UAV Adaptive Flight Control

1MAE707- Adaptive Control

Authors : SHARMA Akash, MALANDKAR Yash, BELAPURKAR Atharva

Under the kind guidance of

BORDENEUVE-GUIBE, Joël

Submitted on : 19th June, 2022

Table of Contents

1. Introduction

The objective of this report is to consider the adaptive control of a quadrotor drone submitted to a severe mass drop and to control its movement according to the specified requirements. The brutal mass drop directly affects the dynamics of the system and alters the behavior of the flight control system. Adaptive control is used to provide good tracking performance and robust control when plant parameters are time-varying as it does not require prior information about time-varying parameters. This strategy is implemented in various stages. First on a pre-build Simulink model with quadrotor dynamics and then implementing it onto the SiMoDrones simulator.

The implementation of the control strategies was divided into four sections:

- Altitude Control with PID using Simplified Model
- Altitude Control with Adaptive PID using Simplified Model
- Altitude Control with Adaptive PID using SiMoDrones Simulator
- All Axis Control with Adaptive PID using SiMoDrones Simulator

1.1. Requirements

	Design Requirements	Software Requirements
1.	Quadrotor	MATLAB
2.	Opti-Track System	Simulink
3.	Wifi-Link	
4.	Ground-Station	

1.2. Flight Trajectories and Scenarios

The altitude control and entire trajectory are subject to certain parameters that the UAV must follow. These requirements are listed below in terms of various flight scenarios.

For the control along the z-axis of the simplified model, the following reference trajectory is defined:

- A step of 1.5 meters for 10 s

- A step of 2 meters for 10 s
- Mass drop at 20 s
- Landing to the ground with a ramp of 10 s.

For the control along the z-axis of the SiMoDrones model, the following reference trajectory is defined:

- A ramp of 1.5 meters for 5 s
- A step of 2 meters for 20 s
- Mass drop at 27 s
- Landing to the ground with a ramp of 5 s.

For the control along all 3-axis of the SiMoDrones model, the following reference trajectory is defined:

- A ramp of 1.5 meters for 5 s
- An horizontal hourglass shape with dimensions of 1 meter in both x and y directions
- Mass drop at 27 s
- Landing to the ground with a ramp of 4 s
- Total simulation lasted for 60 s.

2. Case Studies

2.1. Altitude Control with PID using Simplified Model

The quadrotor dynamics in the z-direction is approximated by using a double integrator as the plant. The input to the plant is thrust and the states used are position and velocity. PID tuning method was implemented for the control of the quadrotor on the z-axis. Figure 1 shows the simplified quadrotor model.

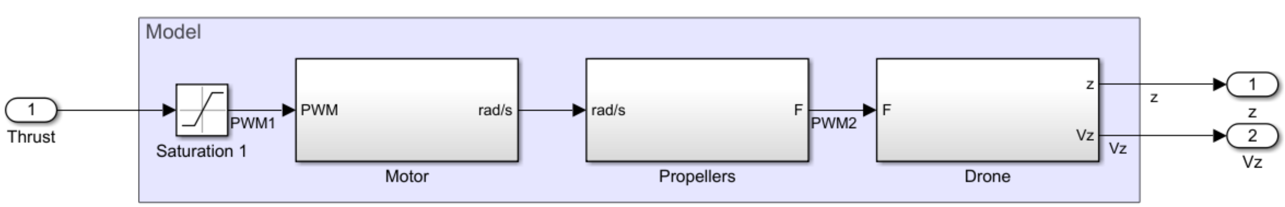

Figure 1: Simplified Quadrotor Model

The objective here was to control the altitude of the quadrotor according to the specifications mentioned above in section 1.2. This was made possible by adding a proportional and an integral controller for the error in position and also a proportional controller for the error in velocity. This proportional controller acted as a derivative controller for the error in position. A Simulink schematic is given in figure 2 to give a better understanding of the system.

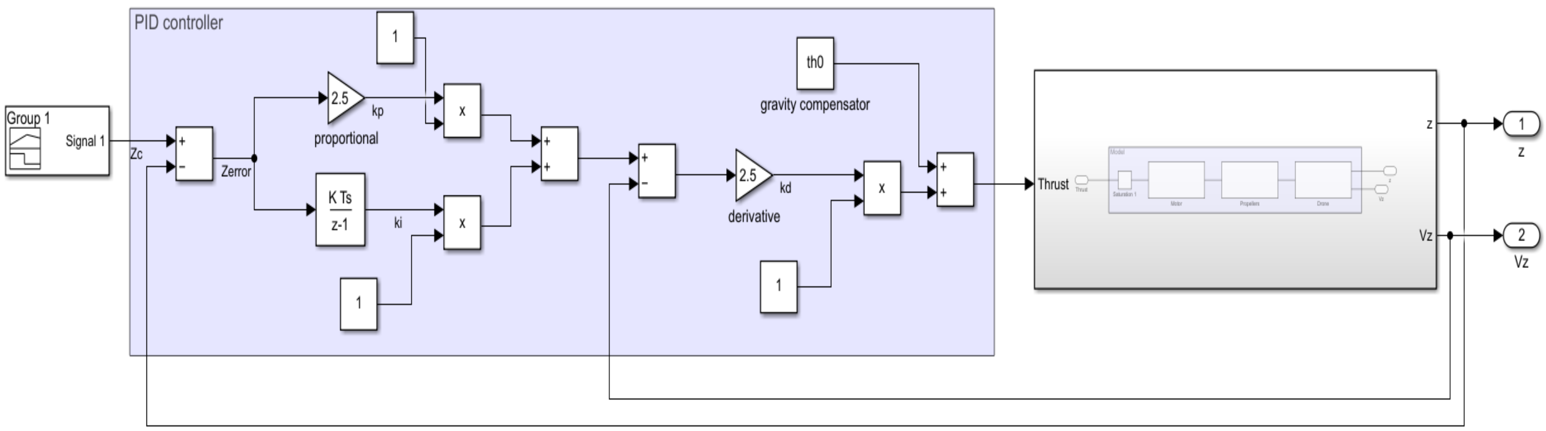

Figure 2: Simulink Schematic PID model

The velocity being the faster of the two states was controlled using a derivative gain Kd, whereas the altitude was controlled using a proportional and an integral gain Kp and Ki respectively. Since the model is highly non-linear, Simulink's PID tuner was unable to linearize the model and therefore could not provide any gain values. The gains of the PID were tuned in small steps until a satisfactory control output was achieved. The final values of the gains are mentioned in table 1.

	Kp	**Ki**	**Kd**
Z-axis	2.5	0.02	2.5

Table 1: PID Gains

The Control Law used to calculate Thrust is:

$$U = (Kp + Ki)*(Zref - Z) + Kd*(Vref-V)$$

Figure() shows the controlled output obtained after the PID tuning technique. It can be seen in the figure that the control output smoothly follows the commanded input with a slight delay which can be explained by the internal dynamics of the drone. Another observation is that the controlled output follows the ramp input with a steady-state error, this is because to cancel out the steady-state error we will require a double integrator, which is not the case in our system. Here, a small deviation can be observed around 20 seconds, this is because of the mass drop simulation. The PID alone cannot trace back the input signal once disturbed due to a mass drop, therefore an adaptive control strategy shall be implemented.

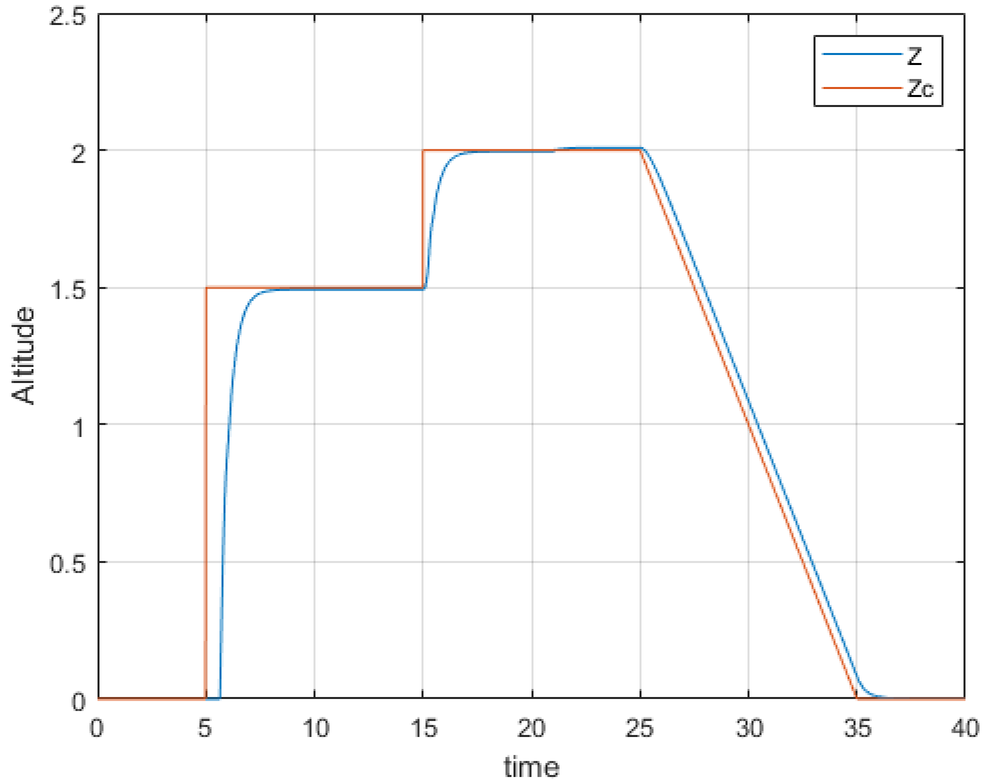

Figure 3: Z-axis control

2.2. Altitude Control with Adaptive PID using Simplified Model

To improve the results obtained using the PID control technique, the altitude is subjected to an adaptive control strategy designed on the basis of the MIT rule. The rule is described as follows:

$$\frac{\partial \theta}{\partial t} = -\gamma_e \frac{\partial e}{\partial \theta}$$

where θ is the adaptation parameter, γ is the gain, e is the error between the outputs of the plant and the model, and $\partial e/\partial \theta$ is the sensitivity derivative. A reference model was taken depending on the provided values of natural frequency ω= 2 rad/s and damping ratio σ=0.8. This is done to ensure a non-oscillatory transient and a fast response of the model reference.

The reference model used for the MIT rule was:

$$\frac{z_m}{u} = \frac{\omega^2}{s^2 + 2\sigma\omega + \omega^2}$$

Figure 4 gives a better understanding of the Simulink model used for the adaptive control technique.

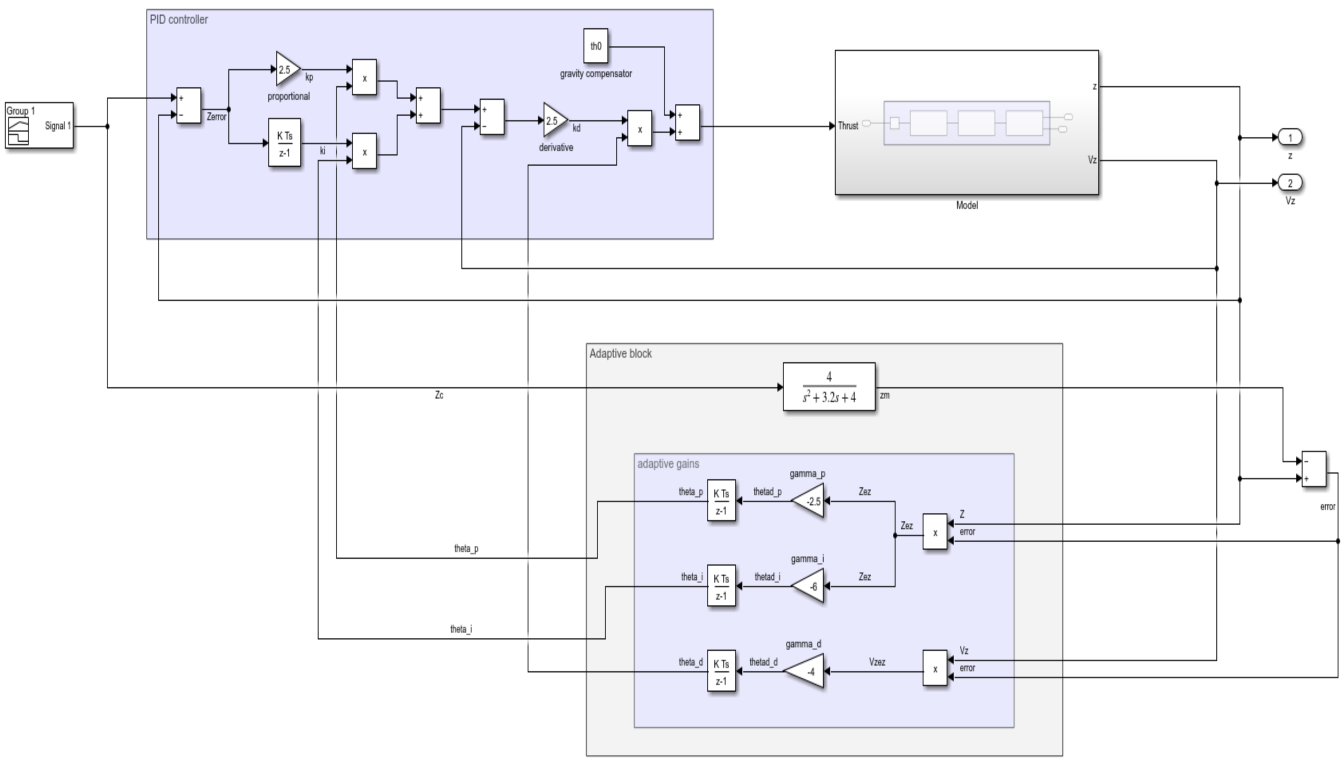

Figure 4: Simulink schematic of the adaptive model

The control law used for calculating thrust was:

U= (Kpa + Kia)*(Zref - Z) + Kda*(Vref - V)

Where,

$$Kpa = \theta_p * Kp$$

$$Kia = \theta_i * Ki$$

$$Kda = \theta_d * Kd$$

Where, Kp, Ki, and Kd are the tuned P, I and D gains from the previous section and $\boldsymbol{\theta}_p$, $\boldsymbol{\theta}_i$ and $\boldsymbol{\theta}_d$ are the adaptive gains, which are updated using Lyapunov law, as below:

$$\boldsymbol{\theta}_{pdot} = -\gamma_p z e_z$$

$$\boldsymbol{\theta}_{idot} = -\gamma_i z e_z$$

$$\boldsymbol{\theta}_{ddot} = -\gamma_d z e_z$$

The adaptation terms γ_p, γ_i, γ_d need to be tuned in order to get a good tracking performance. Table 2 below shows the P, I, and D gains as well as the adaptation gains used for perfect tracking.

	Kpa	Kia	Kda	γ_p	γ_i	γ_d
Z-axis	2.5	0.02	2.5	-2.5	-6	-4

Table 2 : PID and adaptive gains respectively

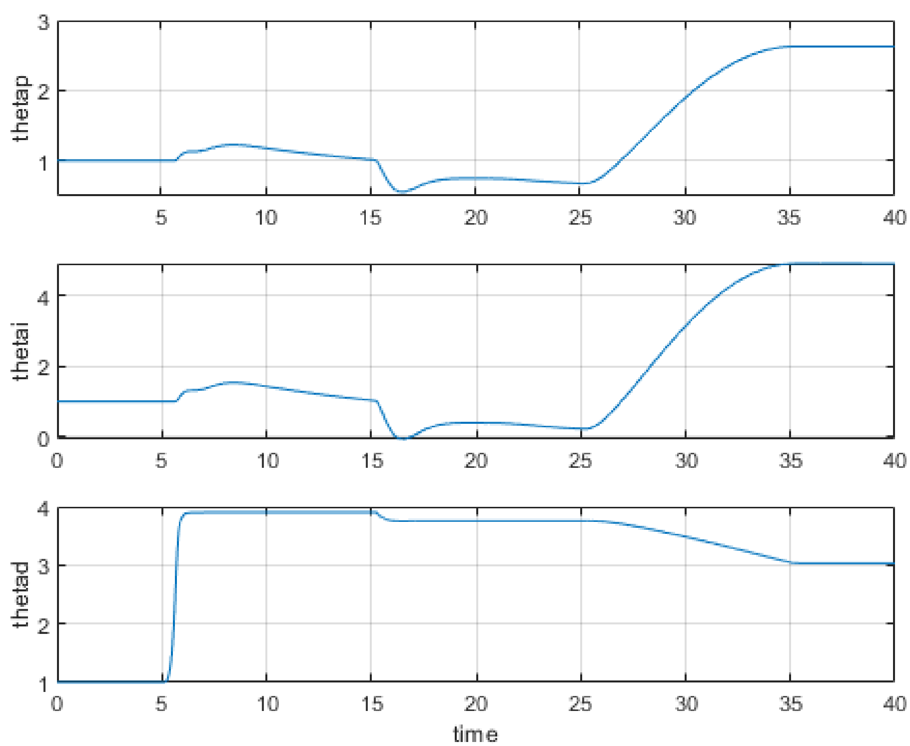

Figure 5 : Adaptation Gains vs time

figure 5 shows the adaptation of γ_p, γ_i, γ_d over the simulated time period. It can be noticed that all the adaptation terms reach a constant value after some period of time, which signals that the error has been brought to zero and there is a proper tracing of the input.

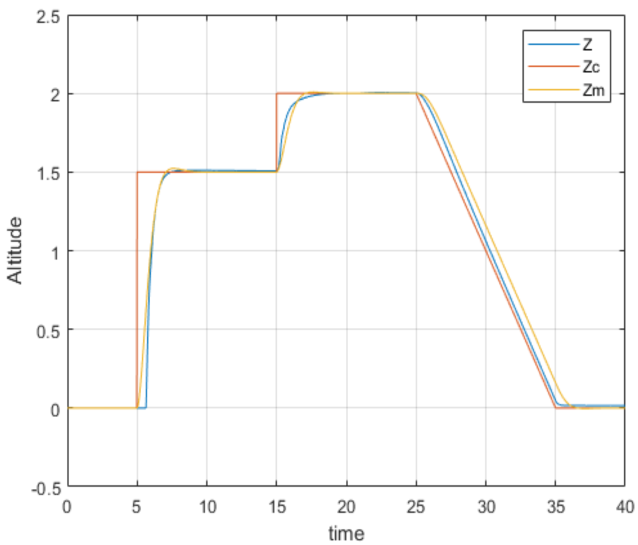

Figure 6: z-axis control using adaptive controller

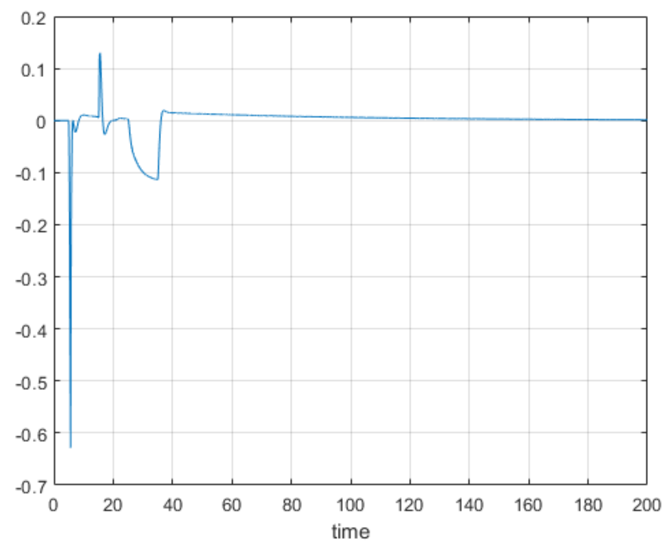

Figure 7: error vs time

Figure 6 shows the controlled output obtained after the adaptive control technique. It can be seen in the figure that the control output (Z) smoothly follows the commanded input with a slight delay which can be explained by the internal dynamics of the drone. Here, a small deviation can be observed around 20 seconds, this is because of the mass drop simulation. The difference between adaptive control and just PID control is that after the mass drop (perturbation) the control output immediately starts following the input signal. The response of the system to the perturbation gets faster and the overshoot decreases quickly. Figure 7 shows the variation of error over the simulated time period. It can be seen that after various overshoots the error eventually gets back to zero which guarantees perfect signal tracking of the system.

2.3. *Altitude Control with Adaptive PID using SiMoDrones Simulator*

After the successful implementation of the adaptive PID controller for a payload drop, it was then implemented in SimoDrones which closely mimics the non-linear dynamics of the actual drone taking the payload drop into account. The presence of these non-linearities necessitates the re-tuning of the PID controller. Therefore, first, all three adaptive gains are set to 0 which essentially cuts off the adaptive scheme from the Simulink model, then the PID is tuned. This is followed by setting appropriate values of adaptive gains that give acceptable tracking performance before as well as after the payload drop. All the final gains as well as the Simulink models constructed are shown below.

	Kp	**Ki**	**Kd**	**ɣp**	**ɣi**	**ɣd**
Z-axis	2	0.6	0.09	-0.12	-0.12	-0.12

Table 3: PID and adaptive gains respectively

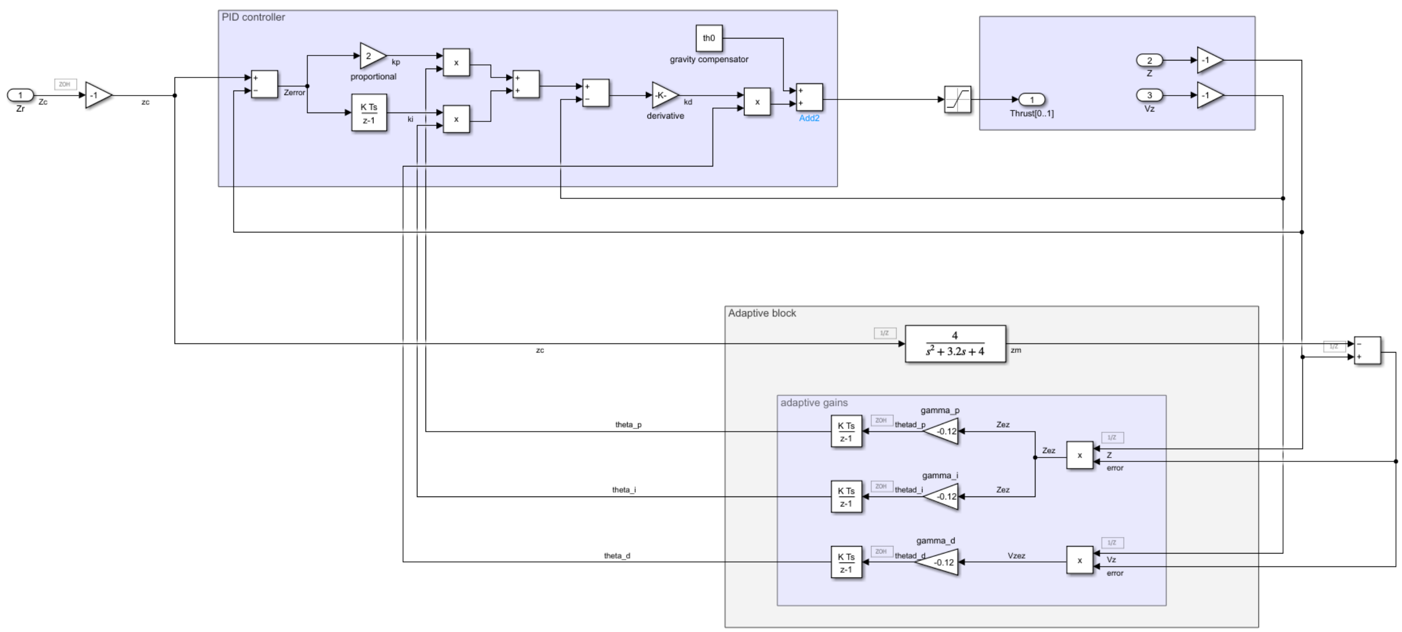

Figure 8: Z-axis Adaptive PID controller

The tracking performance of the quadrotor vehicle is acceptable with the above-mentioned controller model as can be seen in the following graphs.

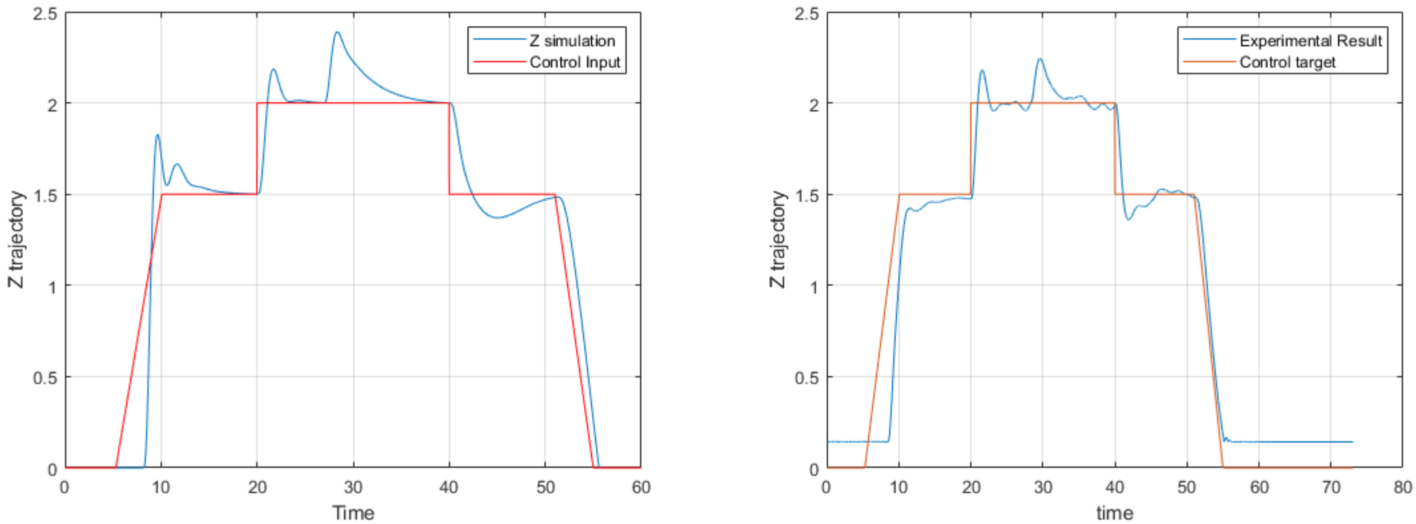

Figure 9: Z control Tracking Performance (Simulation and Experiment)

The Z control is acceptable in the simulation as it quickly adapts to the changing dynamics of the mass drop at 27 seconds. This is also observed in the theta variations, all of which suddenly change and converge to a new value at 27 seconds. In general, the second half of the flight is better tracked than the first half since the adaptive controller becomes more robust due to the perturbations that occur.

In the graph of Z experiment vs time, the adaptive controller is able to quickly track the target values. One of the major drawbacks of the model reference adaptive control is that the presence of an error is necessary to be able to learn from it and adapt the gains accordingly. Hence, we see that the descending ramp at the end of the flight is much better tracked than the ascending ramp in the beginning. After the mass drop at t=27s, the dynamic of the model changes, so the controller brings the resulting overshoot back to the target by adapting the gains. Finally, we also see that the initial and final Z values of the vehicle in the experiment are not 0. This is due to the fact that the spherical markers attached to the vehicle that is responsible for acquiring the position and orientation of the vehicle through the optitrack system are at a slight altitude of a couple of centimeters. Therefore, from the graph, a resting altitude of around 20cm is observed.

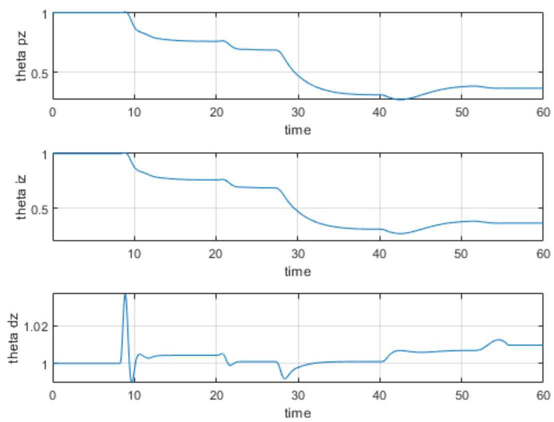

Figure 10: Adaptive Gains Variation

2.3.1 Trajectory Analysis

The trajectory is mapped by giving a ramp input at t=5s till t=10s to 1.5m. The drone then encounters a step input at t=20 to an altitude of 2m. The descent approach is symmetric in which there is a step input down to 1.5m at t=40s followed by a ramp input to the ground at t=50s.

The uncertainties are mostly aptly demonstrated in the 3D graph of the test where the vehicle deviates in the X and Y direction. These uncertainties are attributed to factors such as motor actuation limits, sensor noise, control loop latencies, and external perturbations, etc. However, these deviations are insignificant; the largest deviations observed are 6.5cm & 11.7cm in the X & Y directions respectively.

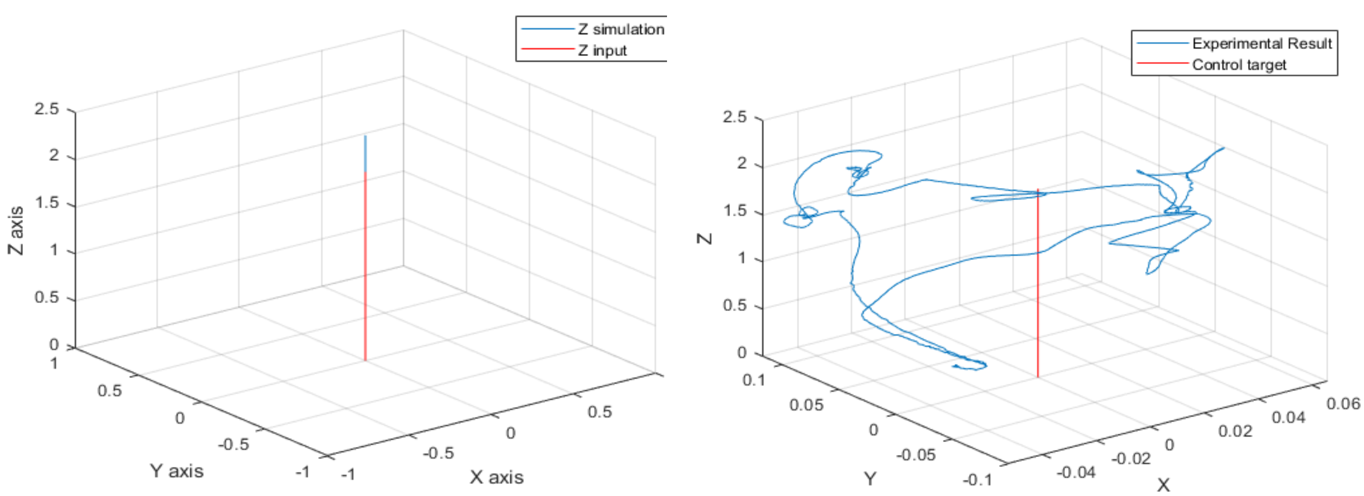

Figure 11: 3D trajectory (Simulation and Experiment)

2.4. All Axis Control with Adaptive PID using SiMoDrones Simulator

The underlying principle of control in the X and Y direction is the same as that of the Z direction, the difference being the output of all the 3 directions. X-control outputs the target pitch angle, Y-control outputs the target the roll angle and the Z control outputs the thrust. For assessing the performance of this controller in the lateral directions, taking into account the non-linearities that would be introduced in the experiment, a complex trajectory has been designed to be tracked so that the controllers are proven to be robust enough. This trajectory takes the shape of an hourglass from the top view.

A similar approach has been taken to tune these lateral controllers like that of the Z controller where after setting the adaptive gains to 0, the PID is tuned. The final gains corresponding to an acceptable tracking performance as well as the Simulink models are shown below.

	Kp	**Ki**	**Kd**	**ɣp**	**ɣi**	**ɣd**
X-axis	1	0.02	0.35	-0.05	-0.05	-0.05
Y-axis	1	0.01	0.35	-0.05	-0.05	-0.05

Table 4: PID and adaptive gains respectively

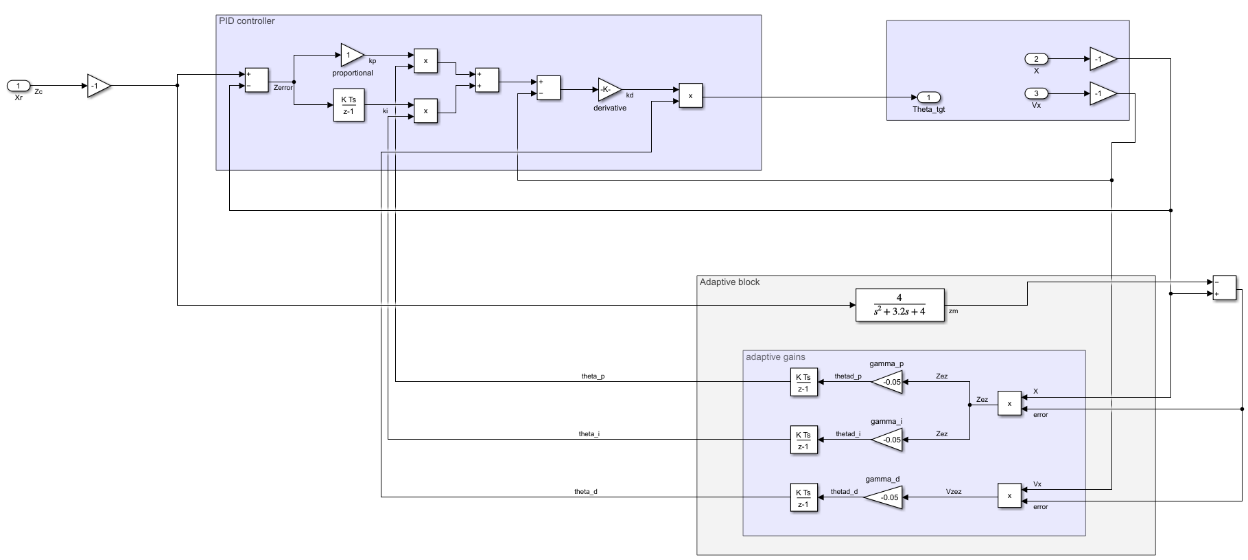

Figure 12: X control Full System

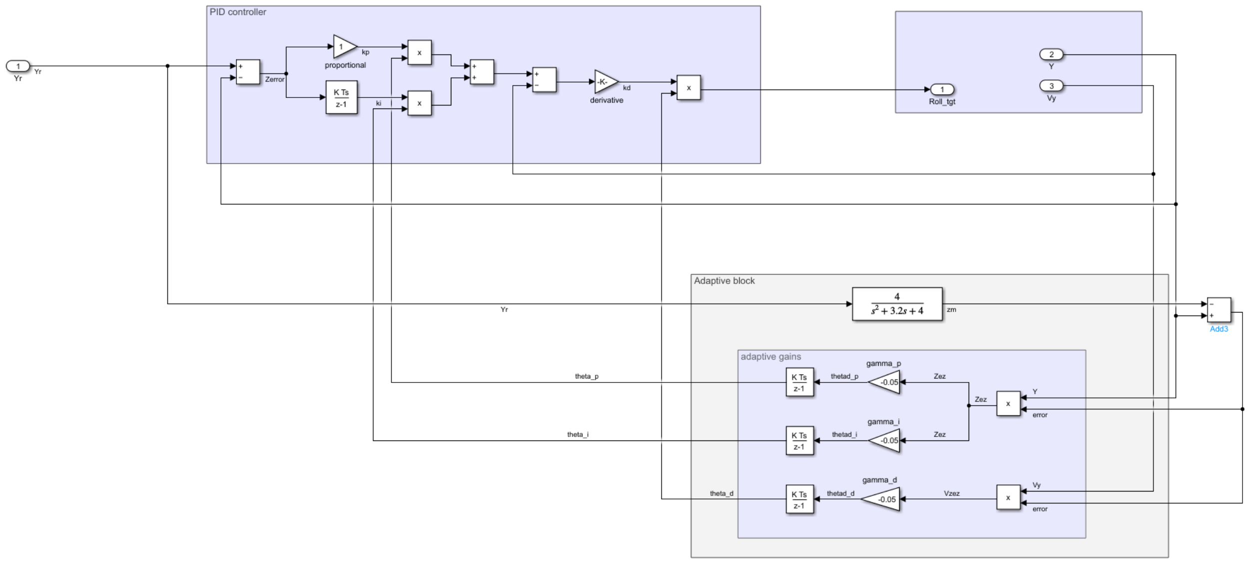

Figure 13: Y control Full System

The tracking performance of the quadrotor vehicle is acceptable with the above-mentioned controller models as can be seen in the following graphs.

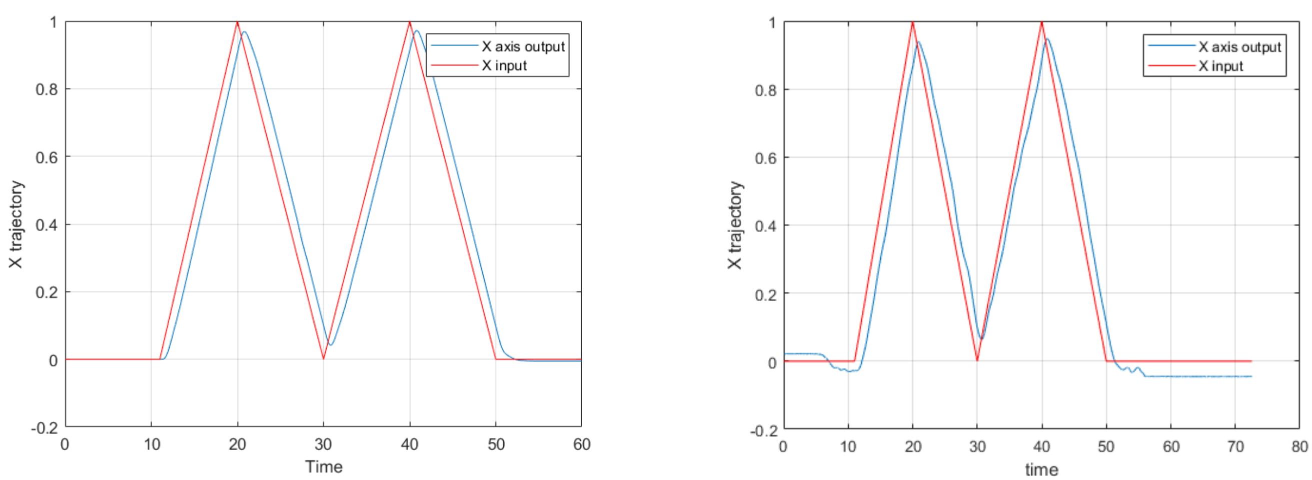

Figure 14: X control Tracking performance (Simulation and Experiment)

The simulation and experiment curves are nearly identical. The initial and final phases of the flight are in the takeoff and landing phase and therefore reflect slight oscillations.

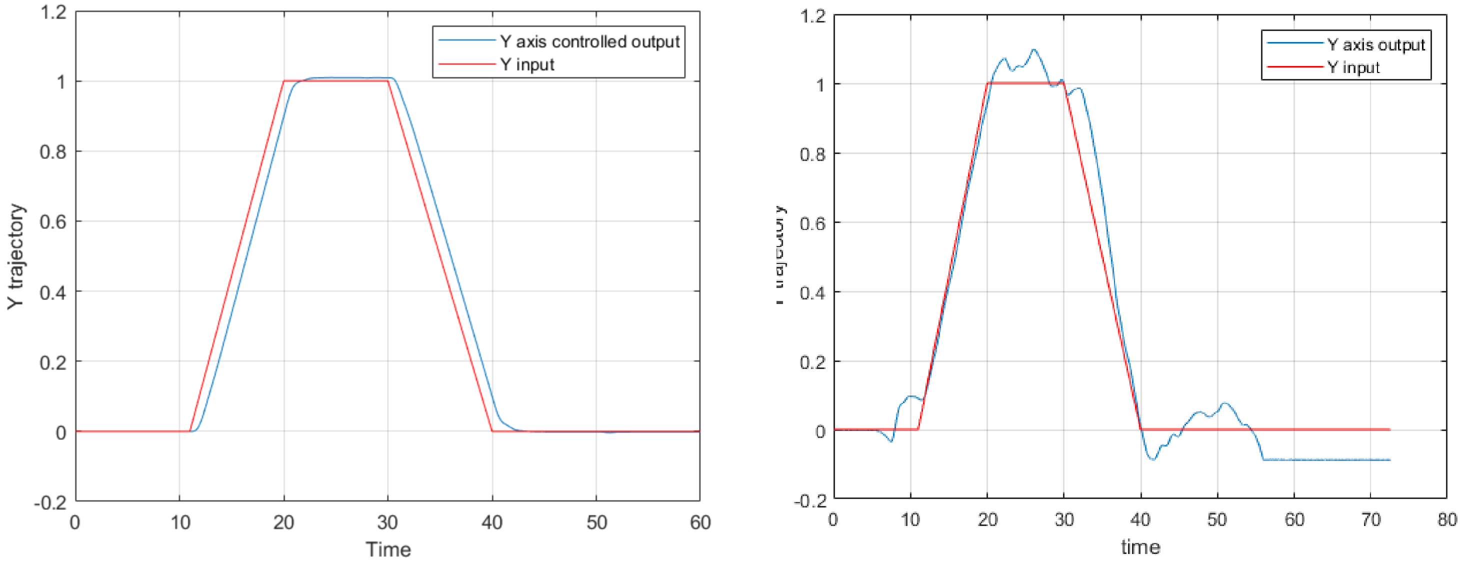

Figure 15: Y control Tracking Performance (Simulation and Experiment)

These graphs show a good tracking performance of the drone with the target. We see that the Y control was particularly affected due to the sudden mass drop. This is because the mass drop happened at t=27s which was around the time when the drone was making a sharp turn in its trajectory. However, all the three controllers quickly adapt to the new dynamic.

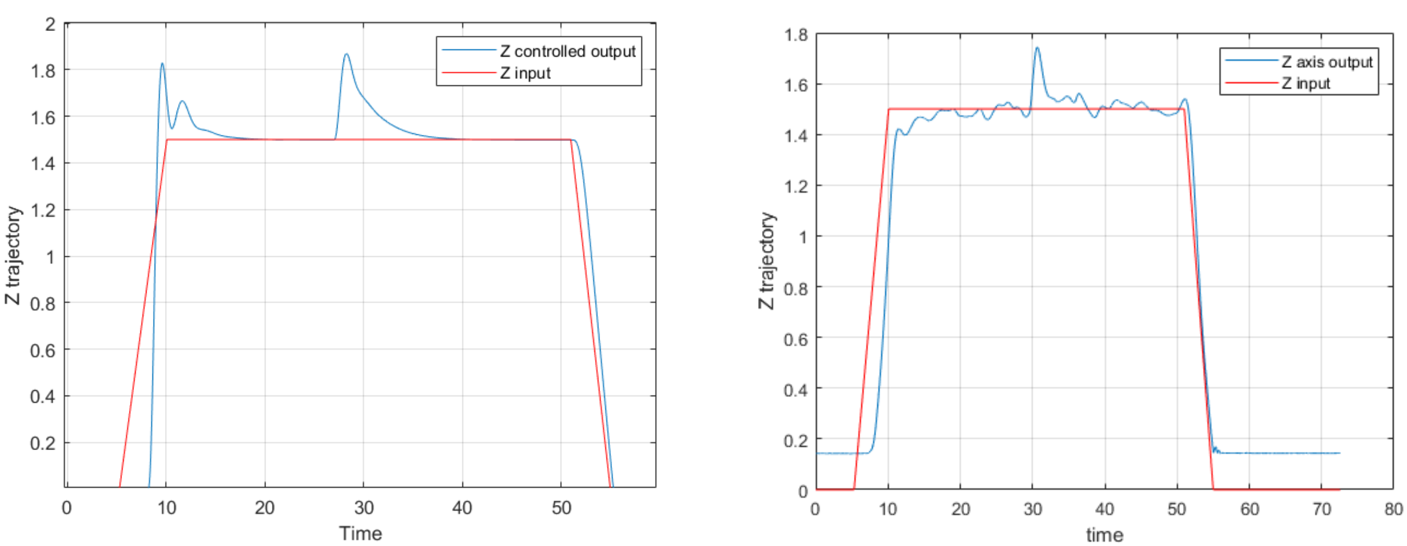

Figure 16: Z control Tracking Performance (Simulation and Experiment)

The Z control graph remains acceptable as well. The aerodynamic fluctuations seen in the experiment in the Z-axis are a byproduct of the changing pitch and roll angles of the drone since changing the orientation of the drone entails redistributing the thrusts in each of the propellers which may cause a slight loss in the lift.

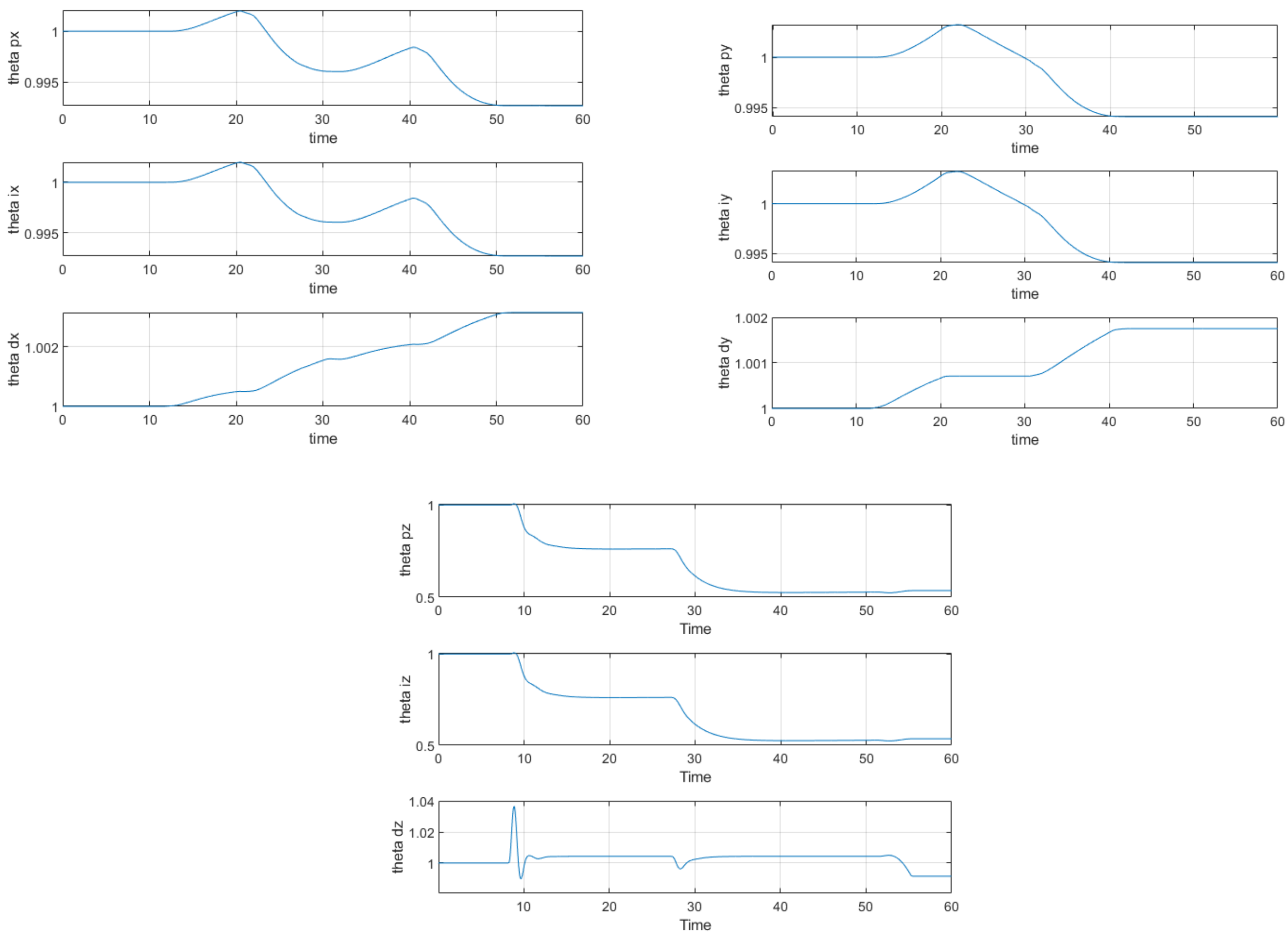

Figure 17: Adaptive Gains Variations

As we can see, the simulations give great tracking performance and quick adaptation with the controller. The adaptive gains display expected behavior in the experiment as well where each one of them changes suddenly due to the mass drop and finally converges.

2.4.1 Trajectory Analysis

This trajectory is created as follows.

1) A ramp input of 1.5m at t=5s is given in the Z-axis.
2) The drone then traverses the first diagonal where both X and Y axes are given a ramp input of 1m each at t=11s
3) The X-axis is given a ramp down input back to 0 at t=20s. Towards the end of this movement, the mass drop is triggered.
4) The drone then follows the second diagonal where X and Y axes are given a ramp up and ramp down input of magnitude 1m each at t=30s.
5) Finally, the drone comes back to its initial flight point when the X-axis is given a ramp down input back to 0 at t=40s

The adaptive controller is shown to be robust enough to tolerate and respond to the mass drop at a critical point in the trajectory where the drone makes a sharp turn towards the second hourglass diagonal. This point is critical due to the high maneuverability demands of the control target. The adaptive gain variations for all the three controllers also converge successfully after encountering the mass drop.

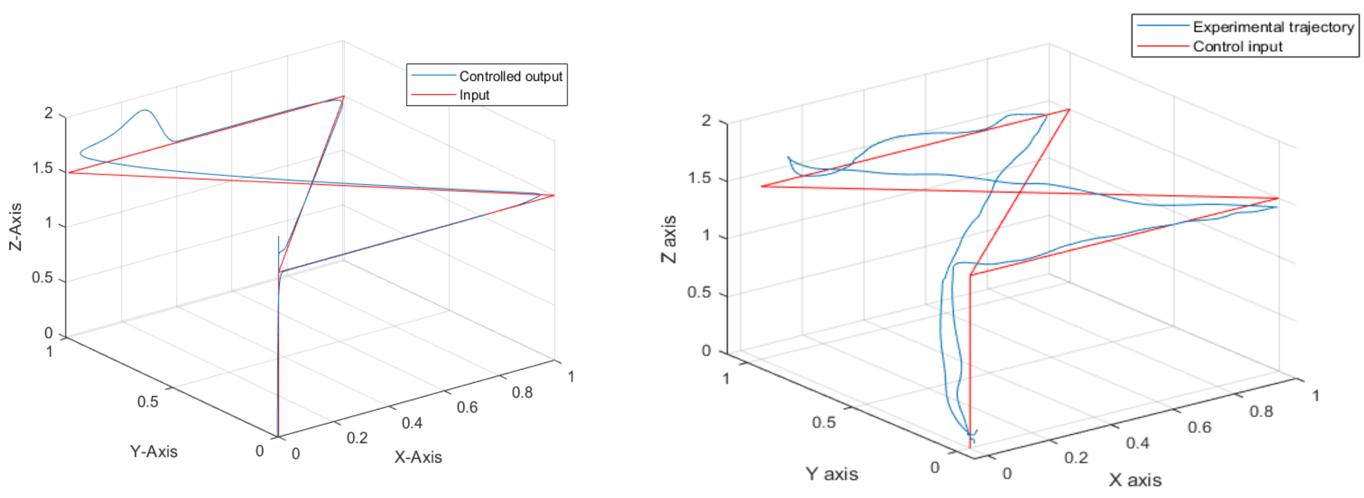

Figure 18: 3D Trajectory (Simulation and Experiment)

3. Conclusion

The results and analysis of this study show that compared to a classical PID controller, a better tracking performance was demonstrated by the designed adaptive controller in the X, Y, and Z axes in simulations as well as flight tests. The performance did not decline drastically in the real-time case and sustained all of its behavioral expectations. Consequently, on dropping the payload, a safe flight of the quadrotor UAV can be ensured. As part of the future scope of this project, the adaptive controller of the drone can be simulated taking into account external perturbations such as wind and the same tests can be conducted on the real drone as well.