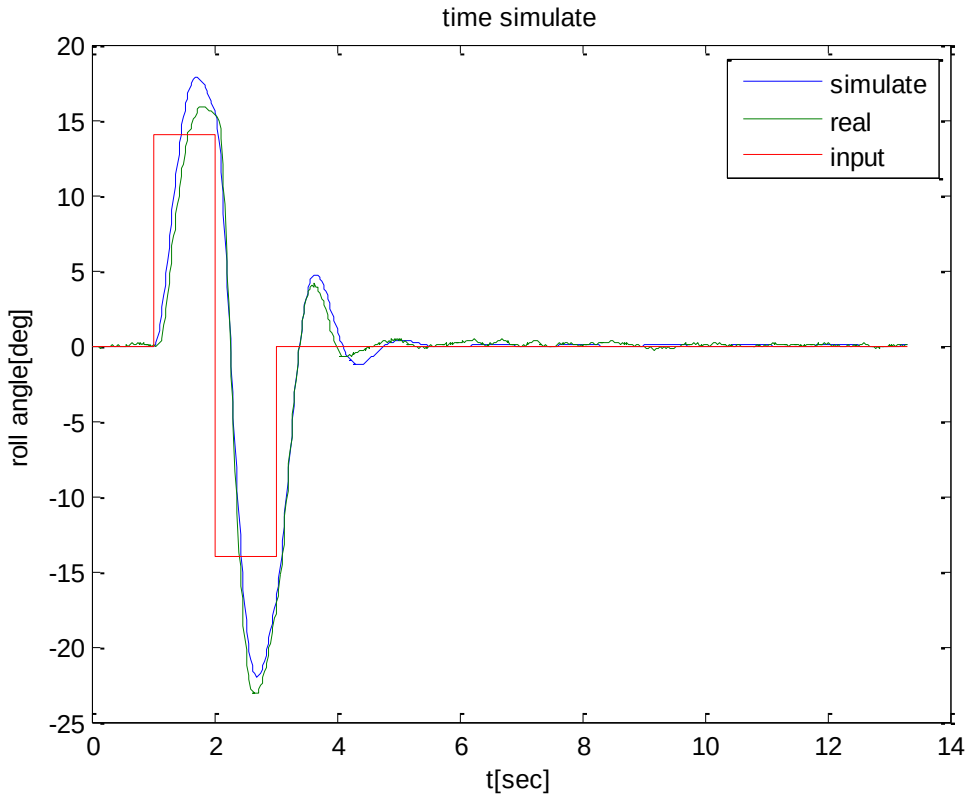


Figure 5.20: pulse response

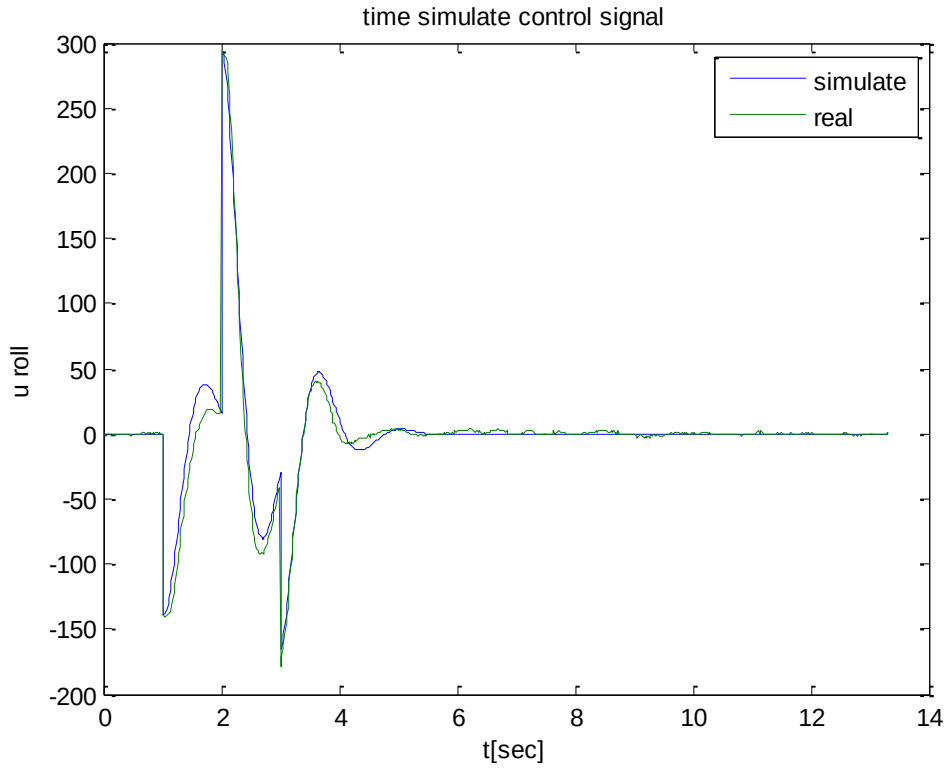


Figure 5.21: control signal to pulse response

Test 4

Graphs:

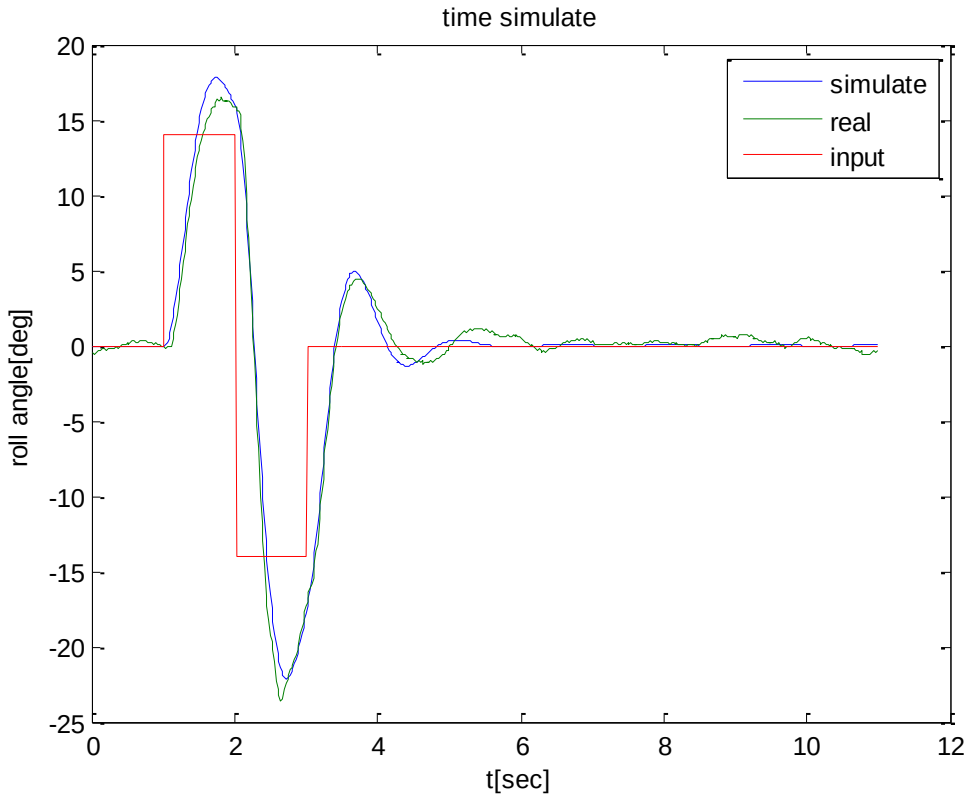


Figure 5.22: pulse response

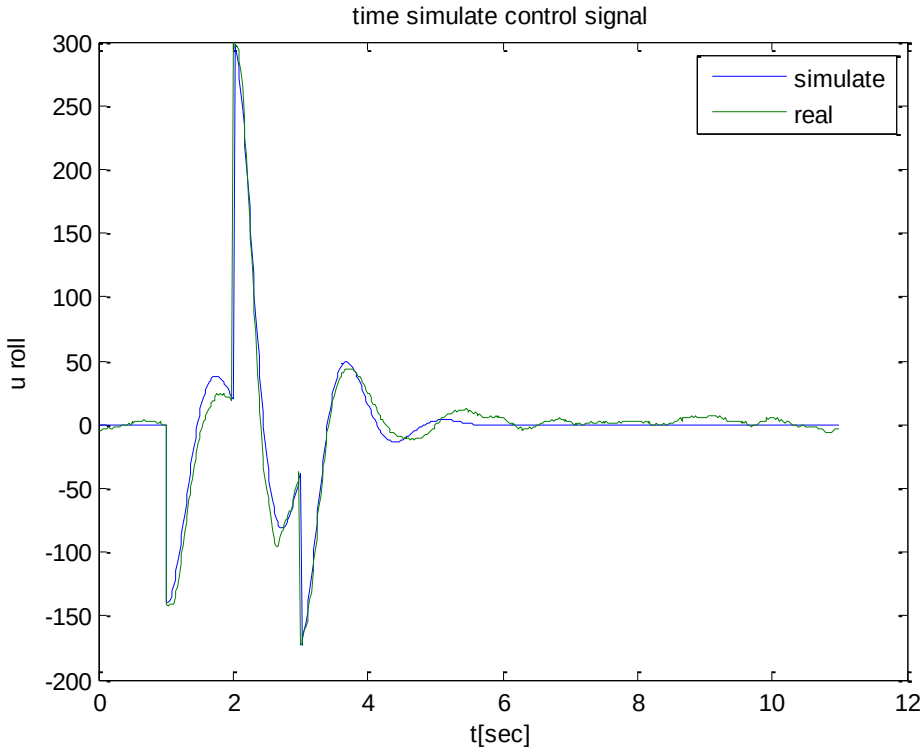


Figure 5.23: control signal to pulse response

Summary table

Test	Chosen controller	Fit percentage
1	2.1708 ---------- s (s+3.348)	0.85
2	2.1476 ---------- s (s+3.461)	0.89
3	2.3466 ---------- s (s+3.702)	0.87
4	2.1994 ---------- s (s+3.591)	0.88

Table 5.2: our entire chosen controller in the tests

Final result

$$P(s) = \frac{2.2157 \pm 0.13}{s\left(s + \left(3.525 \pm 0.1775\right)\right)}$$

Bode robust diagram

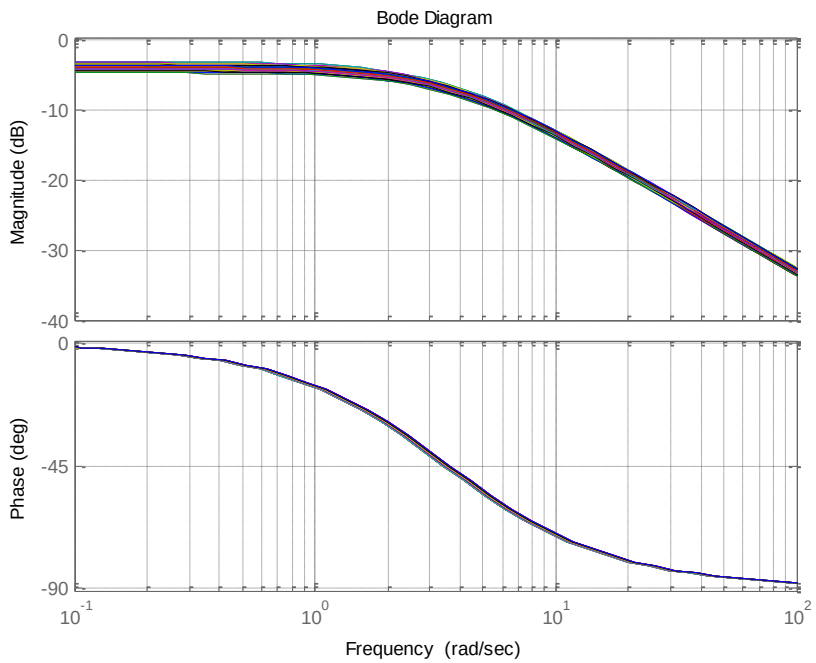


Figure 5.24: bode diagram of robust process

Nichols robust chart

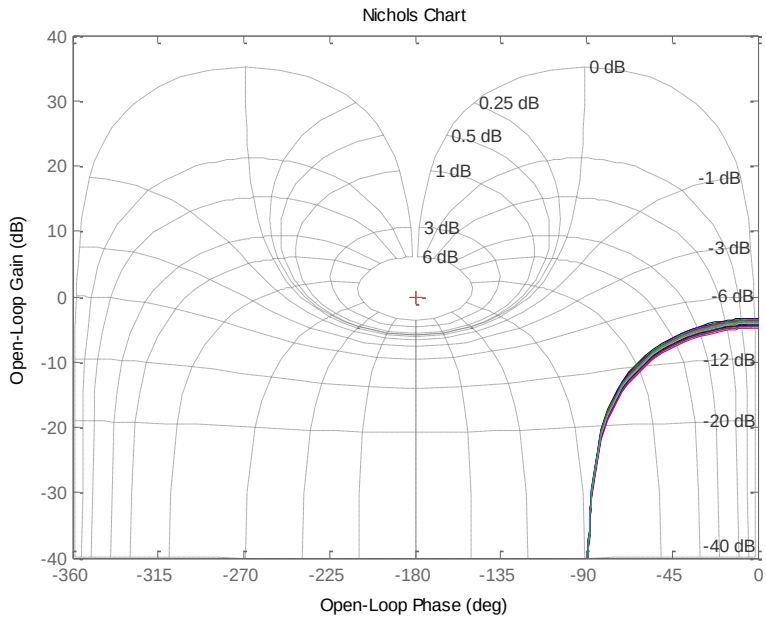


Figure 5.25: Nichols chart of robust process

We can see from our result that we receive second order transfer function with one integrator.

This transfer function isn't a complex transfer function and can be stable with only proportional controller.

Later on we stable this process with a PID controller.

5.6 Yaw model estimation

Model estimation of yaw channel is simpler then roll or pitch, and that is because step reference is possible without any concerns for the system (unlike pitch \roll step reference which results movement of the quadrotor towards the boundaries of the room).

Our estimation of the yaw open-loop transfer function is a 2nd order model. The question is whether we have integrator in the system or not.

We choose proportional controller P=5, and simulate with the rference input of 60 degrees.

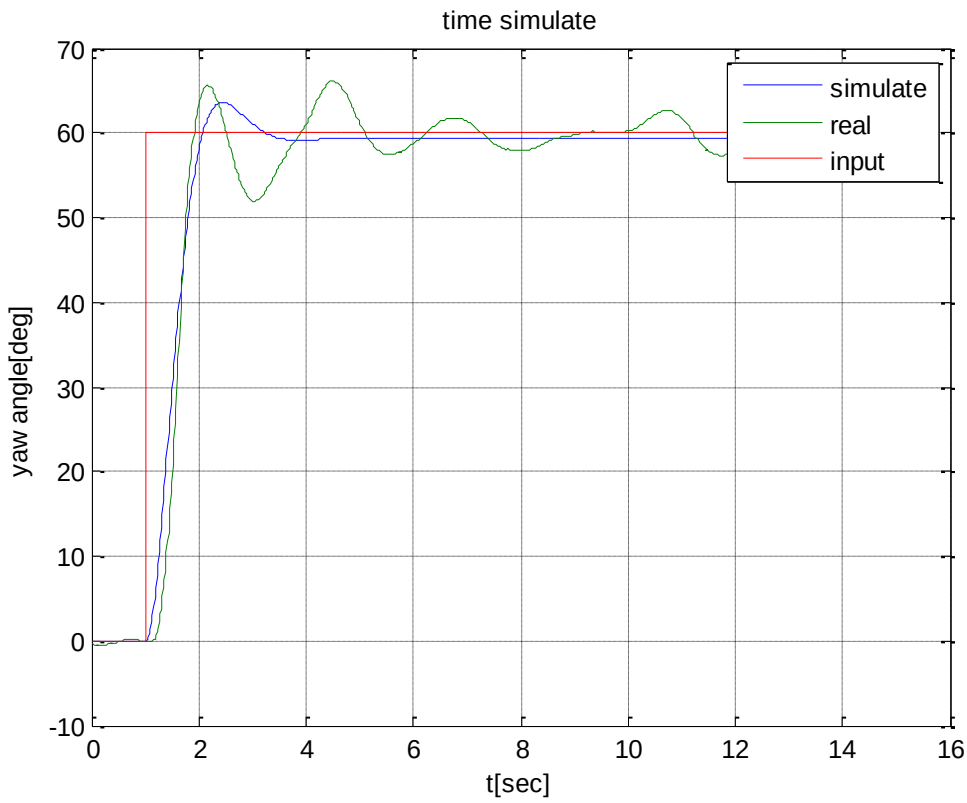


Figure 5.26: step response of model without integrator

With parameter search we found the open loop to be:

$$P_{yaw} = \frac{1.4471}{(s+3.105)(s+0.0212)}$$

We can see that one pole is very close to zero. There for we shell examine a model with integrator.

Additional tests has mad in a different input reference (30 degrees)

Test 1:

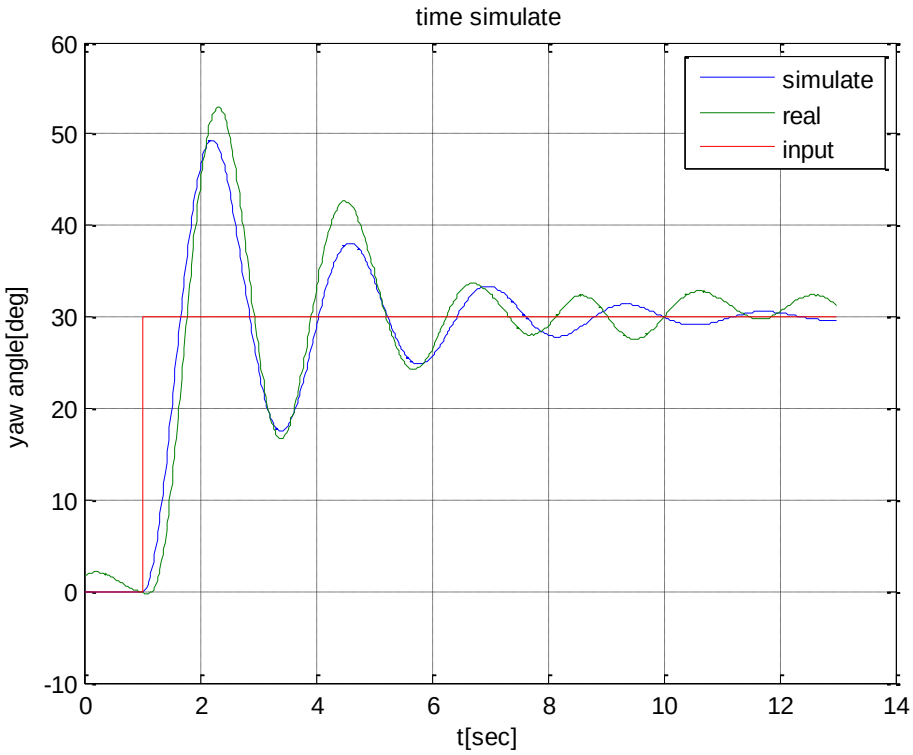


Figure5.27: step response

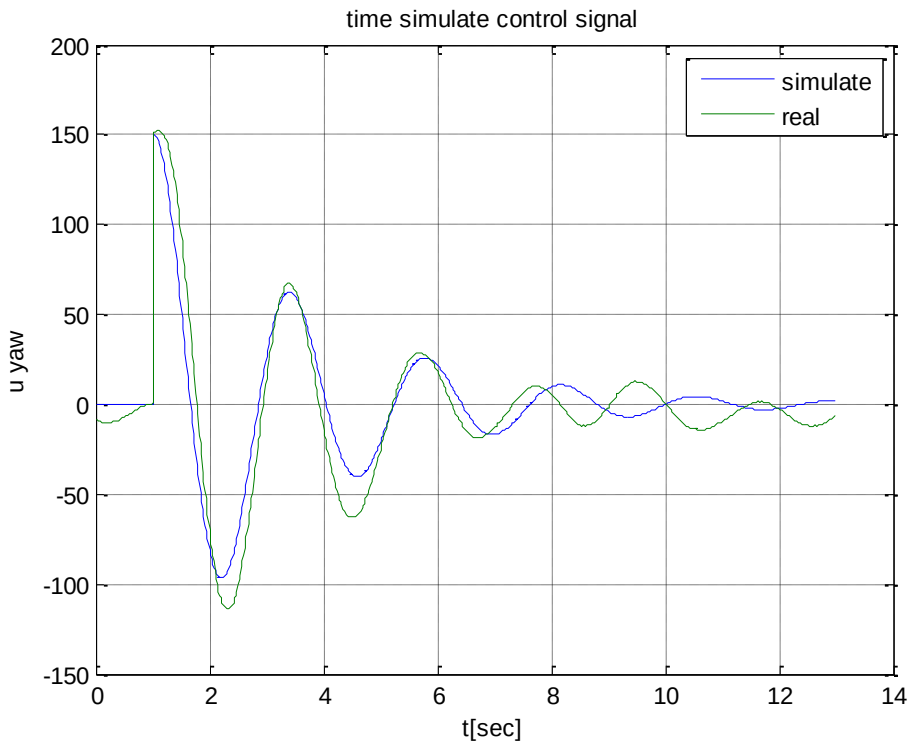


Figure 5.28: control signal to step response

Test 2:

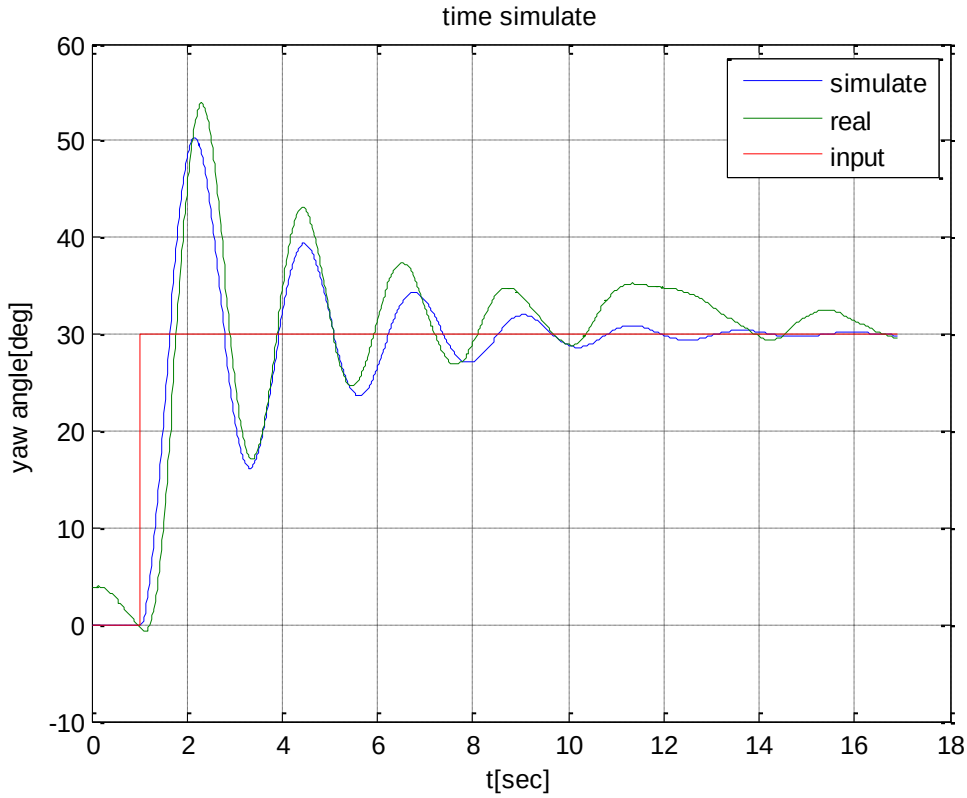


Figure 5.29: step response

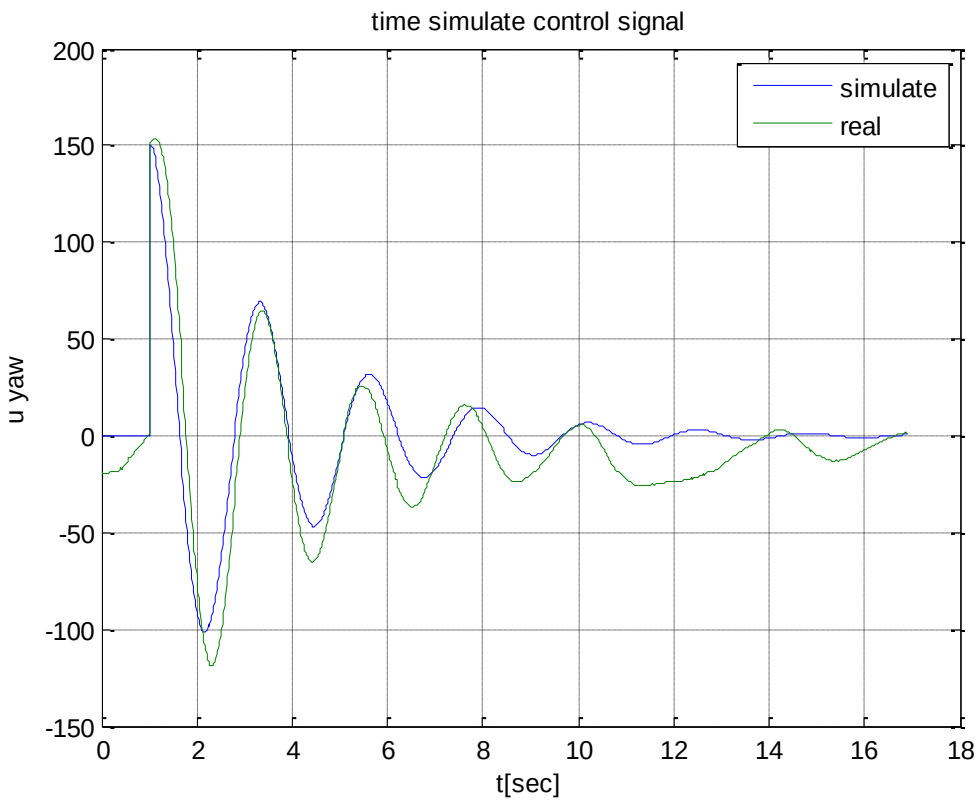


Figure 5.30: control signal to step response

Test 3:

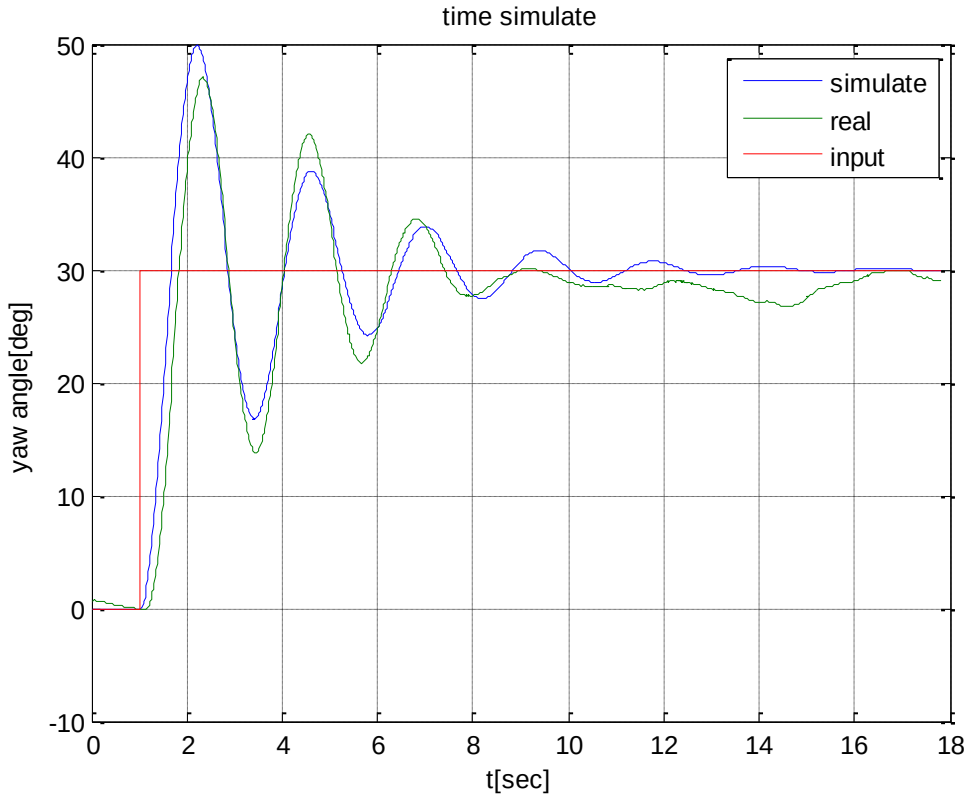


Figure 5.31: step response

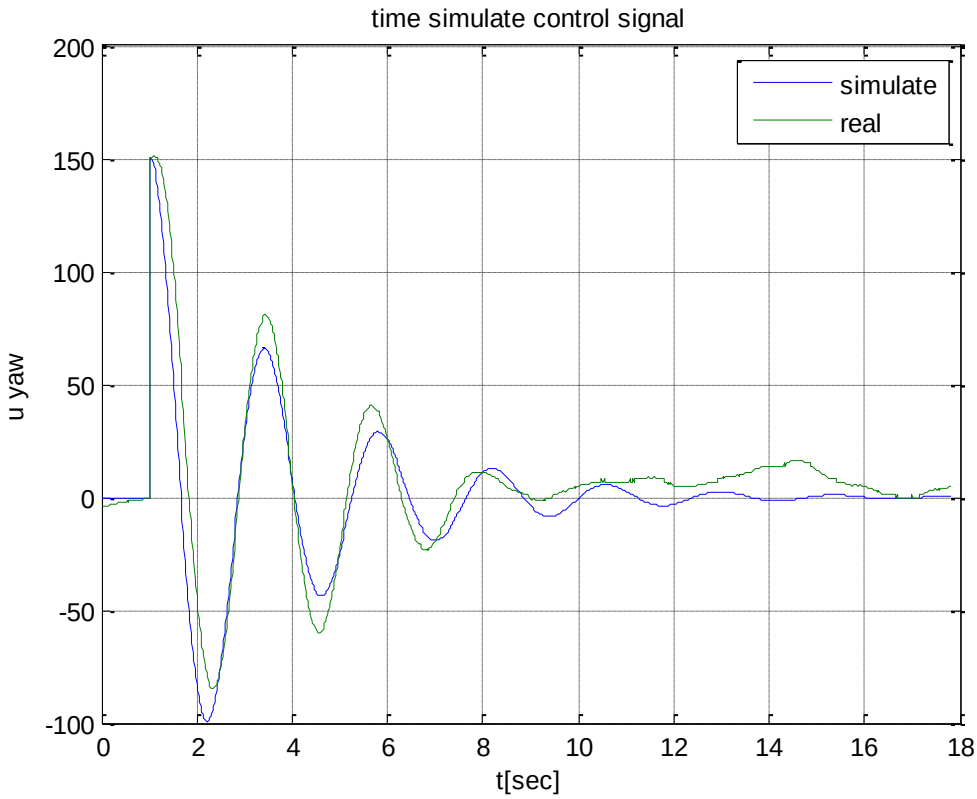


Figure 5.32: control signal to step response

Test 4:

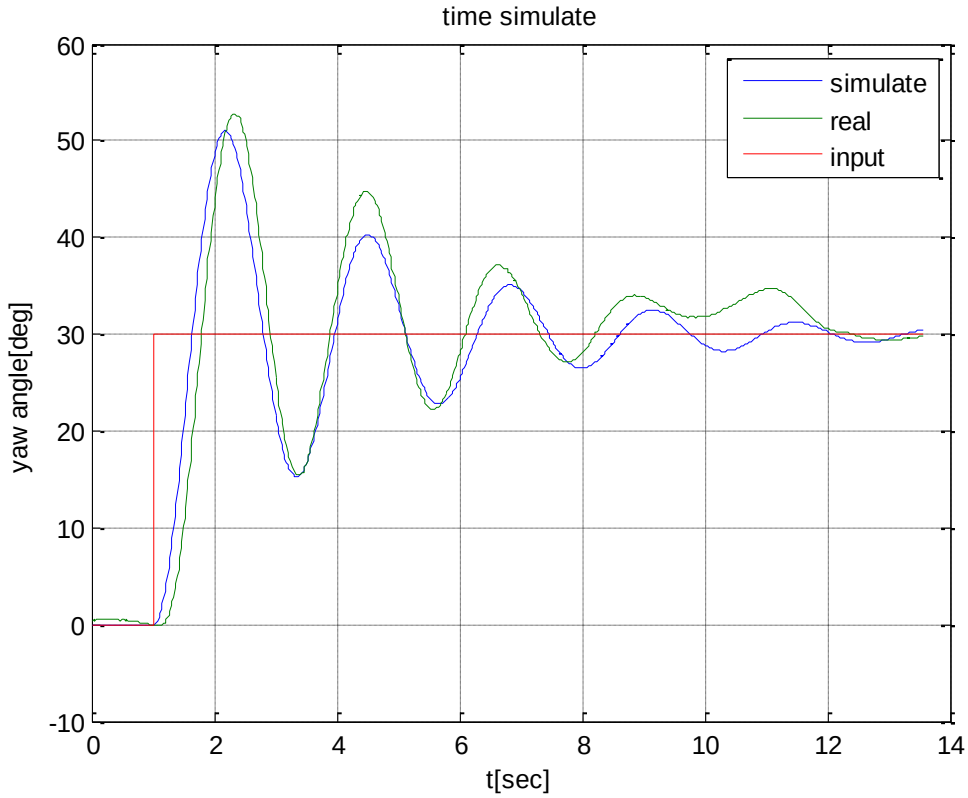


Figure 5.33: step response

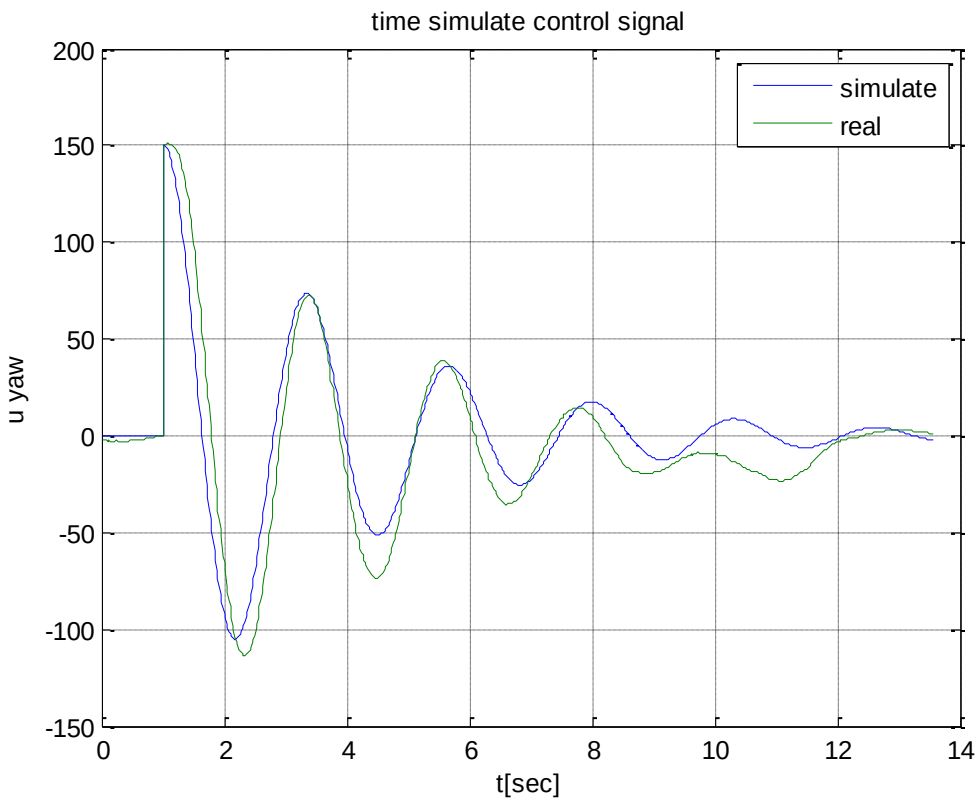


Figure 5.34: control signal to step response

Summary table

Test	Chosen controller	Fit percentage
1	1.4139 ---------- s (s+0.7351)	0.914
2	1.5203 ---------- s (s+0.6739)	0.88
3	1.4042 ---------- s (s+0.6863)	0.91
4	1.4809 ---------- s (s+0.6129)	0.88

Table 5.3: our entire chosen controller in the tests

Our process eventually after several tests:

$$P_{yaw} = \frac{1.45 \pm 0.1}{s\left(s + 0.67 \pm 0.1\right)}$$

We can see that model with integrator is better than the model without.

Bode robust diagram

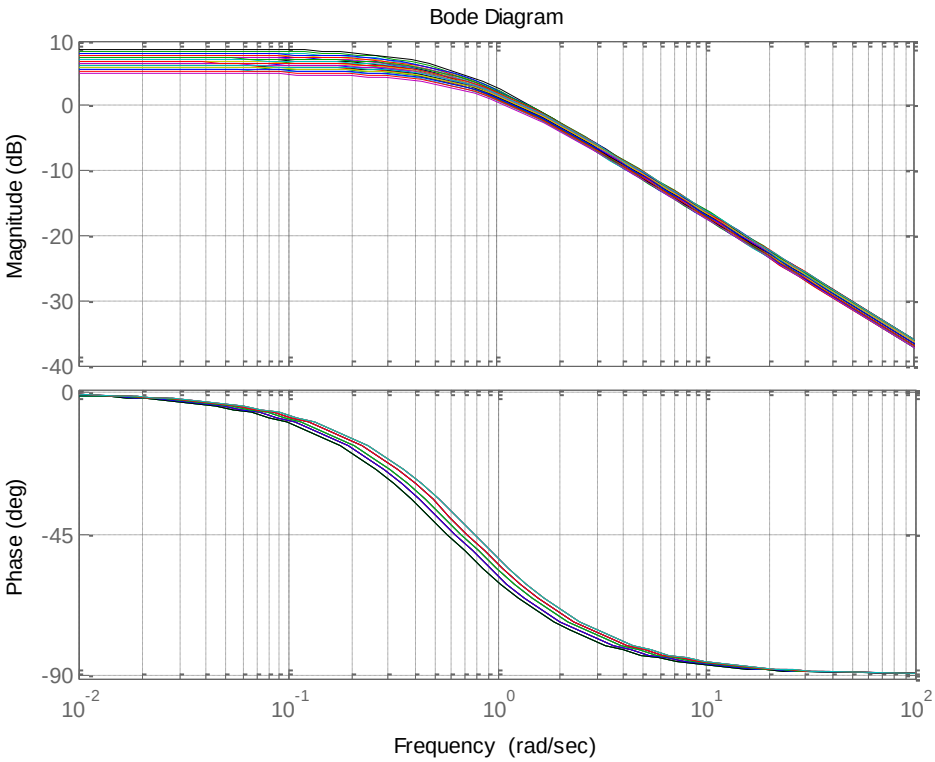


Figure 5.35: bode diagram of the robust process

Nichols robust chart

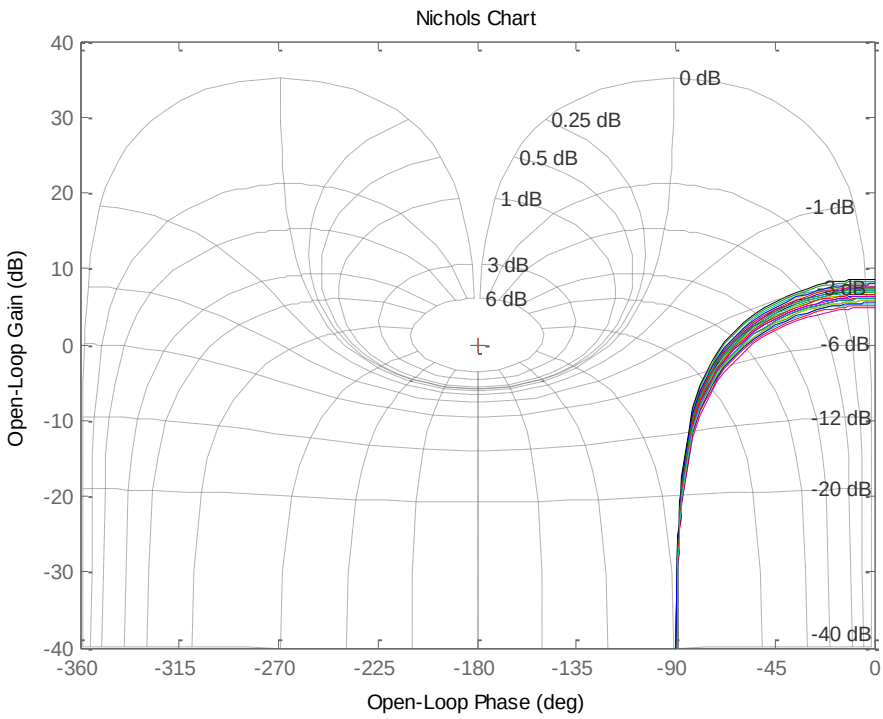


Figure 5.36: Nichols chart for our robust process

5.7 Coupling between roll and pitch

In our system there are four inputs (pitch_cmd, roll_cmd, yaw_cmd, throttle) and four outputs (pitch, roll, yaw, altitude).
Our assumption is that there must be coupling between the difference inputs and outputs.
In order to examine those coupling effects, we perform a test that we set a reference signal only in the roll channel and we examine the pitch angle in order to see the correlation between those channels.

We use a similar proportional controller for both of this channels c=7;
in order to examine this correlation we first need to isolated the coupling transfer function, as we can see in figure 5.37 we try to find a correlation between the roll reference signal and the pitch output signal.
That scheme is reduced scheme in order to simplify the calculation the correlation.

In that scheme we neglect the transfer function of $P_{roll/pitch}$ because we assume that the roll output signal that came as outcome from control signal of pitch is very small, this is why we can neglect him.

Of course we see that in this scheme we neglect the yaw and throttle coupling for the same reason.

In this test we focus on the transfer function from u_{roll} to y_{pitch}.

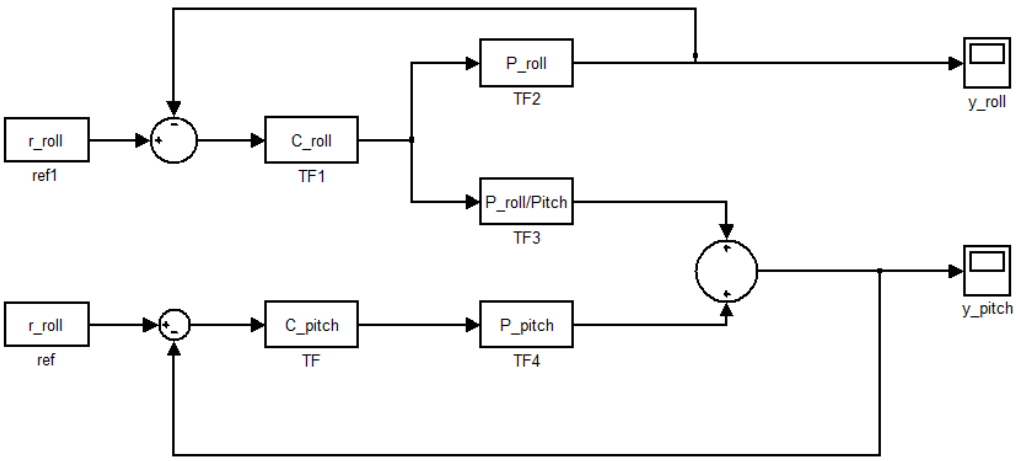


Figure 5.37: reduced scheme of pitch and roll channels

With the assistants of the scheme we will receive the transfer function between roll reference signal and pitch output signal:

$$y_{pitch} = u_{roll} \cdot P_{pitch/roll} + \begin{pmatrix} r_{pitch} - y_{pitch} \\ 0 \end{pmatrix} \cdot C_{pitch} \cdot P_{pitch}$$

$$u_{roll} = \frac{C_{roll}}{1 + C_{roll} \cdot P_{roll}} \cdot r_{roll}$$

$$y_{pitch} \cdot \left(1 + C_{pitch} \cdot P_{pitch}\right) = u_{roll} \cdot P_{pitch/roll} = \frac{C_{roll}}{1 + C_{roll} \cdot P_{roll}} \cdot r_{roll} \cdot P_{pitch/roll}$$

$$\frac{y_{pitch}}{r_{roll}} = \frac{\frac{C_{roll}}{1 + C_{roll} \cdot P_{roll}} \cdot P_{pitch/roll}}{\left(1 + C_{pitch} \cdot P_{pitch}\right)}$$

After we received the required transfer function we perform the test we can see in figure 5.38 the response of pitch angle to a reference in roll.

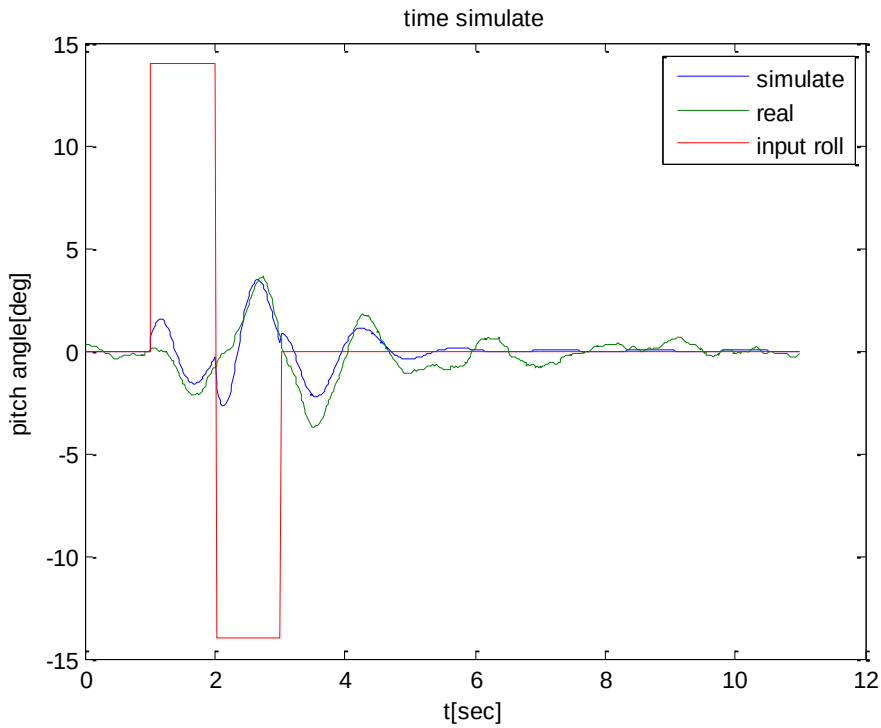


Figure 5.38: pitch response to roll reference

From this plot we shell try to found a correlation between the two channels.

Our best correlation is second order transfer function

$$\frac{-0.0046391\,(s+22.78)\,(s+11.88)}{(s+7.01)\,(s+6.727)}$$

- From this transfer function we see that the k product is a very small one in relative to other transfer function in the system.
- This tf is contras to the input signal, because the negative product in k.
- This coupling tf contain a poles in 6-7 rad/sec , it mean as long as we don't set the system to work on fast modes(higher frequency), this coupling effect is irrelevant to us.

5.8 Summary

- Our theoretic model predictionof all channels contains one integrator and one stable pole at least.
- Several methods of modeling systems exists, but for us, only the step and pulse method are practical, because of our boundaries and limitations in our testing laband our sensors.
- As we modeled the roll/pitch process we conclude that this process is indeed a second order process and contains one integrator.
- The process was verified in several tests and showed appropriate fit to the simulations.
- Yaw process also contain an integrator, we can see this following our tests.
- Second order model for yaw process, gives us an excellent fit between the response and the simulations (95% fit).
- From roll input we saw that there is coupling between the channels.
- The poles and the zeros of the coupling transfer function are much higher from those of the pure process, that's why we can prefer the system as SISO at lower frequency usage.
- The system is considered symmetrical in bots x and y axis, there for we consider pitch and roll transfer functions to be identify.

6 Orientation control

In this chapter our goal is to study the fastest controller that the system can endure.

The method of finding this controller is by inserting a PID controller with different bandwidth and analyzing the system response. The process will end when the system response becomes on the verge of stability.

It means the last controller that we found before instability represents the fastest response of the system.

Our objective is to choose a controller that will fast enough and provide reasonable response and will be robust enough.

In our test we examine the system response and compere this response to our simulate response that we achieve from knowing the process.

Our controller form is: $C(s) = K_p + {K_I}/{s} + K_d \cdot s$

6.1 Pitch and roll controller

As aforesaid, our assumption is that pitch and roll channels have identical process. Because of that we design a similar controller for both of them.

Bandwidth: 2.5 rad/sec

Parameters: P=4.75 I=0.216 D=0.213

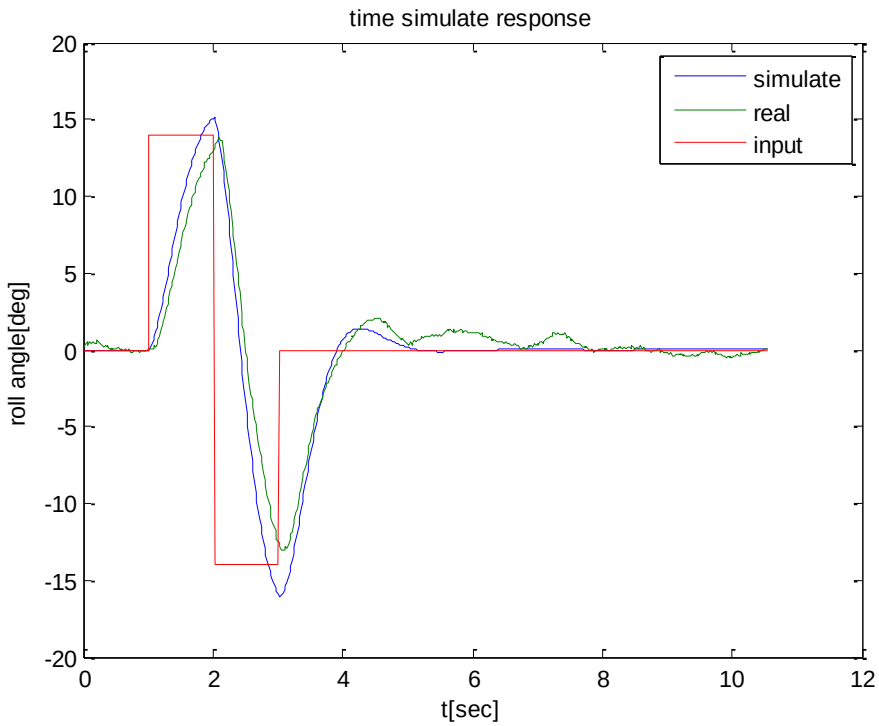


Figure6.1: response of controller with bandwidth of 2.5 rad/sec

Bandwidth: 3 rad/sec

Parameters: P=6.253 I=0.328 D=0.417

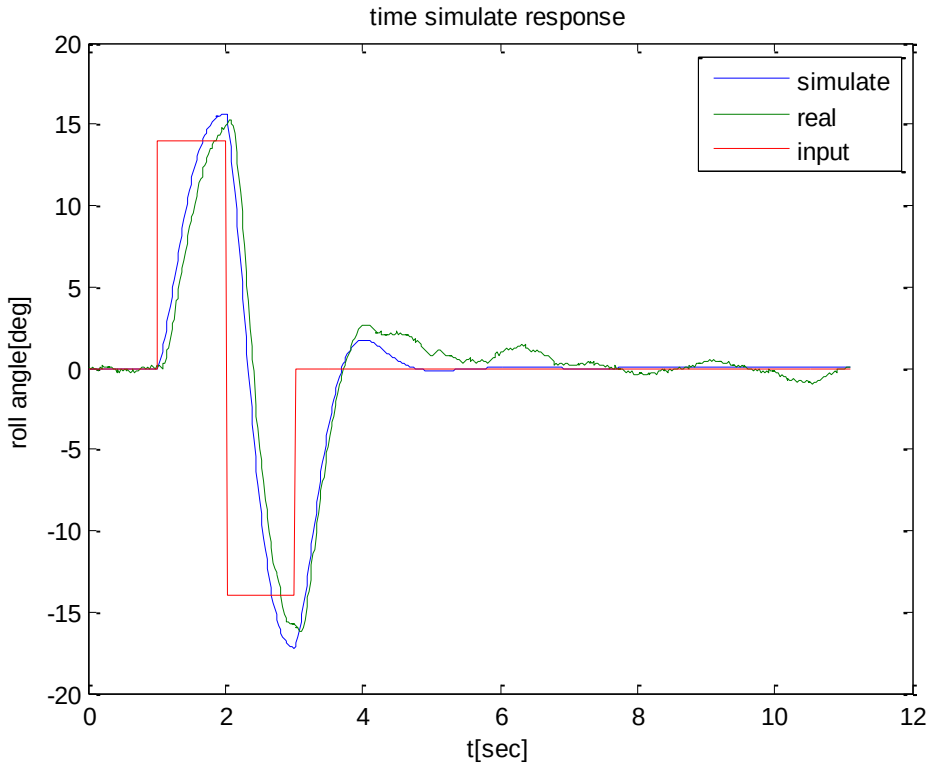


Figure 6.3: response of controller with bandwidth of 3 rad/sec

Bandwidth: 4 rad/sec

Parameters: P=9.09 I=0.06453 D=0.804

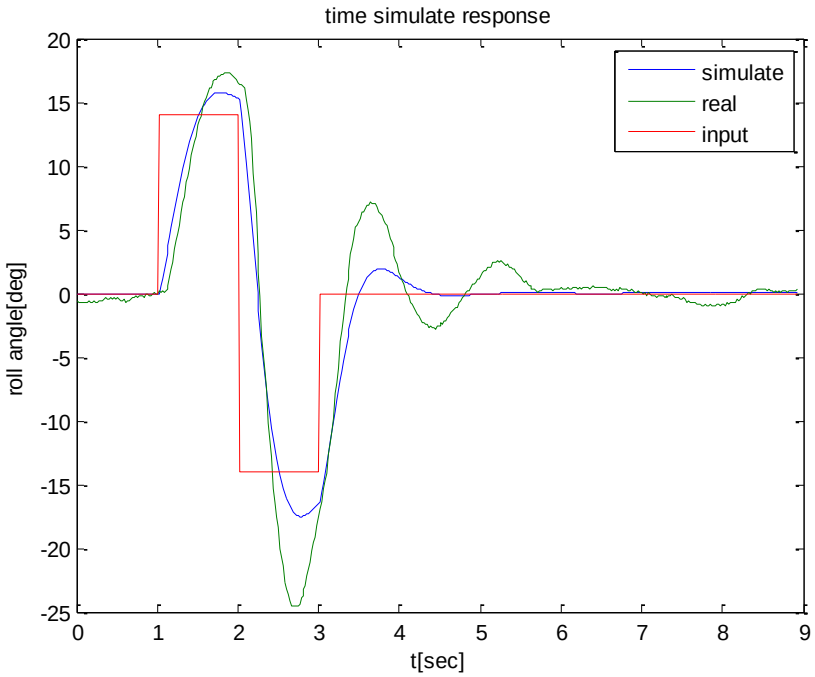


Figure 6.4: response of controller with bandwidth of 4 rad/sec

Bandwidth: 5 rad/sec

Parameters: P=12.2688 I=1.1056 D=1.1796

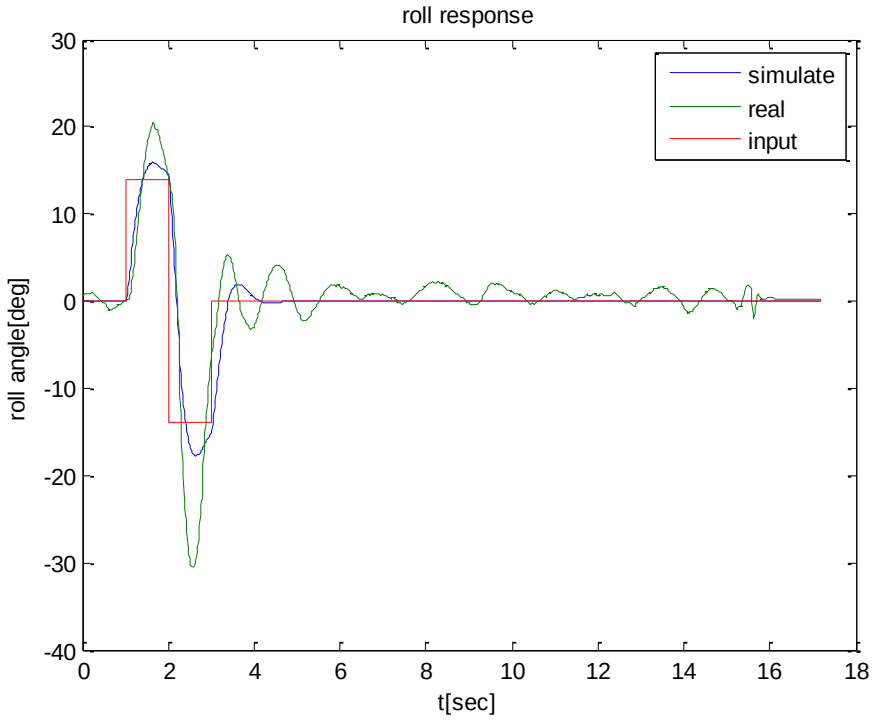


Figure 6.5: response of controller with bandwidth of 5 rad/sec

Bandwidth:5.5 rad/sec

Parameters: P=14.1028 I=1.3966 D=1.3686

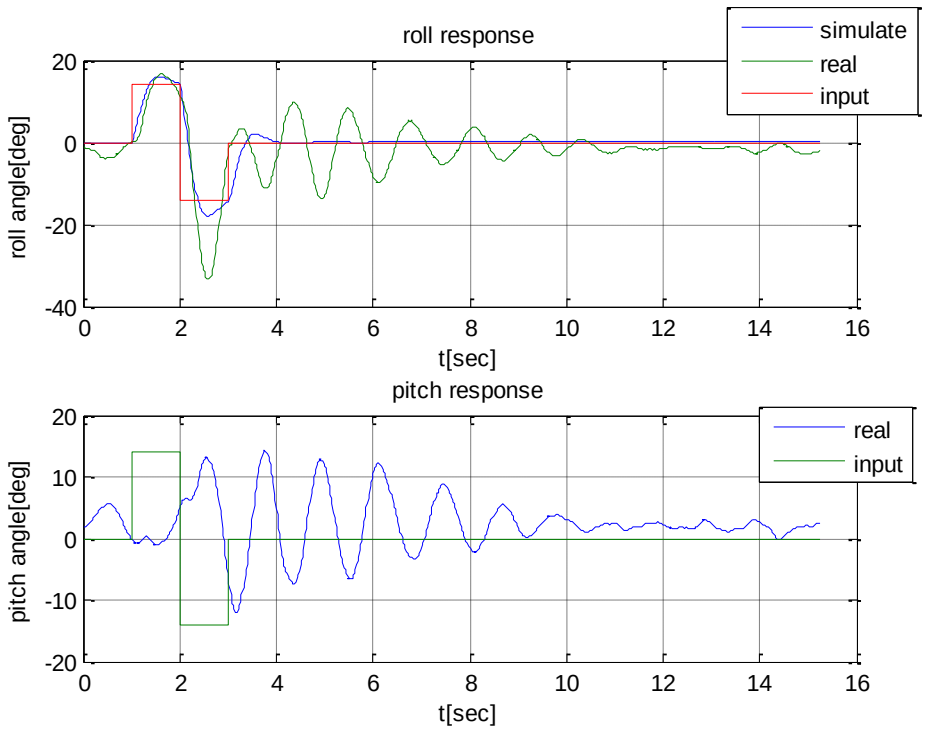


Figure 6.6: response of controller with bandwidth of 5.5 rad/sec

6.1.1 Conclusions

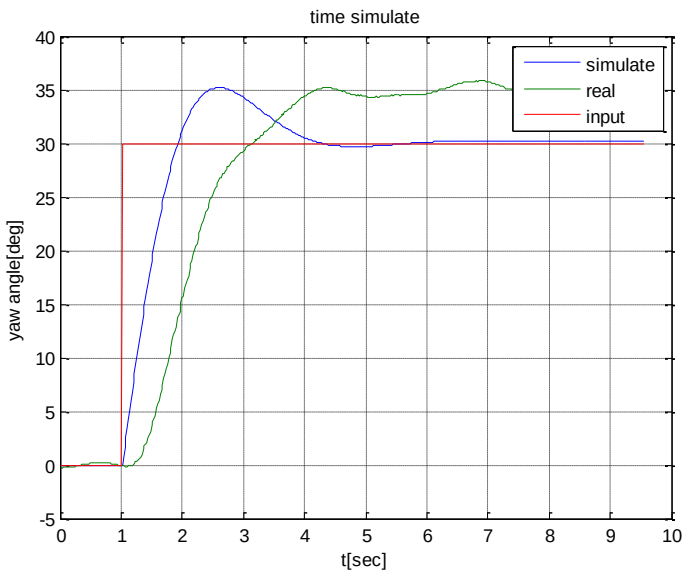
- As we increase the controller bandwidth our system response to the given input becomes faster
- As we increase the controller bandwidth our simulation is less accurate: overshot is higher, and an oscillation starts.
- In frequency of 5.5 rad/sec the system becomes on the verge of stability and we can see well the coupling between pitch and roll.
- Because the transfer function between roll and pitch have a poles and zeros with higher frequency, when we modeled the process, they were irrelevant, but now as we increase our system frequency those zeros and poles start to excited and changing our predicted response.
- In order to achieve large bandwidth we must refer the system as MIMO (multiple input multiple output) system, at those higher frequency we can't neglect the coupling effect.
- Our chosen controller is the controller with bandwidth of 3 rad/sec, because of the fast response and small oscillation.
- $C_{chosen}(s) = 6.253 + {0.328}/{s} + 0.417 \cdot s$

6.2 Yaw controller

Similarly to roll controller we shell stable the yaw process with a PID controller.
As our bandwidth increase until we receive instability.

Bandwidth: 2 rad/sec

Parameters: P=2.17 I=0.076 D=0.98



We can see that clearly this isn't a good result; our assumptionto this is because this controller is a very slow one, and it is also very sensitive to interruption.
As aforesaid, the yaw measure is the less accurate from all of Euler angels because he doesn't have a compensator like the others.

Figure 6.7: response of controller with bandwidth of 2rad/sec

Bandwidth: 3 rad/sec

Parameters: P=4.36 I=0.2252 D=1.62

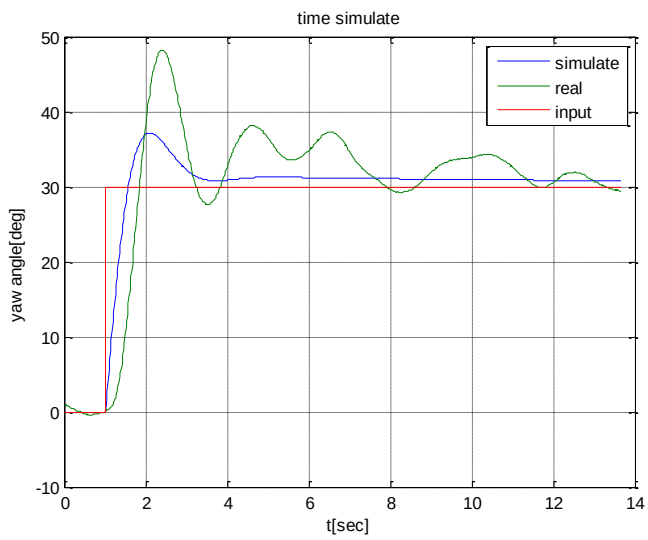


Figure 6.8: response of controller with bandwidth of 3 rad/sec

Bandwidth: 4 rad/sec

Parameters: P=7.09 I=0.495 D=2.12

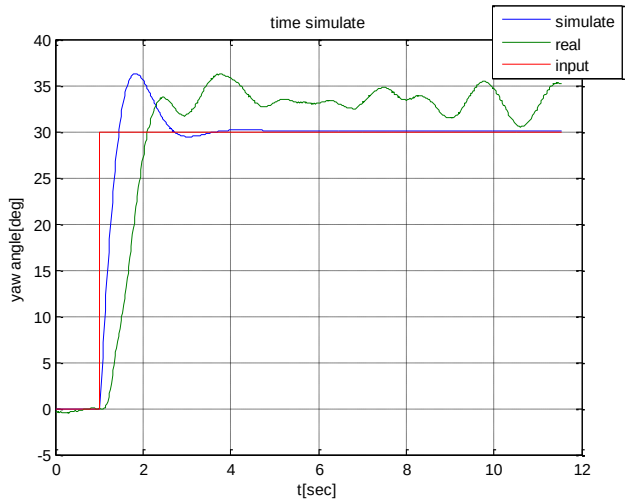


Figure 6.9: response of controller with bandwidth of 4 rad/sec

6.2.1 Conclusion

- We can see from all of the figures that the yaw angels is not coverage to its target value.
- The measure of yaw angles is not entirely accurate.
- Because of the bed measurement we can't verify if our model is correct.
- Another assumption is because we use a PID controller we cause to the frequency of the signal to increase across the limit that our sampling time is allows us, and by that our sensor is cant measure an accurate result.
- In addition we shall examine the effect of coupling, perhaps the PID controller cause to excite of the coupling effect.
- The helicopter is getting power from a wired cable. There for when preforming tests like step the power cable twists and results torque in the opposite direction to the step witch results inaccurate model.
- Our chosen controller is $C(s) = 4.38 + {0.22}/{s} + 1.62 \cdot s$

7 Summary

In this project we had forced various issues:

- Sensors analysis
- developed layout of tests
- Analysis frequency response
- Modeling system
- Handing with uncertainty system
- Un-linearized model
- The effect of measurement noise
- Classic control theory
- The effect of folding frequencies (alias effect)
- MIMO system
- Discrete controller
- Signal analysis

The open loop siso transfer functions that has identified are:

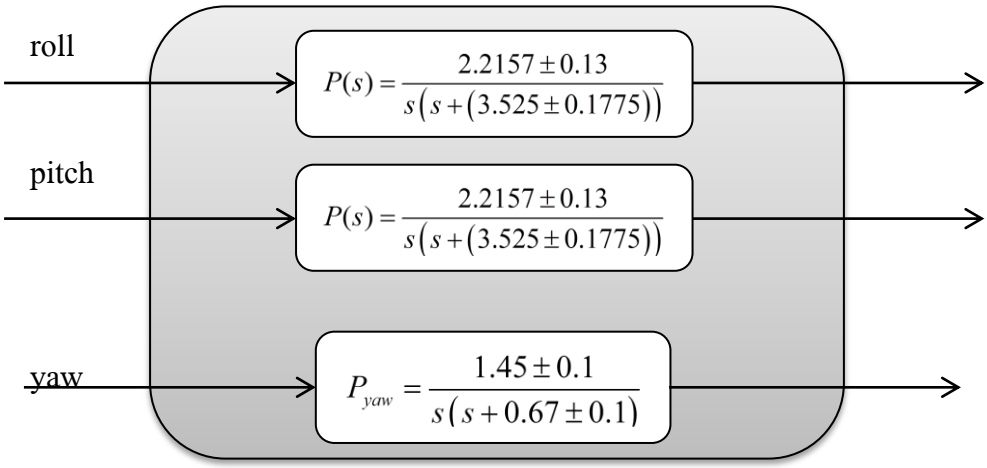


Figure 6.10: The process of all Euler angles

It was a very interesting and complex project because of this varies of issues that we had to understand in order to succeed.

At the end, we proved that this helicopter can be stable and control by simple controller.

The next step for this project will be to inquire the effecting coupling and the non-linearized model.

But in order to that, our experiment system has to upgrade.
We must change all of the wires connection to wireless and substitute the current voltage supply to a battery, and change the control interface.

8 References

[1] Ascending Technologies GmbH. X-3D-BL User's Manual, Germany, 2009.MTi-G User Manual and Technical Documentation

[2] Modelling and Control of Autonomous Group 10833 Faculty of Engineering, Science and Medicine University of Aalborg, Denmark

[3] L. Mirkin. lecture notes - Linear Control Systems

[4] J. G. Leishman, Principles of Helicopter Aerodynamics

[5] P. Castillo et al., Modelling and Control of Mini-Flying Machines

[6] G. Angeletti, J. R. Pereira Valente, L. Iocchi, D. Nardi , Autonomous Indoor Hovering with a Quadrotor

[7] G. M. Hoffmann, H. Huang, S.L. Waslander and C.J. Tomlin – "Quadrotor Helicopter Flight Dynamics and Control: Theory and Experiment", AIAA Guidance, Navigation and Control Conference and Exhibit (AIAA paper 2007-6461).

[8] S. Bouabdallah and R. Siegwart – "Design and Control of a Miniature Quadrotor", Advances in Unmanned Aerial Vehicles, pp. 171-210.

[9] A.O. Kivrak – "Design of control systems for a quadrotor flight vehicle equipped with inertial sensors", Atilim University, December 2006.

[10] R.J. Mantz and E.J. Tacconi – "A regulating and Tracking PID Controller", University of La Plata, Argentina.

[11] R.J. Mantz, E.J. Tacconi – "Complementary rules to Ziegler and Nichols' rules for a regulating and tracking controller", International Journal of Control, 49: 5, 1464-1471.

[12] E. Eitelberg – "A regulating and tracking PI(D) controller", International Journal of Control, 45: 1, 91-95.

[13] "Linear Feedback Control" by DingyuXue, YangQuan Chen, and Derek P. Atherton.

[14] W. Wang, G. Song, K. Nonami, M. Hirata, and O. Miyazawa. Autonomous control for micro-flying robot and small wireless helicopter. In IEEE/RSJ International Conference on Intelligent Robots and Systems, Beijing, China, 2006.IEEE/RSJ.

[15] J. McLean, G. Parker, and N. Seal. Basic control for four rotor autonomous aerial agent. ISORA (International Symposium on Operations Research and Its Applications),
2008.

[16] M. E. Greene and V. Trent. Software algorithms in air data attitude heading reference
systems. Aircraft Engineering and Aerospace Technology, 75(5): 470–476, 2003.

9 Acknowledgements

We would like to express our appreciation to the following people for their support along the way:

Leonid Mirkin

Aram Movsisian

Sergey Danielian

Yair Atzmon

Hector P. Rotstein

Appendix A

Matlab codes

```
%% LISSAJOU FIGURES
clc
clearall
formatlong

dt=0.02; %sampling time
A=13;
B=20;
pai=0.5;
w=1.5;
N=1000;
t=(0:N-1)*dt;
t1=t(200:end);
r=A*sin(w*t1);
y=B*sin(w*t1+pai);
X=-13:13;
Y=X*B/A;
X1=X;
Y1=X1*-A/B;
plot(r,y,'LineWidth',2)
holdon
%plot(X,Y,'.-.r')
%plot(X1,Y1,'.-.r')
title('lliasu figure')
xlabel('reference signal')
ylabel('output signal')
axis([-15 15 -21 21])
axisequal
grid
%%

%%CONTROLLER EFFECT///////////////
clc
clearall
formatlong

for w=0.3:0.1:3
R=0.1;
t=0:0.05:2.1*pi;
P=2.3/(-w^2+3.7*1i*w);
disc=R*(cos(t)+1i*sin(t));
P_area=P+disc;
C1=10;
s=w*1i;
C2=10+0.1/s+0.5*s;
T1=C1.*P_area./(1+C1.*P_area);
T_mid1=C1*P/(1+C1*P);
T2=C2.*P_area./(1+C2.*P_area);
T_mid2=C2*P/(1+C2*P);
%L=C.*P_area;
figure(1)
grid
plot(real(P_area),imag(P_area))
grid
holdon
```

```
plot(real(P),imag(P),'*')

plot(real(T1),imag(T1),'black')
plot(real(T2),imag(T2),'g')
%plot(real(T_mid1),imag(T_mid1),'*black')
legend('P','mid','T1','T2')


end

%%/////////////////////////////////////

function [w,F]=fouriertrans(t,f,N);
%[w,F]=fouriertrans(t,f,N). Continuous-time Fourier transform
%
%   IN: t, vector of equidistant sampled time
%       f, vector of sampled function values 'f(t)'
%       N, number of grid points (preferably N=2^k)
%  OUT: w, row vector of sampled frequencies
%       F, row vector of sampled Fourier transform 'F(w)'
%
%                     Author: GjerritMeinsma, Univ. of Twente

if N < length(f)
disp('Not all data is used.');
end
T=t(2)-t(1);                  % sampling time
ws=2*pi/T;                    % sampling frequency
points=1:(N/2);
w=(points-1)*ws/N;            % N/2 grid points from [0,ws/2]
F=T*fft(f(:)',N);             % note: f(:)' is a row vector
F=F(points).*exp(-j*t(1)*w); % T sum f[n]e^(-jnwT)rect_ws(w)

clc
clearall
formatlong

loadtest_2_512_sf_0.mat
dt=1/512;
N=16384;%number of sampels
G_X=data(1:N,5);
G_Y=data(1:N,6);
G_Z=data(1:N,7);
A_X=data(:,2);
A_Y=data(:,3);
A_Z=data(:,4);
N=length(G_Y);
t=(0:N-1)*dt;
wb=50;
%%  data after L.P.F
G_X_fil=G_X_filter.signals.values;
G_Y_fil=G_Y_filter.signals.values;
G_Z_fil=G_Z_filter.signals.values;
N1=length(G_X_fil);
t1=(0:N1-1)*dt;
%% down sampling after filter
n=floor(N/5);
G_X_fil_d_s=zeros(n,1);  % gyro X after filter down sampling
G_Y_fil_d_s=zeros(n,1);
G_Z_fil_d_s=zeros(n,1);
for i=1:n
```

```
G_X_fil_d_s(i)=G_X_fil(5*(i-1)+1);
G_Y_fil_d_s(i)=G_X_fil(5*(i-1)+1);
G_Z_fil_d_s(i)=G_X_fil(5*(i-1)+1);
end
dt_new=6/512;
t2=(0:n-1)*dt_new;
%% down sampling no filter
n=floor(N/5);

G_Y_d_s=zeros(n,1);  % gyro X no filter down sampling
for i=1:n
G_Y_d_s(i)=G_Y(5*(i-1)+1);
end

%% down sampling with averaging
n=floor(N/5);
G_Y_ave_d_s=zeros(n,1);  % gyro Y  averaging  filter down sampling
for i=1:n
G_Y_ave_d_s(i)=mean(G_Y(5*(i-1)+1:5*(i-1)+5));
end

%% spectrum analysis data

[w1,G_X_spec_high]=fouriertrans(t1,G_X_fil,N1);  % high freq with
filter
[w1,G_Y_spec_high]=fouriertrans(t1,G_Y_fil,N1);
[w1,G_Z_spec_high]=fouriertrans(t1,G_Z_fil,N1);

f1=w1/2/pi;

[w2,G_X_spec]=fouriertrans(t2,G_X_fil_d_s,n);  %low frq  with filter
[w2,G_Y_spec]=fouriertrans(t2,G_Y_fil_d_s,n);
[w2,G_Z_spec]=fouriertrans(t2,G_Z_fil_d_s,n);

[w2,G_Y_spec_no_fil]=fouriertrans(t2,G_Y_d_s,n);% low freq no filter

[w2,G_Y_spec_ave_fil]=fouriertrans(t2,G_Y_ave_d_s,n);% low freq
average filter
f2=w2/pi/2;
%A_X_spec=abs(fft(A_X));
%A_Y_spec=abs(fft(A_Y));
%A_Z_spec=abs(fft(A_Z));

%% gyro high sample
figure(1)
subplot(3,1,1)
plot(f1,G_X_spec_high)
title('X Angular velocity  spectrum')
xlabel('f[HZ]')
ylabel('amplitude')
subplot(3,1,2)
plot(f1,G_Y_spec_high)
title('Y Angular velocity  spectrum')
xlabel('f[HZ]')
ylabel('amplitude')
subplot(3,1,3)
plot(f1,G_Z_spec_high)
title('Z Angular velocity  spectrum')
xlabel('f[HZ]')
ylabel('amplitude')
```

```
%% gyro spectrum low sample
figure(1)
subplot(3,1,1)
plot(f2,G_X_spec)
title('X Angular velocity  spectrum')
xlabel('f[HZ]')
ylabel('amplitude')
subplot(3,1,2)
plot(f2,G_Y_spec)
title('Y Angular velocity  spectrum')
xlabel('f[HZ]')
ylabel('amplitude')
subplot(3,1,3)
plot(f2,G_Z_spec)
title('Z Angular velocity  spectrum')
xlabel('f[HZ]')
ylabel('amplitude')

%% fft for NO FILTER average filter and L.P.F


figure(1)
plot(f2,G_Y_spec_no_fil,f2,G_Y_spec,f2,G_Y_spec_ave_fil)
title('Y Angular velocity  spectrum')
xlabel('f[HZ]')
ylabel('amplitude')
legend('spec no filter','spec with L.P. filter','spec with average
filter')
grid





%% acc  spectrum



figure(2)
subplot(3,1,1)
plot(f(ind),A_X_spec(ind))
title('X acceleration spectrum')
xlabel('f[HZ]')
ylabel('amplitude')
subplot(3,1,2)
plot(f(ind),A_X_spec(ind))
title('Y acceleration spectrum')
xlabel('f[HZ]')
ylabel('amplitude')
subplot(3,1,3)
plot(f(ind),A_X_spec(ind))
title('Z acceleration spectrum')
xlabel('f[HZ]')
ylabel('amplitude')
```

```
function [error ] = error_min_comp_model( var )
%found error of complex model
loadtest7.mat

dt=0.02;
roll=data(451:end,14);
counter=ones(1,length(data(:,14)));
shift_theo=roll(49);
roll=roll-shift_theo;

t=(1:length(roll))*dt;
sb=450; %shift beginning
A1=0; %%build input
t1=counter(1:500-sb);
A2=14;
t2=counter(501-sb:550-sb);
A3=-14;
t3=counter(551-sb:600-sb);
A4=0;
t4=counter(601-sb:end-sb);
input=[A1*t1 A2*t2 A3*t3 A4*t4]';

s=tf('s');
k=var(1);
z1=var(2);
p1=var(3);
p2=var(4);
p3=var(5);

P=k*(z1+s)/s/(s+p1)/(s+p2)/(s+p3);
P1=10;
I1=0.1;
D1=0.5;
C1=P1+I1/s+D1*s;
T=feedback(C1*P,1);

y=lsim(T,input,t);
error=norm(y-roll);
end


clc
clearall
formatlong

loadtest7.mat


[var_new,error,exitflag,output] = fminsearch(@(var)
error_min_comp_model(var),[10 10 10 10 10]);
%% edit simulate respode
dt=0.02;
roll=data(451:end,14);
counter=ones(1,length(data(:,14)));
shift_theo=roll(49);
roll=roll-shift_theo;
t=(1:length(roll))*dt;
sb=450; %shift beginning

A1=0;
t1=counter(1:500-sb);
```

```
A2=14;
t2=counter(501-sb:550-sb);
A3=-14;
t3=counter(551-sb:600-sb);
A4=0;
t4=counter(601-sb:end-sb);
input=[A1*t1 A2*t2 A3*t3 A4*t4]';
s=tf('s');

k=var_new(1);
z1=var_new(2);
p1=var_new(3);
p2=var_new(4);
p3=var_new(5);

P=k*(z1+s)/s/(s+p1)/(s+p2)/(s+p3);

P1=10;
I1=0.1;
D1=0.5;
C1=P1+I1/s+D1*s;
T=feedback(C1*P,1);
[F,T1]=balreal(P);
 [A,B,C,D]=dssdata(F);
 r=2;
A_min=A(1:r,1:r);
B_min=B(1:r);
C_min=C(1:r);
Pr=tf(ss(A_min,B_min,C_min,D));
Tr=feedback(C1*Pr,1);
yr=lsim(Tr,input,t);
y=lsim(T,input,t);
figure(1)
plot(t,y,t,roll,t,input,t,yr);
title('time simulate')
xlabel('t[sec]')
ylabel('roll angle[deg]')
legend('comlex model','real','input','simp model')
axis([0 6 -23 23])
figure(2)
rlocus(P)
```

Appendix B

Theoretic model

From Newton's second low equation we know:

$$\sum M = \ddot{\theta} I + \dot{\theta} c + k\theta$$

In our system we don't have the k product, so our equation know is:

$$\sum M = \ddot{\theta} I + \dot{\theta} c$$

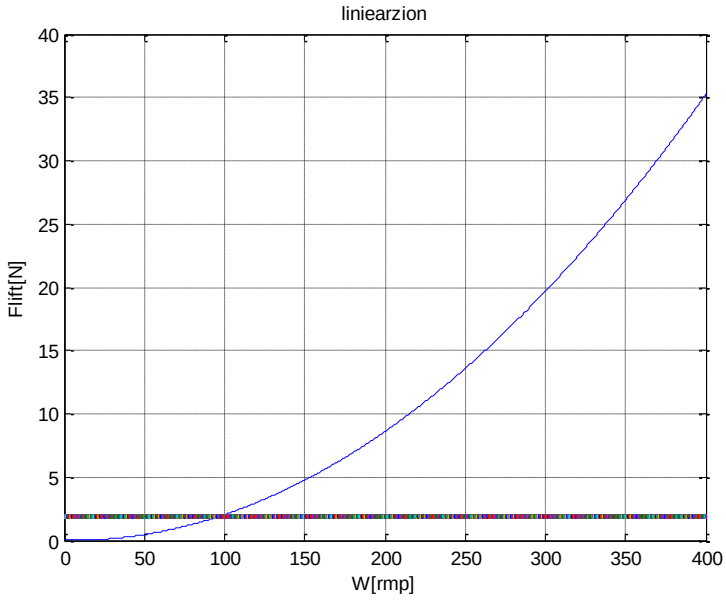
Our control signal is cause to rotor speed to be:

$$\Omega_i = \Omega_{nom} \pm \Delta\Omega \quad \underset{after\ \text{linearizing}}{\rightarrow} \quad F_i = F_{nom} \pm \Delta F$$

After linearizing the force: $K_\omega = \frac{\partial F_{lift}}{\partial \Omega}\Big|_{\Omega_{eq}}$

$$\frac{\Delta F}{\Delta \Omega} = K_\omega$$

From experiment the made at Pennsylvania they receive a correlation between the speed rotor and the lift force that the rotor produces:



The equation that they receive was:

$$F_{Lift}(\Omega) = 0.2263 \cdot 10^{-3} \cdot \Omega^2 - 2.448 \cdot 10^{-3} \cdot \Omega + 11.97 \cdot 10^{-3}\ [N]$$

Our weight is about 800 gr, that mean from this equation that our equilibrium speed rotor is:

$$F_{lif}\left(\Omega_{eq}\right)=\frac{8[N]}{4[rotors]}=2\rightarrow \qquad \Omega_{eq}=100[r.p.m]$$

Of course this is a nominal speed rotor, when we want to hovering we need to increase that speed rotor.

From this we can calculate:

$$\frac{\partial F_{lift}}{\partial\Omega}=0.4526\cdot10^{-3}\Omega-2.448\cdot10^{-3} \qquad \rightarrow \qquad K_{\omega}=\frac{\partial F_{lift}}{\partial\Omega}\Big|_{\Omega_{eq}=100}=0.0428\left[N/rpm\right]$$

Back to our equation we receive:

$$M=r\cdot(F_{nom}+\Delta F-(F_{nom}-\Delta F))=2r\Delta F$$

$$\frac{u_1}{\Delta\Omega}=K_{system}\rightarrow M=K_{system}\cdot K_{\omega}\cdot2\cdot r\cdot u_i$$

$\overline{u}_i$ - Our control signal.

We can short this term by define: $K_{system}\cdot K_{\omega}\cdot2\cdot r\equiv K$

Now we transfer the equation to Laplace domain

$$K\overline{u}_i=I\cdot s^2\theta+c\cdot s\theta\rightarrow\frac{\theta}{\overline{u}_i}=\frac{K}{s\left(sI+c\right)}=P(s)$$

We know that the control signal is pass through inner controller, that have a gyro, but our assuming is that the inner control don't change the model shape, it mean the inner control only change our constant variable: I,c,K .

This model is true about all the our Euler angles, the difference between is again only on the constant variable: I,c,K .