

TABLE OF CONTENTS

	Page
CHAPTER 1 INTRODUCTION	3
1.1 Overview of Control	4
1.2 Literature Review and Motivation	5
1.3 Research Objectives.....	11
1.4 Originality of the Research and Contribution.....	12
1.5 Thesis Progress and Methodology	14
1.6 Mathematical Concepts.....	15
1.6.1 Lyapunov's direct method of stability	15
1.6.2 Sliding mode control.....	17
1.6.3 Perturbation compensation systems.....	19
CHAPTER 2 MODELLING SYSTEM AND APPROACH OF CONTROL	23
2.1 Introduction.....	23
2.2 Description and Movement of Quadrotor	24
2.3 Mechanical Model of the Quadrotor	27
2.3.1 Linear and angular velocity	28
2.3.2 Development of the mathematical model	30
2.4 General Structure of Quadrotor Control	35
2.5 Conclusion	36
CHAPTER 3 HIERARCHICAL PERTURBATION COMPENSATION SYSTEM WITH EXPONENTIAL REACHING LAW SLIDING MODE CONTROLLER IN A QUADROTOR.....	37
3.1 Introduction.....	38
3.2 Quadrotor Dynamics.....	41
3.3 Problem Statement.....	44
3.4 Perturbation Compensators.....	45
3.5 Hierarchical Perturbation Compensator.....	46
3.6 Integrated System Design	48
3.7 Simulation.....	51
3.8 Experimental Results	57
3.8.1 Real-time setup:	58
3.8.2 Practical implementation:	60
3.9 Conclusion	65

CHAPTER 4	THREE-LOOP UNCERTAINTIES COMPENSATOR AND SLIDING MODE QUADROTOR UAV CONTROL WITH EXPONENTIAL REACHING LAW	67
4.1	Introduction.....	68
4.2	Quadrotor Model.....	71
4.3	Three-Loop Uncertainty Compensator	74
	4.3.1 Main loop uncertainties compensator	74
	4.3.2 Three-loop Uncertainties Compensator	75
4.4	Boundedness of perturbation compensators	76
4.5	The Control System	77
4.6	Simulation	80
4.7	Practical Implementation	86
	4.7.1 Experiment setup	86
	4.7.2 Experimental results.....	89
4.8	Conclusion	93
CHAPTER 5	VISION BASED LEADER FOLLOWER APPROACH FOR UNCERTAIN QUADROTOR DYNAMICS USING FEEDBACK LINEARIZATION SLIDING MODE CONTROL (FLSMC)	95
5.1	Introduction.....	96
5.2	Quadrotor Dynamics.....	99
5.3	Problem Definition.....	102
5.4	Leader Position Visual Estimation.....	104
5.5	Control Design.....	105
5.6	Simulation	111
5.7	Conclusion	120
CHAPTER 6	POSITION AND ATTITUDE TRACKING OF UNCERTAIN QUADROTOR BASED ON NON-SINGULAR TERMINAL SUPER- TWISTING ALGORITHM	121
6.1	Introduction.....	122
6.2	Preliminaries	125
6.3	Controller Design	127
	6.3.1 Position controller design	128
	6.3.2 Attitude controller design	134
6.4	Numerical Simulations.....	137
6.5	Practical Implementation	145
6.6	Conclusion	151
	CONCLUSION AND FUTURE WORK	153

LIST OF PUBLICATIONS155

APPENDIX I PARROT “ROLLING-SPIDER” QUADROTOR157

APPENDIX II COMPARITIVE STUDY165

LIST OF BIBLIOGRAPHICAL REFERENCES.....167

LIST OF TABLES

	Page
Table 3.1	Quadrotor parameters.....52
Table 3.2	Error RMS comparison.....57
Table 4.1	Quadrotor parameters as used in the numerical simulations81
Table 4.2	Error RMS comparison.....86
Table 5.1	Quadrotor parameters.....111
Table 5.2	Root mean square of errors in three different cases.....114
Table 6.1	Physical parameters of the quadrotor.....137
Table 6.2	Proposed controller gains (simulation).....138
Table 6.3	Classical STA gains (simulation).....139
Table 6.4	Comparative results141
Table 6.5	Constants and gains (experiment).....146

LIST OF FIGURES

	Page
Figure 1.1	General view on systems proposed to solve perturbation problem9
Figure 1.2	The proposed control solution in this thesis11
Figure 1.3	Robotics challenges and thesis progress15
Figure 1.4	Principle of sliding mode control.....17
Figure 1.5	Perturbation compensation system20
Figure 1.6	The output without external disturbance.....21
Figure 1.7	The output (a) White noise is applied (b) The compensator is applied22
Figure 2.1	Quadrotor structure, forces, angles and frames23
Figure 2.2	Quadrotor hovering.....25
Figure 2.3	Quadrotor Yaw movement in two directions.....25
Figure 2.4	Quadrotor Roll and movement in y direction.....26
Figure 2.5	Quadrotor pitch and movement in x direction.....27
Figure 2.6	Control block diagram35
Figure 3.1	Quadrotor structure, forces, angles and frames42
Figure 3.2	HPC -ERLSM system.....47
Figure 3.3	Trajectory in 3D-with perturbation, using ERLSM without the HPC53
Figure 3.4	Position trajectory-with perturbation, using ERLSM without the HPC53
Figure 3.5	Control signals-with perturbation, using ERLSM without the HPC54
Figure 3.6	Trajectory in 3D-with perturbation, using ERLSM and the HPC54
Figure 3.7	Position trajectory-with perturbation, using ERLSM and the HPC.....55
Figure 3.8	Control signals-with perturbation, using ERLSM and the HPC.....55
Figure 3.9	The HPC signal.....56

Figure 3.10 Control signals- with perturbation and using the HPC-SM57

Figure 3.11 Implementation workflow.....58

Figure 3.12 (a) The trajectory in 3D (b) The trajectory in x-y61

Figure 3.13 Position and altitude trajectory62

Figure 3.14 Euler angles response.....62

Figure 3.15 Velocities of x and y63

Figure 3.16 Errors in position, altitude and orientation63

Figure 3.17 Control signals64

Figure 3.18 HPC signals.....64

Figure 4.1 Quadrotor structure, forces, angles and frames73

Figure 4.2 Three Loop Uncertainty Compensator block diagram.....75

Figure 4.3 Trajectory in 3D-with perturbation, without the TLUC82

Figure 4.4 Position and attitude-with perturbation, without the TLUC82

Figure 4.5 Control signals-with perturbation, without the TLUC.....83

Figure 4.6 Trajectory in 3D-with perturbation, using the TLUC.....83

Figure 4.7 Position and attitude-with perturbation, using the TLUC.....84

Figure 4.8 Control signals-with perturbation, using the TLUC85

Figure 4.9 The TLUC signal85

Figure 4.10 Implementation workflow.....87

Figure 4.11 Experimental trajectory.....89

Figure 4.12 Position and altitude trajectory90

Figure 4.13 Euler angles response.....90

Figure 4.14 Errors in position, altitude and orientation91

Figure 4.15 Velocities of x and y92

Figure 4.16 Control signals92

Figure 4.17 TLUC signals93

Figure 5.1 Quadrotor configuration, forces, inertial & body frames102

Figure 5.2 Different frames definition and leader relative position105

Figure 5.3 Control System block diagram.....107

Figure 5.4 Quadrotor following the leader quadrotor112

Figure 5.5 Translational trajectory113

Figure 5.6 Translation and Rotational error113

Figure 5.7 Velocity estimation114

Figure 5.8 The sliding surface in FLSMC.....115

Figure 5.9 The sliding surface in SM control.....116

Figure 5.10 Control signals for the FLSMC Control117

Figure 5.11 Control signals for the SM control.....117

Figure 5.12 Control signals for the FLSMC control with (*tanh*) function118

Figure 5.13 The sliding surface in FLSMC with (*tanh*) function.....119

Figure 5.14 Translation and Rotational error with (*tanh*) function119

Figure 6.1 Considered quadrotor structure, forces, angles and frames127

Figure 6.2 Architecture of the closed loop system.....127

Figure 6.3 3D trajectory tracking140

Figure 6.4 Position tracking141

Figure 6.5 Errors in position tracking142

Figure 6.6 Euler angles response via proposed approach142

Figure 6.7 Euler angles response via standard STA.....143

Figure 6.8 Errors in orientation143

Figure 6.9	Control Signals.....	144
Figure 6.10	Implementation workflow.....	146
Figure 6.11	Workspace trajectory tracking: (a) In 3D (b) In x-y	148
Figure 6.12	Position and altitude trajectory	149
Figure 6.13	Euler angles response.....	149
Figure 6.14	Velocities of x and y	150
Figure 6.15	Errors in position, altitude and orientation	150
Figure 6.16	Control signals	151

LIST OF ABBREVIATIONS

UAV	Unmanned aerial vehicles.
HPC	Hierarchical Perturbation Compensators.
FFPC	Feed-Forward Perturbation compensator.
FBPC	Feed-Back Perturbation compensator.
SMPC	Sliding Mode Perturbation compensator.
ERLSM	Exponential Reaching Law Sliding Mode.
TLUC	Three Loop Uncertainty Compensator.
ERSM	Exponential Reaching Law Sliding Mode.
TDE	Time Delay Estimation.
SOED	Second Order Sliding Mode Exact Differentiation estimator.
SMC	Sliding Mode Control.
SOSM	Second Order Sliding Mode.
STA	Super-Twisting Algorithm.

LIST OF SYMBOLS

F	Thrust force generated by each motor.
R	Rotation matrix.
η^I	Position vector in the inertial frame $\eta^I = [x \ y \ z]$.
x, y, z	The coordinates of the center of gravity in the inertial frame.
Θ^I	Angles vector in the inertial frame $\Theta^I = [\phi \ \theta \ \psi]$.
ϕ, θ, ψ	Euler angles, the roll, pitch and yaw in inertial frame respectively.
u	The control command.
X	State variable.
X_d	Desired state.
S	Sliding surface.
t	Time variable.
V	Lyapunov Function.
v^I	Linear velocity in the inertial frame.
v^B	Linear velocity in the body frame.
u, v, w	Linear velocities in X, Y and Z respectively in the body frame.
m	Total mass and g is gravity acceleration.
b	Lift coefficient.

d	Drag coefficient.
T_h	Drag in the propellers.
F_t	The drag along the axes.
$k_{ftx}, k_{fity}, k_{ftz}$	Translational drag coefficient in X, Y and Z axes.
$k_{fax}, k_{fay}, k_{faz}$	Coefficient of aerodynamic friction in X, Y and Z axes.
p, q, r	Angular velocities around x, y and z respectively in the body frame.
J_r	Rotor inertia.
I_x, I_y, I_z	Inertia in X, Y and Z respectively.

CHAPTER 1

INTRODUCTION

Unmanned aerial vehicles (UAVs) known as drones are very popular type of aircrafts. Research on UAVs has been increasing since the mid-1990s. A significant focus is placed on rotorcrafts primarily due to the variety of its possible applications. Four-rotor UAVs or quadrotors; equipped with electric motors and fixed-pitch propellers have gained most popularity. Quadrotors are controlled autonomously by an onboard microcontroller or by a remote controller in the base station. The quadrotor has a simple design and it is easy for maintenance, no mechanical linkages are required to vary rotors blade pitch angle as they spin, it has small four rotors, each has a smaller diameter and less kinetic energy when compared to an equivalent helicopter rotor. It has the ability to perform a vertical take-off and landing, fly with high maneuverability and at low speed. Quadrotors disadvantages include energy consumption due to the use of four motors that also gives more weight. Its control requires very precise and accurate rotor-speed changes, which makes it more suitable for electric motors. Large quadrotor engines with gearbox system that has slow response could not be satisfactory (Garcia, Lozano, & Dzul, 2006) when compared to a single rotor helicopter. Despite that, the number of possible applications using this type of UAV encouraged a lot of research in this field. The quadrotors are used in applications such as inspections and security mission, pipe/power line surveillance, real-estate mapping, traffic monitoring, disaster response and relief, infrastructure monitoring, agricultural applications, aerial photography, movie productions, sports events coverage, mining detection and fishery control and it can help in search and rescue missions.

There are many challenges and issues concerned controlling quadrotors. As most of existing nonlinear dynamic systems, accurate modeling of this type of robots is difficult to obtain. Because of quadrotors small size, it is sensitive to wind disturbance, air friction, uncertain/changing parameters and non-modelled dynamics. The mentioned problems reduce system performance and affect the control and trajectory tracking negatively.

1.1 Overview of Control

Quadrotors stability and trajectory tracking depend on the control of four propellers. Six degrees of freedom are controlled by four inputs. For this reason, it is considered as an under-actuated system and a highly coupled dynamic structure. These issues increase the complication of the control task. Moreover, complex applications which requires aggressive maneuvers arise the need for a robust control system.

Mechanical systems suffer from uncertainties. Examples of such problems can be seen in aircraft control where the change in air density at ground level compared to 30'000 ft altitude. The aerodynamics and control characteristics will change with altitude. In missile control, the change in mass and change in center of gravity is the major problem as fuel is consumed. Environmental effects and aging factor play additional role in plant parameters change.

Quadrotors are not an exception, one of the biggest problems in quadrotors is the uncertainties. This problem becomes worse when external disturbances are added. This problem will be referred as perturbation. Perturbation includes wind disturbance, nonlinear friction, inertial cross coupling, air friction, uncertain or changing parameters and non-modelled dynamics. Because quadrotors are small-in-size (relatively) and due to the lack of damping and the cross-coupling between degrees of freedom, the quadrotor is considered very sensitive to the perturbation. Perturbation affects system performance critically.

The aforementioned problem creates challenges in the control of robotics systems. Designing an auto adjustable nonlinear control and compensation for perturbation system to overcome such problems is required. There is a need to measure changes in parameters which occur within the dynamic system. Perturbation problems arise the need for a system to sense and correct itself whenever disturbances or change in parameters occur. Such a system needs to be designed so as to guarantee stability and robustness in the presence of disturbances and noise. The high speed of the adaptation algorithm is needed and computational cost in terms of time

and capacity has to be considered in practical implementations. Such system can be called self-organizing control, on-line perturbation-rejection system or perturbation-compensator control. Under-actuated robot systems come across coupled dynamic behavior which requires complex nonlinear control solutions. The quadrotor, as a robotic system, suffers from hard nonlinearities, unmodelled dynamics and external disturbance.

All robots are nonlinear dynamic systems. Nonlinear control systems are important to ensure stability. A lot of research was carried out to improve the quality of nonlinear control and to avoid any possible flaw. For example, Sliding mode control (SMC) is one of the most effective nonlinear control systems, it suffers from chattering phenomenon (Boiko, Fridman, & Iriarte, 2005; L. M. Fridman, 2001). This flaw is reduced by using Second Order sliding Mode (SOSM), Super Twisting Algorithm (STA) and Terminal Sliding Mode (TSM). A nonlinear control system is required to avoid the weakness of the mentioned controllers such as high chattering and the singularity problem, as well as improving convergence time and using lower gains.

1.2 Literature Review and Motivation

Perturbation problem in robotics is an important area of research. In order to eliminate or reduce its undesired effects, researchers used the following main approaches:

- 1) Building robust controllers that is able to handle the perturbation.
- 2) Building adaptive controllers, which has the ability to adapt to perturbation.
- 3) Designing observers to reduce uncertainties and disturbance.

Under the first approach, an “active disturbance-rejection” controller is designed to eliminate the impact of the state coupling and uncertainties for an autonomous quadrotor (Chang, Xia, Huang, & Ma, 2016; Sanz, Garcia, Zhong, & Albertos, 2016), a cascade control law is designed as a robust control (H. Liu, Zhao, Zuo, & Zhong, 2017), a backstepping controller is developed

(Cabecinhas, Cunha, & Silvestre, 2015). Fuzzy logic-based tracking controller is used (Kayacan & Maslim, 2017). However, the lack of adaptation property of such controllers reduces the performance. To solve this issue, other researchers developed adaptation functionality in their control. For example, adaptive control method is implemented to adjust disturbance and actuator failures (F. Chen, Lu, Jiang, & Tao, 2014), an adaptive output feedback compensator is used (Marino & Tomei, 2016a), an adaptive time-varying compensator is constructed for a quadrotor under uncertainties (Ton, McCourt, & Mehta, 2016) and prediction-based control is developed (Alexis, Nikolakopoulos, & Tzes, 2012). Despite the good performance of the above mentioned systems, they lack estimation and compensation of the perturbation during real time operations. In some control systems, uncertainties are represented by the unstructured uncertainty such as additive uncertainty which are lacking in phase information and whose upper bound of magnitude is assumed to cover the worst case of plant uncertainty. Thus, it inevitably includes a class of plants, which may practically never happen (S. J. Kwon & W. K. Chung, 2004). Therefore, these controllers are designed with high-gain, which makes a control system very conservative in performance although they guarantee robust stability for the assumed plant uncertainty.

Hence, many observers were designed to solve this problem, such as (X. Wang, Shirinzadeh, & Ang, 2015; Yin & Xiao, 2017), Luenberger observer is used with feedback linearization (Mokhtari, M'Sirdi, Meghriche, & Belaidi, 2006), disturbance observers as a part of the control is used (F. Chen, Lei, Zhang, Tao, & Jiang, 2016a), a sliding mode-based disturbance observer is designed (Lénaïck Besnard, Yuri B Shtessel, & Brian Landrum, 2007; Besnard, Shtessel, & Landrum, 2012; Zhang, Sun, Zhang, & Zhou, 2013), an acceleration-based observer is built for attitude control (Jeong, Jung, & Tomizuka, 2012b) and an extended observer with feedback sliding mode is used (Rongting Zhang, Quan Quan, & K-Y Cai, 2011). The existing perturbation observer systems guarantee good performance in the quadrotor, even though; most of them lack multilevel tracking of perturbation and they suffer from some drawbacks such as the influence of measurement lag and sensor noise. The aforementioned state-of-art can be summarized as in Figure 1.1.

In an effort to overcome the mentioned drawbacks, this study focuses on integrating a system of Hierarchical Perturbation Compensators (HPC) with a control system (S. Kwon & W. K. Chung, 2004). The HPC comprises three subsystems to provide estimation and compensation hierarchically. The first subsystem is built to provide estimation of perturbation based on the desired dynamics, which is lag-free, and noise-free signals, meanwhile the second subsystem is built to provide estimation based on the real dynamics. In order to track perturbations in different levels, the third subsystem compensates for the sliding mode dynamic error. The proposed HPC estimates and compensates perturbation simultaneously based on time delay estimation. The HPC has adaptive control property as it generates control effort that is required to compensate the current perturbation. In addition, it has an integral control feature as the current compensation value is estimated based on one-step delayed input. One of the concerns to consider in implementing compensators/observers is the addition of new dynamics to the system, which might as well increase the computational burden on a limited onboard computing ability of small-sized robots. The proposed system decreases computational burden through using the HPC.

The Hierarchical Perturbation Compensator HPC is detailed in Chapter three, despite its great advantages in attenuating perturbation, there is an unavoidable estimation error, if we assume ideal sensors, the estimation error is $\tilde{\Gamma}(t) = \Gamma(t) - \Gamma(t - \tau)$ (where $\Gamma(t)$ is the perturbation, (t) is the time, (τ) is the sampling time) resulting from one step delayed input/output. Consequently, the performance of HPC depends on the norm $\|\tilde{\Gamma}(t)\|$. A satisfactory performance is possible on a condition that perturbation is continuous and differentiable and doesn't vary greatly during a small period of time (τ) , which is a reasonable assumption in most observer applications. For that reason, an efficient method to further attenuate error variations and to enhance the performance of the controlled system is required.

With an endeavor desire to overcome such concerns, this study implements Three-Loop Uncertainty Compensator (TLUC) in order to track uncertainties in three loops. The loops have the ability to track perturbation and residual perturbation. Each loop provides estimation and compensation of perturbation simultaneously based on time delay estimation. The TLUC has

adaptive control property as it generates control effort that is required to compensate the current perturbation. Furthermore, the TLUC has an integral control feature as the current compensation value is estimated based on delayed input. The gyroscopic term in the model cannot be measured in quadrotors because real angular velocities are not measured. On the other hand, rotor inertia J_r is very small. This value is estimated as unmodelled dynamics and compensated in the proposed compensator systems. Perturbation is considered as the unmodelled dynamics and uncertain parameters. In the real experiment an external wind source is added.

On the other hand, the quadrotor has a highly coupled dynamic structure. One of the most popular techniques used to resolve the problem of the nonlinear decoupling is feedback linearization (FL) (Slotine & Li, 1991). FL in general is aimed to transform algebraically nonlinear systems into an equivalent linear one in closed loop in order to avoid complex nonlinear control solutions and to reduce the effect of highly coupled dynamics. This technique is employed to address some practical control problems. In spite of that, the hard nonlinear parameters and/or uncertainties of the system do not permit conventional linear controls to provide a high level of accuracy (Slotine & Li, 1991). Actually, control of hard nonlinearities and uncertainties in nonlinear dynamics is an interesting topic of nonlinear control engineering. Numerous nonlinear control systems have been designed to overcome the effect of the nonlinearities and nonlinear uncertainties. A manipulator system simplifies the control law to become linear for joint decoupling is designed (C. Fallaha & Saad, 2018), H_∞ control system is built (Xiangjian, Kun, & Di, 2016) and a robust nonlinear H_∞ controller takes into account the uncertainties in a quadrotor (Jasim & Gu, 2018). On the other hand, sliding mode control (Kurode & Dixit, 2013; J. Mu, Yan, Spurgeon, & Mao, 2017; Slotine & Li, 1991; Xia, Zhu, & Qi, 2010; Youcef-Toumi & Ito, 1988), which is one of the most attractive control techniques, shows lower performance if the system suffers from hard nonlinearities. Integral backstepping combined with sliding mode control is built to provide robustness to external disturbances (Jia et al., 2017).

In our proposed system, an auxiliary approach is used to support the control by estimating and compensating all disturbances. The control system can deal with nonlinearity without linearizing the model. It just makes use of the general structure of the feedback linearization and based on adaptive sliding mode control. The system reduces the effect of the hard nonlinearity and the highly coupled dynamics and to provide a robust and an accurate control, (Figure 1.2).

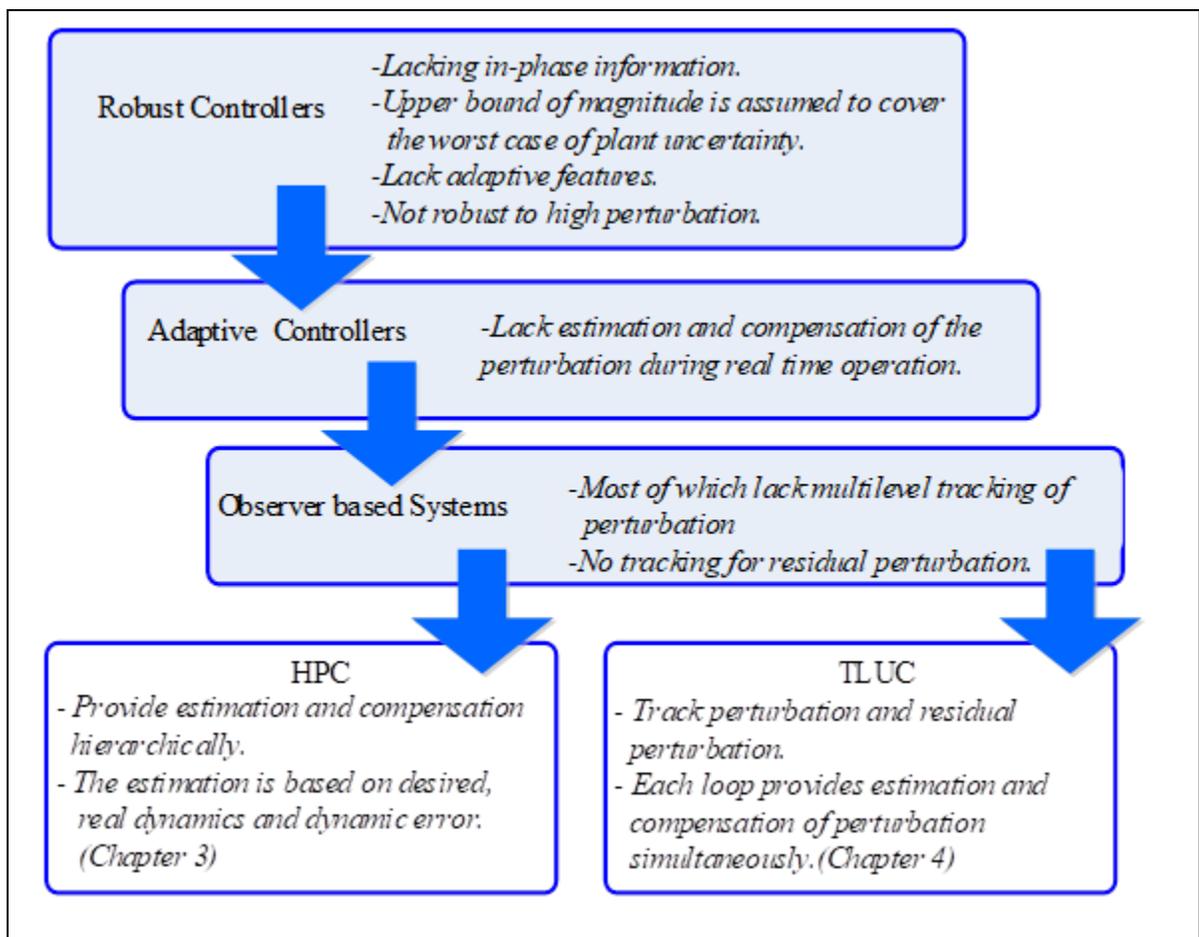


Figure 1.1 General view on systems proposed to solve perturbation problem

Despite that Sliding Mode Control (SMC) is one of the most effective nonlinear control systems, it has an obstacle that represents its major disadvantage, that is chattering phenomenon (Boiko et al., 2005; L. M. Fridman, 2001). The chattering phenomenon results in undesirable performance, damage to mechanical parts in the system, heat and energy loss. In

order to avoid chattering many approaches have been proposed (Ali, Samar, Shah, Bhatti, & Munawar, 2017; Hwachou Chen, Chen, & Xu, 2019; Kali, Saad, & Benjelloun, 2019; Kali, Saad, Benjelloun, & Khairallah, 2018; Razmi & Afshinfar, 2019; Y. Wang, Li, Yan, & Chen, 2019). In recent years, Second Order Sliding Mode (SOSM) control has been widely studied for a class of second-order nonlinear systems and has been considered as a good solution to reduce chattering (Bartolini, Pisano, Punta, & Usai, 2003; Levant, 1993). In practical problems, SOSM control has been successfully implemented in many nonlinear systems as robotic manipulators (Azar, Serrano, Vaidyanathan, & Albalawi, 2019; Kali, Saad, Benjelloun, & Fatemi, 2017), induction machine drives (Benderradji, Benamor, Chrifi-Alaoui, Bussy, & Makouf, 2012; Kali, Rodas, et al., 2017), energy systems (Krim, Abbas, Krim, & Mimouni, 2018; Merabet, Labib, Ghias, Aldurra, & Debbouza, 2019) and others. However, the design of SOSM control law requires the measurement of the first time derivative of the designed sliding surface, which is in many cases not available. Thus, this problem makes the implementation difficult.

As a solution, Super-Twisting Algorithm (STA) has been proposed (Guzmán & Moreno, 2015; Moreno, 2014; Moreno & Osorio, 2008). In addition to the fact that STA is a robust approach that produces less chattering and ensures fast finite time convergence, STA does not need the derivative of the sliding surface. The STA has been implemented for attitude tracking of quadrotor UAV system (Derafa, Benallegue, & Fridman, 2012). However, the convergence time during the sliding phase depends on the designed switching surface. If the latter is not well selected, unacceptable or undesirable performance might be obtained.

In the literature, a terminal sliding surface that is nonlinear has been proposed to improve the convergence time during the sliding phase (Feng, Yu, & Man, 2002). In spite of that, it suffers from the problem of singularity that has been covered as a nonsingular terminal sliding mode (Feng, Yu, & Han, 2013). However, the chattering phenomenon increases with the use of this nonlinear switching surface. Moreover, to the best of the authors' knowledge, all the developed STA control systems use classical linear sliding surface because the use of STA-based the nonsingular terminal sliding surface complicates the stability analysis and might increase the chattering.

Inspired by the above-mentioned published papers and by the good features of second order sliding mode, we propose a position and attitude tracking based on super-twisting control algorithm with a new non-singular terminal sliding surface that proposes a solution to the well-known singularity problem.

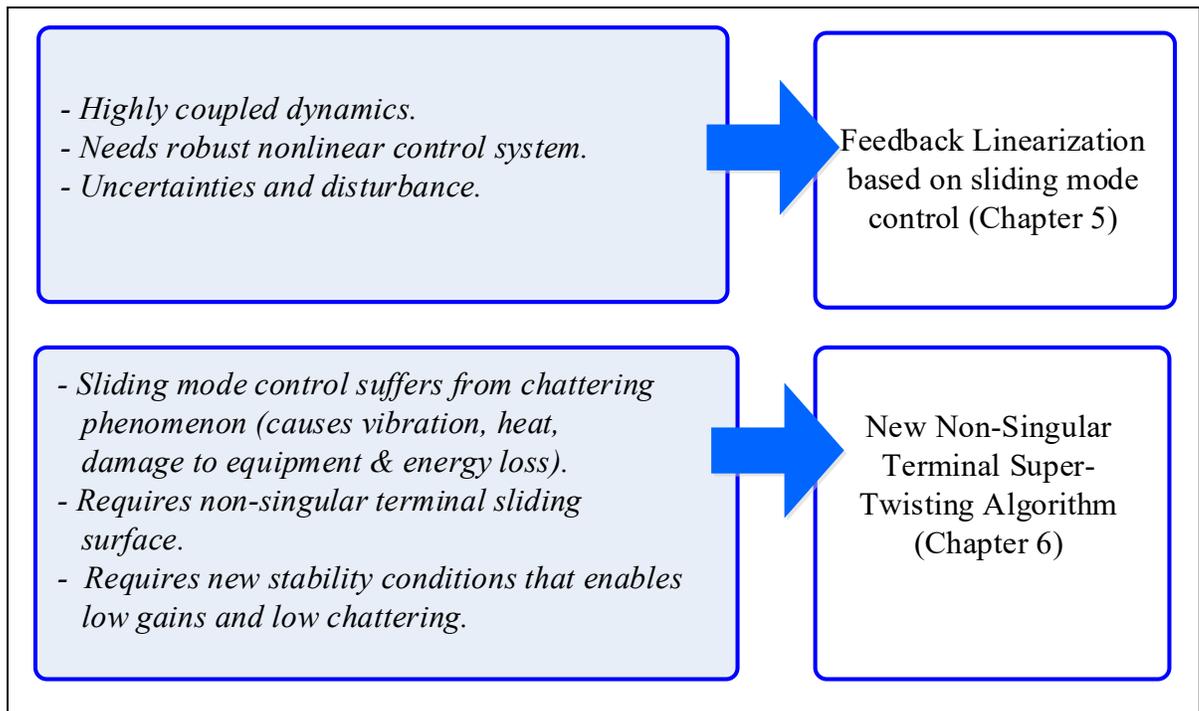


Figure 1.2 The proposed control solution in this thesis

1.3 Research Objectives

Motivated by the problems mentioned above, the main objectives in this research are summarized as follows:

- Build and test Perturbation Compensator systems applied to robots with the ability to provide precise error tracking in the presence of uncertainties and disturbances, with a

focus on estimating and compensating perturbation. The compensators and the controllers are required to be easily implemented in quadrotor aircrafts.

- Design a control system capable of dealing with nonlinearity without linearizing the model. The system should reduce the effect of the hard nonlinearity and the highly coupled dynamics and to provide a robust and an accurate control.
- Design and test a new non-singular terminal sliding surface control to avoid the singularity problem in the classical STA. In addition to provide robust control with low chattering.

1.4 Originality of the Research and Contribution

The research in this thesis concentrate on the development of perturbation compensation systems and nonlinear control laws to ensure the stability of a quadrotor robot. Following the literature review, despite the fact that a lot of researchers have studied perturbation and disturbance in robotic systems, some important points need further study. In contrast with the cited research in the literature in section 1.2, this thesis enriches the knowledge in the robotics field through the following contributions:

Article 1: By using the Hierarchical Perturbation Compensator (HPC) estimation and compensation in addition to the exponential reaching law sliding mode control, the contributions of this paper can be described as:

- Hierarchical Perturbation Compensator (HPC) system is built and applied to a six degree-of-freedom under-actuated robot, quadrotor.
- The combined system of the HPC and the ERLSM provide higher upper bound of perturbation compensation magnitude, this gives more ability to attenuate higher perturbation.

Article 2: By using the Three-Loop Uncertainties Compensator (TLUC) and sliding mode control with Exponential Reaching Law, the contributions of this paper can be described as:

- Design a Three-Loop Uncertainty Compensator (TLUC) in order to track perturbation and residual perturbation in three loops. Each loop provides estimation and compensation of perturbation simultaneously based on time delay estimation.
- The adaptive and integral features of the TLUC give the system the ability to provide real-time estimation and compensation of uncertainties and disturbance.

Article 3: By using the Feedback Linearization based on sliding mode control, the contribution of this paper can be summarized as follows:

- Design a control system capable of dealing with nonlinearity without linearizing the model. It makes use of the general structure of the feedback linearization and based on adaptive sliding mode control. The system reduces the effect of the hard nonlinearity and the highly coupled dynamics to provide a robust and an accurate control.
- Afford accurate, continuous, bounded and smooth estimation of velocity and acceleration of the leader to provide a reference trajectory to the follower by applying Second Order Sliding Mode Exact Differentiation estimator (SOED), which is also able to reduce noise and chattering phenomenon.

Article 4: By using the Position and Attitude tracking of Uncertain Quadrotor UAV based on New Non-Singular Terminal Super-Twisting Algorithm, The contribution of this paper is an extension and improvement of the earlier mentioned control method in the following two aspects:

- It provides better comprehensive performance by proposing a new non-singular terminal sliding surface that uses an exponent that switches between two values to bypass the problem of singularity.

- In conventional STA approach, the gain must be chosen large to overcome the effects of the unmodelled dynamics. In our work, a new stability condition that will allow a small choice of gain while keeping good performances is established using Lyapunov theory. Hence, less chattering will be ensured.

1.5 Thesis Progress and Methodology

Robots working in application projects to serve humans involve change of parameters, for example, load/unload operations cause uncertain change in mass and inertia. Furthermore, uncertain or unmodelled parameters or dynamics have negative effect on the performance. The articles in this thesis are studying nonlinear control systems and perturbation compensation systems, (Figure 1.3). Robust nonlinear control systems related to this thesis are “Position and Attitude tracking of Uncertain Quadrotor UAV based on New Terminal Super-Twisting Algorithm”, “Vision based Leader Follower Approach for Uncertain Quadrotor Dynamics Using Feedback Linearization Sliding Mode Control (FLSMC)” and “Multivariable Super-Twisting Control in a Vision based Quadrotor Utilized in Agricultural Application”. Perturbation estimators are “Hierarchical Perturbation Compensation System with Exponential Reaching Law Sliding Mode Controller in a Quadrotor”, “Three-Loop Uncertainties Compensator and Sliding Mode Quadrotor UAV Control with Exponential Reaching Law” and “Adaptive Control Based on RBF Neural Network Approximation in a Quadrotor”.

Tuning the controllers in a six-degree of freedom robot is a long process. In the quadrotor, we choose the gains based on the stability conditions and we try to keep them as low as possible provided that they achieve good tracking. Altitude controller is tuned first since its performance affects all other degrees of freedom. The second step is to tune the controller of the roll and pitch angles; because of the symmetrical structure of the quadrotor both will have close tuning values. The third step is to tune the yaw angle controller and in the end we tune the position controller in (x, y) .

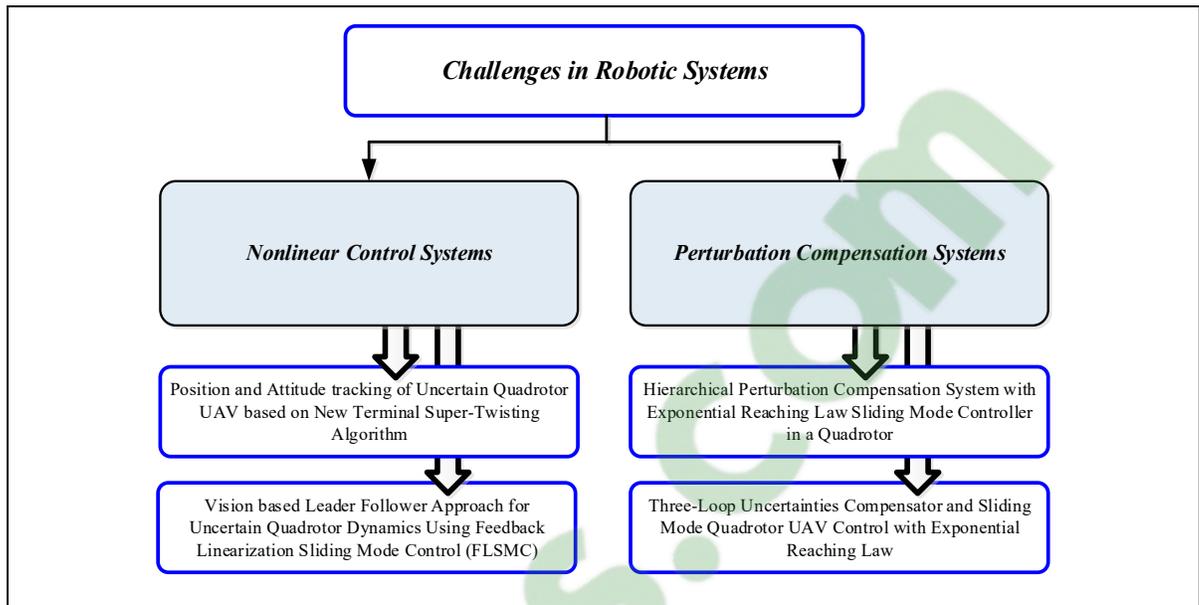


Figure 1.3 Robotics challenges and thesis progress

1.6 Mathematical Concepts

1.6.1 Lyapunov's direct method of stability

In this section, we look at stability theory in the sense of the “Lyapunov”, a Russian mathematician and engineer who put forward this theory which carries his name. He built his philosophy on two methods to study systems stability. The first method called indirect method is based on power series expansions which does not find much favor today. The second method is known as direct method, the basic principle of this method is the physical behavior of the system, if the mechanical or electrical energy in the system is decreasing continuously, then we can predict that the system eventually settles down whether it is linear or nonlinear to an equilibrium point. Therefore the stability of can by examined by analyzing a proper Lyapunov function of the system (Slotine & Li, 1991). Let us consider a dynamic system given as:

$$\dot{x} = f(x, t) \quad (1.1)$$

This system is considered to satisfy $x(t_0) = x_0$, $x \in R^n$. We will consider $f(x, t)$ is Lipschitz continuous with respect to x , uniformly and continuous in t . If point x^* satisfies $f(x^*, t) = 0$ then the point x^* is an equilibrium point (Murray, 2017). Roughly speaking, if all solutions which start near x^* remain near x^* for all time, an equilibrium point is considered locally stable. If x^* is locally stable and all the solutions starting near x^* tend towards x^* as $t \rightarrow \infty$, then the equilibrium point x^* is considered to be locally asymptotically stable.

By shifting the origin of the system, we may assume that the equilibrium point of interest occurs at $x^* = 0$. If multiple equilibrium points exist, we will need to study the stability of each by appropriately shifting the origin. In the sense of Lyapunov, the equilibrium point $x^* = 0$ is stable at $t = t_0$ if for any $\epsilon > 0$ there exists a $\delta(t_0, \epsilon) > 0$ such that (Murray, 2017):

$$\|x(t_0)\| < \delta \Rightarrow \|x(t)\| < \epsilon, \quad \forall t \geq t_0 \quad (1.2)$$

Previous definitions describe the behavior of a system near an equilibrium point. The equilibrium point x^* is globally stable if it is stable for all initial conditions $x_0 \in R^n$. Global stability is very desirable but in many applications it can be difficult to achieve.

It is important to note that the definitions of asymptotic stability do not quantify the rate of convergence. There is a strong form of stability which demands an exponential rate of convergence. The equilibrium point $x^* = 0$ is an exponentially stable equilibrium point if there exist constants $m, \alpha > 0$ and $\epsilon > 0$ such that:

$$\|x(t)\| \leq m e^{-\alpha(t-t_0)} \|x(t_0)\| \quad (1.3)$$

For all $\|x(t_0)\| \leq \epsilon$ and $t \geq t_0$. The constant α is called the rate of convergence. Exponential stability is a strong form of stability, it implies uniform and asymptotic stability.

Exponential convergence is important in applications because it can be shown to be robust to perturbations and it is essential for the consideration of more advanced control algorithms,

such as adaptive ones. A system is globally exponentially stable if the bound in (1.1) holds for all $x_0 \in R^n$ (Murray, 2017).

1.6.2 Sliding mode control

Sliding mode control (SMC) is a control method that alters the dynamics of a system by multiple control structures designed so as to ensure that trajectories always move towards a switching condition. The control law switches from one continuous structure to another based on the error and its derivative. The control is designed to guarantee that trajectories move towards the switching condition. The ultimate trajectory will slide along the boundaries of the control structures. The motion of the system as it slides along these boundaries is called a sliding mode. The geometrical locus consisting of the boundaries is called the sliding surface (J. Liu & Wang, 2012a), (Figure 1.4). The sliding surface is described by $s = 0$, and the sliding mode along the surface commences after a finite time when system trajectories have reached the surface (J. Liu & Wang, 2012a).

Sliding mode based on reaching law includes reaching phase and sliding phase. The reaching phase drive system is to maintain a stable manifold and the sliding phase drive system ensures slide to equilibrium. The idea of sliding mode can be described as in Figure 1.4.

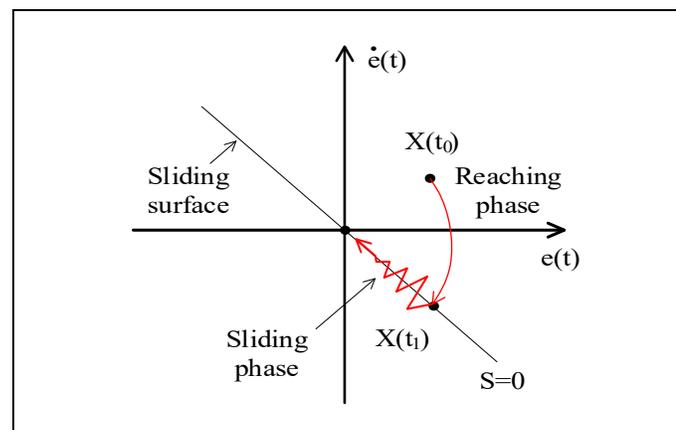


Figure 1.4 Principle of sliding mode control

The structure of SMC law $U(t)$ is based on two main parts; a continuous part $U_c(t)$ and a discontinuous part $U_d(t)$. That is (Sivaramakrishnan, Hemavathy, & Anitha, 2017):

$$U(t) = U_c(t) + U_d(t) \quad (1.4)$$

The continuous part of the controller that maintains the output of the system restricted to the sliding surface, the continuous part of SMC is given as:

$$U_c = f(R(t), Y(t)) \quad (1.5)$$

Where $R(t), Y(t)$ are the reference and the controlled value respectively. The discontinuous part of the SMC, $U_d(t)$ comprises the switching element of the control law. The SMC aims to make the error and its derivative go to zero. The sliding surface function is given as:

$$s(t) = \left(\frac{d}{dt} + \lambda\right)^{n-1} e \quad (1.6)$$

Where e is the error, λ is the slope of the sliding surface. For a second order system $n = 2$, we have:

$$s(t) = \dot{e} + \lambda e \quad (1.7)$$

The discontinues control law of the SMC can be given in different methods, the simplest is as follows (Slotine & Li, 1991):

$$U_d(t) = -k \operatorname{sign}(s) \quad (1.8)$$

Where the parameter k is a positive constant and it is responsible for the reaching mode. The discontinuous switching function causes oscillations around the desired equilibrium point. This

undesired oscillation is known as chattering phenomenon. The latter causes vibration and overheat which may damage the mechanical parts.

Higher values of k cause higher chattering while low values cause slow reaching phase. In order to solve this dilemma, a dynamic variable can be given to the constant k . Power Rate Reaching Law is proposed (J. Liu & Wang, 2012a) and exponential reaching law (C. J. Fallaha, Saad, Kanaan, & Al-Haddad, 2010), in these methods, the constant k takes high values when the error is high and takes low value when the error is small. The variation of k ensures quick convergence and avoids high chattering.

1.6.3 Perturbation compensation systems

Advanced nonlinear control methods enable to scientifically design stabilizing controllers which meet robust stability and performance on the plant uncertainty. In such control systems, uncertainties are represented by the unstructured uncertainty such as additive uncertainty which are lacking in phase information and whose upper bound of magnitude is assumed to cover the worst case of plant uncertainty. Thus, it inevitably includes a class of plants, which may practically never happen (S. J. Kwon & W. K. Chung, 2004). Therefore, these controllers are designed with high-gain, which makes a control system very conservative in performance although they guarantee robust stability for the assumed plant uncertainty. As a solution, perturbation compensators are investigated. Perturbation compensator can be regarded as a “model regulator” which drives the physical plant with uncertainty to the nominal model.

It is necessary to reduce the system’s sensitivity to the perturbation by applying an additional perturbation compensator as well as the nominal feedback controller. Consider we have the following second-order system:

$$\begin{aligned}\dot{x}_1 &= \dot{x}_2 \\ \ddot{x}_2 &= u + \Gamma(t)\end{aligned}\tag{1.9}$$

Where x_1, x_2 are the system states, $\Gamma(t)$ is the perturbation that includes nonlinear friction, uncertain dynamics, unmodelled dynamics and unpredictable external disturbances. The real perturbation in (1.9) can be equivalently expressed as:

$$\Gamma(t) = \ddot{x}_2(t) - u(t) \quad (1.10)$$

It is required to generate equivalent compensation based on time delay as:

$$\hat{\Gamma}(t) = \Gamma_{eq}(t - \tau) = \ddot{x}_2(t - \tau) - u(t - \tau) \quad (1.11)$$

Where τ is the time constant. The estimated perturbation is added to the control as in Figure 1.5, this can be expressed as:

$$\begin{aligned} \ddot{x}_2 &= u + \Gamma(t) - \hat{\Gamma}(t) \\ \ddot{x}_2 &= u + \tilde{\Gamma}(t) \end{aligned} \quad (1.12)$$

Where $\tilde{\Gamma}(t) = \Gamma(t) - \hat{\Gamma}(t)$ is perturbation error. The perturbation in the system is reduced significantly from $\Gamma(t)$ to $\tilde{\Gamma}(t)$.

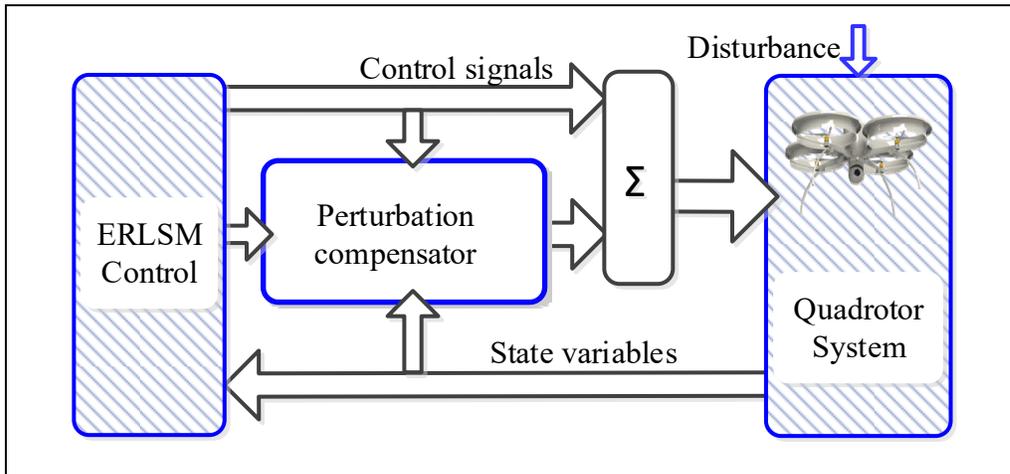


Figure 1.5 Perturbation compensation system

If the controller in this system is selected as:

$$u = -k_2 e_2 - k_1 e_1 \quad (1.13)$$

Where $e_2 = x_2 - x_{2d}$ and $e_1 = x_1 - x_{1d}$ are the errors, k_2 and k_1 are tunable gains, x_{2d} and x_{1d} are the desired states. To the system described in (1.9) and the control system given in (1.13), the normal response of the system is displayed in Figure 1.6. It proves the control ability to stabilize the system. A white noise of power equals 0.1 is added in order to verify the performance of the compensator.

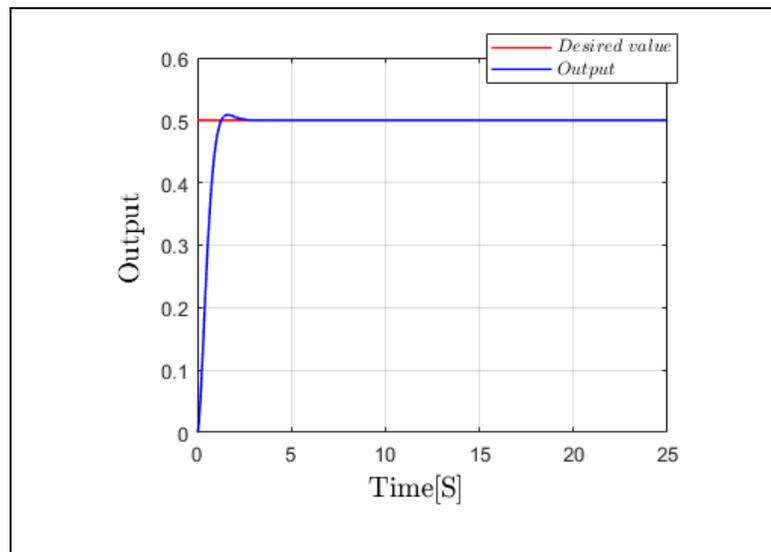


Figure 1.6 The output without external disturbance

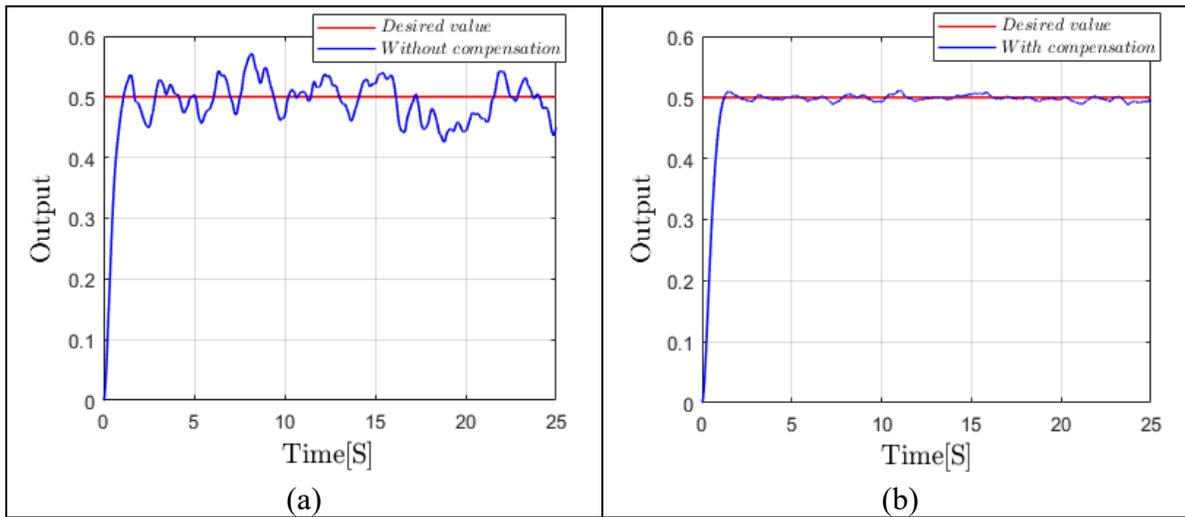


Figure 1.7 The output (a) White noise is applied (b) The compensator is applied

The system is disturbed by the white noise as seen in Figure 1.7-a. After applying the compensator, the performance is improved considerably as seen in Figure 1.7-b.

The perturbation compensator provides great performance as it provides continuous estimation and compensation of perturbation. Compensators have different structures like the Hierarchical Perturbation compensator (HPC) as in Chapter 3 and the Three Loop Perturbation Compensator (TLUC) as in Chapter 4. Convergence time of errors in systems which are subject to bounded perturbation and uncertainty is analyzed and proved in (C. J. Fallaha, Saad, Kanaan, & Al-Haddad, 2011a).

CHAPTER 2

MODELLING SYSTEM AND APPROACH OF CONTROL

2.1 Introduction

The quadrotor is classified as one of the most complex robotic systems due to the number of physical effects, forces and moments that affect its dynamics such as aerodynamic effects, gravity, gyroscopic effects, friction and moments of inertia (Samir Bouabdallah, 2007b; Hwangbo, Sa, Siegwart, & Hutter, 2017; Powers, Mellinger, & Kumar, 2015; Richter, Bry, & Roy, 2016). This type of rotorcraft achieves stable hovering and precise trajectory tracking by balancing the forces produced by the four rotors. Quadrotor configurations, frames, and forces are shown in Figure 2.1. In order to design a flight controller, the movements of the aircraft and its dynamics must be clearly understood. This understanding is necessary not only for the design of the controller, but also to ensure that the simulation of the vehicle behavior is closer to reality when the control is applied.

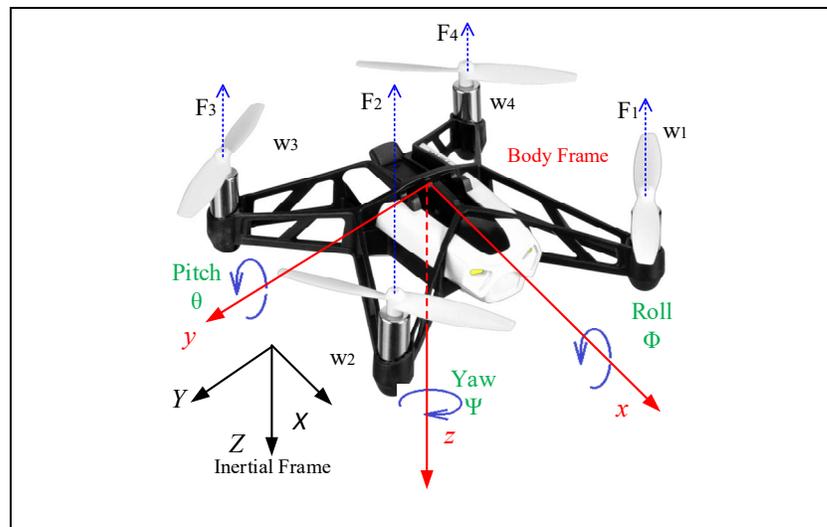


Figure 2.1 Quadrotor structure, forces, angles and frames
("Parrot Minidrone," 2018)

2.2 Description and Movement of Quadrotor

A quadrotor is an aerial mobile robot with four rotors defined in space by six degrees of freedom. The four rotors are placed at the ends of a cross, and the electronics parts are placed in the center. Each two opposite propellers rotate in the same direction, and the other two propellers rotate in the other direction in order to prevent the quadrotor from spinning around z axis. Moving the quadrotor is performed by varying motor speeds, it can be moved up / down, tilted left / right (roll) or forward / backward (pitch) or rotate around itself (yaw). The quadrotor has six degrees of freedom, three translational movements and three rotational movements. The six degrees must be controlled using only four actuators, therefore the quadrotor is known to be as an under-actuated system. In conventional helicopters, when the main rotor rotates, it produces a reactive torque that would cause the helicopter's body to turn in the opposite direction. This is usually balanced by adding a tail rotor that produces a thrust into a lateral direction. However, this rotor with its associated power supply does not provide thrust. In the quadrotor, basic movements are achieved by varying the speed of each rotor thereby changing the thrust produced. The quadrotor inclines towards the slower rotor direction, which then makes translation along the corresponding axis. Therefore, the motion is coupled, meaning that the quadrotor cannot perform the translation without rolling or pitching, which means that a change in the speed of a rotor translates into a movement.

In order to hover, all the thrust forces should work in the same speed (Figure 2.2), increasing thrust leads to upward movement along z axis with a magnitude exactly opposite to the gravitational force. Moreover, the thrust force created by each rotor must be equal to prevent the vehicle from tilting. Therefore, the thrust produced in each rotor must be identical. The upward and downward movement is obtained by the variation of the speed of rotation of the motors (consequently the thrust produced), if the thrust force is greater than the weight of the quadrotor, the movement is ascending, and if the lift force is lower than the weight of quadrotor movement is descending.

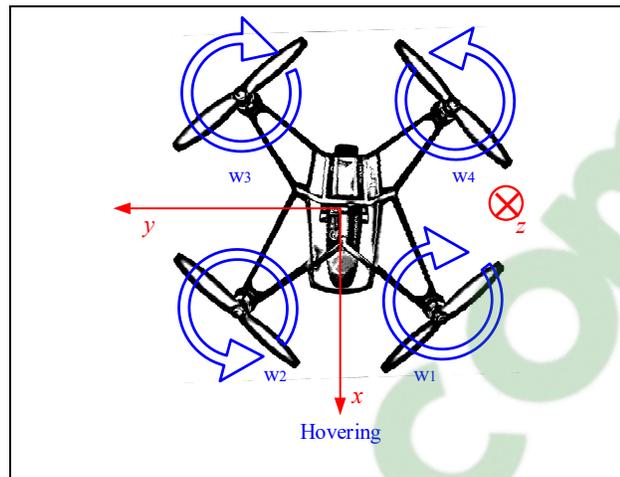


Figure 2.2 Quadrotor hovering

The rotation around z axis is known as Yaw (Figure 2.3) by applying a speed difference between rotors (1, 3) and rotors (2, 4) a torque is generated around z axis, either clockwise or counterclockwise. The direction of the thrust force does not shift during movement. Increasing thrust forces on one pair of rotors must be equal to the decrease of the other pair of rotors and so the total thrust force remains the same.

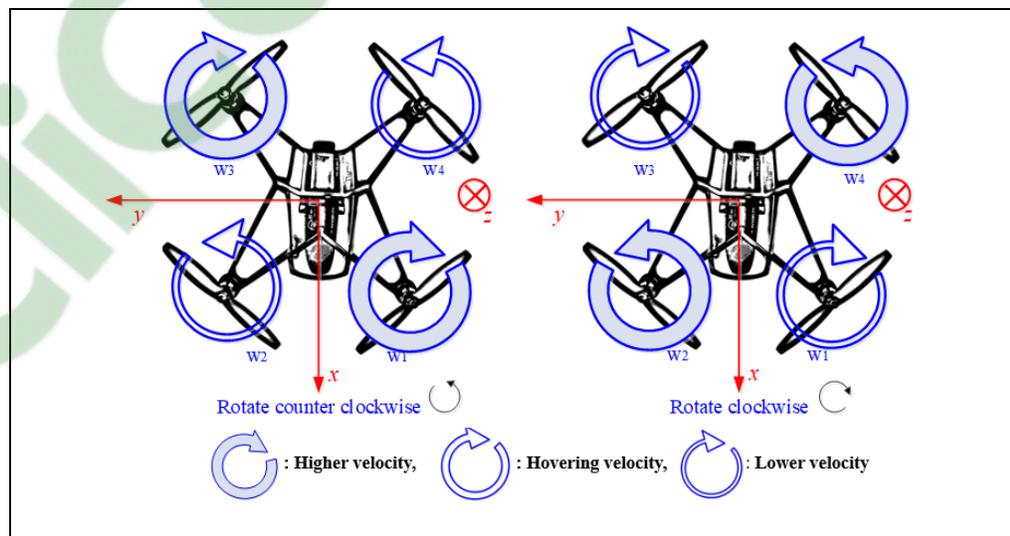


Figure 2.3 Quadrotor Yaw movement in two directions

The angle of rotation around x axis is known as Roll angle. This movement is coupled with a movement of translation along y axis. Applying differential thrust causes the quadrotor to roll and to move in y direction accordingly as it can be seen in Figure 2.4. Higher thrust on motors (1, 4) will cause the quadrotor to move towards y^+ axis, in the same way, higher thrust on motors (2, 3) will cause the quadrotor to move towards y^- axis.

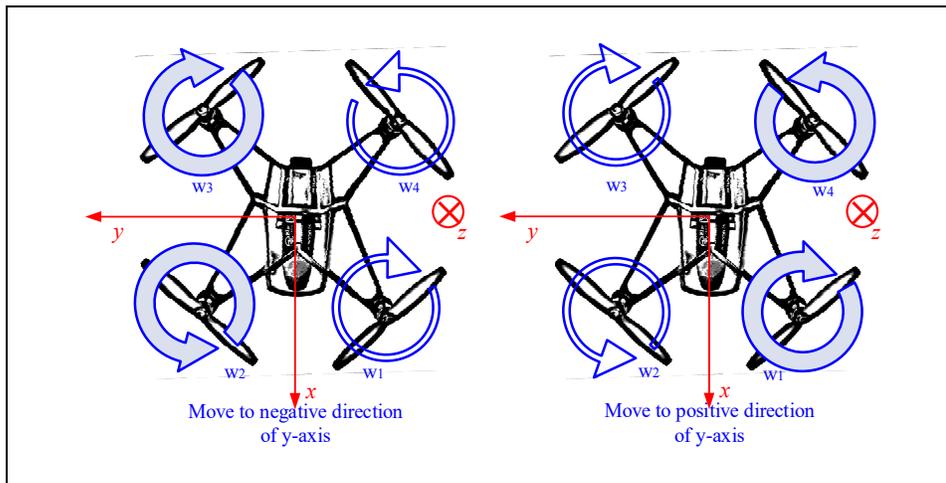


Figure 2.4 Quadrotor Roll and movement in y direction

The angle of rotation around y axis is known as Pitch. This movement is coupled with a movement of translation along x axis. Applying differential thrust causes the quadrotor to pitch and to move in x direction accordingly as it can be seen in Figure 2.5. Higher thrust on motors (3, 4) will cause the quadrotor to move towards x^+ axis, in the same way, higher thrust on motors (1, 2) will cause the quadrotor to move towards x^- axis.

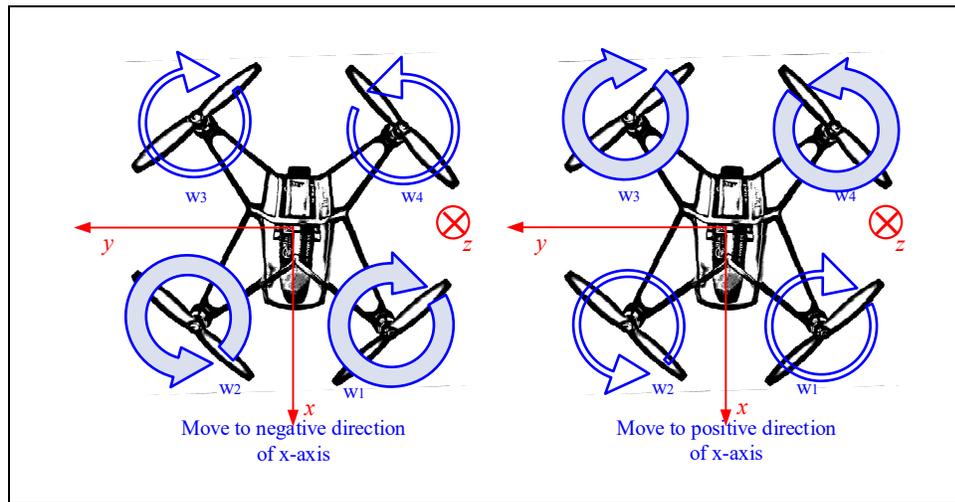


Figure 2.5 Quadrotor pitch and movement in x direction

It can be concluded that horizontal translation is performed by tilting the quadrotor in angular rotations Roll, Pitch and Yaw. These angular rotations are caused by a combination of different thrust forces in the rotors.

2.3 Mechanical Model of the Quadrotor

The quadrotor is a complex mechanical system, it goes under many internal and external physical effects in the aerodynamic and the mechanic domains. The model of the quadrotor should consider all important forces and moments including the gyroscopic effects. The model developed in this thesis assumes the following (Samir Bouabdallah, 2007b; S. Bouabdallah & Siegwart, 2007):

- The structure of the quadrotor is rigid and symmetrical, which induces that the matrix of inertia is diagonal.
- The center of gravity and the body fixed frame origin are aligned.
- The propellers are supposed to be rigid so as to neglect the effect of their deformation during the rotation.

- Thrust and drag forces are proportional to the square of the speed of rotation (ω^2), which is an approximation that is very close to the aerodynamic behavior.

To derive the mathematical model of the quadrotor we use two reference frames, the first is the Inertial frame $I (O^I, X^I, Y^I, Z^I)$, where O^I is the axes origin. The inertial frame is the fixed reference to the earth. The second frame is the quadrotor Body frame $B (O^B, X^B, Y^B, Z^B)$. Where O^B is the axis origin which is aligned to the center of gravity of the quadrotor. The linear position and the orientation of the quadrotor are defined as:

$$\eta^I = [x \ y \ z]^T \quad , \quad \theta^I = [\phi \ \theta \ \psi]^T \quad (2.1)$$

Where x, y and z are the coordinates of the center of gravity in the inertial frame and ϕ, θ and ψ are the Euler angles representing the roll, pitch and yaw respectively, in the inertial frame.

2.3.1 Linear and angular velocity

2.3.1.1 Euler angles

Euler angles $\theta^I = [\phi \ \theta \ \psi]^T$ or attitude angles of the quadrotor are the orientation of the body frame with respect to the inertial frame. A rotation matrix is needed to map the orientation from the body frame to the inertial frame (Samir Bouabdallah, 2007b; Garcia et al., 2006; Uebe, 2008). The rotation matrix is obtained by multiplying three basic rotation matrices around the axes z, y and x , which are denoted as $R_z(\psi), R_y(\theta)$ and $R_x(\phi)$ respectively. At the beginning, body frame is coincident with the fixed inertial frame, after making a rotation around x -axis of a roll angle ($-\frac{\pi}{2} < \phi < \frac{\pi}{2}$) followed by a rotation around y -axis of a pitch angle ($-\frac{\pi}{2} < \theta < \frac{\pi}{2}$) followed by a rotation around z -axis angle of yaw ($-\pi < \psi < \pi$) we have the formula of the rotation matrix R :

$$\begin{aligned}
R &= R_z(\psi) \times R_y(\theta) \times R_x(\phi) \\
R &= \begin{bmatrix} c\psi & -s\psi & 0 \\ s\psi & c\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} c\theta & 0 & s\theta \\ 0 & 1 & 0 \\ -s\theta & 0 & c\theta \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\phi & -s\phi \\ 0 & s\phi & c\phi \end{bmatrix} \\
R &= \begin{bmatrix} c\psi c\theta & s\phi s\theta c\psi - s\psi c\phi & c\phi s\theta c\psi + s\psi s\phi \\ s\psi c\theta & s\phi s\theta s\psi + c\psi c\theta & c\phi s\theta s\psi - s\phi c\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{bmatrix} \\
&(c: \cos, \quad s: \sin)
\end{aligned} \tag{2.2}$$

The rotation matrix R is orthonormal ($R^{-1} = R^T$) and its determinant is equal to one ($\det(R) = 1$).

2.3.1.2 Angular Velocities

The angular velocity in the body frame is $\Omega = [p \ q \ r]^T$, where p, q and r are the angular velocities around x, y and z respectively in the body frame. The relation between the Euler angular rates in the inertial frame $\dot{\theta}^I = [\dot{\phi} \ \dot{\theta} \ \dot{\psi}]^T$ and the angular velocity in the body frame $\Omega = [p \ q \ r]^T$ is introduced by using the transfer matrix (Samir Bouabdallah, 2007b; Garcia et al., 2006; Uebe, 2008), we have:

$$\begin{aligned}
\Omega &= \begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + R_x(\phi)^{-1} \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + (R_y(\theta)R_x(\phi))^{-1} \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \\
\Omega &= \begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \dot{\theta} c\phi \\ -\dot{\theta} s\phi \end{bmatrix} + \begin{bmatrix} -\dot{\psi} s\theta \\ \dot{\psi} s\phi c\theta \\ \dot{\psi} c\phi c\theta \end{bmatrix} = \begin{bmatrix} \dot{\phi} - \dot{\psi} s\theta \\ \dot{\theta} c\phi + \dot{\psi} s\phi c\theta \\ \dot{\psi} s\phi c\theta - \dot{\theta} s\phi \end{bmatrix} \\
\Omega &= \begin{bmatrix} 1 & 0 & -s\theta \\ 0 & c\phi & s\phi c\theta \\ 0 & -s\phi & c\phi c\theta \end{bmatrix} \times \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}
\end{aligned} \tag{2.3}$$

When the quadrotor makes small rotations, the following approximations are possible, $c\phi = c\theta = c\psi = 1$ and $s\phi = s\theta = s\psi = 0$. Therefore, the angular velocity can be considered as:

$$\Omega \approx [\dot{\phi} \ \dot{\theta} \ \dot{\psi}]^T \quad (2.4)$$

2.3.1.3 Linear Velocities

The relation between the linear velocities in the inertial frame v^I is a function of the velocity in the body frame v^B as follows:

$$v^I = R v^B$$

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = R \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad (2.5)$$

Where u, v and w are the linear velocities in x, y and z respectively in the body frame.

2.3.2 Development of the mathematical model

The mathematical model of the quadrotor can be derived by considering the affecting forces and moments. The quadrotor model can be described by the translational equations of motion and the rotational equations of motion {Ghommam, 2017 #5} as in (2.6) and (2.12).

2.3.2.1 Translational motion equations

To develop the translational equations of the quadrotor, we start from the translational equations of motion:

$$\dot{\eta} = v^I$$

$$m\ddot{\eta} = F_f + F_t + F_g \quad (2.6)$$

Where, $\dot{\eta}, \ddot{\eta}$ are the velocity and acceleration of the position vector $\eta = [x \ y \ z]^T$ and m is the total mass of the quadrotor.

F_f : is the force generated by the four rotors, it is given as:

$$\begin{aligned}
 F_{f0} &= R \begin{bmatrix} 0 \\ 0 \\ \sum_{i=1}^4 F_i \end{bmatrix} = \begin{bmatrix} (c\phi s\theta c\psi + s\phi s\psi) \sum_{i=1}^4 F_i \\ (c\phi s\theta s\psi - s\phi c\psi) \sum_{i=1}^4 F_i \\ (c\phi c\theta) \sum_{i=1}^4 F_i \end{bmatrix} \\
 F_f &= E F_{f0} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} (c\phi s\theta c\psi + s\phi s\psi) \sum_{i=1}^4 F_i \\ (c\phi s\theta s\psi - s\phi c\psi) \sum_{i=1}^4 F_i \\ (c\phi c\theta) \sum_{i=1}^4 F_i \end{bmatrix} \\
 F_f &= \begin{bmatrix} (c\phi s\theta c\psi + s\phi s\psi) \sum_{i=1}^4 F_i \\ (c\phi s\theta s\psi - s\phi c\psi) \sum_{i=1}^4 F_i \\ -(c\phi c\theta) \sum_{i=1}^4 F_i \end{bmatrix}, \quad F_i = b \omega_i^2
 \end{aligned} \tag{2.7}$$

F_t : is the drag force along the axes (x, y, z) , it is given as:

$$F_t = \begin{bmatrix} -k_{ftx} & 0 & 0 \\ 0 & -k_{f ty} & 0 \\ 0 & 0 & -k_{ftz} \end{bmatrix} \dot{\eta} \tag{2.8}$$

Where k_{ftx} , $k_{f ty}$, k_{ftz} are the translational drag coefficients.

F_g : is the gravity force given as:

$$F_g = \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} \tag{2.9}$$

Where g is gravity acceleration. After substituting the forces in (2.6), we have:

$$m \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} (c\phi s\theta c\psi + s\phi s\psi) \sum_{i=1}^4 F_i \\ (c\phi s\theta s\psi - s\phi c\psi) \sum_{i=1}^4 F_i \\ -(c\phi c\theta) \sum_{i=1}^4 F_i \end{bmatrix} + \begin{bmatrix} -k_{ftx} \dot{x} \\ -k_{f ty} \dot{y} \\ -k_{ftz} \dot{z} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} \tag{2.10}$$

Then, we find the dynamic equations which represent the dynamic movement of the quadrotor:

$$\begin{aligned}
\ddot{x} &= \frac{1}{m}(c\phi s\theta c\psi + s\phi s\psi)(\sum_{i=1}^4 F_i) - \frac{k_{ftx}}{m}\dot{x} \\
\ddot{y} &= \frac{1}{m}(c\phi s\theta s\psi - s\phi c\psi)(\sum_{i=1}^4 F_i) - \frac{k_{fity}}{m}\dot{y} \\
\ddot{z} &= \frac{-1}{m}(c\phi c\theta)(\sum_{i=1}^4 F_i) - \frac{k_{ftz}}{m}\dot{z} + g
\end{aligned} \tag{2.11}$$

2.3.2.2 Rotational motion equations

To develop the rotational equations of the quadrotor, we start from the rotational equations of motion:

$$\begin{aligned}
\dot{R} &= RS(\Omega) \\
J\dot{\Omega} &= -\Omega \times J\Omega - M_{gh} - M_a + M_f
\end{aligned} \tag{2.12}$$

Where (\times) is the cross product. J is the inertia matrix and it is given as:

$$J = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \tag{2.13}$$

$S(\Omega)$ is the skew matrix for the velocity $\Omega = [p \ q \ r]^T$, and it is given as:

$$S(\Omega) = \begin{bmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{bmatrix} \tag{2.14}$$

M_{gh} : is the gyroscopic moment due to the rotors inertia J_r and relative velocity Ω_r (Nagaty, Saeedi, Thibault, Seto, & Li, 2013). The gyroscopic moment s is given by the following relation:

$$M_{gh} = \begin{bmatrix} J_r \Omega_r \dot{\theta} \\ -J_r \Omega_r \dot{\phi} \\ 0 \end{bmatrix} \quad (2.15)$$

M_f : is the moment caused by thrust and drag forces:

$$M_f = \begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix} \quad (2.16)$$

Where M_x and M_y are two moments occur due to the rotation around x and y axes which are caused by the difference between the lift forces of rotors (3,4) and rotors (1,2) in x -direction and the difference between lift forces of rotors (2,3) and rotors (1,4) in y - direction, these moments are given by the following relation:

$$\begin{aligned} M_x &= l(F_1 - F_2 - F_3 + F_4) = lb (\omega_1^2 - \omega_2^2 - \omega_3^2 + \omega_4^2) \\ M_y &= l(F_1 + F_2 - F_3 - F_4) = lb (\omega_1^2 + \omega_2^2 - \omega_3^2 - \omega_4^2) \end{aligned} \quad (2.17)$$

M_z is a moment occur due to the difference of velocity in the two couples of propellers around the z axis, this moment is given by the following relation:

$$M_z = d (-\omega_1^2 + \omega_2^2 - \omega_3^2 + \omega_4^2) \quad (2.18)$$

M_a : is the moment resulting from aerodynamic friction, it is given by:

$$M_a = \begin{bmatrix} k_{fax} \dot{\phi}^2 \\ k_{fay} \dot{\theta}^2 \\ k_{faz} \dot{\psi}^2 \end{bmatrix} \quad (2.19)$$

Where $k_{fax}, k_{fay}, k_{faz}$ are the coefficients of aerodynamic friction. Substituting the corresponding formulas in (2.12), we find:

$$\begin{aligned}
\begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} &= - \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \times \left(\begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \right) - \begin{bmatrix} J_r \Omega_r \dot{\theta} \\ -J_r \Omega_r \dot{\phi} \\ 0 \end{bmatrix} - \begin{bmatrix} k_{fax} \dot{\phi}^2 \\ k_{fay} \dot{\theta}^2 \\ k_{faz} \dot{\psi}^2 \end{bmatrix} \\
&+ \begin{bmatrix} l b (\omega_1^2 - \omega_2^2 - \omega_3^2 + \omega_4^2) \\ l b (+\omega_1^2 + \omega_2^2 - \omega_3^2 - \omega_4^2) \\ d (-\omega_1^2 + \omega_2^2 - \omega_3^2 + \omega_4^2) \end{bmatrix}
\end{aligned} \tag{2.20}$$

We then obtain the differential equations defining the rotational movement:

$$\begin{aligned}
I_x \ddot{\phi} &= -\dot{\theta} \dot{\psi} (I_z - I_y) - J_r \Omega_r \dot{\theta} - k_{fax} \dot{\phi}^2 + l b (\omega_1^2 + \omega_2^2 - \omega_3^2 - \omega_4^2) \\
I_y \ddot{\theta} &= -\dot{\phi} \dot{\psi} (I_z - I_x) + J_r \Omega_r \dot{\phi} - k_{fay} \dot{\theta}^2 + l b (\omega_1^2 + \omega_2^2 - \omega_3^2 - \omega_4^2) \\
I_z \ddot{\psi} &= -\dot{\phi} \dot{\theta} (I_y - I_x) - k_{faz} \dot{\psi}^2 + d (-\omega_1^2 + \omega_2^2 - \omega_3^2 + \omega_4^2)
\end{aligned} \tag{2.21}$$

As a result, the complete dynamic model governing the quadrotor is as follows:

$$\begin{aligned}
\ddot{\phi} &= \frac{(I_y - I_z)}{I_x} \dot{\theta} \dot{\psi} - \frac{J_r}{I_x} \Omega_r \dot{\theta} - \frac{k_{fax}}{I_x} \dot{\phi}^2 + \frac{1}{I_x} u_2 \\
\ddot{\theta} &= \frac{(I_z - I_x)}{I_y} \dot{\phi} \dot{\psi} + \frac{J_r}{I_y} \Omega_r \dot{\phi} - \frac{k_{fay}}{I_y} \dot{\theta}^2 + \frac{1}{I_y} u_3 \\
\ddot{\psi} &= \frac{(I_x - I_y)}{I_z} \dot{\phi} \dot{\theta} - \frac{J_r}{I_z} \Omega_r \dot{\psi} - \frac{k_{faz}}{I_z} \dot{\psi}^2 + \frac{1}{I_z} u_4 \\
\ddot{x} &= -\frac{k_{ftx}}{m} \dot{x} + \frac{1}{m} u_x u_1 \\
\ddot{y} &= -\frac{k_{fty}}{m} \dot{y} + \frac{1}{m} u_y u_1 \\
\ddot{z} &= -\frac{k_{ftz}}{m} \dot{z} + g - \frac{1}{m} (c\phi c\theta) u_1
\end{aligned} \tag{2.22}$$

The control inputs for the altitude and attitude are u_1, u_2, u_3 and u_4 while u_x, u_y are auxiliary control input designed to generate the reference signals of the roll and pitch angles (desired roll ϕ_d and desired pitch θ_d) then, the roll and pitch are controlled in u_2, u_3 . The auxiliary control signals and the desired roll and pitch are given in the following formulas (Gupta & Kothari, 2017; Khebbache, 2018):

$$\begin{aligned}
 u_x &= c\phi s\theta c\psi + s\phi s\psi \\
 u_y &= c\phi s\theta s\psi - s\phi c\psi \\
 \phi_d &= \arcsin(u_x s\psi_d - u_y c\psi_d) \\
 \theta_d &= \arcsin\left(\frac{u_x c\psi_d + u_y s\psi_d}{c\phi_d}\right)
 \end{aligned} \tag{2.23}$$

2.4 General Structure of Quadrotor Control

In this section, we explain the control strategy in the quadrotor. As known, the quadrotor is an under-actuated system. Four rotors are used to control six degrees of freedom. Control strategy is based on two loops of control. The first loop is “Internal control loop” which controls roll ϕ , pitch θ , yaw ψ and altitude z as shown in block diagram in Figure 2.6. The internal control loop uses the reference value to generate the proper control signal. The second loop is “External Control loop” which controls the position x and y . The purpose of the external control is to calculate the desired roll ϕ_d and the desired pitch θ_d based on the desired position by using the control of position u_x , u_y and the desired yaw ψ_d . The desired position (x_d, y_d, z_d) and the desired Yaw ψ_d comes directly from the user.

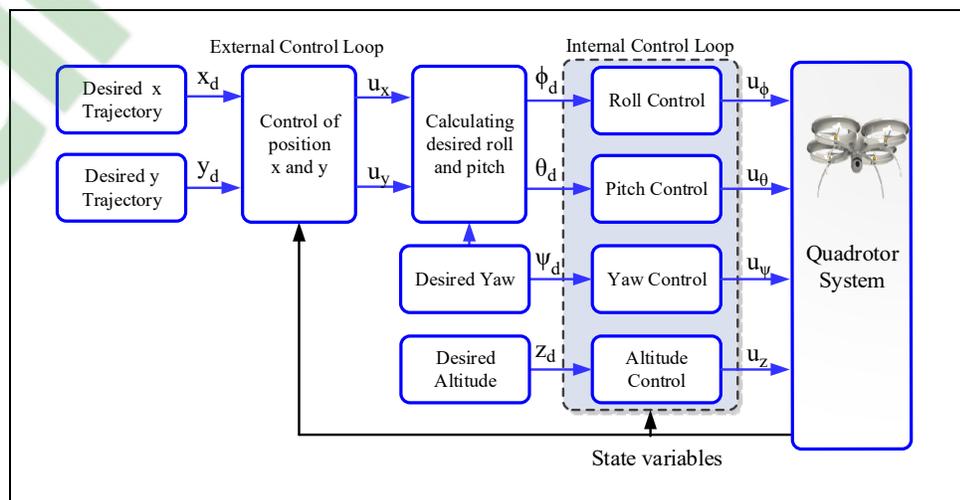


Figure 2.6 Control block diagram

2.5 Conclusion

This chapter demonstrates the modelling of the quadrotor flying robots. The quadrotor is subject to different forces and moments. This system is a six degrees of freedom robot and it is controlled by the velocity of four rotors. By varying the rotational speeds of these rotors, the quadrotor can make different translational and rotational movements. The complexity, nonlinearity and the interaction between system states can be seen clearly.

The obtained dynamic model allows analysis, simulation and control design of high-level controllers in four different articles. Two articles are nonlinear control systems, “Vision based Leader Follower Approach for Uncertain Quadrotor Dynamics Using Feedback Linearization Sliding Mode Control (FLSM)” and “Position and Attitude tracking of Uncertain Quadrotor UAV based on Non-Singular Terminal Super-Twisting Algorithm”. The other two articles are perturbation compensation systems, “Hierarchical Perturbation Compensation System with Exponential Reaching Law Sliding Mode Controller” and “Three Loop Uncertainty Compensation System with Exponential Reaching Law Sliding Mode Controller”.

CHAPTER 3

HIERARCHICAL PERTURBATION COMPENSATION SYSTEM WITH EXPONENTIAL REACHING LAW SLIDING MODE CONTROLLER IN A QUADROTOR

Walid Alqaisi¹, Brahim Brahmi¹, Jawhar Ghommam², Maarouf Saad¹ and Vahé Nerguizian¹

¹ Department of Electrical Engineering, École de technologie supérieure,
1100 Notre-Dame West, Montreal, Quebec, Canada H3C 1K3

² Department of Electrical Engineering, Sultan Quaboos University,
Al Khoudh, Muscat 123, Oman

Paper submitted for publication, July 2019

Abstract:

This article addresses the problem of perturbation in UAV quadrotors. Three subsystems are designed to provide continuous and precise estimation of perturbation and residual perturbation. The three subsystems form a Hierarchical Perturbation Compensator HPC, which is built to compensate for system uncertainties, non-modelled dynamics and external disturbances. The nonlinear control Exponential Reaching Law Sliding Mode ERLSM is utilized with the HPC. Lyapunov stability analysis proves the stability of the entire compensator-controller system. This system has superior proficiency to decrease unknown perturbation either external or internal. It also has the ability to achieve full control of the six-degree-of-freedom quadrotor. This work is an improvement and extension to our previous work (Alqaisi, Brahmi, Ghommam, Saad, & Nerguizian, 2018b). The system performance for position, altitude and attitude control is demonstrated by analysis, simulation and experiments.

Keywords: Exponential Reaching Law; Sliding Mode Controller system; Feedback and Feed-forward Compensators; Hierarchical Perturbation Compensation; UAV-quadrotor.

3.1 Introduction

The large number of applications in UAV quadrotors opened wide area of research projects in this field. One of the biggest problems in this kind of UAVs is the perturbation. Because it is small-in-size (relatively) and due to the lack of damping and the cross-coupling between degrees of freedom, the quadrotor is considered very sensitive to perturbation. Perturbation reduces system performance and affects the control negatively. Perturbation includes wind disturbance, air friction, uncertain/changing parameters and non-modelled dynamics. In certain applications, task of loading or unloading of materials makes great change in inertia and mass parameters which adds extra perturbation. Perturbation in robotics is an important area of research. In order to eliminate or to reduce it, researchers used three main approaches:

- 1) Building robust controllers that is able to handle the perturbation.
- 2) Building adaptive controllers, which has the ability to adapt to perturbation.
- 3) Designing observers to reduce uncertainties and disturbance.

Under the first approach, an “active disturbance-rejection” controller is designed to eliminate the impact of the state coupling and uncertainties for an autonomous quadrotor (Chang et al., 2016; Sanz et al., 2016), a cascade control law is designed as robust control in (H. Liu, Zhao, Zuo, & Zhong, 2016), backstepping controller is developed in (Cabecinhas, Cunha, & Silvestre, 2014), Fuzzy logic-based tracking controller is used in (Kayacan & Maslim, 2016). However, robust controllers lack adaptation property which reduces the performance. Advanced nonlinear control methods enable to scientifically design stabilizing controllers, which meet robust stability and performance on the plant uncertainty. In some systems, uncertainties are represented by unstructured uncertainty such as additive uncertainties which lack in-phase information and whose upper bound of magnitude is assumed to cover the worst case of plant uncertainty. Thus, it inevitably includes a class of plants, which may practically never happen. Therefore, these controllers are designed with high-gain, which makes a control system very conservative in performance although they guarantee robust stability for the assumed plant uncertainty.

To solve this issue, other researchers developed adaptation functionality in their control. For example, adaptive control method is implemented to adapt to disturbance and actuator failures in (F. Chen et al., 2014), an adaptive output feedback compensator is used in (Marino & Tomei, 2016b), an adaptive time-varying compensation is constructed for a quadrotor under uncertainties in (Ton et al., 2016) and prediction-based control is developed in (Alexis et al., 2012).

Despite the good performance of the mentioned systems, they lack estimation and compensation of the perturbation during robot operations. Hence, many observers were designed to solve this problem, such as (X. Wang, Shirinzadeh, & Ang, 2014; Yin & Xiao, 2016), Luenberger observer is used with feedback linearization (Mokhtari et al., 2006), disturbance observers as a part of the control is used in (F. Chen, Lei, Zhang, Tao, & Jiang, 2016b), a sliding mode-based disturbance observer is designed in (Lenaick Besnard, Yuri B Shtessel, & Brian Landrum, 2007; Besnard et al., 2012; Zhang et al., 2013), an acceleration-based observer is built for attitude control (Jeong, Jung, & Tomizuka, 2012a) and an extended observer with feedback sliding mode is used in (Ruifeng Zhang, Quan Quan, & K-Y Cai, 2011). The existing perturbation observers systems guarantee good performance for the quadrotor, even though; they suffer from some drawbacks such as the influence of measurement lag and sensor noise. On the other hand, they lack comparing the system with the desired states' behavior. Moreover, they lack tracking and rejecting of residual perturbation.

In an effort to overcome the mentioned drawbacks, this study focuses on integrating a system of Hierarchical Perturbation Compensators HPC (S. Kwon & W. K. Chung, 2004), The HPC comprises three subsystems to provide estimation and compensation hierarchically. The first subsystem is built to provide estimation of perturbation based on the desired dynamics, which is lag-free, and noise-free signals, meanwhile the second subsystem is built to provide estimation based on the real dynamics. In order to track perturbations in different levels, the third subsystem compensates for the dynamic error of the sliding mode. As we see, each subsystem covers for the drawback of other subsystems. The proposed HPC estimates and

compensates perturbation simultaneously based on time delay estimation. The HPC has adaptive control property (S. Kwon & W. K. Chung, 2004), as it generates control effort that is required to compensate the current perturbation. In addition, it has an integral control feature as the current compensation value is estimated based on one-step delayed input.

To control the quadrotor, sliding mode control SMC is one of the most popular nonlinear control methods. To achieve fast response in the SMC, high gains need to be used, but on the other hand, large gains lead to high chattering. In order to solve this dilemma, Exponential Reaching Low Sliding Mode ERLSM (C. J. Fallaha et al., 2011a) is utilized, which ensures fast response and lowest possible chattering at the same time. Its advantage over other techniques of chattering reducers is that the exponential term can adapt to the variations of the sliding surface (S) smoothly.

Both the HPC and the ERLSM contribute in stabilizing the whole system. This can be explained as follows; lowering the gain by the ERLSM provides support to the HPC in order not to reach the upper bound of magnitude when encountering higher perturbation. This means that the HPC can reach higher magnitude to compensate for higher perturbation. On the other hand, the ERLSM, as a robust control system, needs support if there is a large perturbation as it can be seen in Section 3-7.

The aforementioned advantages of the HPC over other systems (robust controls, adaptive controls and observers) encouraged us to utilize it in a quadrotor. This work extends the results in our previous work (Alqaisi et al., 2018b) to include position control, ERLSM control, comparison with SMC and practical experiment. To the best of our knowledge, this system is implemented for the first time on full control six degrees of freedom robot/quadrotor, including stability analysis, simulation and experiment. By using the HPC estimation and compensation in addition to the ERLSM control, the contributions of this paper can be described as:

- 1) Hierarchical Perturbation Compensator (HPC) system is built and applied to a six degrees of freedom under-actuated robot, quadrotor.

- 2) The combined system of the HPC and the ERLSM provide higher upper bound of perturbation compensation magnitude, this gives more ability to attenuate higher perturbation.

This paper is organized as follows; quadrotor dynamics is described in section 3-2. Problem statement is in section 3-3. Perturbation Compensators and the HPC are demonstrated in sections 3-4 and 3-5. The entire system is designed and the stability is analyzed in section 3-6. Simulation and analysis are given in section 3-7 and experimental results are in section 3-8, finally the conclusion is in section 3-9.

3.2 Quadrotor Dynamics

The quadrotor chassis is built by four motors in cross structure as in Figure 3.1. It is designed in a way that each opposite rotor rotates in the same direction. By controlling each motor angular velocity, they produce forces and moments as desired. Two coordinate frames are used, the earth inertial frame I and the Body-fixed frame B. Quadrotor dynamic model based on Lagrange or Newton-Euler is recognized by many researchers (Samir Bouabdallah, 2007a; S. Bouabdallah & Siegwart, 2007; Bresciani, 2008; Erginer & Altuğ, 2007; Hicham, 2012) and is described as:

$$\begin{aligned}
 \ddot{\phi} &= \frac{i_y - i_z}{i_x} \dot{\theta} \dot{\psi} - \frac{j_r}{i_x} \dot{\theta} \omega_r + \frac{1}{i_x} u_2 \\
 \ddot{\theta} &= \frac{i_z - i_x}{i_y} \dot{\phi} \dot{\psi} + \frac{j_r}{i_y} \dot{\phi} \omega_r + \frac{1}{i_y} u_3 \\
 \ddot{\psi} &= \frac{i_x - i_y}{i_z} \dot{\theta} \dot{\phi} + \frac{1}{i_z} u_4 \\
 \ddot{x} &= \frac{1}{m} u_x u_1 \\
 \ddot{y} &= \frac{1}{m} u_y u_1 \\
 \ddot{z} &= g - \frac{(\cos\phi \cos\theta)}{m} u_1
 \end{aligned} \tag{3.1}$$

Where $X(t) = [\phi(t), \theta(t), \psi(t), x(t), y(t), z(t)]^T$ is the state vector, and $\dot{X}(t), \ddot{X}(t)$ are velocity and acceleration vectors, respectively. Where x, y, z is the quadrotor position. ϕ, θ, ψ are the three Euler angles roll, pitch and yaw angles respectively. The moments of inertia in the body frame are i_x, i_y, i_z . The total mass is m . Gravity acceleration is g and j_r is the rotor inertia. The control inputs for attitude and altitude u_1, u_2, u_3, u_4 can be described as:

$$\begin{aligned} u_1 &= b(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \\ u_2 &= b l_e(\omega_1^2 + \omega_4^2 - \omega_3^2 - \omega_2^2) \\ u_3 &= b l_e(\omega_1^2 + \omega_2^2 - \omega_3^2 - \omega_4^2) \\ u_4 &= d_r(-\omega_1^2 + \omega_2^2 - \omega_3^2 + \omega_4^2) \end{aligned} \quad (3.2)$$

Where b is the thrust coefficient and d_r is the drag coefficient. l_e is the half-length of the helicopter. ω_i ($i = 1, 2, 3, 4$) is the angular velocity of the quadrotor motors and $\omega_r = -\omega_1 + \omega_2 - \omega_3 + \omega_4$. The auxiliary inputs are:

$$\begin{aligned} u_x &= (\sin \theta \cos \psi \cos \phi + \sin \psi \sin \phi) \\ u_y &= (\sin \theta \sin \psi \cos \phi - \cos \psi \sin \phi) \end{aligned} \quad (3.3)$$

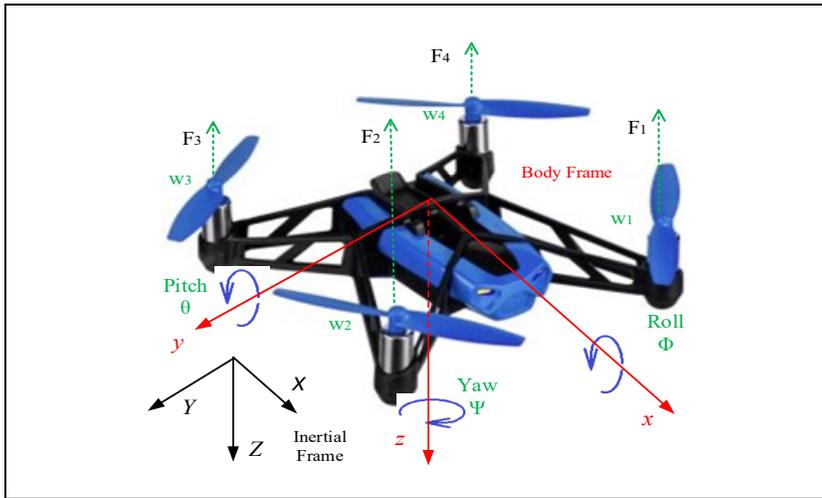


Figure 3.1 Quadrotor structure, forces, angles and frames
("Parrot Minidrone," 2018)

The nonlinear quadrotor system can be described by the companion form, or controllability canonical form (Slotine & Li, 1991):

$$\ddot{X} = F_T(\dot{X}) + G_T(X)U \quad (3.4)$$

Where $F_T(\dot{X})$ and $G_T(X)$ are the total nonlinear dynamics of the quadrotor system that includes the known and the unknown dynamics. By detailing nominal and uncertain dynamic parts, equation (3.4) can be written as:

$$\ddot{X} = (F(\dot{X}) + \Delta F(\dot{X})) + (G(X) + \Delta G(X))U \quad (3.5)$$

Where, $\Delta F(\dot{X})$ and $\Delta G(X)$ are uncertain dynamics. U is the input vector given as below. $F(\dot{X})$ and $G(X)$ are the nominal nonlinear parts and can be given as:

$$F(\dot{X}) = \begin{bmatrix} \frac{i_y - i_z}{i_x} \dot{\theta} \dot{\psi} + \frac{J_r}{i_x} \dot{\theta} \omega_r \\ \frac{i_z - i_x}{i_y} \dot{\phi} \dot{\psi} - \frac{J_r}{i_y} \dot{\phi} \omega_r \\ \frac{i_x - i_y}{i_z} \dot{\theta} \dot{\phi} \\ 0 \\ 0 \\ g \end{bmatrix}, \quad U = \begin{bmatrix} u_2 \\ u_3 \\ u_4 \\ u_1 u_x \\ u_1 u_y \\ u_1 \end{bmatrix}$$

$$G(X) = \begin{bmatrix} 1/i_x & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/i_y & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/i_z & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/m & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/m & 0 \\ 0 & 0 & 0 & 0 & 0 & -(\cos\phi\cos\theta)/m \end{bmatrix}$$

The actual input to the quadrotors are u_1, u_2, u_3 and u_4 as in (3.2). While u_x, u_y are auxiliary control input used to calculate the desired roll ϕ_d and desired pitch θ_d , then, the roll and pitch

will be controlled in u_2 , u_3 . The desired roll and pitch are found as (Gupta & Kothari, 2017; Khebbache, 2018):

$$\begin{aligned}\phi_d &= \sin^{-1}(u_x \sin \psi_d - u_y \cos \psi_d) \\ \theta_d &= \sin^{-1}\left(\frac{u_x \cos \psi_d + u_y \sin \psi_d}{\cos \phi_d}\right)\end{aligned}\tag{3.6}$$

Where ψ_d is the desired yaw angle. The following assumptions are needed for stability analysis:

Assumption 1: Matrix $\mathbf{G}(X)$ is invertible

Assumption 2: The perturbation $\Gamma(t)$ is a globally Lipschitz function.

Assumption 3: The trajectory x_d, y_d and z_d are smooth and their first and second derivatives are bounded.

Remark 1: For Assumption 1, matrix $\mathbf{G}(x)$ is invertible means that the quadrotor is not allowed to perform aggressive maneuvering and therefore the roll ϕ and the pitch θ angles are not equal to $\pi/2$. As for Assumption 2, $\Gamma(t)$ is globally Lipschitz function means $\Gamma(t)$ is continuous and differentiable and don't vary greatly during a small period of time τ .

3.3 Problem Statement

Robot quadrotors are subject to different types of perturbation that adversely affect their performance. Perturbation includes uncertain parameters and modeling, alterations of parameters and weight, wind resistance and other kinds of disturbance. In the proposed system, the main objective can be stated as follows: Given a desired trajectory X_d , under the above assumptions, devise a control input U for the quadrotor dynamics (17), such that the tracking errors of the closed-loop system are bounded and the following limits hold for $t > 0$:

$$\begin{aligned} \lim_{t \rightarrow \infty} |X - X_d| &= 0 \\ \lim_{t \rightarrow \infty} |\dot{X} - \dot{X}_d| &= 0 \end{aligned} \quad (3.7)$$

3.4 Perturbation Compensators

The perturbation can be described as in the following formula (S. Kwon & W. K. Chung, 2004):

$$\Gamma(t) = \Delta F(\dot{X}) + \Delta \mathbf{G}(X) U + F_c(t) + D(t) \quad (3.8)$$

Where, $F_c(t)$ is the system non-modelled dynamics. The external disturbance is $D(t)$. By incorporating perturbation term (3.8) in the general equation (3.5), leads to the following formula:

$$\ddot{X} = F(\dot{X}) + \mathbf{G}(X)U + \Gamma(t) \quad (3.9)$$

A perturbation compensator is required to provide estimation and compensation of the perturbation. Based on (3.9) the current perturbation can be stated as:

$$\Gamma(t) = \ddot{X} - F(\dot{X}) - \mathbf{G}(X)U \quad (3.10)$$

Since the disturbance is unknown, a time delay approach is used to estimate its value. In the designed perturbation compensator, a compensation signal is required that is equal to one step time-period delay by making use of the system variables i.e. the goal is to make the compensator provide estimated perturbation as $\hat{\Gamma}(t)$:

$$\hat{\Gamma}(t) = \Gamma(t - \tau) = \ddot{X}(t - \tau) - F(\dot{X}(t - \tau)) - \mathbf{G}(X(t - \tau))U(t - \tau) \quad (3.11)$$

Where τ is the process step time. The first and second subsystems of Hierarchical Perturbation Compensator HPC is designed based on formula (3.11).

3.5 Hierarchical Perturbation Compensator

The HPC system (S. Kwon & W. K. Chung, 2004) is applied to a quadrotor with Exponential Reaching Law Sliding Mode ERLSM control. The HPC comprises three subsystems to provide estimation and compensation hierarchically, (Figure 3.2). The first subsystem is a Feed-Forward Perturbation Compensator FFPC, which is built with respect to the desired dynamics; the second is a Feed-Back Perturbation Compensator FBPC, which is built with respect to nominal dynamics. The third is a Sliding Mode dynamic error Perturbation Compensator SMPC that is built with respect to the sliding mode dynamic error.

The compensation involvement of the FFPC makes the quadrotor behave similarly to the desired dynamics. The feed-forward signal has the advantages of being lag-free and noise free and so the FFPC compensates the shortcoming of the FBPC. The FBPC depends on the measurements which suffer from dynamic lag and inaccuracy, but also it is functioning based on the real dynamics. The FFPC and the FBPC both work in the inner loop while the SMPC works from the outside loop.

The FFPC provides compensation value $\hat{\Gamma}_{FF}$. The compensation error or the residual perturbation is to be rejected by FBPC and have the value $\hat{\Gamma}_{FB}$. Both of $\hat{\Gamma}_{FF}$ and $\hat{\Gamma}_{FB}$ reduce the closed loop error to a very small value. The SMPC as a third level works from the outer loop to compensate for the remaining error. This can be explained as:

$$\begin{aligned}\tilde{\Gamma}_{FF}(t) &= \Gamma(t) - \hat{\Gamma}_{FF}(t) \triangleq \Gamma_{FB}(t) \\ \tilde{\Gamma}_{FB}(t) &= \Gamma(t) - \hat{\Gamma}_{FF}(t) - \hat{\Gamma}_{FB}(t) = \Gamma_{FB}(t) - \hat{\Gamma}_{FB}(t) \triangleq \Gamma_{SM}(t)\end{aligned}\tag{3.12}$$

The design of the three subsystems can be demonstrated as follows (S. Kwon & W. K. Chung, 2004):

$$\hat{\Gamma}_{FF}(t) = \ddot{X}_d(t - \tau) - F(\dot{X}_d(t - \tau)) - \mathbf{G}(X_d(t - \tau))U(t - \tau) \quad (3.13)$$

$$\hat{\Gamma}_{FB}(t) = \ddot{X}(t - \tau) - F(\dot{X}(t - \tau)) - \mathbf{G}(X(t - \tau))U(t - \tau) - \hat{\Gamma}_{FF}(t - \tau) \quad (3.14)$$

$$\hat{\Gamma}_{SM}(t) = \hat{\Gamma}_{SM}(t - \tau) + \dot{S}(t - \tau) + \mathbf{K} \text{Sign}(t - \tau) \quad (3.15)$$

Where, \dot{S} is the derivative of the sliding surface. \mathbf{K} is a positive-definite, dynamic value and diagonal matrix. \mathbf{K} and \dot{S} are defined in the following section. The implemented HPC incorporates the three aforementioned compensator subsystems:

$$\hat{\Gamma}(t) = \hat{\Gamma}_{HPC}(t) = \hat{\Gamma}_{FF}(t) + \hat{\Gamma}_{FB}(t) + \hat{\Gamma}_{SM}(t) \quad (3.16)$$

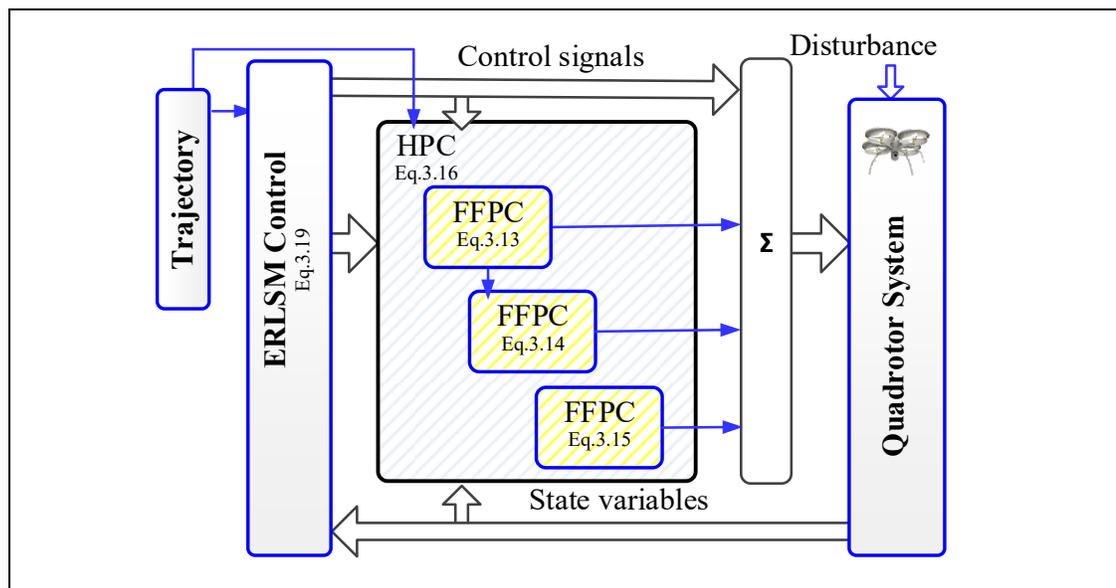


Figure 3.2 HPC -ERLSM system

3.6 Integrated System Design

Sliding mode controller builds a systematic methodology for retaining stability by using a sliding surface to attract the error and its derivative. The method used in this paper is based on (Slotine & Li, 1991). In sliding mode approach, mechanical systems are designed to drag and force the system state to remain within a region of a predetermined switching function. The advantage of this type of control is that the plant dynamic behavior can be adjusted by a certain choice of a desired switching function (J. Liu & Wang, 2012a).

The tracking error is defined as, $E = X - X_d$, where X_d is the desired trajectory, $X_d = [\phi_d, \theta_d, \psi_d, x_d, y_d, z_d]^T$. The sliding surface and its derivative are defined as:

$$\begin{aligned} S &= \dot{E} + \Lambda E \\ \dot{S} &= \ddot{E} + \Lambda \dot{E} = \ddot{X} - (\ddot{X}_d - \Lambda \dot{E}) \\ \dot{S} &= \ddot{X} - \ddot{X}_r \end{aligned} \quad (3.17)$$

We select $\ddot{X}_r = \ddot{X}_d - \Lambda \dot{E}$, where $\Lambda = \text{diag}(\lambda_i)$, ($i = 1, 2 \dots 6$) is a definite positive diagonal matrix.

The following reaching law (3.18) attracts the error to the sliding surface. In order to have rapid reaching time, a high value should be given to the constant \mathbf{K} . However, this will increase the undesired chattering. In order to solve this dilemma, a dynamic value can be given to the constant \mathbf{K} (C. J. Fallaha, Saad, Kanaan, & Al-Haddad, 2011b). In this method, the constant \mathbf{K} takes high values when the error is high and takes low value when the error is small. The variation of \mathbf{K} ensures quick convergence and avoids high chattering. The proposed exponential reaching law is (C. J. Fallaha et al., 2011a):

$$\dot{S} = -\mathbf{K} \text{Sign}(s_i) \quad (3.18)$$

Where $\mathbf{K} = \text{diag}\left(\frac{k_1}{N(s_1)}, \frac{k_2}{N(s_2)}, \dots, \frac{k_n}{N(s_n)}\right)$ (C. J. Fallaha et al., 2011a), and $k_i > 0$ for $i = 1, 2, \dots, n$, $N(s_i) = \delta_{0i} + (1 - \delta_{0i})e^{-\alpha_i |s_i|^{p_i}}$. δ_{0i} is a strictly positive offset $0 < \delta_{0i} < 1$, p_i and α_i are strictly positive. It can be noticed that the exponential reaching law (3.18) does not affect the system stability because $N(s_i)$ is always positive.

Remark 2: If $|s_i|$ increases, $N(s_i)$ approaches to δ_{0i} therefore $k_i/N(s_i)$ converges to k_i/δ_{0i} which is greater than or equal to k_i . This means that $k_i/N(s_i)$ increases in the reaching phase, accordingly the movement to the sliding surface will be faster (C. J. Fallaha et al., 2011b).

The control system based on the classical SMC (Behal, Dixon, Dawson, & Xian, 2009) and after incorporating the HPC and the ERLSM is given as:

$$U = \mathbf{G}^{-1}(X) [\ddot{X}_r - F(\dot{X}) - \mathbf{K} \text{Sign}(S) - \hat{F}] \quad (3.19)$$

The *Sign* function is defined as:

$$\begin{aligned} \text{Sign}(S) &= [\text{sign}(s_1), \dots, \text{sign}(s_4)]^T \\ \text{sign}(s_i) &= \begin{cases} 1 & \text{for } s_i > 0 \\ 0 & \text{for } s_i = 0 \\ -1 & \text{for } s_i < 0 \end{cases}, \quad i = 1, 2, 3, 4 \end{aligned} \quad (3.20)$$

Remark 3: The designed controller is free from uncertainties as \hat{F} in contrast to adaptive technique is estimated using (3.16), which shows that its actual value can be forecasted based on the knowledge of the free-uncertainty model.

Proposition 1: Consider the quadrotor dynamic system (3.1), under Assumption 1, the design of the HPC (3.16) along with the controller (3.19), ensures that the solutions of the closed-loop systems are bounded, furthermore, the tracking error E converges asymptotically to zero as time goes to infinity.

Proof: To prove the stability, the following Lyapunov function is selected:

$$V = \frac{1}{2} S^T S \quad (3.21)$$

$$\dot{V} = S^T \dot{S} \quad (3.22)$$

Substituting \dot{S} from (3.17), then:

$$\dot{V} = S^T [\ddot{X} - \ddot{X}_r] \quad (3.23)$$

Substituting \ddot{X} from (3.9):

$$\dot{V} = S^T (F(\dot{X}) + \mathbf{G}(X)U + \Gamma - \ddot{X}_r) \quad (3.24)$$

Substituting the control (3.19) in (3.24):

$$\begin{aligned} \dot{V} &= S^T (F(\dot{X}) + \mathbf{G}(X)\mathbf{G}^{-1}(X)[\ddot{X}_r - F(\dot{X}) - \mathbf{K} \text{Sign}(S) - \hat{\Gamma}] + \Gamma(t) - \ddot{X}_r) \\ \dot{V} &= S^T (-\mathbf{K} \text{Sign}(S) + \tilde{\Gamma}) \end{aligned} \quad (3.25)$$

Where, $\tilde{\Gamma}(t) = \Gamma(t) - \hat{\Gamma}(t)$ is the estimation error. Equation (3.25) is rewritten as:

$$\begin{aligned} \dot{V}(s) &= \sum_{i=1}^n [s_i (-\frac{k_i}{N(s_i)} \text{sign}(s_i) + \tilde{\Gamma}_i(t))] \\ \dot{V}(s) &= \sum_{i=1}^n [-\frac{k_i}{N(s_i)} |s_i| + s_i \tilde{\Gamma}_i(t)] \end{aligned} \quad (3.26)$$

Where, $\tilde{\Gamma}_i(t) = \Gamma_i(t) - \hat{\Gamma}_i(t) = [\tilde{\Gamma}_1, \dots, \tilde{\Gamma}_6]^T$ and based on Assumption 2, it can be found:

$$\begin{aligned} |\tilde{\Gamma}_i(t)| &= |\Gamma_i(t) - \hat{\Gamma}_i(t)| \\ &= |\Gamma_i(t) - \Gamma_i(t - \tau)| \\ &\leq \delta_i |t - (t - \tau)| \\ &\leq \delta_i \tau \end{aligned} \quad (3.27)$$

Where $\delta_i > 0$ is the constant of Lipschitz function. In order to have a stable system, the following condition is required:

$$\begin{aligned}
-\frac{k_i}{N(s_i)} |s_i| + s_i \tilde{I}_i(t) &< 0 \\
-\frac{k_i}{N(s_i)} |s_i| + |s_i| |\tilde{I}_i(t)| &< 0 \\
-\frac{k_i}{N(s_i)} + \delta_i \tau &< 0 \\
\delta_i \tau \cdot N(s_i) &< k_i
\end{aligned} \tag{3.28}$$

From (3.26) and (3.28), it can be implied that:

$$\begin{aligned}
\dot{V} &\leq -\frac{k_i}{N(s_i)} |s_i| \\
&\leq -\frac{\sqrt{2}k_i}{N(s_i)} \sqrt{V}
\end{aligned} \tag{3.29}$$

Clearly, the solutions of the inequality (3.27) are bounded, therefore by the construction of the sliding surface S , the tracking errors E and \dot{E} are also bounded. From (3.27), it is clear that $\dot{V} \leq 0$ which implies that the sliding surface converges to zero asymptotically. Convergence of S to zero immediately implies the convergence of the tracking errors $E = X - X_d$ and its derivative \dot{E} to zero asymptotically. This completes the proof.

3.7 Simulation

The simulation is performed based on the “rolling-spider parrot” minidrone. The quadrotor parameters are given in Table 3.1. The exponential reaching law sliding mode control (3.17) is applied to the quadrotor system (3.1) to track the trajectory with stability, in addition to the HPC compensator (3.14) to attenuate perturbation. The trajectory is chosen to be a circular shape with one-meter diameter where the desired height is one meter given by a smooth fifth-order polynomial. The simulated perturbation is a continuous sine wave signal $\alpha = a \sin(\omega t)$, $a = 0.05 u_{i-max}$, where u_{i-max} is the maximum value of the control input,

$w = 2\pi f$, $f = 1$ Hz. We will compare the results of the ERLSM controller with and without the HPC in the presence of the mentioned perturbation in order to observe the effectiveness of the proposed system.

Table 3.1 Quadrotor parameters

Parameter	Value	Unit
m	0.068	[kg]
i_x	0.0686×10^{-3}	[kg.m ²]
i_y	0.0920×10^{-3}	[kg.m ²]
i_z	0.1366×10^{-3}	[kg.m ²]
j_r	1.0209×10^{-7}	[kg.m ²]
g	9.81	[m/s ²]
l_e	0.1	[m]

It can be seen in Figures 3.3 and 3.4 that the perturbation causes clear distortion in the trajectory. Figure 3.5 displays the control signals. It can be seen obviously that the control alone is not able to reject the perturbation. This explains the needs of a perturbation compensator.

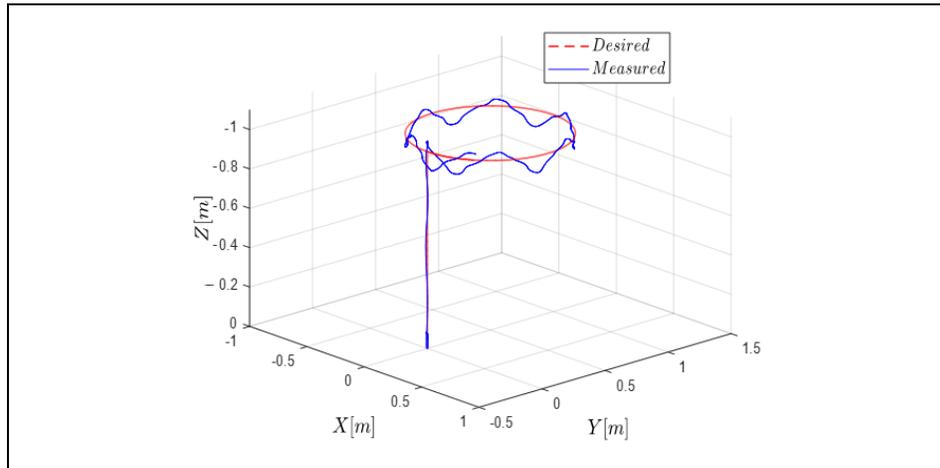


Figure 3.3 Trajectory in 3D-with perturbation, using ERLSM without the HPC

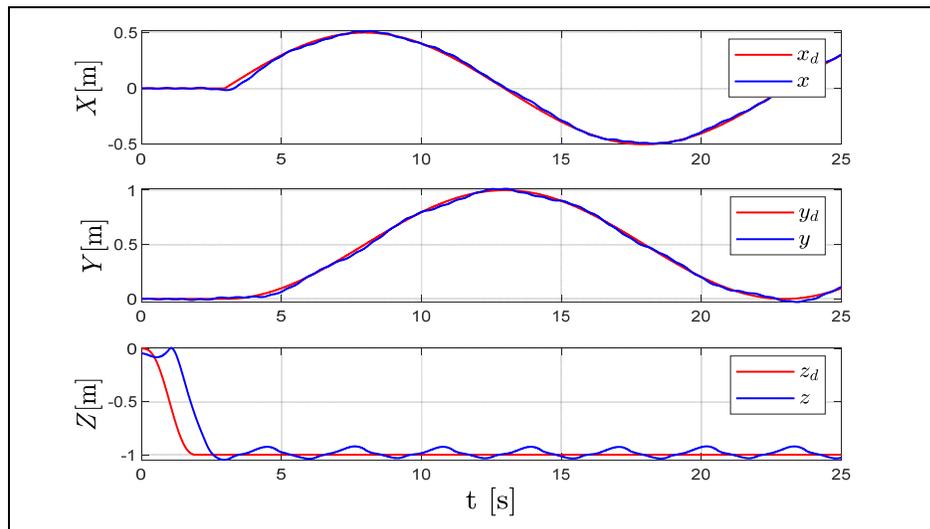


Figure 3.4 Position trajectory-with perturbation, using ERLSM without the HPC

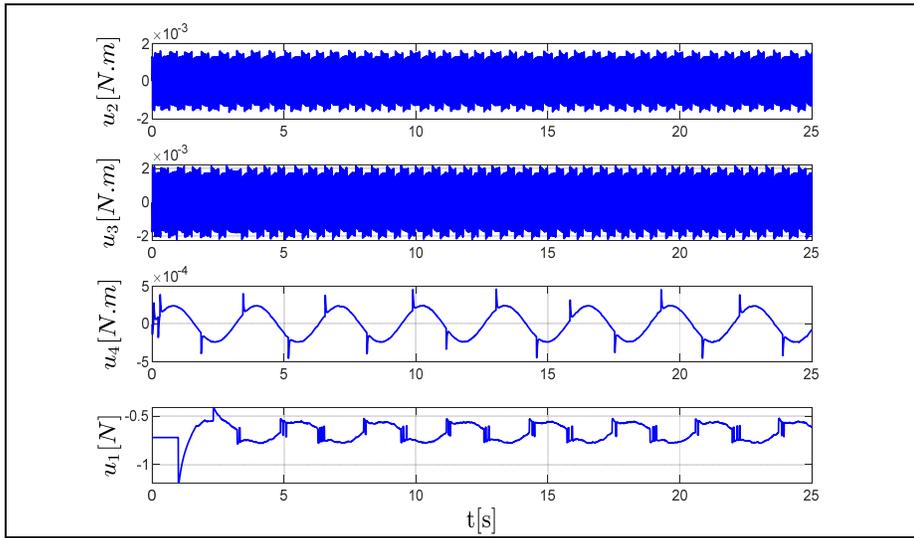


Figure 3.5 Control signals-with perturbation, using ERLSM without the HPC

After involving the HPC, the improved response can be seen as obtained in Figures 3.6 and 3.7. Figures 3.8 and 3.9 display the entire system control signals and the HPC signals. The performance is noticeable in the entire system. The proposed compensators are able to compensate perturbation in the speed of one time interval, which provides fast error compensation. This rapid action decreases the burden on the utilized control system.

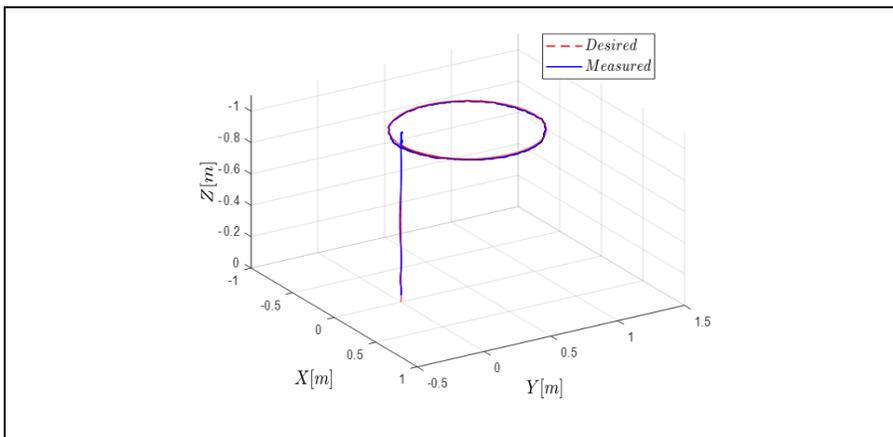


Figure 3.6 Trajectory in 3D-with perturbation, using ERLSM and the HPC

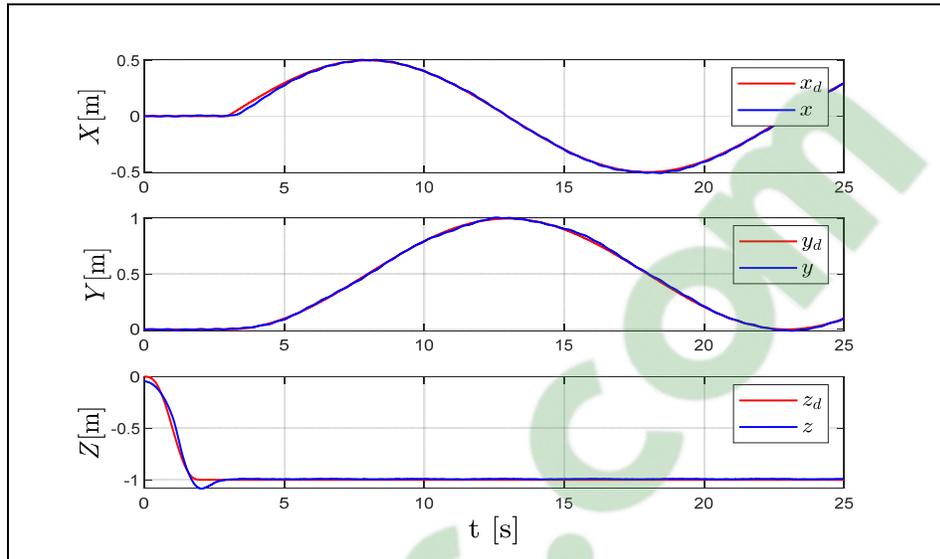


Figure 3.7 Position trajectory-with perturbation, using ERLSM and the HPC

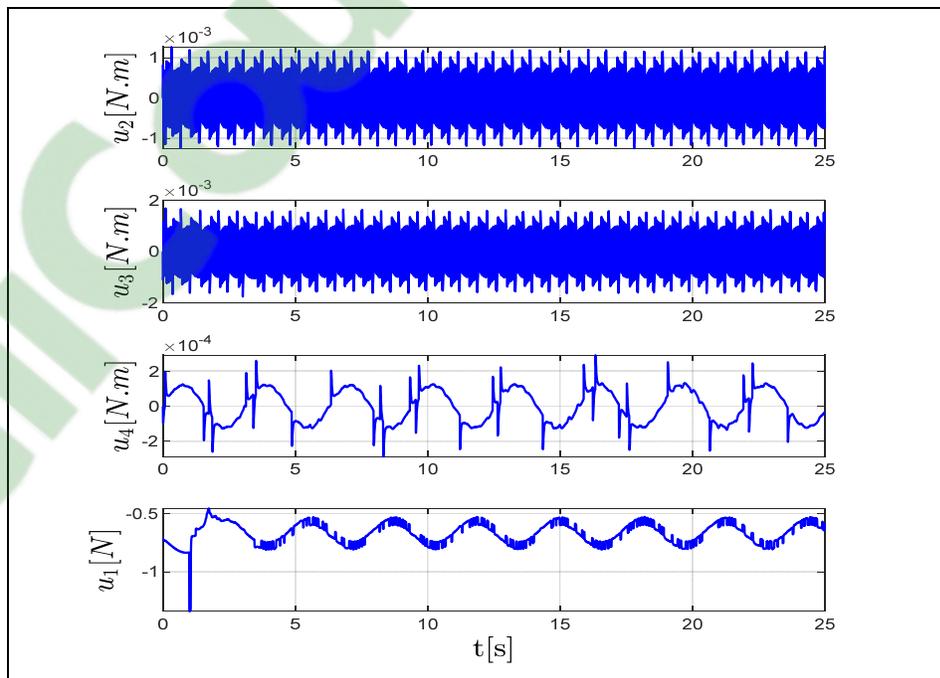


Figure 3.8 Control signals-with perturbation, using ERLSM and the HPC

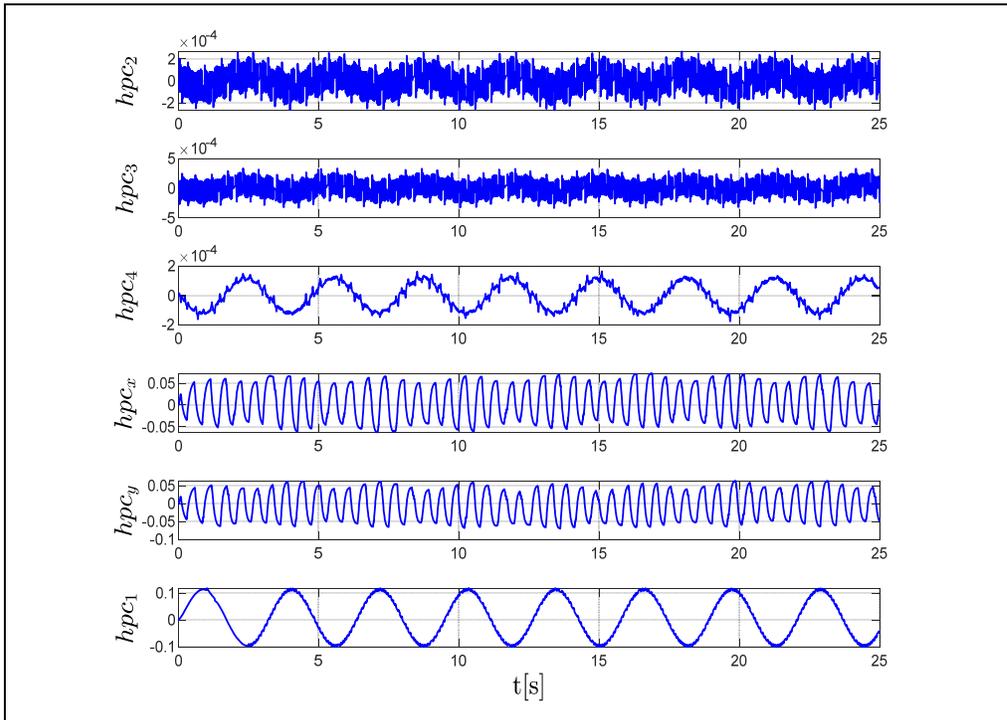


Figure 3.9 The HPC signal

Furthermore, the error root mean square (RMS) value is compared in both cases to provide numerical values of perturbation attenuation in Table 3.2. It is clear that the HPC verified good performance to keep the entire system stable and to reduce the effect of the applied perturbation.

In order to demonstrate the ability to reduce chattering in the exponential reaching law sliding mode in comparison with the traditional sliding mode controller, the simulation is repeated this time as HPC-SM (conventional sliding mode). By comparing Figure 3.8 and Figure 3.10, we can see that the HPC-ERLSM produce less chattering in the control.

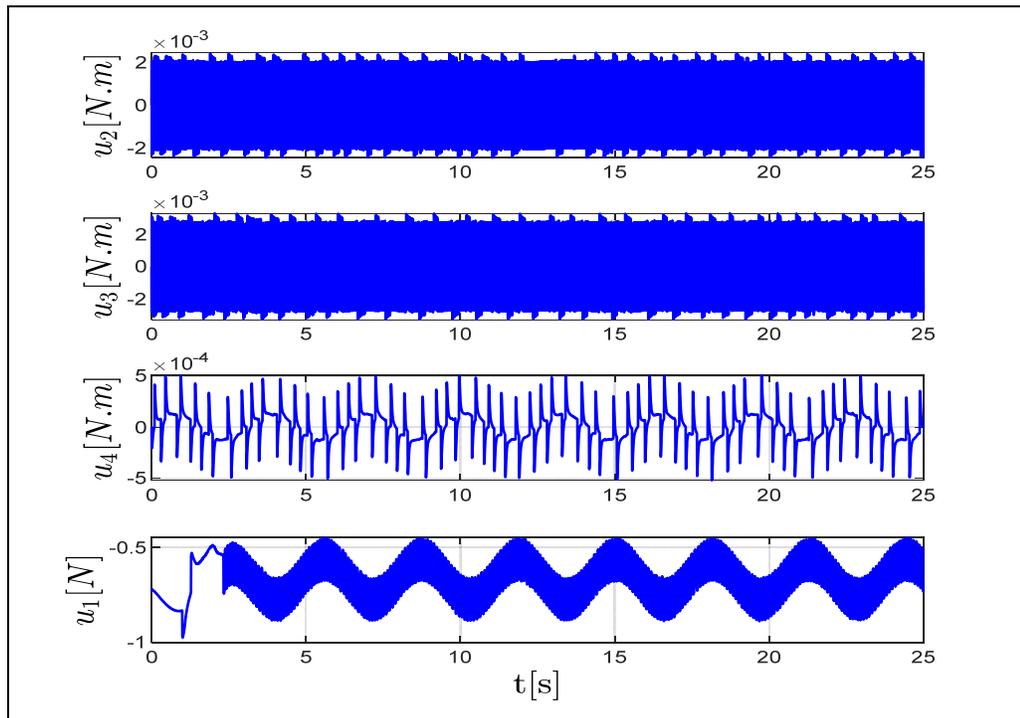


Figure 3.10 Control signals- with perturbation and using the HPC-SM

Table 3.2 Error RMS comparison

Parameter	HPC not applied	HPC applied
e_{x-rms}	0.0496	0.0208
e_{y-rms}	0.111	0.1056
e_{z-rms}	0.1155	0.0337

3.8 Experimental Results

Experimental results are demonstrated in this section to show the efficiency of the proposed HPC compensator to reject perturbation as well as the ERLSM to stabilize a quadrotor aircraft.

3.8.1 Real-time setup:

The experiment platform consists of a Parrot quadrotor minidrone. Parrot has an integrated IMU with a three-axis gyroscope three-axis accelerometer, a compass, as well as altitude sonar and pressure sensors. It is also equipped with a downward-facing camera 160x120 pixels and have a battery lifetime up to eight minutes.

The practical implementation is based on Simulink support package for PARROT minidrones (Mathworks, 2018). It facilitates building and deploying the flight control algorithm on the PARROT minidrones. Control algorithms were deployed wirelessly over Bluetooth and can access quadrotor onboard sensors such as the ultrasonic, accelerometer, gyroscope, and air pressure sensors. Simulink Coder™ allows recording flight data on the minidrone and access the C-code generated from Simulink models (Mathworks, 2018).

Implementation workflow can be summarized as in Figure 3.11. The inertial measurement unit (IMU) measures the body-fixed frame angular velocity vector $\Omega = [p \ q \ r]^T$ and body-fixed frame translational acceleration $\ddot{T}_B = [\ddot{x}_B \ \ddot{y}_B \ \ddot{z}_B]^T$.

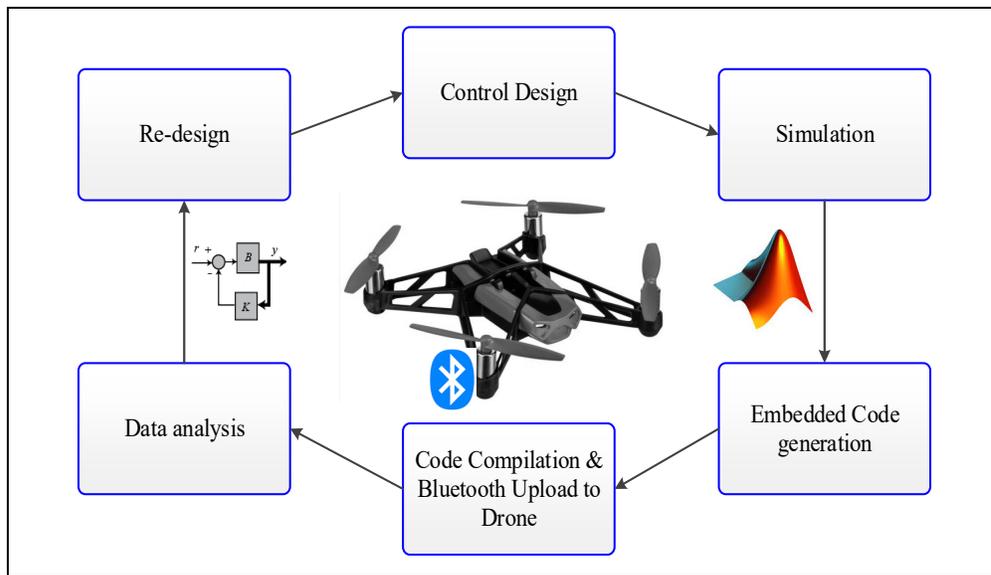


Figure 3.11 Implementation workflow

Euler angles rate of change in the inertial frame $\dot{O} = [\dot{\phi} \ \dot{\theta} \ \dot{\psi}]^T$ can be identified by using the transformation matrix (Emran, 2014):

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & s_{\phi} t_{\theta} & c_{\phi} t_{\theta} \\ 0 & c_{\phi} & -s_{\phi} \\ 0 & s_{\phi}/c_{\theta} & c_{\phi}/c_{\theta} \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (3.30)$$

Where, $s_{(\bullet)}$, $c_{(\bullet)}$, $t_{(\bullet)}$ are $\sin(\bullet)$, $\cos(\bullet)$, $\tan(\bullet)$ respectively. The delayed rotational acceleration $\ddot{O}(t - \tau)$ at time $(t - \tau)$ is found by the following approximation (Kali, Saad, & Benjelloun, 2018):

$$\ddot{O}(t - \tau) = \frac{1}{\tau^2} (O(t - \tau) - 2O(t - 2\tau) + O(t - 3\tau)) \quad (3.31)$$

Where $\ddot{O} = [\ddot{\phi} \ \ddot{\theta} \ \ddot{\psi}]^T$ is the rotational acceleration. The complementary filter is used to give the orientation based on the data from the gyroscope and the accelerometer as in (Mathworks, 2018) (Pieter-Jan, 2013), the gyroscope is precise and not susceptible to external forces while the accelerometer does not drift. The filter looks as follows:

$$\begin{aligned} O(t) &= 0.999(O(t - \tau) + R \Omega(t) \tau + 0.001 \gamma) \\ \mathbf{R} &= \begin{bmatrix} c_{\psi} c_{\theta} & -s_{\psi} c_{\phi} + c_{\psi} s_{\theta} s_{\phi} & s_{\psi} s_{\phi} + c_{\psi} s_{\theta} c_{\phi} \\ s_{\psi} c_{\theta} & c_{\psi} c_{\phi} + s_{\psi} s_{\theta} s_{\phi} & c_{\psi} s_{\phi} + s_{\psi} s_{\theta} c_{\phi} \\ -s_{\theta} & c_{\theta} s_{\phi} & c_{\theta} c_{\phi} \end{bmatrix} \\ \gamma &= [a \sin(\ddot{x}_B/g) \quad a \tan(\ddot{y}_B/\ddot{z}_B) \quad 0]^T \end{aligned} \quad (3.32)$$

Where, \mathbf{R} is the rotation matrix. The gyroscope data is integrated every time step with the current angle value. Then it is combined with the low-pass data from the accelerometer. The constants (0.999 and 0.001) have to add up to 1 but can be changed to tune the filter properly, they are selected based on (Mathworks, 2018). The translational acceleration in the inertial frame $\ddot{T} = [\ddot{x} \ \ddot{y} \ \ddot{z}]^T$ is found by the relation (Mathworks, 2018; Zipfel, 2007):

$$\ddot{T} = \mathbf{R} \ddot{T}_B \quad (3.33)$$

The velocity and the position are calculated by the following formulas (Mathworks, 2018; Zipfel, 2007) :

$$\begin{aligned} \dot{X} &= \mathbf{R} \left[\int \frac{F}{m} + V_B \times \Omega \right] \\ X &= \frac{k_1 \tau}{z^* - 1} \dot{X} \end{aligned} \quad (3.34)$$

Where $F = F_f + F_g$ contains the applied forces in body-fixed coordinate frame $F_f = R[0 \ 0 \ \sum_{i=1}^4 F_i]$, $F_i = b\omega_i^2$ for $i = 1,2,3,4$. $F_g = [0 \ 0 \ -mg]$. V_B is the velocity w.r.t. to the body frame, k_1 is a constant and it is a value of 0.01, and $X = [x \ y \ z]^T$ is the position vector and z^* is z-transform operator.

3.8.2 Practical implementation:

The performance of the HPC-ERLSM controller is evaluated experimentally in this part. The parrot quadrotor parameters are shown in Table 3.1. The HPC-ERLSM gains used in the experiment were determined experimentally $\mathbf{K} = \text{diag} \left(\frac{10}{N_1}, \frac{10}{N_2}, \frac{1}{N_3}, \frac{0.7}{N_4}, \frac{0.7}{N_5}, \frac{1.7}{N_6} \right)$, $\mathbf{A} = \text{diag}(6,6,5,1,1,3)$, $\delta_{0i} = 0.7$, $\alpha_i = 1$, $p_i = 1$ for $(i = 1,2..6)$. The trajectory is chosen as in the simulation part. Experimental results are presented in Figures 3.11 to 3.17. Figure 3.11 shows 3D and x-y task space tracking of the desired trajectory. Figure 3.12 shows trajectory tracking for each axis. Both figures show good error tracking during the whole operation time.

Orientation angles response is displayed in Figure 3.13, which shows fast response of the angles to stabilize the system. The velocity is found simultaneously as shown in Figure 3.14. Figure 3.15 shows the error signals, small value of the errors can be noticed. It can be seen in Figure 3.16 that the control torque inputs are small values. The proposed controller ensures

good tracking of the desired trajectory with accuracy due to HPC estimation of uncertain dynamics. The HPC signals are shown in Figure 3.17.

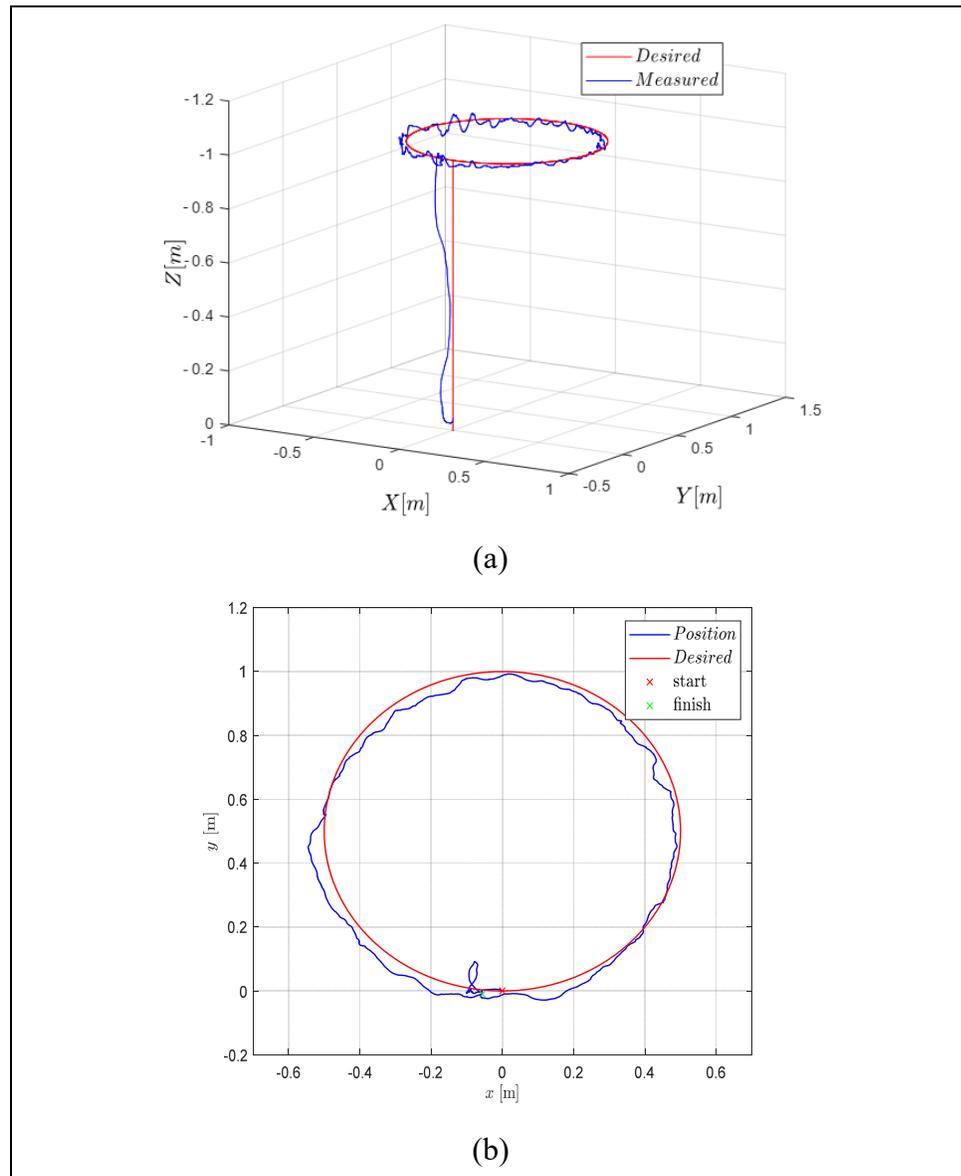


Figure 3.12 (a) The trajectory in 3D (b) The trajectory in x-y

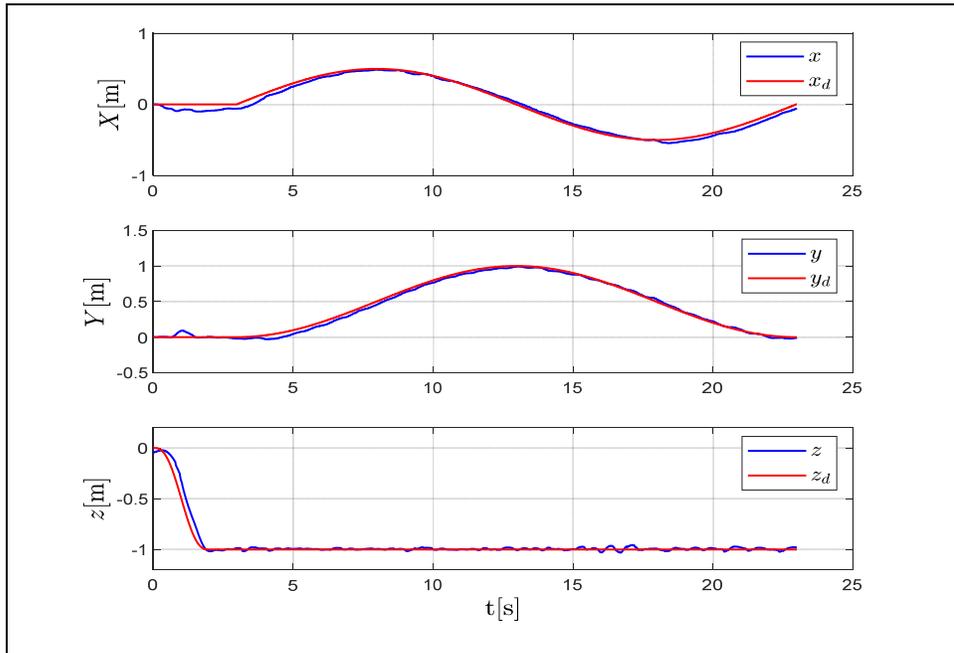


Figure 3.13 Position and altitude trajectory

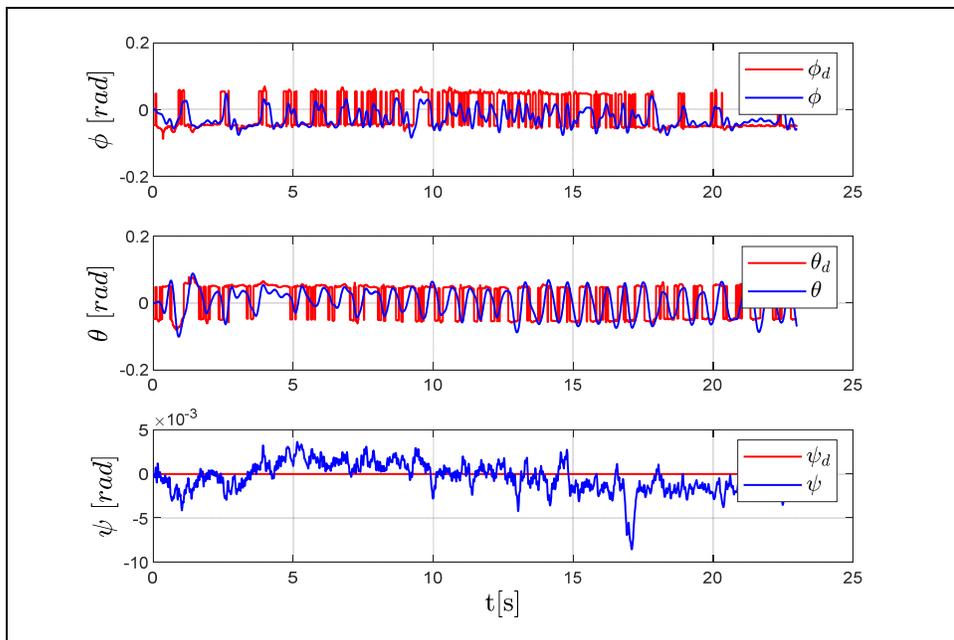


Figure 3.14 Euler angles response

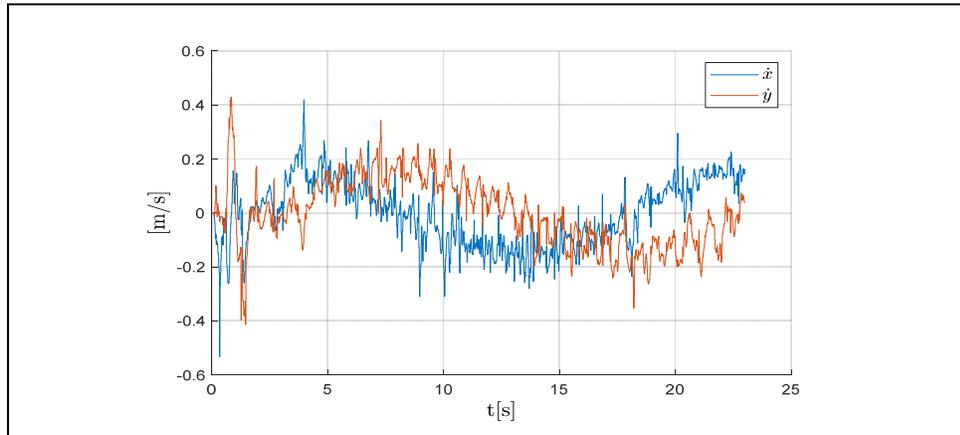


Figure 3.15 Velocities of x and y

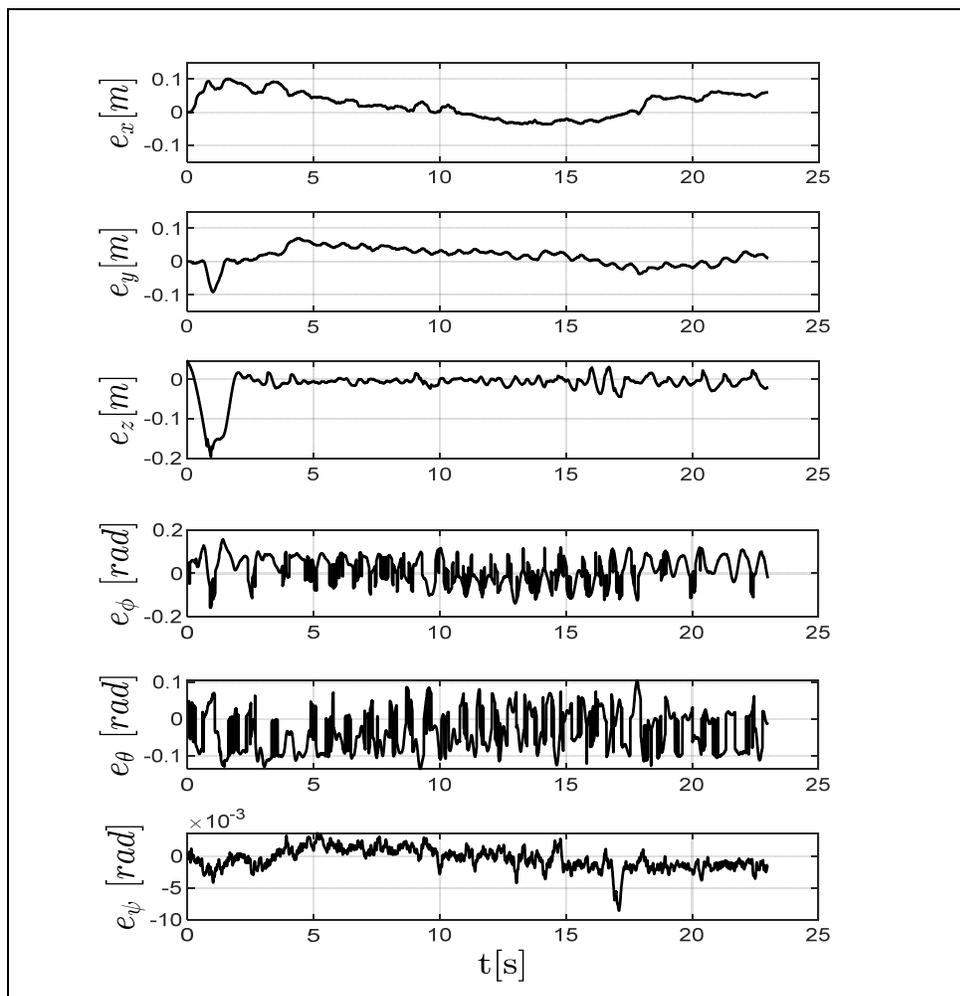


Figure 3.16 Errors in position, altitude and orientation

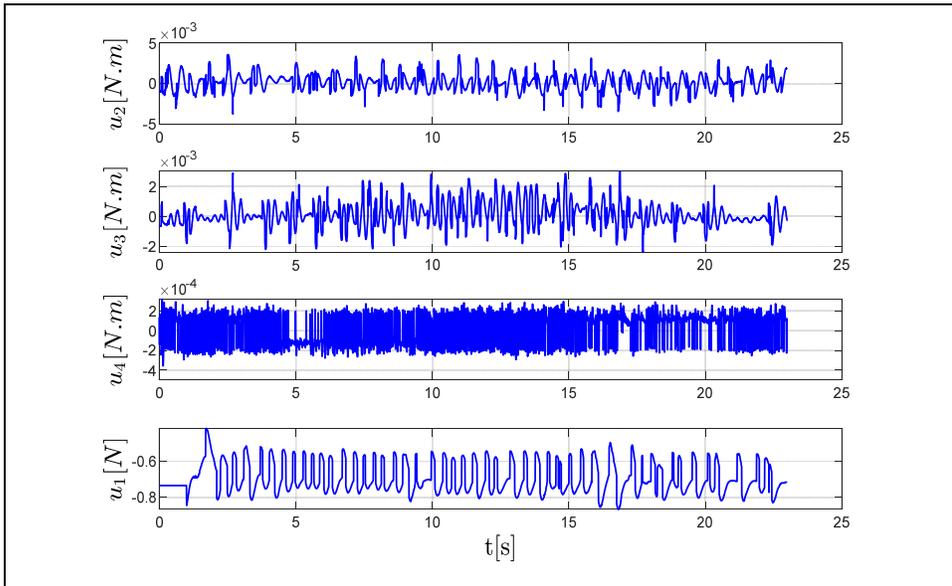


Figure 3.17 Control signals

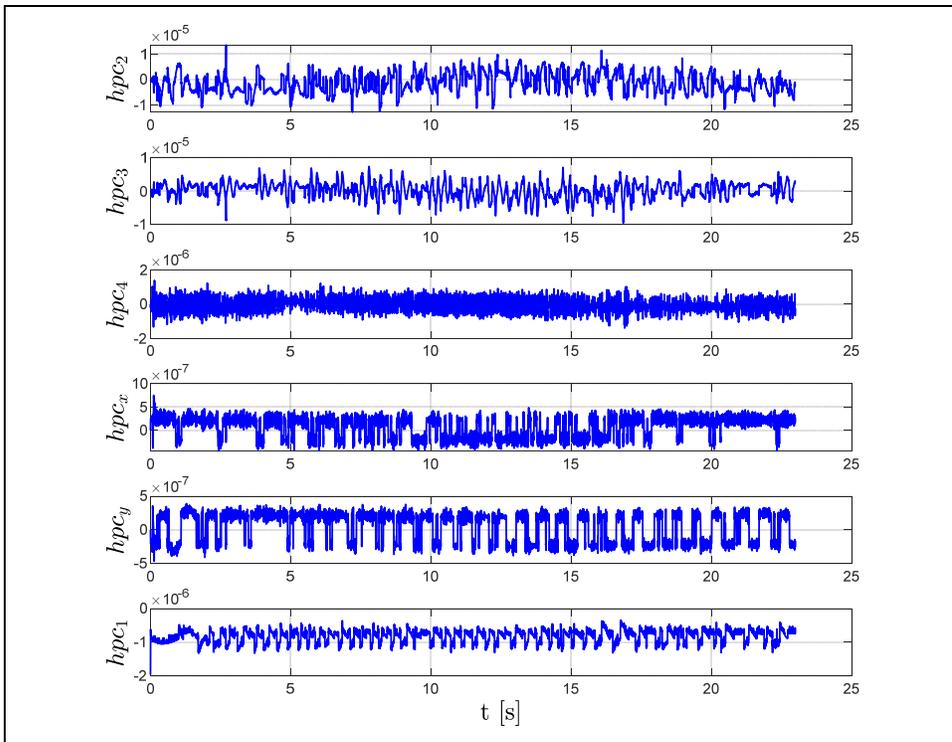


Figure 3.18 HPC signals

3.9 Conclusion

Perturbation in the UAV quadrotor is attenuated in this paper hierarchically by using the HPC. The HPC is combined of three subsystems FFPC, FBPC and SMPC. They are designed to reduce the perturbation and residual perturbation by comparing the system with the desired and the actual dynamics. Perturbation is rejected in the inner loop of the control by the FFPC and the FBPC while the closed loop dynamic error is rejected by the SMPC. ERLSM controller is implemented to provide fast response to the control with lowest possible chattering. The stability of the combined system of the HPC and the ERLSM is studied by Lyapunov analysis, simulation and experimental implementation, which verified the high performance of the HPC in reducing the effects of the perturbation.

CHAPTER 4

THREE-LOOP UNCERTAINTIES COMPENSATOR AND SLIDING MODE QUADROTOR UAV CONTROL WITH EXPONENTIAL REACHING LAW

Walid Alqaisi¹, Jawhar Ghommam², Anas Alazzam^{3,1} Maarouf Saad¹ and Vahé Nerguizian¹

¹ Department of Electrical Engineering, École de technologie supérieure;
1100 Notre-Dame West, Montreal, Quebec, Canada H3C 1K3

² Department of Electrical Engineering, Sultan Quaboos University,
Al Khoudh, Muscat 123, Oman

³ Department of Mechanical Engineering, Khalifa University,
Abu Dhabi, United Arab Emirates

Paper submitted for publication, April 2019

Abstract:

In this paper, a Three Loop Uncertainties Compensator (TLUC) and Exponential Reaching Law Sliding Mode Controller (ERSM) is proposed and successfully applied to a UAV quadrotor. The TLUC estimates unknown time-varying uncertainties and perturbations to reduce their effects and to preserve stability. The ERSM is integrated based on the Lyapunov stability theory to obtain fast response with lowest possible chattering. The novelty of this paper is that the TLUC can estimate and compensate for uncertainties and unknown time-varying disturbances in three loops. This provides tracking of residual uncertainty to provide higher level of support to the controller. The performance is verified through analyses, simulations and experiments.

Keywords: Feedback Compensation, Loop compensation, Quadrotor, Reaching law sliding mode, Time delay, UAV, Uncertainty compensation, Uncertainty estimation,

4.1 Introduction

During the last years, the research community showed a significant increase of interest in the flying vehicles in general and in particular, quadrotors. The ability to take-off and land vertically, fly at low speed and its simple structure encouraged implementing a lot of quadrotor applications.

Development of an effective flight system that is robust to the perturbation has been one of the primary objectives. Because of being an under-actuated system and a relatively small-sized robot, the quadrotor is more sensitive to uncertainties and disturbances than other types of robots. Uncertainties include, but not limited to, wind disturbance, air friction, uncertain parameters, and non-modelled dynamics. Uncertainties problem in robotics system is a wide area of research. To deal with this problem, some studies proposed a robust controller system, such as, a disturbance rejection control (Chang et al., 2016; Sanz et al., 2016), a cascade control law (H. Liu et al., 2017), backstepping controller (Cabecinhas et al., 2015), and fuzzy logic-based controller (Kayacan & Maslim, 2017). However, such controllers do not have adaptation properties that could reduce their performance.

Other researchers developed adaptation functionality in their control. For example, adaptive control method is used to adapt to disturbance (F. Chen et al., 2014), an adaptive output feedback compensator (Marino & Tomei, 2016a), an adaptive time-varying compensation is constructed for a quadrotor under uncertainties (Ton et al., 2016). Nevertheless, they lack estimation and compensation of the perturbation during practical operations, this encouraged researchers to design disturbance observers (X. Wang et al., 2015; Yin & Xiao, 2017), Luenberger observer with feedback linearization (Mokhtari et al., 2006), a sliding mode-based disturbance observer (Lénaïck Besnard et al., 2007; Besnard et al., 2012; Zhang et al., 2013), an acceleration-based observer for attitude control (Jeong et al., 2012b), and using an extended observer with feedback sliding mode (Rongting Zhang et al., 2011). However, such systems suffer from some drawbacks such as the influence of measurement lag and sensor noise which adds disturbance, meanwhile they lack tracking and rejecting residual perturbation.

A Hierarchical Perturbation Compensator HPC is detailed in (S. Kwon & W. K. Chung, 2004), despite its great advantages in attenuating perturbation, there is an unavoidable estimation error, if we assume ideal sensors, then the estimation error is $\tilde{\Gamma}(t) = \Gamma(t) - \Gamma(t - \tau)$ (where $\Gamma(t)$ is the perturbation, (t) is the time, (τ) is the sampling time), resulting from one step delayed input/output. Consequently, the performance of HPC depends on the norm $\|\tilde{\Gamma}(t)\|$. A satisfactory performance is possible on a condition that perturbation is continuous and differentiable and doesn't vary greatly during a small period of time τ , which is a reasonable assumption in most observer applications. For that reason, an efficient method to further attenuate error variations and to enhance the performance of the controlled system is required.

With an endeavor desire to overcome such concerns, this study implements Three-Loop Uncertainty Compensator TLUC in order to track uncertainties in three loops. The loops have the ability to track perturbation and residual perturbation. Each loop provides estimation and compensation of perturbation simultaneously based on time delay estimation. The TLUC has adaptive control property as it generates control effort that is required to compensate the current perturbation. Furthermore the TLUC has an integral control feature as the current compensation value is estimated based on delayed input.

Sliding mode control is one of the robust nonlinear control systems. Its control law is not a continuous function of time, conversely it switches from one continuous structure to another based on the current position in the state space. The fact that the sliding mode control is a variable structure control method it causes an undesired phenomenon called chattering. The chattering produce vibration and heat which cause damage to the used equipment. There are many solutions proposed to reduce the chattering problem, such as high order sliding mode (Benallegue, Mokhtari, & Fridman, 2008), super-twisting algorithm (Dávila, Moreno, & Fridman, 2010; Derafa et al., 2012) and modified super-twisting control (Kamal, Chalanga, Bera, & Bandyopadhyay, 2012) and multivariable super twisting (Alqaisi, Brahmi, Ghommam, Saad, & Nerguizian, 2018a). Perturbation compensation (Alqaisi et al., 2018b) is a technique used to reduce errors and so mitigating the chattering effect, despite the fact that it has no direct effect on the sliding surface. However, when the errors are far from the desired

sliding surface, the finite time of convergence of the selected surface is not ensured. To deal with the mentioned problems. Power rate reaching law is introduced (J. Liu & Wang, 2012a) to decrease the gain near the sliding surface, at the same time the gain rapidly decreases because of the fractional power thus it reduces the robustness of the controller near the sliding surface. To overcome this shortcoming we utilize Exponential Reaching Law Sliding Mode ERSM (C. J. Fallaha et al., 2011a), which uses a dynamic gain value. In this method, the gain takes high value when the error is high and takes low value when the error is small. The variation of the gain ensures quick convergence and avoids high chattering.

The system of TLUC-ERSM is implemented on six degrees of freedom quadrotor, the implementation includes Lyapunov analysis, simulations, and experiments.

The contributions of this paper can be summarized as:

- 1) Design a three-Loop Uncertainty Compensator TLUC in order to track perturbation and residual perturbation in three loops. Each loop provides estimation and compensation of perturbation simultaneously based on one-step time delay.
- 2) Real-time estimation and compensation involve adaptive and integral features of the proposed TLUC.

The major outlines of this paper can be described as:

- 1) Three-Loop Uncertainty Compensator TLUC is built and applied to the UAV quadrotor to reduce uncertainties and disturbances and to track residual perturbation in three loops.
- 2) The proposed system has adaptive control property as it generates control effort that is required to compensate the current perturbation.
- 3) The proposed system has an integral control feature as the current compensation value is estimated based residual perturbation.

- 4) The ERSM ensures full control to the position, attitude and altitude. It also guarantees low chattering and fast response as a result, the closed-loop system can be driven to asymptotic stability.
- 5) The entire system of the TLUC-ERSM confirmed high trajectory tracking performance as proved by analysis, simulations, and experiments.

This article is organized as follows; Section 5.2 describes the quadrotor model. Section 5.3 explains the TLUC structure. Section 5.4 studies the boundedness of the proposed TLUC. Section 5.5 includes ERSM control design and Lyapunov stability analysis for the whole system. Section 5.6 demonstrates the simulations with and without applying the TLUC. Section 5.7 demonstrates experimental results and analysis. The conclusion is in section 5.8.

4.2 Quadrotor Model

Based on Lagrange and Newton-Euler the quadrotor dynamics is built and used by many researchers (Samir Bouabdallah, 2007a; S. Bouabdallah & Siegwart, 2007; Bresciani, 2008; Erginer & Altuğ, 2007). Quadrotor configurations, frames, and forces are shown in Figure 4.1. The nonlinear quadrotor system is described as (Slotine & Li, 1991):

$$\ddot{X}(t) = F_T(\dot{X}(t)) + G_T(X(t))U(t) + D(t) \quad (4.1)$$

Where $X(t) = [\phi(t), \theta(t), \psi(t), x(t), y(t), z(t)]^T$ is the state vector, and $\dot{X}(t), \ddot{X}(t)$ are velocity and acceleration vectors, respectively. $F_T(\dot{X}(t))$ and $G_T(X(t))$ are the total nonlinear dynamics of the quadrotor system which include known and unknown dynamics, $D(t)$ denotes the external disturbance vector.

The dynamic system in (4.1) can be rewritten by describing the nominal part and the uncertain/changing dynamic part as:

$$\ddot{X}(t) = F(\dot{X}(t)) + \Delta F(\dot{X}(t)) + (G(X(t)) + \Delta G(X(t))) U(t) + D(t) \quad (4.2)$$

Where $F(\dot{X}(t)), G(X(t))$ are the nominal dynamics and they are given as:

$$F(\dot{X}(t)) = \begin{bmatrix} \frac{i_y - i_z}{i_x} \dot{\theta} \psi - \frac{j_r}{i_x} \dot{\theta} \omega_r \\ \frac{i_z - i_x}{i_y} \dot{\phi} \psi + \frac{j_r}{i_y} \dot{\phi} \omega_r \\ \frac{i_x - i_y}{i_z} \dot{\theta} \dot{\phi} \\ 0 \\ 0 \\ g \end{bmatrix} \quad (4.3)$$

$$G(X(t)) = \text{diag} \left[\frac{1}{i_x}, \frac{1}{i_y}, \frac{1}{i_z}, \frac{1}{m}, \frac{1}{m}, \frac{-1}{m} \cos \phi \cos \theta \right]$$

Functions $\Delta F(\dot{X}(t))$ and $\Delta G(X(t))$ are the uncertain terms of the dynamics. The input vector $U(t) = [u_2(t), u_3(t), u_4(t), u_1(t)u_x(t), u_1(t)u_y(t), u_1(t)]^T$ is defined as (S. Bouabdallah & Siegwart, 2007). The control inputs for the attitude and altitude are u_1, u_2, u_3 and u_4 while u_x, u_y are auxiliary control input designed to generate the reference signals of the roll and pitch angles, desired roll ϕ_d and desired pitch θ_d , then, the roll and pitch are controlled in u_2, u_3 . The control signals and the auxiliary controls are given as:

$$\begin{aligned} u_1 &= b(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \\ u_2 &= b l_e(\omega_1^2 + \omega_4^2 - \omega_3^2 - \omega_2^2) \\ u_3 &= b l_e(\omega_1^2 + \omega_2^2 - \omega_3^2 - \omega_4^2) \\ u_4 &= d(-\omega_1^2 + \omega_2^2 - \omega_3^2 + \omega_4^2) \\ u_x &= (\cos \psi \sin \theta \cos \phi + \sin \psi \sin \phi) \\ u_y &= (\sin \psi \sin \theta \cos \phi - \cos \psi \sin \phi) \end{aligned} \quad (4.4)$$

The desired roll and pitch can be found as (Gupta & Kothari, 2017; Khebbache, 2018):

$$\begin{aligned}\phi_d &= \sin^{-1}(u_x \sin \psi_d - u_y \cos \psi_d) \\ \theta_d &= \sin^{-1}\left(\frac{u_x \cos \psi_d + u_y \sin \psi_d}{\cos \phi_d}\right)\end{aligned}\quad (4.5)$$

Knowing that,

ϕ, θ and ψ : Roll, pitch and yaw angles respectively [rad].

i_x, i_y and i_z : Moments of inertia about body frame in x, y and z axes respectively [kg.m²].

m : Total mass [kg].

g : Gravity force [m/s²]

j_r : Rotor inertia [kg.m²].

b : Thrust coefficient [kg.m].

d : Drag coefficient [kg.m²].

l_e : Length of the moment arm [m].

ψ_d : The desired yaw angle [rad].

ω_i : Angular velocity. ($i = 1,2,3,4$) [rad/s].

ω_r : The balance around z-axis, $\omega_r = -\omega_1 + \omega_2 - \omega_3 + \omega_4$ [rad/s].

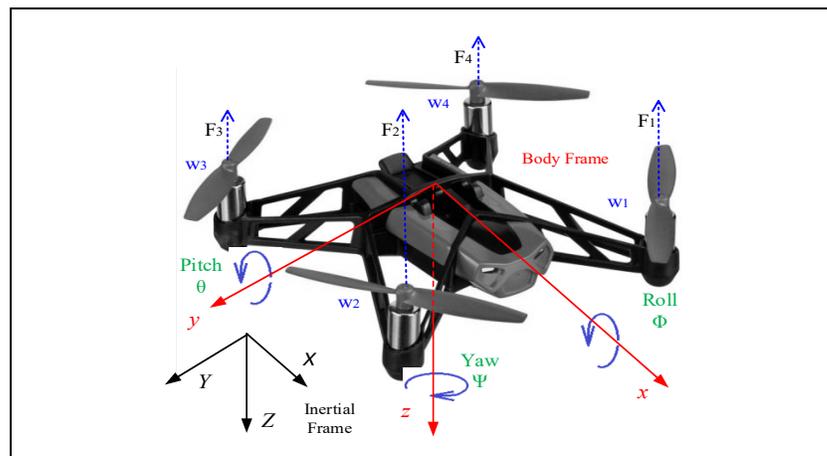


Figure 4.1 Quadrotor structure, forces, angles and frames ("Parrot Minidrone," 2018)

4.3 Three-Loop Uncertainty Compensator

Uncertainties in robotic systems include imperfection of modeling, air friction, and external disturbances. The uncertainties of the considered system can be described by the following equation:

$$\Gamma(t) = \Delta F(\dot{X}(t)) + \Delta G(X(t))U(t) + D(t) \quad (4.6)$$

Henceforth, the system dynamics in (2) can be written as:

$$\ddot{X}(t) = F(\dot{X}(t)) + \mathbf{G}(X(t))U(t) + \Gamma(t) \quad (4.7)$$

In this paper, it is required to provide estimation $\hat{\Gamma}(t)$ and compensation of uncertainties equivalent to the real perturbation $\Gamma(t)$. The estimated $\hat{\Gamma}(t)$ plays an important role in maintaining the system to the desired behavior.

Assumption: In the perturbation vector $\Gamma(t) = [\Gamma_1(t), \Gamma_2(t), \dots, \Gamma_6(t)]$, we assume the functions $\Gamma_i(t)$ for $i = 1, \dots, 6$ to be globally Lipschitz function. In other words, the functions $\Gamma_i(t)$ are continuous and differentiable and don't vary greatly during a small period of time (τ).

4.3.1 Main loop uncertainties compensator

The utilized main loop in the three-loop uncertainties compensator system consists of time delay estimation method. This loop utilizes a measured feedback to provide estimation based on the real system. The estimation of the main loop can be described in the following equations:

$$\hat{\eta}_1(t) \cong \Gamma(t - \tau) = \ddot{X}(t - \tau) - F(\dot{X}(t - \tau)) - G(X(t - \tau))U(t - \tau) \quad (4.8)$$

Where (τ) is the process step time and the delayed $\ddot{X}(t - \tau)$ is calculated as follows:

$$\begin{aligned}
\dot{X}(t - \tau) &= \frac{X(t - \tau) - X(t - 2\tau)}{\tau} \\
\dot{X}(t - 2\tau) &= \frac{X(t - 2\tau) - X(t - 3\tau)}{\tau} \\
\ddot{X}(t - \tau) &= \frac{\dot{X}(t - \tau) - \dot{X}(t - 2\tau)}{\tau} \\
\ddot{X}(t - \tau) &= \frac{1}{\tau^2} [X(t - \tau) - 2X(t - 2\tau) + X(t - 3\tau)]
\end{aligned} \tag{4.9}$$

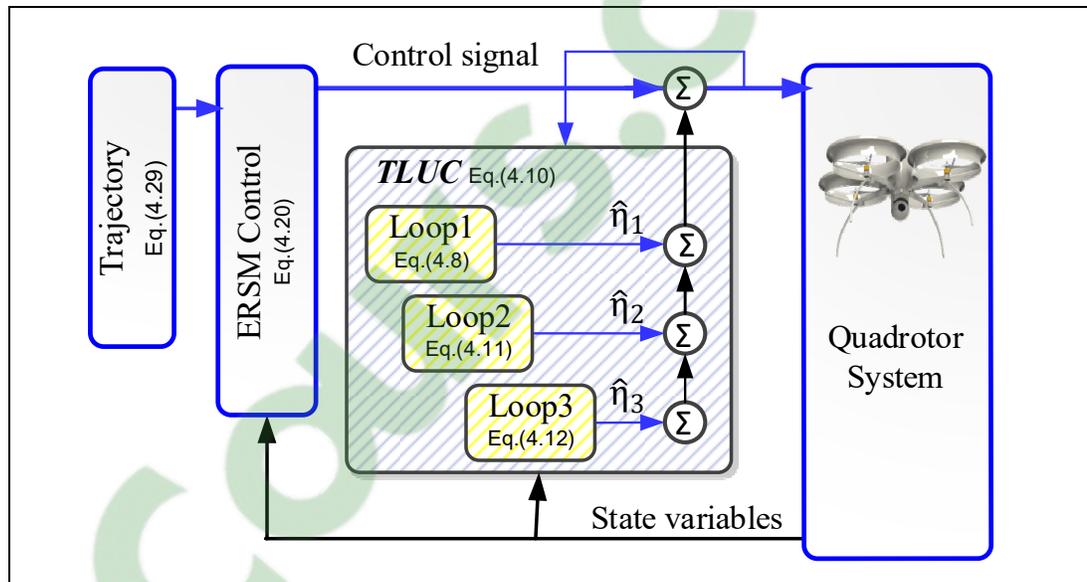


Figure 4.2 Three Loop Uncertainty Compensator block diagram

4.3.2 Three-loop Uncertainties Compensator

The proposed three-loop uncertainties compensator (Figure 4.2) is composed of three loops, the second and the third loops compensate the residual perturbation of the previous loop in order to reach a very small value of compensation error $\tilde{F}(t)$. The total compensation can be described as:

$$\hat{F}(t) = \hat{\eta}_1(t) + \hat{\eta}_2(t) + \hat{\eta}_3(t) \tag{4.10}$$

The compensation provided by the second loop $\hat{\eta}_2(t)$ is the current estimation $\Gamma(t - \tau)$ subtracted by the estimation of the first loop $\hat{\eta}_1(t - \tau)$ i.e. $\hat{\eta}_2(t) = \Gamma(t - \tau) - \hat{\eta}_1(t - \tau)$, by using (4.8) we can write:

$$\begin{aligned}\hat{\eta}_2(t) &= \Gamma(t - \tau) - \hat{\eta}_1(t - \tau) \\ &= \Gamma(t - \tau) - \Gamma(t - 2\tau)\end{aligned}\quad (4.11)$$

Similarly, $\hat{\eta}_3(t)$ is defined as:

$$\begin{aligned}\hat{\eta}_3(t) &= \Gamma(t - \tau) - \hat{\eta}_1(t - \tau) - \hat{\eta}_2(t - \tau) \\ &= \Gamma(t - \tau) - 2\Gamma(t - 2\tau) + \Gamma(t - 3\tau)\end{aligned}\quad (4.12)$$

4.4 Boundedness of perturbation compensators

Analyzing boundedness of the proposed three loop uncertainties compensator TLUC is an important step to evaluate the stability of the whole system. Starting from the definition of compensation error, and using (4.10), (4.11) and (4.12) we find:

$$\begin{aligned}\tilde{\Gamma}(t) &= \Gamma(t) - \hat{\Gamma}(t) \\ \tilde{\Gamma}(t) &= \Gamma(t) - (\hat{\eta}_1(t) + \hat{\eta}_2(t) + \hat{\eta}_3(t)) \\ \tilde{\Gamma}(t) &= \Gamma(t) - 3\Gamma(t - \tau) + 3\Gamma(t - 2\tau) - \Gamma(t - 3\tau)\end{aligned}\quad (4.13)$$

Then we can write:

$$\begin{aligned}\tilde{\Gamma}_i(t) &= \Gamma_i(t) - 3\Gamma_i(t - \tau) + 3\Gamma_i(t - 2\tau) - \Gamma_i(t - 3\tau) \\ \tilde{\Gamma}_i(t) &= [\Gamma_i(t) - \Gamma_i(t - \tau)] + 2[\Gamma_i(t - 2\tau) - \Gamma_i(t - \tau)] + [\Gamma_i(t - 2\tau) - \Gamma_i(t - 3\tau)] \\ \tilde{\Gamma}_i(t) &\leq |\Gamma_i(t) - \Gamma_i(t - \tau)| + 2|\Gamma_i(t - 2\tau) - \Gamma_i(t - \tau)| + |\Gamma_i(t - 2\tau) - \Gamma_i(t - 3\tau)|\end{aligned}\quad (4.14)$$

As $\Gamma_i(t)$ is assumed to be a Lipschitz function, the following relationship is true:

$$|\Gamma_i(t - a\tau) - \Gamma_i(t - b\tau)| \leq \gamma_i \tau |b - a| \quad (4.15)$$

Where $\gamma_i > 0$ is Lipschitz constant which is a very small value. Based on (4.14) and using (4.13) it can be found that:

$$\begin{aligned} \tilde{I}_i(t) &\leq \gamma_i \tau + 2\gamma_i \tau + \gamma_i \tau \\ \tilde{I}_i(t) &\leq 4\gamma_i \tau \end{aligned} \quad (4.16)$$

Both, sampling step time τ and Lipschitz constant γ_i are very small. It can be seen that estimation error is bounded.

4.5 The Control System

The quadrotor is well-known to be a highly nonlinear system. Sliding mode nonlinear controller maintains stability by using the error and its first derivative in the sliding surface (Slotine & Li, 1991). As a nonlinear control, it is designed to drive and force the system to remain within a region of a predetermined switching function. The dynamics can be adjusted by the chosen desired switching function. The undesired phenomenon of chattering can be solved by using the exponential reaching law sliding mode ERSM [25]. The idea of the ERSM is to give high values to the gains when the error is high to achieve quick convergence. At the same time, it provides low values to K when the error is small to avoid high chattering. The proposed reaching law is

$$\dot{S}(t) = -K \text{Sign}(S(t)) \quad (4.17)$$

Where $K = \text{dig} \left(\frac{k_1}{N(s_1)}, \frac{k_2}{N(s_2)}, \dots, \frac{k_n}{N(s_n)} \right)$, and $k_i > 0$ for $i = 1, 2, \dots, n$,

$N(s_i) = \delta_{0i} + (1 - \delta_{0i})e^{-\alpha_i |s_i|^{P_i}}$. δ_{0i} is a strictly positive offset $0 < \delta_{0i} < 1$, P_i and α_i are strictly positive adjustable parameters (C. J. Fallaha et al., 2011a).

We start by defining a tracking error $E(t) = X(t) - X_d(t)$, where X_d is the desired trajectory $X_d = [\phi_d, \theta_d, \psi_d, x_d, y_d, z_d]^T$. We define the sliding surface as $S(X, t) = 0$, it is also given as:

$$S = \dot{E} + \Lambda E \quad (4.18)$$

The sliding surface first derivative is given as:

$$\begin{aligned} \dot{S}(t) &= \ddot{E}(t) + \Lambda \dot{E}(t) \\ &= \ddot{X}(t) - (\ddot{X}_d(t) - \Lambda \dot{E}(t)) \\ &= \ddot{X}(t) - \ddot{X}_r(t) \end{aligned} \quad (4.19)$$

Where $\ddot{X}_r(t) = \ddot{X}_d(t) - \Lambda \dot{E}(t)$, $\Lambda = \text{diag}(\lambda_i)$, $\lambda_i (i = 1, 2, \dots, 6)$ are positive definite constants. The proposed controller based on the exponential reaching law sliding mode control is given as follows (Behal, Dixon, Xian, & Dawson, 2009; C. J. Fallaha et al., 2011a):

$$U(t) = G^{-1}(X) [\ddot{X}_r(t) - F(X(t)) - K \text{Sign}(S(t)) - \hat{\Gamma}(t)] \quad (4.20)$$

The function $\text{Sign}(S(t)) = [\text{sign}(s_1(t)), \dots, \text{sign}(s_6(t))]^T$ is given as:

$$\text{sign}(s_i(t)) = \begin{cases} 1 & \text{for } s_i(t) > 0 \\ 0 & \text{for } s_i(t) = 0 \\ -1 & \text{for } s_i(t) < 0 \end{cases}, \quad i = 1, 2, \dots, 6 \quad (4.21)$$

Theorem: Consider the nonlinear dynamics of six degrees of freedom quadrotor given by the dynamic equations (4.7). Let the uncertainty compensator TLUC be designed as in (4.10). If the control input for the quadrotor $U(t)$ is designed based on ERMS (4.20), then the closed loop system is asymptotically stable. That is:

$$\lim_{t \rightarrow \infty} E(t) = 0 \quad (4.22)$$

Provided that the gain controllers are selected such that the stability condition is met, $k_i > 4\gamma_i\tau N(s_i(t))$.

Proof: To prove the stability, the following Lyapunov function is selected:

$$\begin{aligned} V(t) &= \frac{1}{2} S^T(t) S(t) \\ \dot{V} &= S^T(t) \dot{S}(t) \end{aligned} \quad (4.23)$$

Substituting \dot{S} from (4.17) leads to:

$$\dot{V}(t) = S^T(t) [\ddot{X}(t) - \ddot{X}_r(t)] \quad (4.24)$$

Substituting \ddot{X} from (4.7) gives:

$$\dot{V}(t) = S^T(t) (F(X(t)) + G(X(t))U(t) + \Gamma(t) - \ddot{X}_r(t)) \quad (4.25)$$

Substituting the control (4.20) in (4.25) gives:

$$\begin{aligned} \dot{V}(t) &= S^T(t) (F(X(t)) \\ &\quad + G(X(t))G^{-1}(X(t)) [\ddot{X}_r(t) - F(X(t)) - K \text{Sign}(S(t)) - \hat{\Gamma}(t)] \\ &\quad + \Gamma(t) - \ddot{X}_r(t)) \end{aligned} \quad (4.26)$$

$$\dot{V}(t) = S^T(t) (-K \text{Sign}(S(t)) + \tilde{\Gamma}(t))$$

Where, $\tilde{\Gamma}(t) = \Gamma(t) - \hat{\Gamma}(t)$ is the estimation error or TDE error. Equation (4.26) can be written as:

$$\begin{aligned} \dot{V}(t) &= \sum_{i=1}^n [s_i(t) (\frac{-k_i}{N(s_i(t))} \text{sign}(s_i(t)) + \tilde{\Gamma}_i(t))] \\ &= \sum_{i=1}^n [\frac{-k_i}{N(s_i(t))} |s_i(t)| + s_i \tilde{\Gamma}_i(t)] \end{aligned} \quad (4.27)$$

$$\begin{aligned}
&\leq \sum_{i=1}^n |s_i(t)| \left[\frac{-k_i}{N(s_i(t))} + |\tilde{f}_i(t)| \right] \\
&\leq \sum_{i=1}^n |s_i(t)| \left[\frac{-k_i}{N(s_i(t))} + 4\tau\gamma_i \right]
\end{aligned}$$

In order to have a stable system, the following condition has to be met:

$$k_i > 4\gamma_i\tau N(s_i(t)) \quad (4.28)$$

Knowing that $0 < N(s_i(t)) < 1$ and γ_i, τ are very small values. Equation (4.28) proves that $\dot{V}(t) < 0$, which shows that $s_i \rightarrow 0$ as $t \rightarrow \infty$, this confirms that error converges to zero asymptotically.

4.6 Simulation

Numerical simulations are performed to prove the entire system functionality. The TLUC (4.10) and the ERSM (4.20) are applied to the quadrotor (4.1). The objective is to stabilize the system, track the trajectory, and to attenuate perturbation. The used parameters in Table 4.1 belong to the “rolling-spider” minidrone by “Parrot”.

The following reference trajectory is built to assess the quadrotors tracking performance:

$$\begin{aligned}
x_d &= \begin{cases} 0 & \text{if } t < 5 \\ 0.4 \sin(2\pi t/10) & \text{if } t \geq 5 \end{cases} \\
y_d &= \begin{cases} 0 & \text{if } t < 5 \\ 0.4 \sin(2\pi t/20) & \text{if } t \geq 5 \end{cases} \\
z_d &= \begin{cases} a_0 + a_1 t + a_2 t^2 + a_3 t^3 & \text{if } t < 2 \\ 1 & \text{if } t \geq 2 \end{cases} \\
\psi_d &= 0
\end{aligned} \quad (4.29)$$

The trajectory z_d is a third order polynomial, $a_0 = z_0$, $a_1 = 0$, $a_2 = \frac{3}{t_f^2}(z_f - z_0)$ and $a_3 = \frac{-2}{t_f^3}(z_f - z_0)$ where z_0 and z_f are the initial and final position values, t_f is the final time (C. J. Fallaha et al., 2011a). As seen in Figure 4.3, the trajectory is an infinity shape in x and y . The perturbation is a continuous sinusoidal signal $p = a * \sin(\omega t)$, $a = 0.05 * (u_i)_{max}$, where $(u_i)_{max}$ is the maximum value of the control input. $\omega = 2\pi f$, $f = 1$ Hz.

Table 4.1 Quadrotor parameters as used in the numerical simulations

Parameter	Value
Mass (m)	= 0.068 [kg].
Moment of Inertia (i_x)	= 0.6860×10^{-3} [kg.m ²].
Moment of Inertia (i_y)	= 0.0920×10^{-3} [kg.m ²].
Moment of Inertia (i_z)	= 0.1366×10^{-3} [kg.m ²].
Motor inertia (j_r)	= 1.0209×10^{-7} [kg.m ²].
Gravity (g)	= 9.81 [m/s ²].

A comparison is made with and without the TLUC to observe the effectiveness of the proposed system. The applied disturbance causes clear distortion in the response as seen in Figures 4.3 and 4.4. Figure 4.5 displays the ERSM control signals. The control has difficulties in rejecting the high perturbation which explains the need of a perturbation compensator. After involving the TLUC, the response is improved and as obtained in Figures 4.6 and 4.7, it can be noticed that the TLUC reduced the effect of the perturbation. Figures 4.8 and 4.9 display the entire system control signals and TLUC signals respectively. The influence of the proposed TLUC is noticeable over the complete system. The TLUC is able to provide estimation and compensation in the speed of a single time interval, which provides fast error convergence. This rapid action decreases the burden on the control system. The TLUC verified good performance to bring the entire system to stability and to reduce the effect of the applied perturbation.

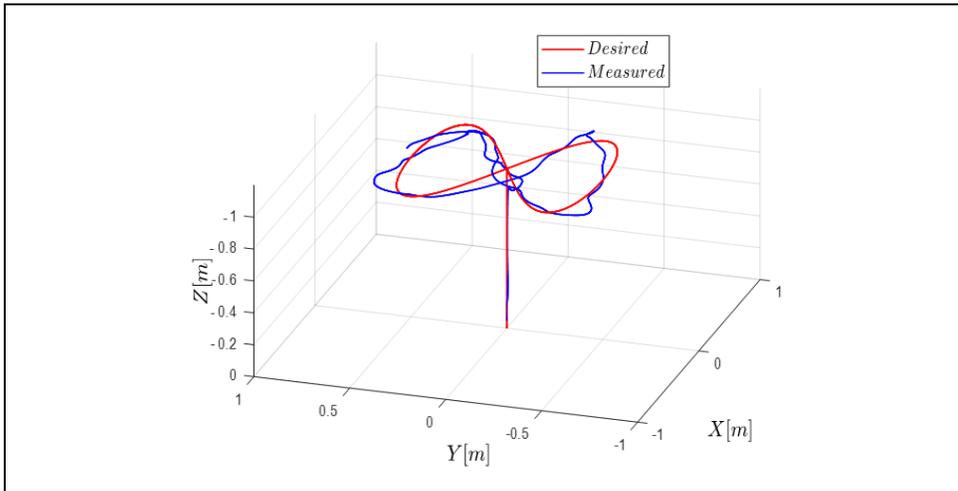


Figure 4.3 Trajectory in 3D-with perturbation, without the TLUC

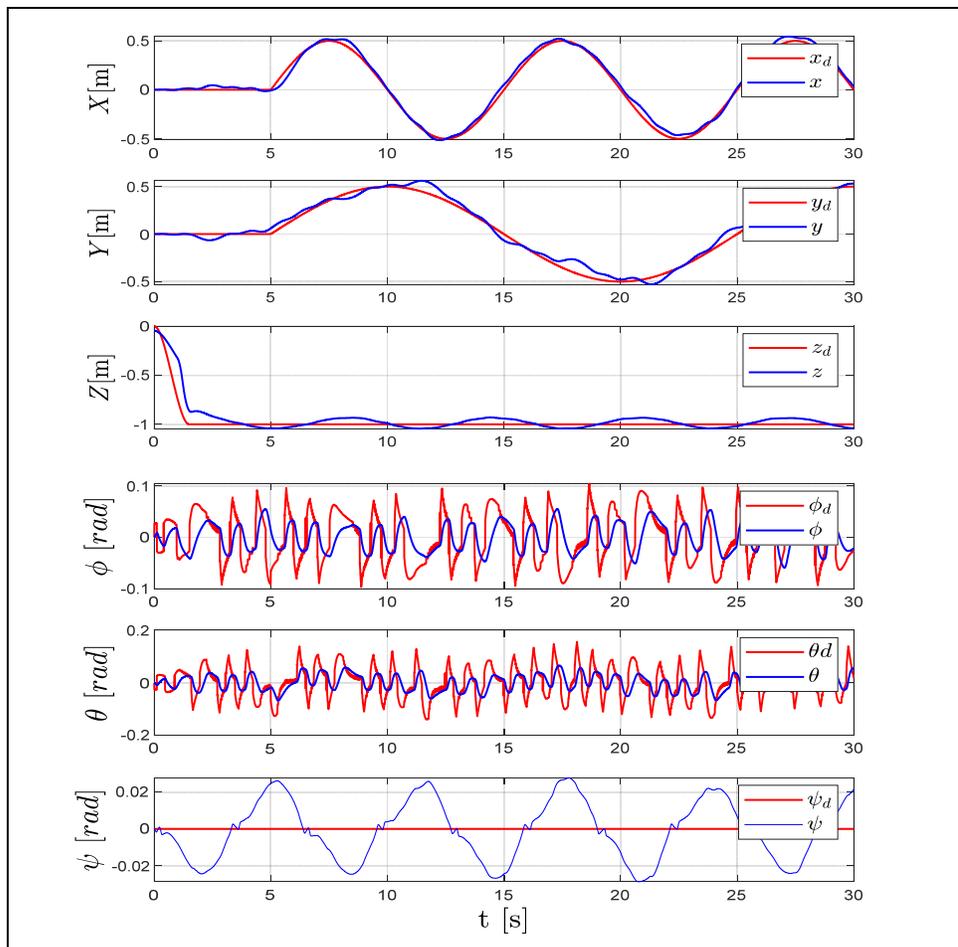


Figure 4.4 Position and attitude-with perturbation, without the TLUC

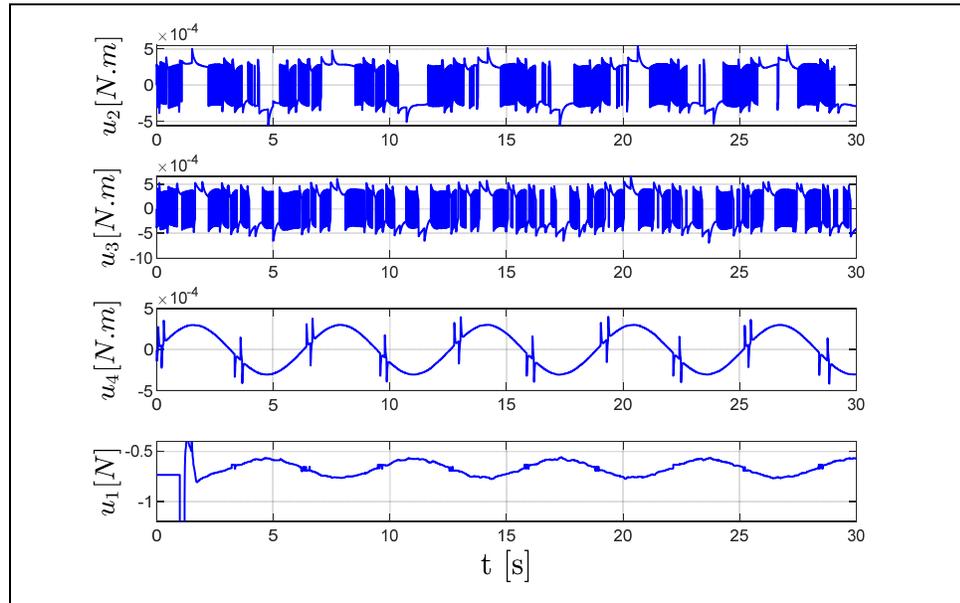


Figure 4.5 Control signals-with perturbation, without the TLUC

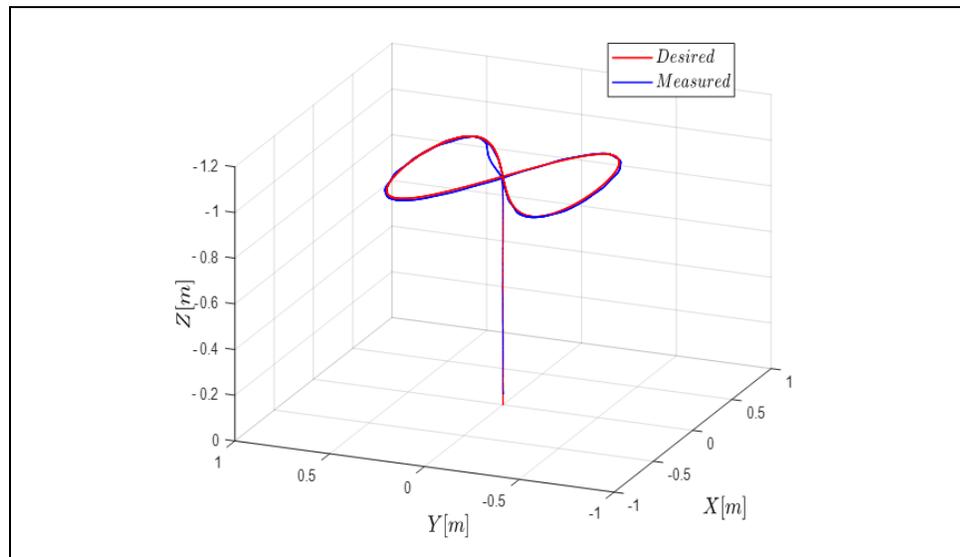


Figure 4.6 Trajectory in 3D-with perturbation, using the TLUC

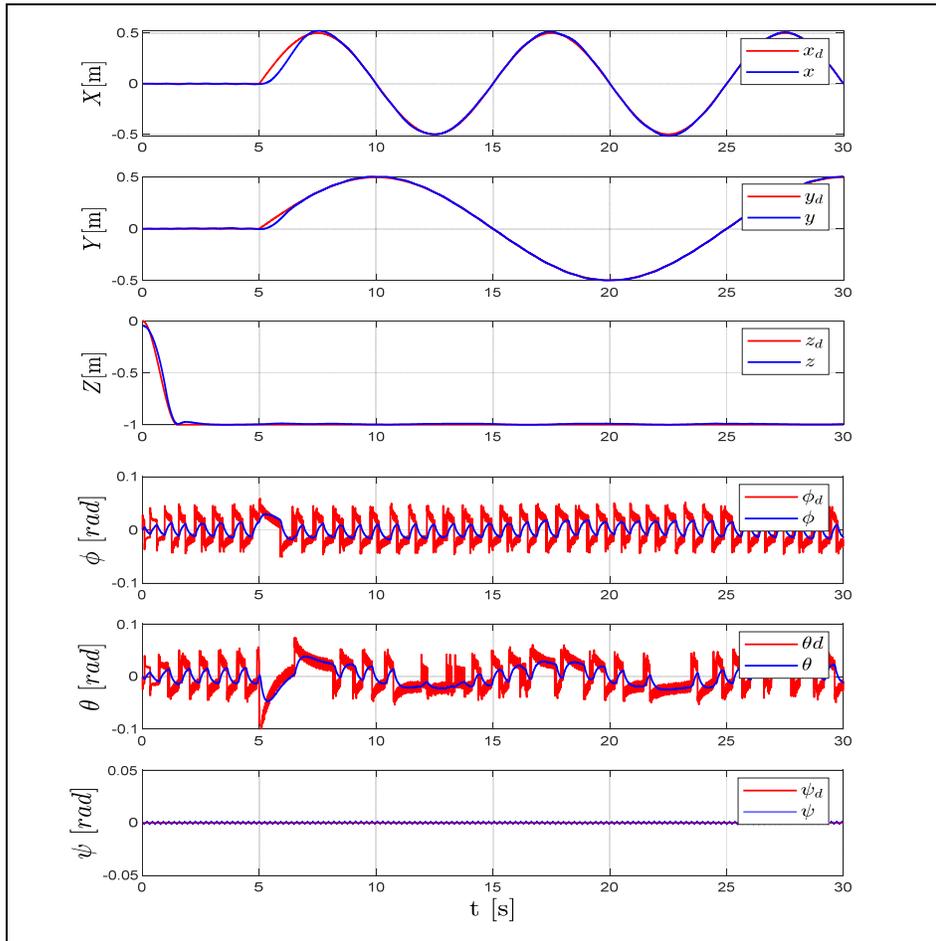


Figure 4.7 Position and attitude-with perturbation, using the TLUC

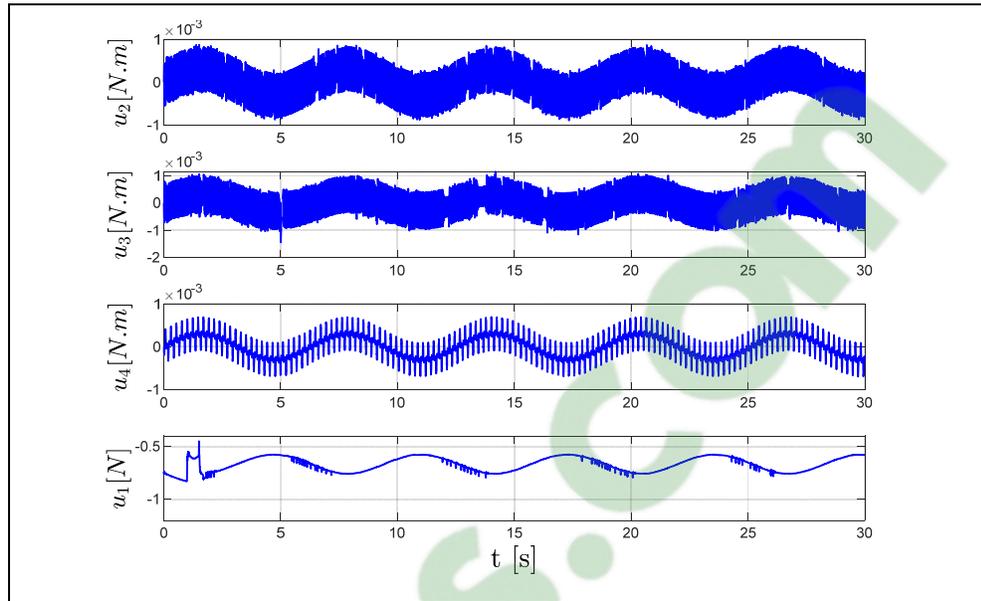


Figure 4.8 Control signals-with perturbation, using the TLUC

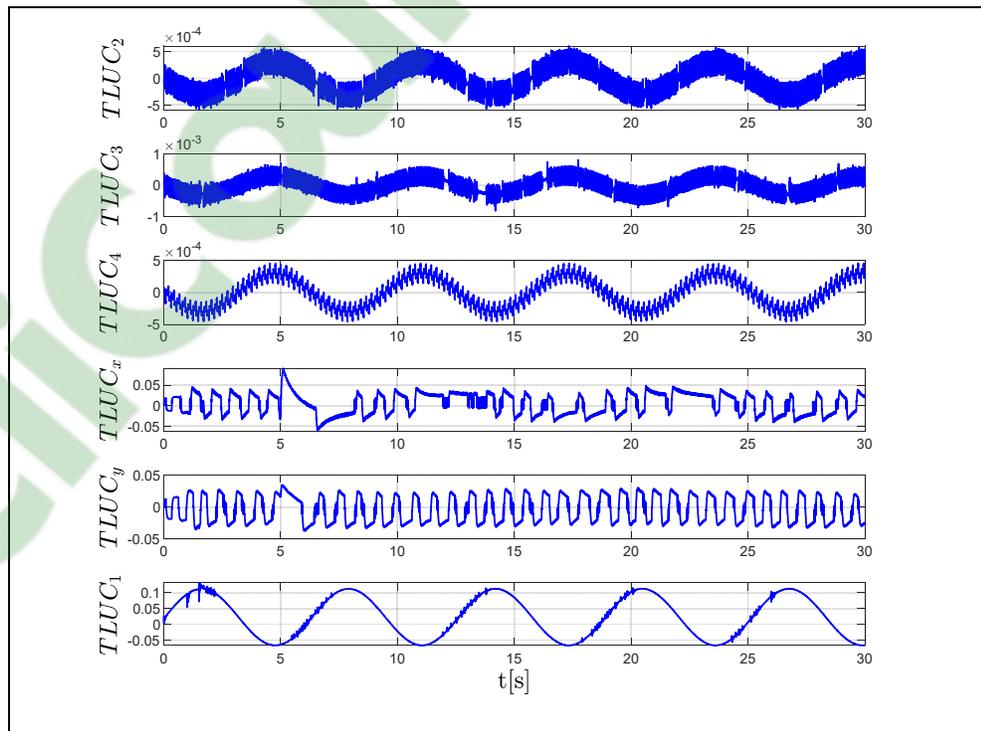


Figure 4.9 The TLUC signal

Furthermore, the root mean square value of the error (RMS) is compared in both cases to provide numerical values of perturbation attenuation, Table 4.2. As seen from the data in Table 4.2, the TLUC verified good performance to keep the entire system stable and to reduce the effect of the applied perturbation.

Table 4.2 Error RMS comparison

	TLUC not applied	TLUC applied
e_{x-rms}	0.0168	0.00183
e_{y-rms}	0.0055	0.000016
e_{z-rms}	0.0251	0.0105

4.7 Practical Implementation

Experiments are carried out in order to demonstrate the effectiveness of the proposed TLUC to attenuate perturbation as well as the ERSM to track the trajectory.

4.7.1 Experiment setup

Experimental platform consists of a “Parrot Rolling-spider” minidrone which is equipped with a three-axis gyroscope, three-axis accelerometer, altitude sonar, and a pressure sensor. In addition to a downward-facing camera 160x120 pixels attached to the drone. The practical experiment is based on Simulink support package for PARROT Minidrones (Mathworks, 2018) which facilitates building and deploying the flight control algorithm on Parrot minidrones. Implementation workflow can be summarized as in Figure 4.10. Control algorithms are deployed wirelessly over Bluetooth and can access quadrotor onboard sensors such as the ultrasonic, accelerometer, gyroscope and air pressure. Simulink Coder™ allows

recording flight data on the minidrone and access the C code generated from Simulink models (Mathworks, 2018).

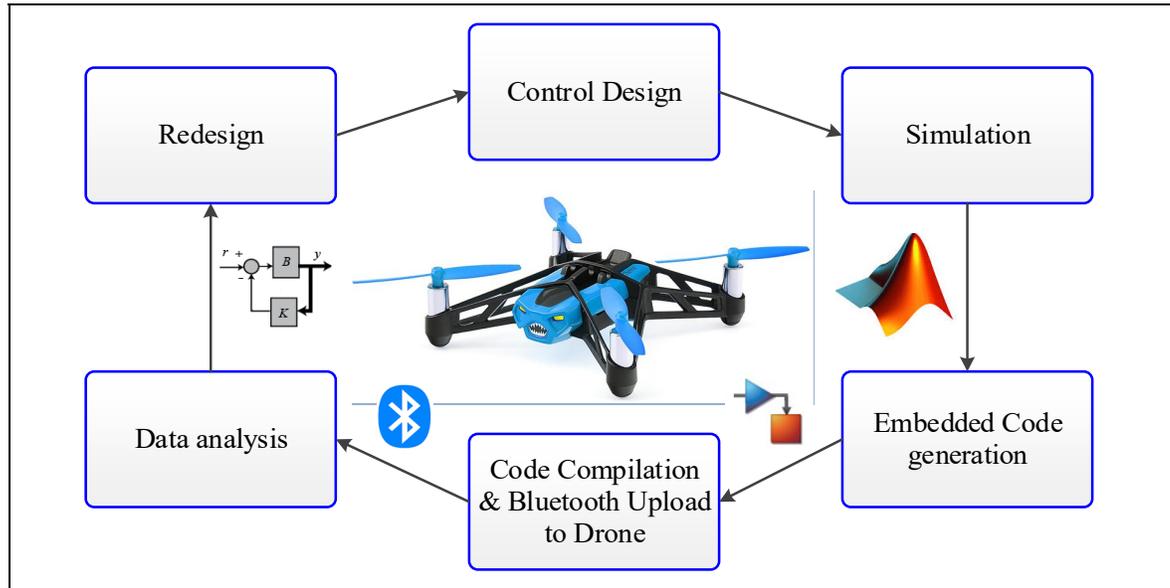


Figure 4.10 Implementation workflow

The inertial measurement unit (IMU) measures the body-fixed frame angular velocity vector $W = [p \ q \ r]^T$ and body-fixed frame translational accelerations $\ddot{T}_B = [\ddot{x}_B \ \ddot{y}_B \ \ddot{z}_B]^T$. Euler angles rate of change in the inertial frame $\dot{O} = [\dot{\phi} \ \dot{\theta} \ \dot{\psi}]^T$ can be identified by using the transformation matrix as:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & s_\phi t_\theta & c_\phi t_\theta \\ 0 & c_\phi & -s_\phi \\ 0 & s_\phi/c_\theta & c_\phi/c_\theta \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (4.30)$$

Where $s_{(\bullet)}$, $c_{(\bullet)}$, $t_{(\bullet)}$ are $\sin(\bullet)$, $\cos(\bullet)$, $\tan(\bullet)$ respectively. The complementary filter is used to give the orientation of the drone based on the data from the gyroscope and the accelerometer (Mathworks, 2018; Van de Maele), the gyroscope is precise and not susceptible to external forces while the accelerometer does not drift. The filter looks as follows:

$$\begin{aligned}
O(t) &= 0.999(O(t - \tau) + R W(t) \tau + 0.001 \gamma) \\
R &= \begin{bmatrix} c_\psi c_\theta & -s_\psi c_\phi + c_\psi s_\theta s_\phi & s_\psi s_\phi + c_\psi s_\theta c_\phi \\ s_\psi c_\theta & c_\psi c_\phi + s_\psi s_\theta s_\phi & c_\psi s_\phi + s_\psi s_\theta c_\phi \\ -s_\theta & c_\theta s_\phi & c_\theta c_\phi \end{bmatrix} \\
\gamma &= [\text{asin}(\ddot{x}_B/g) \quad \text{atan}(\ddot{y}_B/\ddot{z}_B) \quad 0]^T
\end{aligned} \tag{4.31}$$

Where R is the rotation matrix. The gyroscope data is integrated every time step with the current angle value. Then, it is combined with the low-pass data from the accelerometer. The constants (0.999 and 0.001) have to add up to 1 but can be changed to tune the filter properly. The translational acceleration in the inertial frame $\ddot{T} = [\ddot{x} \quad \ddot{y} \quad \ddot{z}]^T$ is found by the relation (Mathworks, 2018; Zipfel, 2007):

$$\ddot{T} = R \ddot{T}_B \tag{4.32}$$

The velocity and the position are calculated by the following formulas (Mathworks, 2018; Zipfel, 2007) :

$$\begin{aligned}
\dot{X} &= R \left[\int \frac{F}{m} + V_B \times W \right] \\
X &= \frac{k_1 \tau}{z - 1} \dot{X}
\end{aligned} \tag{4.33}$$

Where V_B is the velocity with respect to the body frame, k_1 is a constant and its value is 0.01 and $X = [x \quad y \quad z]^T$ is the position vector. If the quadrotor is required to follow the trajectory in a cyclic process, the error may increase significantly due to the increase of the bias in the position information. The TLUC fails to overcome this error because it is intrinsically designed based on the position information.

4.7.2 Experimental results

Experiments are made to evaluate the performance of the proposed TLUC-ERSM system. The trajectory is chosen to be as in the simulation part. Figure 4.11 shows the 3D task space tracking of the desired trajectory. Tracking in the three axes are displayed in Figure 4.12. Figures 4.11 and 4.12 show good tracking during the whole operation time.

Orientation angles responses are displayed in Figure 4.13. Error signals in Figure 4.14 show a small value. The velocity is shown in Figure 4.15. In Figure 4.16, we can see the control torque inputs. The proposed controller ensures good tracking of the space desired trajectory with accuracy due to the TLUC estimation of uncertain dynamics which is shown in Figure 4.17.

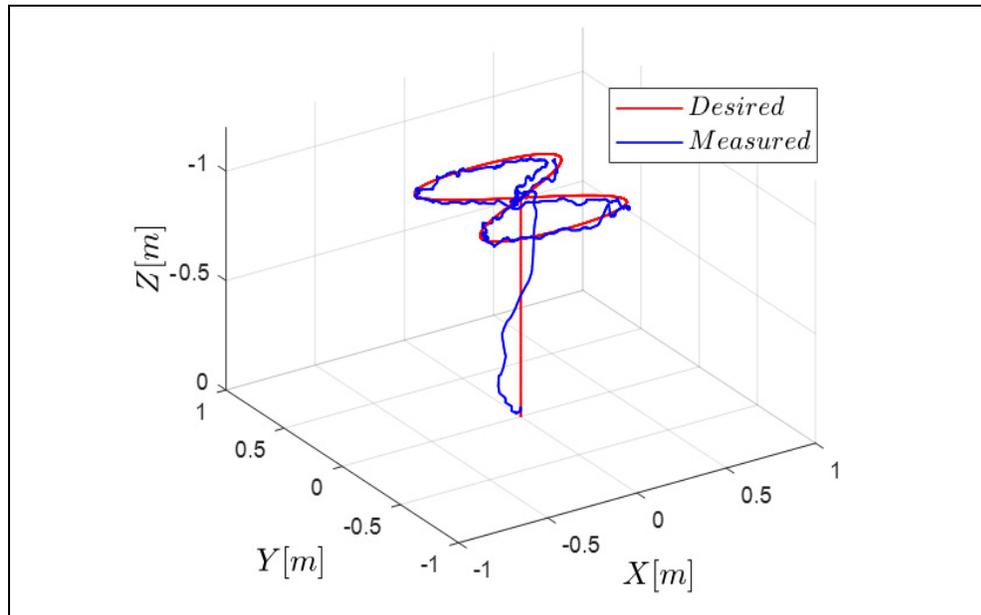


Figure 4.11 Experimental trajectory

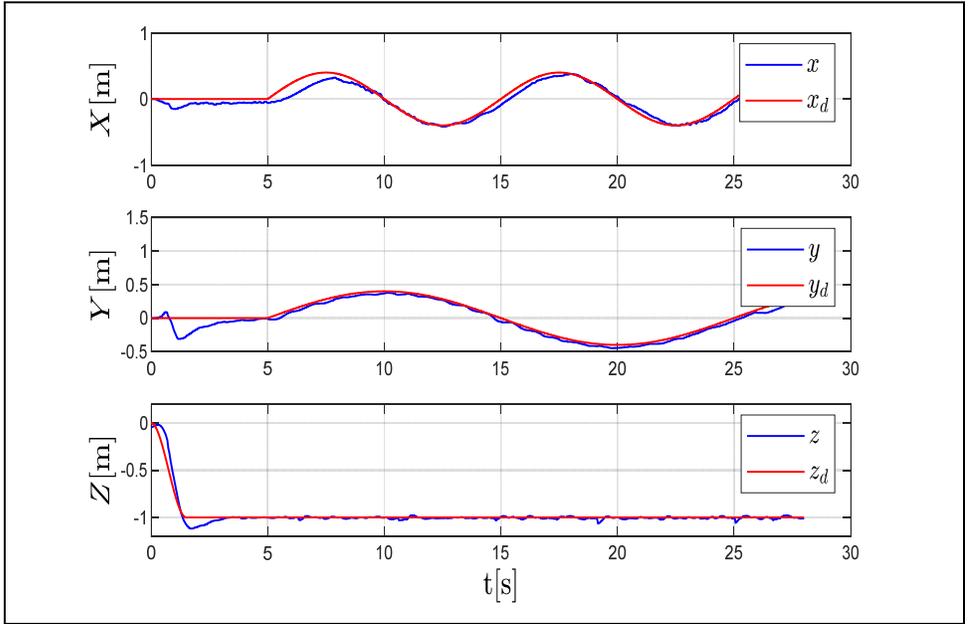


Figure 4.12 Position and altitude trajectory

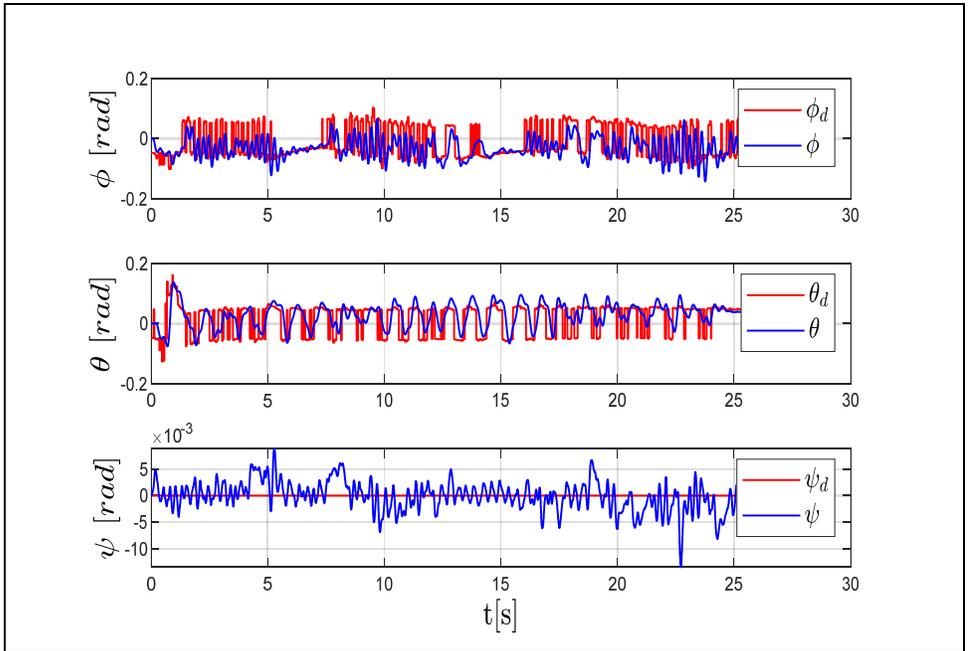


Figure 4.13 Euler angles response

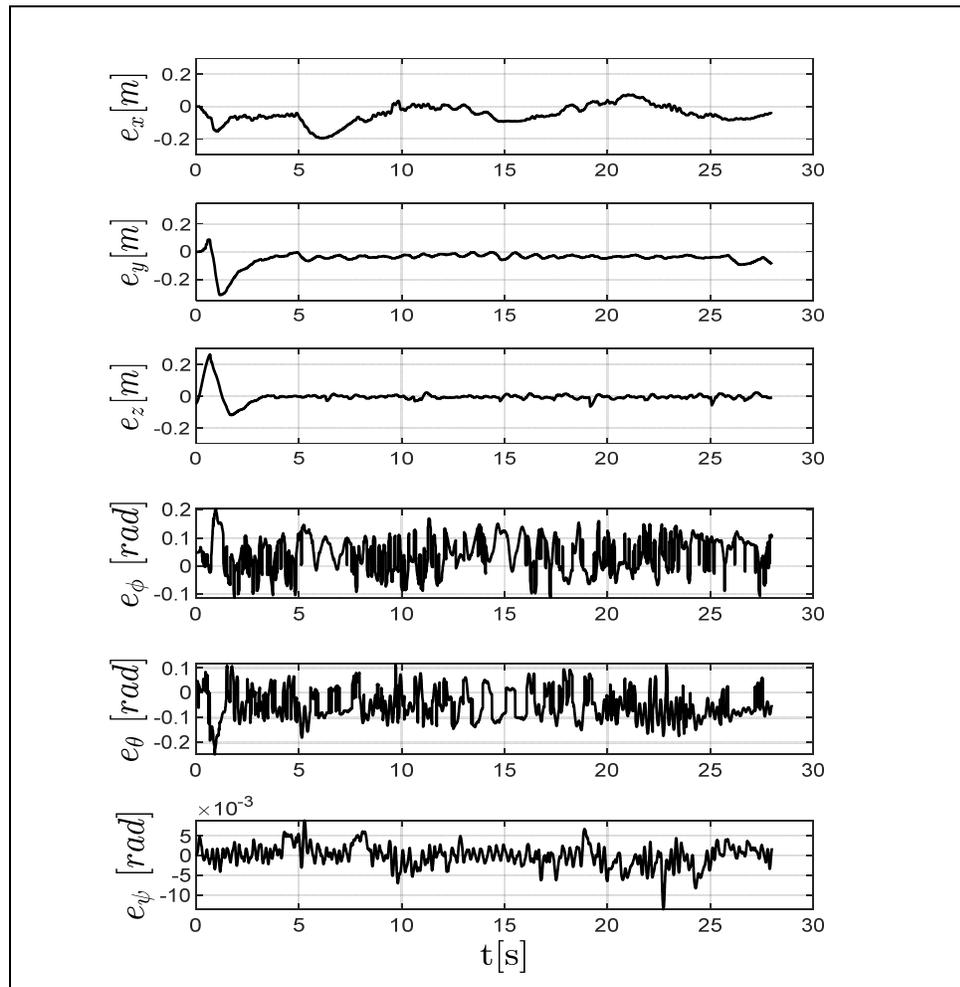


Figure 4.14 Errors in position, altitude and orientation

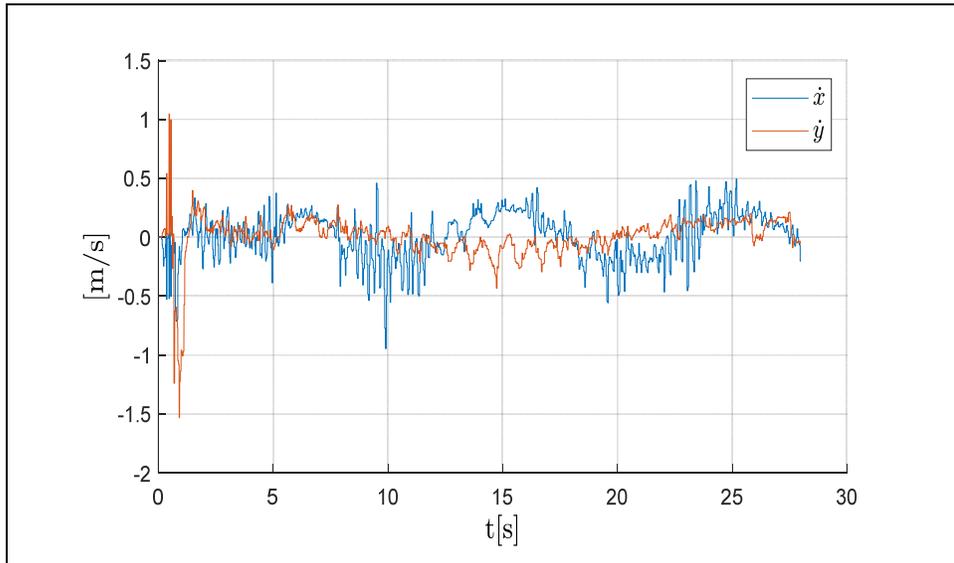
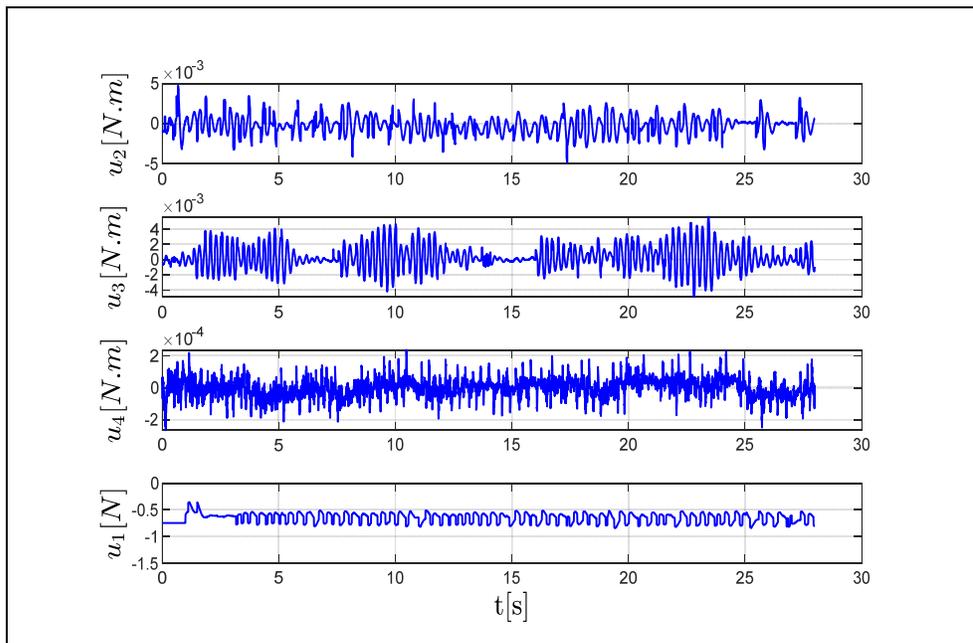
Figure 4.15 Velocities of x and y 

Figure 4.16 Control signals

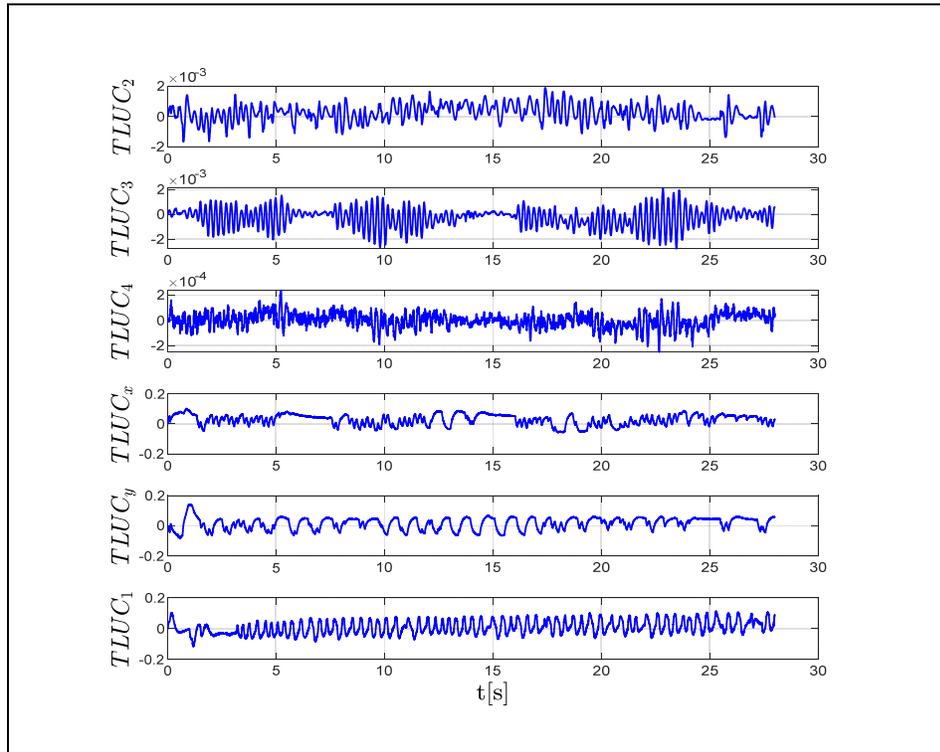


Figure 4.17 TLUC signals

4.8 Conclusion

We addressed the problem of uncertainties in the quadrotor by a proposed system designed to track perturbation in three-loop approach. The adaptive and the integral features of the TLUC give the ability to provide estimation and compensation of disturbance and uncertainties in real time. The ERSM ensures full control of the position, attitude and altitude and also guarantees low chattering and fast response. As a result, the closed-loop system can be driven to asymptotic stability. The performance of the complete system is analyzed by Lyapunov function, simulations, and experiments. The results show high performance of the proposed system in minimizing the effects of uncertainties and disturbances.

CHAPTER 5

VISION BASED LEADER FOLLOWER APPROACH FOR UNCERTAIN QUADROTOR DYNAMICS USING FEEDBACK LINEARIZATION SLIDING MODE CONTROL (FLSMC)

Walid Alqaisi¹, Brahim Brahmi¹, Jawhar Ghommam², Maarouf Saad¹ and Vahé Nerguizian¹

¹ Department of Electrical Engineering, École de technologie supérieure,
1100 Notre-Dame West, Montreal, Quebec, Canada H3C 1K3

² Department of Electrical Engineering, Sultan Quaboos University,
Al Khoudh, Muscat 123, Oman

Paper published in the International Journal of Modelling, Identification and Control, April,
2019

Abstract:

In this paper, a leader-follower quadrotor based on a visual system is presented. It is assumed that the follower quadrotor is equipped with a single onboard camera for determining the position of the leader. In the following quadrotor, feedback linearization based on sliding mode control FLSMC is designed. The latter reduces complex nonlinear control solutions and highly coupled dynamic behavior of the quadrotor. Uncertain dynamics and unexpected disturbances such as the change of payload, wind variation is overcome by designing time delay estimation TDE which help in reducing chattering. The proposed controller uses a second order sliding mode exact differentiator SOED to estimate the leader velocity and acceleration. The effectiveness of the proposed system is analyzed by Lyapunov function and studied by Matlab simulation.

Keywords: Leader-Follower, Feedback Linearization Sliding Mode Control, Quadrotor, second order sliding mode estimator, Time Delay Estimation.

5.1 Introduction

Tracking operations in robotics technology involve many possible applications like surveillance monitoring, load transportation and robot formation. A lot of research is carried out in this field. Following a leader robot in many cases is based on communication and position information sent by the leader based on an onboard GPS (Loria, Dardemir, & Jarquin, 2016; Mercado, Castro, & Lozano, 2013) or communication topology is utilized (Ranjbar, Ghasemi, & Akramizadeh, 2018). The major drawback with using GPS or a communication topology is its low accuracy and high noise. The signal could also be frequently lost. The follower robot might already know the trajectory of the leader (Li & Xiao, 2005). This is not possible in most robot following problems. In some cases, sensor-based measurements are used (Ali Dehghani & Bagher Menhaj, 2016; Dehghani & Menhaj, 2016). In such cases higher battery consumption, sensor accuracy, sensor noise and sensor cost become the real problem. Vision detection systems are used by many researchers (Sequeira, 2007; Jian Wang, Liu, & Yi, 2015). A single light-weight, low cost, pinhole camera is assumed to give the position in this paper. On the other hand, the quadrotor has a highly coupled dynamic structure. An accurate modeling of this type of robots cannot be obtained in a straight forward method. One of the most popular techniques used to resolve the problem of the nonlinear decoupling is feedback linearization FL (Slotine & Li, 1991). FL in general is aimed to transform algebraically nonlinear systems into an equivalent linear one in closed loop in order to avoid complex nonlinear control solutions and to reduce the effect of highly coupled dynamics. This technique is employed to address some practical control problems. In spite of that, the hard nonlinear parameters and/or uncertainties of the system do not permit conventional linear controls to provide a high level of accuracy (Slotine & Li, 1991). Actually, control of hard nonlinearities and uncertainties in nonlinear dynamics is an interesting topic of nonlinear control engineering. Numerous nonlinear control systems have been designed to overcome the effect of the nonlinearities and nonlinear uncertainties. A manipulator system to simplify the control law to become linear to decouple joint is designed (C. Fallaha & Saad, 2018), H_∞ control system is built (Xiangjian et al., 2016) and a robust nonlinear H_∞ controller takes into account the uncertainties in a quadrotor (Jasim & Gu, 2018). On the other hand, sliding mode

control (Kurode & Dixit, 2013; J. Mu et al., 2017; Slotine & Li, 1991; Xia et al., 2010; Youcef-Toumi & Ito, 1988), which is one of the most attractive control techniques, suffers from hard nonlinearities, unmodelled dynamics and external disturbance could reduce the performance. In (B. Mu, Zhang, & Shi, 2017) disturbance and uncertainties are reduced by an integral sliding mode flight controller incorporating reference angular signals and desired position information. Although it avoids output sensor noise and sensor accuracy problems, it does not incorporate the real states with control. In (Antonelli et al., 2017; Yoshimura, 2008), the robustness of the adaptive control is implemented to reduce external disturbance and uncertainty. In (Jia et al., 2017) an integral backstepping combined with sliding mode control is built to provide robustness to external disturbances. Nonlinear control systems, such as sliding mode control and backstepping are robust enough to reduce the effect of disturbances. However, they lack estimation and compensation of disturbances, mainly when disturbances are high. In our proposed system, an auxiliary approach is used to support the control by estimating and compensating all disturbances in order to provide higher rejection to disturbances and uncertainty. Furthermore, disturbance can be eliminated regardless of the control system used. In another approach (Yang, Cheng, Xia, & Yuan, 2017), a disturbance observer is used in a quadrotor system with a linear PD controller used in the outer loop, similar to (Zhou, Deng, Shi, & Zhong, 2017) where a cascade PID with a compensator is utilized. Anyway, a linear control applied to highly nonlinear system like the quadrotor does not guarantee robustness in all flying conditions. In (Jun Wang, Xin, & Zhang, 2017) a fuzzy logic controller FLC is designed to study the behavior of quadrotor subject to external disturbances. In spite of the advantages of FLC, it is still not robust to large disturbances variation.

Furthermore, in practical applications of sliding mode control, engineers may experience the undesirable phenomenon of oscillations having finite frequency and amplitude, which is known as ‘chattering’. Chattering is a harmful phenomenon because it leads to low control accuracy, high wear of moving mechanical parts, and high heat losses in power circuits (L. Fridman, 1999). Many approaches combine different techniques with the sliding mode to reduce the undesired effect of chattering (P. Chen, Chen, & Chiang, 2009; Slotine & Li, 1991; L. Wang, Chai, & Zhai, 2009). There is still the need to have a control system that reduces

chattering and high coupling of some robotic systems like the quadrotor with high performance and accuracy. The presence of disturbance and sensor noise makes the problem worse. These facts urge the need to build a system that has the ability to overcome the mentioned problems.

Motivated by different research work, a new adaptive sliding mode controller based on feedback linearization FL incorporating with time delay estimation TDE is implemented. It is important to mention here that the FL is used as a nonlinear design methodology.

The equivalent linear quadrotor model cannot perform efficiently in all operation points due to the nonlinear behavior of the quadrotor. The general form of the feedback linearization is utilized without applying an equivalent linear quadrotor model. This controller is achieved in two loops, inner and outer loops. The inner loop reduces the effect of the hard nonlinearity of the quadrotor parameters. The outer loop contains the robust term of the sliding mode controller and provides an estimation of disturbances using TDE in order to take into account the nonlinear uncertain disturbance problem (Youcef-Toumi & Ito, 1988). The latter is used widely in some research work and it provides decent results (Jin, Lee, & Ahn, 2015; Kim, Joe, Yu, Lee, & Kim, 2016; Singh, Goyal, Deolia, & Sharma, 2017). This approach is not affected by the size of robot parameters. It uses only delayed information of control input of the system and its delayed response states in order to provide an accurate estimation of disturbances. The visual system provides the leader position. However, the control design also needs velocity and acceleration to provide a good tracking. Practically, velocity and acceleration are not available. To overcome this dilemma, the second order sliding mode estimator SOED is applied (Levant, 2003) to provide an estimation of the velocity and acceleration of the leader. Its capability to reduce the noise of position measurement provides good tracking results. The stability of the quadrotor system and its finite time convergence of the errors are proved based on Lyapunov function.

The contribution of this paper can be summarized as follows:

- Design a control system capable of dealing with nonlinearity without linearizing the model. It just makes use of the general structure of the feedback linearization, and

based on adaptive sliding mode control. The system reduces the effect of the hard nonlinearity and the highly coupled dynamics and to provide a robust and an accurate control.

- Provide accurate and simultaneous estimation and compensation of external and internal disturbances in real time at the speed of one time-step by incorporating the proposed control with Time Delay Estimation (TDE).
- Afford accurate, continuous, bounded and smooth estimation of velocity and acceleration of the leader to provide a reference trajectory to the follower, by applying Second Order Sliding Mode Exact Differentiation estimator (SOED), which is also able to reduce noise and chattering phenomenon.

The rest of the paper is organized as follows: section (5-2) provides the quadrotor dynamic system. Problem definition is given in section (5-3). The visual estimation is described in section (5-4). The control is developed in section (5-5). The simulation is shown in section (5-6). Finally, the conclusion is given in section (5-7).

5.2 Quadrotor Dynamics

The quadrotor has four propellers in cross configuration as in Figure 5-1. Changing the velocity of each pair of motors causes the quadrotor to tilt and move to all possible directions. In this modeling, the quadrotor is assumed to be a rigid and symmetric. The development of the mathematical model is based on Newton-Euler formulation (Amin, Aijun, & Shamshirband, 2016; Samir Bouabdallah, 2007a; Bouadi, Bouchoucha, & Tadjine, 2007a; Bresciani, 2008). The dynamic equations are written in the following form:

$$\begin{aligned}
\ddot{\phi} &= \dot{\theta}\dot{\psi} \frac{i_{yy} - i_{zz}}{i_{xx}} - \dot{\theta} w_r \frac{j_r}{i_{xx}} - \frac{k_{fax}}{i_{xx}} \dot{\phi}^2 + \frac{1}{i_{xx}} u_2 \\
\ddot{\theta} &= \dot{\phi}\dot{\psi} \frac{i_{zz} - i_{xx}}{i_{yy}} + \dot{\phi} w_r \frac{j_r}{i_{yy}} - \frac{k_{fay}}{i_{yy}} \dot{\theta}^2 + \frac{1}{i_{yy}} u_3 \\
\ddot{\psi} &= \dot{\theta}\dot{\phi} \frac{i_{xx} - i_{yy}}{i_{zz}} - \frac{k_{faz}}{i_{zz}} \dot{\psi}^2 + \frac{1}{i_{zz}} u_4 \\
\ddot{x} &= -\frac{k_{ftx}}{m} \dot{x} + u_x \frac{u_1}{m} \\
\ddot{y} &= -\frac{k_{f ty}}{m} \dot{y} + u_y \frac{u_1}{m} \\
\ddot{z} &= -\frac{k_{ftz}}{m} \dot{z} - g + (\cos\phi \cos\theta) \frac{u_1}{m}
\end{aligned} \tag{5.1}$$

Knowing that, ϕ , θ and ψ are the roll, pitch and yaw angles respectively. i_{xx} , i_{yy} , i_{zz} are the moments of inertia about body frames x , y and z axes respectively. j_r is the rotor inertia. m is the total mass. g is the gravity force. k_{ftx} , $k_{f ty}$, k_{ftz} are the drag coefficients of translation and k_{fax} , k_{fay} , k_{faz} are the coefficients of the aerodynamic friction. u_1 , u_2 , u_3 and u_4 are the control inputs for altitude, roll, pitch and yaw:

$$\begin{aligned}
u_1 &= b(w_1^2 + w_2^2 + w_3^2 + w_4^2) \\
u_2 &= b(w_4^2 - w_2^2)l_e \\
u_3 &= b(w_3^2 - w_1^2)l_e \\
u_4 &= d_r(-w_1^2 + w_2^2 - w_3^2 + w_4^2)
\end{aligned} \tag{5.2}$$

Where, d_r is the drag coefficient, b is the thrust coefficient. l_e is the length of the moment arm. w_i ($i = 1,2,3,4$) is the angular velocity of the quadrotor motors and $w_r = -w_1 + w_2 - w_3 + w_4$. The auxiliary inputs are:

$$\begin{aligned}
u_x &= (\cos\phi \sin\theta \cos\psi + \sin\phi \sin\psi) \\
u_y &= (\cos\phi \sin\theta \sin\psi - \sin\phi \cos\psi)
\end{aligned} \tag{5.3}$$

Remark 1: In this paper, matrices are referred as bold capital letters, to vectors as capital letters and small letters to scalars.

The nonlinear quadrotor system can be described by the so-called companion form, or controllability canonical form (Slotine & Li, 1991):

$$\ddot{X} = F(X) + \mathbf{G}(X)U + D(t) \quad (5.4)$$

Where, $D(t)$ is the bounded uncertainty, the state vector is given as, $X = [\phi, \theta, \psi, x, y, z]^T$, $F(X)$ and $\mathbf{G}(X)$ are known nonlinear functions, U is the input vector, they are given as:

$$F(X) = \begin{bmatrix} (\dot{\theta}\dot{\psi} \frac{(i_{yy} - i_{zz})}{i_{xx}} - \dot{\theta} \frac{w_r j_r}{i_{xx}} - \frac{k_{fax}}{i_{xx}} \dot{\phi}^2) \\ (\dot{\phi}\dot{\psi} \frac{(i_{zz} - i_{xx})}{i_{yy}} + \dot{\phi} w_r \frac{j_r}{i_{yy}} - \frac{k_{fay}}{i_{yy}} \dot{\theta}^2) \\ (\dot{\theta}\dot{\phi} \frac{(i_{xx} - i_{yy})}{i_{zz}} - \frac{k_{faz}}{i_{zz}} \dot{\psi}^2) \\ -\frac{k_{ftx}}{m} \dot{x} \\ -\frac{k_{fity}}{m} \dot{y} \\ -\frac{k_{ftz}}{m} \dot{z} - g \end{bmatrix}, \quad U = \begin{bmatrix} u_2 \\ u_3 \\ u_4 \\ u_1 u_x \\ u_1 u_y \\ u_1 \end{bmatrix}$$

$$\mathbf{G}(X) = \begin{bmatrix} 1/i_{xx} & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/i_{yy} & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/i_{zz} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/m & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/m & 0 \\ 0 & 0 & 0 & 0 & 0 & (\cos\phi\cos\theta)/m \end{bmatrix}$$

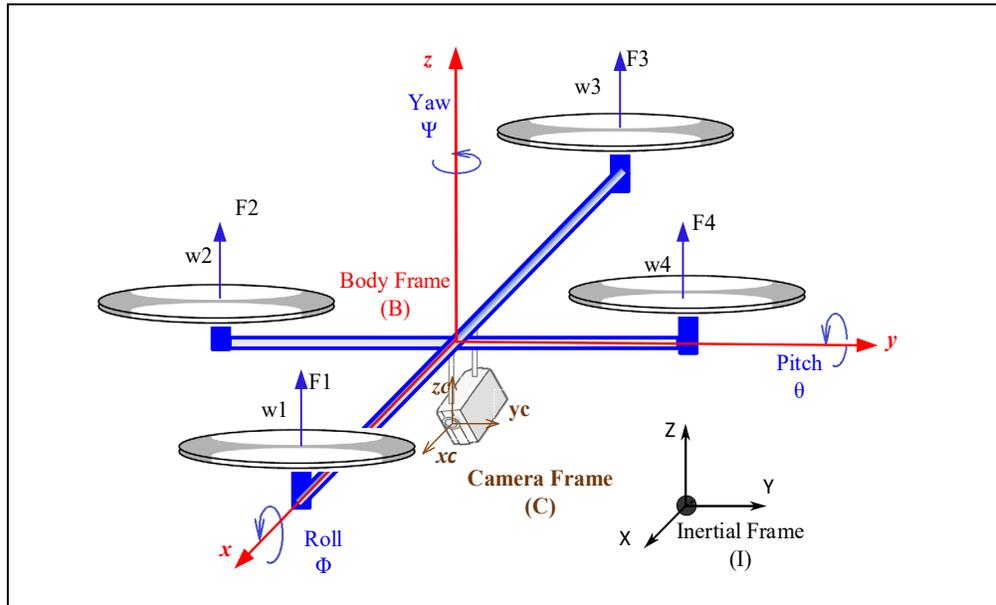


Figure 5.1 Quadrotor configuration, forces, inertial & body frames

5.3 Problem Definition

The follower-leader mission between robots is performed in this paper such that a follower quadrotor tracks a leader quadrotor and keeps a certain distance. The position of the leader is provided by a visual system using a single camera as in (Kaminer, Ghabchelo, Dobrokhodov, & Jones, 2011). However, the designed controller needs velocity and acceleration to provide a good tracking and to avoid the collision between the quadrotors. On the other hand, the quadrotor as a robotic system has a highly coupled dynamic structure. The hard nonlinearities and uncertainties of the system prevent conventional control system from providing high accuracy. The problems of nonlinear uncertainty, imperfection of modeling, disturbance and sensor noise reduce the performance also increase chattering. The general objective is to build a control system that is able to handle such difficulties with high accuracy. The mathematical problem boils down into designing a control input U such that the following formula is satisfied in the follower quadrotor:

$$\lim_{t \rightarrow \infty} \|X - X_d\| = 0 \quad (5.5)$$

Where X is the state vector of the follower quadrotor, X_d is the desired trajectory vector that combined desired rotation R_d and desired position P_d :

$$\begin{aligned} X_d &= [R_d, P_d]^T \\ R_d &= [\phi_d, \theta_d, \psi_d]^T \\ P_d &= [x_d, y_d, z_d]^T \end{aligned} \quad (5.6)$$

The desired trajectory X_d is obtained as follows:

- 1) The desired pitch and roll angles are given as (Bouadi, Bouchoucha, & Tadjine, 2007b):

$$\begin{aligned} \phi_d &= \arcsin \left\{ \frac{[-(\ddot{x} - \frac{k_{ftx}}{m} \dot{x}) \sin \psi + (\ddot{y} - \frac{k_{fty}}{m} \dot{y}) \cos \psi]}{\sqrt{(\ddot{x} - \frac{k_{ftx}}{m} \dot{x})^2 + (\ddot{y} - \frac{k_{fty}}{m} \dot{y})^2 + (\ddot{z} + g - \frac{k_{ftz}}{m} \dot{z})^2}} \right\} \\ \theta_d &= \arctan \left\{ \frac{[(\ddot{x} - \frac{k_{ftx}}{m} \dot{x}) \cos \psi + (\ddot{y} - \frac{k_{fty}}{m} \dot{y}) \sin \psi]}{\ddot{z} + g - \frac{k_{ftz}}{m} \dot{z}} \right\} \end{aligned}$$

- 2) The desired yaw angle will not affect the position of the quadrotor, therefore it could be set to zero.
- 3) The desired position of the follower is $P_d = [x_d, y_d, z_d]^T = P_L^I - [\mu_x, \mu_y, \mu_z]^T$, where $P_L^I = [x_L, y_L, z_L]^T$ is the position of the leader with respect to the inertial frame as demonstrated in section 5.6. μ_x, μ_y and μ_z are the desired distances between the leader and the follower in x, y and z directions respectively.

5.4 Leader Position Visual Estimation

In this paper, it is considered to have a quadrotor equipped with a single pinhole camera pointing at the moving leader as in Figure 5.2. The reason behind using a single light-weight, low cost, pinhole camera rather than other types of measurements is that sensor-based measurements suffer from sensor inaccuracy, sensor noise and sensor cost in addition to higher battery consumption. Furthermore, GPS or other communication signals could also be frequently lost. Meanwhile, vision systems avoid such drawbacks.

Let I, B, C denote the inertial reference frame, the quadrotor body fixed frame and the camera frame respectively. The coordinate of the leader with respect to the camera frame is $P_L^C = [x_c, y_c, z_c]^T$. R_C^B is the coordinate transformation matrix from frame C to frame B , R_B^I is the coordinate transformation matrix from frame B to frame I . R_C^I is the coordinate transformation matrix from frame C to frame I then $R_C^I = R_B^I \cdot R_C^B$. The transformation R_C^B is computed onboard of the UAV noting that camera frame is shifted from the body frame. R_B^I is calculated using attitude angles given by UAV. The position of the leader can be found as (Kaminer et al., 2011):

$$\begin{aligned} P_L^I &= R_B^I \cdot R_C^B \cdot P_L^C \\ P_L^I &= R_C^I \cdot P_L^C \end{aligned} \quad (5.7)$$

$P_L^I = [x_L, y_L, z_L]^T$ is the position of the leader with respect to the inertial frame I as in Figure 5.2. The visual system provides the control with the position of the leader where the velocity and acceleration are estimated as described in section 5.5.

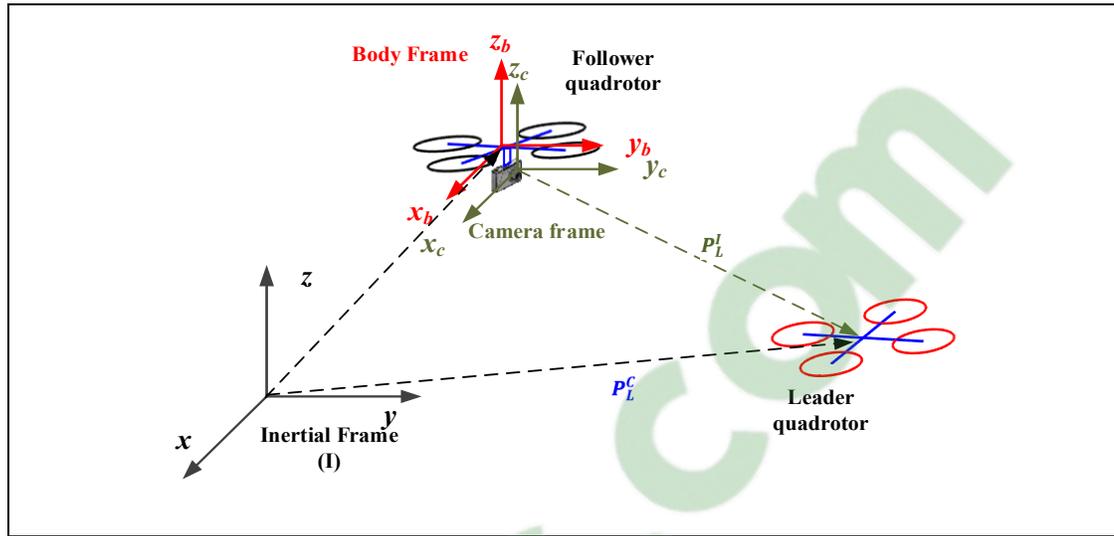


Figure 5.2 Different frames definition and leader relative position

5.5 Control Design

The controller synthesis procedure aims to ensure stability and makes the quadrotor follow a requested trajectory while keeping the roll and pitch angles bounded and small enough to be near the linearization trajectory. The following assumptions are needed in the control design:

Assumption 1: The visual system provides the control with the position of the leader where the leader remains in the field of view of the camera.

Assumption 2: In the follower quadrotor, the position and its derivative are measured.

Assumption 3: The term of uncertainties $D(t)$ is globally Lipschitz function.

Assumption 4: Matrix $\mathbf{G}(x)$ is nonsingular and invertible.

Control Algorithm:

In this section, a robust sliding mode controller incorporated with time delay estimation TDE approach is implemented. The purpose is to have asymptotic convergence of the error in the presence of nonlinear uncertainty and external disturbance. The linearization procedure is based on input/output feedback linearization approach. This latter is achieved by two loops,

inner and outer. The inner loop is designed to reduce the effect of the hard nonlinearity of the dynamic system, to create input/output state relation and build a nonlinear control law. The outer loop aimed to control the input/output system, to realize the stabilization of closed-loop system and to provide an estimation of nonlinear uncertainties. The control system block diagram is described in Figure 5.3. The objective of the input-output system is to obtain a direct relationship between the output system and the input control action of the system. Based on (5.4) the desired input U can be written as:

$$U = \mathbf{G}(X)^{-1}(V_b - F(X)) \quad (5.8)$$

Where V_b is an auxiliary control input to the system and $V_b = [v_\phi, v_\theta, v_\psi, v_x, v_y, v_z]^T$. The inverse of the matrix $\mathbf{G}(x)$ exists according to Assumption 4. According to (5.8) there is an explicit relation between the control input and the output of the system where the system can be rewritten such that:

$$\ddot{X} = V_b + D(t) \quad (5.9)$$

The desired trajectory X_d is obtained as in (5.6). The error and its derivative are $E = X - X_d \in R^n$ and $\dot{E} = \dot{X} - \dot{X}_d \in R^n$. The sliding variable and its derivative are selected as:

$$\begin{aligned} S &= \dot{E} + \mathbf{C}E \\ \dot{S} &= \ddot{E} + \mathbf{C}\dot{E} \end{aligned} \quad (5.10)$$

Where, $\mathbf{C} = \text{diag}(c_{ii})$ for $i = 1, \dots, n$ is a diagonal definite positive matrix, n is the length of the state vector.

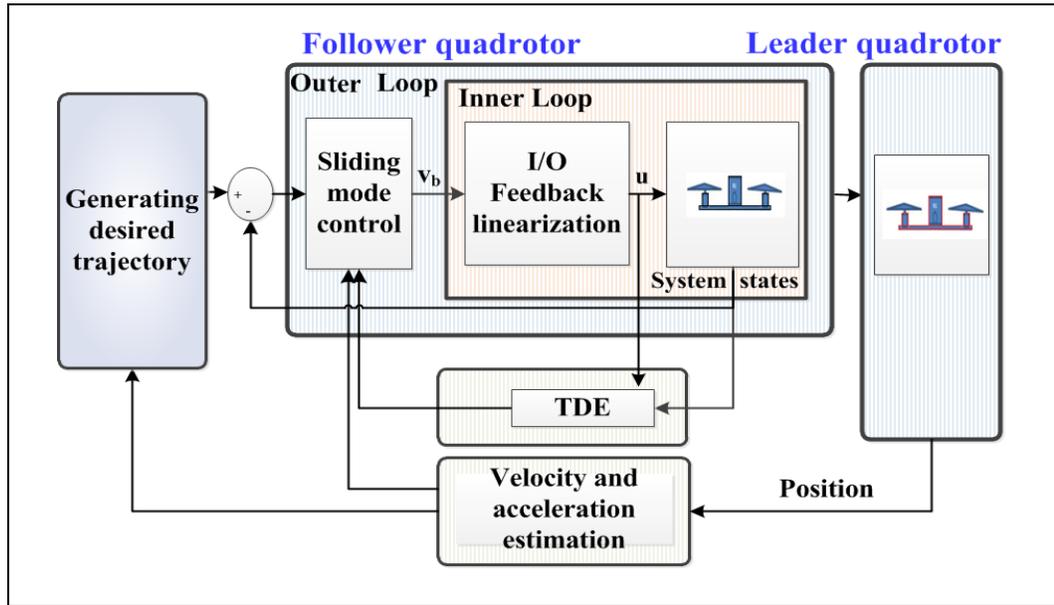


Figure 5.3 Control System block diagram

The designed controller needs velocity and acceleration of the desired trajectory \dot{X}_d, \ddot{X}_d to provide good trajectory tracking. To find robust real-time estimations in the absence of measurement noises, an estimator based on “Second Order sliding mode Exact Differentiation SOED” is built (Levant, 2003) as follows:

$$\begin{aligned}
 \dot{Y}_0 &= -\mathbf{B}_1 \mathbf{\Pi}_1 \text{Sign}(Y_0 - P_L^l) + Y_1 \\
 \dot{Y}_1 &= -\mathbf{B}_2 \mathbf{\Pi}_2 \text{Sign}(Y_1 - \dot{Y}_0) + Y_2 \\
 \dot{Y}_2 &= -\mathbf{B}_3 \text{Sign}(Y_2 - \dot{Y}_1)
 \end{aligned} \tag{5.11}$$

Where $\mathbf{B}_1, \mathbf{B}_2$ and $\mathbf{B}_3 \in R^{3 \times 3}$ are positive diagonal constants, the matrices $\mathbf{\Pi}_1$ and $\mathbf{\Pi}_2$ are given as:

$$\mathbf{\Pi}_1 = \begin{bmatrix} |y_{01} - x_d|^{\frac{2}{3}} & 0 & 0 \\ 0 & |y_{02} - y_d|^{\frac{2}{3}} & 0 \\ 0 & 0 & |y_{03} - z_d|^{\frac{2}{3}} \end{bmatrix}$$

$$\mathbf{\Pi}_2 = \begin{bmatrix} |y_{11} - \dot{y}_{01}|^{\frac{1}{2}} & 0 & 0 \\ 0 & |y_{12} - \dot{y}_{02}|^{\frac{1}{2}} & 0 \\ 0 & 0 & |y_{13} - \dot{y}_{03}|^{\frac{1}{2}} \end{bmatrix}$$

Where y_{01}, y_{02}, y_{03} represent the x, y, z position. The Second Order Exact Differentiation (SOED) provides:

$$\begin{aligned} Y_1 &= \dot{P}_L^I = \hat{P}_d \\ Y_2 &= \ddot{P}_L^I = \ddot{P}_d \end{aligned} \quad (5.12)$$

Then we have $\hat{X}_d = [\hat{R}_d, \hat{P}_d]^T$ and $\hat{X}_d = [\hat{R}_d, \hat{P}_d]^T$, where $(\hat{\bullet})$ is the estimated value of (\bullet) . The desired angular velocity and acceleration are found onboard. If the constants are correctly chosen, the equalities $\hat{X}_d = \dot{X}_d$ and $\hat{X}_d = \ddot{X}_d$ are true after a certain time of a transient process. The corresponding solutions of the dynamic systems are Lyapunov stable (Levant, 2003).

Because of velocity estimator, the derivative of the error becomes $\dot{E} = \dot{X} - \hat{X}_d$. The sliding variable and its derivative in (5.10) becomes:

$$\begin{aligned} \hat{S} &= \dot{E} + \mathbf{C}E \\ \dot{\hat{S}} &= \ddot{E} + \mathbf{C}\dot{E} \end{aligned} \quad (5.13)$$

The auxiliary input V_b is designed as described below (J. Liu & Wang, 2012b):

$$V_b = \hat{X}_d - \mathbf{C}\dot{E} - \mathbf{K} \text{Sign}(\hat{S}) - D(t) \quad (5.14)$$

Where, $\mathbf{K} = \text{diag}(k_{ii})$ for $i = 1, \dots, n$ is a diagonal positive-definite matrix. The function $\text{Sign}(s_i)$ is defined such that:

$$\text{Sign}(s_i) = \begin{cases} 1 & \text{for } s_i > 0 \\ 0 & \text{for } s_i = 0 \\ -1 & \text{for } s_i < 0 \end{cases} \quad (5.15)$$

Since $D(t)$ is uncertain, this may influence the performance of the quadrotor tracking. If Assumption 3 is verified, it is possible to use time delay estimation TDE to obtain the estimated disturbances by using (5.4) where it is possible to adopt one step delayed signals to satisfy the causality between input and output samples such that:

$$\widehat{D}(t) \approx D(t - \tau) = \ddot{X}(t - \tau) - F(X(t - \tau)) - \mathbf{G}(x(t - \tau)) U(t - \tau) \quad (5.16)$$

Where, τ is the smallest constant that can be achieved. In real time implementation, τ is the sampling time. $\widehat{D}(t)$ is the estimated values of $D(t)$. The external control input can be rewritten such that:

$$V_b^* = \ddot{X}_d - \mathbf{C}\dot{E} - \mathbf{K} \text{Sign}(\hat{S}) - \widehat{D}(t) \quad (5.17)$$

By using (5.17), the control system proposed in (5.8) becomes:

$$U = G(X)^{-1}(\ddot{X}_d - \mathbf{C}\dot{E} - \mathbf{K} \text{Sign}(\hat{S}) - \widehat{D}(t) - F(X)) \quad (5.18)$$

Theorem: For the quadrotor system described in (5.4), the combined system of the control input with the time delay estimation TDE in addition to the velocity estimator proposed in (5.18), (5.16) and (5.11) yields to finite time convergence of the sliding surface $S(x, t) = 0$. The tracking error E and \dot{E} will asymptotically converge to zero.

Proof: Let us select the following Lyapunov function:

$$V = \frac{1}{2} S^T S \quad (5.19)$$

The time derivative of Lyapunov function is given by:

$$\dot{V} = S^T \dot{S} \quad (5.20)$$

By substituting the derivative of the selected surface from (5.10), then:

$$\begin{aligned} \dot{V} &= S^T (\ddot{E} + \mathbf{C}\dot{E}) \\ &= S^T [\ddot{X} - \ddot{X}_d + \mathbf{C}\dot{E}] \end{aligned} \quad (5.21)$$

Substituting (5.4) in (5.21), gives:

$$\dot{V} = S^T [F(x) + \mathbf{G}(x) U + D(t) - \ddot{X}_d + \mathbf{C}\dot{E}] \quad (5.22)$$

By substituting the control input (5.18) in (5.22):

$$\begin{aligned} \dot{V} &= S^T [\ddot{\hat{X}}_d - \mathbf{C}\dot{\hat{E}} - \mathbf{K} \text{Sign}(\hat{S}) - \widehat{D}(t) + D(t) - \ddot{X}_d + \mathbf{C}\dot{E}] \\ &= S^T [-\mathbf{K} \text{Sign}(\hat{S}) - (\ddot{X}_d - \ddot{\hat{X}}_d) + \mathbf{C}(\dot{E} - \dot{\hat{E}}) + (D(t) - \widehat{D}(t))] \\ &\leq -|S^T| |\mathbf{K}| - |S^T| [|\ddot{X}_d - \ddot{\hat{X}}_d| + \mathbf{C} |\dot{E} - \dot{\hat{E}}| + |D(t) - \widehat{D}(t)|] \end{aligned} \quad (5.23)$$

We have $|\bullet|$ denotes the Euclidian norm. $\widetilde{D}(t) = D(t) - \widehat{D}(t)$ is the estimation error or TDE error. Assumption 4 implies:

$$\begin{aligned} |\widetilde{D}(t)| &= |D(t) - \widehat{D}(t)| \\ &= |D(t) - D(t - \tau)| \\ &\leq \delta |t - (t - \tau)| \\ &\leq \delta \tau \end{aligned} \quad (5.24)$$

Where $\delta > 0$ is Lipschitz constant that is known to have a small value. On the other hand, because of the convergence of the extended observer as proved in (Levant, 2003) it can be

found that $|\ddot{X}_d - \ddot{\hat{X}}_d| \rightarrow 0$ and $|\dot{E} - \dot{\hat{E}}| \rightarrow 0$. A proper selection of \mathbf{K} gives $\dot{V} \leq 0$, which proves the stability of the closed loop of the control system.

5.6 Simulation

The simulation results are obtained based on real quadrotor parameters as in Table 5-1.

Table 5.1 Quadrotor parameters

Parameter	Value	Parameter	Value
m	1.83 [kg]	c	1.140×10^{-7} [N.s ²]
i_{xx}	21.6×10^{-3} [kg.m ²]	l_e	1[m]
i_{yy}	21.6×10^{-3} [kg.m ²]	k_{ftx}	0.0320 [N.m.s ²]
i_{zz}	43.2×10^{-3} [kg.m ²]	k_{fity}	0.0320 [N.m.s ²]
d_r	0.3 [N.m.s ²]	k_{ftz}	0.0480 [N.m.s ²]
g	9.81 [m/s ²]	k_{ftax}	5.5670×10^{-4} [N.s/m]
j_r	3.357×10^{-5} [kg.m ²]	k_{ftay}	5.5670×10^{-4} [N.s/m]
b	2.98×10^{-5} [N.s ²]	k_{ftaz}	6.3540×10^{-4} [N.s/m]

The tuned gain values are $\mathbf{K} = \text{diag}[0.5, 0.5, 1.2, 1.5, 1.5, 1.2]$, $\mathbf{C} = \text{diag}[30, 30, 30, 30, 30, 30]$, $\mathbf{B}_1 = \text{diag}[160, 160, 100]$, $\mathbf{B}_2 = \text{diag}[520, 520, 200]$ and $\mathbf{B}_3 = \text{diag}[10^{-2}, 10^{-2}, 10^{-1}]$. The simulation is performed using sliding mode based on feedback linearization. The Leader quadrotor start position is at point (2, 0, 0) and the end point is (4, 0.3, 2). The follower position starts at point (-1, -0.5, 0). The desired distance between the quadrotors is $[\mu_x, \mu_y, \mu_z] = [2, 0, 0]$. The goal is to keep $[\mu_x, \mu_y, \mu_z]$ meters distance with the leader quadrotor as an in Figure 5.4.

Figure 5.4 shows the simulation of the follower quadrotor tracks the leader quadrotor in 3D space. The output trajectory in translational and rotational movement is shown in Figure 5.5. It can be noticed that in x-direction, the error converges at $t = 20$ while in y-direction it converges at $t = 10$, this is due to the initial position in x and y $(-1, -0.5, 0)$, as a result it takes a longer time to converge in x-direction. It can be seen that the control system is able to robustly stabilize the quadrotor and move it to the desired trajectory with the desired angles. Errors in translation and rotation are shown in Figure 5.6. The chattering in the errors are very small and it exists because of the *(Sign)* function in (5.14) which is known to have an aggressive nature if compared to *(Sat)* or *(Tanh)* functions (Slotine & Li, 1991). Velocity estimation of the leader quadrotor is shown in Figure 5.7. The estimated velocity is continuous and smooth as desired and could easily be applied to a real-time model.

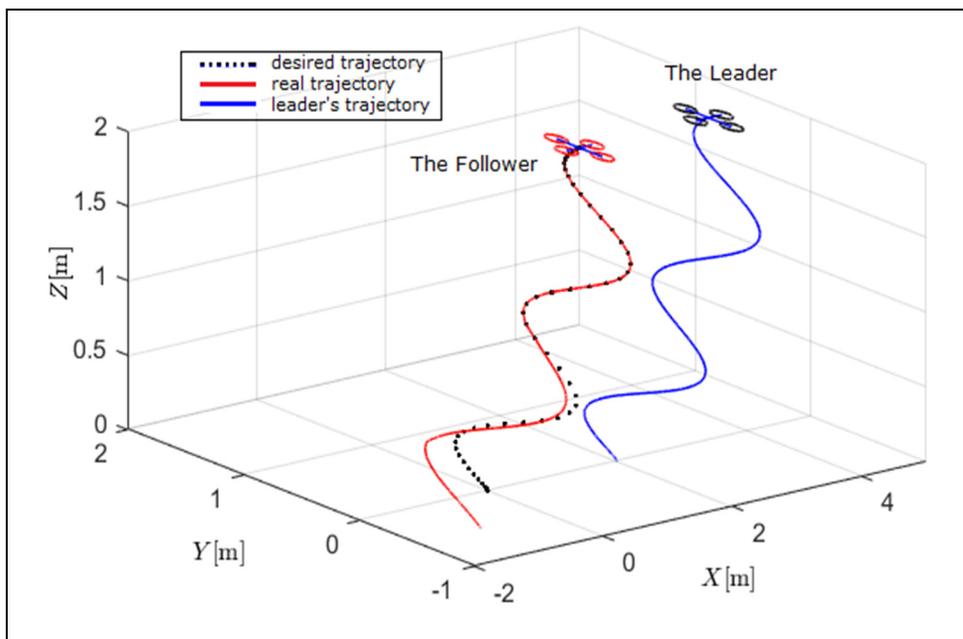


Figure 5.4 Quadrotor following the leader quadrotor

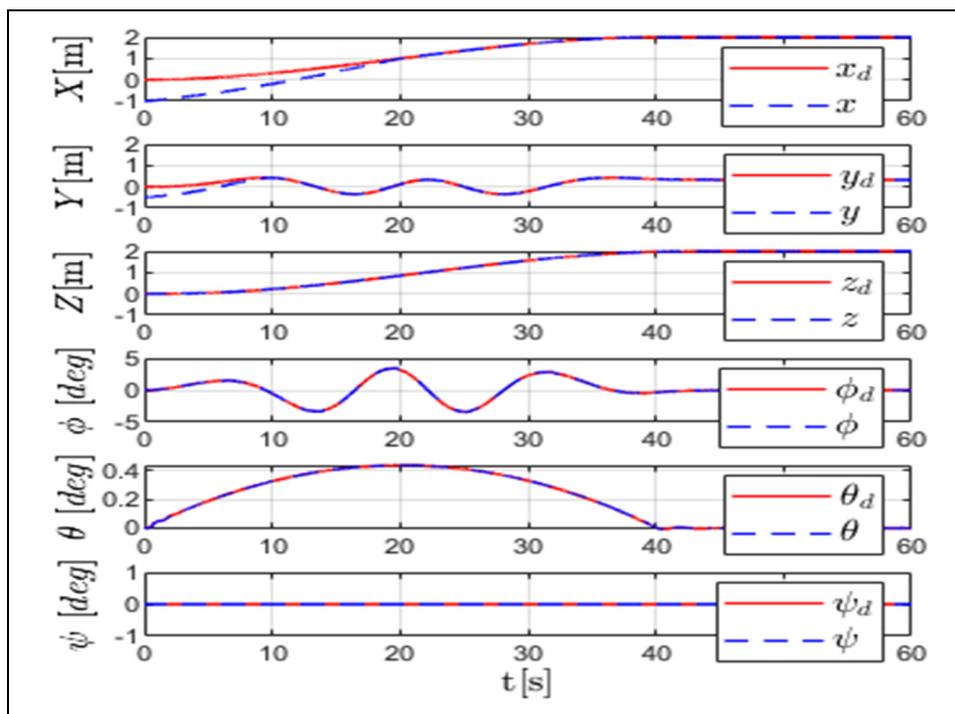


Figure 5.5 Translational trajectory

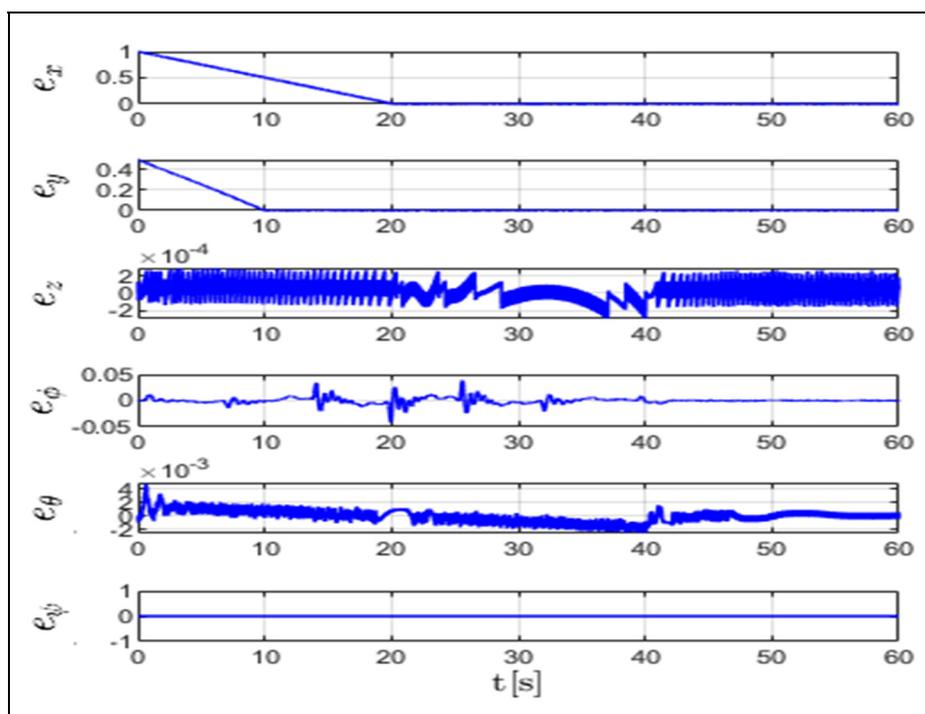


Figure 5.6 Translation and Rotational error

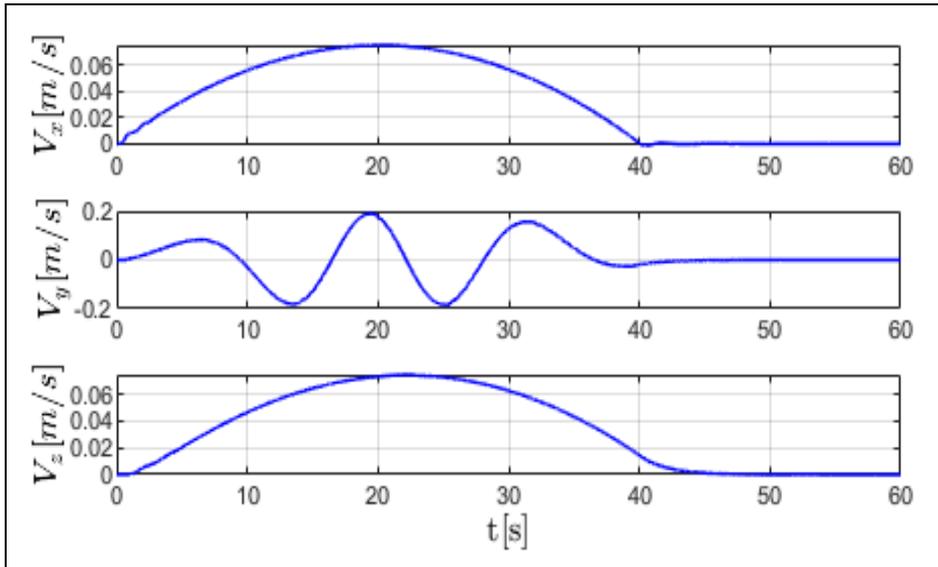


Figure 5.7 Velocity estimation

In order to verify the quality of the proposed scheme, a comparison is made in three different cases using the same parameters and the same trajectory. At first, a conventional sliding mode control is used (Hicham, 2012). After that, the FLSMC control with time delay estimation TDE is used. Then the FLSMC is used with TDE and SOED. The Root mean square (RMSE) value of the errors in each case is shown in Table (5.2).

Table 5.2 Root mean square of errors in three different cases

RMS	Sliding mode	FLSMC with TDE	FLSMC with TDE & SOED
x	0.3818	0.3739	3.36×10^{-1}
y	0.1377	0.1329	0.1195
z	3.00×10^{-3}	4.59×10^{-4}	1.00×10^{-4}

The numbers in the table show the advantage of FLSMC with the support of TDE and SOED over the traditional sliding mode. The proposed combined system provides good performance as shown in the figures and the table.

Figure 5.8 shows the sliding surface in the case of applying FLSMC with TDE and SOED. Figure 5.9 displays the sliding surface for the traditional sliding mode control (Hicham, 2012). If Figures 5.8 and 5.9 are compared, it can be seen that sliding surface is smaller in the proposed algorithm, which shows another advantage over the traditional sliding mode control.

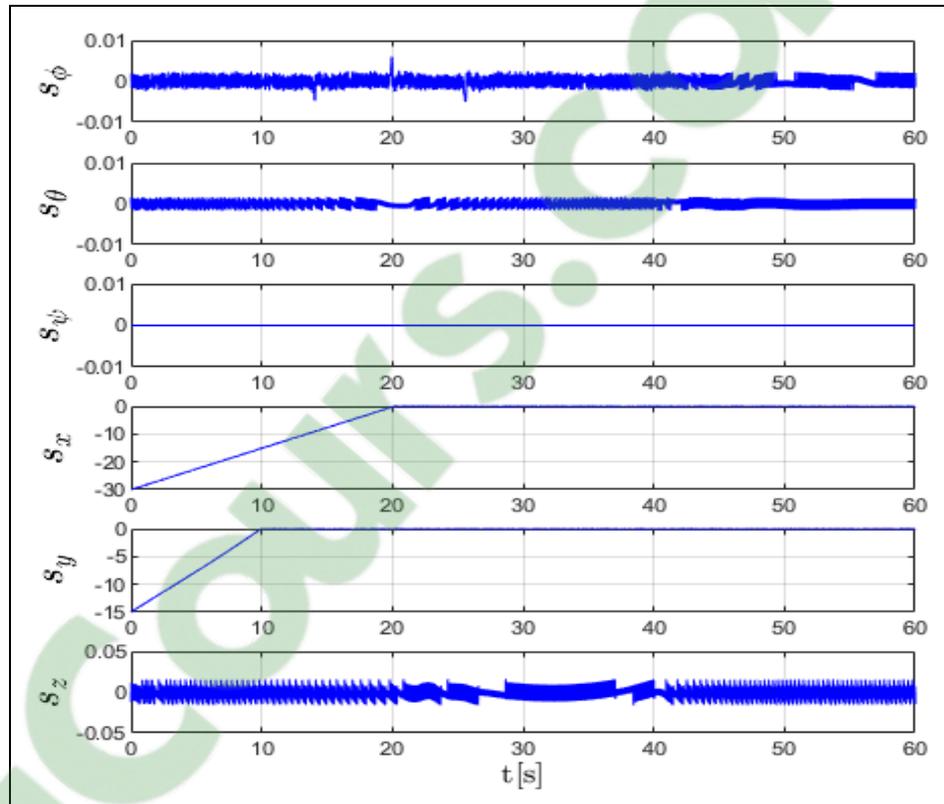


Figure 5.8 The sliding surface in FLSMC

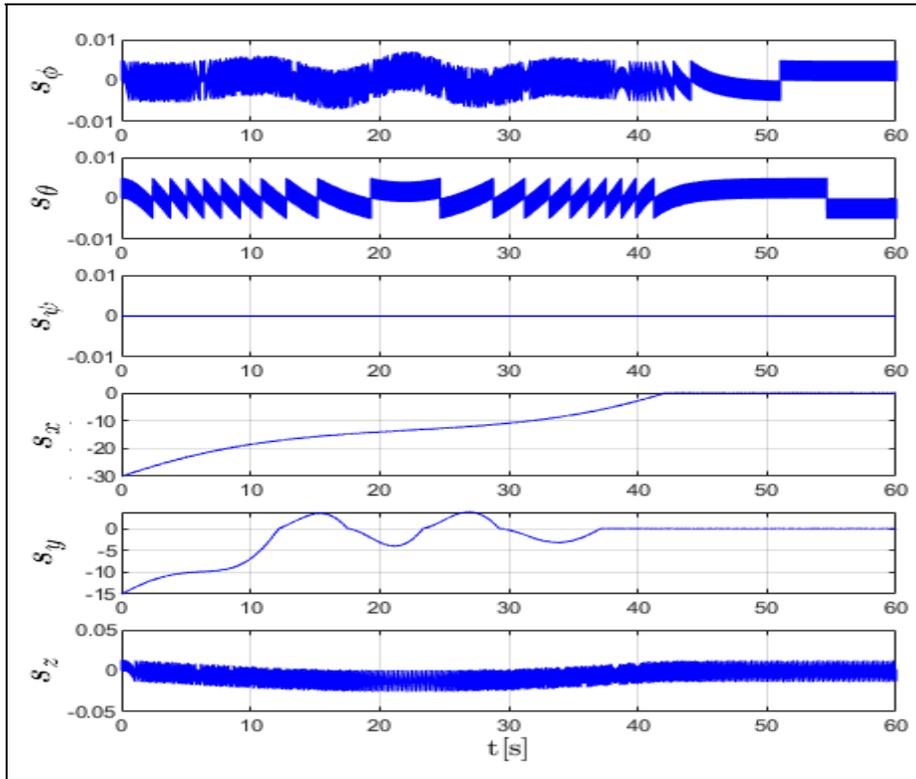


Figure 5.9 The sliding surface in SM control

In Figure 5.10, the control is shown for the FLSMC control with TDE and SOED. Figure 5.13 displays the same for the traditional sliding mode control (Hicham, 2012). The proposed system helped in reducing control signals value. On the other hand, motor torque commands are continuous as desired and could easily be applied to a real-life model. It can be noted that the command values are small and never reach saturation during the flight which is an indicator of the stability of the controller. In the simulation, (*Sign*) function is used to illustrate the chattering clearly for comparison purposes.

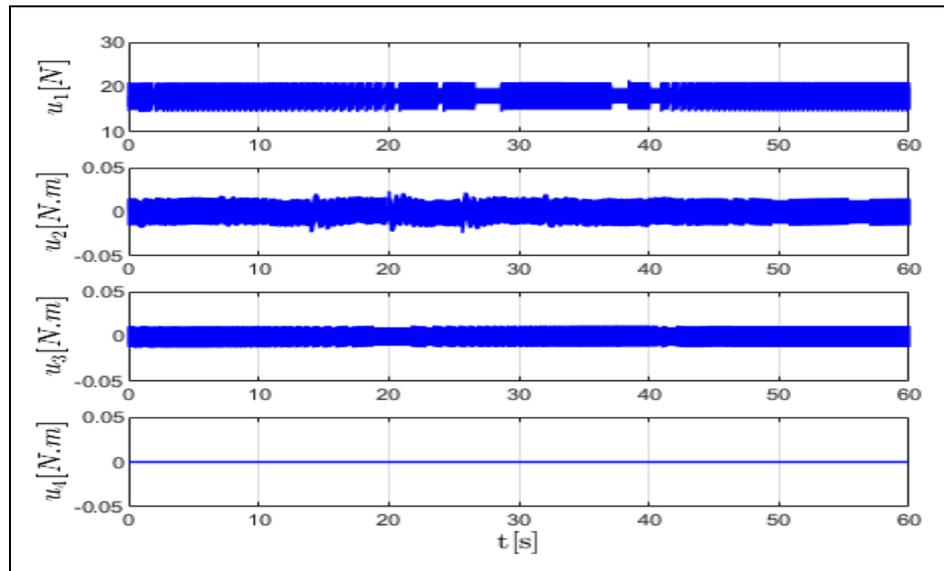


Figure 5.10 Control signals for the FLSMC Control

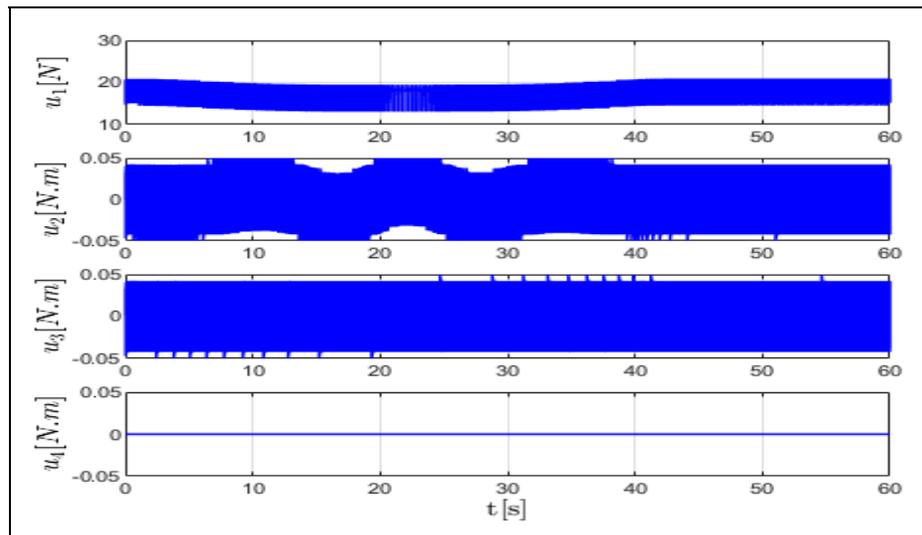


Figure 5.11 Control signals for the SM control

In the following figures, the aggressive (*sign*) function in the control is replaced with the smooth function (*tanh*). As it can be seen in Figure 5.12, the control signal is smooth. In addition to the sliding surface in Figure 5.13 becomes smoother. Furthermore, if Figure 5.6 is compared with Figure 5.14, the effect of chattering is noticed when (*sign*) function is used where it is smooth in Figure 5.14 due to the use of (*tanh*) function. On the other hand, the

error is smaller when using (*sign*) function due to the rapid reaction control. This is not clear in x and y because of the big error in the beginning. However, the errors in both cases are small values.

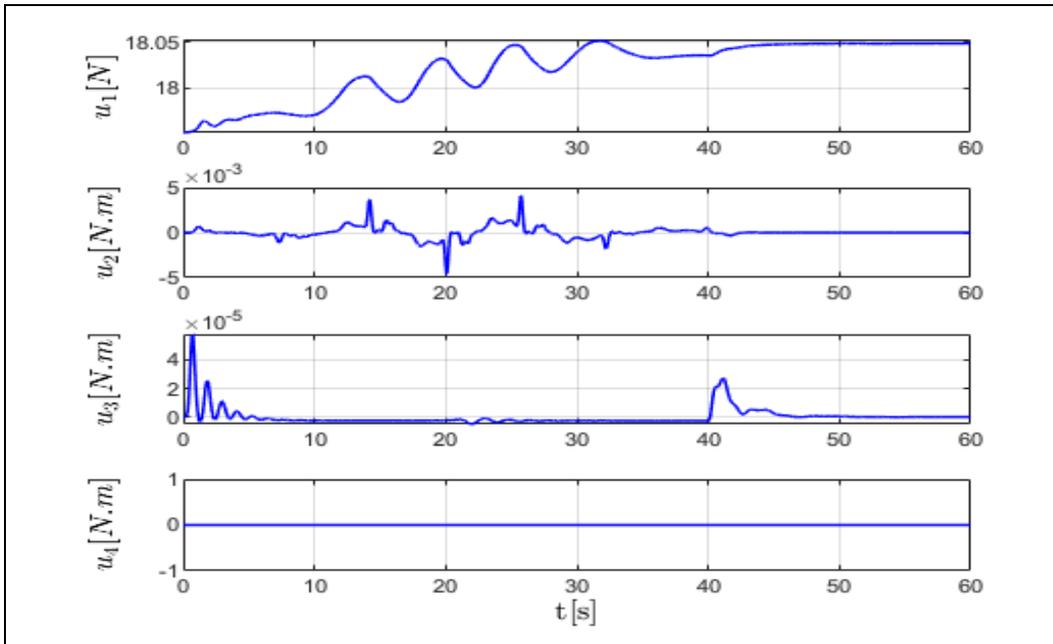


Figure 5.12 Control signals for the FLSMC control with (*tanh*) function

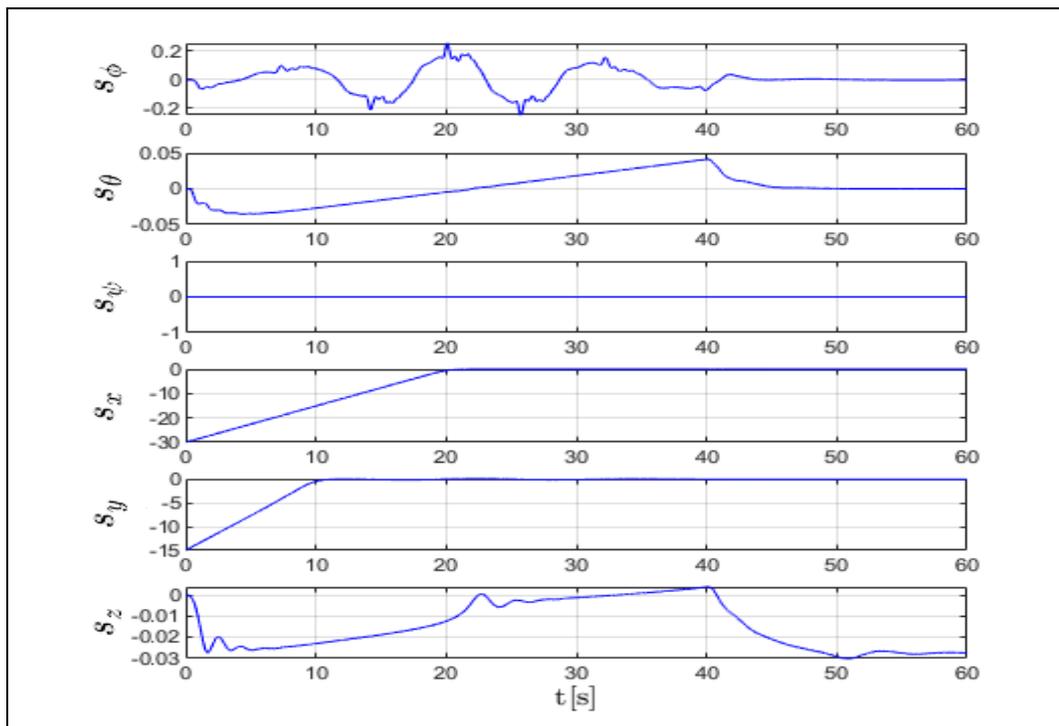


Figure 5.13 The sliding surface in FLSMC with (\tanh) function

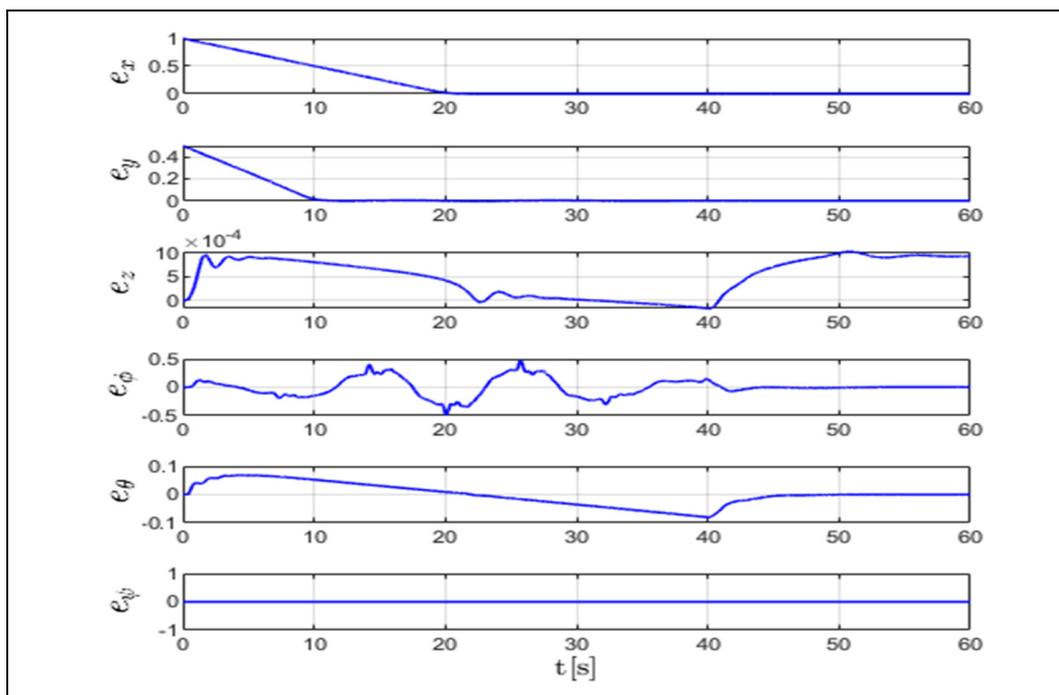


Figure 5.14 Translation and Rotational error with (\tanh) function

The performance in trajectory tracking and the small values of errors are good indicators of the accuracy of the proposed control. The smoothness of the estimated velocity indicates the quality of the SOED. The comparison with the traditional sliding mode in the table and in the figures proves the high quality of the proposed FLSMC control with TDE and SOED.

5.7 Conclusion

In this paper, a leader-follower quadrotor based on a visual system is presented. The proposed feedback linearization based on sliding mode controller deals directly with nonlinearity of the system without linearizing the model. The adaptive feature of the controller reduces the effect of the highly coupled dynamics in order to provide a robust and an accurate tracking. The velocity and the acceleration are estimated by a second order sliding mode estimator. Unmodelled dynamics and disturbance are handled by a time delay estimation TDE, which provides estimation of the external disturbances to impose desired stability and robustness properties on the global closed-loop system. The stability is studied by Lyapunov analysis and the dynamic model is implemented in Matlab/Simulink. The effectiveness of the proposed system is proved by the results and by a comparison with a conventional sliding mode, the error RMS values for (x, y, z) were reduced from $(0.3818, 0.1377, 3.00E - 03)$ to $(3.36E - 01, 0.1195, 1.00E - 04)$ respectively. The results show good performance and accuracy.

A suggested future work is to compare the proposed system with feedback linearization based on other types of control system in order to come up with the most efficient method. In addition to implement a real-life experimentation with external disturbance and to compare the effectiveness of each system.

CHAPTER 6

POSITION AND ATTITUDE TRACKING OF UNCERTAIN QUADROTOR BASED ON NON-SINGULAR TERMINAL SUPER-TWISTING ALGORITHM

Walid Alqaisi¹, Yassine Kali¹, Jawhar Ghommam², Maarouf Saad¹ and Vahé Nerguizian¹

¹ Department of Electrical Engineering, École de technologie supérieure,
1100 Notre-Dame West, Montreal, Quebec, Canada H3C 1K3

² Department of Electrical Engineering, Sultan Quaboos University,
Al Khoudh, Muscat 123, Oman

Paper published in Journal of Systems and Control Engineering, August, 2019

Abstract:

This paper proposes an improved non-singular terminal super twisting control for the problem of position and attitude tracking of quadrotor systems suffering from uncertainties and disturbances. The super-twisting algorithm STA is a second order sliding mode known to be a very effective control used to provide high precision and less chattering for uncertain nonlinear electromechanical systems. The proposed method is based on a non-singular terminal sliding surface with new exponent that solves the problem of singularity. The design procedure and the stability analysis of the closed loop system using Lyapunov theory are detailed for the considered system. Finally, the proposed control scheme is tested in simulations and by experiments on the parrot-rolling spider quadrotor. The results obtained show adequate performance in trajectory tracking and chattering reduction.

Keywords: Non-singular terminal sliding surface, Super-twisting algorithm, unmanned aerial vehicle, Altitude and attitude tracking, Finite time convergence.

6.1 Introduction

Nowadays, control of flying robotic systems has become an interesting topic of research. This interest is due to the fact that Unmanned Aerial Vehicles (UAVs) are used in many applications such as inspection, exploration, agriculture and transportation. Moreover, as all second order nonlinear systems, UAVs are affected by uncertainties due to the parameter changes and outward disturbances due to wind. For such reasons, designing nonlinear controller while taking into account the effect of the uncertainties and disturbances is a must to ensure high tracking performances. Recently, many nonlinear controllers have been developed for altitude and/or attitude trajectory tracking, such as feedback linearization (Abbasi, Ghayour, & Danesh, 2017; Al-Hiddabi, 2009; Voos, 2009), backstepping (Barikbin & Fakharian, 2019; Jiang, Lin, & Song, 2018; Mohd Basri, Husain, & Danapalasingam, 2015), sliding mode control (SMC) (Haitao Chen, Song, & Li, 2019; Jiang et al., 2018; Kali, Rodas, Gregor, Saad, & Benjelloun, 2018), finite time controller (Wu, Du, & Zhu, 2017; Zheng & Xian, 2018) and others. Since it was introduced, SMC (Utkin, Guldner, & Shi, 2009) has attracted great interest due its good features namely its insensitivity to matched uncertainties, its finite time convergence property and its simplicity of design. The SMC design consists of two steps:

- 1) Design of the sliding surface that represents the desired behavior of the system trajectory.
- 2) Design of a switching (discontinuous) control input that will force the system trajectory to reach the selected sliding surface in a finite time.

Despite of its good characteristics, SMCs real time implementation has an obstacle that represents its major disadvantage, namely the well-known chattering phenomenon (Boiko & Fridman, 2005). Chattering is caused by the fast unmodelled dynamics and/or the use of digital controllers with fixed sampling time. The chattering phenomenon results in undesirable performance, damage to mechanical parts in the system, and high energy loss (Utkin & Lee, 2006). In order to avoid the chattering, many propositions and approaches have been proposed (Ali et al., 2017; Hwachou Chen et al., 2019; Kali et al., 2019; Kali, Saad, Benjelloun, et al.,

2018; Razmi & Afshinfar, 2019; Y. Wang et al., 2019). In recent years, Second Order Sliding Mode SOSM control has been widely studied for a class of second-order nonlinear systems and has been considered a good solution to the chattering phenomenon while keeping the same robustness properties (Bartolini et al., 2003; Levant, 1993). In practical problems, SOSM control has been successfully implemented in many nonlinear systems as robotic manipulators (Azar et al., 2019; Kali, Saad, et al., 2017), induction machine drives (Benderradji et al., 2012; Kali, Rodas, et al., 2017), energy systems (Krim et al., 2018; Merabet et al., 2019) and others. However, the design of SOSM control law requires the measurement of the first time derivative of the designed sliding surface, which is often not available. Thus, this problem makes the implementation difficult.

As a solution, Super-Twisting Algorithm (STA) has been proposed (Guzmán & Moreno, 2015; Moreno, 2014; Moreno & Osorio, 2008). In addition to the fact that STA is a robust approach that produces less chattering and ensures fast finite time convergence, STA does not need the derivative of the sliding surface.

Actually, good control performance has been observed with this algorithm for lots of practical systems, such as robot manipulators (Kali, Saad, & Benjelloun, 2018; Kali, Saad, Benjelloun, et al., 2018), wind energy conversion system (Evangelista, Puleston, Valenciaga, & Fridman, 2012), switched reluctance motor (Rafiq, Rehman, Rehman, Butt, & Awan, 2012) and others. In addition, this algorithm has been implemented for attitude tracking of quadrotor UAV system (Derafa et al., 2012). However, the convergence time during the sliding phase depends on the designed switching surface. If this latter is not well selected, unacceptable or undesirable performance might be obtained. In the literature, a terminal sliding surface that is nonlinear has been proposed to improve the convergence time during the sliding phase (Zhihong & Yu, 1996). In spite of that, it suffers from the problem of singularity that has been covered and a non-singular terminal sliding mode has been proposed (Feng et al., 2002). However, the chattering phenomenon increases with the use of this nonlinear switching surface. Moreover, to the best of the authors' knowledge, all the developed STA control systems use classical

linear sliding surface because the use of STA-based in non-singular terminal sliding surface complicates the stability analysis and might increase the chattering.

Inspired by the above-mentioned published papers and by the good features of second order sliding mode, this paper proposes a position and attitude tracking based on super-twisting control algorithm with a new non-singular terminal sliding surface that proposes a solution to the well-known singularity problem. The contribution of this paper is an extension and improvement of the above-mentioned conventional STA method in the following two aspects:

- 1) It provides better comprehensive performance by proposing a new non-singular terminal sliding surface that uses an exponent that switches between two values to bypass the problem of singularity. The proposed modification will not affect the chattering while improving the convergence during the sliding phase.
- 2) In conventional STA approach, the gain must be chosen large to overcome the effects of the unmodelled dynamics. In our work, a new stability condition that will allow a small choice of gain while keeping good performances is established using Lyapunov theory. Hence, less chattering will be ensured.

The proposed method is tested on an uncertain quadrotor UAV system to show its improvement. The paper is divided into four sections. Section 5.2 introduces the position and attitude dynamic equations of a quadrotor UAV system. Section 5.3 demonstrates the design of the proposed super-twisting control algorithm based on the new non-singular terminal sliding surface and the stability of the closed loop system is proved theory. In Section 5.4, numerical simulation is provided to show the effectiveness of the proposed control scheme. The experimental results and analysis are given in Section 5.5. Finally, Section 5.6 concludes the paper.

6.2 Preliminaries

The quadrotor UAVs are flying robotic systems that consist of four independent motors fixed on a rigid cross structure. The considered one is shown in Figure 6.1. Their mathematical model is based on six-degrees-of freedom (DOF) given as:

$$[x, y, z, \phi, \theta, \psi]^T \in R^6 \quad (6.1)$$

Where $[x, y, z]^T \in R^3$ is the position vector including the altitude z and $[\phi, \theta, \psi]^T \in R^3$ is the Euler angles vector (roll ϕ , pitch θ and yaw ψ) that describes the attitude. Hence, the dynamic model can be divided into two parts (Samir Bouabdallah & Siegwart, 2005). The first part is the position dynamic model given by:

$$\begin{aligned} \ddot{x} &= -\frac{k_{ftx}}{m}\dot{x} + \frac{1}{m}(\cos\phi \sin\theta \cos\psi + \sin\phi \sin\psi)u_1 + d_x \\ \ddot{y} &= -\frac{k_{fty}}{m}\dot{y} + \frac{1}{m}(\cos\phi \sin\theta \sin\psi - \sin\phi \cos\psi)u_1 + d_y \\ \ddot{z} &= -\frac{k_{ftz}}{m}\dot{z} + g - \frac{1}{m}(\cos\phi \cos\theta)u_1 + d_z \end{aligned} \quad (6.2)$$

Where m denotes the mass of the quadrotor, g is the constant of gravity, k_{ftx} , k_{fty} and k_{ftz} are drag coefficients of translation, d_x , d_y and d_z are uncertain functions that satisfy $|\dot{d}_i| \leq D_i$ with $D_i > 0$ for $i = x, y, z$ and u_1 is the collective or the vertical force. In the second part, the attitude dynamic model is given by:

$$\begin{aligned} \ddot{\phi} &= \frac{I_y - I_z}{I_x}\dot{\theta}\dot{\psi} - \frac{J_r}{I_x}\dot{\theta}\omega_r - \frac{k_{fax}}{I_x}\dot{\phi}^2 + \frac{1}{I_x}u_2 + d_\phi \\ \ddot{\theta} &= \frac{I_z - I_x}{I_y}\dot{\phi}\dot{\psi} + \frac{J_r}{I_y}\dot{\phi}\omega_r - \frac{k_{fay}}{I_y}\dot{\theta}^2 + \frac{1}{I_y}u_3 + d_\theta \\ \ddot{\psi} &= \frac{I_x - I_y}{I_z}\dot{\theta}\dot{\phi} - \frac{k_{faz}}{I_z}\dot{\psi}^2 + \frac{1}{I_z}u_4 + d_\psi \end{aligned} \quad (6.3)$$

Where u_2 , u_3 and u_4 represent the torques in roll, pitch and yaw, respectively, k_{fax} , k_{fay} and k_{faz} are the coefficients of the aerodynamic friction, I_x , I_y and I_z denote the moments of inertia, J_r denotes the motor inertia, d_ϕ , d_θ and d_ψ are uncertain functions that satisfy $|\dot{d}_i| \leq D_i$ with $D_i > 0$ for $i = \phi, \theta, \psi$ and ω is the rotor speed that is linked to the torques by the following equations:

$$\begin{aligned}
 u_1 &= b(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \\
 u_2 &= b l_e (\omega_1^2 - \omega_2^2 - \omega_3^2 + \omega_4^2) \\
 u_3 &= b l_e (\omega_1^2 + \omega_2^2 - \omega_3^2 - \omega_4^2) \\
 u_4 &= c(-\omega_1^2 + \omega_2^2 - \omega_3^2 + \omega_4^2) \\
 \omega_r &= -\omega_1 + \omega_2 - \omega_3 + \omega_4
 \end{aligned} \tag{6.4}$$

Where c is the drag coefficient and b is the thrust coefficient and l_e is the lengths of the moment arm.

The control objective is to ensure that the quadrotor position tracks the desired known trajectory in a finite time with good accuracy in spite of suffering from uncertainties and disturbances. In the subsequent section, the controller and the stability analysis are performed by assuming that the 6-DOF vector $[x, y, z, \phi, \theta, \psi]^T$ and its time derivative are available for measurements.

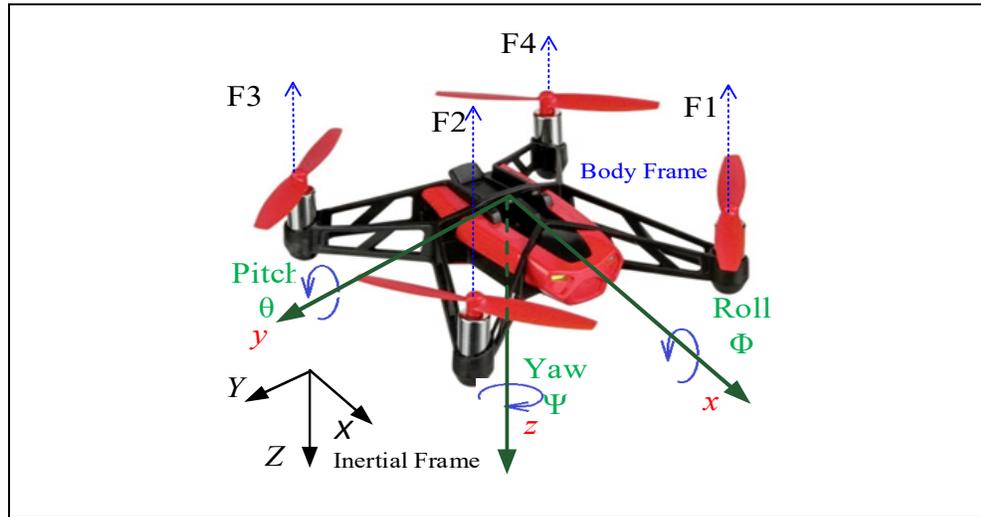


Figure 6.1 Considered quadrotor structure, forces, angles and frames ("Parrot Minidrone," 2018)

6.3 Controller Design

Super-twisting control algorithm based on the new nonsingular terminal sliding surface is designed in this section for uncertain quadrotor UAV systems in order to ensure a fast finite time convergence of the error and its derivative to zero. Figure 6.2 shows the block diagram of the control algorithm.

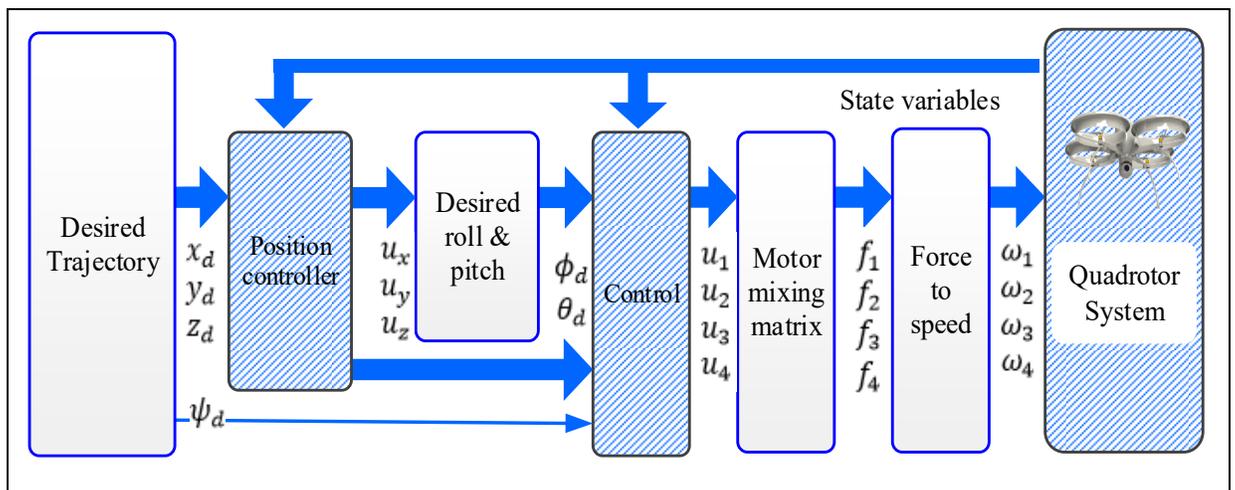


Figure 6.2 Architecture of the closed loop system

6.3.1 Position controller design

The position dynamic model given in (6.2) can be rewritten as follows:

$$\begin{aligned}
 \ddot{x} &= -\frac{k_{ftx}}{m}\dot{x} + u_x + d_x \\
 \ddot{y} &= -\frac{k_{fty}}{m}\dot{y} + u_y + d_y \\
 \ddot{z} &= -\frac{k_{ftz}}{m}\dot{z} + u_z + d_z
 \end{aligned} \tag{6.5}$$

Where u_x , u_y and u_z are virtual control inputs defined as:

$$\begin{aligned}
 u_x &= \frac{1}{m} (\cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi) u_1 \\
 u_y &= \frac{1}{m} (\cos \phi \sin \theta \sin \psi - \sin \phi \cos \psi) u_1 \\
 u_z &= g - \frac{1}{m} (\cos \phi \cos \theta) u_1
 \end{aligned} \tag{6.6}$$

Then, the first step in the proposed method design procedure is the selection of the new non-singular terminal sliding surface for $i = x, y, z$ as follows:

$$S_i = \dot{e}_i + \lambda_i |e_i|^{\beta_i} \text{sign}(e_i) \tag{6.7}$$

Where $e_x = x - x_d$, $e_y = y - y_d$ and $e_z = z - z_d$ denote the position tracking errors with x_d, y_d and z_d are the known desired trajectories, λ_x, λ_y and λ_z are positive constants and β_i for $i = x, y, z$ is defined by:

$$\beta_i = \begin{cases} 1 & , \text{ if } e_i < \epsilon \\ \beta_{i0} & , \text{ if } e_i > \epsilon \end{cases} \tag{6.8}$$

With $0.5 < \beta_{i0} < 1$ and ϵ is small positive constant used to provide limitation when $|e_i|$ is very small as explained below. Equation (6.8) is designed in a way to avoid the singularity in Terminal Sliding Mode (TSM) and it can be explained as follows, after taking the time derivative of S given in (6.7), we have:

$$\dot{S}_i = \ddot{e}_i + \lambda_i \beta_i |e_i|^{\beta_i - 1} \dot{e}_i \quad (6.9)$$

The conventional selection of β_i is given as $0.5 < \beta_i < 1$ therefore, the exponent $(\beta_i - 1)$ will be negative $-0.5 < (\beta_i - 1) < 0$. Hence, singularity (Feng et al., 2013) occurs in the transient response when $e_i = 0$ and $\dot{e}_i \neq 0$. It can be seen that if β_i is designed as proposed in (6.8), the singularity will be avoided since the term $|e_i|^{\beta_i - 1}$ will converge to one if $e_i = 0$ and the sliding surface will become linear. Small values of $|e_i|$ will not cause singularity (or very high value). In this application, we select $\epsilon = 0.01$, if we select, for example, $|e_i| = 0.01$, then the term $\beta_i |e_i|^{\beta_i - 1} \dot{e}_i = 4.870 \dot{e}_i$ for $\beta_i = 0.51$ and $\beta_i |e_i|^{\beta_i - 1} \dot{e}_i = 1.036 \dot{e}_i$ for $\beta_i = 0.99$. It means that the highest possible value for $\beta_i |e_i|^{\beta_i - 1}$ is 4.870 which does not cause a singularity problem. Therefore, to ensure the occurrence of the sliding motion, the equivalent control is designed such as the derivative of the sliding surface is equal to zero. Hence, developing (6.10) for $i = x, y, z$ gives:

$$\begin{aligned} \dot{S}_i &= \ddot{e}_i + \lambda_i \beta_i |e_i|^{\beta_i - 1} \dot{e}_i \\ &= \ddot{i} - \ddot{i}_d + \lambda_i \beta_i |e_i|^{\beta_i - 1} \dot{e}_i \\ &= -\frac{k_{fti}}{m} \dot{i} + u_i + d_i - \ddot{i}_d + \lambda_i \beta_i |e_i|^{\beta_i - 1} \dot{e}_i \end{aligned} \quad (6.10)$$

According to the super-twisting design procedure, the proposed controller can be obtained as follows:

$$u_i = u_{ieq} + u_{ista} \quad (6.11)$$

The aim of the equivalent part u_{ieq} is to control the nominal model. Its expression for x , y and z position is obtained by solving for $\dot{S} = 0$ using the nominal model as:

$$u_{ieq} = \frac{k_{fti}}{m} \dot{i} + \ddot{i}_d - \lambda_i \beta_i |e_i|^{\beta_i - 1} \dot{e}_i \quad (6.12)$$

While the aim of the STA term u_{ista} is used to ensure robustness against uncertainties and disturbances and to reduce the major problem of classical sliding mode control. Its expression is given by:

$$u_{ista} = -k_{1i} |S_i|^{0.5} \text{sign}(S_i) - k_{2i} \int_0^t \text{sign}(S_i) dt \quad (6.13)$$

Where k_{1i} and k_{2i} for $i = x, y, z$ are positive constants that will be determined later. The function $\text{sign}(S_i)$ is defined as:

$$\text{sign}(S_i) = \begin{cases} 1, & \text{if } S_i > 0 \\ 0, & \text{if } S_i = 0 \\ -1, & \text{if } S_i < 0 \end{cases} \quad (6.14)$$

Finally, the total thrust u_1 can be computed by the following equation (Zhao, Xian, Zhang, & Zhang, 2015):

$$u_1 = m \sqrt{u_x^2 + u_y^2 + (u_z + g)^2} \quad (6.15)$$

Theorem 1: Consider the nonlinear quadrotor UAV system (6.5), if the super-twisting gains are chosen for $i = x, y, z$ as follows:

$$\begin{aligned} k_{1i} &> 2\varepsilon, \quad \varepsilon > 0, \quad k_{2i} = ak_{1i} \\ a &> \frac{(\varepsilon^2 - \varepsilon k_{1i})^2 + D_i^2 + 4\varepsilon^4 - 2\varepsilon^3 k_{1i}}{2\varepsilon k_{1i}^2 + 4\varepsilon^2 k_{1i}} \end{aligned} \quad (6.16)$$

Where ε is a positive constant, then proposed controller (6.11) ensures the convergence of the sliding surface to zero in a maximal finite time T_{ri} :

$$T_{ri} \leq 2 \frac{\sqrt{\lambda_{\max}[R]}}{\lambda_{\min}[Q]} \sqrt{V(0)} \quad (6.17)$$

Proof: Substituting the proposed method (6.11) into the position dynamic model (6.5) gives the following closed loop error dynamics for $i = x, y, z$:

$$\begin{aligned} \dot{S}_i &= -k_{1i} |S_i|^{0.5} \text{sign}(S_i) + v_i \\ \dot{v}_i &= -k_{2i} \text{sign}(S_i) + \dot{d}_i \end{aligned} \quad (6.18)$$

The stability of the closed-loop system is proved by analysis, let us select the following Lyapunov function:

$$V = \xi^T R \xi \quad (6.19)$$

Where $\xi = [\xi_1 \ \xi_2]^T$ with $\xi_1 = |S_i|^{0.5} \text{sign}(S_i)$ and $\xi_2 = v_i$ and R is a symmetric positive definite matrix. By choosing appropriate matrix R as:

$$R = \begin{bmatrix} 2\varepsilon^3 + 2a\varepsilon k_{1i} & -2\varepsilon^2 \\ -2\varepsilon^2 & \varepsilon \end{bmatrix} \quad (6.20)$$

The above matrix is a symmetric positive definite if $a > 0$ and $\varepsilon > 0$. Therefore, choosing $k_{2i} = a k_{1i}$, the first time derivative of ξ is calculated as follows:

$$\begin{aligned} [\dot{\xi}_1 \ \dot{\xi}_2]^T &= \left[\frac{1}{2|S_i|^{0.5}} \dot{S}_i \quad \dot{v}_i \right]^T \\ &= \frac{1}{|\xi_1|} A \xi + \frac{1}{|\xi_1|} B \dot{d}_i |\xi_1| \end{aligned} \quad (6.21)$$

Where,

$$A = \begin{bmatrix} -\frac{1}{2}k_{1i} & \frac{1}{2} \\ -ak_{1i} & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Hence, the time derivative of V is computed as:

$$\begin{aligned} \dot{V} &= \dot{\xi}^T R \xi + \xi^T R \dot{\xi} \\ &= \frac{1}{|\xi_1|} \xi^T (A^T R + RA) \xi + \frac{2\dot{d}_i}{|\xi_1|} |\xi_1| B^T R \xi \\ &\leq \frac{1}{|\xi_1|} \xi^T (A^T R + RA) \xi + \dot{d}_i^2 |\xi_1|^2 + \xi^T R B B^T R \xi \\ &\leq \frac{1}{|\xi_1|} \xi^T (A^T R + RA + D_i^2 C^T C + R B B^T R) \xi \\ &\leq \frac{1}{|\xi_1|} \xi^T Q \xi \end{aligned} \tag{6.22}$$

Where $C = [1 \ 0]^T$. Then, the obtained Q is calculated as follows:

$$\begin{aligned} Q &= -(A^T R + RA + D_i^2 C^T C + R B B^T R) \\ &= \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix} \end{aligned} \tag{6.23}$$

Where,

$$Q_{11} = a(2\varepsilon k_{1i}^2 - 4\varepsilon^2 k_{1i}) + 2\varepsilon^3 k_{1i} - D_i^2 - 4\varepsilon^4$$

$$Q_{12} = Q_{21} = \varepsilon^3 - \varepsilon^2 k_{1i}$$

$$Q_{22} = \varepsilon^2$$

The obtained Q matrix is symmetrical definite positive if:

$$Q_{11} > 0 \tag{6.24}$$

$$Q_{22} > 0 \tag{6.25}$$

$$\det(Q) > 0 \tag{6.26}$$

The above conditions can be respectively rewritten as follows:

$$a(2\varepsilon k_{1i}^2 - 4\varepsilon^2 k_{1i}) + 2\varepsilon^3 k_{1i} - D_i^2 - 4\varepsilon^4 > 0 \quad (6.27)$$

$$\varepsilon > 0 \quad (6.28)$$

$$Q_{11}Q_{22} - Q_{12}^2 > 0 \quad (6.29)$$

By using (6.27), the first condition can be obtained as follows:

$$a > (D_i^2 + 4\varepsilon^4 - 2\varepsilon^3 k_{1i})(2\varepsilon k_{1i}^2 - 4\varepsilon^2 k_{1i})^{-1} \quad (6.30)$$

With $k_{1i} > 2\varepsilon$. Moreover, by using (6.29), the second condition can be obtained as follows:

$$a > \frac{(\varepsilon^2 - \varepsilon k_{1i})^2 + D_i^2 + 4\varepsilon^4 - 2\varepsilon^3 k_{1i}}{2k_{1i}^2 - 4\varepsilon^2 k_{1i}} \quad (6.31)$$

With $k_{1i} > 2\varepsilon$. Hence, if the conditions in (6.16) are verified, \dot{V} is negative definite. Therefore, the closed-loop stability of the system is proved.

In order to prove the finite time convergence, at first, we recall that the Lyapunov function V is bounded:

$$\lambda_{\min}[R]\|\xi\|_2^2 \leq V \leq \lambda_{\max}[R]\|\xi\|_2^2 \quad (6.32)$$

With $\lambda_{\min}[R]$ and $\lambda_{\max}[R]$ denote the minimum and maximum eigenvalues of R . Then, equation (6.32) can be rewritten as:

$$\frac{\sqrt{V}}{\sqrt{\lambda_{\max}[R]}} \leq \|\xi\|_2 \leq \frac{\sqrt{V}}{\sqrt{\lambda_{\min}[R]}} \quad (6.33)$$

Moreover, equation (6.22) gives:

$$\begin{aligned}
\dot{V} &\leq -\frac{1}{|\xi_1|} \xi^T Q \xi \\
&\leq -\frac{1}{|\xi_1|} \lambda_{\min}[Q] \|\xi\|_2^2
\end{aligned} \tag{6.34}$$

Where $\lambda_{\min}[Q]$ is the minimum eigenvalue of Q . Since it is obvious that $\|\xi\|_2 \geq |\xi_1|$ then:

$$\dot{V} \leq -\frac{\lambda_{\min}[Q]}{\sqrt{\lambda_{\max}[P]}} \sqrt{V} \tag{6.35}$$

According to the last equation, the system trajectories converge to the sliding surface in a finite time. This completes the proof.

6.3.2 Attitude controller design

The objective of this part is to control the Euler angles. To this end, the same methodology will be used. First of all, let us rewrite the attitude dynamic model given in (6.3) as follows:

$$\begin{aligned}
\ddot{\phi} &= f_{\phi} + g_{\phi} u_{\phi} + d_{\phi} \\
\ddot{\theta} &= f_{\theta} + g_{\theta} u_{\theta} + d_{\theta} \\
\ddot{\psi} &= f_{\psi} + g_{\psi} u_{\psi} + d_{\psi}
\end{aligned} \tag{6.36}$$

Where $u_{\phi} = u_2$, $u_{\theta} = u_3$ and $u_{\psi} = u_4$ while f_i and g_i for $j = \phi, \theta, \psi$ are defined as:

$$\begin{aligned}
f_{\phi} &= \frac{1}{I_x} (-k_{fax} \dot{\phi}^2 + (I_y - I_z) \dot{\theta} \dot{\psi} - J_r \dot{\theta} \omega_r) \\
f_{\theta} &= \frac{1}{I_y} (-k_{fay} \dot{\theta}^2 + (I_z - I_x) \dot{\phi} \dot{\psi} + J_r \dot{\phi} \omega_r) \\
f_{\psi} &= \frac{1}{I_z} (-k_{faz} \dot{\psi}^2 + (I_x - I_y) \dot{\theta} \dot{\phi}) \\
g_{\phi} &= \frac{1}{I_x}, \quad g_{\theta} = \frac{1}{I_y}, \quad g_{\psi} = \frac{1}{I_z}
\end{aligned} \tag{6.37}$$

Now, let us select the new non-singular terminal sliding surface for $j = \phi, \theta, \psi$ the roll, pitch and yaw as follows:

$$S_j = \dot{e}_j + \lambda_j |e_j|^{\beta_j} \text{sign}(e_j) \quad (6.38)$$

Where $e_\phi = \phi - \phi_d$, $e_\theta = \theta - \theta_d$ and $e_\psi = \psi - \psi_d$ are the attitude tracking errors. The desired roll ϕ_d and the desired pitch θ_d are found as in (6.40) and the desired yaw ψ_d is an input. are known desired orientations. $\lambda_\phi, \lambda_\theta$ and λ_ψ are positive constants and β_j for $j = \phi, \theta, \psi$ is defined by:

$$\beta_j = \begin{cases} 1 & , \text{ if } e_j < \epsilon \\ \beta_{j0} & , \text{ if } e_j > \epsilon \end{cases} \quad (6.39)$$

Here, the desired roll and pitch angles are generated from the virtual controllers (Zhao et al., 2015) as follows:

$$\begin{aligned} \phi_d &= \arcsin\left(\frac{m}{u_1}(u_x \sin(\psi_d) - u_y \cos(\psi_d))\right) \\ \theta_d &= \arctan\left(\frac{1}{u_z + g}(u_x \cos(\psi_d) + u_y \sin(\psi_d))\right) \end{aligned} \quad (6.40)$$

Therefore, the first time derivative of the sliding surfaces (6.38) is calculated for $j = \phi, \theta, \psi$ as follows:

$$\begin{aligned} \dot{S}_j &= \ddot{e}_j + \lambda_j \beta_j |e_j|^{\beta_j - 1} \dot{e}_j \\ &= \ddot{j} - \ddot{j}_d + \lambda_j \beta_j |e_j|^{\beta_j - 1} \dot{e}_j \\ &= f_j + g_j u_j + d_i - \ddot{j}_d + \lambda_j \beta_j |e_j|^{\beta_j - 1} \dot{e}_i \end{aligned} \quad (6.41)$$

Then, based on the nominal model, the equivalent control for the attitude tracking is obtained as:

$$u_{jeq} = g_j^{-1}(-f_i + \ddot{j}_d - (\lambda_j \beta_j |e_j|^{\beta_j - 1} \dot{e}_j)) \quad (6.42)$$

While the STA terms are obtained as:

$$u_{jsta} = -g_j^{-1}(k_{1j}|S_j|^{0.5} \text{sign}(S_j) + k_{2j} \int_0^t \text{sign}(S_j) dt) \quad (6.43)$$

Where k_{1j} and k_{2j} are positive constants. Finally, the proposed controller for the attitude tracking for $j = \phi, \theta, \psi$ is given as:

$$u_j = u_{jeq} + u_{jsta} \quad (6.44)$$

Theorem 2: Consider the attitude model of the quadrotor UAV system (6.3), if the super-twisting gains for $j = \phi, \theta, \psi$ are chosen as follows:

$$\begin{aligned} k_{1j} &> 2\varepsilon, \quad \varepsilon > 0, \quad k_{2j} = bk_{ij} \\ b &> \frac{(\varepsilon^2 - \varepsilon k_{1j})^2 + D_j^2 + 4\varepsilon^4 - 2\varepsilon^3 k_{1j}}{2\varepsilon k_{1j}^2 - 4\varepsilon^2 k_{1j}} \end{aligned} \quad (6.45)$$

Then, the proposed controller (6.44) ensures the convergence of the sliding surfaces (6.38) to zero in a finite time.

Proof: The proof is similar to the one of tracking position.

6.4 Numerical Simulations

In this section, the simulation results are presented to validate the effectiveness of the proposed non-singular terminal super twisting algorithm. Simulation is carried out by Matlab/Simulink software for the studied quadrotor described in equations (6.2) and (6.3). The parameters are used based on parrot-rolling spider quadrotor (Technology, 2018) as given in Table 6.1. Moreover, to prove the effectiveness of our method, it is compared to the standard STA (Derafa et al., 2012).

Table 6.1 Physical parameters of the quadrotor.

Parameter	Value	Unit
m	0.068	[kg]
I_x	0.0686×10^{-3}	[kg.m ²]
I_y	0.0920×10^{-3}	[kg.m ²]
I_z	0.1366×10^{-3}	[kg.m ²]
J_r	1.0209×10^{-7}	[kg.m ²]
g	9.81	[m/s ²]

The following reference trajectory is built to assess the quadrotor's tracking performance:

$$\begin{aligned}
 x_d &= \begin{cases} 0 & \text{if } t < 3 \text{ seconds} \\ 0.5 \sin(0.5t) & \text{if } t > 3 \text{ seconds} \end{cases} \\
 y_d &= \begin{cases} 0 & \text{if } t < 3 \text{ seconds} \\ 0.5 + \sin(0.5t) & \text{if } t > 3 \text{ seconds} \end{cases} \\
 z_d &= 1, \quad \psi_d = 0
 \end{aligned} \tag{6.46}$$

The initial altitude positions are chosen to be $x(0) = y(0) = z(0) = 0$, while initial values of Euler angles are $\phi(0) = \theta(0) = \psi(0) = 0$. For the adopted scenario, the following disturbances are added:

$$d_x = d_y = d_z = 0.3 \cos(10\pi t) \quad (6.47)$$

During the simulations, the chosen gains for the proposed controller are as in Table 6.2 while the chosen gains for the classical STA are listed in Table 6.3. The gains are chosen based on the stability conditions of each controller.

Table 6.2 Proposed controller gains (simulation)

Parameter	Value	Parameter	Value	Parameter	Value
β_{x0}	0.8	β_{y0}	0.8	β_{z0}	0.8
λ_x	10	λ_y	10	λ_z	1
K_{1x}	7	K_{1y}	7	K_{1z}	7
K_{2x}	12.65	K_{2y}	12.65	K_{2z}	12.65
$\beta_{\phi 0}$	0.8	$\beta_{\theta 0}$	0.8	$\beta_{\psi 0}$	0.8
λ_ϕ	1	λ_θ	1	λ_ψ	1
$K_{1\phi}$	7	$K_{1\theta}$	7	$K_{1\psi}$	7
$K_{2\phi}$	12.65	$K_{2\theta}$	12.65	$K_{2\psi}$	12.65

Table 6.3 Classical STA gains (simulation)

Parameter	Value	Parameter	Value	Parameter	Value
λ_x	10	λ_y	10	λ_z	1
λ_ϕ	1	λ_θ	1	λ_ψ	1
K_{1x}	13.82	K_{1y}	13.82	K_{1z}	13.82
$K_{1\phi}$	13.82	$K_{1\theta}$	13.82	$K_{1\psi}$	13.82
K_{2x}	10.37	K_{2y}	10.37	K_{2z}	10.37
$K_{2\phi}$	10.37	$K_{2\theta}$	10.37	$K_{2\psi}$	10.37

The simulation results are presented in the given figures. Free space 3D tracking is presented in Figure 6.3 where it can be noticed that the proposed approach gave better result. In Figures. 6.4, 6.6 and 6.7, position and angles tracking are shown. It can be seen that both methods allow good performance in position trajectory tracking. However, based on the error signals for position and orientation shown in Figures 6.5 and 6.8, it is clear that the smallest values of errors are obtained using the proposed method. This is good indicators of its accuracy and good features. Finally, it can be seen in Figure 6.9, the control signal is kept to small values with very low chattering in comparison with the standard STA. The small effort obtained by the proposed approach justify the fact that the variations of the generated Euler angles are smaller. This encourages the real-time implementation that will be presented in the next section. To support these results, we compared both controllers based on the Root-Mean-Squared (RMS) error defined for $l = x, y, z, \phi, \theta, \psi$ by:

$$\text{RMS}(e_l) = \left(\frac{1}{N} \sum_{k=1}^N \|e_l(k)\|^2 \right)^2 \quad (6.48)$$

Where N is the number of simulation samples. As reported in Table 6.4, all the RMS error values show the superiority of the proposed method except x-tracking.

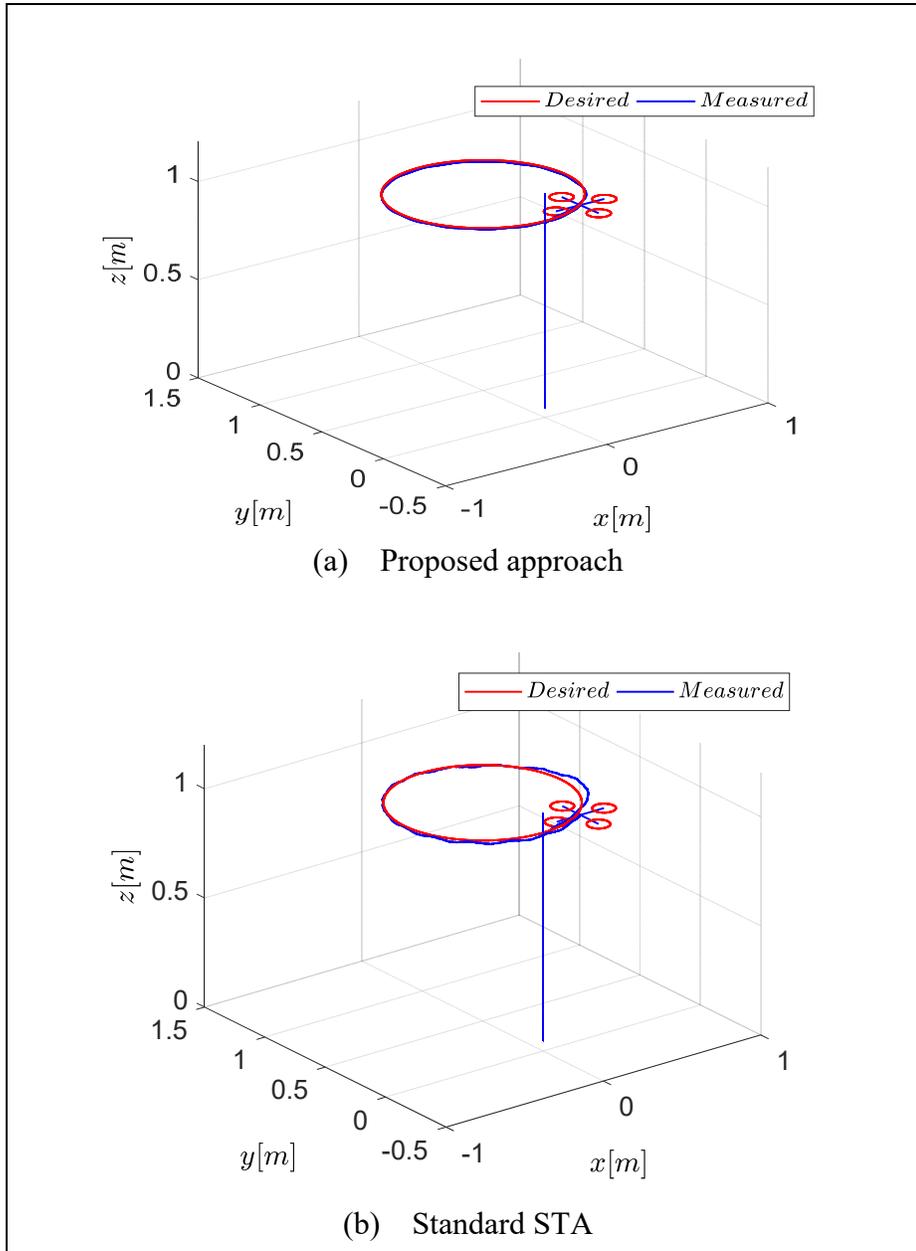


Figure 6.3 3D trajectory tracking

Table 6.4 Comparative results

Parameter	Proposed Approach	Standard STA
$RMS e_x$	0.021	0.004
$RMS e_y$	1.86×10^{-4}	0.0011
$RMS e_z$	0.0272	0.031
$RMS e_\phi$	3.83×10^{-7}	4.41×10^{-4}
$RMS e_\theta$	6.19×10^{-7}	6.35×10^{-4}
$RMS e_\psi$	1.84×10^{-7}	7.081×10^{-4}

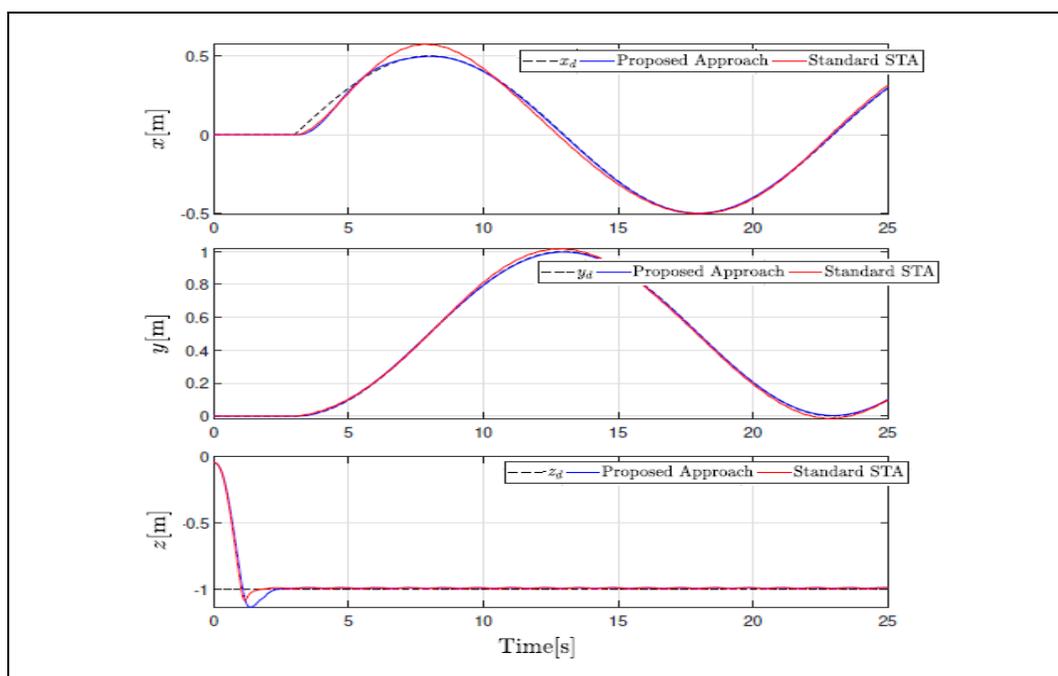


Figure 6.4 Position tracking

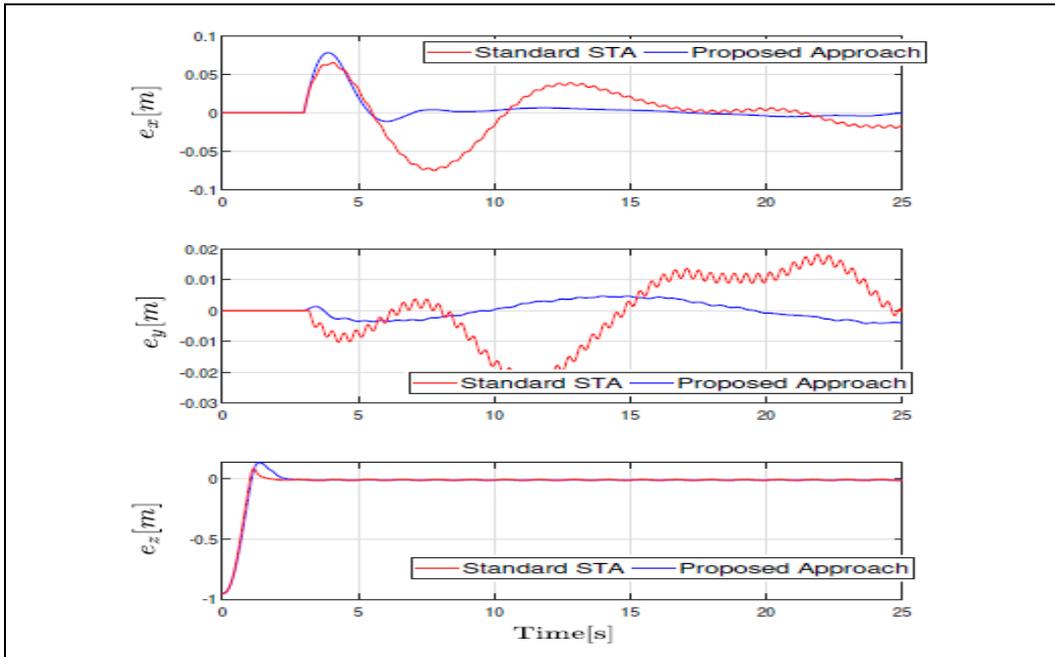


Figure 6.5 Errors in position tracking

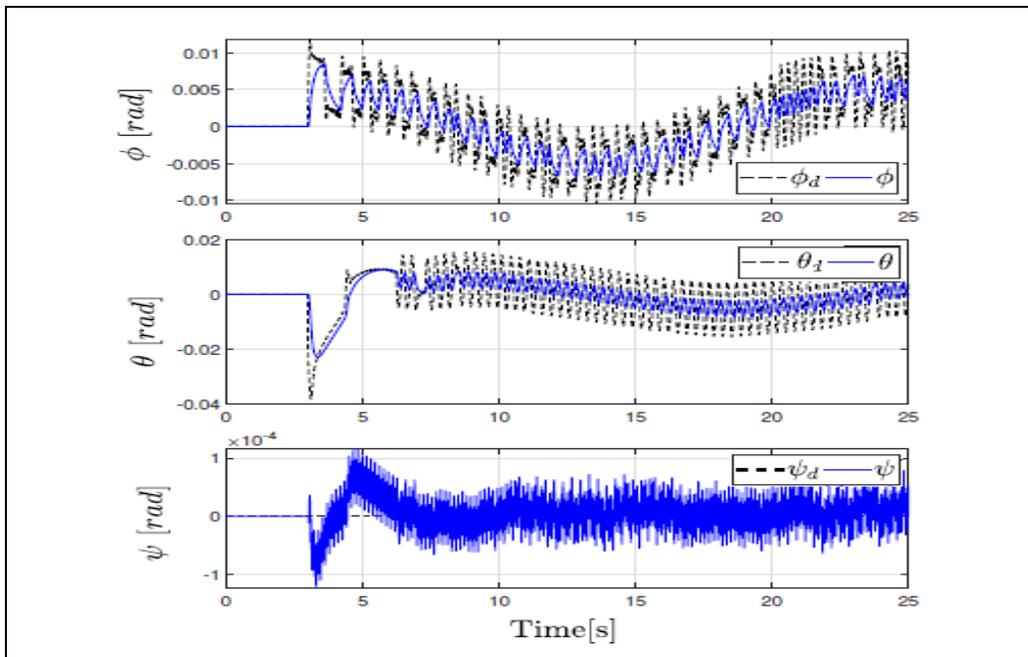


Figure 6.6 Euler angles response via proposed approach

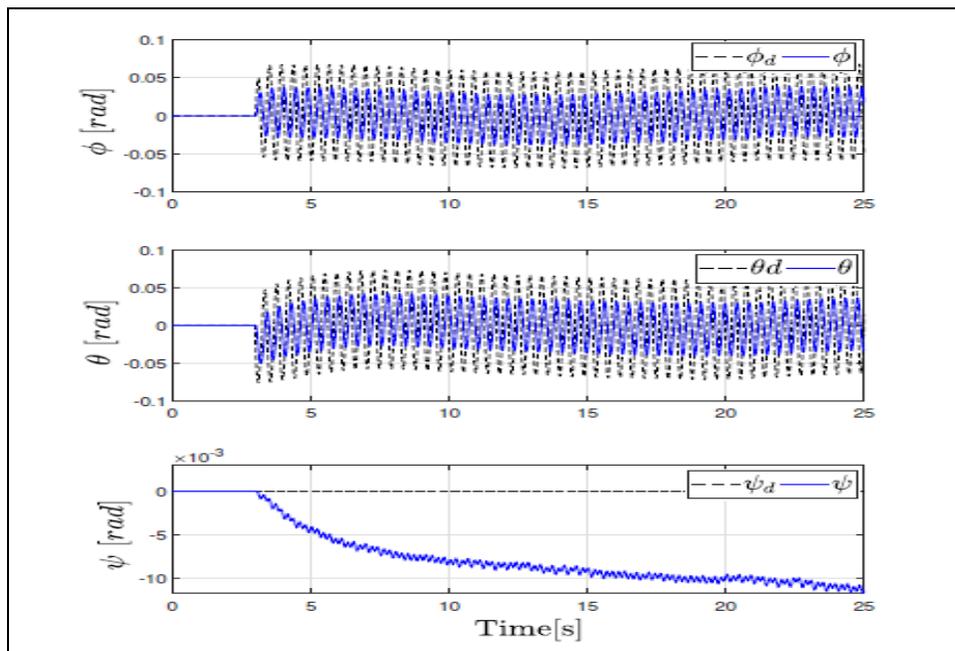


Figure 6.7 Euler angles response via standard STA

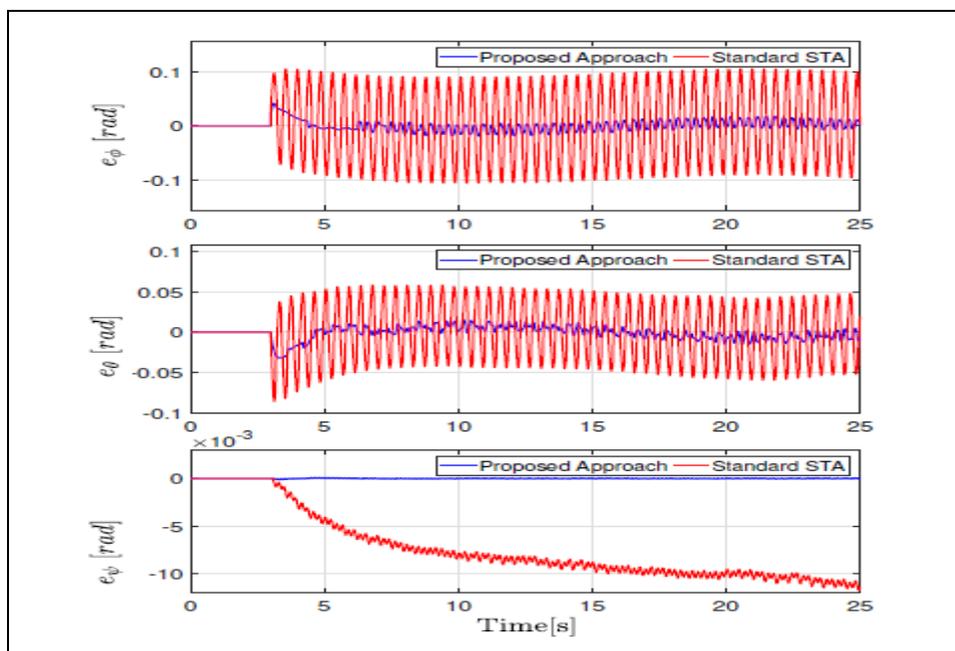


Figure 6.8 Errors in orientation

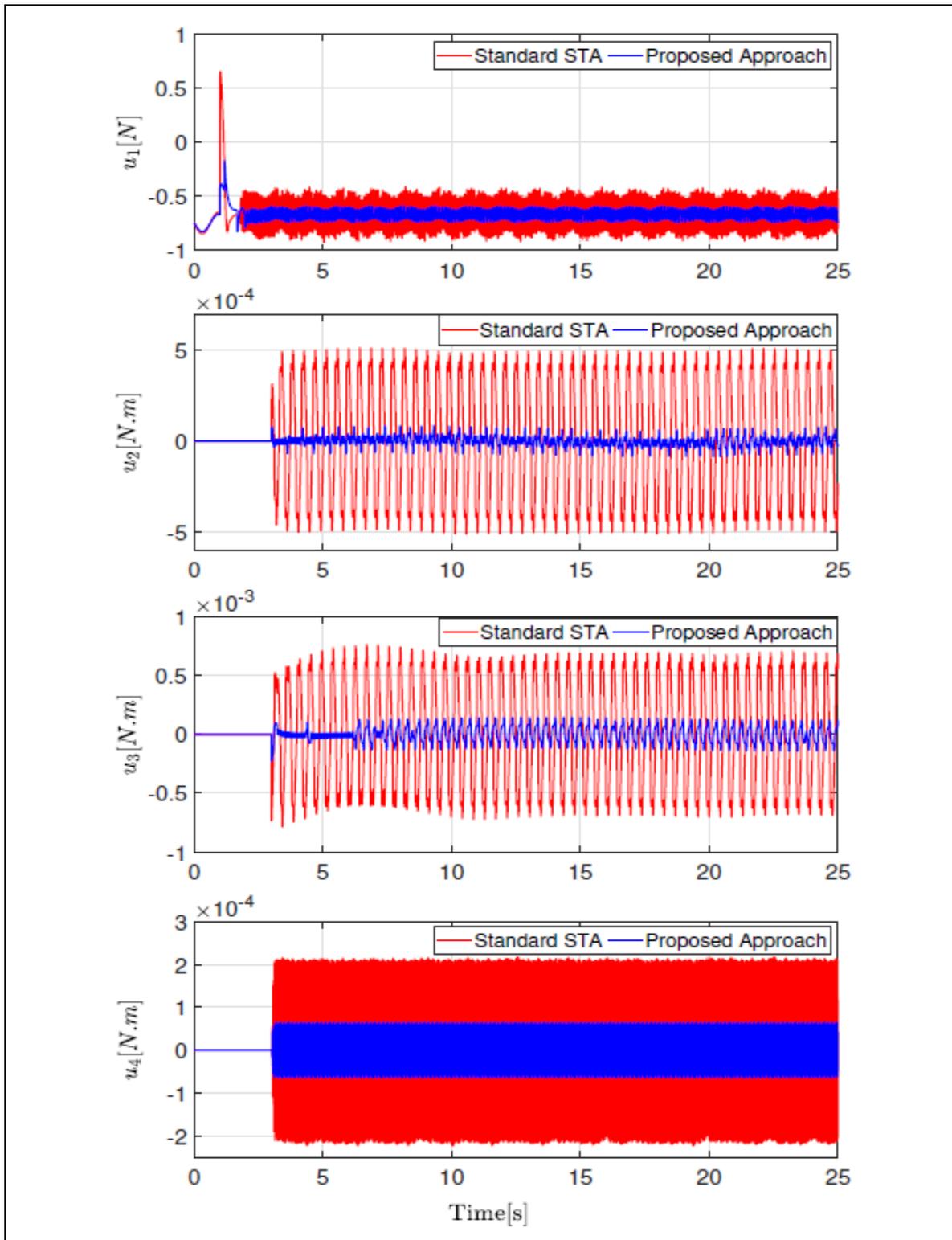


Figure 6.9 Control Signals

6.5 Practical Implementation

The performance of the proposed system is demonstrated experimentally in this part. The practical implementation is based on Simulink support package for PARROT minidrone. Parrot has an integrated IMU with a three-axis gyroscope, three-axis accelerometer, a compass, as well as altitude sonar and pressure sensors. It is also equipped with a battery of life-time up to 8 minutes.

This software was developed jointly by the Massachusetts Institute of Technology MIT, and the Parrot minidrones company. This type of minidrones facilitates building and deploying flight control algorithm on Parrot minidrones. After building the desired control, Simulink Coder is used to generate and executes the code from the Simulink model. The generated source code can be used for real time and non real-time applications, including simulation, rapid prototyping, and hardware-in-the-loop testing. The generated code then is deployed to the drone wirelessly by Bluetooth Smart technology V4.0 communication. The workflow of the implementation is summarized in Figure 6.10. The quadrotors inertial measurement unit IMU provides measurement of the translational accelerations in the body frame $\ddot{X}_B = [\ddot{x}_B \ \ddot{y}_B \ \ddot{z}_B]^T$. The translational acceleration in the inertial frame $\ddot{X} = [\ddot{x} \ \ddot{y} \ \ddot{z}]^T$, velocity $\dot{X}_B = [\dot{x}_B \ \dot{y}_B \ \dot{z}_B]^T$ and position $X_B = [x_B \ y_B \ z_B]^T$ are found in the developed support package (Mathworks, 2018; Technology, 2018), based on the following relationships:

$$\begin{aligned}\ddot{X} &= \mathbf{R} \ddot{X}_B \\ \dot{X} &= \mathbf{R} \left[\int \frac{F}{m} + V_B \times \Omega \right] \\ X &= \frac{k_1 \tau}{z - 1} \dot{X}\end{aligned}\tag{6.49}$$

Where F represents the applied forces in quadrotor body, V_B is the velocity with respect to the body frame, Ω is the body-fixed frame angular velocity vector, k_1 is a constant and its value is 0.01. Simulink Coder enables recording flight data on the minidrone and access the code generated from Simulink model. The reference trajectory and starting conditions are described

as in the simulation part. During the experiment, the drag coefficients k_{fti} are assumed to be part of the vector of perturbations since they are not easily obtained in real time. Hence the controller gains and constants are chosen as in Table 6.5.

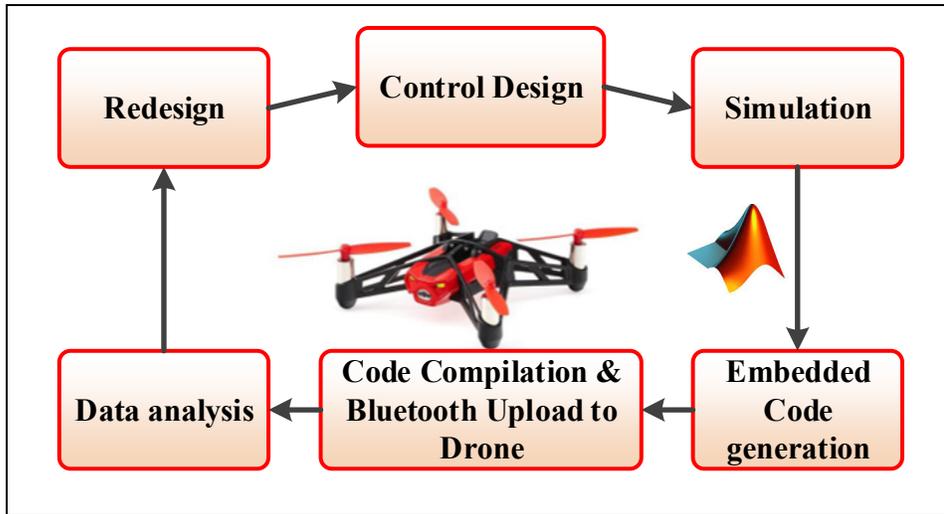


Figure 6.10 Implementation workflow

Table 6.5 Constants and gains (experiment)

Parameter	Value	Parameter	Value	Parameter	Value
β_{x0}	0.2	β_{y0}	0.2	β_{z0}	0.2
λ_x	1	λ_y	1.3	λ_z	3
K_{1x}	2.1	K_{1y}	2.1	K_{1z}	0.7
K_{2x}	0.01	K_{2y}	0.01	K_{2z}	0.1
$\beta_{\phi 0}$	0.2	$\beta_{\theta 0}$	0.2	$\beta_{\psi 0}$	0.2
λ_{ϕ}	10	λ_{θ}	10	λ_{ψ}	5
$K_{1\phi}$	0.3	$K_{1\theta}$	0.4	$K_{1\psi}$	3
$K_{2\phi}$	0.01	$K_{2\theta}$	0.01	$K_{2\psi}$	0.1

Experimental results are presented in Figures 6.11 to 6.17. Figure 6.11-(a) shows 3D task space tracking of the desired trajectory, while Figure 6.11-(b) shows the tracking of the trajectory in x-y direction. Figure 6.12 shows trajectory tracking for each axis. Both Figures 6.11 and 6.12 show good tracking during the whole operation time. Orientation angles response is displayed in Figure 6.13 which shows fast response of the angles to stabilize the system. The velocity is found simultaneously as shown in Figure 6.14. Figures 6.15 and 6.16 show the error signals, small value of the errors can be noticed. It can be seen in Figure 6.17 that the control torque input efforts are small values that are quite similar to the ones obtained in simulation. The proposed controller ensures good tracking of the desired trajectory with accuracy.

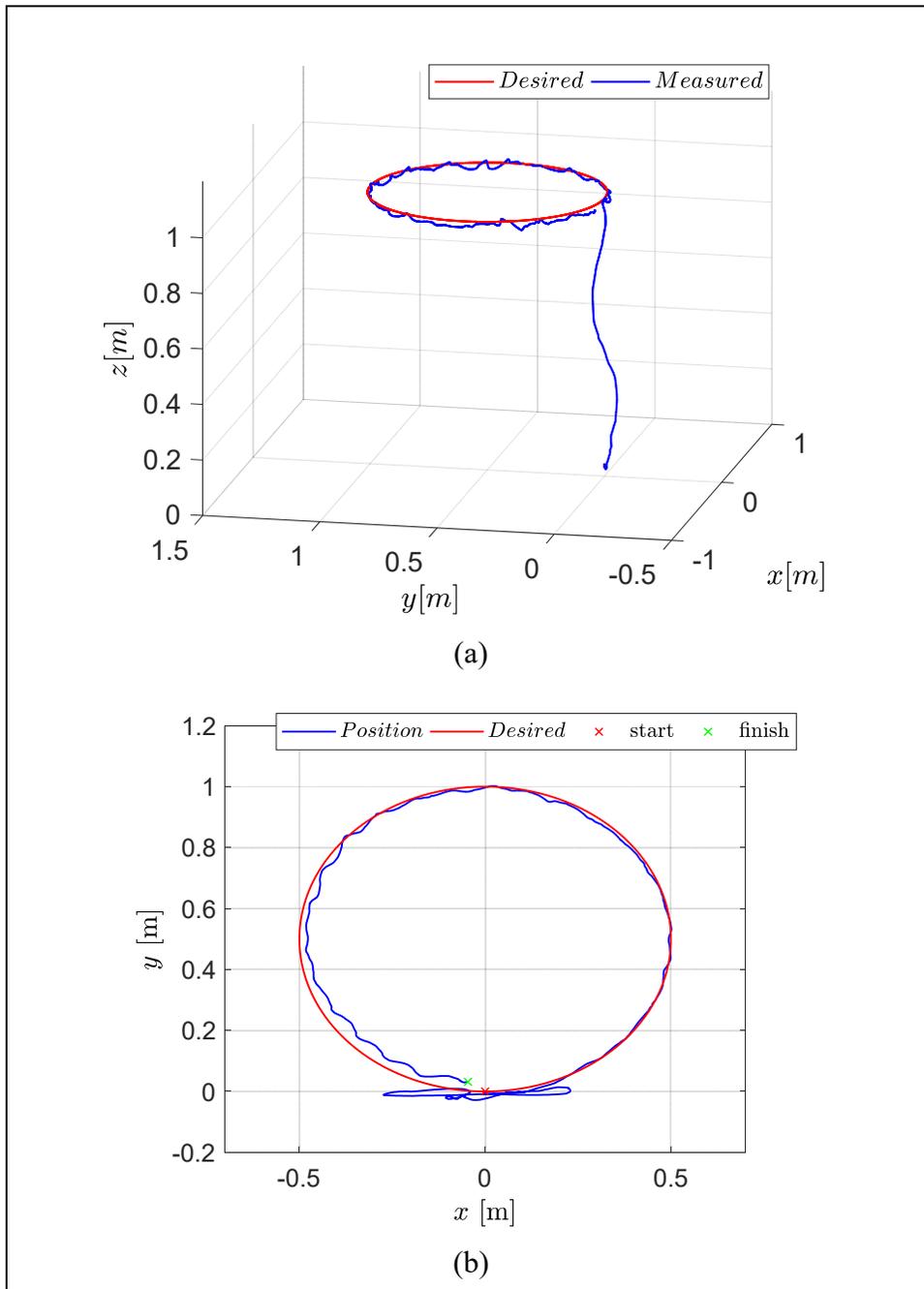


Figure 6.11 Workspace trajectory tracking: (a) In 3D
(b) In x - y

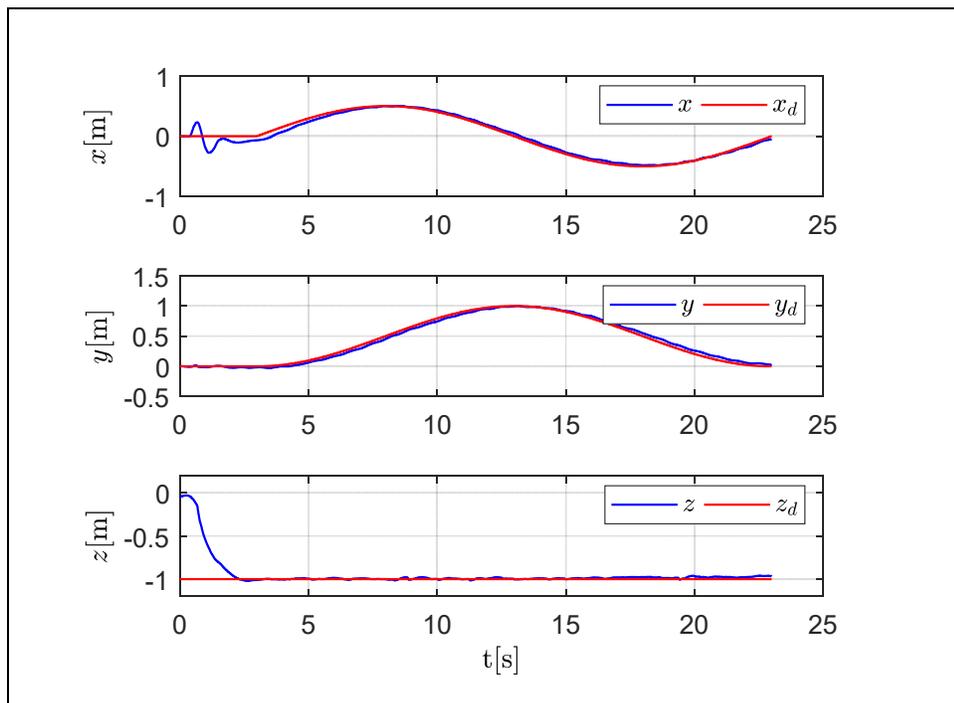


Figure 6.12 Position and altitude trajectory

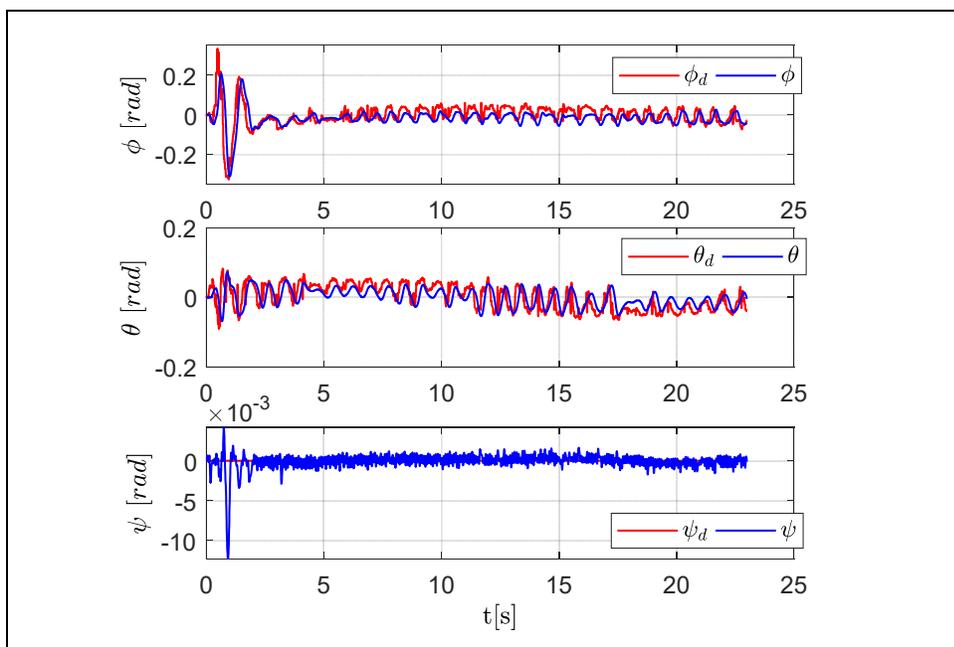


Figure 6.13 Euler angles response

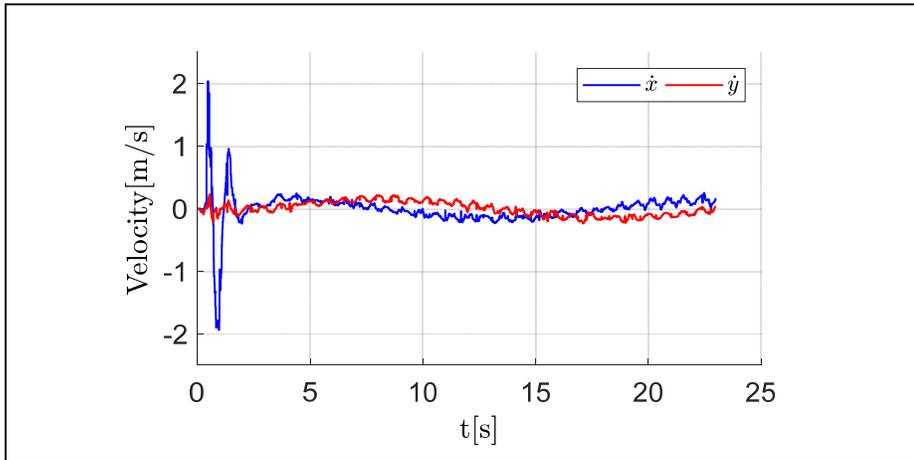
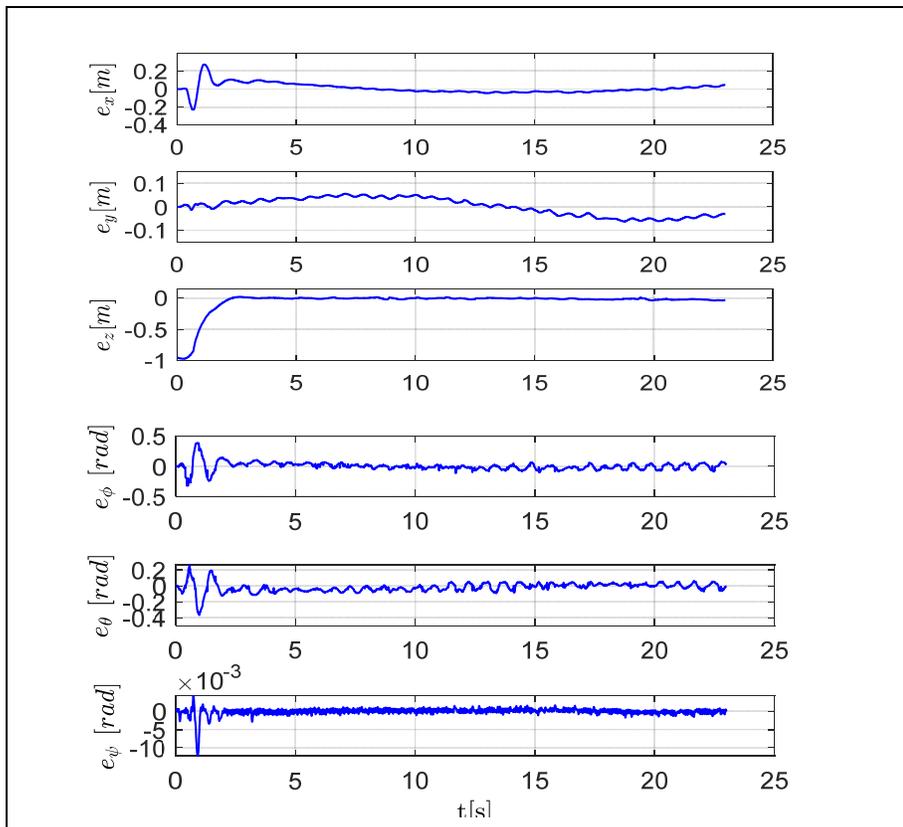
Figure 6.14 Velocities of x and y 

Figure 6.15 Errors in position, altitude and orientation

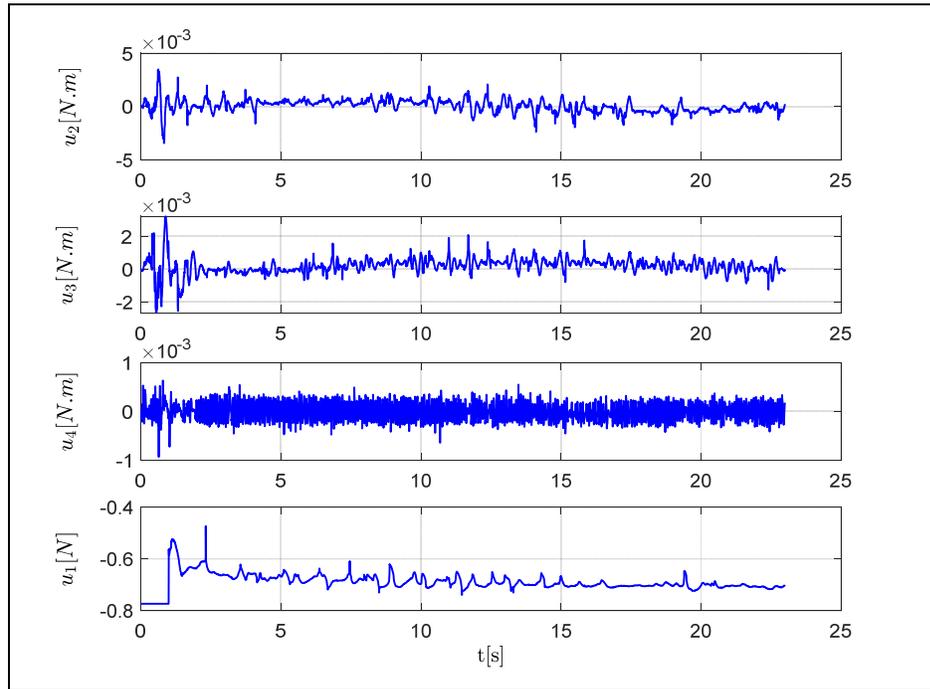


Figure 6.16 Control signals

6.6 Conclusion

In this work, a super twisting second order sliding mode controller based on a modified non-singular terminal sliding surface has been developed and successfully implemented on the minidrone parrot-rolling spider quadrotor. The proposed nonlinear switching surface has been designed using a new exponent that bypass the singularity problem that occurs when only the position error converges to zero. Then, the chosen control algorithm has been derived and a new stability conditions have been established to reduce the chattering phenomenon, to ensure finite time convergence and robustness such as the effects of the uncertainties and the perturbations are rejected. The simulation and experimental results obtained on the considered quadrotor showed clearly the efficiency of the proposed approach in position and attitude tracking and rejection of perturbations and uncertainties.

CONCLUSION AND FUTURE WORK

This thesis focuses on solving important challenges in robotics applications. The problem of uncertainty or the change in robot parameters and unmodelled dynamics is studied and solved in two papers. At first, perturbation is reduced hierarchically by using the Hierarchical Perturbation Compensator (HPC). The later uses three subsystems depend on a feedback signal, a feedforward signal and the dynamic error. Each subsystem provides advantages that cover the drawbacks of the other subsystems.

The second paper proposed a system designed to track perturbation in three-loop approach called Three-Loop Uncertainty Compensator (TLUC). This system has adaptive and integral features that give the ability to track perturbation and residual perturbation in the mechanical system. The HPC and the TLUC provide estimation and compensation of disturbance and uncertainties in real time. The Exponential Reaching Law Sliding Mode (ERSM) ensures full control of position, attitude and altitude and guarantees low chattering and fast response. As a result, the closed loop system can be driven to asymptotic stability. The performance of the complete system is analyzed by Lyapunov function, simulation, and experiment. The results show high performance of the proposed system in minimizing the effects of uncertainties and disturbances.

Hard Nonlinearities and highly coupled dynamics in UAV Quadrotor are studied in this thesis. Feedback linearization based on sliding mode controller is proposed to deal directly with nonlinearity of the system without linearizing the model. Uncertainties are evaluated and velocity and the acceleration are estimated by a second order sliding mode estimator. The stability is studied by Lyapunov analysis and by simulation.

New non-Singular Terminal Super-Twisting Algorithm is proposed in order to reduce chattering problem, utilize terminal sliding surface, ensure finite-time convergence, prevent singularity in the terminal sliding mode control and to restrict high values of the super twisting gains. New stability conditions are proposed to reduce finite time convergence. The system

performance is analyzed by Lyapunov function, tested and compared with standard super twisting and implemented experimentally. The results proved the performance of the proposed control.

A comparative study is performed in this thesis for the different proposed systems, the HPC, TLUC, FLSM and NTST as in Appendix II. The objective of this study is to compare the performance of each system with the other systems. In this comparison, all of the systems go through same conditions.

The developed perturbation compensators and controller are found to provide some possible solutions for the field of control of nonlinear systems. They could improve performance in the presence of different perturbations. As a future work in this thesis:

- More research and testing is possible to analyze the performance on larger quadrotors with big change of load in order to verify to what limits the compensators can be effective.
- The proposed systems can be tested on different types of nonlinear robotics systems such as manipulators, car-like robots and underwater vehicles.
- The developed systems can be tested using observers in the case where measurements are not available then a comparative analysis can be provided.
- Comparative study can be done to different types of trajectories with more aggressive turns in directions involving the orientation $[\phi, \theta, \psi]$ and the position $[x, y, z]$.

LIST OF PUBLICATIONS

Published:

- W. Alqaisi, B. Brahmi, J. Ghommam, M. Saad, and V. Nerguizian, "Sliding mode controller and hierarchical perturbation compensator in a UAV quadrotor," in 2018 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA), 2018, pp. 1-6: IEEE.
- W. Alqaisi, B. Brahmi, J. Ghommam, M. Saad, and V. Nerguizian, "Multivariable super-twisting control in a vision-based quadrotor utilized in agricultural application," in 2018 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA), 2018, pp. 1-6: IEEE.
- W. Alqaisi, B. Brahmi, J. Ghommam, M. Saad, and V. Nerguizian, "Adaptive Control Based on RBF Neural Network Approximation in a Quadrotor", in IEEE International Symposium on Robotic and Sensors Environments (ROSE), 2019, pp. 1-6: IEEE.

Accepted:

- W. Alqaisi, B. Brahmi, J. Ghommam, M. Saad, and V. Nerguizian, "Vision based Leader Follower Approach for Uncertain Quadrotor Dynamics Using Feedback Linearization Sliding Mode Control (FLSMC)", International Journal of Modelling, Identification and Control, April 2019.
- W. Alqaisi, Y. Kali, J. Ghommam, M. Saad, and V. Nerguizian, "Position and Attitude tracking of Uncertain Quadrotor UAV based on New Terminal Super-Twisting Algorithm", in Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering, August 2019, pp. 1-12.
- Y. Kali, J. Rodas, M. Saad, R. Gregor, W. Alqaisi and K. Benjelloun. 2019. "Robust Finite-Time Position and Attitude Tracking of a Quadrotor UAV using Super-Twisting Control Algorithm with Linear Correction Terms", The 16th International Conference on Informatics in Control, Automation and Robotics: ICINCO 2019At: Prague, Czech Republic, July 2019.

Under review (submitted):

- W. Alqaisi, B. Brahmi, J. Ghommam, M. Saad, and V. Nerguizian, "Hierarchical Perturbation Compensation System with Exponential Reaching Law Sliding Mode Controller in a Quadrotor", in Journal of Control, Automation and Electrical Systems, 2019, pp. 1-10. (under review, submitted July 2019)

- W. Alqaisi, J. Ghommam, A. Alazzam, M. Saad, and V. Nerguizian, "Three-Loop Uncertainties Compensator and Sliding Mode Quadrotor UAV Control with Exponential Reaching Law", in *Journal of Computer and Electrical Engineering*, 2019, pp. 1-10. (under review, submitted April 2019)

APPENDIX I

PARROT “ROLLING-SPIDER” QUADROTOR

The Simulink Support Package for Parrot Rolling-Spider is a useful tool made to design and build flight control algorithms for Parrot minidrones. The software that supports this project was developed jointly by the Massachusetts Institute of Technology (MIT), and the Parrot Company. The Parrot “Rolling-Spider” quadrotor can access signals from three-axis accelerometers and three-axis gyroscope, pressure and ultrasonic sensors for altitude as well as a down-facing camera.

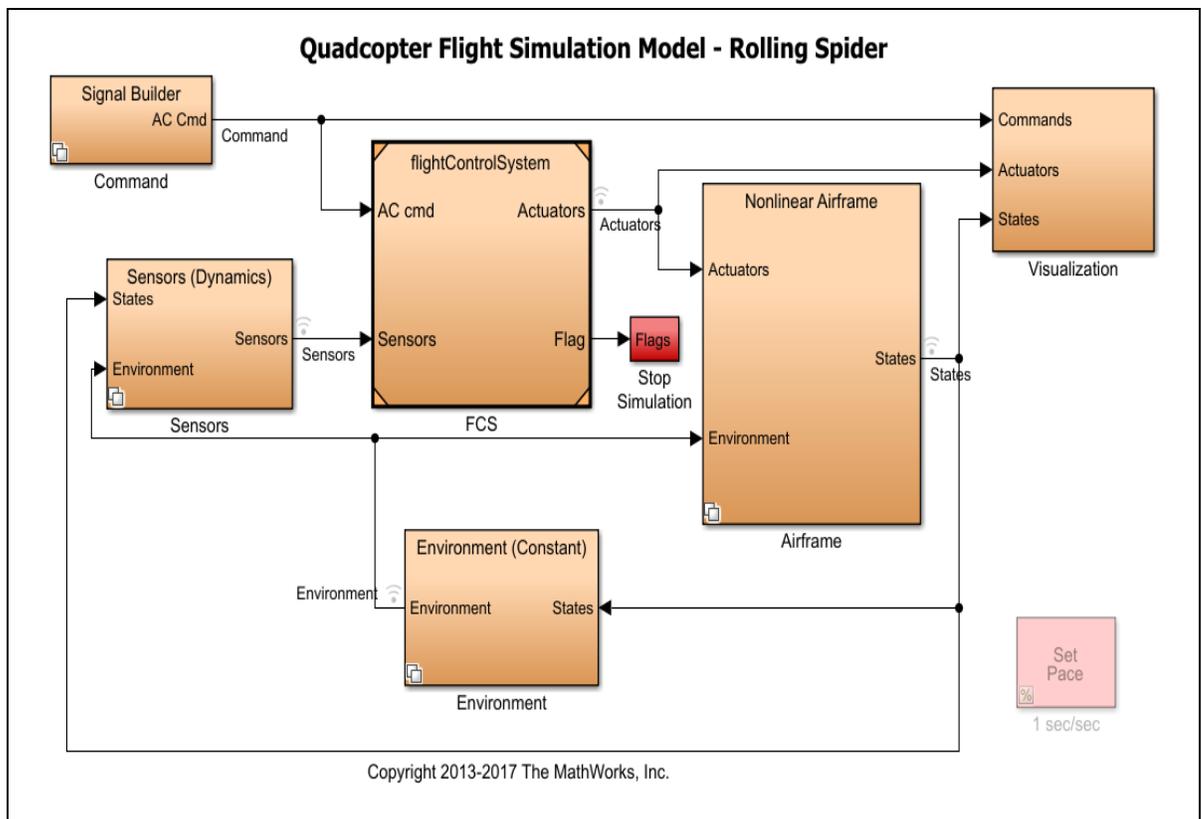


Figure-AI-1 Flight control model by Simulink

The support package includes a starting control model as in Figure-AI-1, which lets you model six degrees of freedom equations of motion and simulate aircraft behavior under various flights

and environmental conditions. The model consists of six blocks that contain mathematical representation of the dynamic system. There is mathematical representation of the airframe, environment and the sensors. The airframe block includes Euler angle representation of six degrees of freedom equations of motion. The developed control algorithm can be designed and built in Flight Control System (FCS) block. Input trajectory and results are obtained by the command block and the visualization block respectively (Technology, 2018). After building the desired control in FCS block, Simulink Coder is used to generate and executes the code from the Simulink model. The generated source code can be used for real-time and non real-time applications, including simulation acceleration, rapid prototyping, and hardware-in-the-loop testing. You can tune and monitor the generated code using Simulink or run and interact with the code outside MATLAB and Simulink. Simulink Coder lets you record flight data on the minidrone and access the code generated from Simulink models (Technology, 2018). The generated code then is deployed to the drone wirelessly by Bluetooth Smart technology V4.0 communication. The workflow of programing and testing is summarized in Figure-AI-2, and the quadrotor parameters are shown in Table-AI-1.

Table-AI-1 Parrot “rolling-spider” parameters

Parameter	Value
Mass (m)	= 0.068 [kg]
Moment of Inertia (I_x)	= 0.6860×10^{-3} [kg.m ²]
Moment of Inertia (I_y)	= 0.0920×10^{-3} [kg.m ²]
Moment of Inertia (I_z)	= 0.1366×10^{-3} [kg.m ²]
Motor inertia (j_r)	= 1.0209×10^{-7} [kg.m ²]
Gravity (g)	= 9.81 [m/s ²]

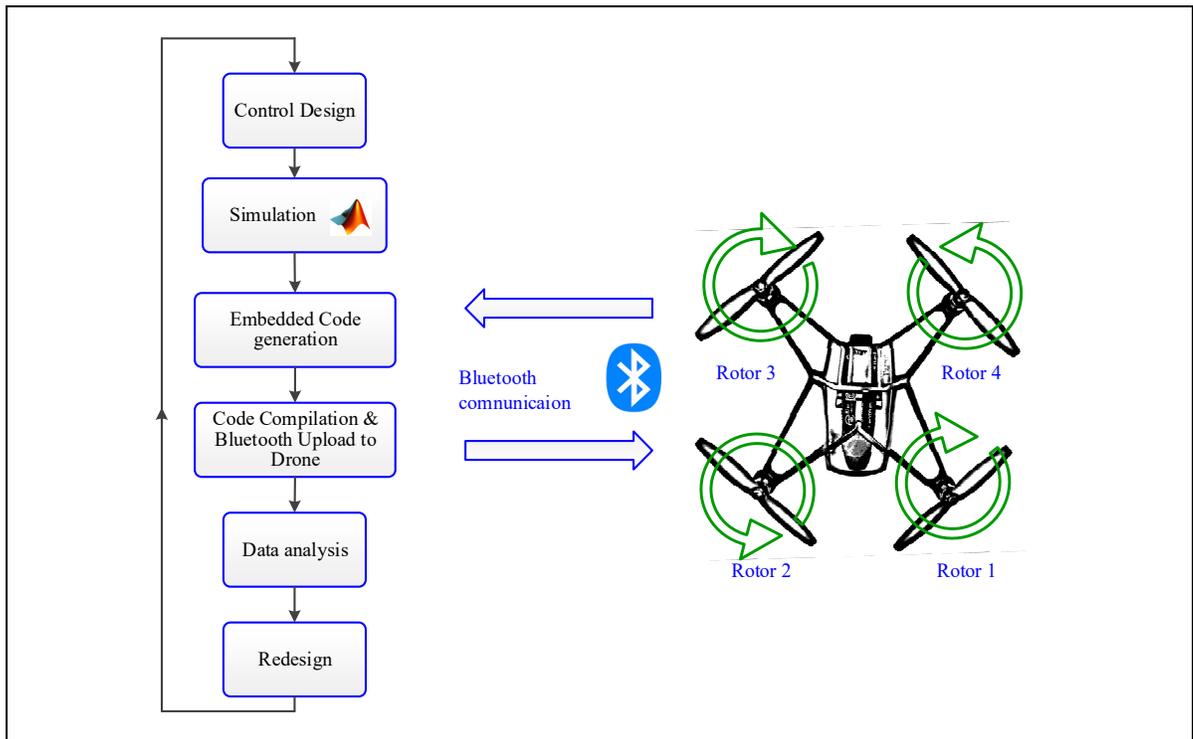


Figure-AI-2 Quadrotor-Simulink Implementation flow

The small size of Parrot rolling-spider drone makes it safe for flights in enclosed spaces. The option of adding protection wheels, protects the quadrotor in case of collision.

The four electric motors have the dimensions of 20mm height by 8mm diameter, with a 65mm diameter propeller per motor. The motors are 'coreless' type which makes their inertia low, results in a rapid response to sudden changes in acceleration. Furthermore, they have low current loss as iron losses disappear, which increases efficiency. They create low levels of electromagnetic field which reduces the problem of electromagnetic interference with other electronic devices.

Minidrone Setup and Configuration:

In this section, we introduce Parrot rolling-spider setup on window 7 (or higher) operating system. This is a brief description; more details are available online (Mathworks, 2018). In order to get started, the software/hardware requirements of the computer are as follows:

Software:

- Matlab R2017 or higher,
- Simulink,
- Simulink support package for Parrot minidrone (v.17.2.1 or higher),
- Aerospace block set, Simulink coder and Simulink 3D Animation,
- Simulink Coder.

Hardware:

- Parrot rolling-spider minidrone,
- Micro USB cable,
- Bluetooth low energy dongle,
- Battery charger.

Running the project goes through some main steps, description of each step is given as follows:

- 1) Switch ON the Parrot minidrone. The On/Off button is located under the minidrone, near the vertical camera. The LEDs glow or blink to indicate the On or Off status of the Parrot minidrone.
- 2) Connect the minidrone to a USB port of the computer using a micro USB type-B cable (Figure-AI-3-a) and wait for the LED indications to be stable. If the minidrone is recognized you see a confirmation message on the screen (Figure-AI-3-b).

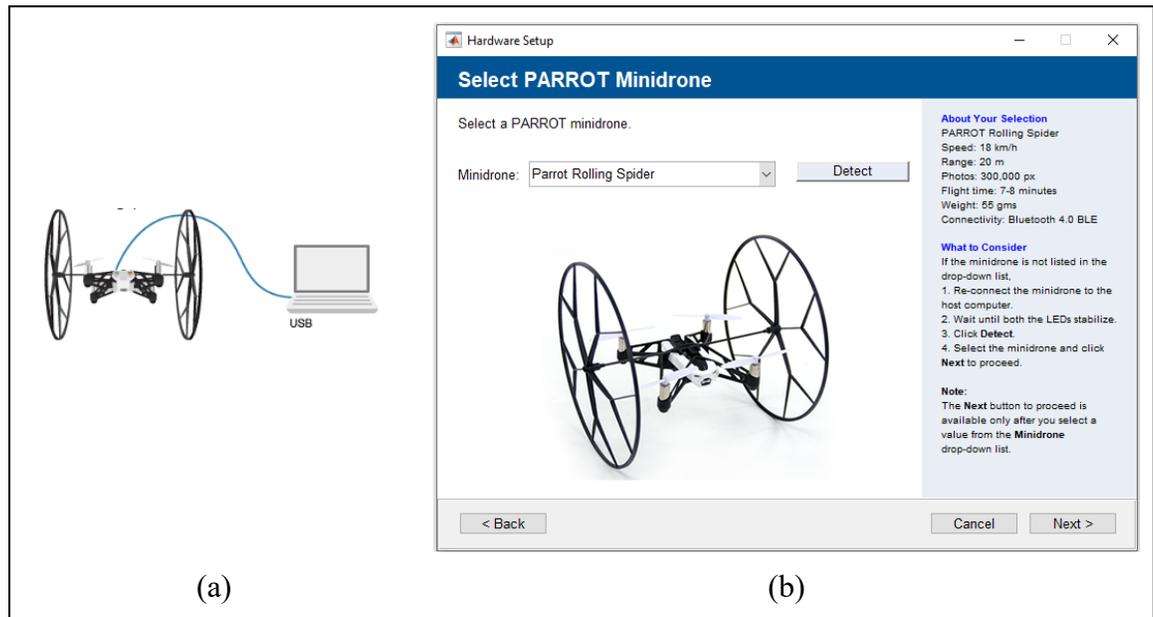


Figure-AI-3 (a) Parrot rolling-spider Connection, (b) Recognizing the quadrotor

- 3) Install Remote Network Driver Interface Specification (RNDIS) before working with the Simulink support package for Parrot Minidrones. During the entire process ensure that the minidrone is switched on and the battery is charged.
- 4) Replace or update the main firmware on the quadrotor.
- 5) Connect the quadrotor using Bluetooth on Windows; use an adapter that uses CSR bluetooth stack. For example, CSR 4.0 or Cinolink bluetooth 4.0 USB dongle adapters. These adapters must also support Personal Area Networking (GN). Switch on the drone. The On/Off button is located under the minidrone, near the vertical camera. Wait for both the LEDs to be green and stable. Insert your bluetooth 4.0 adapter into a USB port on your computer. Install the correct bluetooth driver for your adapter. After installing the driver, restart the computer. Turn on the bluetooth support on your computer. In this point, the configuration will be confirmed as seen in Figure-AI-4).

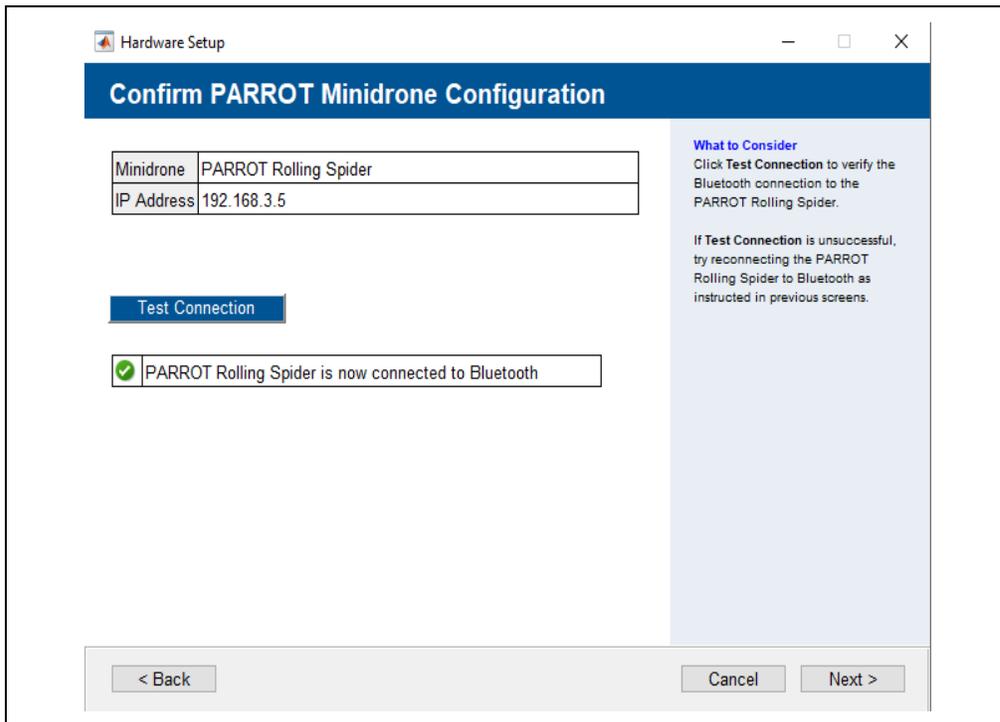


Figure-AI-4 Quadrotor-computer configuration confirmed

After the computer-quadrotor configuration is made, Simulink is now ready to deploy the code to the quadrotor:

- 1) Generate the code and deploy to the quadrotor using the flight control interface, click  on Simulink window. The model will be coded and deployed to the drone (Figure-AI-5). After coding and deploying goes through successfully, the flight can Start/Stop as we need (Figure-AI-6) also motor power can be set as required, in most applications it is should be set to the maximum. We set time for the model as the duration of flight is required for the minidrone.
- 2) The MAT-File for the signals logged in the model can be downloaded for data analysis. It is required to enable MAT-File logging in the model, click MAT File to download the MAT-File on your current MATLAB directory.

```
▼ Code Generation ⓘ 1
Elapsed: 50 sec

Downloading shared object to PARROT Rolling Spider ....
Terminating the currently executing shared object (if any) ....
Connecting to FTP Server 192.168.3.5 ....
Connected to FTP Server at 192.168.3.5 successfully.
Shared object 'librsedu.so' loaded successfully.
Initiating execution of shared object...
Execution of shared object initiated successfully.
Click here to launch the PARROT Flight Control Interface.
### Successful completion of build procedure for model: parrot_gettingstarted

Build process completed successfully
```

Figure-AI-5 Code generation and deploying

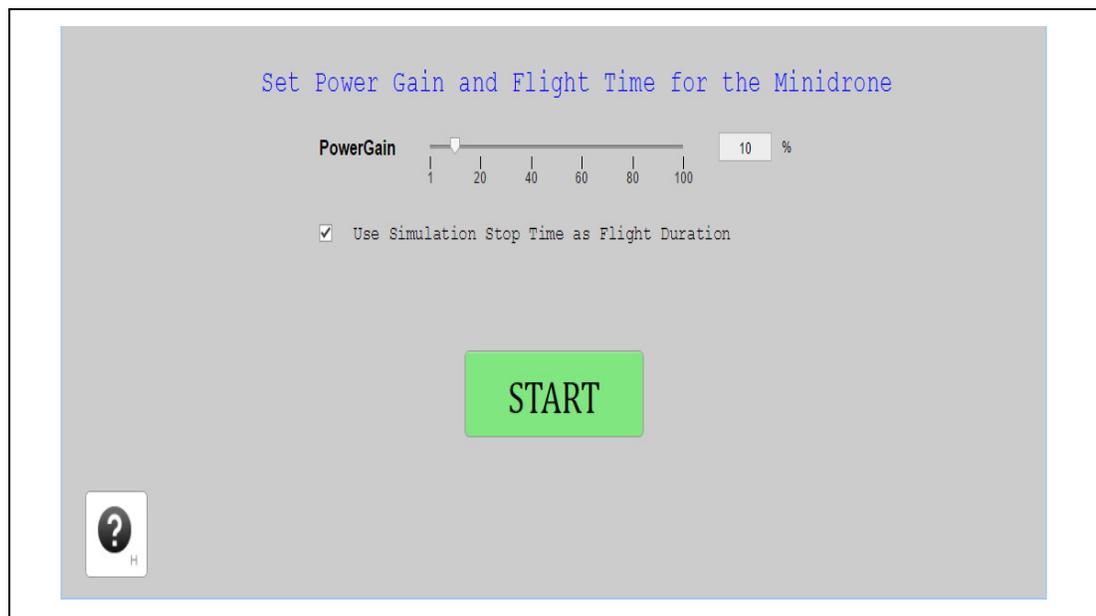


Figure-AI-6 Start/Stop and setting motor power.

Mathworks support package for Parrot (Mathworks, 2018) provides many useful examples start from beginners' level for new users and up to higher level of users. Such tool facilitates building more sophisticated control systems and many applications.

APPENDIX II

COMPARITIVE STUDY

A comparative study is performed in this thesis for the different proposed systems, the HPC, TLUC, FLSM and NTST. The target of this study is to compare the performance of each system with other systems. The trajectory is chosen as a circular shape with one-meter diameter where the desired height is one meter given by a smooth fifth-order polynomial. The four systems go under perturbation that is a continuous sinusoidal wave signal $\alpha = a \sin(\omega t)$, $a = 0.05 u_{i-max}$, where u_{i-max} is the maximum value of the control input, $\omega = 2\pi f$, $f = 1\text{Hz}$. In this method, it is required to have all the system work under same conditions. The objective of this study is to compare the performance of each system with other systems.

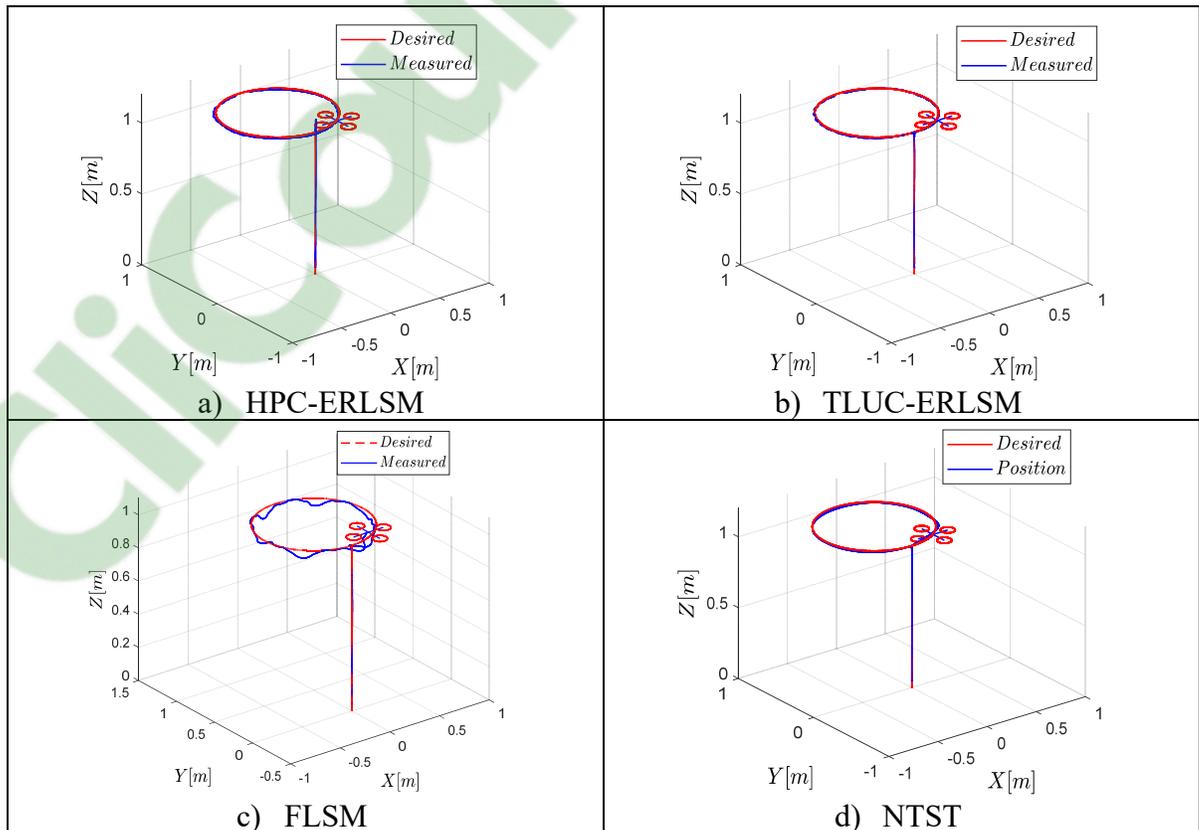


Figure-AII-1 Comparison in 3D trajectory

Figure AII-1 shows 3D trajectory of the proposed systems, it can be noticed that the all the systems have good response while the trajectory in the FLSM suffers from some distortion. In the errors figure (Figure-AII-2), it can be noticed that the TLUC give the smallest errors.

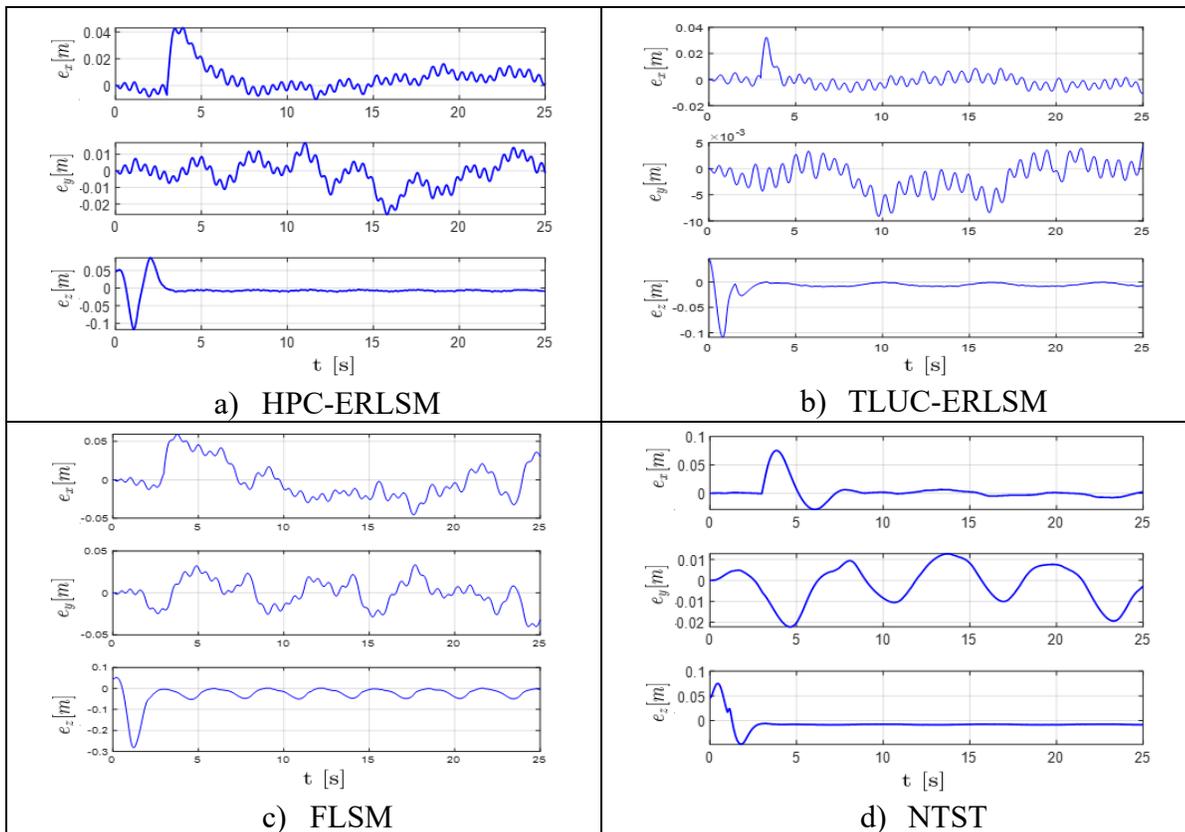


Figure-AII-2 Error signals

As a conclusion, the aforementioned systems went under unexpected and high disturbance. In this comparative study it can be seen that the TLUC give the best performance while the lowest performance is in the FLSM. The reason is that the FLSM, for example, is designed and tuned to work under normal conditions, it lacks estimation and compensation of disturbance. On the other hand, the TLUC is designed to estimate, compensate and track perturbation and residual perturbation in three loops. Thus, it can attenuate perturbation to a very small values.

LIST OF BIBLIOGRAPHICAL REFERENCES

- Abbasi, E., Ghayour, M., & Danesh, M. (2017). *Virtual Leader-Follower Formation Control of Multi Quadrotors by using Feedback Linearization Controller*. Paper presented at the 2017 5th RSI International Conference on Robotics and Mechatronics (ICRoM).
- Al-Hiddabi, S. A. (2009). *Quadrotor control using feedback linearization with dynamic extension*. Paper presented at the 2009 6th International Symposium on Mechatronics and its Applications.
- Alexis, K., Nikolakopoulos, G., & Tzes, A. (2012). Model predictive quadrotor control: attitude, altitude and position experimental studies. *IET Control Theory & Applications*, 6(12), 1812-1827.
- Ali Dehghani, M., & Bagher Menhaj, M. (2016). Takagi-Sugeno system for supervisory formation control of seeker mounted unmanned aerial vehicles. *Assembly Automation*, 36(2), 111-119.
- Ali, S. U., Samar, R., Shah, M. Z., Bhatti, A. I., & Munawar, K. (2017). Higher-order sliding mode based lateral guidance for unmanned aerial vehicles. *Transactions of the Institute of Measurement and Control*, 39(5), 715-727.
- Alqaisi, W., Brahmi, B., Ghommam, J., Saad, M., & Nerguizian, V. (2018a). *Multivariable super-twisting control in a vision-based quadrotor utilized in agricultural application*. Paper presented at the 2018 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA).
- Alqaisi, W., Brahmi, B., Ghommam, J., Saad, M., & Nerguizian, V. (2018b). *Sliding mode controller and hierarchical perturbation compensator in a UAV quadrotor*. Paper presented at the 2018 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA).
- Amin, R., Aijun, L., & Shamshirband, S. (2016). A review of quadrotor UAV: control methodologies and performance evaluation. *International Journal of Automation and Control*, 10(2), 87-103.
- Antonelli, G., Cataldi, E., Arrichiello, F., Giordano, P. R., Chiaverini, S., & Franchi, A. (2017). Adaptive trajectory tracking for quadrotor MAVs in presence of parameter uncertainties and external disturbances. *IEEE Transactions on Control Systems Technology*.

- Azar, A. T., Serrano, F. E., Vaidyanathan, S., & Albalawi, H. (2019). *Adaptive Higher Order Sliding Mode Control for Robotic Manipulators with Matched and Mismatched Uncertainties*. Paper presented at the International Conference on Advanced Machine Learning Technologies and Applications.
- Barikbin, B., & Fakharian, A. (2019). Trajectory tracking for quadrotor UAV transporting cable-suspended payload in wind presence. *Transactions of the Institute of Measurement and Control*, 0142331218774606.
- Bartolini, G., Pisano, A., Punta, E., & Usai, E. (2003). A survey of applications of second-order sliding mode control to mechanical systems. *International journal of control*, 76(9-10), 875-892.
- Behal, A., Dixon, W., Dawson, D. M., & Xian, B. (2009). *Lyapunov-based control of robotic systems* (Vol. 36): CRC Press.
- Behal, A., Dixon, W., Xian, B., & Dawson, D. M. (2009). *Lyapunov-based control of robotic systems*: CRC Press.
- Benallegue, A., Mokhtari, A., & Fridman, L. (2008). High-order sliding-mode observer for a quadrotor UAV. *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal*, 18(4-5), 427-440.
- Benderradji, H., Benamor, A., Chrifi-Alaoui, L., Bussy, P., & Makouf, A. (2012). *Second order sliding mode induction motor control with a new Lyapunov approach*. Paper presented at the International Multi-Conference on Systems, Signals & Devices.
- Besnard, L., Shtessel, Y. B., & Landrum, B. (2007). *Control of a quadrotor vehicle using sliding mode disturbance observer*. Paper presented at the 2007 American Control Conference.
- Besnard, L., Shtessel, Y. B., & Landrum, B. (2007). *Control of a quadrotor vehicle using sliding mode disturbance observer*. Paper presented at the American Control Conference, 2007. ACC'07.
- Besnard, L., Shtessel, Y. B., & Landrum, B. (2012). Quadrotor vehicle control via sliding mode controller driven by sliding mode disturbance observer. *Journal of the Franklin Institute*, 349(2), 658-684.
- Boiko, I., & Fridman, L. (2005). Analysis of chattering in continuous sliding-mode controllers. *IEEE Transactions on Automatic Control*, 50(9), 1442-1446.
- Boiko, I., Fridman, L., & Iriarte, R. (2005). *Analysis of chattering in continuous sliding mode control*. Paper presented at the American Control Conference, 2005. Proceedings of the 2005.

- Bouabdallah, S. (2007a). *Design and control of quadrotors with application to autonomous flying*. Ecole Polytechnique Federale de Lausanne,
- Bouabdallah, S. (2007b). *Design and control of quadrotors with application to autonomous flying*. Retrieved from
- Bouabdallah, S., & Siegwart, R. (2005). *Backstepping and sliding-mode techniques applied to an indoor micro quadrotor*. Paper presented at the None.
- Bouabdallah, S., & Siegwart, R. (2007, Oct. 29 2007-Nov. 2 2007). *Full control of a quadrotor*. Paper presented at the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems.
- Bouadi, H., Bouchoucha, M., & Tadjine, M. (2007a). Modelling and stabilizing control laws design based on backstepping for an UAV type-quadrotor. *IFAC Proceedings Volumes*, 40(15), 245-250.
- Bouadi, H., Bouchoucha, M., & Tadjine, M. (2007b). Sliding mode control based on backstepping approach for an UAV type-quadrotor. *World Academy of Science, Engineering and Technology*, 26(5), 22-27.
- Bresciani, T. (2008). Modelling, identification and control of a quadrotor helicopter. *MSc Theses*.
- Cabecinhas, D., Cunha, R., & Silvestre, C. (2014). A globally stabilizing path following controller for rotorcraft with wind disturbance rejection. *IEEE Transactions on Control Systems Technology*, 23(2), 708-714.
- Cabecinhas, D., Cunha, R., & Silvestre, C. (2015). A globally stabilizing path following controller for rotorcraft with wind disturbance rejection. *IEEE Transactions on Control Systems Technology*, 23(2), 708-714.
- Chang, K., Xia, Y., Huang, K., & Ma, D. (2016). Obstacle avoidance and active disturbance rejection control for a quadrotor. *Neurocomputing*, 190, 60-69.
- Chen, F., Lei, W., Zhang, K., Tao, G., & Jiang, B. (2016a). A novel nonlinear resilient control for a quadrotor UAV via backstepping control and nonlinear disturbance observer. *Nonlinear Dynamics*, 1-15.
- Chen, F., Lei, W., Zhang, K., Tao, G., & Jiang, B. (2016b). A novel nonlinear resilient control for a quadrotor UAV via backstepping control and nonlinear disturbance observer. *Nonlinear Dynamics*, 85(2), 1281-1295.

- Chen, F., Lu, F., Jiang, B., & Tao, G. (2014). Adaptive compensation control of the quadrotor helicopter using quantum information technology and disturbance observer. *Journal of the Franklin Institute*, 351(1), 442-455.
- Chen, H., Chen, H., & Xu, P. (2019). *Global Fast Terminal Sliding Mode Control Law Design of a Quadrotor*. Paper presented at the 2019 International Conference on Computer, Network, Communication and Information Systems (CNCI 2019).
- Chen, H., Song, S., & Li, X. (2019). Robust spacecraft attitude tracking control with integral terminal sliding mode surface considering input saturation. *Transactions of the Institute of Measurement and Control*, 41(2), 405-416.
- Chen, P., Chen, C.-W., & Chiang, W.-L. (2009). GA-based modified adaptive fuzzy sliding mode controller for nonlinear systems. *Expert Systems with Applications*, 36(3), 5872-5879.
- Dávila, A., Moreno, J. A., & Fridman, L. (2010). *Variable gains super-twisting algorithm: A Lyapunov based design*. Paper presented at the American Control Conference, Baltimore, MD, June.
- Dehghani, M. A., & Menhaj, M. B. (2016). Communication free leader–follower formation control of unmanned aircraft systems. *Robotics and Autonomous Systems*, 80, 69-75.
- Derafa, L., Benallegue, A., & Fridman, L. (2012). Super twisting control algorithm for the attitude tracking of a four rotors UAV. *Journal of the Franklin Institute*, 349(2), 685-699.
- Emran, B. J. (2014). *Nonlinear Adaptive Control of Quadrotor*. (Master of Science), American university of Sharjah, AUS DSpace. Retrieved from <http://hdl.handle.net/11073/6059>.
- Erginer, B., & Altuğ, E. (2007). *Modeling and PD control of a quadrotor VTOL vehicle*. Paper presented at the Intelligent Vehicles Symposium, 2007 IEEE.
- Evangelista, C., Puleston, P., Valenciaga, F., & Fridman, L. M. (2012). Lyapunov-designed super-twisting sliding mode control for wind energy conversion optimization. *IEEE Transactions on Industrial Electronics*, 60(2), 538-545.
- Fallaha, C., & Saad, M. (2018). Model-based sliding functions design for sliding mode robot control. *International Journal of Modelling, Identification and Control*, 30(1), 48-60.
- Fallaha, C. J., Saad, M., Kanaan, H. Y., & Al-Haddad, K. (2010). Sliding-mode robot control with exponential reaching law. *IEEE Transactions on Industrial Electronics*, 58(2), 600-610.

- Fallaha, C. J., Saad, M., Kanaan, H. Y., & Al-Haddad, K. (2011a). Sliding-mode robot control with exponential reaching law. *IEEE Transactions on Industrial Electronics*, 58(2), 600-610.
- Feng, Y., Yu, X., & Han, F. (2013). On nonsingular terminal sliding-mode control of nonlinear systems. *Automatica*, 49(6), 1715-1722.
- Feng, Y., Yu, X., & Man, Z. (2002). Non-singular terminal sliding mode control of rigid manipulators. *Automatica*, 38(12), 2159-2167.
- Fridman, L. (1999). The problem of chattering: an averaging approach. *Variable structure systems, sliding mode and nonlinear control*, 363-386.
- Fridman, L. M. (2001). An averaging approach to chattering. *IEEE Transactions on Automatic Control*, 46(8), 1260-1265.
- Garcia, P. C., Lozano, R., & Dzul, A. E. (2006). *Modelling and control of mini-flying machines*: Springer Science & Business Media.
- Gupta, N., & Kothari, M. (2017). Modeling and Control of Inverted Flight of a Variable-Pitch Quadrotor. *arXiv preprint arXiv:1709.06407*.
- Guzmán, E., & Moreno, J. A. (2015). Super-twisting observer for second-order systems with time-varying coefficient. *IET Control Theory & Applications*, 9(4), 553-562.
- Hicham, K. (2012). *Tolérance aux défauts via la méthode backstepping des systèmes non linéaires application: Système uav de type quadrirotor*. Université Ferhat Abbas de Sétif 1,
- Hwangbo, J., Sa, I., Siegwart, R., & Hutter, M. (2017). Control of a quadrotor with reinforcement learning. *IEEE Robotics and Automation Letters*, 2(4), 2096-2103.
- Jasim, W., & Gu, D. (2018). Robust path tracking control for quadrotors with experimental validation. *International Journal of Modelling, Identification and Control*, 29(1), 1-13.
- Jeong, S. H., Jung, S., & Tomizuka, M. (2012a). *Attitude control of a quad-rotor system using an acceleration-based disturbance observer: An empirical approach*. Paper presented at the 2012 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM).
- Jeong, S. H., Jung, S., & Tomizuka, M. (2012b). *Attitude control of a quad-rotor system using an acceleration-based disturbance observer: An empirical approach*. Paper presented at the Advanced Intelligent Mechatronics (AIM), 2012 IEEE/ASME International Conference on.

- Jia, Z., Yu, J., Mei, Y., Chen, Y., Shen, Y., & Ai, X. (2017). Integral backstepping sliding mode control for quadrotor helicopter under external uncertain disturbances. *Aerospace Science and Technology*, 68, 299-307.
- Jiang, T., Lin, D., & Song, T. (2018). Finite-time backstepping control for quadrotors with disturbances and input constraints. *IEEE Access*, 6, 62037-62049.
- Jin, M., Lee, J., & Ahn, K. K. (2015). Continuous nonsingular terminal sliding-mode control of shape memory alloy actuators using time delay estimation. *IEEE/ASME Transactions on Mechatronics*, 20(2), 899-909.
- Kali, Y., Rodas, J., Gregor, R., Saad, M., & Benjelloun, K. (2018). *Attitude tracking of a tri-rotor UAV based on robust sliding mode with time delay estimation*. Paper presented at the 2018 International Conference on Unmanned Aircraft Systems (ICUAS).
- Kali, Y., Rodas, J., Saad, M., Gregor, R., Benjelloun, K., & Doval-Gandoy, J. (2017). *Current control based on super-twisting algorithm with time delay estimation for a five-phase induction motor drive*. Paper presented at the 2017 IEEE International Electric Machines and Drives Conference (IEMDC).
- Kali, Y., Saad, M., & Benjelloun, K. (2018). Optimal super-twisting algorithm with time delay estimation for robot manipulators based on feedback linearization. *Robotics and Autonomous Systems*, 108, 87-99.
- Kali, Y., Saad, M., & Benjelloun, K. (2019). Control of robot manipulators using modified backstepping sliding mode. In *New Developments and Advances in Robot Control* (pp. 107-136): Springer.
- Kali, Y., Saad, M., Benjelloun, K., & Fatemi, A. (2017). Discrete-time second order sliding mode with time delay control for uncertain robot manipulators. *Robotics and Autonomous Systems*, 94, 53-60.
- Kali, Y., Saad, M., Benjelloun, K., & Khairallah, C. (2018). Super-twisting algorithm with time delay estimation for uncertain robot manipulators. *Nonlinear Dynamics*, 1-13.
- Kamal, S., Chalanga, A., Bera, M. K., & Bandyopadhyay, B. (2012). *State estimation and non vanishing unmatched disturbance reconstruction using modified super twisting algorithm*. Paper presented at the Electrical & Computer Engineering (ICECE), 2012 7th International Conference on.
- Kaminer, I. I., Ghabchelo, R., Dobrokhodov, V. N., & Jones, K. D. (2011). Vision-Based Tracking and Motion Estimation for Moving Targets Using Small UAVs.

- Kayacan, E., & Maslim, R. (2016). Type-2 fuzzy logic trajectory tracking control of quadrotor VTOL aircraft with elliptic membership functions. *IEEE/ASME Transactions on Mechatronics*, 22(1), 339-348.
- Kayacan, E., & Maslim, R. (2017). Type-2 fuzzy logic trajectory tracking control of quadrotor VTOL aircraft with elliptic membership functions. *IEEE/ASME Transactions on Mechatronics*, 22(1), 339-348.
- Khebbache, H. (2018). *Tolérance aux défauts via la méthode backstepping des systèmes non linéaires: application système UAV de type quadrirotor*.
- Kim, J., Joe, H., Yu, S.-c., Lee, J. S., & Kim, M. (2016). Time-delay controller design for position control of autonomous underwater vehicle under disturbances. *IEEE Transactions on Industrial Electronics*, 63(2), 1052-1061.
- Krim, Y., Abbes, D., Krim, S., & Mimouni, M. F. (2018). Classical vector, first-order sliding-mode and high-order sliding-mode control for a grid-connected variable-speed wind energy conversion system: A comparative study. *Wind Engineering*, 42(1), 16-37.
- Kurode, S., & Dixit, P. (2013). Sliding mode control of flexible link manipulator using states and disturbance estimation. *International Journal of Advanced Mechatronic Systems*, 5(2), 129-137.
- Kwon, S., & Chung, W. K. (2004). *Perturbation compensator based robust tracking control and state estimation of mechanical systems* (Vol. 307): Springer Science & Business Media.
- Kwon, S. J., & Chung, W. K. (2004). Robust tracking control with hierarchical perturbation compensation. *Perturbation Compensator based Robust Tracking Control and State Estimation of Mechanical Systems*, 11-39.
- Levant, A. (1993). Sliding order and sliding accuracy in sliding mode control. *International journal of control*, 58(6), 1247-1263.
- Levant, A. (2003). Higher-order sliding modes, differentiation and output-feedback control. *International journal of Control*, 76(9-10), 924-941.
- Li, X., & Xiao, J. (2005). Robot formation control in leader-follower motion using direct Lyapunov method. *International Journal of Intelligent Control and Systems*, 10(3), 244-250.
- Liu, H., Zhao, W., Zuo, Z., & Zhong, Y. (2016). Robust control for quadrotors with multiple time-varying uncertainties and delays. *IEEE Transactions on Industrial Electronics*, 64(2), 1303-1312.

- Liu, H., Zhao, W., Zuo, Z., & Zhong, Y. (2017). Robust control for quadrotors with multiple time-varying uncertainties and delays. *IEEE Transactions on Industrial Electronics*, 64(2), 1303-1312.
- Liu, J., & Wang, X. (2012b). *Advanced Sliding Mode Control for Mechanical Systems: Design, Analysis and MATLAB Simulation*: Springer.
- Loria, A., Dasdemir, J., & Jarquin, N. A. (2016). Leader–follower formation and tracking control of mobile robots along straight paths. *IEEE Transactions on Control Systems Technology*, 24(2), 727-732.
- Marino, R., & Tomei, P. (2016a). Adaptive disturbance rejection for unknown stable linear systems. *Transactions of the Institute of Measurement and Control*, 0142331215625769.
- Marino, R., & Tomei, P. (2016b). Adaptive disturbance rejection for unknown stable linear systems. *Transactions of the Institute of Measurement and Control*, 38(6), 640-647.
- Mathworks. (2018). PARROT Minidrones Support from Simulink. Retrieved from <https://www.mathworks.com/hardware-support/parrot-minidrones.html>.
- Merabet, A., Labib, L., Ghias, A. M., Aldurra, A., & Debbouza, M. (2019). Dual-mode operation based second-order sliding mode control for grid-connected solar photovoltaic energy system. *International Journal of Electrical Power & Energy Systems*, 111, 459-474.
- Mercado, D., Castro, R., & Lozano, R. (2013). *Quadrotors flight formation control using a leader-follower approach*. Paper presented at the Control Conference (ECC), 2013 European.
- Mohd Basri, M. A., Husain, A. R., & Danapalasingam, K. A. (2015). Intelligent adaptive backstepping control for MIMO uncertain non-linear quadrotor helicopter systems. *Transactions of the Institute of Measurement and Control*, 37(3), 345-361.
- Mokhtari, A., M'Sirdi, N. K., Meghriche, K., & Belaidi, A. (2006). Feedback linearization and linear observer for a quadrotor unmanned aerial vehicle. *Advanced Robotics*, 20(1), 71-91.
- Moreno, J. A. (2014). On strict Lyapunov functions for some non-homogeneous super-twisting algorithms. *Journal of the Franklin Institute*, 351(4), 1902-1919.
- Moreno, J. A., & Osorio, M. (2008). *A Lyapunov approach to second-order sliding mode controllers and observers*. Paper presented at the Decision and Control, 2008. CDC 2008. 47th IEEE Conference on.

- Mu, B., Zhang, K., & Shi, Y. (2017). Integral sliding mode flight controller design for a quadrotor and the application in a heterogeneous multi-agent system. *IEEE Transactions on Industrial Electronics*, 64(12), 9389-9398.
- Mu, J., Yan, X.-G., Spurgeon, S. K., & Mao, Z. (2017). Nonlinear sliding mode control of a two-wheeled mobile robot system. *International Journal of Modelling, Identification and Control*, 27(2), 75-83.
- Murray, R. M. (2017). *A mathematical introduction to robotic manipulation* (1st Edition ed.): Taylor and Francis Group.
- Nagaty, A., Saeedi, S., Thibault, C., Seto, M., & Li, H. (2013). Control and navigation framework for quadrotor helicopters. *Journal of intelligent & robotic systems*, 70(1-4), 1-12.
- Parrot Minidrone. (2018). Retrieved from <http://abihuynh.com/nova-scotia/parrot-minidrone-rolling-spider-manual.php#>).
- Pieter-Jan. (2013). Reading a IMU Without Kalman: The Complementary Filter. Retrieved from <http://www.pieter-jan.com/node/11>.
- Powers, C., Mellinger, D., & Kumar, V. (2015). Quadrotor kinematics and dynamics. *Handbook of Unmanned Aerial Vehicles*, 307-328.
- Rafiq, M., Rehman, S.-u., Rehman, F.-u., Butt, Q. R., & Awan, I. (2012). A second order sliding mode control design of a switched reluctance motor using super twisting algorithm. *Simulation Modelling Practice and Theory*, 25, 106-117.
- Ranjbar, M. A., Ghasemi, R., & Akramizadeh, A. (2018). Time-varying leader-following consensus of high order multi-agent systems. *International Journal of Modelling, Identification and Control*, 30(4), 333-341.
- Razmi, H., & Afshinfar, S. (2019). Neural network-based adaptive sliding mode control design for position and attitude control of a quadrotor UAV. *Aerospace Science and Technology*, 91, 12-27.
- Richter, C., Bry, A., & Roy, N. (2016). Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments. In *Robotics Research* (pp. 649-666): Springer.
- Sanz, R., Garcia, P., Zhong, Q.-C., & Albertos, P. (2016). Robust control of quadrotors based on an uncertainty and disturbance estimator. *Journal of Dynamic Systems, Measurement, and Control*, 138(7), 071006.

- Sequeira, G. (2007). *Vision based leader-follower formation control for mobile robots*. (M.S. in Electrical Engineering), Missouri University of Science and Technology, University of Missouri--Rolla.
- Singh, P., Goyal, V., Deolia, V. K., & Sharma, T. N. (2017). Sliding Mode Control of Uncertain Nonlinear Discrete Delayed Time System Using Chebyshev Neural Network. In *Advances in Computer and Computational Sciences* (pp. 527-540): Springer.
- Sivaramakrishnan, N., Hemavathy, P., & Anitha, G. (2017). Design of Hybrid control for Isothermal Continuous stirred tank Reactor. *International Journal of Pure and Applied Mathematics*, 117(20), 999-1009.
- Slotine, J.-J. E., & Li, W. (1991). *Applied nonlinear control* (Vol. 199): Prentice hall Englewood Cliffs, NJ.
- Technology, M. I. o. (2018). \Matlab toolbox for parrot rolling spider." [Online]. Available: <http://karaman.mit.edu/software.html>.
- Ton, C. T., McCourt, M., & Mehta, S. S. (2016). *Robust tracking control of a quadrotor with time-varying gain in the presence of uncertainty and disturbances*. Paper presented at the AIAA Guidance, Navigation, and Control Conference.
- Uebe, G. (2008). JE Gentle: Matrix Algebra—Theory, Computations, and Applications in Statistics. *AStA Advances in Statistical Analysis*, 92(3), 343-344.
- Utkin, V., Guldner, J., & Shi, J. (2009). *Sliding mode control in electro-mechanical systems*: CRC press.
- Utkin, V., & Lee, H. (2006). *Chattering problem in sliding mode control systems*. Paper presented at the International Workshop on Variable Structure Systems, 2006. VSS'06.
- Van de Maele, P.-J. Reading a IMU Without Kalman: The Complementary Filter. 2013. *Pieter-Jan Van de Maele. Kättesaadav: <http://www.pieterjan.com/node/11> (02.02. 2014)*.
- Voos, H. (2009). *Nonlinear control of a quadrotor micro-UAV using feedback-linearization*. Paper presented at the 2009 IEEE International Conference on Mechatronics.
- Wang, J., Liu, J.-y., & Yi, H. (2015). *Formation control of unmanned surface vehicles with vision sensor constraints*. Paper presented at the OCEANS'15 MTS/IEEE Washington.
- Wang, J., Xin, S., & Zhang, Y. (2017). Modeling and Control of a Quadrotor Vehicle Subject to Disturbance Load.

- Wang, L., Chai, T., & Zhai, L. (2009). Neural-network-based terminal sliding-mode control of robotic manipulators including actuator dynamics. *IEEE Transactions on Industrial Electronics*, 56(9), 3296-3304.
- Wang, X., Shirinzadeh, B., & Ang, M. H. (2014). Nonlinear double-integral observer and application to quadrotor aircraft. *IEEE Transactions on Industrial Electronics*, 62(2), 1189-1200.
- Wang, X., Shirinzadeh, B., & Ang, M. H. (2015). Nonlinear double-integral observer and application to quadrotor aircraft. *IEEE Transactions on Industrial Electronics*, 62(2), 1189-1200.
- Wang, Y., Li, B., Yan, F., & Chen, B. (2019). Practical adaptive fractional-order nonsingular terminal sliding mode control for a cable-driven manipulator. *International Journal of Robust and Nonlinear Control*, 29(5), 1396-1417.
- Wu, D., Du, H., & Zhu, W. (2017). *Finite-time position tracking control of a quadrotor aircraft*. Paper presented at the 2017 36th Chinese Control Conference (CCC).
- Xia, Q., Zhu, J., & Qi, Z. (2010). A composite sliding mode autopilot design. *International Journal of Modelling, Identification and Control*, 9(1-2), 37-42.
- Xiangjian, C., Kun, S., & Di, L. (2016). H infinity control design for Eight-Rotor MAV attitude system based on identification by interval type II fuzzy neural network. *International Journal of Aeronautical and Space Sciences*, 17(2), 195-203.
- Yang, H., Cheng, L., Xia, Y., & Yuan, Y. (2017). Active Disturbance Rejection Attitude Control for a Dual Closed-Loop Quadrotor Under Gust Wind. *IEEE Transactions on Control Systems Technology*.
- Yin, S., & Xiao, B. (2016). Tracking control of surface ships with disturbance and uncertainties rejection capability. *IEEE/ASME Transactions on Mechatronics*, 22(3), 1154-1162.
- Yin, S., & Xiao, B. (2017). Tracking control of surface ships with disturbance and uncertainties rejection capability. *IEEE/ASME Transactions on Mechatronics*, 22(3), 1154-1162.
- Yoshimura, T. (2008). Adaptive sliding mode control for a class of mismatched uncertain systems. *International Journal of Modelling, Identification and Control*, 5(2), 154-165.
- Youcef-Toumi, K., & Ito, O. (1988). *A time delay controller for systems with unknown dynamics*. Paper presented at the American Control Conference, 1988.
- Zhang, R., Quan, Q., & Cai, K.-Y. (2011). Attitude control of a quadrotor aircraft subject to a class of time-varying disturbances. *Control Theory & Applications, IET*, 5(9), 1140-1146.

- Zhang, R., Quan, Q., & Cai, K.-Y. (2011). Attitude control of a quadrotor aircraft subject to a class of time-varying disturbances. *IET Control Theory & Applications*, 5(9), 1140-1146.
- Zhang, R., Sun, C., Zhang, J., & Zhou, Y. (2013). Second-order terminal sliding mode control for hypersonic vehicle in cruising flight with sliding mode disturbance observer. *Journal of control theory and applications*, 11(2), 299-305.
- Zhao, B., Xian, B., Zhang, Y., & Zhang, X. (2015). Nonlinear robust adaptive tracking control of a quadrotor UAV via immersion and invariance methodology. *IEEE Transactions on Industrial Electronics*, 62(5), 2891-2902.
- Zheng, G., & Xian, B. (2018). Nonlinear robust control of a quadrotor helicopter with finite time convergence. *Control Theory and Technology*, 16(2), 133-144.
- Zhihong, M., & Yu, X. H. (1996). *Terminal sliding mode control of MIMO linear systems*. Paper presented at the Proceedings of 35th IEEE conference on decision and control.
- Zhou, J., Deng, R., Shi, Z., & Zhong, Y. (2017). *Robust cascade PID attitude control of quadrotor helicopters subject to wind disturbance*. Paper presented at the Control Conference (CCC), 2017 36th Chinese.
- Zipfel, P. H. (2007). *Modeling and simulation of aerospace vehicle dynamics*: American Institute of Aeronautics and Astronautics.