# Sliding mode based control of an unmanned aerial manipulator

Hardy Azmir Bin Anuar

COLLEGE SCIENCES DOCTORAL DE L'INGENIERIE PAYS DE LA LOIRE ET DES SYSTEMES

CENTRALE NANTES

# MEMOIRE DE DOCTORAT DE

L'ÉCOLE CENTRALE DE NANTES

ÉCOLE DOCTORALE N° 602
*Sciences de l'Ingénierie et des Systèmes*
Spécialité : *Robotique*

Par

**Hardy Azmir BIN ANUAR**

# Sliding mode based control of an unmanned aerial manipulator

**Projet de recherche doctoral présenté et soutenu à l'École Centrale de Nantes, le 4 décembre 2023**
**Unité de recherche : UMR 6004, Laboratoire des Sciences du Numérique de Nantes (LS2N)**

**Rapporteurs avant soutenance :**

| | |
|---|---|
| Xavier BRUN | Professeur des Universités, INSA de Lyon |
| Patrick COIRAULT | Professeur des Universités, Université de Poitiers |

**Composition du Jury :**

| | | |
|---|---|---|
| Président : | Xavier BRUN | Professeur des Universités, INSA de Lyon |
| Examinateurs : | Patrick COIRAULT | Professeur des Universités, Université de Poitiers |
| | Claire DUNE | Maître de Conférences, Université de Toulon |
| Directeur de recherches doctorales : | Franck PLESTAN | Professeur des Universités, École Centrale de Nantes |
| Co-enc. de recherches doctorales : | Abdelhamid CHRIETTE | Maître de Conférences, École Centrale de Nantes |
| Co-enc. de recherches doctorales : | Olivier KERMORGANT | Maître de Conférences, École Centrale de Nantes |

**Invité :**
Khairol Amali BIN AHMAD Professor, National Defence University of Malaysia

# ACKNOWLEDGEMENT

All praises to the Most Gracious, the Most Merciful for His willingness to giving me this opportunity and thus letting me complete this PhD journey with His blessings, guidance, merciful and incomparable love.

I am particularly thankful to Professor Dr. Franck Plestan, Dean of Research at Ecole Centrale de Nantes for giving me the opportunity to work with him at Laboratoire des Sciences du Numerique de Nantes. Special appreciation to my co-supervisors Associate Professor Dr. Abdelhamid Chriette and Associate Professor Dr. Olivier Kermorgant with their support and never ending guidance from the very first day of my enrolment. Their scientific advice, patience, determination, motivations and confidence in my work are never to be forgotten. Thanks to them, who always takes times for discussions and progress meetings as well as correcting all my publications and this manuscript. Special thanks go to Caroline Kerello, Mobility Scholarship Manager who helped me and my family enormously when it comes to administrative.

I would like to thank my family. The advice and unwavering confidence in my abilities from my parents, coupled with their prayers, have been a constant source of support leading me to this point. I am profoundly thankful to my wife, Mazarina Azlin, for her selfless dedication of time and tremendous support. Her motivation and constant encouragement, acting as my cheerleader, proved instrumental, especially during challenging times. I express deep appreciation to all my children Batrisyia Syahirah, Hardy Danish Danial, Batrisyia Qasdina and Batrisyia Basyirah for their love, understanding, prayers and steadfast support throughout this PhD journey.

Lastly, but certainly not least, I extend my gratitude to both the National Defence University of Malaysia (NDUM) and Naval Group (formerly DCNS) for generously funding my studies at Ecole Centrale de Nantes.

*Hardy Azmir Bin Anuar*
*Ecole Centrale de Nantes, France*
*December 2023*

# ABSTRACT

This manuscript presents a study on the application of adaptive sliding mode control (first-order and supertwisting) to a robotic system. An aerial manipulator composed of a quadrotor equipped with a rigid two-degree-of-freedom manipulator arm is considered, the work focusing on the tracking of complex trajectories, by acting on the altitude and attitude of the quadrotor and the positioning of the manipulator arm. A complete simulator has been developed. The challenge was to assess the system's ability to follow helical motion trajectories in the presence of disturbances (external forces) or uncertainties (on-board mass). To achieve this, control laws based on the theory of sliding modes were proposed: the robust nature of this type of approach, coupled with their adaptive feature, has motivated their use for the first time in this field of application. A detailed analysis of the performance of these new control approaches has been carried out in simulation, along with a comparison with more conventional approaches, in particular with/without the presence of external disturbances and with/without the presence of point mass loading at the end of the manipulator arm.

**Keywords**: *Unmanned aerial manipulator, sliding mode control, supertwisting, adaptive gain.*

# TABLE OF CONTENTS

# List of acronyms

| | |
|---|---|
| 2-SMC | Second-order Sliding Mode Control |
| ADRC | Active Disturbance Rejection Control |
| COG | Center of Gravity |
| COM | Center of Mass |
| CLIK | Closed Loop Inverse Kinematic |
| DOF | Degree of Freedom |
| ESO | Extended State Observer |
| ESO-FSMC | Fuzzy Sliding Mode Control with Extended State Observer |
| ESRA | European Strategic Research Agenda |
| FASTESO | Fuzzy Adaptive Saturation Super-twisting Extended State Observer |
| FSMC | Fuzzy Sliding Mode Controller |
| GPS | Global Positioning System |
| HOSM | High-order Sliding Mode |
| IMU | Inertial Measurement Units |
| LQR | Linear Quadratic Regulator |
| MIMO | Multi Input Multi Output |
| PID | Proportional Integral Derivative |
| RNE | Recursive Newton-Euler algorithm |
| RTK-DGPS | Real-time Kinematic Differential Global Positioning System |
| SMC | Sliding Mode Control |
| SMCA | Adaptive Gain Sliding Mode Control |
| STWC | Super-Twisting Sliding Mode Control |
| STWCA | Adaptive Gain Super-Twisting Sliding Mode Control |
| TWC | Twisting Sliding Mode Control |
| UAM | Unmanned Aerial Manipulator |
| UAV | Unmanned Aerial Vehicle |
| VPIB | Variable Parameter Integral Backstepping |
| VToL | Vertical Taking-off and Landing |
| WGS84 | World Geodetic System |

# List of Figures

# List of Tables

# INTRODUCTION

An unmanned aerial vehicle (UAV), also known as drone, is an aircraft that has no pilot on board. Its flight can be remotely controlled, semi-autonomous or autonomous. UAVs have been a topic of research in diverse domains, such as military, civilian, academic and industrial applications. For various reasons, most studies tend to focus on quadrotor UAV that has four rotors. One of its advantages is that it does not require mechanical linkages to vary the rotor blade pitch angle as compared to scaled helicopter because its motion is controlled by changing the four rotors speed. The use of four rotors allows each rotor to have smaller diameter, that reduces the posibility of damaging the rotors if quadrotor hits anything. Quadrotor also requires only a small area for vertical taking-off and landing (VToL). Another advantage is its capability to fly in every direction, vertically and horizontally, to hover in a fixed position and to fly at a low altitude. These abilities make them an ideal instrument for detailed surveillance, remote sensing or flying through hard-to-reach areas. Extending this application to active action such as grasping, carrying or manipulating is an interesting way to expand the field of application of the quadrotor. However, grasping, carrying and manipulating an object with a quadrotor during the flight arise several problems due to both the unstable dynamics of the vehicle and the coupling effects and the uncertainties introduced by the presence of an object.

## Unmanned Aerial Manipulator (UAM)

According to European Strategic Research Agenda (ESRA) 2014 - 2020 [1], aerial robots are intended to be employed as robotic workers and co-workers, logistic robots and robots for exploration and inspection. To perform such tasks, a quadrotor due to its advantages is suitable for the base of aerial manipulator. Active task performed by aerial manipulator can cover in general the grasping, transporting, positioning, measuring, interaction or assembly/disassembly of mechanical parts of any objects. The use of aerial manipulator to access high-location and hard-to-reach areas that involves significant risks and accidents is very helpful. This aerial manipulator then can be remotely controlled, semi-autonomous or autonomous.

However in case of UAVs equipped with a manipulator, physical interactions with external objects while flying would induce several problems such as [2]:

— stability of the aircraft subject to forces and torques generated during the interactions;
— accuracy in trajectory tracking or positioning with respect to the targets while grasping or moving due to physical contact with objects;

— consideration of aerodynamic perturbations due to the proximity of surfaces;
— payload capability to carry devices for physical interaction;
— required flight time for moving in large areas, positioning and accurate interventions;
— reactivity to cancel the effect of external perturbations flying close to objects;
— planning taking into account the constraints involved in the physical interactions.

In order that UAV accomplish manipulation task in the air, they have to be equipped with a well adapted tool. The two most popular solutions [3] are either to mount a griper or multi fingered hand directly (or through cable suspension) on the UAV, known as *flying hand* or to equip the UAV with one or more robotic manipulators, hence the name *unmanned aerial manipulator* (UAM). With a flying hand it is possible to do pick and place operation only which is not enough for active task of UAV where the object cannot independently moved from the UAV.

(a) Quadrotor with mounted gripper known as *flying hand*. Source:[4].

(b) Quadrotor with robotic manipulator known as *unmanned aerial manipulator*. Source:[5].

Figure 1 – Quadrotor equipped with tool for manipulation task in the air.

UAM can consist of flying base such as quadrotor, robotic arm to perform active task and sensors like cameras, laser scanners, and so on for positioning of robotic arm on object. An UAM can be categorized as follows according to the different technological solutions addressing the problem of mounting a robotic manipulator on a UAV [3]:

— the number of degree of freedom (DOF) of robotic arm: 1-DOF, 2-DOF or more;
— the type of joint of robotic arm. Most of the joints are revolute with some are prismatic;
— how robotic arm are control: motion-controlled (kinematic) or torque-controlled (dynamic);
— configuration of robotic arm controlled as serial, parallel or redundant;

The modeling and control of UAM are quite challenging given that the attachment of robotic arm to a UAV creates coupling effects in the dynamic model of the system [3]. Indeed the dynamic behavior changes due to the movement of robotic arm interacting or manipulating objects. The main effects that appear and have to be considered are the displacement of the center of mass (COM), the variation of the moments of inertia and the dynamic reaction forces and torques

generated by the motion of the robotic arm [6, 7].

There are two approaches to address modeling and control problems [3, 8]. The first one is an overall approach, in which the UAV and the robotic arm are considered as a whole unit, and thus the planning and the controller are designed from the complete kinematic and dynamic models. This approach is quite difficult and complex but accurate [8]. The second approach is to consider the UAV and the robotic arm as two separate independent systems, and then separately builds models and controllers. The effects of the robotic arm on the UAV dynamic are then considered as external disturbances and vice versa. This approach is not as difficult as the first approach and simplifies the process of modeling and control of UAM [8].

Equipping aerial vehicle with manipulator arm, creating unmanned aerial manipulators, requires addressing issues like stability, accuracy, payload capacity and planning constraints. Aerial manipulators typically consist of a quadrotor base, a manipulator arm and sensors. Modeling and control of aerial manipulators can be approached holistically or by treating the quadrotor base and manipulator arm separately.

Chapter 1

# Literature Review and Problem Statement

In the past decade, there has been a growing interest in aerial robots within the field of robotics research. This is primarily due to the vast range of potential application areas they offer. More recently, the advancement of unmanned aerial vehicle equipped with robotic manipulators has further expanded these applications. This development has significantly improved their ability to interact with the natural environment. This chapter aims to provide a background study on various aspects related to aerial platforms, manipulator types, modeling and control of aerial manipulator. Additionally, it will address the research problems that arise in this context and outline the objectives of the study.

## 1.1 Floating platform based on single rotor system

Numerous researcher have displayed interest in the field of aerial manipulation. Work by Forte *et al.* [9] focuses on the modeling and the control of an aerial robot which consists of a VToL airframe with a ducted-fan configuration and a fully-actuated robotic arm composed of a prismatic joint. The aim is to allow this aerial platform to stabilize in a free flight and then with end effector come in contact with the vertical surface. Using impedance controller, it is shown that the system is stable both in the free flight and in the presence of contacts without the knowledge of the interaction forces. However this ducted-fan aerial vehicle used a single fixed pitch propellers for generating the main thrust. It rely on a number of mechanical flaps below this propellers both to generate anti-torque and for controlling the movement of the vehicle at which the design of control is complex.

Yang *et al.* [10] build a multi link aerial manipulator that is based on an helicopter as displayed by Figure 1.1. The system dynamics model is constructed using Euler-Lagrange formulation consist of three main components, the body dynamics model, the mid-dynamics model, and the actuator model. Each component serves a specific purpose in describing the behavior of the system. Full-state feedback linear quadratic regulator (LQR) controller is designed based on the linearized system model and simulations are conducted to show the control performances of the system. This controller takes into account the full state of the system, including position,

velocity, and attitude, to calculate the control actions. By utilizing the full-state feedback, the LQR controller can provide precise and robust control over the rotor-flying manipulator system. The study shows the LQR controller is able to stabilize the system near steady state but has very limited operating region; furthermore, the performance of the closed-loop system with the LQR controller is sensitive to the external disturbance.

Figure 1.1 – Helicopter with multijoint manipulator [10].

An other study of helicopter platform with gripper has been made by Pound *et al.* [11] and shows that dynamic load disturbances introduced by payload can be rejected thanks to proportional-integral-derivative (PID) based control. The study demonstrates grasping and retrieval of various objects while hovering using Yale aerial manipulator test-bed as shown in Figure 1.2. They successfully grasps and lifts objects with PID based control for a variety of objects (different masses, sizes and shapes). Yet this helicopter platform need to hover close to ground surface which generate ground effect that can cause instability to the vehicle.


Figure 1.2 – Yale aerial manipulator test-bed with mounted gripper [11].

The control of a VToL airframe with a ducted-fan configuration and helicopter platform involves two phases. The first phase is controlling the main propeller to generate thrust, while the second phase involves controlling actuated flaps for lateral, longitudinal, and vertical movement. However, the collective and cyclic pitch commands of this platform can be mechanically complex and less reliable, especially when dealing with miniature scaled machines.

## 1.2 Floating platform based on multi rotor system

Mellinger *et al.* [4] addresses the modeling and control for aerial grasping using a quadrotor. Two types of grippers, impactive and ingressive gripper, have been developed for this aerial grasping (Figure 1.3). This study takes into consideration the changes in dynamic of the system due to the grasped object. The inertial parameters of grasped object is estimated using least-squared method. The study also takes into account the displacement of the center of mass in the controller design. The inclusion of these parameters led to a significant improvement in tracking performance as shown by experimental results.

(a) The beam is grasped with an impactive gripper.

(b) The flat piece of wood is held by an ingressive gripper.

Figure 1.3 – Quadrotor with mounted gripper [4].

Study described in Kim *et al.* [5], Kannan *et al.* [12][13], Stergiopoulos *et al.* [14] and Gkountas *et al.* [15] present aerial manipulation using a quadrotor with a 2-DOF manipulator. Kim *et al.* [5], propose a model of the quadrotor and the manipulator arm as a combined system (see Figure 1.4). The study focuses on aerial manipulation during hovering flight, specifically the tasks of picking up and releasing objects. It emphasizes the need for a robust controller capable of handling disturbances caused by changes in quadrotor attitude during object manipulation, as well as uncertainties like battery drain and measurement errors. To address these challenges, an adaptive sliding mode controller is employed with estimated uncertainty for the adaptation law to cope with parametric uncertainties and external disturbance. An experimental results affirm the successful control of the quadrotor and manipulator arm throughout the entire flight, including object manipulations.

Kannan *et al.* [12, 13] proposed an independent modelling and control of quadrotor and manipulator as shown in Figure 1.5. Both quadrotor and manipulator also have a separate reference trajectory. The quadrotor model is based on Newton-Euler equation and Recursive Newton-Euler (RNE) method for manipulator. Finally the aerial manipulator is modeled based on coupling effect. The proposed scheme is first test with linear PID control and subsequently an Image Based Visual Servo system to perform specific tasks [12]. A visual servo system is used with the quadrotor to perform precise positioning and hover stabilization with manipulation task.

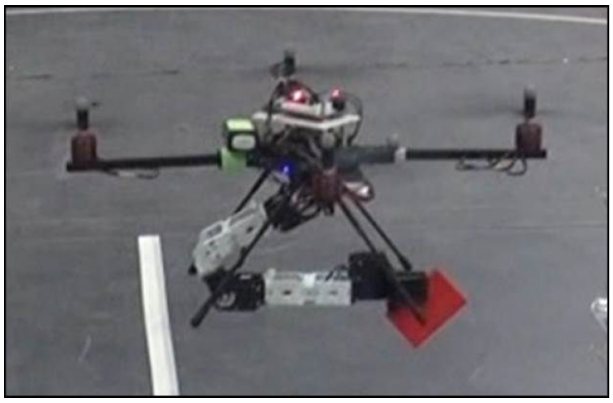


Figure 1.4 – Experimental demonstration by Kim *et al.* [5].

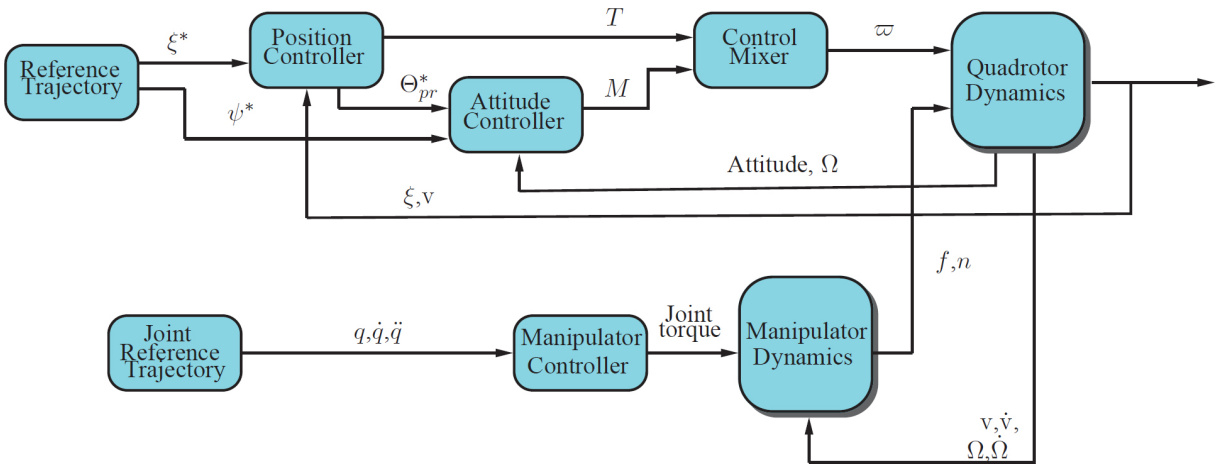


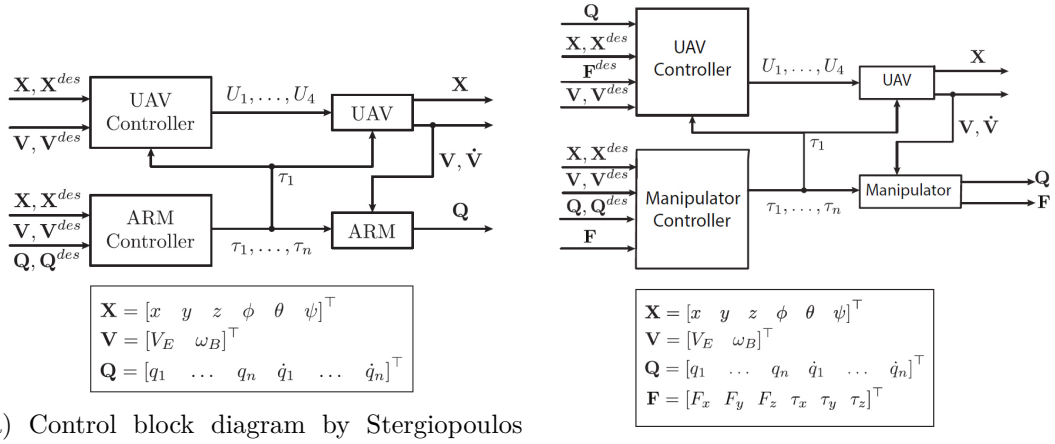

Figure 1.5 – Controller block diagram of quadrotor with manipulator by Kannan *et al.* [12, 13].

The adaptive position control of the aerial manipulator is then discussed in [13]. A hypothesis is considered that a manipulator with payload will prevent quadrotor to track a position reference. Hence an adaptive position control for quadrotor should be sufficient to handle the problem as long as the inner attitude loop is stabilized properly. The control law includes an

estimate value of mass and moment of inertia of the attached payload. Here the adaptive control is able to stabilize the quadrotor from the disturbances caused due to coupling forces and moments acting on it by the manipulator.

The papers by Stergiopoulos *et al.* [14] and Gkountas *et al.* [15] also discuss about independent modeling and control of aerial manipulator. In this case, the floating base effect is also taken into account in the RNE equations that transmit the velocities and accelerations of the quadrotor from the base to the terminal link of the manipulator; in return the forces and moments exerted on the quadrotor at the base of the manipulator are computed. This allows interconnection between the two subsystems (quadrotor and manipulator) gives a coupled dynamic modeling and control policy.

Stergiopoulos *et al.* [14] show the efficiency of proportional-derivative controller for quadrotor and computed torque controller for manipulator. Additonally Gkountas *et al.* [15] has considered a fictional force applied at the tip of terminal manipulator to simulate the interaction with environment. The dynamic model of the manipulator then includes the force applied and compute the reaction force on the UAV. Both studies show that the coupled dynamic model able to provide realistic simulation environment as well as the base for control design of UAM.




(a) Control block diagram by Stergiopoulos et al. [14].

(b) Block diagram by Gkountas et al. [15].

Figure 1.6 – Control block diagram of quadrotor/manipulator coupled system [14, 15].

In Lippiello *et al.* [16] (see Figure 1.7), an impedance control is considered which realize dynamic relationship between external force acting on aerial manipulator structure and whole system motion. The dynamic model of the whole system is presented in a symbolic matrix using Euler Lagrangian formalism. The formalism do not allow separation between slow rate position control and fast rate attitude control as in [12, 13]. The hovering control of a quadrotor with 3-DOF manipulator in the presence of contact forces and external disturbance acting on the system is proposed.

In Jimenez *et al.* [17], considered the quadrotor model with an articulated body, which causes

the center of mass to be displaced from the geometric centre of the quadrotor and modifies the inertia; both of them are dependent on the arm joint angles. On the other hand, the arm controller takes as information the state of the quadrotor, and is acting on the arm to compensate the motion of its base (quadrotor). The objective is to maintain the end effector of the manipulator at a desired position, or follows a predefined trajectory. By simulation and outdoor experiments Jimenez *et al.* [17] shows that Variable Parameter Integral Backstepping controller can be used to control the aerial platform in order to compensate the motion of the arm. It is also shown that the proposed method outperforms the results that can be obtained with conventional PID controllers.

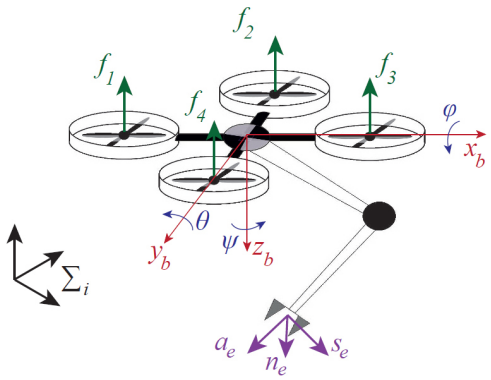



Figure 1.7 – Dynamic model schematic representation for [16].

Figure 1.8 – Quadrotor with 3-DOF manipulator arm developed by [17].

The papers by Arleo *et al.* [18] and Caccavale *et al.* [19] deal with the trajectory tracking control for a quadrotor equipped with a 5-DOF robotic manipulator. The proposed approach is based on a two-layer controller. First, in the top layer, an inverse kinematics algorithm computes the motion references for the position and yaw angle of the quadrotor and joint variables of the

manipulator. Secondly in the bottom layer, a motion control algorithm is in charge of tracking the motion references. In [18], a standard PID control is proposed for the tracking of motion references. While in [19] present an adaptive motion control algorithm which improve the motion tracking. This adaptive motion control takes into account disturbance and modeling uncertainties given by estimated value.

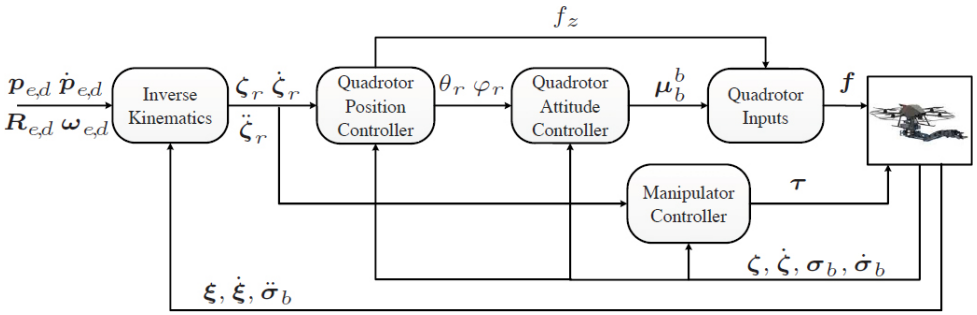



Figure 1.9 – Block scheme of the control architecture of [18] and [19].

The design of an adaptive control for a quadrotor equipped with 6-DOF manipulator is presented in [20], the manipulator being assumed to be already driven by a joint-based controller. Here the system is modeled by assuming the quadrotor and the manipulator to be rigid bodies; the whole system is regarded as an open kinematic chain with floating base. The proposed approach is based on adaptive control principles and is designed to estimate and compensate for the dynamics of the entire system. The proposed controller shows better performance, with smaller position errors and improved orientation control. The adaptive controller compensates for the presence of the manipulator arm and exhibits better stability and robustness in the presents of disturbances caused by arm movements.

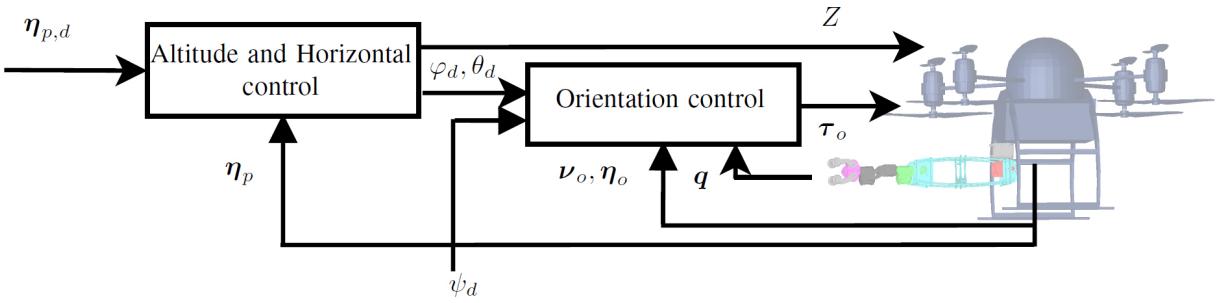



Figure 1.10 – Sketch of the control loop [20].

Finally in Heredia *et al.* [6], an UAV with eight rotors for additional payload capability is equipped by 7-DOF arm (see Figure 1.11) and carry sensors and processing hardware needed for outdoor positioning. A backstepping-based controller that uses a coupled dynamic model is proposed, and position-based Cartesian impedance controller for the manipulator arm is pro-

posed. The contoller using sensor measurement and visual servoing from inertial measurement units (IMU), barometer, ultrasonic sensor, RTK-DGPS and three cameras, for positioning and stabilization of the system. The proposed controller is able to effectively dampen the oscillation due to the motion of the arm to a large extent.

Figure 1.11 – Octorotor aerial manipulator [6] in flight.

The previous section has shown that there exist different kinds of floating platform (based on ducted-fan, helicopter and multi rotor) and different kind of manipulator type (based on gripper, prismatic and multi link). There also exist different kinds of modeling architecture and controlling design.

## 1.3 Robust control of unmanned aerial manipulator

Based on the compensation of the dynamic coupling between a UAV and a manipulator, Mimmo *et al.* [21] propose a control scheme that allows the end-effector of an aerial manipulator to track a desired trajectory while keeping the UAV in a fixed position. The control architecture includes two internal loops for stabilizing the UAV's position and attitude, as well as maintaining stability between the end-effector and the UAV. Nested saturation control is employed to ensure stability and robustness, limiting control inputs to prevent excessive responses and instability. These saturation functions are applied at different levels within the control system hierarchy, enabling effective handling of disturbances and uncertainties. Simulation results demonstrate accurate tracking performance despite external disturbances, demonstrating improved stability and efficiency in aerial manipulation tasks.

Modeling the UAV and the manipulator as a unified system in the form of a standard

robot dynamics equation, Bulut *et al.* [22] proposed a computed torque control with a PID outer loop to achieve accurate trajectory tracking. In this study, the kinematic and dynamic model of the system is derived using the Lagrange-D'Alembert formulation. The controller's gains are optimized using a multi-objective nonlinear least squares method to minimize errors between observed data and model predictions, considering 8 trajectories simultaneously. The MATLAB Global Optimization Toolbox provides a solver for this purpose. The performance of the controller is evaluated through simulation, incorporating realistic factors such as UAV's motor transfer function and torque filtering. Simulation results demonstrate the controller's ability to achieve accurate trajectory tracking and handle interaction forces to some extent, demonstrating its effectiveness in dealing with disturbances. The integral time weighted absolute errors serves as a performance criterion in assessing control system performance.

As the motion of the manipulator may disturb the attitude of UAV that affects flight stability and operation accuracy, Zhiyuan *et al.* [23] proposed a fuzzy sliding mode controller (FSMC) and fuzzy sliding mode control with extended state observer (ESO-FSMC). The proposed strategy combines fuzzy logic control, sliding mode control and an extended state observer to improve control system performance. Fuzzy logic handles uncertainties and nonlinearities, sliding mode control provides robustness while the extended state observer estimates and compensates for unmeasured states or disturbances. In simulation, FSMC shows faster adjustment speed and reduced attitude oscillation compared to sliding mode control, but with larger overshoot and chattering. The ESO-FSMC effectively estimates and suppresses disturbances, ensuring stable control. Overall, the integrated controller offers promising results for disturbance mitigation and system stability for UAM.

Chen *et al.* [24] proposes a robust control strategy to address the challenges faced by the UAM system when grasping objects in uncertain environmental conditions, such as continuous rain and wind. The strategy involves categorizing disturbances into internal disturbances caused by center of gravity shifts and external disturbances from the environment. In the position control, both internal and external disturbances are mitigated using a sliding mode controller. In the attitude control, an adaptive law is developed to estimate internal disturbances and a disturbance observer is designed to estimate external disturbances. Simulations are conducted to compare the proposed method with a controller without disturbances compensation as in [25]. The results demonstrate the effectiveness of the proposed approach with higher tracking accuracy and smoother tracking performance. The proposed method also effectively reduces overshoot and improves convergence speed to achieve desired setpoint in attitude control, especially in rapidly changing environments. The root mean squared error criterion is used as a standard for control performance evaluation. However, future study may account for uncertainties in the UAM's total mass after grasping an object.

The study by Jiao *et al.* [26] propose a control strategy for UAM operation to address chal-

lenges posed by disturbances like wind and model uncertainties, which can significantly impact the stability of the entire system. The strategy combines a fuzzy adaptive saturation super-twisting extended state observer called FASTESO to estimate and compensate for disturbances, along with a backstepping attitude controller to mitigate their effects. FASTESO incorporates a saturation function to overcome the chattering problem and employs fuzzy logic rules to adaptively adjust observer switching gains. The control strategy integrates FASTESO in the feedforward loop and a backstepping controller in the feedback loop, incorporating saturation function, tracking differentiator, and fuzzy logic methods. Simulations reveal the superiority of FASTESO over a simplified version without adaptive gain. Experimental tests confirm the control strategy's superior performance in reducing attitude offset, angular rate fluctuation and the impact of external disturbances.

Aerial manipulator pose significant challenges in control design due to issues such as interaction with the environment, nonlinear couplings and uncertainties in the system's structure and dynamics. Many existing control methods simplify the UAM dynamics, making the control design process more manageable but sacrificing accuracy. Samadikhoshkho *et al.* [27] propose four effective control algorithms that consider the complete dynamics of the system without simplification. These control methods aim to improve accuracy, robustness and ease of implementation in UAM tracking control. The proposed control method are

1. An enhanced inverse dynamics technique.
2. A modified hierarchical LQR scheme.
3. An improved adaptive sliding mode method.
4. A new semi-optimal nonlinear control.

The control methods are evaluated based on their ability to track a helixcal trajectory and a Lissajous curve trajectory. The simulation results demonstrate the efficiency and performance of the control methods in terms of accuracy, control effort, and robustness. Both the sliding mode and semi-optimal nonlinear control approaches exhibit good performance in controlling both the quadrotor and the manipulator. Conversely, the hierarchical LQR method shows satisfactory performance specifically for manipulator control. While inverse dynamics technique unable to indicate acceptable robustness and are not suitable for the manipulator control in the presence of significant uncertainties.

The application of super-twisting sliding mode control (STWC) to aerial manipulators has potential benefits. Aiming for better tracking performance in the presence of manipulator motion and disturbances Kuchwa-Dube *et al.* [28] presents the altitude and attitude tracking using STWC for a quadrotor-based aerial manipulator. Both first-order sliding mode control (SMC) and STWC are investigated, utilizing a saturation function to limit chattering. The STWC demonstrates superior performance with reduced chattering and lower control effort compared to the first-order SMC. Various performance metrics, such as the integral squared control inputs,

integral squared errors and integral time weighted absolute errors, are provided to compare the two control methods. Although STWC can mitigate chattering compared to first-order SMCs, some chattering still exists. Previous studies [29, 30, 31, 32] have shown that adaptive STWC exhibit lower chattering and better tracking performance then standard STWC. To further reduce chattering, Kuchwa-Dube *et al.* [33] propose the use of an adaptive STWC. The adaptive STWC control replaces fixed gains with adaptive gains as defined by [32]. The results demonstrated that the adaptive STWC control yielded superior tracking performance compared to the standard STWC control. The study also proposed a modified version of the adaptive gains, leading to a smoother adaptive gain transition, a slightly slower decrease in gain and a prolonged period of higher gain achieved by incorporating the tanh function. The modified adaptive STWC control demonstrated comparable or improved performance compared to the original adaptive STWC control.

In this section several control strategies for UAM to achieve accurate trajectory tracking and disturbance mitigation are proposed. These strategies include nested saturation control, computed torque control with PID outer loop, fuzzy sliding mode control, adaptive sliding mode control, and super-twisting SMC. Simulation and experimental results demonstrate improved stability, efficiency, and tracking performance of these control methods, considering factors like disturbances, uncertainties, and environmental conditions. The proposed control strategies address challenges posed by nonlinear couplings, uncertainties, and disturbances in UAM systems, aiming to improve accuracy, robustness, and ease of implementation in aerial manipulation tasks.

Researchers have proposed various control strategies, such as nested saturation control, computed torque control, fuzzy sliding mode control, adaptive sliding mode control, and super-twisting SMC, to improve trajectory tracking and disturbance handling in UAM. These strategies aim to enhance stability, efficiency, and tracking performance while addressing challenges associated with nonlinear couplings, uncertainties, and disturbances in UAM systems.

## 1.4 Problem statement and objective

Industry's expectations for the performance of industrial robots are constantly increasing. To be efficient, the industry will increasingly need adaptable robots, able to perceive their environment, easy to program, reconfigurable and mobile, interconnected and having a very large workspace. Aerial robotics, which has experienced significant growth for many years, has the potential to answer many of these requirements. The application of aerial robotics to access high-location and hard-to-reach areas (that involves significant risks and accidents), can be very helpful. However, to be able to tackle the industrial needs, researchers on aerial robots must still

solve many issues, one of them being the ability for the aerial robot to be stable and robust in order to manipulate a large class of objects.

The aerial manipulation has become a robotics research field that proposes the integration of one or more robotic arms with multirotors aerial platform, for allowing the manipulator to perform certain operations at high altitude or in areas out of the reach of standard fixed-based or ground-mobile manipulator arms. Possible applications span many areas such as disaster response, maintenance of infrastructure, inspection of remote sites. However, aerial manipulation is a real challenge when considering that the grasping, carrying and manipulating of an object with a multirotor platform during the flight cause unstable dynamics of the vehicle.

One of the biggest challenges arise from their limited payload. Another challenge is that the dynamics of the robot are significantly altered by the addition of payloads. According to the background and challenges on quadrotor UAV carrying payloads or manipulators ([4] - [33]), the contribution of this study is centred on the modelling and the design of a nonlinear control allowing stabilization of a quarotor UAV carrying a rigid manipulator arm.

The objectives of the work in this thesis can be summarized as:

— Design a complete model of a quadrotor carrying a rigid manipulator arm;

— design a robust control for the pose stabilization which takes into account the arm motion effects;

— design a robust control in order to take into account of external disturbance;

— use a new adaptive controller solutions based on super-twisting approach.

Chapter 2

# System Modelling

The mathematical dynamic models of robotic system are usually based on Lagrangian and Newton-Euler formulation [3, 34]. Lagrangian formulation is based on the computation of the kinetic and potential energies and is a systematic approach. It allows to get the analytical equations describing the dynamic model but becomes unfeasible for complex system with many degrees of freedom [34]. On the other hand, the Newton-Euler formulation is based on the computation of forces and torques. By using Newton-Euler recursive method the coding is simple and efficient for real time computation [35]. To summarize the mathematical model of system dynamics either is derived using the standard Lagrangian formalism, in which it is possible to have a general matrix form of the whole dynamic model [16, 36], or is derived using the Newton-Euler recursive formulation which has more simplifications from a coding point of view [12, 20].

## 2.1 Introduction

In this chapter the dynamic model of Unmanned Aerial Manipulator which consist of quadrotor UAV as a flying platform with attached 2-DOF manipulator arm is discuss. The mathematical dynamic models of this robotic system is based on Newton-Euler formulation. Newton-Euler formulation is used as the method for this study because it is simple and efficient which based on computation of forces and torque. The modeling of UAM is address independently by separating the quadrotor and manipulator arm apart and then built model and controller separately. The coupling effect of the manipulator arm on the quadrotor are then considered as the external disturbances and vice versa. This approach simplifies the process of modeling and control of UAM.

## 2.2 Reference Frame

Reference frames are required for the description of the position of aerial manipulator. This is an important aspect of describing the position of aerial manipulator. A reference frame is a conceptual coordinates system with an origin and three main axis in three dimensional geometric space. In order to get a model of quadrotor UAV equipped with a two degrees of freedom rigid

manipulator arm, there are six reference frame to be considered. It is a fixed frame ($F_W$), a local frame on the quadrotor body ($F_B$), a local frame on the base of manipulator ($F_0$) for coupling of the two system and three local frames on the manipulator arm ($F_1, F_2, F_e$) (refer Figure 2.1). There are two reference frames for the quadrotor model and four reference frames for the manipulator model. The list of all symbols used in this chapter are given in appendix A.

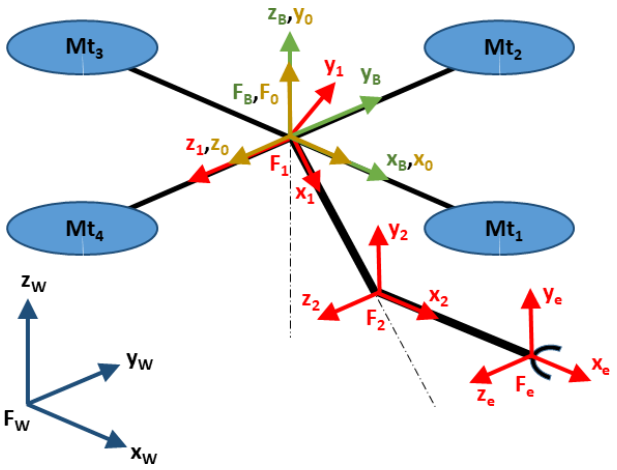



Figure 2.1 – Quadrotor UAV with a 2-DOF manipulator arm and the associated frame.

Inertial frame, also referred as world frame, is a fixed reference frame with the position of its origin depending on the method used to locate the quadrotor. This position of the origin can be taken at the corner of a laboratory or a World Geodetic System (WGS84) if using Global Positioning System (GPS). The inertial frame, denoted $F_W$ is defined by its origin $O_W$ and three unit vectors along the main axes denoted $\{x_W, y_W, z_W\}$ (see Figure 2.1 blue frame). The compact notation of this frame is $F_W$:$O_W - \{x_W, y_W, z_W\}$. Quadrotor body frame or simply body frame denoted $F_B$ is a rotating frame associated to the quadrotor with its origin $O_B$ at the center of mass of the quadrotor. The compact notation for this body frame is $F_B$:$O_B - \{x_B, y_B, z_B\}$ (see Figure 2.1 green frame).

The distance between the inertial frame ($F_W$) and the body frame ($F_B$) correspond to the a position of quadrotor center of mass given by vector $\boldsymbol{\xi} = [x\ y\ z]^T$ in the inertial frame. The attitude of the quadrotor is defined in the inertial frame using three Euler angles $\boldsymbol{\eta} = [\phi\ \theta\ \psi]^T$. Roll angle $\phi$ is rotation about the $x$-axis, pitch angle $\theta$ is the rotation about the $y$-axis and yaw angle $\psi$ is the rotation about $z$-axis. $\boldsymbol{\xi}$ and $\boldsymbol{\eta}$ are respectively the position and orientation coordinates (or the quadrotor state vector) with respect to the inertial frame $F_W$. Linear and angular velocities of the quadrotor are denoted as $\boldsymbol{\nu} = [u\ v\ w]^T$ (for linear velocities) and $\boldsymbol{\omega} = [p\ q\ r]^T$ (for angular velocities).

The two DOF rigid manipulator arm is composed of a sequence of 3 links and 2 joints (see

Figure 2.2). The links are numbered with link 0 representing the platform or base of the arm, while link 2 is the terminal link. Joint 1 connects link 1 to link 0 while joint 2 connects link 2 to link 1. These joints are revolute and assumed to be ideal (i.e. the centers of journal and bearing are always coincident). The $z$-axis of each link is assigned along the axis of its joint and $x$-axis is aligned with the common normal between two consecutive joint [37].

The base frame denoted $F_0$ and defined by its origin $O_0$ coincides with the quadrotor body frame. The base is where the manipulator arm is attached to the quadrotor body, or where the two systems model is coupled. The three unit vector along its main axes is denoted $\{x_0, y_0, z_0\}$ that yields the compact notation $F_0$:$O_0 - \{x_0, y_0, z_0\}$. Coordinate frame of manipulator arm joint 1 is denoted $F_1$ with its origin $O_1$ is the beginning of the link 1. The compact notation for this first link frame yields $F_1$:$O_1 - \{x_1, y_1, z_1\}$. As for the link 2 reference frame which is connected at joint 2, the compact notation for the terminal link frame yields $F_2$:$O_2 - \{x_2, y_2, z_2\}$. Another coordinates frame to take into consideration is the end effector frame denoted $F_e$ at the end of the second link where tools can be attached to the manipulator arm. The compact notation of end effector reference frame is $F_e$:$O_e - \{x_e, y_e, z_e\}$. Representation of all the frames can be found in Figure 2.1.

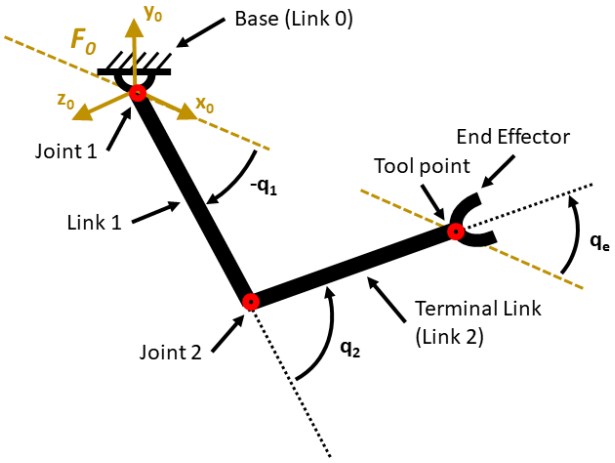



Figure 2.2 – 2-DOF manipulator arm structure.

Now that all the frame linked to the system have been defined, the position and orientation can be define. The position vector of the first frame $F_1$ with respect to the base frame $F_0$ is defined by $\boldsymbol{\xi}_1 = [x_1\ y_1\ z_1]^T$. This is the distance of joint 1 from the base frame along $x_0$-axis. This distance can be assumed to be equal to zero ($\boldsymbol{\xi}_1 = [0\ 0\ 0]^T$). Its orientation is given by an angle $q_1$ about $z_1$-axis with respect to base frame. Likewise, the position vector of the second frame $F_2$ with respect to the first frame $F_1$ is given by $\boldsymbol{\xi}_2 = [x_2\ y_2\ z_2]^T$ and its orientation about $z_2$-axis given by an angle $q_2$. The orientation of $F_1$ and $F_2$ can be denoted by a vector of joint positions given by $\boldsymbol{q} = [q_1\ q_2]^T$. The distance between base frame $F_0$ and end effector frame $F_e$

is defined by $\boldsymbol{\xi}_e = [x_e\ y_e\ z_e]^T$. This is the position of end effector frame in the base frame. The orientation of this frame is given by an angle $q_e$ which is equal to $q_1 + q_2$.

For the description of aerial manipulator position, six reference frame is introduced; the inertial fixed reference frame, $F_W$, the quadrotor body frame, $F_B$, the manipulator base frame, $F_0$, the manipulator joints frame, $F_1$ and $F_2$ and finally the end effector frame, $F_e$.

## 2.3 Rotation Matrix

To describe a vector associated to an object in a reference frame $F_K$:$O_k - \{x_k, y_k, z_k\}$ to a difference reference frame $F_J$:$O_j - \{x_j, y_j, z_j\}$ (see Figure 2.3), a rotation matrix is required. In a three dimensional space, the rotation matrix is a $3 \times 3$ matrix which contains the components of the unit vectors $\{\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}, \hat{\boldsymbol{z}}\}$ along each principle axis. It describes unit vectors in frame $K$ relative to frame $J$ and can be represented with a notation ${}^J\boldsymbol{R}_K = [{}^J\hat{\boldsymbol{x}}_K, {}^J\hat{\boldsymbol{y}}_K, {}^J\hat{\boldsymbol{z}}_K]$. Rotation matrix ${}^J\boldsymbol{R}_K$ when pre-multiplied by a vector $\boldsymbol{\xi}_K \in \mathbb{R}^3$ expressed in frame $K$ yields a vector $\boldsymbol{\xi}_J \in \mathbb{R}^3$ expressed in the frame $J$ that changes the direction while preserving its magnitude. It can be summarized by the following equation. (In this writing the subscript represent the base frame of vector value while the superscript is the reference frame. In the absent of superscript it is reference in the same frame.)

$$\begin{aligned} \boldsymbol{\xi}_J &= {}^J\boldsymbol{R}_K\, \boldsymbol{\xi}_K \\ \begin{bmatrix} x_j \\ y_j \\ z_j \end{bmatrix} &= \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} x_k \\ y_k \\ z_k \end{bmatrix} \end{aligned} \tag{2.1}$$

Consider first a rotation of a frame about its $x$-axis. The $y$-axis and $z$-axis are rotated by angle $\phi$ while the $x$-axis remains unchanged. From Figure 2.4, it can concluded that the components of the unit vectors $\{\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}, \hat{\boldsymbol{z}}\}$ along the axis $\{x', y', z'\}$ of the rotated frame as follows

$$\begin{aligned} \hat{\boldsymbol{x}} &= [1\ 0\ 0]^T \\ \hat{\boldsymbol{y}} &= [0\ C\phi\ S\phi]^T \\ \hat{\boldsymbol{z}} &= [0\ -S\phi\ C\phi]^T \end{aligned} \tag{2.2}$$

where $C\phi$ and $S\phi$ represent $cos(\phi)$ and $sin(\phi)$ respectively. Therefore, a rotation by $\phi$ radians

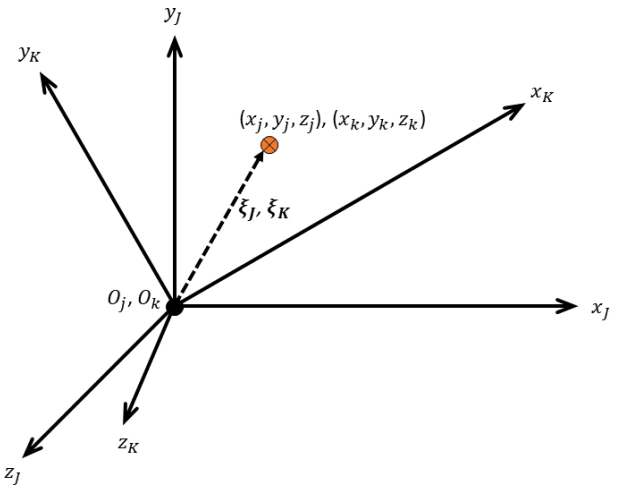



Figure 2.3 – A point in space can be described by position vector $\boldsymbol{\xi}_J = [x_j\ y_j\ z_j]^T$ in frame $J$ or $\boldsymbol{\xi}_K = [x_k\ y_k\ z_k]^T$ in frame $K$.

about the $x$-axis allows to defined the rotation matrix

$$\begin{aligned} \boldsymbol{R_x} &= [\hat{\boldsymbol{x}}\ \hat{\boldsymbol{y}}\ \hat{\boldsymbol{z}}] \\ \boldsymbol{R_x} &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & C\phi & -S\phi \\ 0 & S\phi & C\phi \end{bmatrix} \end{aligned} \tag{2.3}$$

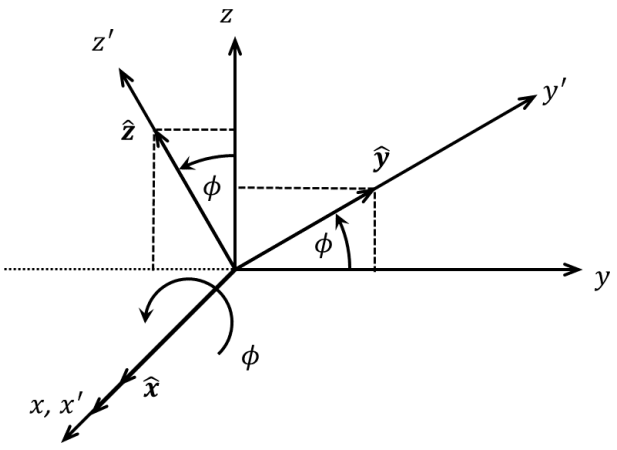



Figure 2.4 – Rotation by $\phi$ radians about the $x$-axis.

Similarly for a rotation of a frame about its $y$-axis, the y-axis remains unchanged and the $x$-axis and $z$-axis are rotated by angle $\theta$ (Figure 2.5). The rotation matrix about this $y$-axis is

then defined as

$$\boldsymbol{R_y} = \begin{bmatrix} C\theta & 0 & S\theta \\ 0 & 1 & 0 \\ -S\theta & 0 & C\theta \end{bmatrix} \tag{2.4}$$

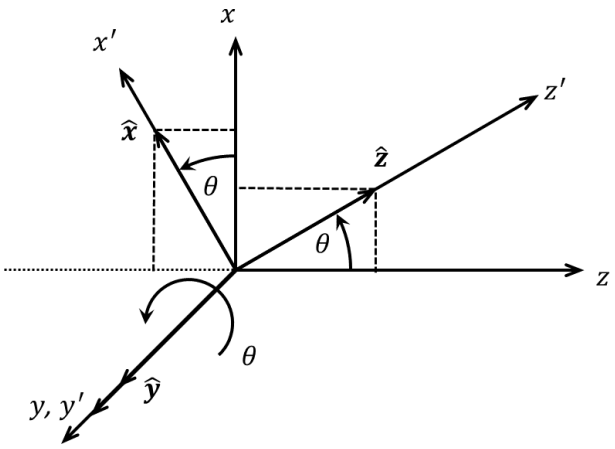



Figure 2.5 – Rotation by $\theta$ radians about the $y$-axis.

Finally for a rotation of a frame about its $z$-axis, the $z$-axis remains unchanged while the $x$-axis and $y$-axis are rotated by angle $\psi$ (Figure 2.6). The rotation matrix about this $z$-axis is defined as

$$\boldsymbol{R_z} = \begin{bmatrix} C\psi & -S\psi & 0 \\ S\psi & C\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.5}$$

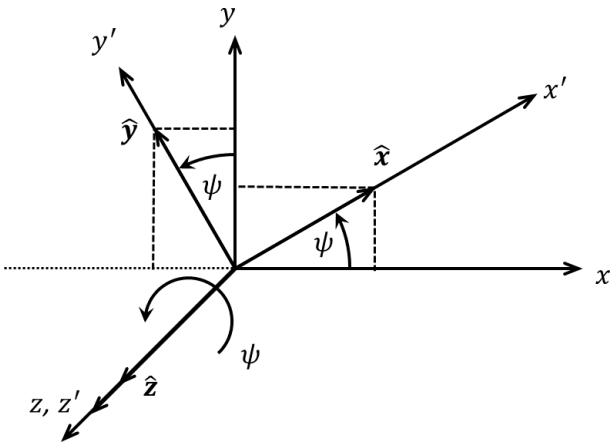



Figure 2.6 – Rotation by $\psi$ radians about the $z$-axis.

$\boldsymbol{R_x}$, $\boldsymbol{R_y}$ and $\boldsymbol{R_z}$ are the fundamental rotation matrices defined about the $x$-axis, $y$-axis and $z$-

axis respectively. Any transformation can be accomplished by combining these three rotations. Premultiplying these rotation matrices by its transpose gives rise to an identity matrix (e.g $\boldsymbol{R}_x \boldsymbol{R}_x^T = \boldsymbol{I}$). Since it is also an orthogonal matrix, hence $\boldsymbol{R}_x^{-1} = \boldsymbol{R}_x^T$. In general case, if $\boldsymbol{V}_J \in \mathbb{R}^3$ is a vector in frame $J$ and $\boldsymbol{V}_K \in \mathbb{R}^3$ is a vector in frame $K$, then the following relations hold:

$$\boldsymbol{V}_J = {}^J\boldsymbol{R}_K \boldsymbol{V}_K \tag{2.6}$$

$$\boldsymbol{V}_K = {}^J\boldsymbol{R}_K^T \boldsymbol{V}_J \tag{2.7}$$

## 2.4 Quadrotor modeling

Quadrotors are under-actuated systems in which four inputs are used to control motion in six degrees of freedom, three translational and three rotational. Quadrotors are lifted and propelled using four individual actuator motor attached to a rigid cross frame. These actuator motors are arranged with two motors in counter-clockwise direction (Mt1 and Mt3) placed diagonally with respect to each other and the other two motors in clockwise direction (Mt2 and Mt4) placed diagonally as shown in Figure 2.7.

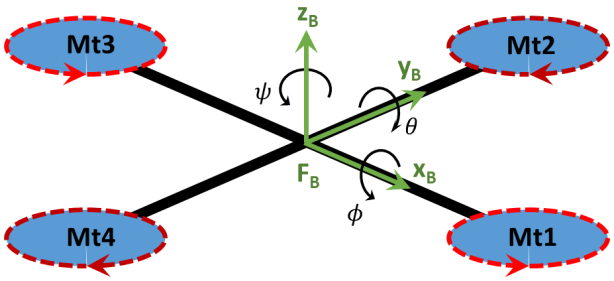



Figure 2.7 – Configuration of a quadrotor.

The motion is achieved by differential control of thrust generated by each motor. Up and down or lifting motions are achieved by collectively increasing or decreasing the speeds of all four motors. This leads to vertical force with respect to quadrotor body frame which lifts or lowers the quadrotor. If quadrotor is in flat position, the inertial frame and body frame coincide. Otherwise, the thrust generates both vertical and horizontal acceleration in the inertial frame.

Taking $x_B$ as forward direction of the quadrotor, left and right, or lateral motions are achieved by increasing right motor (Mt4) thrust and decreasing left motor (Mt2) thrust or vice versa while proportionally preserving the total collective thrust. This leads to a rolling torque about the quadrotor $x$-axis which makes the quadrotor to move to the left or right direction. Then, the forward and backward motions are achieved by increasing back motor (Mt3) thrust and decreasing front motor (Mt1) thrust or vice versa while proportionally preserving the total collective thrust. It leads to a pitching torque with respect to the quadrotor $y$-axis which makes

the quadrotor to move forward or backward.

Finally, yaw motion is achieved by increasing the thrust of diagonal pair of motors and decreasing the other while proportionally preserving the total collective thrust. This leads to a yawing torque with respect to the quadrotor $z$-axis which makes the quadrotor to turn clockwise or counter clockwise. The yaw movement is generated due to the fact that the left-right propellers rotate clockwise while the front-rear propellers rotate counter clockwise. Hence, when the overall torque is unbalanced, the quadrotor turns on itself around $z$-axis. The mathematical formulation for the quadrotor model is derived from Newton-Euler formalism with the following assumptions [38]:

— The structure is rigid and symmetrical such that the inertia matrix about x-axis is equal to inertia matrix about y-axis ($I_{xx} = I_{yy}$).
— The center of gravity coincides with the body fixed frame.
— The propellers are rigid.
— Thrust and drag are proportional to the square of propellers speed.

### 2.4.1 Quadrotor kinematics

Kinematics deal with motion of objects without taking into account the forces and moments acting on these objects. Deriving quadrotor kinematics (and later dynamic) equation requires the quadrotor states vector as well as rotation matrix for transformation between their different reference frames.

The orientation of the quadrotor is described using roll ($\phi$), pitch ($\theta$) and yaw ($\psi$) angles about the $x$, $y$ and $z$ -axis respectively (see Figure 2.7). The sequences of roll→pitch→yaw angle are used in this study to describe Euler angles rotation about fixed inertial frame. It can be expressed by a successive multiplication of the fundamental rotation matrix as

$$
\begin{aligned}
{}^{W}\boldsymbol{R}_{B} &= \boldsymbol{R_z}(\psi)\,\boldsymbol{R_y}(\theta)\,\boldsymbol{R_x}(\phi) \\
&= \begin{bmatrix} C\psi & -S\psi & 0 \\ S\psi & C\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C\theta & 0 & S\theta \\ 0 & 1 & 0 \\ -S\theta & 0 & C\theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & C\phi & -S\phi \\ 0 & S\phi & C\phi \end{bmatrix} \\
{}^{W}\boldsymbol{R}_{B} &= \begin{bmatrix} C\psi C\theta & C\psi S\theta S\phi - S\psi C\phi & C\psi S\theta C\phi + S\psi S\phi \\ S\psi C\theta & S\psi S\theta S\phi + C\psi C\phi & S\psi S\theta C\phi - C\psi S\phi \\ -S\theta & C\theta S\phi & C\theta C\phi \end{bmatrix}
\end{aligned}
\tag{2.8}
$$

With the rotation matrix ${}^{W}\boldsymbol{R}_{B}$ describing the orientation of the quadrotor from the body $F_B$ frame to the inertial frame $F_W$. Notice that this rotation matrix is orthogonal, thus ${}^{B}\boldsymbol{R}_{W} =$

${}^{W}\boldsymbol{R}_{B}^{T}$, that is the rotation matrix from inertial frame to the body frame.

$$ {}^{B}\boldsymbol{R}_{W} = \begin{bmatrix} C\psi C\theta & S\psi C\theta & -S\theta \\ C\psi S\theta S\phi - S\psi C\phi & S\psi S\theta S\phi + C\psi C\phi & C\theta S\phi \\ C\psi S\theta C\phi + S\psi S\phi & S\psi S\theta C\phi - C\psi S\phi & C\theta C\phi \end{bmatrix} \tag{2.9} $$

In order to link the angular rates ($\dot{\boldsymbol{\eta}} = [\dot{\phi}, \dot{\theta}, \dot{\psi}]^T$) of the quadrotor in the inertial frame to the angular velocity ($\boldsymbol{\omega} = [p, q, r]^T$) in the body frame a transfer matrix is used instead of rotation matrix. Obtaining this transformation is a bit complicated. The $\dot{\phi}$ transformation is achieved by a single rotation, the $\dot{\theta}$ transformation by two successive rotations and the $\dot{\psi}$ transformation by three successive rotations [39, 40, 41] as follows

$$ \begin{aligned} \begin{bmatrix} p \\ q \\ r \end{bmatrix} &= \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + \boldsymbol{R_x}(-\phi) \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + \boldsymbol{R_x}(-\phi)\boldsymbol{R_y}(-\theta) \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & -S\theta \\ 0 & C\phi & S\phi C\theta \\ 0 & -S\phi & C\phi C\theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \\ \begin{bmatrix} p \\ q \\ r \end{bmatrix} &= {}^{B}\boldsymbol{T}_{W} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \end{aligned} \tag{2.10} $$

Therefore, transfer matrix ${}^{B}\boldsymbol{T}_{W}$ from the inertial frame $F_W$ to the body frame $F_B$ reads as

$$ {}^{B}\boldsymbol{T}_{W} = \begin{bmatrix} 1 & 0 & -S\theta \\ 0 & C\phi & S\phi C\theta \\ 0 & -S\phi & C\phi C\theta \end{bmatrix} \tag{2.11} $$

The inverse of this transfer matrix ${}^{W}\boldsymbol{T}_{B}$ from the body frame $F_B$ to the inertial frame $F_W$ reads as

$$ {}^{W}\boldsymbol{T}_{B} = \begin{bmatrix} 1 & S\phi T\theta & C\phi T\theta \\ 0 & C\phi & -S\phi \\ 0 & \dfrac{S\phi}{C\theta} & \dfrac{C\phi}{C\theta} \end{bmatrix} \tag{2.12} $$

Together, the rotation matrix and the transfer matrix link the state vector of the quadrotor as follows

$$ \dot{\boldsymbol{\xi}} = {}^{W}\boldsymbol{R}_{B}\,\boldsymbol{\nu} \tag{2.13} $$

$$ \dot{\boldsymbol{\eta}} = {}^{W}\boldsymbol{T}_{B}\,\boldsymbol{\omega} \tag{2.14} $$

The rotation matrix is utilized to transform linear velocities from the quadrotor body frame to an inertial fixed frame. Conversely, the transfer matrix is employed to transform angular velocities from the quadrotor body frame to an inertial fixed frame.

Detailing the kinematic equations of the quadrotor, gives the following nonlinear equations

$$\dot{x} = uC\psi C\theta + v(C\psi S\theta S\phi - S\psi C\phi) + w(C\psi S\theta C\phi + S\psi S\phi) \tag{2.15}$$

$$\dot{y} = u(S\psi C\theta) + v(S\psi S\theta S\phi + C\psi C\phi) + w(S\psi S\theta C\phi - C\psi S\phi) \tag{2.16}$$

$$\dot{z} = u(-S\theta) + v(C\theta S\phi) + w(C\theta C\phi) \tag{2.17}$$

$$\dot{\phi} = p + q(S\phi T\theta) + r(C\phi T\theta) \tag{2.18}$$

$$\dot{\theta} = qC\phi - rS\phi \tag{2.19}$$

$$\dot{\psi} = q\frac{S\phi}{C\theta} + r\frac{C\phi}{C\theta} \tag{2.20}$$

### 2.4.2 Quadrotor force and moment

Each $i_{th}$ rotating propeller produces two kinds of forces on the body frame of quadrotor as shown in Figure 2.8. Considering the forces are an upward thrust $F_i$ and a moment of force or torque $M_i$.

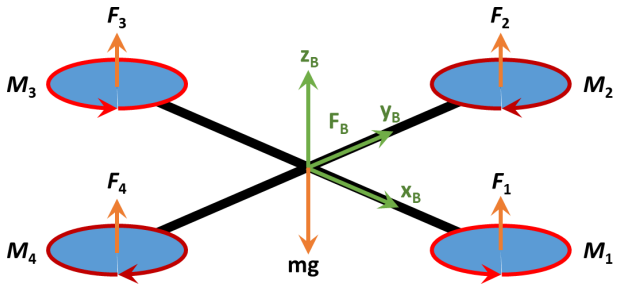



Figure 2.8 – Forces, $F_i$ and moments, $M_i$ on quadrotor.

Each propeller has an angular velocity $\boldsymbol{\omega_i}$ that produces thrust $F_i$ and torque $M_i$ according to [38]

$$F_i = \frac{1}{2}\rho A C_T r^2 \omega_i^2 \tag{2.21}$$

$$M_i = \frac{1}{2}\rho A C_D r^2 \omega_i^2 \tag{2.22}$$

where $\rho$ is the air density, $A$ is the propeller area, $C_T$ is the aerodynamic thrust coefficients, $C_D$ is the aerodynamic torque coefficients and $r$ is the radius of propeller.

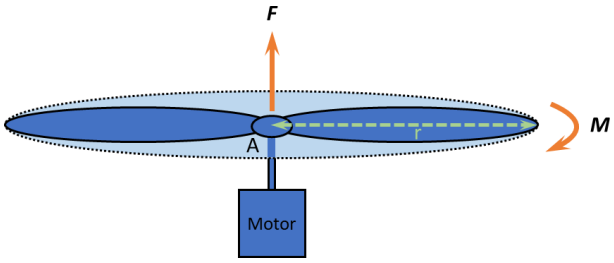



Figure 2.9 – Force and torque on propeller.

Equations can be simplified to

$$F_i = C_F \omega_i^2 \tag{2.23}$$

$$M_i = C_M \omega_i^2 \tag{2.24}$$

with $C_F$ and $C_M$ is the coefficient of the thrust and torque of the propeller respectively; the both coefficients can be experimentally determined [42, 38]. By identifying the force and torque generated by each propeller, one can get the total force ($\boldsymbol{F}$) and torque ($\boldsymbol{\tau}$) acting on the quadrotor. The total force acting on the quadrotor is defined as, $\boldsymbol{F} = [0\ 0\ f_z]^T$ where $f_z$ the lift reads as

$$\begin{aligned} f_z &= F_1 + F_2 + F_3 + F_4 && (2.25)\\ &= (C_F\omega_1^2) + (C_F\omega_2^2) + (C_F\omega_3^2) + (C_F\omega_4^2) \\ &= C_F(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) && (2.26) \end{aligned}$$

The total torque acting on the quadrotor, $\boldsymbol{\tau} = [\tau_x\ \tau_y\ \tau_z]^T$. Firstly, consists of the yawing torque $\tau_z$ generated around the $z$-axis due the drag of the propellers and reading as

$$\begin{aligned} \tau_z &= M_1 - M_2 + M_3 - M_4 && (2.27)\\ &= (C_M\omega_1^2) - (C_M\omega_2^2) + (C_M\omega_3^2) - (C_M\omega_4^2) \\ &= C_M(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) && (2.28) \end{aligned}$$

The pitching torque $\tau_y$ is evaluated about $y$-axis and is produced by the forces of the front and back propellers. It reads as

$$\begin{aligned} \tau_y &= -F_1 d + F_3 d && (2.29)\\ &= -(C_F\omega_1^2)d + (C_F\omega_3^2)d \\ &= C_F d(-\omega_1^2 + \omega_3^2) && (2.30) \end{aligned}$$

Finally, the rolling torque $\tau_x$ evaluated about $x$-axis and is produced by the forces of the right and left propellers, reads as

$$\begin{aligned} \tau_x &= F_2 d - F_4 d && (2.31)\\ &= (C_F \omega_2^2) d - (C_F \omega_4^2) d \\ &= C_F d(\omega_2^2 - \omega_4^2) && (2.32) \end{aligned}$$

With $d$ the distance from the center of the propeller to the center of mass of the quadrotor.

Thus, forces and torques from the four propellers of the quadrotor can be combined as four control inputs defined as

$$\begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} = \begin{bmatrix} C_F(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \\ C_F d(\omega_2^2 - \omega_4^2) \\ C_F d(-\omega_1^2 + \omega_3^2) \\ C_M(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) \end{bmatrix} \tag{2.33}$$

The input $U_1$ is the upward thrust $f_z$ whereas $U_2$, $U_3$ and $U_4$ are the torques $\tau_x$, $\tau_y$ and $\tau_z$ respectively.

### 2.4.3 Quadrotor dynamics

The quadrotor is assumed to be symmetrical such that the inertia matrix $\boldsymbol{I}$ is a diagonal matrix in which $I_{xx} = I_{yy}$, and is defined as

$$\boldsymbol{I} = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \tag{2.34}$$

where $I_{xx}, I_{yy}$ and $I_{zz}$ are the moment of inertia about $x$-axis, $y$-axis and $z$-axis respectively. The external forces acting on the center of mass are given by the gravitational force, $mg$ (where $m$ is the quadrotor mass and $g$ is the gravitational acceleration) and the total upward thrust ($\boldsymbol{F}$) which contribute to the acceleration of the quadrotor. It yields from the Newton equation [42]

$$m\ddot{\boldsymbol{\xi}} = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} + {}^{W}\boldsymbol{R}_B \boldsymbol{F} \tag{2.35}$$

Thus, the acceleration is defined as

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} + \begin{bmatrix} C\psi S\theta C\phi + S\psi S\phi \\ S\psi S\theta C\phi - C\psi S\phi \\ C\theta C\phi \end{bmatrix} \frac{f_z}{m} \tag{2.36}$$

The Euler equation [42] represents the rotational equation of motion specifically for the torque produced by a rotating propeller. This equation quantifies how the torque determines the angular acceleration of the system as follows.

$$\boldsymbol{\tau} = \boldsymbol{I}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times (\boldsymbol{I}\boldsymbol{\omega}) \tag{2.37}$$

$$\boldsymbol{I}\dot{\boldsymbol{\omega}} = \boldsymbol{\tau} - \boldsymbol{\omega} \times (\boldsymbol{I}\boldsymbol{\omega}) \tag{2.38}$$

$$\dot{\boldsymbol{\omega}} = \boldsymbol{I}^{-1}(\boldsymbol{\tau} - \boldsymbol{\omega} \times (\boldsymbol{I}\boldsymbol{\omega})) \tag{2.39}$$

Thus, the angular acceleration reads as

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix}^{-1} \left( \begin{bmatrix} \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} - \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \left( \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \right) \right) \tag{2.40}$$

To summarize, the dynamic model of quadrotor is given by

$$\ddot{x} = \frac{U_1}{m}(C\psi S\theta C\phi + S\psi S\phi) \tag{2.41}$$

$$\ddot{y} = \frac{U_1}{m}(S\psi S\theta C\phi - C\psi S\phi) \tag{2.42}$$

$$\ddot{z} = \frac{U_1}{m} C\theta C\phi - g \tag{2.43}$$

$$\dot{p} = \frac{U_2}{I_{xx}} + \frac{I_{yy} - I_{zz}}{I_{xx}} qr \tag{2.44}$$

$$\dot{q} = \frac{U_3}{I_{yy}} + \frac{I_{zz} - I_{xx}}{I_{yy}} pr \tag{2.45}$$

$$\dot{r} = \frac{U_4}{I_{zz}} + \frac{I_{xx} - I_{yy}}{I_{zz}} pq \tag{2.46}$$

Notice that this quadrotor system can be written as a nonlinear system affine in the control input with the state vector define as $[x\ y\ z\ \dot{x}\ \dot{y}\ \dot{z}\ \phi\ \theta\ \psi\ p\ q\ r]^T$ and the input as $[U_1\ U_2\ U_3\ U_4]^T$.

## 2.5 Manipulator modeling

Two methods can be used to derive the mathematical model of the dynamics of the robot arm. These methods are the Lagrangian formulation and Newton-Euler formulation [35, 37]. The Lagrangian formulation is described in terms of work and energy using generalized coordinates. The obtained equations are generally compact and provide a closed-form expression in terms of joint torques and joint displacements. On the other hand, the Newton-Euler formulation is derived from Newton's second law of motion which describes dynamic systems in terms of force and momentum. The equations consider all the forces and moments acting on the individual robot links, including the reaction forces and moments between the links. Newton-Euler formulation is used here because of its simplification and its efficiency from a coding point of view.

### 2.5.1 Static relations

Consider the free body diagram of Figure 2.10 showing the forces and torques acting on the link $j$. This link $j$ is connected to links $j-1$ and $j+1$ by joint $j$ and $j+1$ respectively. Let $O_j$ be a point fixed on joint $j$ at the origin of link $j$ and $O_{j+1}$ be a point fixed on joint $j+1$ at the origin of link $j+1$. Through these connections with the adjacent links, link $j$ receives forces and moments from its both sides.

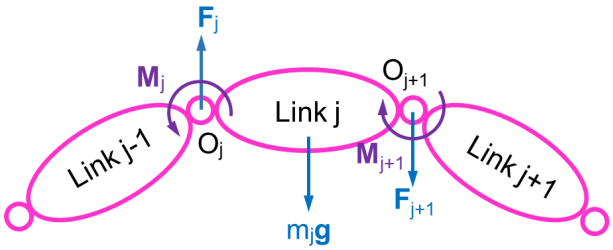



Figure 2.10 – Forces and torques on link j.

Let $\boldsymbol{F}_j$ be a three dimensional force acting from link $j-1$ to link $j$. Likewise, let $\boldsymbol{F}_{j+1}$ be the force from link $j$ to link $j+1$. Hence, the coupling force applied to link $j$ from link $j+1$ is then given by $-\boldsymbol{F}_{j+1}$. The gravity force acting at the center of mass of link $j$ is denoted $m_j\boldsymbol{g}$, with $m_j$ the mass of link $j$ and $\boldsymbol{g}$ is the $3 \times 1$ vector representing the acceleration due to the gravity. The static balance of linear forces is then given by

$$\boldsymbol{F}_j - \boldsymbol{F}_{j+1} + m_j\boldsymbol{g} = 0 \tag{2.47}$$

In the sequel, the balance of moments is derived. The moment applied to link $j$ by link $j-1$ is denoted $\boldsymbol{M}_j$, and therefore the moment applied to link $j$ by link $j+1$ is $-\boldsymbol{M}_{j+1}$. Furthermore, the linear forces $\boldsymbol{F}_{j+1}$ and gravity force also cause moments about the origin of link $j$. The static

balance of moments with respect to the origin of link $j$ is thus given by

$$\boldsymbol{M}_j - \boldsymbol{M}_{j+1} - \boldsymbol{L}_{j+1} \times \boldsymbol{F}_{j+1} + \boldsymbol{MS}_j \times \boldsymbol{g} = 0 \tag{2.48}$$

where

$\boldsymbol{L}_{j+1}$ is the position vector of point $O_{j+1}$ from $O_j$;
$\boldsymbol{MS}_j$ is the vector of the first moment of inertia;

### 2.5.2 Inverse dynamics

The inverse dynamic model provides the joint forces or torques in terms of the joint positions, velocities and accelerations. It is described by:

$$\boldsymbol{\Gamma} = \boldsymbol{f}(\boldsymbol{q}, \dot{\boldsymbol{q}}, \ddot{\boldsymbol{q}}, \boldsymbol{F}_e) \tag{2.49}$$

where:

$\boldsymbol{\Gamma}$ is the vector of joint forces or torques;
$\boldsymbol{q}$ is the vector of joint positions;
$\boldsymbol{F}_e$ is the vector of forces and moments exerted by the robot on the environment.

The closed-form expression of inverse dynamic model reads as [35]:

$$\boldsymbol{\Gamma} = \boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\boldsymbol{q} + \boldsymbol{Q}(\boldsymbol{q}) \tag{2.50}$$

where:

$\boldsymbol{M}(\boldsymbol{q})$ is the inertia matrix of the manipulator.
$\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ is the vector of Coriolis and centrifugal torque.
$\boldsymbol{Q}(\boldsymbol{q})$ is the vector of gravity effects.

To analyze the dynamic behavior of link $j$, the inertial force and moment (known as wrench) due to the dynamic motion of the link have to be considered. The equation of motion of a rigid body given by Newton-Euler formulation can be represented by two equations: the first one describes the translational motion with respect to $O_j$ while the other one describes the rotational motion about $O_j$. The former is known as Newton's equation of motion, and the latter is called Euler's equation of motion.

The Newton-Euler equations of the inertial force, $\boldsymbol{F}_{tj}$ and moment, $\boldsymbol{M}_{tj}$, of link $j$ at $O_j$ are given by [35]

$$\boldsymbol{F}_{tj} = m_j\dot{\boldsymbol{v}}_j + \dot{\boldsymbol{\omega}}_j \times \boldsymbol{MS}_j + \boldsymbol{\omega}_j \times (\boldsymbol{\omega}_j \times \boldsymbol{MS}_j) \tag{2.51}$$

$$\boldsymbol{M}_{tj} = \boldsymbol{I}_{Oj}\dot{\boldsymbol{\omega}}_j + \boldsymbol{MS}_j \times \dot{\boldsymbol{v}}_j + \boldsymbol{\omega}_j \times (\boldsymbol{I}_{Oj}\boldsymbol{\omega}_j) \tag{2.52}$$

with $\boldsymbol{v}_j$ and $\boldsymbol{\omega}_j$ are the linear and angular velocities at $O_j$ respectively. While, $\boldsymbol{I}_{Oj}$ is the inertia tensor of link $j$ about $O_j$. The dynamic behavior of link $j$ is determined by equating the inertial force and moment to the static balance of forces and moments. This can be expressed as follows:

$$\boldsymbol{F}_{tj} = \boldsymbol{F}_j - \boldsymbol{F}_{j+1} + m_j\boldsymbol{g} - \boldsymbol{F}_{ej} \tag{2.53}$$

$$\boldsymbol{M}_{tj} = \boldsymbol{M}_j - \boldsymbol{M}_{j+1} - \boldsymbol{L}_{j+1} \times \boldsymbol{F}_{j+1} + \boldsymbol{MS}_j \times \boldsymbol{g} - \boldsymbol{M}_{ej} \tag{2.54}$$

With $\boldsymbol{F}_{ej}$ and $\boldsymbol{M}_{ej}$ the force and moment exerted by the link $j$ on the environment; it may include contributions from springs, dampers and contact with the environment.

By rearranging Equation (2.53) and (2.54) the force, $\boldsymbol{F}_j$ and moment, $\boldsymbol{M}_j$ acting on joint j reads as:

$$\boldsymbol{F}_j = \boldsymbol{F}_{tj} + \boldsymbol{F}_{j+1} - m_j\boldsymbol{g} + \boldsymbol{F}_{ej} \tag{2.55}$$

$$\boldsymbol{M}_j = \boldsymbol{M}_{tj} + \boldsymbol{M}_{j+1} + \boldsymbol{L}_{j+1} \times \boldsymbol{F}_{j+1} - \boldsymbol{MS}_j \times \boldsymbol{g} + \boldsymbol{M}_{ej} \tag{2.56}$$

### 2.5.3 Recursive Newton-Euler

The recursive Newton-Euler (RNE) algorithm [35, 37] which is considered as one of the efficient algorithms for real time computation of the inverse dynamic model, consists of two recursive computations: forward recursion and backward recursion. During the forward recursion, the algorithm calculates the velocities and accelerations of each link, starting from the base and moving towards the terminal link. This computation allows for the determination of the dynamic wrench acting on each link. Conversely, the backward recursion begins at the terminal link and progresses towards the base. In this step, the algorithm determines the reaction wrenches exerted on each link, which in turn enables the calculation of the joint torques.

#### Forward Recursive

The forward recursive algorithm for joint $j = 1$ to $n$ is given as:

$${}^j\boldsymbol{\omega}_j = {}^j\boldsymbol{R}_{j-1}\,{}^{j-1}\boldsymbol{\omega}_{j-1} + \dot{q}_j\hat{\boldsymbol{z}}_0 \tag{2.57}$$

$${}^j\dot{\boldsymbol{\omega}}_j = {}^j\boldsymbol{R}_{j-1}\,{}^{j-1}\dot{\boldsymbol{\omega}}_{j-1} + \ddot{q}_j\hat{\boldsymbol{z}}_0 + {}^j\boldsymbol{\omega}_{j-1} \times \dot{q}_j\hat{\boldsymbol{z}}_0 \tag{2.58}$$

$${}^j\dot{\boldsymbol{\nu}}_j = {}^j\boldsymbol{R}_{j-1}({}^{j-1}\dot{\boldsymbol{\nu}}_{j-1} + {}^j\dot{\boldsymbol{\omega}}_j \times {}^{j-1}\boldsymbol{L}_j + {}^j\boldsymbol{\omega}_j \times ({}^j\boldsymbol{\omega}_j \times {}^{j-1}\boldsymbol{L}_j)) \tag{2.59}$$

with $\boldsymbol{\nu}$, $\boldsymbol{\omega}$, $\boldsymbol{R}$, $\boldsymbol{L}$, $q$ and $\hat{\boldsymbol{z}}$ are the linear velocity, angular velocity, rotation matrix, position vector, joint position and a unit vector along z-axis respectively. Recall that the subscript represent the base frame of vector value while the superscript is the reference frame. In the absent of superscript

it is reference in the same frame. The initial condition for a fixed based manipulator arm are $\boldsymbol{\omega}_0 = 0$, $\dot{\boldsymbol{\omega}}_0 = 0$ and $\dot{\boldsymbol{\nu}}_0 = 0$. Then the dynamic wrench on each joint reads as

$$ {}^{j}\boldsymbol{F}_{tj} = M_j\, {}^{j}\dot{\boldsymbol{v}}_j + {}^{j}\dot{\boldsymbol{\omega}}_j \times {}^{j}\boldsymbol{MS}_j + {}^{j}\boldsymbol{\omega}_j \times ({}^{j}\boldsymbol{\omega}_j \times {}^{j}\boldsymbol{MS}_j) \tag{2.60} $$

$$ {}^{j}\boldsymbol{M}_{tj} = {}^{j}\boldsymbol{I}_{oj}\, {}^{j}\dot{\boldsymbol{\omega}}_j + {}^{j}\boldsymbol{MS}_j \times {}^{j}\dot{\boldsymbol{v}}_j + {}^{j}\boldsymbol{\omega}_j \times ({}^{j}\boldsymbol{I}_{oj}\, {}^{j}\boldsymbol{\omega}_j) \tag{2.61} $$

**Backward Recursive**

The backward recursive algorithm for joint $j = n$ to 1 reads as

$$ {}^{j}\boldsymbol{F}_j = {}^{j}\boldsymbol{F}_{tj} + {}^{j}\boldsymbol{F}_{j+1} + {}^{j}\boldsymbol{F}_{ej} \tag{2.62} $$

$$ {}^{j-1}\boldsymbol{F}_j = {}^{j-1}\boldsymbol{R}_j\, {}^{j}\boldsymbol{F}_j \tag{2.63} $$

$$ {}^{j}\boldsymbol{M}_j = {}^{j}\boldsymbol{M}_{tj} + {}^{j}\boldsymbol{R}_{j+1}\, {}^{j+1}\boldsymbol{M}_{j+1} + {}^{j}\boldsymbol{L}_{j+1} \times {}^{j}\boldsymbol{F}_{j+1} + {}^{j}\boldsymbol{M}_{ej} \tag{2.64} $$

The gravity terms from equations (2.55) - (2.56) are canceled in equations (2.62) - (2.64) by taking into account their effects in the initial acceleration such that

$$ \dot{\boldsymbol{\nu}}_0 = -\boldsymbol{g} \tag{2.65} $$

By iteratively applying the forward and backward recursions, the RNE algorithm can compute the joint forces and torques acting on each link.

Finally, the torque $\boldsymbol{\Gamma}_j$ on the axis for revolute joint manipulator arm is given as.

$$ \boldsymbol{\Gamma}_j = {}^{j}\boldsymbol{M}_j^T \hat{\boldsymbol{z}}_0 + \boldsymbol{\Gamma}_{fj} \tag{2.66} $$

with $\boldsymbol{\Gamma}_{fj}$ denoting the joint $j$ friction forces and rotor inertia. For this study, supposed that manipulator joint is actuated with direct drives, that gives $\boldsymbol{\Gamma}_{fj} = 0$ [43].

### 2.5.4 Forward dynamics

The computation of the forward dynamic model is used to carry out simulations for the purpose of testing the manipulator performances. For simulation, the dynamic equations are solved for the joint accelerations from the joint torques, positions and velocities of the manipulator. Through integration of the joints accelerations, the manipulator state variables are then determined. In this study, the method based on Newton-Euler inverse dynamic model is considered and requires the computation of the inverse of the inertia matrix $\boldsymbol{M}(\boldsymbol{q})$ of the manipulator.

From closed-form expression of inverse dynamic model Equation (2.50), the forward dynamic

model reads as the solution of the joint accelerations from the following equation

$$\ddot{\boldsymbol{q}} = \boldsymbol{M}(\boldsymbol{q})^{-1}\Big[\boldsymbol{\Gamma} - \boldsymbol{H}(\boldsymbol{q}, \dot{\boldsymbol{q}})\Big] \tag{2.67}$$

with $\boldsymbol{H}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\boldsymbol{q} + \boldsymbol{Q}(\boldsymbol{q})$. The computation of the forward dynamics can be broken down into three steps [35]

1. the calculation of $\boldsymbol{H}(\boldsymbol{q}, \dot{\boldsymbol{q}})$.
2. the calculation of $\boldsymbol{M}(\boldsymbol{q})$.
3. the solution of the linear equation for $\ddot{\boldsymbol{q}}$.

The computation of the first step is accomplished using the inverse dynamics algorithm with the desired joint accelerations equal to zero. Then, the inertia matrix $\boldsymbol{M}(\boldsymbol{q})$ can be calculated one column at a time using inverse dynamics algorithm. The calculation of $j_{th}$ column of inertia matrix is accomplished with the forces, gravity and velocities equal to zero, and setting $j_{th}$ row of $\ddot{\boldsymbol{q}}$ with one and zero elsewhere. Iterating the algorithm for $j = 1$ to $n$ leads to the construction of the entire inertia matrix. Finally, the solution for $\ddot{\boldsymbol{q}}$ is achieved by solving the Equation (2.67) above.

The forward dynamics model computation is utilized to conduct simulations aimed at evaluating manipulator performance and studying control laws. In simulations, the forward dynamics equations are solved to determine the joint accelerations given the input torques and the current state of the manipulator. By integrating the joint accelerations, the manipulator trajectory is established.

## 2.6 Modeling of coupled system

Modeling a coupled system that includes both a quadrotor and a manipulator is a challenging and intricate task. The difficulty arises due to the coupling effects between the dynamics of the aerial vehicle and the manipulator arm. To tackle this problem, this section adopts an approach that treats the aerial vehicle and the manipulator arm as separate entities. Separate models and controllers are developed for each component in the previous section. Consequently, the coupling effect between the arm and the aerial vehicle is treated as external disturbances for both subsystems. By adopting this approach, the process of modeling and controlling an unmanned aerial manipulator is simplified. The complexities arising from the coupling effects are managed by considering them as external disturbances, allowing for more manageable and focused modeling and control strategies for the individual subsystem.

### 2.6.1 Orientation of coupled system

In order to derive the entire system dynamics, a manipulator is attached to the body frame of quadrotor at its COM as shown in Figure 2.11. The configuration of the manipulator in relation to the inertial fixed frame, $F_W$ is given thanks to the rotation matrix

$$ {}^0\boldsymbol{R}_W = {}^0\boldsymbol{R}_B \, {}^B\boldsymbol{R}_W \tag{2.68} $$

The manipulator base frame $F_0$ attached at the body frame of quadrotor $F_B$ as shown in the Figure 2.11 gives the rotation matrix from the body frame to the base frame and reads as

$$ {}^0\boldsymbol{R}_B = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix} \tag{2.69} $$

Therefore from (2.9) and (2.69), one gets

$$ {}^0\boldsymbol{R}_W = \begin{bmatrix} C\psi C\theta & S\psi C\theta & -S\theta \\ C\psi S\theta C\phi + S\psi S\phi & S\psi S\theta C\phi - C\psi S\phi & C\theta C\phi \\ -C\psi S\theta S\phi + S\psi C\phi & -S\psi S\theta S\phi - C\psi C\phi & -C\theta S\phi \end{bmatrix} \tag{2.70} $$

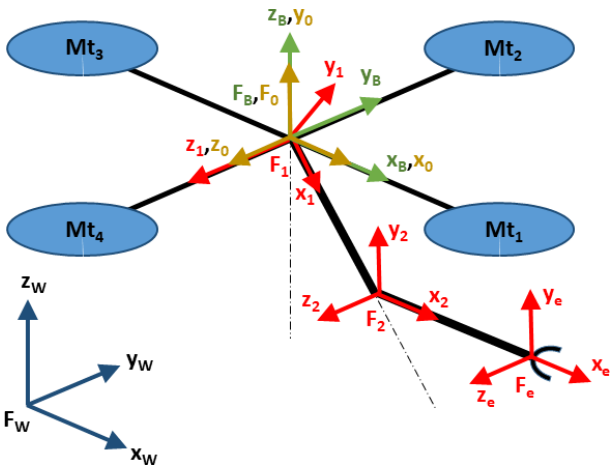



Figure 2.11 – Quadrotor equipped with 2-DOF Robotic Manipulator.

### 2.6.2 Dynamics of coupled system

As a coupled system, the manipulator is no longer stationary and has a floating base. This affects the computation of recursive Newton-Euler algorithm of the manipulator. Indeed, the algorithm provides the forces and torques, that are then applied at the center of mass of the

quadrotor [14, 15] as external disturbance. This is the coupling effect that changes the dynamics of each subsystem as in Figure 2.12. For a floating base, the initial velocity ($\boldsymbol{\omega}_0$) and acceleration $\{\dot{\boldsymbol{\nu}}_0, \dot{\boldsymbol{\omega}}_0\}$ used by the RNE algorithm computing manipulator arm joint torque $\boldsymbol{\Gamma}$, are initialized from quadrotor velocity ($\boldsymbol{\omega}$) and acceleration $\{\ddot{\boldsymbol{\xi}}, \dot{\boldsymbol{\omega}}\}$ as follows;

$$\boldsymbol{\omega}_0 = {}^0\boldsymbol{R}_B\boldsymbol{\omega} \tag{2.71}$$

$$\dot{\boldsymbol{\omega}}_0 = {}^0\boldsymbol{R}_B\dot{\boldsymbol{\omega}} \tag{2.72}$$

$$\dot{\boldsymbol{\nu}}_0 = {}^0\boldsymbol{R}_W(\ddot{\boldsymbol{\xi}} + [0\ 0\ -g]^T) \tag{2.73}$$

These initial velocity and acceleration are in turn transmitted from one link to another by forward recursion and would result in an additional resultant torque [12] (hence affect the dynamic of manipulator).

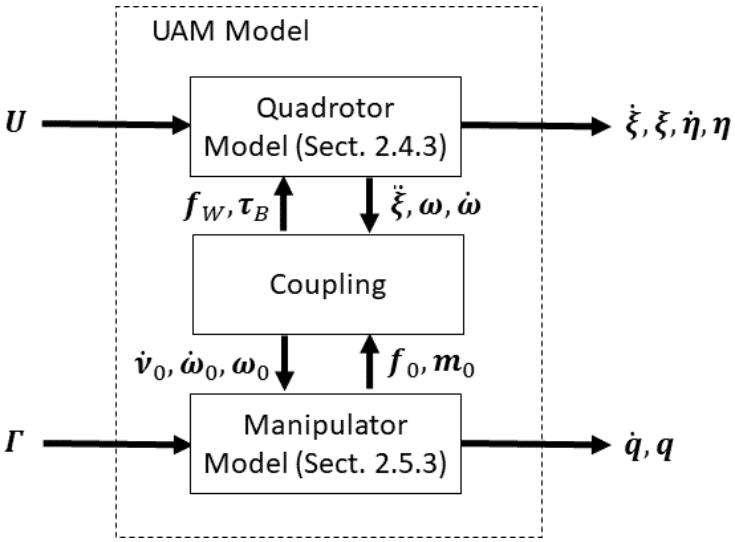



Figure 2.12 – Block diagram of the coupled system.

The force $\boldsymbol{f}_0$ and torque $\boldsymbol{m}_0$ applied to the COM of the quadrotor as external disturbance $\{\boldsymbol{f}_W, \boldsymbol{\tau}_B\}$ by the manipulator is obtained from force $\boldsymbol{f}_1$ and torque $\boldsymbol{m}_1$ of joint 1 with

$$\boldsymbol{f}_0 = {}^0\boldsymbol{R}_1\boldsymbol{f}_1 \tag{2.74}$$

$$\boldsymbol{m}_0 = {}^0\boldsymbol{R}_1\boldsymbol{m}_1 \tag{2.75}$$

and

$${}^0\boldsymbol{R}_1 = \begin{bmatrix} C(q_1) & -S(q_1) & 0 \\ S(q_1) & C(q_1) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.76}$$

where $q_1$ is the angle between link 1 and base frame of the manipulator arm.

For the reaction force $\boldsymbol{f}_W$ applied to the quadrotor COM by manipulator base, the coupling is expressed with respect to the inertial frame while the reaction torque $\boldsymbol{\tau}_B$ is expressed with respect to the body frame. Therefore, it is necessary to express $\boldsymbol{f}_0$ and $\boldsymbol{m}_0$ to the frame, it yields;

$$\boldsymbol{f}_W = {}^W\boldsymbol{R}_0\boldsymbol{f}_0 \tag{2.77}$$

$$\boldsymbol{\tau}_B = {}^B\boldsymbol{R}_0\boldsymbol{m}_0 \tag{2.78}$$

Consequently, the update dynamic equation of quadrotor model, accounted for reaction force $\boldsymbol{f}_W = [f_{W.x}\ f_{W.y}\ f_{W.z}]^T$ and torque $\boldsymbol{\tau}_B = [\tau_{B.x}\ \tau_{B.y}\ \tau_{B.z}]^T$ is then governed by [14]:

$$\ddot{x} = \frac{U_1}{m}(C\psi S\theta C\phi + S\psi S\phi) + \frac{f_{W.x}}{m} \tag{2.79}$$

$$\ddot{y} = \frac{U_1}{m}(S\psi S\theta C\phi - C\psi S\phi) + \frac{f_{W.y}}{m} \tag{2.80}$$

$$\ddot{z} = \frac{U_1}{m}C\theta C\phi - g + \frac{f_{W.z}}{m} \tag{2.81}$$

$$\dot{p} = \frac{U_2 + \tau_{B.x}}{I_{xx}} + \frac{I_{yy} - I_{zz}}{I_{xx}}qr \tag{2.82}$$

$$\dot{q} = \frac{U_3 + \tau_{B.y}}{I_{yy}} + \frac{I_{zz} - I_{xx}}{I_{yy}}pr \tag{2.83}$$

$$\dot{r} = \frac{U_4 + \tau_{B.z}}{I_{zz}} + \frac{I_{xx} - I_{yy}}{I_{zz}}pq \tag{2.84}$$

The coupling effect of the quadrotor and manipulator arm is addressed by treating quadrotor velocities and accelerations as the base velocities and accelerations of the manipulator. They are transmitted from the base to the terminal link of the manipulator affecting joint torques. Conversely, the reaction force and torque exerted on the center of mass of the quadrotor by the manipulator base are considered as external disturbances of the quadrotor. In this way, the dynamic interaction between the quadrotor and the manipulator arm is accounted for.

## 2.7 Differential kinematics

Kinematics is a part of mechanics concerned with the motion of a body without reference to the forces which cause the motion. Here, the differential kinematics relates the velocity of the end the effector and the velocities of quadrotor and manipulator joints of the whole system. Forward kinematics gives the end effector velocity in terms of quadrotor and manipulator joint velocities. While the inverse kinematic gives the quadrotor and manipulator joint velocities for

a desired end effector velocity. Notice that, with inverse kinematics there are often multiple different solutions.

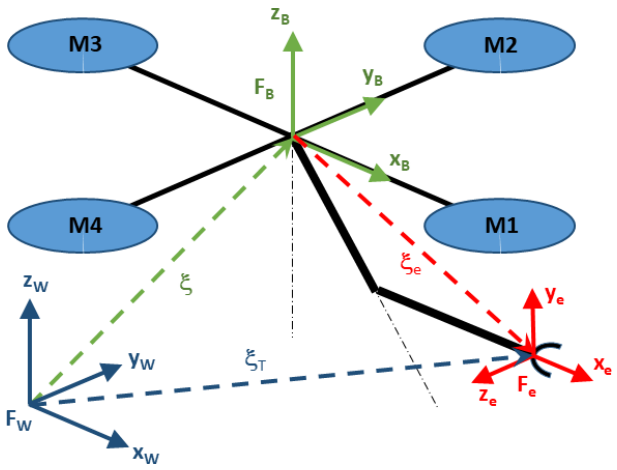



Figure 2.13 – Composition of three main position vector.

Consider the end effector frame $F_e{:}O_e - \{x_e, y_e, z_e\}$ as shown in Figure 2.13. The position vector $\boldsymbol{\xi}_T$ and the orientation, ${}^W\boldsymbol{R}_e$ of the end effector in the inertial frame can be stated as [18]

$$\boldsymbol{\xi}_T = \boldsymbol{\xi} + {}^W\boldsymbol{R}_B\, \boldsymbol{\xi}_e \tag{2.85}$$

$${}^W\boldsymbol{R}_e = {}^W\boldsymbol{R}_B\, {}^B\boldsymbol{R}_e \tag{2.86}$$

With $\boldsymbol{\xi}$ and ${}^W\boldsymbol{R}_B$ respectively the position vector and the orientation of quadrotor in inertial frame, while $\boldsymbol{\xi}_e$ and ${}^B\boldsymbol{R}_e$ are the position vector and orientation of end effector with respect to body frame $F_B$.

Differentiating equations (2.85) - (2.86), gives the linear velocity $\dot{\boldsymbol{\xi}}_T$ and the associated angular velocity $\boldsymbol{\omega}_T$ of the end effector in the inertial frame, $F_W$ as

$$\dot{\boldsymbol{\xi}}_T = \dot{\boldsymbol{\xi}} - \boldsymbol{S}({}^W\boldsymbol{R}_B\, \boldsymbol{\xi}_e)\, \boldsymbol{\omega} + {}^W\boldsymbol{R}_B\, \dot{\boldsymbol{\xi}}_e \tag{2.87}$$

$$\boldsymbol{\omega}_T = \boldsymbol{\omega} + {}^W\boldsymbol{R}_B\, \boldsymbol{\omega}_e \tag{2.88}$$

where $\boldsymbol{S}(\cdot)$ is a $(3 \times 3)$ skew symmetric matrix and $\boldsymbol{\omega}$ is the angular velocity of quadrotor in the inertial frame while $\boldsymbol{\omega}_e$ describes the angular velocity of the end effector in body frame, $F_B$. Equations (2.87) and (2.88) can be rewritten under matrix form as

$$\begin{bmatrix} \dot{\boldsymbol{\xi}}_T \\ \boldsymbol{\omega}_T \end{bmatrix} = \begin{bmatrix} \boldsymbol{I}_3 & -\boldsymbol{S}({}^W\boldsymbol{R}_B\, \boldsymbol{\xi}_e) \\ \boldsymbol{O}_3 & \boldsymbol{I}_3 \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{\xi}} \\ \boldsymbol{\omega} \end{bmatrix} + \begin{bmatrix} {}^W\boldsymbol{R}_B & \boldsymbol{O}_3 \\ \boldsymbol{O}_3 & {}^W\boldsymbol{R}_B \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{\xi}}_e \\ \boldsymbol{\omega}_e \end{bmatrix} \tag{2.89}$$

where $\boldsymbol{I}_3$ and $\boldsymbol{O}_3$ denote a $(3 \times 3)$ identity and null matrices respectively.

As the attitude and heading of this system is expressed in term of Euler angle $(\phi, \theta, \psi)$ a transformation is required. The transformation is between the angular velocity $\boldsymbol{\omega}$ and the Euler rate $\dot{\boldsymbol{\eta}}$ in a same frame and reads as [39]

$$\begin{aligned}\boldsymbol{\omega} &= \begin{bmatrix}0\\0\\\dot{\psi}\end{bmatrix} + \boldsymbol{R_z}(\psi)\begin{bmatrix}0\\\dot{\theta}\\0\end{bmatrix} + \boldsymbol{R_z}(\psi)\boldsymbol{R_y}(\theta)\begin{bmatrix}\dot{\phi}\\0\\0\end{bmatrix}\\ &= \begin{bmatrix}0 & -S\psi & C\psi C\theta\\0 & C\psi & S\psi C\theta\\1 & 0 & -S\theta\end{bmatrix}\begin{bmatrix}\dot{\psi}\\\dot{\theta}\\\dot{\phi}\end{bmatrix}\\ \boldsymbol{\omega} &= \boldsymbol{T}(\boldsymbol{\eta})\,\dot{\boldsymbol{\eta}}\end{aligned} \tag{2.90}$$

Therefore, the transfer matrix $\boldsymbol{T}(\boldsymbol{\eta})$ between the angular velocity $\boldsymbol{\omega}$ and the Euler rate $\dot{\boldsymbol{\eta}}$ in a same frame is given by

$$\boldsymbol{T}(\boldsymbol{\eta}) = \begin{bmatrix}0 & -S\psi & C\psi C\theta\\0 & C\psi & S\psi C\theta\\1 & 0 & -S\theta\end{bmatrix} \tag{2.91}$$

Then, substituting (2.90) into (2.89) for every angular velocity gives

$$\begin{bmatrix}\boldsymbol{I}_3 & \boldsymbol{O}_3\\\boldsymbol{O}_3 & \boldsymbol{T}(\boldsymbol{\eta}_T)\end{bmatrix}\begin{bmatrix}\dot{\boldsymbol{\xi}}_T\\\dot{\boldsymbol{\eta}}_T\end{bmatrix} = \begin{bmatrix}\boldsymbol{I}_3 & -\boldsymbol{S}({}^W\boldsymbol{R}_B\,\boldsymbol{\xi}_e)\\\boldsymbol{O}_3 & \boldsymbol{I}_3\end{bmatrix}\begin{bmatrix}\boldsymbol{I}_3 & \boldsymbol{O}_3\\\boldsymbol{O}_3 & \boldsymbol{T}(\boldsymbol{\eta})\end{bmatrix}\begin{bmatrix}\dot{\boldsymbol{\xi}}\\\dot{\boldsymbol{\eta}}\end{bmatrix} + \begin{bmatrix}{}^W\boldsymbol{R}_B & \boldsymbol{O}_3\\\boldsymbol{O}_3 & {}^W\boldsymbol{R}_B\end{bmatrix}\begin{bmatrix}\dot{\boldsymbol{\xi}}_e\\\boldsymbol{\omega}_e\end{bmatrix} \tag{2.92}$$

The relations of the velocities of the end effector $[\dot{\boldsymbol{\xi}}_e{}^T\ \boldsymbol{\omega}_e{}^T]^T$ with joint velocities $\dot{\boldsymbol{q}}$ are given by [35]:

$$\begin{bmatrix}\dot{\boldsymbol{\xi}}_e\\\boldsymbol{\omega}_e\end{bmatrix} = J(\boldsymbol{q})\,\dot{\boldsymbol{q}} \tag{2.93}$$

where $J(\boldsymbol{q})$ denotes a Jacobian matrix of manipulator joint velocities. Then, Equation (2.92) becomes

$$\begin{bmatrix}\boldsymbol{I}_3 & \boldsymbol{O}_3\\\boldsymbol{O}_3 & \boldsymbol{T}(\boldsymbol{\eta}_T)\end{bmatrix}\begin{bmatrix}\dot{\boldsymbol{\xi}}_T\\\dot{\boldsymbol{\eta}}_T\end{bmatrix} = \begin{bmatrix}\boldsymbol{I}_3 & -\boldsymbol{S}({}^W\boldsymbol{R}_B\,\boldsymbol{\xi}_e)\,\boldsymbol{T}(\boldsymbol{\eta})\\\boldsymbol{O}_3 & \boldsymbol{T}(\boldsymbol{\eta})\end{bmatrix}\begin{bmatrix}\dot{\boldsymbol{\xi}}\\\dot{\boldsymbol{\eta}}\end{bmatrix} + \begin{bmatrix}{}^W\boldsymbol{R}_B & \boldsymbol{O}_3\\\boldsymbol{O}_3 & {}^W\boldsymbol{R}_B\end{bmatrix}J(\boldsymbol{q})\,\dot{\boldsymbol{q}} \tag{2.94}$$

To make it more compressed, let

$$\boldsymbol{T}_T(\boldsymbol{\eta}_T)\dot{\boldsymbol{\kappa}}_T = J_B(\boldsymbol{\eta}, \boldsymbol{q})\dot{\boldsymbol{\kappa}} + J_e(\boldsymbol{\eta}, \boldsymbol{q})\dot{\boldsymbol{q}} \tag{2.95}$$

with

$$\dot{\boldsymbol{\kappa}}_T = \begin{bmatrix} \dot{\boldsymbol{\xi}}_T \\ \dot{\boldsymbol{\eta}}_T \end{bmatrix} \tag{2.96}$$

$$\dot{\boldsymbol{\kappa}} = \begin{bmatrix} \dot{\boldsymbol{\xi}} \\ \dot{\boldsymbol{\eta}} \end{bmatrix} \tag{2.97}$$

$$\boldsymbol{T}_T(\boldsymbol{\eta}_T) = \begin{bmatrix} \boldsymbol{I}_3 & \boldsymbol{O}_3 \\ \boldsymbol{O}_3 & \boldsymbol{T}(\boldsymbol{\eta}_T) \end{bmatrix} \tag{2.98}$$

$$J_B(\boldsymbol{\eta}, \boldsymbol{q}) = \begin{bmatrix} \boldsymbol{I}_3 & -\boldsymbol{S}({}^W\boldsymbol{R}_B\, \boldsymbol{\xi}_e)\, \boldsymbol{T}(\boldsymbol{\eta}) \\ \boldsymbol{O}_3 & \boldsymbol{T}(\boldsymbol{\eta}) \end{bmatrix} \tag{2.99}$$

$$J_e(\boldsymbol{\eta}, \boldsymbol{q}) = \begin{bmatrix} {}^W\boldsymbol{R}_B & \boldsymbol{O}_3 \\ \boldsymbol{O}_3 & {}^W\boldsymbol{R}_B \end{bmatrix} J(\boldsymbol{q}) \tag{2.100}$$

Quadrotor is an underactuated system indeed, only 4 independent control inputs are available to control the 6-DOF of the system. The position vector $\boldsymbol{\xi}$ and yaw angle $\psi$ are the available controlled parameters, while roll $\phi$ and pitch $\theta$ angles are the intermediate control inputs. Hence, vector $\boldsymbol{\kappa}$ can be rewritten as

$$\boldsymbol{\kappa} = \begin{bmatrix} \boldsymbol{\xi} \\ \phi \\ \theta \\ \psi \end{bmatrix} \tag{2.101}$$

and let

$$\boldsymbol{\alpha} = \begin{bmatrix} \boldsymbol{\xi} \\ \psi \end{bmatrix}; \boldsymbol{\beta} = \begin{bmatrix} \theta \\ \phi \end{bmatrix} \tag{2.102}$$

Then, one gets

$$\boldsymbol{T}_T(\boldsymbol{\eta}_T)\dot{\boldsymbol{\kappa}}_T = J_\alpha(\boldsymbol{\eta}, \boldsymbol{q})\dot{\boldsymbol{\alpha}} + J_\beta(\boldsymbol{\eta}, \boldsymbol{q})\dot{\boldsymbol{\beta}} + J_e(\boldsymbol{\eta}, \boldsymbol{q})\dot{\boldsymbol{q}} \tag{2.103}$$

where $J_\alpha(\boldsymbol{\eta}, \boldsymbol{q})$ comprises of first four columns of $J_B(\boldsymbol{\eta}, \boldsymbol{q})$ and $J_\beta(\boldsymbol{\eta}, \boldsymbol{q})$ comprises of last two columns of $J_B(\boldsymbol{\eta}, \boldsymbol{q})$. Merging $\boldsymbol{\alpha}$ and $\boldsymbol{q}$ so that $\boldsymbol{\zeta} = [\boldsymbol{\alpha}^T\ \boldsymbol{q}^T]^T$ is the vector of controller variables and $J_\zeta = [J_\alpha\ J_e]$, then the UAM forward kinematic becomes.

$$\dot{\boldsymbol{\kappa}}_T = \boldsymbol{T}_T(\boldsymbol{\eta}_T)^{-1}[J_\zeta(\boldsymbol{\zeta}, \boldsymbol{\beta})\dot{\boldsymbol{\zeta}} + J_\beta(\boldsymbol{\zeta}, \boldsymbol{\beta})\dot{\boldsymbol{\beta}}] \tag{2.104}$$

The modeling of the quadrotor UAV (Section 2.4) and the manipulator arm (Section 2.5) is discussed independently. The entire system of the aerial manipulator is designed by combining both models (quadrotor + manipulator) using a coupling algorithm (Section 2.6). Consequently, the coupling effect of the manipulator on the quadrotor is considered as an external disturbance, and vice versa. The whole system can be represented with the state vector defined as $[x\ y\ z\ \dot{x}\ \dot{y}\ \dot{z}\ \phi\ \theta\ \psi\ p\ q\ r\ q_1\ q_2\ \dot{q}_1\ \dot{q}_2]^T$ and the control input as $[U_1\ U_2\ U_3\ U_4\ \Gamma_1\ \Gamma_2]^T$. The reference trajectories for the quadrotor and manipulator are determined using differential kinematics (Section 2.7) based on the desired end effector trajectory.

Chapter 3

# Unmanned Aerial Manipulator Control

A control system is a combination of components and processes that act together to force the behavior of a dynamic system to a desired one. Its purpose is to maintain the system output that is the function of the state at a desired value (constant or time-varying), by continuously monitoring the system performance and then making necessary adjustments - that is the principle of the closed loop. The primary components of a control system include the controller, the actuators, the plant, the sensors and the feedback loop. The controller adjusts system inputs based on the error signal between the setpoint and the system output. Actuators convert control signals into physical actions for the plant; sensors measure relevant variables while the feedback loop continuously compares the output to the setpoint for adjustment. Control systems are classified with respect to their characteristics (static feedback, output feedback ...) and their design principles (linear, nonlinear, continuous, discrete ...). The choice of their architecture depends on specific factors related to the controlled system and the desired features (robust, finite time ...).

The control system of an aerial manipulator must regulate and coordinate the movement of both the aerial vehicle, such as a quadrotor, and the attached payload, typically a manipulator arm or gripper, to achieve desired tasks and objectives. In many cases, the control laws design is based on mathematical models of kinematics and dynamics in order to understand the aerial manipulator's behavior. Sensors, including cameras, depth sensors and IMUs, enable sensing of the environment and then to track the aerial manipulator's actual position and orientation [6, 12, 18, 44]. Motion planning algorithms generate feasible paths considering safety and obstacle avoidance. As shown in previous chapter trajectory generation produces desired motion profiles based on the planned paths. Control algorithms, such as linear proportional-integral-derivative [4, 11, 13] or nonlinear sliding mode control [23, 28, 33], are tuned in order to induce precise tracking of trajectories. Communication and integration with other systems facilitate coordinated operation, while safety measures and fault tolerance strategies confirm stability and reliability.

The objective of the control system for an aerial manipulator is to achieve precise and coordinated movements, to ensure stability and safety and efficiently perform the intended tasks. In this chapter, control solution, that are robust and adaptive, are presented in order to fulfill the objective with accuracy and robustness.

## 3.1 Sliding mode control

There is always a mismatch between the mathematical model on which the controller is design and the real behavior of the system, subject to uncertainties and unmodeled dynamics. Additionally, external disturbances are acting on the system. In order to address the inconsistencies arising from these disturbances and uncertainties in model dynamics, robust control methods, such as Sliding Mode Control (SMC), are required [45, 46, 47]. There exist other robust control method for uncertain/perturbed nonlinear systems, for example backstepping [6, 38, 48]. However sliding mode control is quite easy to design and implement, and adaptive version allows to have a quite reduced modeling effort. Sliding mode control is a robust control technique designed to achieve a desired behavior by guiding the system's state trajectory to reach a predefined manifold known as the sliding surface; once the trajectory has reached this manifold, the system is insensitive to uncertainties and perturbation. The fundamental concept behind sliding mode control is the development of discontinuous control laws that guide the system trajectory onto the sliding surface in a finite time and maintain them on this surface, regardless of uncertainties or disturbances.

In sliding mode control, the system dynamics are evolving in two phases: the sliding phase and the reaching phase [47]. The sliding phase is when the system trajectory precisely follows the sliding surface, forced by a discontinuous control law designed for robustness against uncertainties and disturbances. This ensures that the system trajectory remains on the sliding surface once this latter has been reached. Thus during reaching phase, the system trajectory converges in a finite time towards the sliding surface. Control actions in this phase aim to drive the system's state onto the sliding surface.

The main advantage of sliding mode control is its robustness to system uncertainties and disturbances. The control law is designed to actively compensate for uncertainties and to drive the system onto the sliding surface, where the control behavior is deterministic and system behavior insensitive to perturbation and uncertainties.

However, sliding mode control has some drawbacks as well. Due to its discontinuous nature, SMC can generate high-frequency control signals, leading to chattering phenomenon [46] that may cause wear and tear on actuators. Additionally, the discontinuous nature of sliding mode control can introduce high-frequency "noise" into the system, that can affect the performance or the stability.

Despite these challenges, sliding mode control has been successfully applied to a wide range of applications, including robotics, power electronics, aerospace systems, and automotive control. The design and implementation of sliding mode control typically involve careful selection of the sliding surface, control law design and consideration of practical implementation issues. SMC design consists of two steps. The first step is the design of sliding surface that has to be chosen such that the relative degree of the system versus the sliding variable equals 1. Then a discontinuous control law is design to force system trajectory onto sliding surface and to maintain onto it in spite of the presence of inconsistency.

### 3.1.1 Sliding surface design

In order to illustrate consider the following second order system (as can be modeled by a mechanical system)

$$\ddot{x} = u + f(x) \tag{3.1}$$

where $x$ is the variable, $u$ is the control input and $f(x)$ is an unknown disturbance assumed to be bounded, *i.e.*, $|f(x)| \leqslant \rho > 0$. Then, the state of the system is written as

$$\boldsymbol{X} = \begin{bmatrix} x \\ \dot{x} \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \tag{3.2}$$

where $x$ is supposed to be a displacement and $\dot{x}$ the associated velocity. The control objective is to force the variables $x$ to a desired value $x_d$. Defining the error signal $e = x - x_d$, define a sliding variable $\sigma$ such that when $\sigma = 0$, the control objective is achieved *i.e.* $e \to 0$. A standard sliding variable then is defined as [46].

$$\sigma = \dot{e} + \lambda e, \quad \lambda > 0 \tag{3.3}$$

This definition allows to fulfill two properties

- when $\sigma = 0$, $e \to 0$. Indeed, $\sigma = 0$ induces $\dot{e} + \lambda e = 0 \Rightarrow e(t) = e(t_0)\, exp^{(-\lambda(t-t_0))}$
- the selective degree of the system with respect to $\sigma$ equals 1. It is obvious given that $\sigma$ depends on $\dot{e}$, given (3.1), $\dot{\sigma}$ is depending on $u$.

### 3.1.2 Control law design

Once the sliding variables is defined, the objective is to design a discontinuous control that guarantee

- the finite time convergence towards the sliding surface $\sigma = 0$.
- once $\sigma = 0$ reached, the fact that the closed loop system trajectory is kept on the manifold.

Consider the time derivative of $\sigma$

$$\begin{aligned}\dot{\sigma} &= \ddot{e} + \lambda\dot{e} \\ &= \ddot{x} - \ddot{x}_d + \lambda(\dot{x} - \dot{x}_d) \\ &= u + f(x) - \ddot{x}_d + \lambda(\dot{x} - \dot{x}_d)\end{aligned} \tag{3.4}$$

Consider now a candidate Lyapunov function defined as

$$V = \frac{1}{2}\sigma^2 \tag{3.5}$$

This function fulfills $V(0) = 0$ and $V(\sigma) > 0 \; \forall \; \sigma \neq 0$. One sets

$$\dot{V} = \sigma\dot{\sigma} \tag{3.6}$$

To ensure the asymptotic convergence of $\sigma$, the time derivative of $V$ must be negative definite [46], *i.e.*

$$\dot{V} = \sigma\dot{\sigma} < 0 \quad \text{for } \sigma \neq 0 \tag{3.7}$$

one gets

$$\sigma\dot{\sigma} = \sigma(u + f(x) - \ddot{x}_d + \lambda(\dot{x} - \dot{x}_d)) \tag{3.8}$$

Assuming

$$u = \ddot{x}_d - \lambda(\dot{x} - \dot{x}_d) + \mu(\sigma)\,^1 \tag{3.9}$$

$\mu(\sigma)$ being defined in the sequel, one gets

$$\begin{aligned}\sigma\dot{\sigma} &= \sigma(f(x) + \mu(\sigma)) \\ &= \sigma f(x) + \sigma\mu(\sigma)\end{aligned} \tag{3.10}$$

As $f(x)$ is assumed to be bounded that is $|f(x)| \leqslant \rho$ then

$$\sigma\dot{\sigma} \leqslant \sigma\rho\,\mathrm{sign}(\sigma) + \sigma\mu(\sigma) \tag{3.11}$$

Selecting $\mu(\sigma) = -K\,\mathrm{sign}(\sigma)$, one gets

$$\begin{aligned}\sigma\dot{\sigma} &\leqslant \sigma\rho\,\mathrm{sign}(\sigma) - \sigma K\,\mathrm{sign}(\sigma) \\ &\leqslant -\sigma\,\mathrm{sign}(\sigma)(K - \rho) \\ &\leqslant -|\sigma|(K - \rho)\end{aligned} \tag{3.12}$$

1. This control law supposes that $\dot{x}$ is measured or obtained from $x$ thanks to a differentiator. Furthermore, the desired trajectories are supposed to be known

If $K > \rho$, then condition (3.7) is satisfied. Consequently, the control law $u$

$$u = \ddot{x}_d - \lambda(\dot{x} - \dot{x}_d) - K\text{sign}(\sigma) \tag{3.13}$$

ensures that $\sigma \to 0$ in spite of the perturbation $f(x)$. However, condition (3.7) formally does not ensure finite time convergence. Indeed, the so called "sliding condition" reads as

$$\dot{V} \leq -\eta|\sigma| = -\eta V^{1/2}\,, \quad \eta > 0 \tag{3.14}$$

In this case, the gain condition to ensure the finite time convergence and the establishment of a sliding mode, reads as

$$K \geq \eta + \rho \tag{3.15}$$

In this case, the reaching phase duration, *i.e.* the convergence time $t_r$ to reach $\sigma = 0$ is such that

$$t_r \leq \frac{|\sigma(0)|}{\eta} \tag{3.16}$$

**Remark.** *Given the maximum value of* $t_r$, $\frac{|\sigma(0)|}{\eta}$, *faster the convergence, bigger* $\eta$ *and greater the gain* $K$. *It is an important point to limit the chattering magnitude.*

### 3.1.3 Chattering reduction

The main drawback of SMC is the phenomenon called chattering [47, 49]. Chattering causes high frequency oscillations on the control input that leads to reduced accuracy and increases actuator mechanical wear and tear [47]. The chattering is due to the discontinuous sign function in the control input (3.13). There exist different solution to reduce chattering. Among them, one can replace the *sign* function by a continuous one. Thus approximate continuous function in a vicinity of the origin of the sliding surface [50], is the saturation function given by

$$\text{sat}(\sigma, \epsilon) = \begin{cases} \text{sign}(\sigma) & \text{if } |\sigma| > \epsilon \\ \dfrac{\sigma}{\epsilon} & \text{if } |\sigma| \leqslant \epsilon \end{cases} \tag{3.17}$$

where $\epsilon > 0$. However notice that this solution reduces the robustness and the accuracy of the closed loop system.

## 3.2 Twisting sliding mode control

As shown in the previous section, conventional sliding mode control suffers from the chattering phenomenon, which generates high-frequency oscillations that can cause mechanical wear, reduced accuracy and loss of robustness [47]. The chattering effect can be reduced by introducing

a smooth function, as described in Subsection 3.1.3. However, this smooth control function may cause the sliding variable and the state variable to converge to the vicinity of the origin instead of reaching zero in finite time [46]; in fact, around the sliding surface, the control has a linear behavior. As a consequence, the introduction of this smooth control function reduces robustness and results in a loss of accuracy. To address this issue, Levant introduced the concept of High-order Sliding Mode (HOSM) [51]. By a general point-of-view, this class of controller allows to force, not only the sliding variable $\sigma \to 0$, but also finite number of its time derivative. As a consequences, the accuracy is improved (higher the order of sliding mode, better the accuracy) and the chattering is reduced.

There are multiple algorithms available for implementing HOSM [46], with Second-order Sliding Mode controllers (2-SMC) being particularly useful in eliminating chattering and enhancing control performance, resulting in improved tracking capabilities. In the context of the second order system (3.1), in standard SMC, the control law is designed to drive sliding variable $\sigma$ to zero. In the case of 2-SMC, the primary objective extends beyond merely driving the sliding variable to zero. Instead, the goal is to bring both the sliding variable and its first time derivative to zero *i.e.* [47]

$$\sigma = \dot{\sigma} = 0 \tag{3.18}$$

Among the second-order sliding mode algorithms, options include suboptimal controllers [46], twisting controllers [46], and super-twisting controllers [46]. The Twisting Sliding Mode control (TWC), as described by Levant [51], is a discontinuous control method designed for a class of systems having a relative degree of either equal to 1 or 2 in relation to the sliding variable [47]. It is defined by the formula [46]

$$\mu = -(k_1 \text{sign}(\sigma) + k_2 \text{sign}(\dot{\sigma})) \ , \quad k_1 > k_2 > 0 \tag{3.19}$$

Assume that sliding variable $\sigma$ is twice differentiable with respect to time [46], then one gets

$$\ddot{\sigma} = h(x,t) + g(x,t)\mu \tag{3.20}$$

where the function $h(x,t)$ and $g(x,t)$ are some unknown smooth function that are bounded by positive constant $K_m$, $K_M$ and $C$ such that

$$\begin{aligned} 0 < K_m \leq g \leq K_M \ , \ |h| \leq C \\ K_m > 0, K_M > 0, C > 0 \end{aligned} \tag{3.21}$$

If $k_1$ and $k_2$ satisfy these conditions

$$\begin{aligned}(k_1 + k_2)K_m - C &> (k_1 - k_2)K_M + C \\ (k_1 - k_2)K_m &> C\end{aligned} \tag{3.22}$$

The TWC in Equation (3.19) guarantees the realization of 2-SMC in finite time. The twisting algorithm enforces the sliding variable $\sigma$ to have a relative degree of two in the control law, requiring knowledge of $\dot{\sigma}$. On the other hand, the super-twisting algorithm operates with a sliding variable of relative degree one and does not require knowledge of $\dot{\sigma}$.

## 3.3 Super-twisting sliding mode control

Super-twisting Sliding Mode control (STWC) is a control technique used for uncertain nonlinear systems to achieve robust and precise control in the presence of uncertainties and disturbances. STWC attenuates chattering effects, because it provides a continuous control signal.

Consider the following system

$$\dot{\sigma} = \mu + \psi \tag{3.23}$$

with perturbation $|\psi| < \Delta$ in the operating domain. The STWC algorithm reads as, the following control law $\mu$ that does not require the measurement of $\dot{\sigma}$, as proposed by Levant [51].

$$\begin{aligned}\mu &= -K_1|\sigma|^{1/2}\text{sign}(\sigma) + \nu \\ \dot{\nu} &= -K_2\text{sign}(\sigma)\end{aligned} \tag{3.24}$$

where $K_1$ and $K_2$ are positive gains. When $K_1$ and $K_2$ are tuned to adequate condition, the sliding variable $\sigma$ and its first time derivative $\dot{\sigma}$ converge to zero in finite time. This convergence of system trajectories to a vicinity of $(\sigma, \dot{\sigma}) = (0, 0)$ in a finite time can be fulfill if the gains $K_1$ and $K_2$ satisfy either the following conditions

— Conditions from Moreno *et al.* (2014) [53]

$$K_2 > \Delta, \qquad K_1 > \frac{K_2 + \Delta}{\sqrt{K_2 - \Delta}}$$

— Conditions from Kumar P *et al.* (2017) [54]

$$K_2 > \Delta, \qquad K_1 > 1.8\sqrt{K_2 + \Delta}$$

— Conditions from Chalanga *et al.* (2017) [55]

$$K_2 > \Delta, \qquad K_1 > 1.4\sqrt{K_2 + \Delta}$$

If $K_1$ and $K_2$ fulfill these condition, then a second order sliding mode is establish in a finite time, *i.e.* there exists a finite time $t_c$ such that for $t \geq t_c$, $\sigma = \dot{\sigma} = 0$. This fact is called "ideal sliding mode". In practical cases, when there is a sampling period $\tau \neq 0$, then a "real sliding mode" is establish, *i.e.* there exist a time $t_c$ such that, for $t \geq t_c$, $|\sigma| \leq \Omega\tau$. Notice that $K_1$ and $K_2$ depend on the bound of the perturbation. So, selecting gains based on the above conditions requires knowledge of the disturbance bound $\Delta$, which is often difficult to precisely determine [52]. As a consequence in many cases, if an overestimated bound is used this leading to large gains and significant chattering. Therefore, there is a significant interest in reducing the controller gains to attenuate chattering and simplify the identification task (eliminating the need for information on the disturbance bound $\Delta$ to tune the gains).

The super-twisting algorithm is a continuous law providing faster convergence and better tracking accuracy. Unlike the twisting controller, the super-twisting algorithm does not require knowledge of the derivative of the sliding variable. This makes it more practical and applicable to various systems where obtaining accurate derivative information may be challenging or not feasible. STWC exhibits robustness by effectively handling uncertainties and disturbances within the system. This robustness ensures system stability and enables the attainment of accurate tracking performance even in the presence of such uncertainties and disturbances. Then, the inclusion of the super-twisting term significantly reduces chattering, leading to a smoother control action and improved overall system behavior.

By replacing Equation (3.24) into the control law $u$ in Equation (3.9), the sliding variable $\sigma$ is forced to zero in the presence of bounded disturbance in finite time given by

$$u = \ddot{x}_d - \lambda(\dot{x} - \dot{x}_d) - K_1|\sigma|^{1/2}\text{sign}(\sigma) - \int_0^t K_2\text{sign}(\sigma)dT \tag{3.25}$$

## 3.4 Adaptive Gain

In sliding mode control, the adaptive gain refers to a mechanism that dynamically adjusts the control gains based on the system's behavior and operating conditions. It is used to improve the performance and robustness of the control system by adapting the control parameters in real-time. The adaptive gain in sliding mode control is typically implemented through an adaptive law or algorithm that continuously monitors the system's response and updates the control gains accordingly. This adaptive law uses feedback signals, error measurements or other system variables to estimate the system's uncertainties and disturbances. Based on these estimates, the adaptive law modifies the control gains to compensate for the uncertainties and disturbances and achieve desired control objectives. The purpose of using adaptive gains in sliding mode control is to enhance the control system's ability to handle uncertainties and disturbances. By adapting the control gains, the control system can maintain stability, improve tracking accuracy

and reduce the effects of uncertainties and disturbances.

### 3.4.1 Adaptive Gain Sliding Mode Control

Another drawback of conventional SMC is that its requires the knowledge of uncertainties bound ($\rho_*$) which often leads to over-estimated bounds. A way to counteract this drawback is to ensure a dynamical adaptation of the control gain to achieve sliding mode condition and sufficient to overcome the unknown uncertainties. Therefore, a value of $K_*$ for a dynamically tuned Adaptive Gain Sliding Mode control (SMCA) is given as [49]:

$$\dot{K}_* = \begin{cases} \bar{K}_* \cdot \text{sign}(|\sigma_*| - \mu_*) & \text{if } K_* > \alpha_* \\ \alpha_* & \text{if } K_* \leqslant \alpha_* \end{cases} \tag{3.26}$$

with $\bar{K}_* > 0$, $K_{0*} > 0$, $\mu_* > 0$ and $\alpha_* > 0$. The parameter $\alpha_*$ is to ensures that $K_*$ is always positive. $\bar{K}_*$ determine the gain dynamics that is how fast it will react in the presence of disturbance while $\mu_*$ acts as the accuracy or sensitivity of the gain to the deviation of $\sigma_*$ from zero. For adaptive gain law (3.26), the gain $K_*$ will increase at constant rate to force $\sigma_*$ to zero until $|\sigma_*| < \mu_*$, then the gain decreases and is kept at the smallest level as long as $\sigma_*$ remain within $\mu_*$ or close to zero. With this adaptive control gain, the knowledge of uncertainties bound is not required as in Equation (3.12) for conventional SMC. Furthermore, the adaptation gain allows to have the "just sufficient" gain that will help reducing the chattering effect.

### 3.4.2 Adaptive Gain Super-twisting Sliding Mode Control

The computation of gain $K_{1*}$ and $K_{2*}$ requires the knowledge of the disturbance bound as discuss in Chalanga *et al.* [55], Moreno *et al.* [53] and Kumar *et al.* [54]. This value is difficult to obtain and will also leads to over-estimated. A way to counteract this is to conform to a dynamical adaptation of the control gain that do not require any knowledge of the disturbance bound and sufficient to overcome the unknown uncertainties. Therefore, a value of $K_{1*}$ and $K_{2*}$ for a dynamically tuned Adaptive-gain Super-twisting Sliding Mode control (STWCA) is given by [52]:

$$\dot{K}_{1*} = \begin{cases} \dfrac{\alpha_*}{|\dot{\sigma}_*| + \epsilon_*} & \text{if } |\sigma_*| > \epsilon_* \\ -K_{1*} & \text{if } |\sigma_*| \leqslant \epsilon_* \end{cases} \tag{3.27}$$

$$\dot{K}_{2*} = \begin{cases} \dfrac{\alpha_*}{2|\sigma_*|^{1/2}} & \text{if } |\sigma_*| > \epsilon_* \\ -K_{2*} & \text{if } |\sigma_*| \leqslant \epsilon_* \end{cases} \tag{3.28}$$

$$\tag{3.29}$$

where $\alpha_*$ and $\epsilon_*$ are some positive design parameter, while $\dot{\sigma}_*$ it is estimated from sliding variable, $\sigma_*$ in Equation (3.3) fed to a first order differentiator.

The concept of this adaptation law is straightforward: the gains $K_{1*}$ and $K_{2*}$ increase as long as the desired accuracy is not achieved, indicated by $|\sigma_*| > \epsilon_*$, where $\epsilon_*$ defines the desired accuracy level. Once $|\sigma_*| \leqslant \epsilon_*$ the system has converged and the gains are decreased, fulfilling the objective. However, significant disturbances can cause the system trajectories to deviate from the convergence domain. In such cases, the adaptation law rapidly increases both gains $K_{1*}$ and $K_{2*}$ in a short time bringing the system trajectories back to the domain $|\sigma_*| \leqslant \epsilon_*$. Choosing a small value for the parameter $\epsilon_*$ is appropriate and the parameter $\alpha_*$ is chosen to directly influence the dynamics of the gains.

## 3.5 Application to the unmanned aerial manipulator control

In this section, the previous robust control laws are applied for the control of the unmanned aerial manipulator describe in Chapter 2.

### 3.5.1 General architecture

The proposed control strategy for the unmanned aerial manipulator consists of two stages, as depicted in Figure 3.1. The first stage (Stage 1 Figure 3.1) employs a kinematic motion algorithm called Closed-loop Inverse Kinematics (CLIK) being defined in the sequel. This algorithm calculates the reference path and trajectory for the quadrotor and the manipulator arm joints from desired end effector path and trajectory. These computed values serve as references for the next stage. In the second stage (Stage 2 Figure 3.1), the control laws are implemented, that consist of two independent controllers: the first one for the quadrotor and the second one for the manipulator. The objective of these controllers is to track the references provided by Stage 1. The controller of Stage 2 can have different forms in this work, they are based on linear approaches (PID) that are the baseline controllers used to compare performances of other controller based on sliding mode theory.

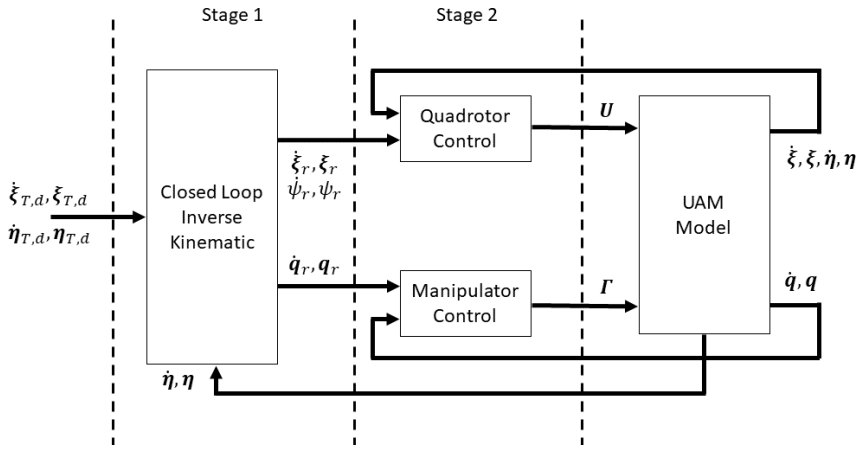



Figure 3.1 – Block scheme of UAM control architecture. Refer Appendix A.7 for symbol description.

### 3.5.2 Closed loop inverse kinematic

Given the desired end effector trajectory ($\boldsymbol{\kappa}_{T.d} = \begin{bmatrix} \boldsymbol{\xi}_{T.d} \\ \boldsymbol{\eta}_{T.d} \end{bmatrix}$) and its derivative ($\dot{\boldsymbol{\kappa}}_{T.d} = \begin{bmatrix} \dot{\boldsymbol{\xi}}_{T.d} \\ \dot{\boldsymbol{\eta}}_{T.d} \end{bmatrix}$), inverse kinematic algorithm determined an appropriate quadrotor and manipulator reference trajectory ($\boldsymbol{\zeta}_r = \begin{bmatrix} \boldsymbol{\xi}_r \\ \boldsymbol{\psi}_r \\ \boldsymbol{q}_r \end{bmatrix}$) and its derivative ($\dot{\boldsymbol{\zeta}}_r = \begin{bmatrix} \dot{\boldsymbol{\xi}}_r \\ \dot{\boldsymbol{\psi}}_r \\ \dot{\boldsymbol{q}}_r \end{bmatrix}$) for which the end effectors move to obtain the desired response. A reliable inverse kinematic solution is necessary to place the end effector at a specific position and orientation to perform a particular tasks. A widely implemented approach to calculate the differential kinematic solution that can be used in real-time is based on the Jacobian of the robot [57] as discussed in Section 2.7.

Recall from Section 2.7, the forward kinematic Equation (2.104) is considered to derive an inverse kinematic for computing the UAM motion references as follows

$$\dot{\boldsymbol{\zeta}} = \boldsymbol{J}_{\zeta}(\boldsymbol{\zeta}, \boldsymbol{\beta})^{-1} \left( \boldsymbol{T}_T(\boldsymbol{\eta}_T) \, (\dot{\boldsymbol{\kappa}}_T) - \, \boldsymbol{J}_{\beta}(\boldsymbol{\zeta}, \boldsymbol{\beta}) \, \dot{\boldsymbol{\beta}} \right) \tag{3.30}$$

where $\boldsymbol{\beta} = \begin{bmatrix} \theta \\ \phi \end{bmatrix}$ and $\boldsymbol{T}_T(\boldsymbol{\eta}_T)$ is a transfer matrix from Euler rate to angular velocity. While $\boldsymbol{J}_{\zeta}$ and $\boldsymbol{J}_{\beta}$ are the Jacobian matrices. This inverse kinematic solution is then integrated using a numerical integration technique to determine the reference position and orientation ($\boldsymbol{\zeta}$) of the system. Considering Figure 3.2 then the inverse kinematic solution (3.30) is rewritten with feedback gain as [18].

$$\dot{\boldsymbol{\zeta}}_r = \boldsymbol{J}_{\zeta}^{\dagger}(\boldsymbol{\zeta}_r, \boldsymbol{\beta}) \left( \boldsymbol{T}_T(\boldsymbol{\eta}_{T.r}) \, (\dot{\boldsymbol{\kappa}}_{T.d} + \boldsymbol{K}\boldsymbol{e}) - \, \boldsymbol{J}_{\beta}(\boldsymbol{\zeta}_r, \boldsymbol{\beta}) \, \dot{\boldsymbol{\beta}} \right) \tag{3.31}$$

where $\boldsymbol{J}_{\zeta}^{\dagger} = \boldsymbol{J}_{\zeta}{}^{T}(\boldsymbol{J}_{\zeta}\boldsymbol{J}_{\zeta}{}^{T} + \lambda^{2}\boldsymbol{I})^{-1}$ is a numerical approximate solution of the inverse Jacobian

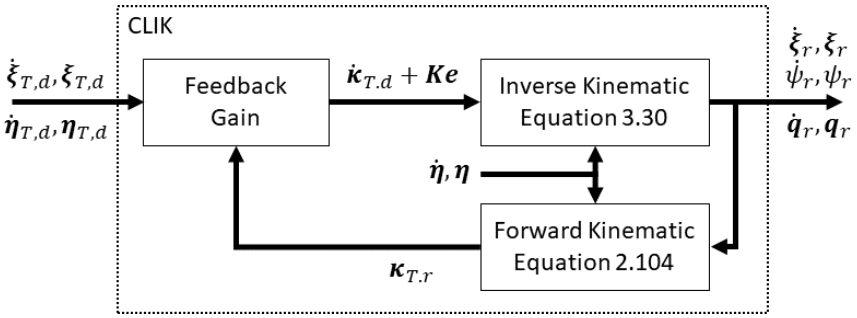



Figure 3.2 – Closed-loop inverse kinematic block diagram of UAM.

of $\boldsymbol{J}_{\zeta}$ named least damped squares methods [58] with $\lambda$ in the equation is a non-zero damping constant.

Since the least damped squares algorithm is a numerical approximation of the inverse Jacobian solution, the inverse kinematic has an inversion error. The inversion error can be reduced by introducing a feedback gain $\dot{\boldsymbol{\kappa}}_{T.d} + \boldsymbol{K}\boldsymbol{e}$ to improve the tracking performance referred to as the closed-loop inverse kinematics algorithm [18]. $\boldsymbol{K}$ here is a symmetric positive definite gain matrix, $\boldsymbol{e} = \boldsymbol{\kappa}_{T.d} - \boldsymbol{\kappa}_{T.r}$ is the kinematic inversion error between desired ($\boldsymbol{\kappa}_{T.d}$) and reference ($\boldsymbol{\kappa}_{T.r}$) end effector position and orientation value and $\boldsymbol{\kappa}_{T.r}$ is calculated using forward kinematic Equation (2.104) from reference value $\boldsymbol{\zeta}_{r}$ and measured value $\boldsymbol{\beta}$.

### 3.5.3 Quadrotor control

The control for the quadrotor is designed hierarchically, utilizing both outer position control and inner attitude control loops, as illustrated in Figure 3.3. The primary objective of this controller is to ensure the stabilization or the good tracking of the quadrotor's position, altitude, attitude and heading, such that the quadrotor is a stable platform for the manipulator arm. The outer position controller is responsible for accurately tracking the quadrotor's reference position along the $x$-axis for back-and-forth movement and along the $y$-axis for side-to-side movement. It computes the desired roll angle ($\phi_r$) and pitch angle ($\theta_r$) for the inner attitude control. The inner attitude control loop focuses on tracking the reference altitude along the $z$-axis ($z_r$), the desired attitude ($\phi_r$ and $\theta_r$ about the $x$-axis and $y$-axis respectively) and the reference heading ($\psi_r$ about the $z$-axis). By processing these values, the inner attitude controller determines the appropriate input force and torque, denoted as $\boldsymbol{U}$, required for the quadrotor's four motors. This control action allows the quadrotor to achieve the desired altitude, attitude, and heading.

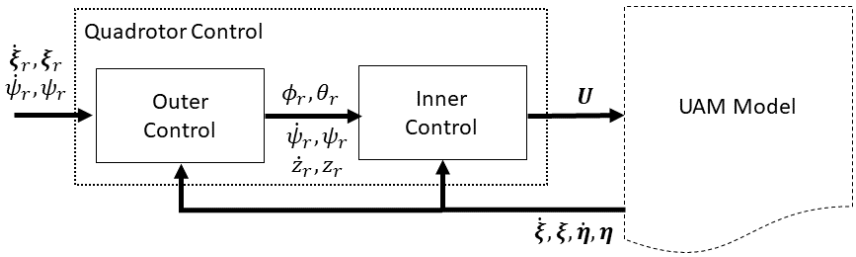



Figure 3.3 – Block scheme of quadrotor hierarchical control architecture.

### 3.5.4 Manipulator control

Controlling the manipulator arm is viewed as a Multi-input Multi-output (MIMO) control problem, where the control law needs to compute a vector of joint actuator torque, $\boldsymbol{\Gamma}$. The control scheme is divided into two main components: feedback linearization and feedback control, as depicted in Figure 3.4. The feedback linearization phase involves the linearization and decoupling of the dynamic model, utilizing the technique of feedback linearization to compensate for the non-linearities present in the robot's dynamics [35, 43, 56]. This stage takes advantage of the inverse dynamic equation of the manipulator arm as discuss in Subsection 2.5.2. On the other hand, the feedback control, that is the "heart" of the control scheme, computes the control input signal, $\boldsymbol{\nu}_q$, based on the manipulator's reference trajectory $\{\boldsymbol{q}_r, \dot{\boldsymbol{q}}_r\}$. This part of manipulator control is based, in the sequel, on PID and sliding mode control strategies. This term $\boldsymbol{\nu}_q$ is then fed into the inverse dynamic equation of the feedback linearization, resulting in the determination of the control torque, $\boldsymbol{\Gamma}$, for the manipulator joints' actuators. This control scheme is commonly referred to as computed torque control or inverse dynamic control [35, 43].

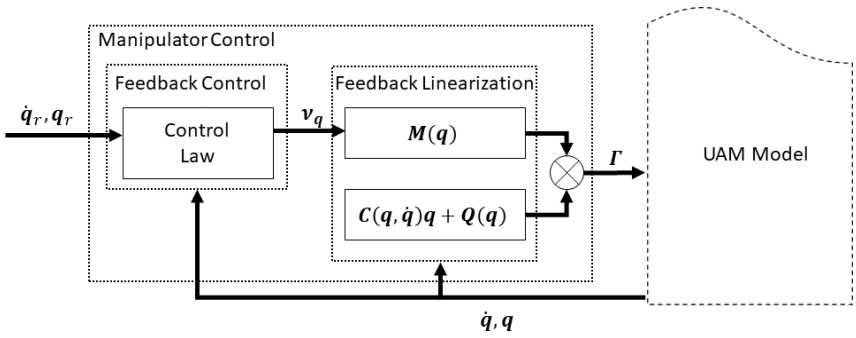



Figure 3.4 – Block scheme of manipulator control architecture.

### 3.5.5 Quadrotor control design

As introduced in the previous subsection, quadrotor altitude, attitude, and heading control refer to the control strategies used to regulate the vertical position (altitude), orientation

(attitude), and direction (heading) of a quadrotor UAV. These control laws ensure that the quadrotor maintains its desired altitude, orientation, and heading during flight, enabling it to perform various tasks with stability and precision.

Altitude control focuses on maintaining the quadrotor's desired vertical position. It involves adjusting the collective thrust generated by the rotors to counteract gravity and achieve the desired altitude. In real situation, altitude control relies on altitude sensors or altimeters to measure the quadrotor's height accurately.

Attitude control involves regulating the quadrotor's orientation in terms of pitch and roll. While the $x$ and $y$ positions cannot be directly controlled by the input forces and torques, these back-and-forth and side-to-side movements are controlled through pitch and roll angles, respectively. Pitch refers to the forward or backward tilt, while roll refers to the right or left tilt. Attitude control ensures that the quadrotor maintains the desired orientation throughout its flight. It utilizes sensors such as gyroscopes, accelerometers, or IMU to measure the quadrotor's current attitude accurately. Control algorithms are employed to compute appropriate torques and adjust the rotor speeds to achieve the desired attitude.

Heading control focuses on maintaining the quadrotor's desired direction or heading during flight. It involves regulating the quadrotor's yaw orientation, which represents the rotation about the vertical axis. Heading control ensures that the quadrotor is aligned with the desired direction. It relies on sensors such as magnetometers to measure the quadrotor's current heading relative to the Earth's magnetic field. The control algorithms adjust the yaw torque applied to the quadrotor's rotors to align it with the desired heading.

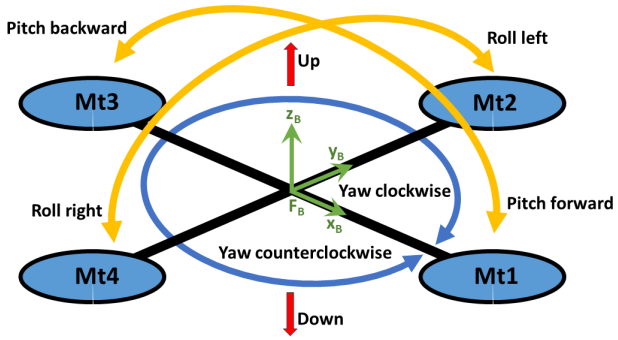



Figure 3.5 – Altitude control handles up and down vertical movements. Attitude control deal with quadrotor orientation, enabling it to pitch forward or backward and roll right or left. Heading control position the quadrotor in the desired direction through yaw adjustment, either clockwise or counterclockwise.

Combining altitude, attitude, and heading control, quadrotor can maintain stable flight and accurately perform tasks such as aerial photography, surveillance, delivery, or search and rescue operations. These controllers are components of the inner control system shown in Figure 3.3.

The control algorithms and strategies employed in quadrotor altitude, attitude and heading control ensure that the UAV responds to external disturbances, maintains its desired position and orientation, and achieves the required flight performance. In the following subsection, we will delve into the detailed design of these control laws.

### 3.5.6 Quadrotor simplified model

The kinematic model (2.15 - 2.20) and dynamic model (2.79 - 2.84) discuss in Subsection 2.4.1 and Subsection 2.6.2 describe the full model of the quadrotor system. Simplifying the models reduces the complexity of the control law and calculations involved. This can lead to faster computation and implementation of the control algorithm, making it more suitable for real-time control applications. A simplification is made by considering the behavior of the system around an operating point. This latter corresponds to the quadrotor nominal hover state. In other words, in hover state [2], $\phi = \theta = 0$, $\psi = \psi_0$, $\dot{\phi} = \dot{\theta} = \dot{\psi} = 0$, $\boldsymbol{\xi} = \boldsymbol{\xi}_0$ and $\ddot{\boldsymbol{\xi}} = \dot{\boldsymbol{\xi}} = 0$, given that the roll and pitch angles are small, one has $C\theta \approx 1$, $C\phi \approx 1$, $S\theta \approx \theta$ and $S\phi \approx \phi$ [42, 41]. Applying this approximation to transfer matrix Equation (2.10) gives [3] [40, 41]

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} \approx \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{3.32}$$

The quadrotor is underactuated and has 4 control inputs. As a consequence four variables, namely altitude ($z$), roll ($\phi$), pitch ($\theta$), and heading ($\psi$), are controlled. Combining Equation (3.32) with Equations (2.43 - 2.46), the dynamic equation used for synthesizing the quadrotor control law for the UAM can be expressed as follows [28, 33, 40]

$$\ddot{z} = \frac{U_1}{m} - g + \delta_z \tag{3.33}$$

$$\ddot{\phi} = \frac{U_2}{I_{xx}} + \frac{I_{yy} - I_{zz}}{I_{xx}}\dot{\theta}\dot{\psi} + \delta_\phi \tag{3.34}$$

$$\ddot{\theta} = \frac{U_3}{I_{yy}} + \frac{I_{zz} - I_{xx}}{I_{yy}}\dot{\phi}\dot{\psi} + \delta_\theta \tag{3.35}$$

$$\ddot{\psi} = \frac{U_4}{I_{zz}} + \frac{I_{xx} - I_{yy}}{I_{zz}}\dot{\phi}\dot{\theta} + \delta_\psi \tag{3.36}$$

where $\delta_z$, $\delta_\phi$, $\delta_\theta$ and $\delta_\psi$ are the external disturbance acted on altitude, roll, pitch and yaw respectively.

Notice that the system is nonlinear and can be viewed as affine in control input. Indeed, one

2. $\phi$, $\theta$ and $\psi$ are quadrotor roll, pitch and yaw angle.
3. $p$, $q$ and $r$ are the angular velocities of quadrotor about $x_B$-axis, $y_B$-axis and $z_B$-axis of the body frame $F_B$ respectively.

has

$$\begin{bmatrix} \ddot{z} \\ \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} -g \\ \frac{I_{yy}-I_{zz}}{I_{xx}}\dot{\theta}\dot{\psi} \\ \frac{I_{zz}-I_{xx}}{I_{yy}}\dot{\phi}\dot{\psi} \\ \frac{I_{xx}-I_{yy}}{I_{zz}}\dot{\phi}\dot{\theta} \end{bmatrix} + \begin{bmatrix} \frac{1}{m} & 0 & 0 & 0 \\ 0 & \frac{1}{I_{xx}} & 0 & 0 \\ 0 & 0 & \frac{1}{I_{yy}} & 0 \\ 0 & 0 & 0 & \frac{1}{I_{zz}} \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} \tag{3.37}$$

### 3.5.7 Quadrotor altitude control

Consider the quadrotor altitude error ($e_z = z - z_r$) that is the difference between the measured altitude $z$ and the reference altitude $z_r$. It can be noted that relative degree of the system (3.33) - (3.36) with $e_z$ as output equals 2. It means that first order sliding mode control and super-twisting control can be applied if the sliding variable defined as

$$\sigma_z = \dot{e}_z + \lambda_z e_z \tag{3.38}$$

with $e_z = z - z_r$ and $\lambda_z > 0$. Consider $U_1$ reading as

$$U_1 = m\Big(\ddot{z}_r - \lambda_z(\dot{z} - \dot{z}_r) - K_z\,\mathrm{sign}(\sigma_z) + g\Big) \tag{3.39}$$

Then from (3.33), one gets

$$\ddot{z} = \ddot{z}_r - \lambda_z(\dot{z} - \dot{z}_r) - K_z\,\mathrm{sign}(\sigma_z) + \delta_z \tag{3.40}$$

that gives

$$\dot{\sigma}_z = -K_z\,\mathrm{sign}(\sigma_z) + \delta_z \tag{3.41}$$

Then, from sliding mode theory, a first order sliding mode is established in a finite time if $K_z > |\delta_z| + \eta_z$. In order to reduce chattering effect, a controller based on super-twisting is proposed. By matching Equation (3.33) with STWC Equation (3.25) for altitude control gives

$$U_1 = m\Big(\ddot{z}_r - \lambda_z(\dot{z} - \dot{z}_r) - K_{1z}|\sigma_z|^{1/2}\mathrm{sign}(\sigma_z) - \int_0^t K_{2z}\mathrm{sign}(\sigma_z)dT + g\Big) \tag{3.42}$$

that gives

$$\dot{\sigma}_z = -K_{1z}|\sigma_z|^{1/2}\mathrm{sign}(\sigma_z) \tag{3.43}$$

### 3.5.8 Quadrotor attitude and heading control

The attitude (roll and pitch) and heading (yaw) control combine in forcing the errors between the desired roll ($\phi_r$), the desired pitch ($\theta_r$) and the reference yaw ($\psi_r$) and the measured values

$\phi$, $\theta$ and $\psi$ respectively.

### Roll control

Roll controller is developed to control the Equation (3.34) determines the roll angle $\phi$ of the quadrotor forcing by the input $U_2$. Considering the roll error $e_\phi = \phi - \phi_r$ and inspired by the previous controller, define the sliding variable $\sigma_\phi = \dot{e}_\phi + \lambda_\phi e_\phi$. Then, the following controller

$$U_2 = I_{xx}\Big(\ddot{\phi}_r - \lambda_\phi(\dot{\phi} - \dot{\phi}_r) - K_\phi \text{sign}(\sigma_\phi)\Big) - (I_{yy} - I_{zz})\dot{\theta}\dot{\psi} \tag{3.44}$$

then from 3.34

$$\ddot{\phi} = \ddot{\phi}_r - \lambda_\phi(\dot{\phi} - \dot{\phi}_r) - K_\phi \text{sign}(\sigma_\phi) + \delta_\phi \tag{3.45}$$

that gives

$$\dot{\sigma}_\phi = -K_\phi \text{sign}(\sigma_\phi) + \delta_\phi \tag{3.46}$$

with adequate tuning of $K_\phi$, *i.e.*

$$K_\phi > |\delta_\phi| + \eta_\phi \quad \eta_\phi > 0 \tag{3.47}$$

$\sigma_\phi$ reaches 0 in a finite time. Then $e_\phi$ exponentially tends to 0 in a time depending on $\eta_\phi$. By a similar way than previously, a STWC based controller is designed as

$$U_2 = I_{xx}\Big(\ddot{\phi}_r - \lambda_\phi(\dot{\phi} - \dot{\phi}_r) - K_{1\phi}|\sigma_\phi|^{1/2}\text{sign}(\sigma_\phi) - \int_0^t K_{2\phi}\text{sign}(\sigma_\phi)dT\Big) - (I_{yy} - I_{zz})\dot{\theta}\dot{\psi} \tag{3.48}$$

with $\sigma_\phi$ and $\dot{\sigma}_\phi$ converging towards 0 in a finite time if $K_{1\phi}$ and $K_{2\phi}$ are correctly tuned.

### Pitch Control

Pitch controller is developed to control the Equation (3.35) determines the pitch angle $\theta$ of the quadrotor forcing by the input $U_3$. Considering the pitch error $e_\theta = \theta - \theta_r$ and inspired by the previous controller, define the sliding variable $\sigma_\theta = \dot{e}_\theta + \lambda_\theta e_\theta$. Then, the following controller

$$U_3 = I_{yy}\Big(\ddot{\theta}_r - \lambda_\theta(\dot{\theta} - \dot{\theta}_r) - K_\theta \text{sign}(\sigma_\theta)\Big) - (I_{zz} - I_{xx})\dot{\phi}\dot{\psi} \tag{3.49}$$

then from 3.35

$$\ddot{\theta} = \ddot{\theta}_r - \lambda_\theta(\dot{\theta} - \dot{\theta}_r) - K_\theta \text{sign}(\sigma_\theta) + \delta_\theta \tag{3.50}$$

that gives

$$\dot{\sigma}_\theta = -K_\theta \text{sign}(\sigma_\theta) + \delta_\theta \tag{3.51}$$

with adequate tuning of $K_\theta$, *i.e.*

$$K_\theta > |\delta_\theta| + \eta_\theta \quad \eta_\theta > 0 \tag{3.52}$$

$\sigma_\theta$ reaches 0 in a finite time. Then $e_\theta$ exponentially tends to 0 in a time depending on $\eta_\theta$. By a similar way than previously, a STWC based controller is designed as

$$U_2 = I_{yy}\Big(\ddot{\theta}_r - \lambda_\theta(\dot{\theta} - \dot{\theta}_r) - K_{1\theta}|\sigma_\theta|^{1/2}\text{sign}(\sigma_\theta) - \int_0^t K_{2\theta}\text{sign}(\sigma_\theta)dT\Big) - (I_{zz} - I_{xx})\dot{\phi}\dot{\psi} \tag{3.53}$$

with $\sigma_\theta$ and $\dot{\sigma}_\theta$ converging towards 0 in a finite time if $K_{1\theta}$ and $K_{2\theta}$ are correctly tuned.

**Yaw Control**

Yaw controller is developed to control the Equation (3.36) determines the yaw angle $\psi$ of the quadrotor forcing by the input $U_4$. Considering the roll error $e_\psi = \psi - \psi_r$ and inspired by the previous controller, define the sliding variable $\sigma_\psi = \dot{e}_\psi + \lambda_\psi e_\psi$. Then, the following controller

$$U_4 = I_{zz}\Big(\ddot{\psi}_r - \lambda_\psi(\dot{\psi} - \dot{\psi}_r) - K_\psi\text{sign}(\sigma_\psi)\Big) - (I_{xx} - I_{yy})\dot{\phi}\dot{\theta} \tag{3.54}$$

then from 3.36

$$\ddot{\psi} = \ddot{\psi}_r - \lambda_\psi(\dot{\psi} - \dot{\psi}_r) - K_\psi\text{sign}(\sigma_\psi) + \delta_\psi \tag{3.55}$$

that gives

$$\dot{\sigma}_\psi = -K_\psi\text{sign}(\sigma_\psi) + \delta_\psi \tag{3.56}$$

with adequate tuning of $K_\psi$, *i.e.*

$$K_\psi > |\delta_\psi| + \eta_\psi \quad \eta_\psi > 0 \tag{3.57}$$

$\sigma_\psi$ reaches 0 in a finite time. Then $e_\psi$ exponentially tends to 0 in a time depending on $\eta_\psi$. By a similar way than previously, a STWC based controller is designed as

$$U_2 = I_{zz}\Big(\ddot{\psi}_r - \lambda_\psi(\dot{\psi} - \dot{\psi}_r) - K_{1\psi}|\sigma_\psi|^{1/2}\text{sign}(\sigma_\psi) - \int_0^t K_{2\psi}\text{sign}(\sigma_\psi)dT\Big) - (I_{xx} - I_{yy})\dot{\phi}\dot{\theta} \tag{3.58}$$

with $\sigma_\psi$ and $\dot{\sigma}_\psi$ converging towards 0 in a finite time if $K_{1\psi}$ and $K_{2\psi}$ are correctly tuned.

## 3.6 Quadrotor position control

The objective of this section is to detail the design of control for $x$ and $y$ position of the quadrotor, given that $z$ control has been made in Subsection 3.5.7. The $x$ and $y$ positions cannot be directly controlled by the four control inputs $U_1$, $U_2$, $U_3$ and $U_4$. However, the back-and-forth

and side-to-side movements of the quadrotor can be controlled through roll and pitch angles. The main idea is that the position control provides the desired roll ($\phi_r$) and pitch ($\theta_r$) trajectories for the attitude control (see previous section). Consider the Equations (2.41) and (2.42) discussing for $x$ and $y$ dynamics and rewrite the both under this following form.

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix} = \frac{U_1}{m} \begin{bmatrix} S\psi & C\psi \\ -C\psi & S\psi \end{bmatrix} \begin{bmatrix} S\phi \\ S\theta C\phi \end{bmatrix} \tag{3.59}$$

that gives

$$\begin{bmatrix} S\phi \\ S\theta C\phi \end{bmatrix} = \frac{m}{U_1} \begin{bmatrix} S\psi & -C\psi \\ C\psi & S\psi \end{bmatrix} \begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix} \tag{3.60}$$

Notice that the matrix $\begin{bmatrix} S\psi & -C\psi \\ C\psi & S\psi \end{bmatrix}$ is invertible $\forall\psi$. Since the objective of position control is to stabilize the quadrotor at an equilibrium point corresponding to the nominal hover state, it is supposed that angles $\phi$ and $\theta$ are small, and $U_1$ is used to just compensated the gravity *i.e.* $U_1 = mg$. By this way, Equation (3.60) can be written as

$$\begin{bmatrix} \phi \\ \theta \end{bmatrix} = \frac{1}{g} \begin{bmatrix} S\psi & -C\psi \\ C\psi & S\psi \end{bmatrix} \begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix} \tag{3.61}$$

Therefore, the desired roll ($\phi_r$) and pitch ($\theta_r$) for the attitude control are given by

$$\begin{bmatrix} \phi_r \\ \theta_r \end{bmatrix} = \frac{1}{g} \begin{bmatrix} S\psi & -C\psi \\ C\psi & S\psi \end{bmatrix} \begin{bmatrix} \nu_x \\ \nu_y \end{bmatrix} \tag{3.62}$$

where $\nu_x = \ddot{x}$ and $\nu_y = \ddot{y}$ are the control input signal for position $x$ and $y$ respectively.

## 3.7 Manipulator computed torque control

When the task requires high dynamic accuracy, it is necessary to take into account manipulator inverse dynamic model to cancel the nonlinearities terms and decouple the dynamic of each link via a computed torque control technique [35, 43, 56]. The control of manipulator can be handled by partitioning the controller (Figure 3.4) into

— a feedback linearization that linearizes and decouples manipulator dynamics using inverse dynamics model,

— and a feedback control which ensures tracking of desired trajectories.

The inverse dynamic model provides the vector of joint actuator torques, $\boldsymbol{\Gamma}$ [35, 43, 56]

$$\boldsymbol{\Gamma} = \boldsymbol{M}(\boldsymbol{q})\boldsymbol{\nu}_q + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\boldsymbol{q} + \boldsymbol{Q}(\boldsymbol{q}) \tag{3.63}$$

with $\boldsymbol{\nu}_q$ the "new" input vector that has to be defined and $\boldsymbol{q} = [q_1\ q_2]^T$ is vector of joints position. Comparing this control law with the dynamic model of the manipulator (2.50) correlates the control signal $\boldsymbol{\nu}_q$ to the joint acceleration, $\ddot{\boldsymbol{q}}$, that gives

$$\boldsymbol{\nu}_q = \ddot{\boldsymbol{q}} \tag{3.64}$$

Considering SMC (3.13) for the control law $\boldsymbol{\nu}_q$ gives

$$\boldsymbol{\nu}_q = \ddot{\boldsymbol{q}}_r - \boldsymbol{\lambda}_q(\dot{\boldsymbol{q}} - \dot{\boldsymbol{q}}_r) - \boldsymbol{K}_q\,\mathrm{sg}(\boldsymbol{\sigma}_q) \tag{3.65}$$

where $\boldsymbol{\lambda}_q = \begin{bmatrix}\lambda_{q_1} & 0\\ 0 & \lambda_{q_2}\end{bmatrix}$, $\boldsymbol{K}_q = \begin{bmatrix}K_{q_1} & 0\\ 0 & K_{q_2}\end{bmatrix}$ and $\mathrm{sg}(\boldsymbol{\sigma}_q) = \begin{bmatrix}\mathrm{sign}(\sigma_{q_1})\\ \mathrm{sign}(\sigma_{q_2})\end{bmatrix}$. Equation (3.64) - (3.65) give, with the sliding variable $\boldsymbol{\sigma}_q$ defined as

$$\begin{aligned}\boldsymbol{\sigma}_q &= \dot{\boldsymbol{e}}_q + \boldsymbol{\lambda}_q \boldsymbol{e}_q\\ \boldsymbol{e}_q &= \boldsymbol{q} - \boldsymbol{q}_r\\ \dot{\boldsymbol{\sigma}}_q &= -\boldsymbol{K}_q \mathrm{sg}(\boldsymbol{\sigma}_q)\end{aligned} \tag{3.66}$$

To summarize the overall manipulator computed torque control reads as

$$\boldsymbol{\Gamma} = \boldsymbol{M}(\boldsymbol{q})\Big(\ddot{\boldsymbol{q}}_r - \boldsymbol{\lambda}_q(\dot{\boldsymbol{q}} - \dot{\boldsymbol{q}}_r) - \boldsymbol{K}_q\,\mathrm{sg}(\boldsymbol{\sigma}_q)\Big) + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\boldsymbol{q} + \boldsymbol{Q}(\boldsymbol{q}) \tag{3.67}$$

with $\boldsymbol{\Gamma} = [\Gamma_1\ \Gamma_2]^T$ is input vector of joints torque. From Equation (3.63) with STWC (3.25) gives the following control law

$$\boldsymbol{\Gamma} = \boldsymbol{M}(\boldsymbol{q})\Big(\ddot{\boldsymbol{q}}_r - \boldsymbol{\lambda}_q(\dot{\boldsymbol{q}} - \dot{\boldsymbol{q}}_r) - \boldsymbol{K}_{1q}\mathrm{st}(\boldsymbol{\sigma}_q)\mathrm{sg}(\boldsymbol{\sigma}_q) - \int_0^t \boldsymbol{K}_{2q}\mathrm{sg}(\boldsymbol{\sigma}_q)dT\Big) + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\boldsymbol{q} + \boldsymbol{Q}(\boldsymbol{q}) \tag{3.68}$$

with $\boldsymbol{K}_{1q} = \begin{bmatrix}K_{1q_1} & 0\\ 0 & K_{1q_2}\end{bmatrix}$, $\boldsymbol{K}_{2q} = \begin{bmatrix}K_{2q_1} & 0\\ 0 & K_{2q_2}\end{bmatrix}$ and $\mathrm{st}(\boldsymbol{\sigma}_q) = \begin{bmatrix}\sqrt{|\sigma_{q_1}|} & 0\\ 0 & \sqrt{|\sigma_{q_2}|}\end{bmatrix}$ ensuring the establishment of a second order sliding mode.

## 3.8 Proportional integral derivative control

The Proportional-Integral-Derivative control is the most common form of linear feedback control, and can be found in various control applications due to its simplicity and good performances. It is widely used in control engineering. With advancements in technology, the PID controller can be enhanced with additional features such as automatic tuning, gain scheduling, and continuous adaptation. Even without these additional features, the controller can still be embedded in specialized control systems like the aerial manipulator as considered here. In this

work, PID controllers are used to determine the appropriate signals that stabilize the quadrotor in a desired position $x$ and $y$ for the given task.

In general form, PID feedback control takes the difference between reference value ($\xi_r$) and measured value ($\xi$) as its input to get the adequate acceleration ($\ddot{\xi}$) [42]. For example, if one has $\ddot{\xi} = \nu$ then defining

$$\nu = \ddot{\xi}_r + K_p(\xi_r - \xi) + K_i \int_0^t (\xi_r - \xi)dT + K_d(\dot{\xi}_r - \dot{\xi}) \tag{3.69}$$

where $K_p$ is the proportional coefficient, $K_i$ is the integral coefficient and $K_d$ is the derivative coefficient makes $e = \xi_r - \xi$ converging towards 0. For quadrotor operating near nominal hover state, gives the derivative $\ddot{\xi}_r = \dot{\xi}_r = 0$ [42, 41], as consider in Subsection 3.5.6. Then the PID control law for $x$ and $y$ positions control input signal of Equation (3.62) is computed as follows.

$$\nu_x = K_{p.x}\,(x_r - x) + K_{i.x} \int_0^t (x_r - x)dT - K_{d.x}\,\dot{x} \tag{3.70}$$

$$\nu_y = K_{p.y}\,(y_r - y) + K_{i.y} \int_0^t (y_r - y)dT - K_{d.y}\,\dot{y} \tag{3.71}$$

with $x_r$ and $y_r$ being the reference trajectory.

### 3.8.1 Quadrotor PID altitude control

As previously, Equation (3.33) is linearized at an equilibrium operating point with small angle approximation as describe in Subsection 3.5.6; then, one gets

$$\ddot{z} = \frac{U_1}{m} - g \tag{3.72}$$

Altitude controller is developed to generate the lifting thrust force signal, $U_1$, which determines the altitude position, $z$ of the quadrotor. The PID control law is then derived from linear Equation (3.72) and reads as

$$U_1 = m\Big(\ddot{z}_r + K_{p.z}\,(z_r - z) + K_{i.z} \int_0^t (z_r - z)dT + K_{d.z}\,(\dot{z}_r - \dot{z}) + g\Big) \tag{3.73}$$

with $z_r$ being the reference trajectory.

### 3.8.2 Quadrotor PID attitude and heading control

As PID is a linear control, the dynamic model Equation (3.34 - 3.36) for roll, pitch and yaw is linearized by supposing small variation of the angles. As in the previous section, the reference

trajectories for the attitude and heading read as

$$\ddot{\phi} = \frac{U_2}{I_{xx}} \tag{3.74}$$

$$\ddot{\theta} = \frac{U_3}{I_{yy}} \tag{3.75}$$

$$\ddot{\psi} = \frac{U_4}{I_{zz}} \tag{3.76}$$

**Remark.** *So, the dynamic equations describing the attitude behavior read as that the system is supposed to be decoupled.*

#### Roll Control

Roll controller generates the rolling torque signal, $U_2$, which is acting on the roll angle, $\phi$ of the quadrotor. The control law is derived from linear Equation (3.74) as follows.

$$U_2 = I_{xx}\Big(\ddot{\phi}_r + K_{p.\phi}\left(\phi_r - \phi\right) + K_{i.\phi}\int_0^t (\phi_r - \phi)dT + K_{d.\phi}\left(\dot{\phi}_r - \dot{\phi}\right)\Big) \tag{3.77}$$

that gives the following dynamics for roll tracking error $e_\phi = \phi_r - \phi$

#### Pitch Control

Pitch controller generates the pitching torque signal, $U_3$, which is acting on the pitch angle, $\theta$ of the quadrotor. The control law is derived from linear Equation (3.75) as follows.

$$U_3 = I_{yy}\Big(\ddot{\theta}_r + K_{p.\theta}\left(\theta_r - \theta\right) + K_{i.\theta}\int (\theta_r - \theta) + K_{d.\theta}\left(\dot{\theta}_r - \dot{\theta}\right)\Big) \tag{3.78}$$

that gives the following dynamics for pitch tracking error $e_\theta = \theta_r - \theta$

#### Yaw Control

Yaw controller generates the yawing torque signal, $U_4$, which is acting the heading angle, $\psi$ of the quadrotor. The control law is derived from linear Equation (3.76) as follows.

$$U_4 = I_{zz}\Big(\ddot{\psi}_r + K_{p.\psi}\left(\psi_r - \psi\right) + K_{i.\psi}\int (\psi_r - \psi) + K_{d.\psi}\left(\dot{\psi}_r - \dot{\psi}\right)\Big) \tag{3.79}$$

that gives the following dynamics for yaw tracking error $e_\psi = \psi_d - \psi$

### 3.8.3 PID steady state analysis

One of the purpose of control system is to provide a stable response *i.e* to maintain the system at a desired set point with minimum error even in the presence of external disturbance.

This section focuses on the steady state analysis of a PID controller to demonstrate how it maintains the desired setpoint by driving the tracking error, $e$, to zero when the system reaches a stable and unchanging condition. Consider a closed loop system error equation given by

$$\ddot{e} + k_d\,\dot{e} + k_p\,e = 0 \tag{3.80}$$

where $k_p$ is the proportional coefficient, $k_d$ is the derivative coefficient and $e$ the tracking error. Analyzing the system at constant state *i.e* a situation where various system parameters and variables no longer change with time. Then the derivatives of all system variables are zero, leads to

$$k_p\,e = 0 \tag{3.81}$$

or

$$e = 0. \tag{3.82}$$

Now, by introducing an external disturbance, $f_{dist}$ acting on the closed loop system, causes the error equation to become

$$\ddot{e} + k_d\,\dot{e} + k_p\,e = f_{dist} \tag{3.83}$$

That yields a steady-state equation

$$k_p\,e = f_{dist} \tag{3.84}$$

or

$$e = \frac{f_{dist}}{k_p} \tag{3.85}$$

at which the error value $e$ represents a steady-state error (if $f_{dist}$ is a constant). In this case, the control appears to be not efficient.

In order to eliminate this steady-state error, an integral term is introduced in the controller which becomes a PID. It gives

$$\ddot{e} + k_d\,\dot{e} + k_p\,e + k_i \int_0^t e\,dT = f_{dist} \tag{3.86}$$

where $k_i$ is the integral coefficient. Taking the time derivation of Equation (3.86) in the presence of constant external disturbance for $t > 0$ as

$$\dddot{e} + k_d\,\ddot{e} + k_p\,\dot{e} + k_i\,e = \dot{f}_{dist} \tag{3.87}$$

This yields a steady-state equation

$$k_i\,e = 0 \tag{3.88}$$

so that the tracking error becomes

$$e = 0 \tag{3.89}$$

This means that with introduction of an integral term the tracking error, $e$, converges to zero when the system reaches a stable and unchanging condition. As a result, the desired setpoint is achieved.

### 3.8.4 Manipulator computed torque with PID feedback control

As previously claimed, the control of the manipulator is handled by inverse dynamic model providing the vector of joint actuator control torque, $\boldsymbol{\Gamma}$ [35, 43, 56]

$$\boldsymbol{\Gamma} = \boldsymbol{M}(\boldsymbol{q})\boldsymbol{\nu}_q + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\boldsymbol{q} + \boldsymbol{Q}(\boldsymbol{q}) \tag{3.90}$$

Comparing this control law with the dynamic model of the manipulator as in Equation (2.50) correlate the "virtual" control input $\boldsymbol{\nu}_q$ to the joint acceleration, $\ddot{\boldsymbol{q}}$, as

$$\boldsymbol{\nu}_q = \ddot{\boldsymbol{q}} \tag{3.91}$$

Now, the manipulator control problem is reduced to that of finding a control law for $\boldsymbol{\nu}_q$ stabilizing the previous double integrator. Here, a PID controller is consider as

$$\boldsymbol{\nu}_q = \ddot{\boldsymbol{q}}_r + \boldsymbol{K}_{p.q}(\boldsymbol{q}_r - \boldsymbol{q}) + \boldsymbol{K}_{i.q}\int_0^t (\boldsymbol{q}_r - \boldsymbol{q})dT + \boldsymbol{K}_{d.q}(\dot{\boldsymbol{q}}_r - \dot{\boldsymbol{q}}) \tag{3.92}$$

with $\boldsymbol{K}_{p.q}$, $\boldsymbol{K}_{i.q}$ and $\boldsymbol{K}_{d.q}$ are the diagonal matrix of proportional coefficient, integral coefficient and derivative coefficient respectively. In which the overall manipulator computed torque control becomes

$$\boldsymbol{\Gamma} = \boldsymbol{M}(\boldsymbol{q})\Big(\ddot{\boldsymbol{q}}_r + \boldsymbol{K}_{p.q}(\boldsymbol{q}_r - \boldsymbol{q}) + \boldsymbol{K}_{i.q}\int_0^t (\boldsymbol{q}_r - \boldsymbol{q})dT + \boldsymbol{K}_{d.q}(\dot{\boldsymbol{q}}_r - \dot{\boldsymbol{q}})\Big) + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\boldsymbol{q} + \boldsymbol{Q}(\boldsymbol{q}) \tag{3.93}$$

Chapter 4

# Simulation

The purpose of this chapter is to evaluate the performances of two proposed control schemes: a first-order sliding mode control with adaptive gain (SMCA) [49] and a super-twisting sliding mode control with adaptive gain (STWCA) [52]. These control law have been chosen for their robustness and for their simplicity to apply. Furthermore, they allow to reduce the effect of perturbation and modeling uncertainties. A comparative analysis is conducted between these two schemes and two other control strategies: standard sliding mode control (SMC) and linear proportional-integral-differential control (PID). The aim is to assess the effectiveness of each control scheme in achieving the desired system behavior. To validate the control schemes, a simulation environment using Matlab/Simulink is used. This platform enables accurate modeling and analysis of the system dynamics and control algorithms. By simulating the system in a controlled environment, various performance metrics can be evaluated, such as tracking accuracy, disturbance rejection and control effort. The simulation employs a helical motion trajectory to mimic real-world scenarios, allowing the assessment of the control schemes under situations both with and without the presence of disturbances. This comprehensive evaluation provides insights into the robustness and effectiveness of the control strategies in handling disturbances and maintaining accurate trajectory tracking. By comparing the performances of SMCA, STWCA, SMC, and PID under the helical motion trajectory simulation, valuable insights can be gained regarding the advantages and limitations of each control scheme. This analysis aids in selecting the most suitable control approach for the specific requirements of the system under consideration.

## 4.1 Model parameters

The control scheme's validation has been performed through simulation of the closed-loop system, considering some specific parameters of the quadrotor. In this setup, the quadrotor is equipped with a 2-DOF revolute manipulator arm as in Figure 4.1, and the following parameters are taken into account:

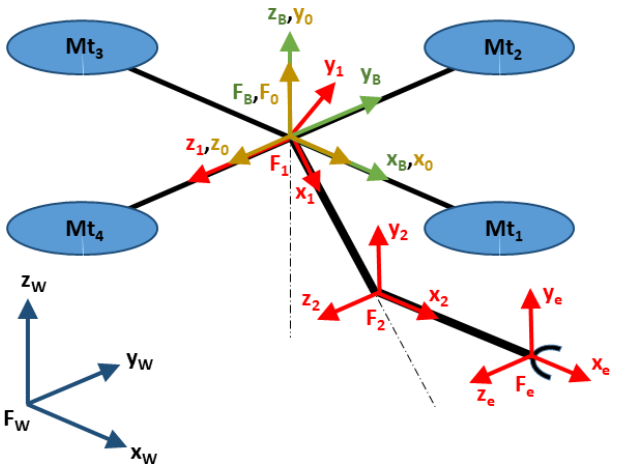



Figure 4.1 – Quadrotor UAV with a 2-DOF manipulator arm and the associated frame.

| Parameter | Value |
|---|---|
| Inertia, $I_{xx}$ | $1.24\,\mathrm{kg\,m^2}$ |
| Inertia, $I_{yy}$ | $1.24\,\mathrm{kg\,m^2}$ |
| Inertia, $I_{zz}$ | $2.48\,\mathrm{kg\,m^2}$ |
| Mass, $m$ | $2.00\,\mathrm{kg}$ |
| Rotor Distance, $d$ | $0.25\,\mathrm{m}$ |
| Thrust Coefficient, $C_f$ | $3 \times 10^{-5}\,\mathrm{N\,s^2}$ |
| Torque Coefficient, $C_m$ | $7.5 \times 10^{-7}\,\mathrm{N\,m\,s^2}$ |
| Gravity, $g$ | $9.81\,\mathrm{m\,s^{-2}}$ |

Table 4.1 – Quadrotor parameters.

| Parameter | Link 1 | Link 2 |
|---|---|---|
| Link length, $L_*$ | $0.25\,\mathrm{m}$ | $0.25\,\mathrm{m}$ |
| Link mass, $m_*$ | $0.1\,\mathrm{kg}$ | $0.1\,\mathrm{kg}$ |
| Inertia $I_*$ | $0.0021\,\mathrm{kg\,m^2}$ | $0.0021\,\mathrm{kg\,m^2}$ |

Table 4.2 – Manipulator arm parameters.

These parameters play a crucial role in characterizing the dynamics of the aerial manipulator. Accurately defining these values allows the simulation to faithfully replicate the system's behavior and effectively evaluate the performance of the control scheme. In addition to the specified parameters, the simulations also take into account external disturbances. The disturbances are a constant and random force acting on quadrotor and a point mass loading at the end of the manipulator arm. By incorporating these elements into the simulation environment,

a more realistic representation of the aerial manipulator's operation is achieved. This enables a comprehensive evaluation of the control scheme's effectiveness in handling disturbances and maintaining stable and precise control.

By validating the control scheme using the specified parameters and conducting simulations in a realistic environment, knowledge of the system can be gained regarding the behavior and performance of the aerial manipulator. These findings provide a deeper understanding of the system's capabilities and limitations, facilitating further improvements to the control strategy. The control scheme can then be applied to enhance the performance of the aerial manipulator in various applications and scenarios, ensuring efficient and reliable operation.

## 4.2 Helical trajectory

The system's tracking capability is evaluated by employing a helical motion trajectory. This trajectory represents a three-dimensional path that combines both translational and rotational motion. Testing the system on such trajectories evaluates its ability to handle complex movements. Furthermore, these trajectories are frequently encountered in real-world applications such as motion required to inspecting cylindrical structures like circular columns and nuclear cooling towers. A comprehensive test is conducted to ensure that the end of the manipulator arm accurately tracks the desired helical path, as depicted in Figure 4.2. The translational and rotational motion required to generate the helical path are illustrated in Figure 4.3 and Figure 4.4, respectively.

To achieve the desired helical motion trajectory, the positions $x$ and $y$ oscillate periodically in a horizontal motion of the inertial frame $F_W$. Simultaneously, the position $z$, representing altitude, steadily increases at a constant rate. This coordinated motion creates the intricate helical pattern. During the tracking of this helical trajectory, an additional level of complexity is introduced. In addition to following the helical path, the end of the manipulator arm periodically pitches at a 20° angle and completes five full cycles of yawing. These additional motions simulate real-world scenarios where the manipulator arm needs to adjust its orientation while tracking a trajectory, enhancing the realism of the test.

This comprehensive test can be performed under various conditions. It can be conducted both with and without the presence of external disturbances also with and without the presence of point mass loading. By evaluating the system tracking capability and response in the presence or absence of disturbances, the robustness and effectiveness of the system can be assessed. This enables a thorough understanding of the system performance and aids in further improvements of the control strategy.

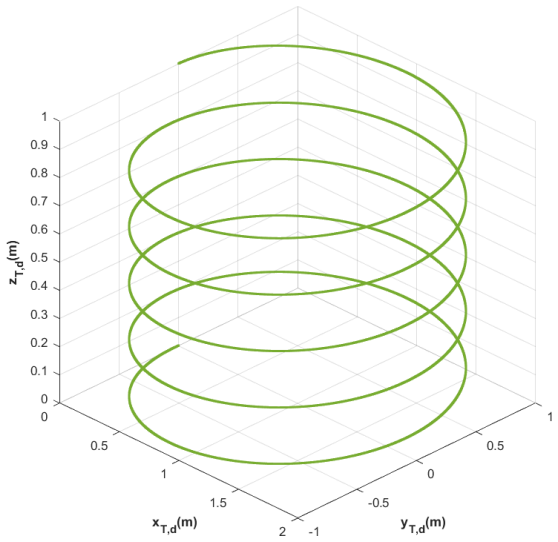
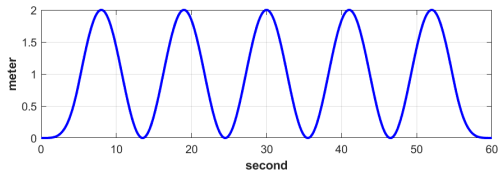
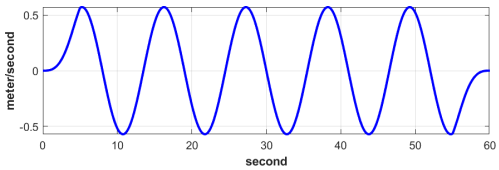


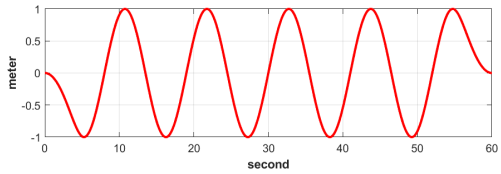

Figure 4.2 – Helical motion trajectory of end of the manipulator arm.



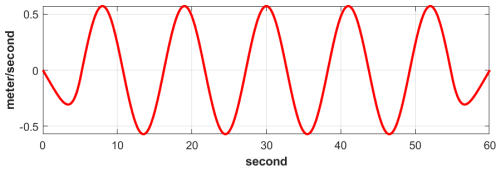

(a) Position $x_{T.d}$.



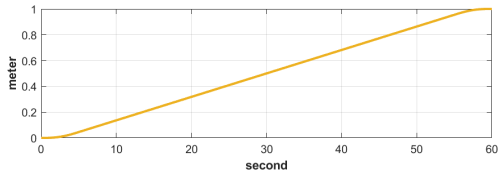

(b) Velocity $x_{T.d}$.



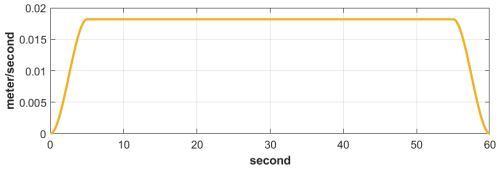

(c) Position $y_{T.d}$.




(d) Velocity $y_{T.d}$.

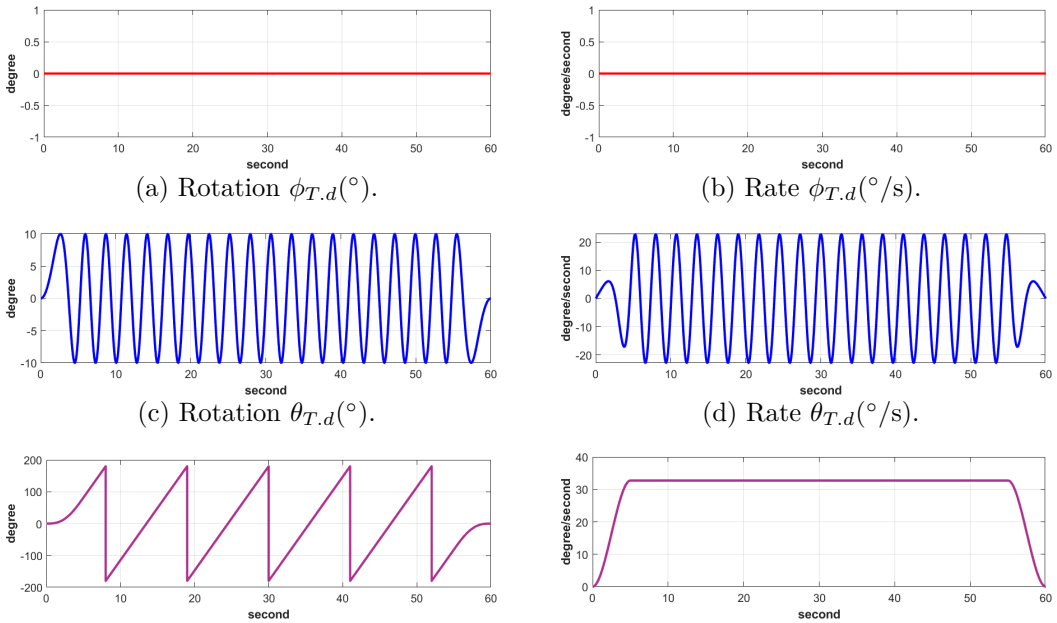



(e) Position $z_{T.d}$.




(f) Velocity $z_{T.d}$.

Figure 4.3 – Desired end of the manipulator arm helical motion translational trajectory (m and m/s) versus time (second).




(a) Rotation $\phi_{T.d}(^\circ)$.
(b) Rate $\phi_{T.d}(^\circ/\text{s})$.
(c) Rotation $\theta_{T.d}(^\circ)$.
(d) Rate $\theta_{T.d}(^\circ/\text{s})$.
(e) Rotation $\psi_{T.d}(^\circ)$.
(f) Rate $\psi_{T.d}(^\circ/\text{s})$.

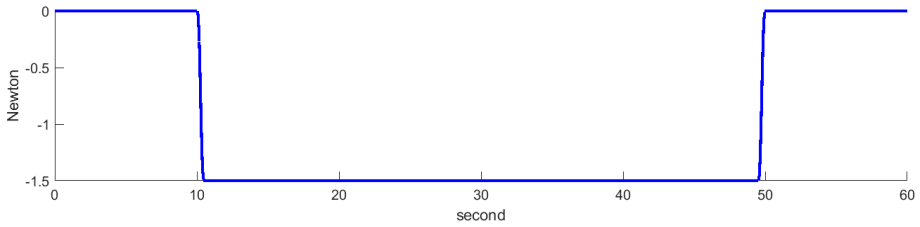
Figure 4.4 – Desired end of the manipulator arm helical motion rotational trajectory (° and °/s) versus time (second).

### 4.2.1 Disturbance

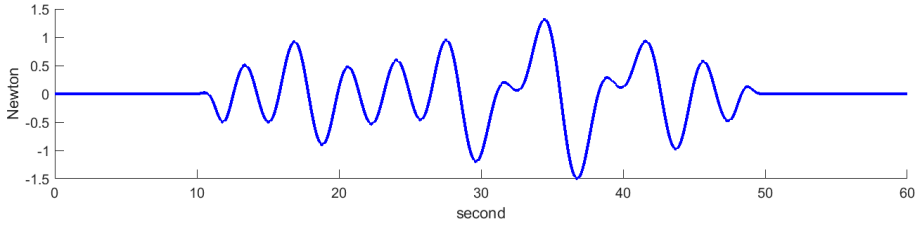
In order to test the system under external disturbances, both the quadrotor and manipulator arm are subjected to specific forces. To begin, a constant and random force is introduced to the quadrotor, as depicted in Figure 4.5. The force acting on the system as given in Equation (3.33) to (3.36). This horizontal force disturbance, with a maximum magnitude of 1.5N, is applied along the $x$-axis of the inertial frame $F_W$. The purpose of this disturbance is to simulate real-world scenarios where the UAV may encounter unpredictable forces during its operation.

Additionally, to assess the system's response to disturbances affecting the manipulator arm, a point mass with a weight of 1N is attached to the end of the arm. This weight is equivalent to a mass of 0.1 kg, which matches the mass of each manipulator link. This point mass represents the load or device that would typically be connected (end effector). By introducing this mass, the system ability to handle external disturbances while manipulating objects can be evaluated.

To comprehensively evaluate the system performance under various scenarios, a combination of horizontal force disturbances on the quadrotor and the presence of the point mass on the manipulator arm are introduced simultaneously. This mixed disturbance scenario provides a more realistic testing environment, enabling a thorough assessment of the system's robustness and ability to maintain tracking accuracy despite external perturbations.




(a) Constant disturbance (N) versus time (second).




(b) Random disturbance (N) versus time (second).

Figure 4.5 – Constant and random external disturbance.

## 4.3 Control test parameter

The parameters for the PID, SMC, SMCA, and STWCA controllers are presented in Tables 4.3, 4.4, 4.5 and 4.6, respectively. These tables provide detailed information on the parameters required for each control scheme, categorized into quadrotor outer position control, quadrotor inner attitude control, and manipulator joint control. Notably, the $x-$ and $y-$position controllers for all control schemes in the outer position control section are implemented as PID controllers. This choice is made to adequately compute the desired trajectories for roll and pitch, which serve as inputs for the inner attitude control. By employing PID controllers for position control, the quadrotor can sufficiently achieve the desired positions in the $x$ and $y$ directions.

The quadrotor inner attitude control parameters are crucial for maintaining stability and controlling the aerial vehicle orientation. These parameters govern the quadrotor roll, pitch, and yaw angles, ensuring precise control and maneuverability. Additionally, the inner attitude control loop focuses on tracking the reference altitude along the z-axis, enabling the quadrotor to maintain the desired height during operation. The manipulator arm joint control parameters play a significant role in achieving accurate control over each joint. These parameters dictate the manipulator movements, allowing it to perform specific tasks and reach the desired orientations necessary for effective manipulation and operation.

Table 4.3 – PID parameters

| Quadrotor PID position control | | | |
|---|---|---|---|
| Parameter | $K_p$ | $K_i$ | $K_d$ |
| Quad Pos X | 5.9 | 4.98 | 1.75 |
| Quad Pos Y | 5.9 | 4.98 | 1.75 |

| Quadrotor PID attitude control | | | |
|---|---|---|---|
| Parameter | $K_p$ | $K_i$ | $K_d$ |
| Quad Pos Z | 20 | 22.36 | 4.47 |
| Quad Roll | 145 | 218.3 | 24.08 |
| Quad Pitch | 155 | 344.6 | 17.43 |
| Quad Yaw | 24 | 24.5 | 5.88 |

| Manipulator PID joint control | | | |
|---|---|---|---|
| Parameter | $K_p$ | $K_i$ | $K_d$ |
| Joint Q1 | 430 | 2787 | 16.59 |
| Joint Q2 | 430 | 2787 | 16.59 |

Table 4.4 – SMC parameters

| Quadrotor PID position control | | | |
|---|---|---|---|
| Parameter | $K_p$ | $K_i$ | $K_d$ |
| Quad Pos X | 7.5 | 6.42 | 2.19 |
| Quad Pos Y | 7.5 | 6.42 | 2.19 |

| Quadrotor SMC attitude control | | | |
|---|---|---|---|
| Parameter | $\lambda$ | $K$ | $\epsilon$ |
| Quad Pos Z | 30 | 6.8 | 0.1 |
| Quad Roll | 60 | 5.7 | 0.1 |
| Quad Pitch | 60 | 6.8 | 0.1 |
| Quad Yaw | 30 | 3.5 | 0.2 |

| Manipulator SMC joint control | | | |
|---|---|---|---|
| Parameter | $\lambda$ | $K$ | $\epsilon$ |
| Joint Q1 | 100 | 130 | 0.15 |
| Joint Q2 | 300 | 330 | 0.3 |

Table 4.5 – SMCA parameters

| Quadrotor PID position control | | | |
|---|---|---|---|
| Parameter | $K_p$ | $K_i$ | $K_d$ |
| Quad Pos X | 7.25 | 6.26 | 2.10 |
| Quad Pos Y | 7.25 | 6.26 | 2.10 |

| Quadrotor SMCA attitude control | | | | | | |
|---|---|---|---|---|---|---|
| Parameter | $\lambda$ | $K_0$ | $\bar{K}$ | $\mu$ | $\alpha$ | $\epsilon$ |
| Quad Pos Z | 30 | 0.001 | 110 | 0.013 | 0.1 | 0.15 |
| Quad Roll | 60 | 0.001 | 150 | 0.014 | 0.5 | 0.7 |
| Quad Pitch | 60 | 0.001 | 180 | 0.01 | 10 | 0.8 |
| Quad Yaw | 30 | 0.001 | 100 | 0.01 | 0.3 | 0.1 |

| Manipulator SMCA joint control | | | | | | |
|---|---|---|---|---|---|---|
| Parameter | $\lambda$ | $K_0$ | $\bar{K}$ | $\mu$ | $\alpha$ | $\epsilon$ |
| Joint Q1 | 100 | 0.001 | 153 | 0.005 | 1 | 0.22 |
| Joint Q2 | 300 | 0.001 | 130 | 0.1 | 1 | 0.4 |

Table 4.6 – STWCA parameters

| Quadrotor PID position control | | | |
|---|---|---|---|
| Parameter | $K_p$ | $K_i$ | $K_d$ |
| Quad Pos X | 5.5 | 4.8 | 1.53 |
| Quad Pos Y | 5.5 | 4.8 | 1.53 |

| Quadrotor STWCA attitude control | | | | | |
|---|---|---|---|---|---|
| Parameter | $\lambda$ | $K_{10}$ | $K_{20}$ | $\epsilon$ | $\alpha$ |
| Quad Pos Z | 30 | 0.001 | 0.001 | 0.005 | 170 |
| Quad Roll | 60 | 0.001 | 0.001 | 0.005 | 60 |
| Quad Pitch | 60 | 0.001 | 0.001 | 0.005 | 60 |
| Quad Yaw | 30 | 0.001 | 0.001 | 0.005 | 75 |

| Manipulator STWCA joint control | | | | | |
|---|---|---|---|---|---|
| Parameter | $\lambda$ | $K_{10}$ | $K_{20}$ | $\epsilon$ | $\alpha$ |
| Joint Q1 | 100 | 0.001 | 0.001 | 0.01 | 515 |
| Joint Q2 | 300 | 0.001 | 0.001 | 0.01 | 900 |

## 4.4 Hovering test

Performing a hovering test is crucial when simulating an aerial manipulator using the proposed control scheme. Hovering involves maintaining a stable position at a fixed altitude, and this test enable the inspection of quadrotor stability. This test involves evaluating the system behavior while the quadrotor hovers with the manipulator arm in a static position, as shown in Figure 4.6. Hovering is achieved by controlling the thrust or lift force generated by the quadrotor along the z-axis. The trajectory required for conducting the hovering test is provided in Figure 4.7.

Hovering is a fundamental capability for an aerial manipulator. By conducting a hovering test, we can assess the stability of the control scheme and verify the system ability to maintain a stationary position in the air without drifting or experiencing significant oscillations. The test allows to determine if the proposed control scheme effectively stabilizes the system during static hovering. Furthermore, hovering serves as a baseline performance evaluation for the control scheme. It allows to observe and analyze the closed-loop system behavior and performances under perfectly known conditions, serving as a reference for subsequent tests and comparisons. The hovering test establishes a benchmark that can be used to measure the improvements achieved by the advanced proposed control scheme. From this hovering test, three key values

are extracted:

— the quadrotor position error, $\xi_e$, in inertial frame $F_W$ calculated as the Euclidean norm of the position $x$ error, position $y$ error and position $z$ error as;

$$\|\xi_e\| = \sqrt{x_e^2 + y_e^2 + z_e^2}$$

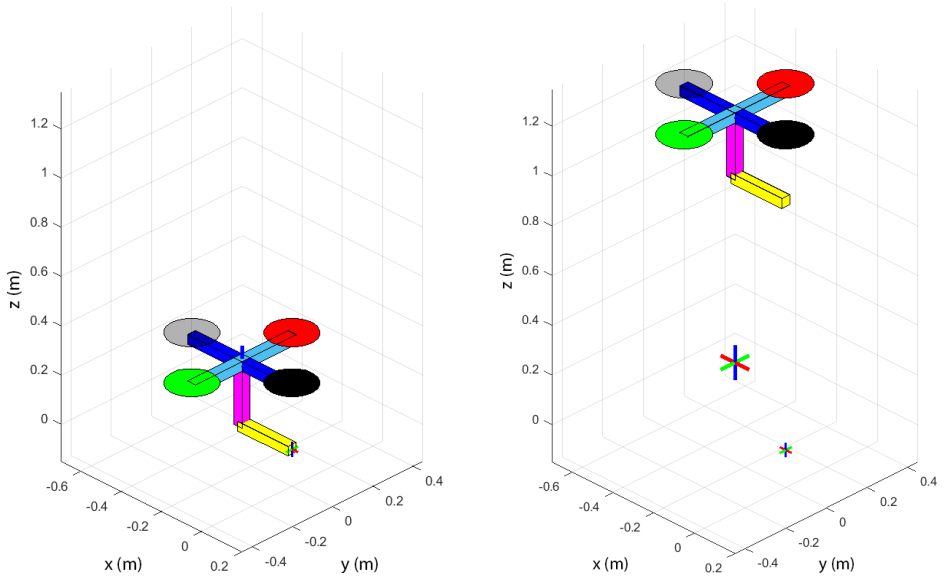
— the quadrotor rotation error, $\eta_e$, in body frame $F_B$ calculated as the absolute value of the error angle, represented as:

$$\|\eta_e\| = \left| arccos\left(\frac{tr(\widehat{{}^W\boldsymbol{R}_B}^T\, {}^W\boldsymbol{R}_B) - 1}{2}\right)\right|$$

where ${}^W\boldsymbol{R}_B$ and $\widehat{{}^W\boldsymbol{R}_B}$ refer to the desired and actual quadrotor rotation matrices, respectively;

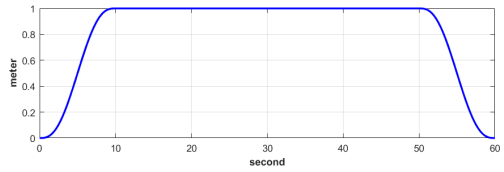
— the manipulator orientation error, $q_e$, in base frame $F_0$ calculated as Manhattan norm of joint $q_1$ error and joint $q_2$ error as:

$$\|q_e\| = |q_{1e}| + |q_{2e}|$$

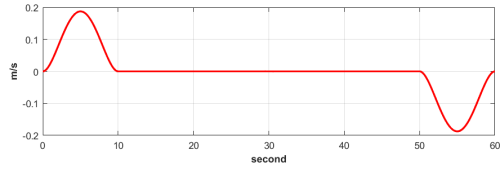



(a) Initial position. (b) Final position.

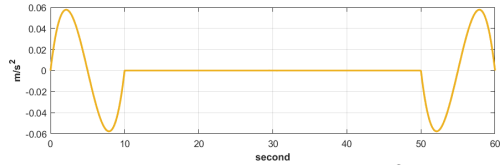
Figure 4.6 – The unmanned aerial manipulator hovers while transitioning from its initial position to the final fixed position.




(a) Position $z_{T.d}(m)$.




(b) Velocity $\dot{z}_{T.d}(m/s)$.




(c) Acceleration $\ddot{z}_{T.d}(m/s^2)$.

Figure 4.7 – Desired unmanned aerial manipulator hovering trajectory.

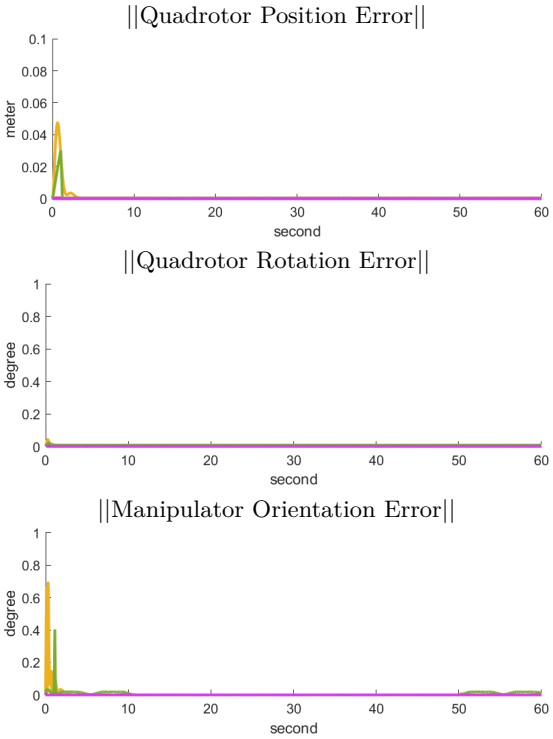



Figure 4.8 – Hovering tracking error.

The hover test, conducted without introducing external disturbances, demonstrates near-zero error for all control schemes, as illustrated in Figure 4.8. (Throughout this writing, the graphs of simulation results follow a consistent color scheme: PID represented in yellow, SMC (sliding mode control) in red, SMCA (adaptive SMC) in green, and STWCA (adaptive super-twisting) in purple.)

During the initial phase of the simulation, when the quadrotor ascends to the hovering height, there is a slight deviation in the position error for the PID and SMCA controllers. However, the control system quickly responds and brings the quadrotor back on track. On the other hand, SMC and STWCA show minimal deviation from the desired set point while the quadrotor is ascending. The same characteristic can be observed for the manipulator's orientation error, while the quadrotor's rotation error does not show any noticeable deviation. The steady-state errors are found to be less than $5 \times 10^{-4}$ meters, $8 \times 10^{-3}$ degrees and $3 \times 10^{-4}$ degrees for quadrotor position, quadrotor rotation, and manipulator orientation, respectively. The slight deviation in the error during the initial stage for the SMCA controller is a result of a gradual increase in adaptive gain, as depicted in Figure 4.9. Conversely, in the case of the STWCA controller, as illustrated in Figure 4.10, the adaptive gain rapidly increases and then gradually decreases to a minimal value.

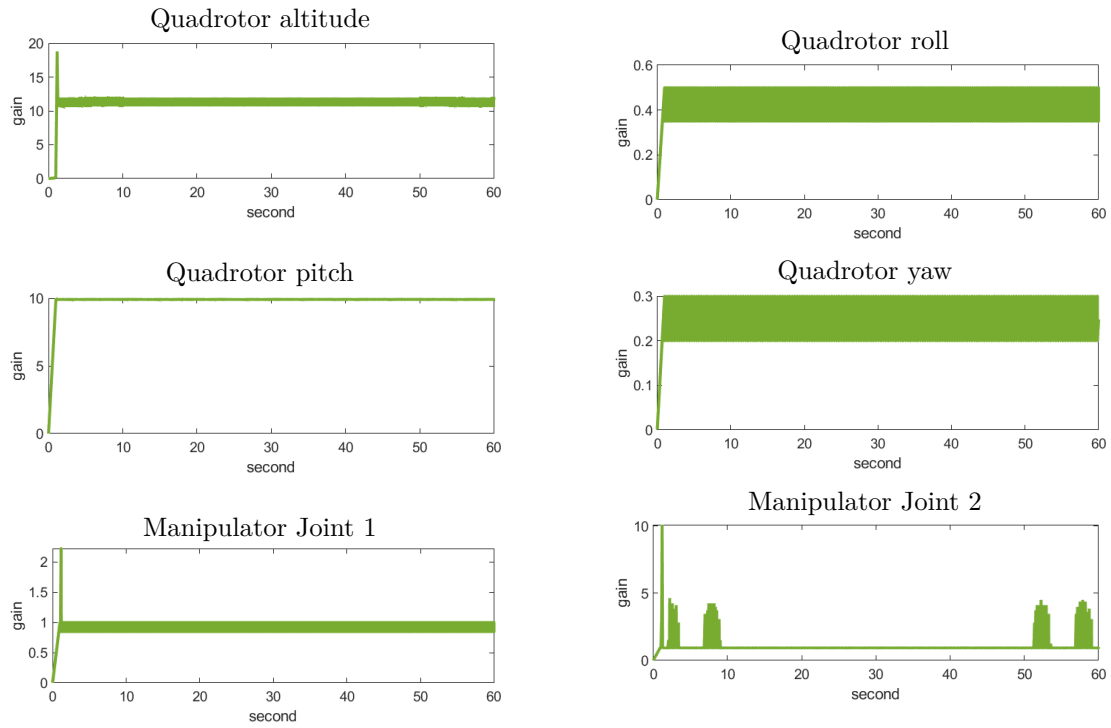



Figure 4.9 – Hovering trajectory tracking SMCA adaptive gain.

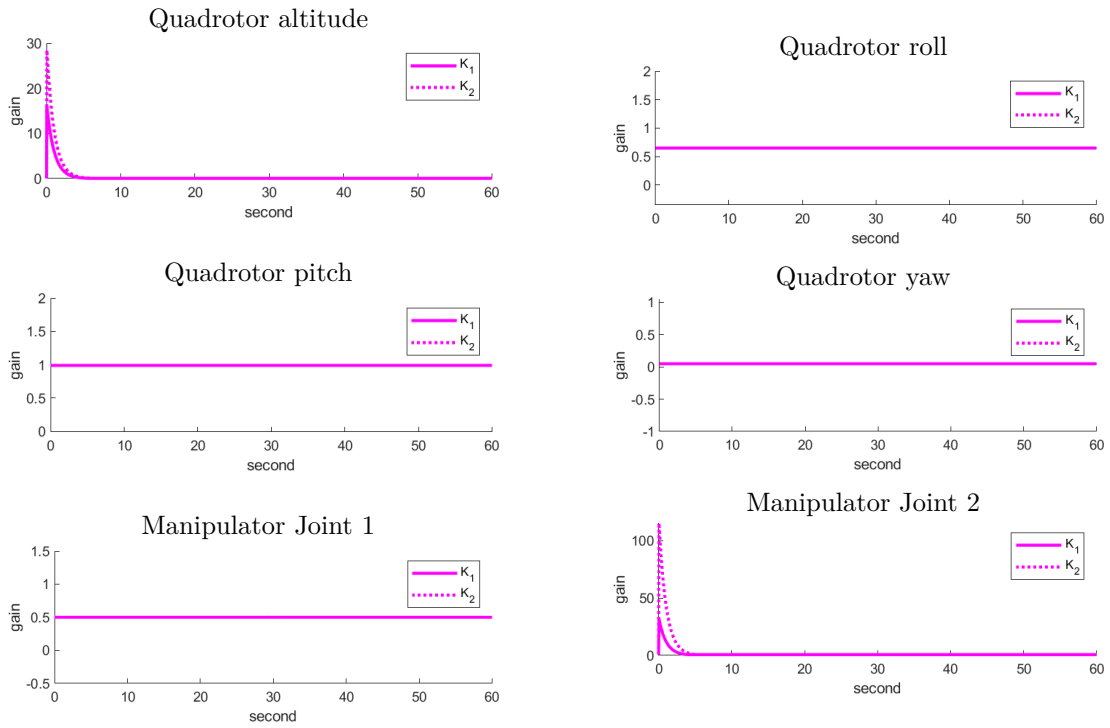



Figure 4.10 – Hovering trajectory tracking STWCA adaptive gain.

When a constant horizontal force is introduced to the quadrotor between 10s and 50s (Figure 4.11), all controllers exhibit momentary oscillations before settling to near-zero error, with STWCA showing the highest quadrotor position deviation. In the position controller, it consists of two PID controls for x- and y-positions, along with an adaptive super-twisting controller for the z-position. It is worth noting that the PID controls for x- and y-positions contributes to the higher position deviation. Conversely, the PID controller exhibits the highest deviation for quadrotor rotation error and manipulator orientation error. Nevertheless, it is evident that all controllers effectively stabilize the system despite the introduction of disturbance during static hovering. The steady-state error remains below $5 \times 10^{-4}$ meters, $8 \times 10^{-3}$ degrees and $25 \times 10^{-3}$ degrees for quadrotor position, quadrotor rotation, and manipulator orientation, respectively. Figure 4.13 and 4.14 illustrate how the introduction of a constant horizontal force affected the adaptive gain of both the SMCA and STWCA controllers, particularly in relation to the quadrotor pitch, manipulator joint 1, and manipulator joint 2.

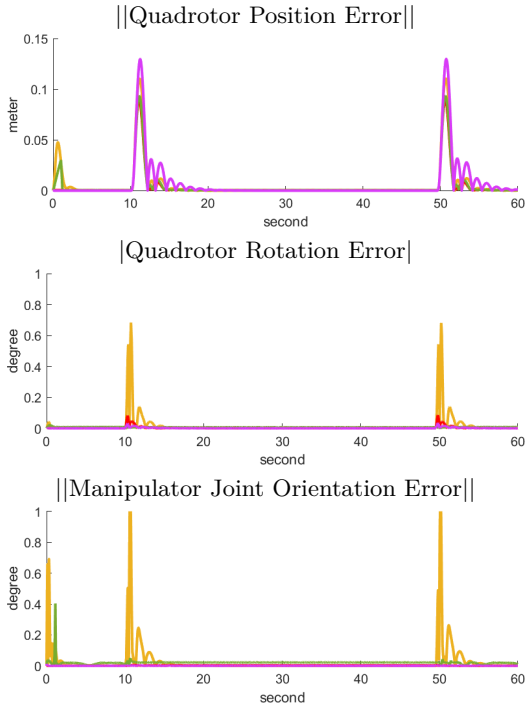



Figure 4.11 – Hovering tracking error with constant horizontal force disturbance.

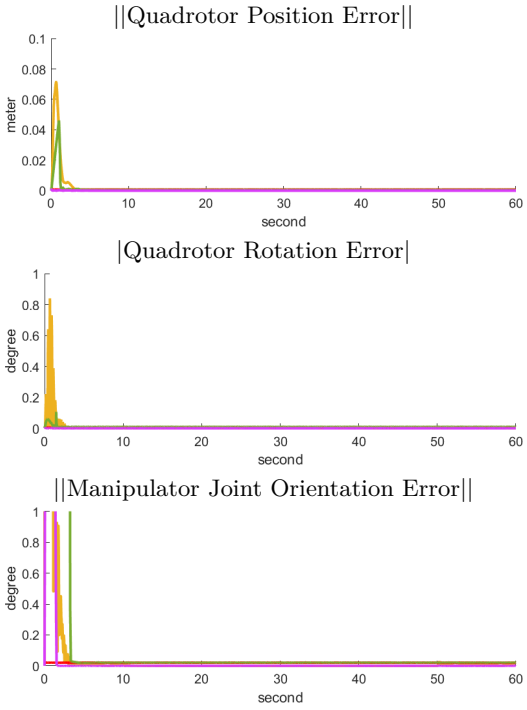



Figure 4.12 – Hovering tracking error with point mass loading.

Furthermore, when a point mass is introduced to the end of a manipulator arm (Figure 4.12), near-zero error is observed for all control schemes. The point mass adds uncertainties to the system, but all control schemes respond instantly to handle these uncertainties. The steady-state error is less than $8 \times 10^{-4}$ meters, $12 \times 10^{-3}$ degrees and $25 \times 10^{-3}$ degrees for quadrotor position, quadrotor rotation, and manipulator orientation, respectively. The introduction of the point mass loading affects the initial-stage adaptive gain of both the SMCA and STWCA controllers, as illustrated in Figure 4.15 and Figure 4.16. As previously mentioned, this effect is particularly noticeable in the quadrotor pitch, manipulator joint 1, and manipulator joint 2. Table 4.7 presents the root mean square error for these hovering test scenarios.

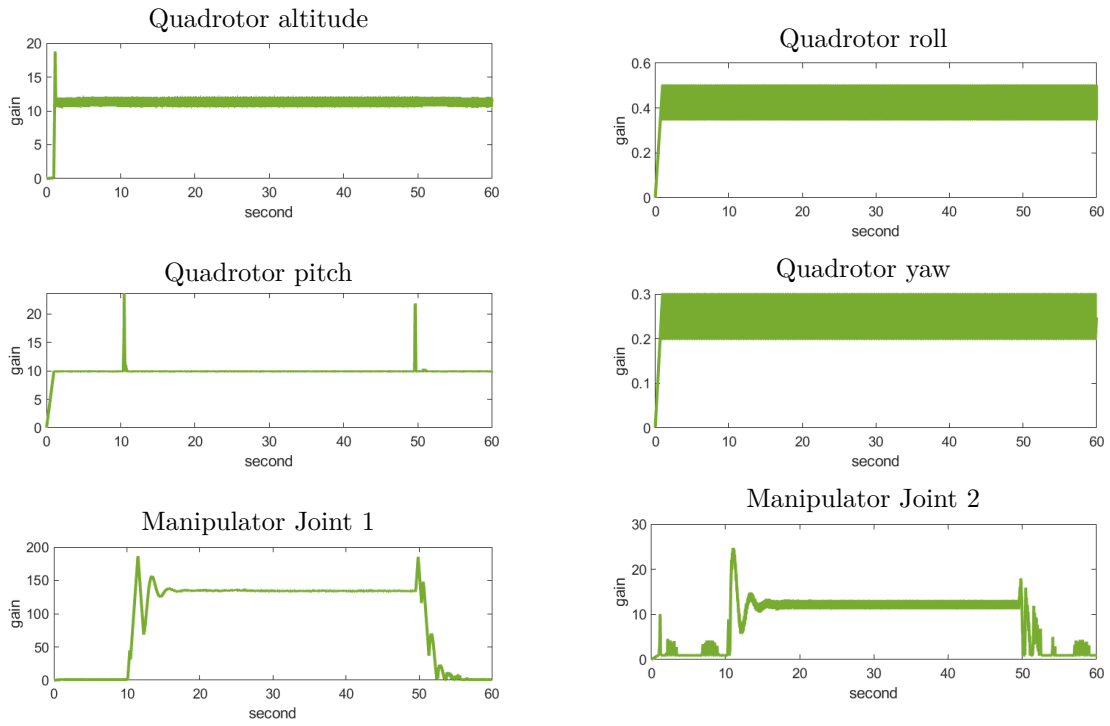



Figure 4.13 – Hovering trajectory tracking SMCA adaptive gain with constant horizontal force disturbance.

Table 4.7 – Hover tracking root mean square error.

| Without Disturbance. | | | | |
|---|---|---|---|---|
| Error | PID | SMC | SMCA | STWCA |
| Quadrotor Position | 2.133e-05 | 4.809e-04 | 4.333e-04 | 2.151e-07 |
| Quadrotor Rotation | 2.434e-05 | 1.389e-03 | 7.623e-03 | 8.601e-06 |
| Joint Orientation | 3.669e-04 | 1.394e-05 | 6.843e-04 | 1.704e-05 |
| **With Horizontal Force.** | | | | |
| Error | PID | SMC | SMCA | STWCA |
| Quadrotor Position | 1.676e-02 | 1.287e-02 | 1.350e-02 | 1.957e-02 |
| Quadrotor Rotation | 7.476e-02 | 1.005e-02 | 6.877e-03 | 4.051e-03 |
| Joint Orientation | 9.972e-02 | 2.597e-03 | 2.213e-02 | 5.646e-04 |
| **With Point Mass.** | | | | |
| Error | PID | SMC | SMCA | STWCA |
| Quadrotor Position | 3.183e-05 | 7.213e-04 | 4.332e-04 | 3.170e-07 |
| Quadrotor Rotation | 4.391e-05 | 4.166e-03 | 9.562e-03 | 1.089e-05 |
| Joint Orientation | 1.196e-03 | 2.046e-02 | 2.116e-02 | 2.437e-05 |

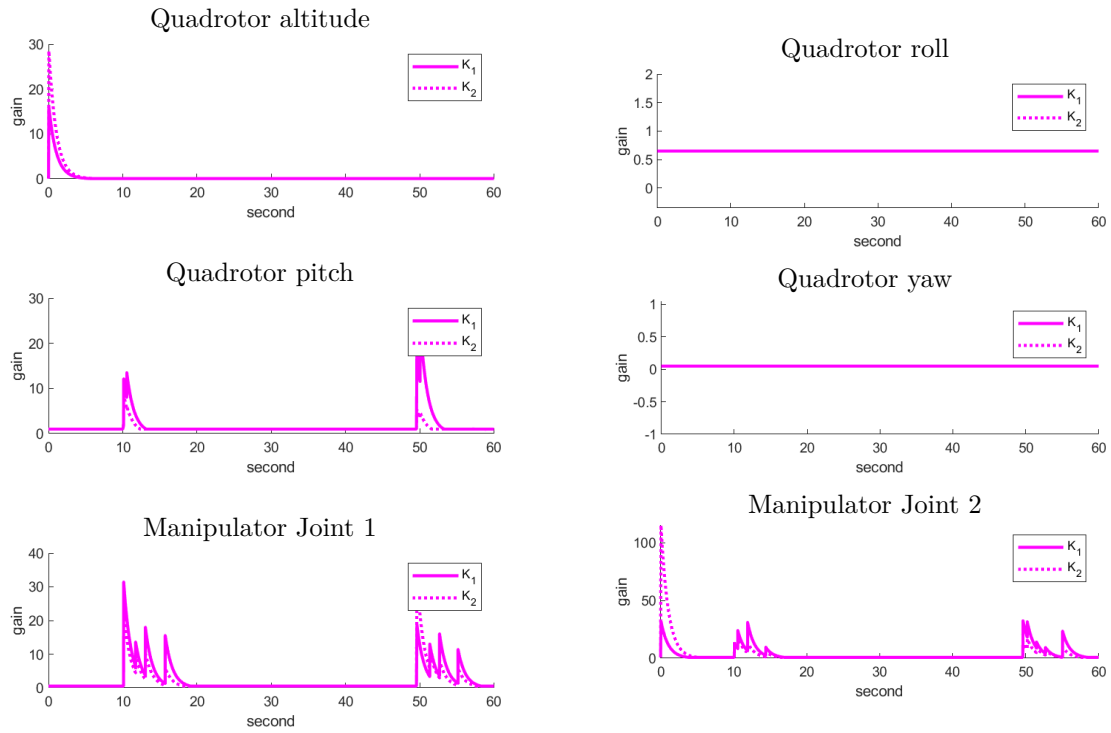



Figure 4.14 – Hovering trajectory tracking STWCA adaptive gain with constant horizontal force disturbance.

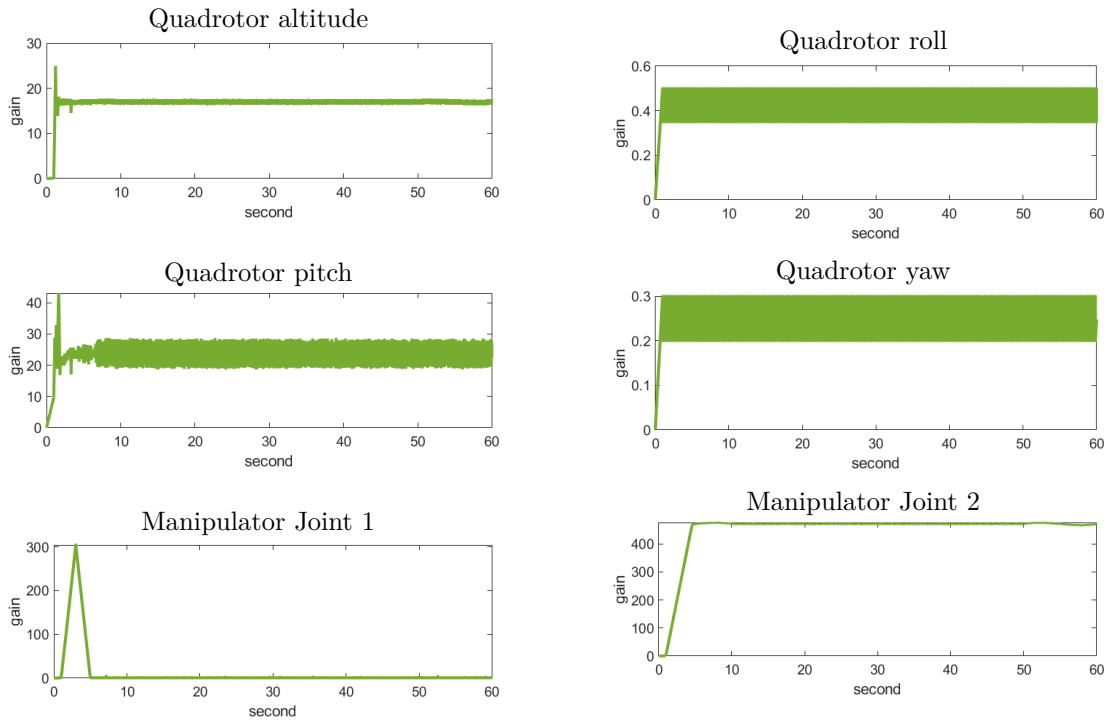



Figure 4.15 – Hovering trajectory tracking SMCA adaptive gain with point mass loading.

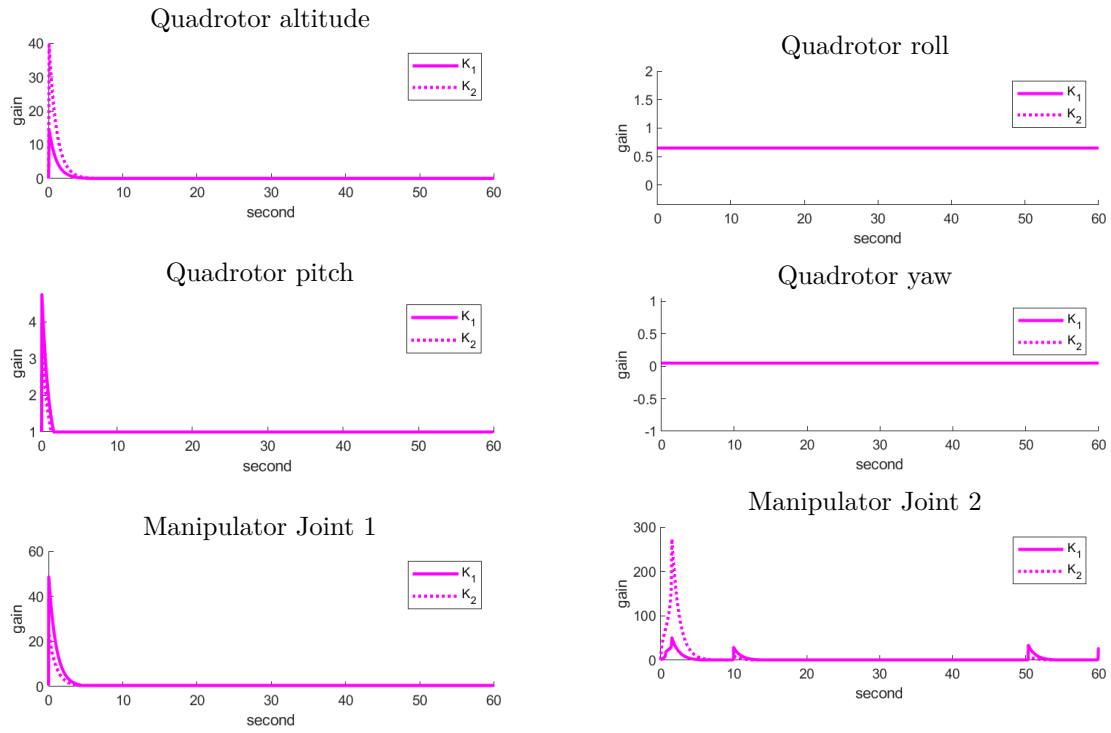



Figure 4.16 – Hovering trajectory tracking STWCA adaptive gain with point mass loading.

This test was conducted to evaluate the stability, control performance and precision of an aerial manipulator during static hovering. Under all three scenarios, the proposed control scheme reached a stable operating condition with minimal steady state errors. These errors were less than 10 mm for position error and less than 0.1 degree for rotational error as depicted in Figure 4.8, 4.11 and 4.12. The precision of the proposed control scheme is reflected by the small value of the RMSE.

## 4.5 Simulation - Helical trajectory

This section presents a simulation study focused on the tracking of a helical trajectory while simultaneously adjusting the manipulator arm orientation. This simulation aims to replicate real-world scenarios where aerial manipulators must adapt their orientation while following a specific trajectory. The study involves dynamically pitching the end of the manipulator arm at a 20° angle, mimicking practical situations where manipulator arms need to adjust their orientation during operation. To achieve precise and robust tracking performance, two proposed control schemes, namely SMCA and STWCA, are compared against conventional linear PID control and standard SMC. The effectiveness of these control strategies is evaluated through an analysis,

comprising quadrotor position error, quadrotor rotation error and manipulator orientation error for each control scheme. The simulation results further illustrate the variations in quadrotor total thrust, joint 1 torque and joint 2 torque across the different control approaches.

To provide quantitative assessment, a table showing the RMSE for helical trajectory tracking and the variations in input force-torque is presented, supporting the comparison and analysis of the proposed control schemes. These metrics offer valuable insights into the tracking accuracy and the effectiveness of actuator control in maintaining stability and orientation adjustment during the helical trajectory execution. The subsequent sections will delve into a detailed analysis of the results and explore the suitability of the proposed control schemes for aerial manipulator trajectory tracking in dynamic real-world scenarios.

### 4.5.1 Without disturbance

Figure 4.17 provides a comparison of aerial manipulator trajectory tracking errors and input force-torque among different controllers: PID (yellow), SMC (red), SMCA (red), and STWCA (purple). The simulations are conducted under nominal conditions, with no perturbation or uncertainty. In Figures 4.17a, 4.17b and 4.17c, the tracking errors of quadrotor position, quadrotor rotation and manipulator orientation are displayed. These errors oscillate as the controllers try to compensate for the helical motion trajectory, which involves both translational and rotational motion, leading to constant changes in the trajectory. This oscillation is also evident in the input force and torque shown in Figure 4.17d, 4.17e and 4.17f for the same reasons.

It is worth noting that the quadrotor position error exhibits an almost similar pattern for all control schemes since the $x$- and $y$-position controllers are implemented as PID controllers across all schemes. For quadrotor rotation, the proposed SMCA and STWCA control schemes demonstrate better tracking errors compared to the PID and SMC controller. Additionally, in the case of the manipulator orientation, only STWCA shows a smaller tracking error compared to the PID and SMC controllers. SMC outperforms SMCA due to its slightly higher constant error in manipulator control. Nevertheless, the analysis indicates that the proposed control scheme still yields improved trajectory tracking accuracy for quadrotor rotation and manipulator orientation.

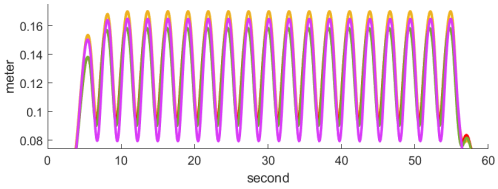
Table 4.8 displays the tracking RMSE and input force-torque variation for the different control schemes. A lower RMSE in trajectory tracking indicates higher tracking accuracy, representing the average distance between the actual trajectory and the desired trajectory that the system aims to tracks. Smaller RMSE values mean better performance in tracking the desired trajectory. The proposed STWCA control scheme demonstrates the lowest RMSE for all tracking tasks, including quadrotor position, quadrotor rotation and manipulator orientation, indicating its superior accuracy compared to other control schemes. Notably, the SMCA control scheme also exhibits better tracking performance, specifically in quadrotor position and quadrotor rotation tasks. The results demonstrate that the STWCA control scheme offers superior tracking

accuracy compared to all other control schemes.

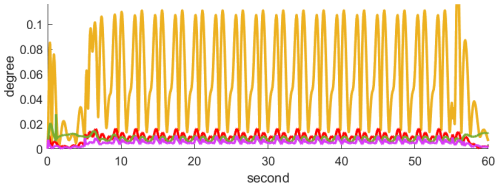
On the other hand, in actuator control, lower variation in input forces and torques reflects more stable control, demonstrating the control algorithm ability to provide consistent and smooth inputs to the system, thus reducing vibration and ensuring stable operation. Upon visual observation of the input forces and torques in Figure 4.17d, 4.17e and 4.17f, it is obvious that all control schemes exhibit almost the same pattern. This suggests that under nominal conditions, all four control schemes yield almost similar input forces and torques. To perform a detailed examination, the force (or torque) variation is calculated as the sum of the absolute differences between two consecutive force values $f_n$ and $f_{n+1}$, expressed as follows:

$$VAR = \sum_{n=1}^{j} |(f_{n+1} - f_n)| \tag{4.1}$$

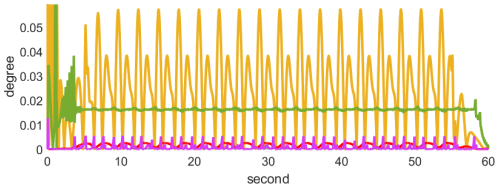
The STWCA control scheme demonstrates the most stable control for quadrotor total thrust, as it exhibits the lowest input force variation. In contrast, the SMCA control scheme shows significant variation, indicating less stable control for this specific parameter. The PID control scheme shows relatively lower input torque variations compared to the SMC, SMCA, and STWCA control schemes, suggesting better stability in controlling manipulator joint 1 and joint 2. Conversely, the SMC and STWCA control schemes exhibit relatively higher variation, indicating less stable control for this particular parameter. Overall, the proposed SMCA and STWCA control schemes shows aggressive actuator control to compensate for the helical motion trajectory, which may lead to increased actuator mechanical wear and tear.



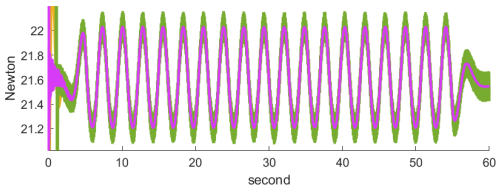

(a) Quadrotor position error.



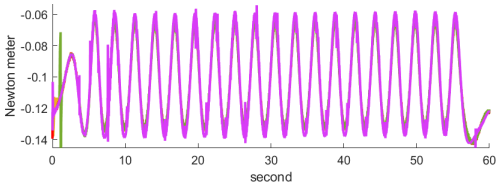

(b) Quadrotor rotation error.



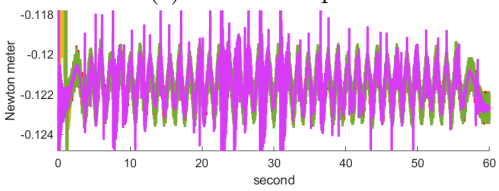

(c) Manipulator orientation error.




(d) Quadrotor total thrust.




(e) Joint 1 torque.




(f) Joint 2 torque.

Figure 4.17 – Helical trajectory tracking error and input force-torque.

Table 4.8 – Helical trajectory tracking RMSE and input force-torque variation.

| Tracking RMSE | | | | |
|---|---|---|---|---|
| Error | PID | SMC | SMCA | STWCA |
| Quadrotor Position | 1.388e-01 | 1.324e-01 | 1.311e-01 | 1.301e-01 |
| Quadrotor Rotation | 7.077e-02 | 1.032e-02 | 8.072e-03 | 5.908e-03 |
| Joint Orientation | 1.153e-01 | 3.182e-03 | 2.195e-02 | 1.285e-03 |

| Input Force and Torque Variation | | | | |
|---|---|---|---|---|
| Force-Torque | PID | SMC | SMCA | STWCA |
| Quadrotor Total Thrust | 49.96 | 92.57 | 710.3 | 24.84 |
| Joint 1 Torque | 2.094 | 28.12 | 20.28 | 33.57 |
| Joint 2 Torque | 1.583e-01 | 20.73 | 6.285 | 20.25 |

Figure 4.18 and 4.18 display the adaptive gains for SMCA and STWCA. The figures reveal that the adaptive gain for SMCA gradually increases during the initial stages of the simulation. This oscillation in the adaptive gain corresponds to the dynamic pitching of the manipulator arm is reflected in the manipulator SMCA adaptive gain. In the case of STWCA, the adaptive gain rapidly increases and then gradually decreases to a minimal value. A similar fluctuation

is observed in the manipulator STWCA adaptive gain, which also reflects the pitching of the manipulator arm.

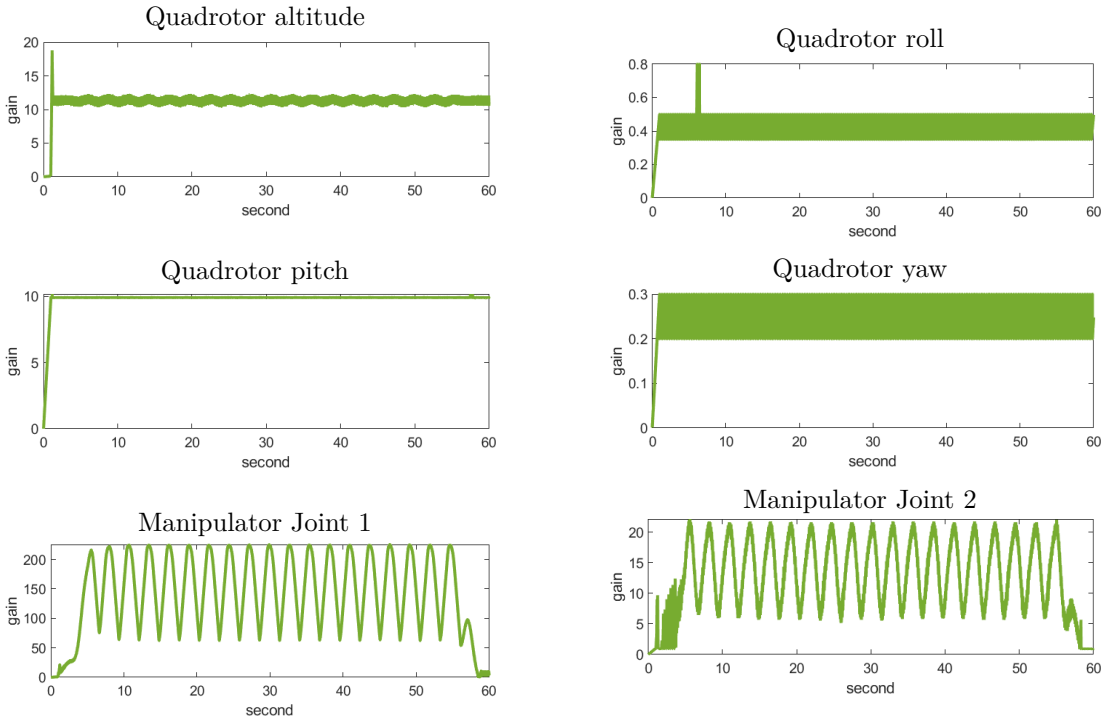



Figure 4.18 – Helical trajectory tracking SMCA adaptive gain.

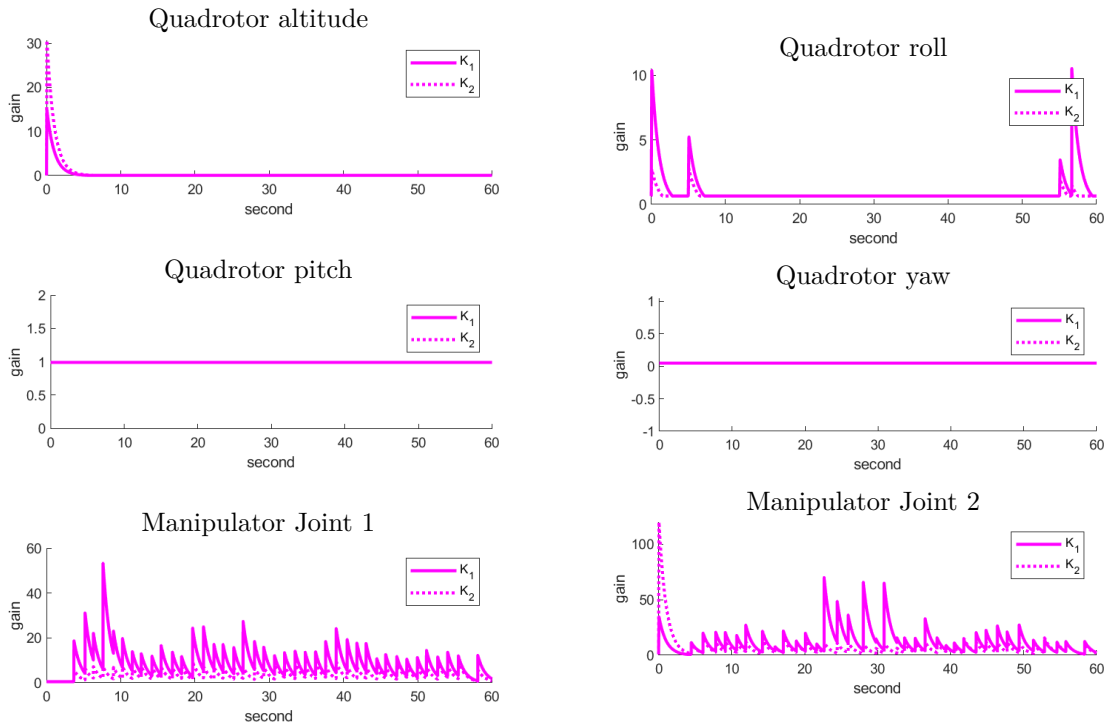


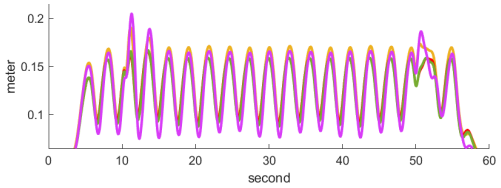

Figure 4.19 – Helical trajectory tracking STWCA adaptive gain.

### 4.5.2 With constant horizontal force

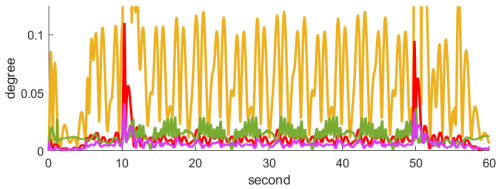
The proposed controller schemes are tested under the presence of a constant horizontal force disturbance while the aerial manipulator tracking the helical trajectory to simulate for the present of external disturbance such as wind. Figure 4.20 illustrates the tracking error and input forces (and torques) of the aerial manipulator in the presence of this kind of disturbance. The quadrotor position RMSE values for PID, SMC, SMCA, and STWCA exhibit only slight differences between these control algorithms, as the $x$- and $y$-position controllers are implemented as PID controllers across all schemes as previously mentioned. The RMSE values increase in a very small amount compared to the nominal conditions.

However, regarding quadrotor rotation and manipulator orientation, STWCA shows much smaller RMSE in comparison to all other control schemes. Despite the constant horizontal force disturbance affecting the stability of STWCA, it still outperforms all other control schemes in terms of performance, as depicted by the Figure 4.20. On the other hand, SMCA shows nearly the same performance in the presence of the horizontal force disturbance as it does under nominal conditions. Similarly, as observed under nominal conditions, for input force and torque variation, the proposed SMCA and STWCA control schemes induce more aggressive actuator control to compensate for the disturbance, which may lead to increase actuator mechanical wear and tear.

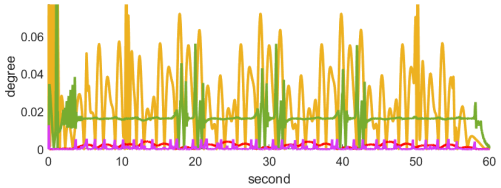
This aggressive control strategy allows the systems to effectively handle the disturbance but may come at the cost of increased wear on the actuators. Figure 4.21 and 4.22 show the adaptive gains for SMCA and STWCA. The fluctuations in gains on the quadrotor roll, quadrotor pitch, and both manipulator joint controls reflect the adaptive response to the horizontal external disturbance.



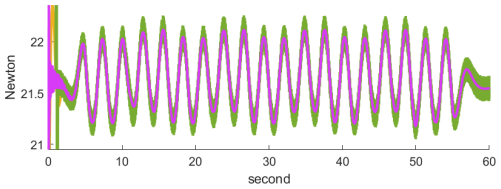

(a) Quadrotor position error.



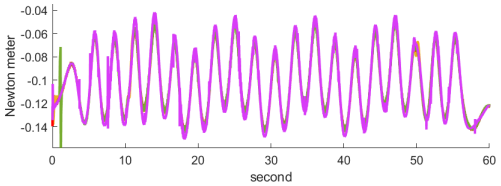

(b) Quadrotor rotation error.



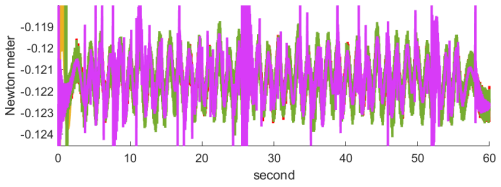

(c) Manipulator orientation error.




(d) Quadrotor total thrust.




(e) Joint 1 torque.




(f) Joint 2 torque.

Figure 4.20 – Helical trajectory tracking error and input force-torque with constant horizontal force disturbance.

Table 4.9 – Helical trajectory tracking RMSE and input force-torque variation with constant horizontal force disturbance.

| Tracking RMSE | | | | |
|---|---|---|---|---|
| Error | PID | SMC | SMCA | STWCA |
| Quadrotor Position | 1.397e-01 | 1.328e-01 | 1.316e-01 | 1.317e-01 |
| Quadrotor Rotation | 1.349e-01 | 1.614e-02 | 1.627e-02 | 7.435e-03 |
| Joint Orientation | 1.301e-01 | 3.791e-03 | 2.251e-02 | 1.283e-03 |

| Input Force and Torque Variation | | | | |
|---|---|---|---|---|
| Force-Torque | PID | SMC | SMCA | STWCA |
| Quadrotor Total Thrust | 47.09 | 94.71 | 732.1 | 24.70 |
| Joint 1 Torque | 2.123 | 28.01 | 27.18 | 28.42 |
| Joint 2 Torque | 1.641e-01 | 20.87 | 7.510 | 18.84 |

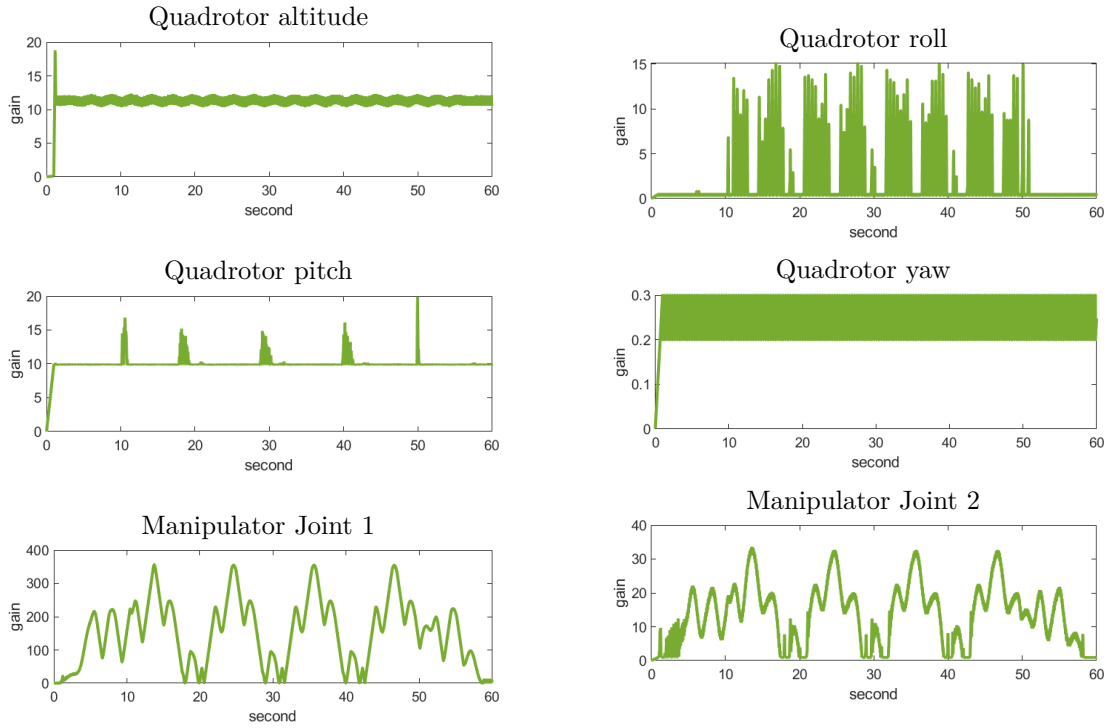



Figure 4.21 – Helical trajectory tracking SMCA adaptive gain with constant horizontal force disturbance.

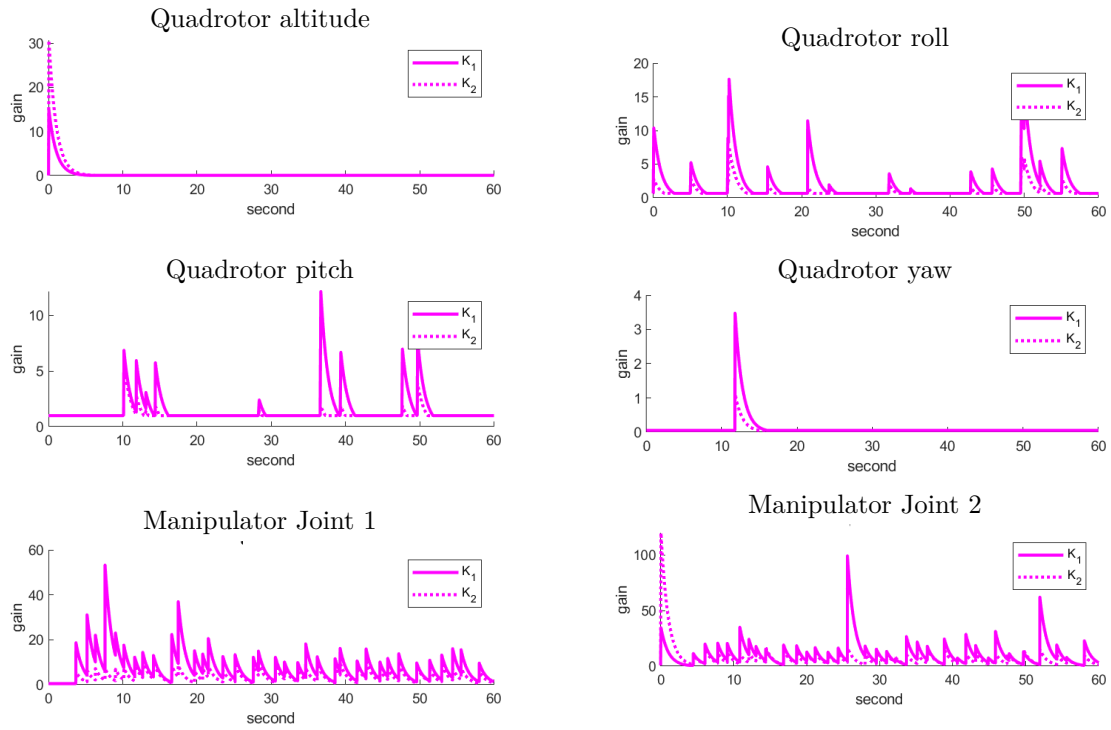



Figure 4.22 – Helical trajectory tracking STWCA adaptive gain with constant horizontal force disturbance.

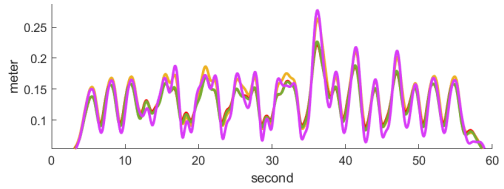
### 4.5.3 With random horizontal force

The proposed controller schemes were further tested under the influence of a random horizontal force disturbance (see Figure 4.5b). Figure 4.23 illustrates the tracking error and input force (and torque) of the aerial manipulator under this disturbance. The STWCA control scheme shows a slight increase in RMSE compared to SMC and SMCA for the quadrotor position tracking task. However, it has the lowest RMSE in quadrotor rotation and manipulator orientation tracking tasks. Notably, the SMCA control scheme also exhibits better tracking performance, specifically in quadrotor position and quadrotor rotation tasks. This indicates considerable accuracy compared to the other two control schemes.

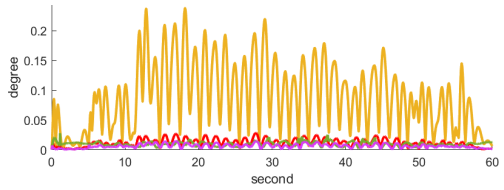
Similar to the previous test scenario, the input force and torque variation of the proposed SMCA and STWCA control schemes show significant values. This demonstrates that under the influence of the random horizontal force, the proposed SMCA and STWCA control schemes also use aggressive actuator control to compensate the disturbance. This aggressive control strategy allows the systems to effectively handle the disturbance. However, it is important to consider the potential long-term effects of aggressive actuator control.

As with the previous simulation, this random horizontal force disturbance also affects the stability of SMCA and STWCA. However, they still exhibit better performance compared to

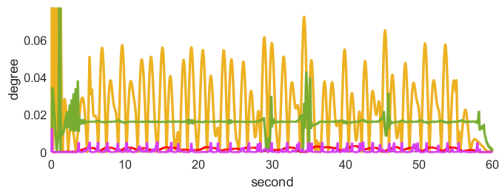
PID and SMC.



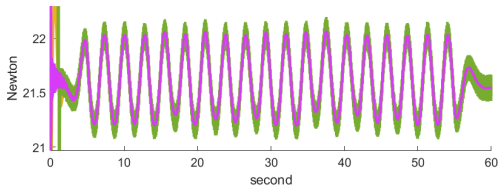

(a) Quadrotor position error.



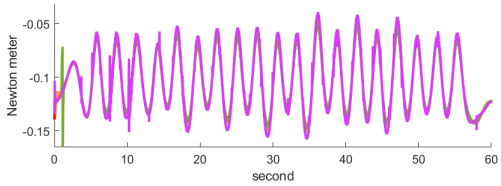

(b) Quadrotor rotation error.



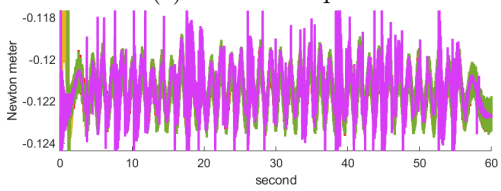

(c) Manipulator orientation error.




(d) Quadrotor total thrust.




(e) Joint 1 torque.




(f) Joint 2 torque.

Figure 4.23 – Helical trajectory tracking error and input force-torque with random horizontal force disturbance.

Table 4.10 – Helical trajectory tracking RMSE and input force-torque variation with random horizontal force disturbance.

| Tracking RMSE | | | | |
|---|---|---|---|---|
| Error | PID | SMC | SMCA | STWCA |
| Quadrotor Position | 1.473e-01 | 1.374e-01 | 1.367e-01 | 1.424e-01 |
| Quadrotor Rotation | 1.204e-01 | 1.509e-02 | 1.054e-02 | 8.098e-03 |
| Joint Orientation | 1.217e-01 | 3.218e-03 | 2.198e-02 | 1.589e-03 |

| Input Force and Torque Variation | | | | |
|---|---|---|---|---|
| Force-Torque | PID | SMC | SMCA | STWCA |
| Quadrotor Total Thrust | 46.63 | 92.95 | 717.4 | 23.86 |
| Joint 1 Torque | 2.270 | 28.25 | 20.76 | 41.08 |
| Joint 2 Torque | 1.659e-01 | 20.73 | 6.406 | 31.36 |

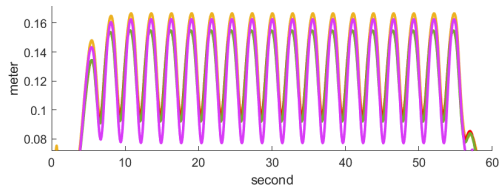
### 4.5.4 With point mass

In this test, a small and concentrated mass is introduced to the tip of the manipulator arm, and the control scheme's response to this additional mass is observed. The point mass accounts for approximately 5% of the total system mass. This test aims to assess how the control algorithms react to these changes and uncertainties in the system.

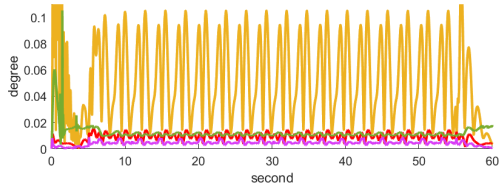
During the initial phase of the simulation, as depicted in Figure 4.24, all control schemes are capable of quickly responding to these uncertainties and bringing the aerial manipulator back on track. This plot closely resembles the simulation without any disturbance, indicating the effectiveness and the robustness of the control schemes.

Regarding quadrotor position RMSE, all control schemes show nearly the same performance as under nominal conditions. However, for quadrotor rotation, STWCA exhibits a decrease in RMSE, while SMCA shows an increase compared to nominal conditions. Additionally, for manipulator orientation, STWCA shows a significant increase in RMSE. Among all four control schemes, STWCA stands out with superior performance, displaying the lowest RMSE in the point mass test.

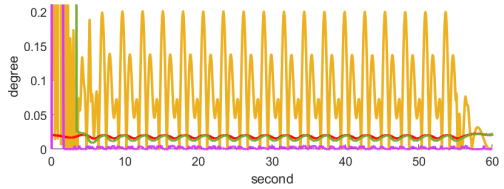
Furthermore, the input force and torque variation of the proposed SMCA and STWCA control schemes show exceptional values. This demonstrates that under the influence of these uncertainties, the proposed SMCA and STWCA control schemes also employ aggressive actuator control to compensate for the additional mass. It is evident that the point mass loading has an impact on the performance of the control schemes, as reflected in the RMSE values and input force-torque variation for each control scheme. As observed in the previous hovering test, the introduction of the point mass loading affects the initial-stage adaptive gain of both the SMCA and STWCA controllers, as illustrated in Figure 4.25 and 4.26.



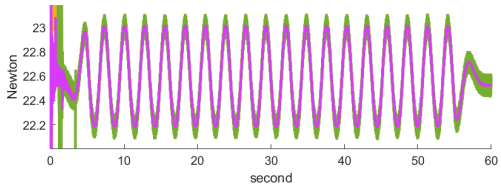

(a) Quadrotor position error.



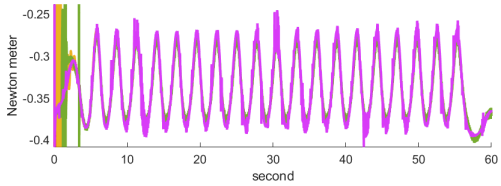

(b) Quadrotor rotation error.



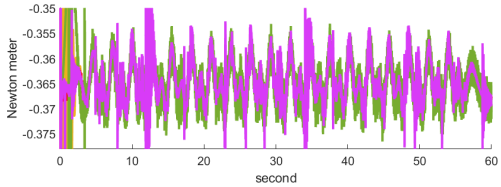

(c) Manipulator orientation error.




(d) Quadrotor total thrust.




(e) Joint 1 torque.




(f) Joint 2 torque.

Figure 4.24 – Helical trajectory tracking error and input force-torque with point mass loading.

Table 4.11 – Helical trajectory tracking RMSE and input force-torque variation with point mass loading.

| Tracking RMSE | | | | |
|---|---|---|---|---|
| Error | PID | SMC | SMCA | STWCA |
| Quadrotor Position | 1.374e-01 | 1.313e-01 | 1.299e-01 | 1.287e-01 |
| Quadrotor Rotation | 6.349e-02 | 9.997e-03 | 1.084e-02 | 4.927e-03 |
| Joint Orientation | 4.897e-01 | 2.448e-02 | 2.255e-02 | 3.254e-03 |

| Input Force and Torque Variation | | | | |
|---|---|---|---|---|
| Force-Torque | PID | SMC | SMCA | STWCA |
| Quadrotor Total Thrust | 46.07 | 101.2 | 709.0 | 55.63 |
| Joint 1 Torque | 3.450 | 39.67 | 79.97 | 141.9 |
| Joint 2 Torque | 4.405e-01 | 32.45 | 72.81 | 97.96 |

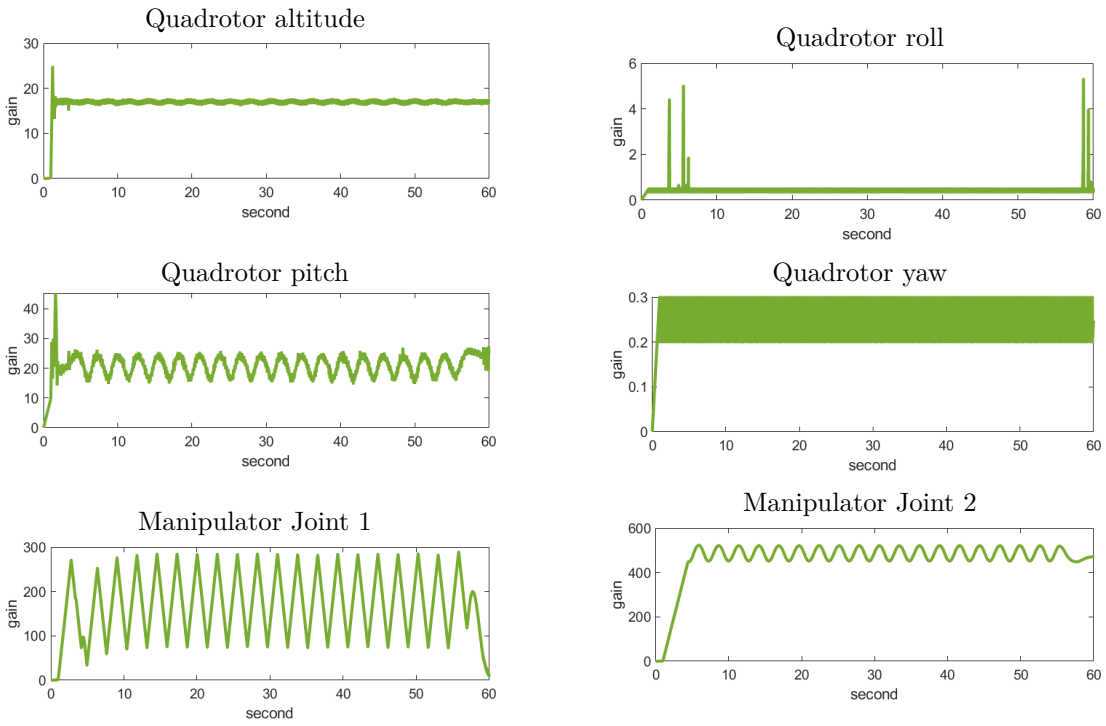



Figure 4.25 – Helical trajectory tracking SMCA adaptive gain with point mass loading.

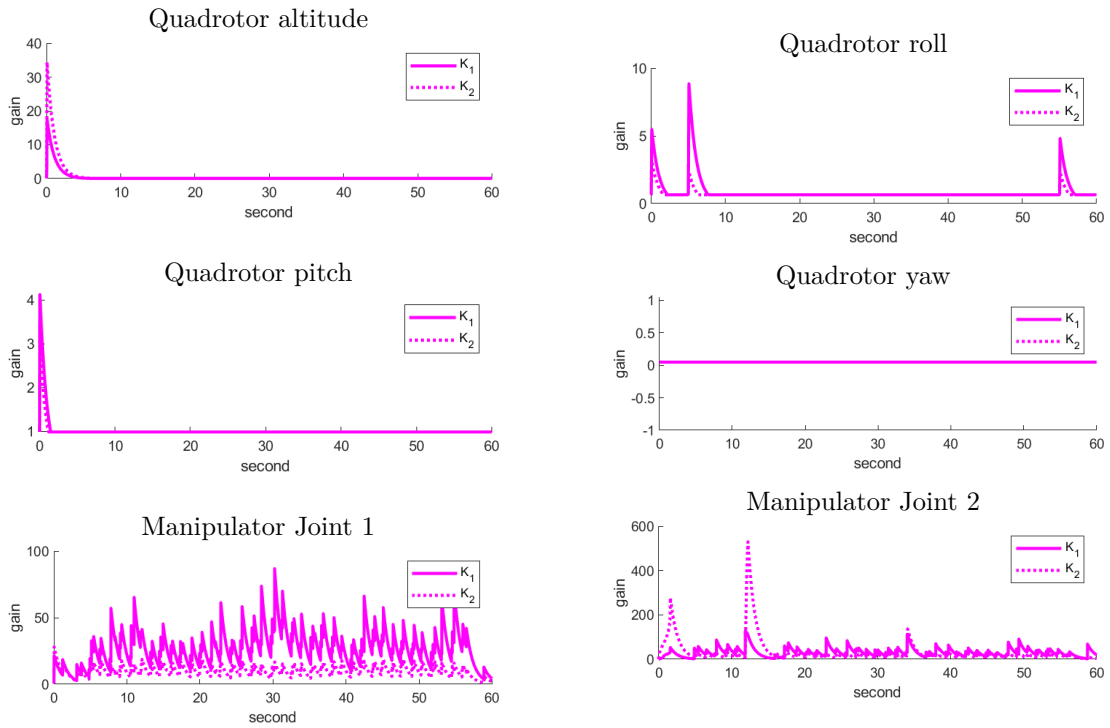


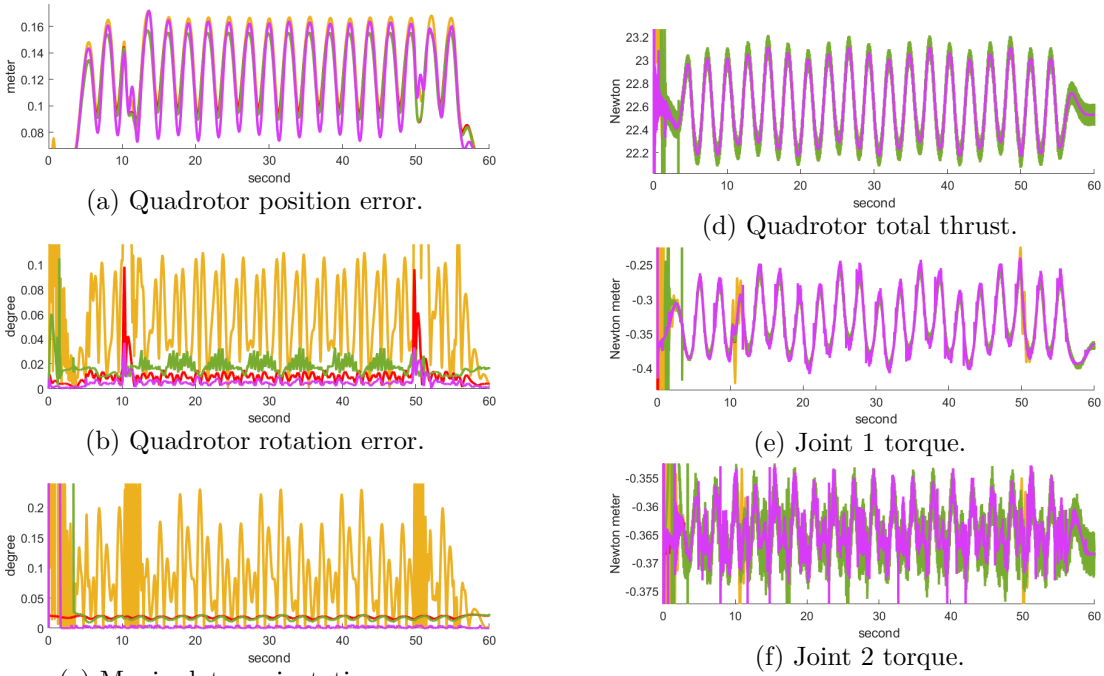

Figure 4.26 – Helical trajectory tracking STWCA adaptive gain with point mass loading.

### 4.5.5 With mix disturbance

Additionally, two more simulations have been conducted with mixed disturbances, combining horizontal force and point mass effects. The first mix disturbance is involving a constant horizontal force and point mass disturbance, while the second mix disturbance includes a random horizontal force and point mass disturbance. Figure 4.27 and 4.28 depict the plots of the helical trajectory tracking error and input force-torque for the constant and random mix disturbances, respectively.

Observing the tracking error plots for both disturbances, it becomes evident that these plots are actually a combination of individual tests for horizontal force and point mass. Table 4.12 presents the tracking RMSE and input force-torque variation for the simulation with the mix of constant horizontal force disturbance, while Table 4.13 shows the corresponding results for the simulation with the mix of random horizontal force disturbance.

### Mix constant horizontal force




(a) Quadrotor position error.

(b) Quadrotor rotation error.

(c) Manipulator orientation error.

(d) Quadrotor total thrust.

(e) Joint 1 torque.

(f) Joint 2 torque.

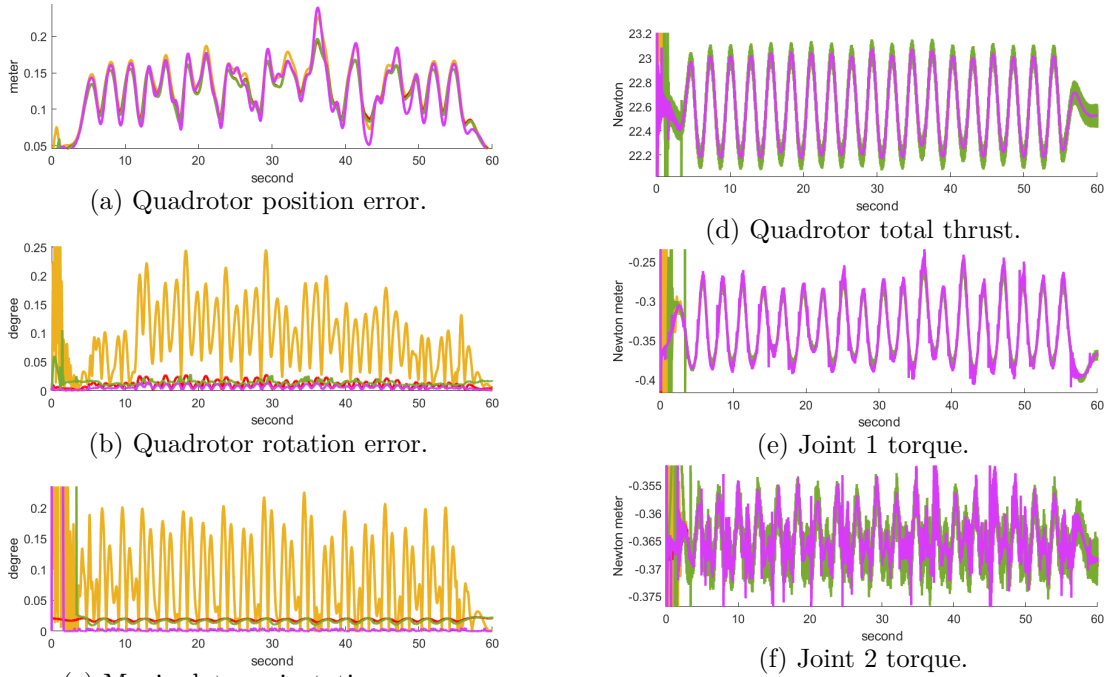
Figure 4.27 – Helical trajectory tracking error and input force-torque with mix constant horizontal force disturbance.

Table 4.12 – Helical trajectory tracking RMSE and input force-torque variation with mix constant horizontal force disturbance.

| Tracking RMSE | | | | |
|---|---|---|---|---|
| Error | PID | SMC | SMCA | STWCA |
| Quadrotor Position | 1.360e-01 | 1.300e-01 | 1.286e-01 | 1.278e-01 |
| Quadrotor Rotation | 1.187e-01 | 1.481e-02 | 1.886e-02 | 6.693e-03 |
| Joint Orientation | 4.943e-01 | 2.470e-02 | 2.262e-02 | 3.321e-03 |

| Input Force and Torque Variation | | | | |
|---|---|---|---|---|
| Force-Torque | PID | SMC | SMCA | STWCA |
| Quadrotor Total Thrust | 46.79 | 103.0 | 720.2 | 36.67 |
| Joint 1 Torque | 3.932 | 40.79 | 92.82 | 82.13 |
| Joint 2 Torque | 5.562e-01 | 34.48 | 86.07 | 43.03 |

### Mix random horizontal force




(a) Quadrotor position error.

(b) Quadrotor rotation error.

(c) Manipulator orientation error.

(d) Quadrotor total thrust.

(e) Joint 1 torque.

(f) Joint 2 torque.

Figure 4.28 – Helical trajectory tracking error and input force-torque with mix random horizontal force disturbance.

Table 4.13 – Helical trajectory tracking RMSE and input force-torque variation with mix random horizontal force disturbance.

| Tracking RMSE | | | | |
|---|---|---|---|---|
| Error | PID | SMC | SMCA | STWCA |
| Quadrotor Position | 1.442e-01 | 1.354e-01 | 1.345e-01 | 1.383e-01 |
| Quadrotor Rotation | 1.214e-01 | 1.475e-02 | 1.429e-02 | 7.545e-03 |
| Joint Orientation | 4.835e-01 | 2.453e-02 | 2.278e-02 | 3.260e-03 |

| Input Force and Torque Variation | | | | |
|---|---|---|---|---|
| Force-Torque | PID | SMC | SMCA | STWCA |
| Quadrotor Total Thrust | 46.56 | 102.5 | 711.8 | 42.58 |
| Joint 1 Torque | 3.324 | 41.55 | 81.80 | 99.69 |
| Joint 2 Torque | 4.440e-01 | 34.70 | 75.17 | 58.42 |

## 4.6 Conclusions

In this simulation, multiple tests were conducted to assess the performance of the proposed SMCA and STWCA control schemes. The first test conducted was a hovering test, which served to evaluate the basic stability and control of the quadrotor, acting as the platform for the manipulator arm. Once the quadrotor demonstrated stable hovering, more complex helical trajectory tests were performed. In these helical trajectory tests, perturbations were introduced, including external disturbances on the quadrotor and the presence or absence of a point mass at the end of the manipulator arm.

The test results indicate that the adaptive gain of both SMCA and STWCA dynamically adjusts to the external disturbances and uncertainties in the system. Despite the fact that both SMCA and STWCA has a few additional parameters, the tuning process is not complicated. With proper parameter tuning, good performance can be achieved.

In summary, the STWCA control scheme proves to be highly effective in achieving accurate trajectory tracking across all scenarios. On the other hand, the SMCA control scheme shows limited performance improvement for helical trajectory tracking. However, it is important to be mindful of the aggressive actuator control used in both schemes, as it may result in increased mechanical wear and tear over time. Careful consideration of these factors is necessary when selecting the most appropriate control scheme for specific applications.

The STWCA control scheme demonstrates accurate helical tracking ability in various test scenarios, while the SMCA control scheme presents some enhancements in performance. When deciding on the most suitable scheme for specific applications, it's important to consider the mechanical wear and tear resulting from the aggressive actuator control in both schemes.

Chapter 5

# UAM Overall Performance

This manuscript presents a study on adaptive gain first-order sliding mode control (SMCA) and adaptive gain super-twisting sliding mode control (STWCA). The study focuses on an aerial manipulator consisting of a quadrotor UAV with a two-degree-of-freedom rigid manipulator arm. The simulation involves position and attitude tracking of quadrotor with the manipulator arm periodically pitches up and down.

To evaluate the system's tracking capability, a helical motion trajectory is used, which simulates both translational and rotational motion simultaneously. A comparison is conducted between the two proposed control schemes mentioned above and two other control strategies: linear PID control and nonlinear SMC. The controllers are simulated under different conditions, including the presence or absence of constant and random horizontal force on the quadrotor and the presence or absence of a point mass loading at the end of the manipulator arm. These simulations allow for an assessment of the controllers' performance under various scenarios.

In this chapter, an overall view of all test scenarios is presented. Comparisons are made across the six test scenarios conducted, which include scenarios without disturbance, with constant horizontal force, with random horizontal force, with point mass, with constant mixed disturbance and with random mixed disturbance. These comparisons allow for a visual indication of the performance of the proposed control scheme under all conditions.

## 5.1 UAM performance

The proposed control scheme simulation for the UAM system in this study consists of six controllers. Among these, four controllers are responsible for controlling the input force of the quadrotor actuator motors, while the other two controllers focus on controlling the input torque of the manipulator arm's joint actuators. Quadrotor UAVs are underactuated systems, with only four input control actions (altitude, roll, pitch, and yaw), but they have six degrees of freedom that need to be controlled. This presents challenges in controlling the UAV motion while serving as the floating base for the manipulator arm. Additionally, the reaction forces and torques resulting from the weight and motion of the manipulator arm introduce additional disturbances to the control of the UAV. From the simulation, three key values are extracted: the quadrotor position error, quadrotor rotation error and manipulator orientation error. Additionally, data on

the input force of the quadrotor and the input torque of the manipulator arm are obtained and represented as quadrotor total thrust, joint 1 torque and joint 2 torque. These values are crucial for assessing the performance of the control schemes in the study. (Throughout this chapter, the graphs of simulation results follow a consistent color scheme: PID represented in yellow, SMC (sliding mode control) in red, SMCA (adaptive SMC) in green, and STWCA (adaptive super-twisting) in purple.)

### 5.1.1 Trajectory tracking normalize RMSE

One of the method for performance indicator in this study is by measuring tracking trajectory root mean square error (RMSE). Figure 5.1, 5.2 and 5.3 shows the normalize version of RMSE for quadrotor position error, quadrotor rotation error and manipulator orientation error respectively for helical trajectory tracking. The normalize RMSE of quadrotor position tracking errors shows that there is a slight performance difference across all simulated conditions for each control scheme. This situation arises due to the control of two out of the three positional axes for all control scheme is employed as PID controller. Specifically, the PID controller is used for both the $x$-axis and the $y$-axis. The normalize RMSE of quadrotor rotation tracking reveals that all variations of SMC exhibit significantly improved performance when compared to PID control. Among these, the adaptive gain super-twisting sliding mode control stands out with the most superior performance. Finally, the normalize RMSE of manipulator orientation error also demonstrates considerably enhanced performance with the adaptive gain super-twisting sliding mode control exhibits the most exceptional performance. Conversely, the adaptive gain first-order sliding mode control consistently maintains its performance at a nearly constant level.

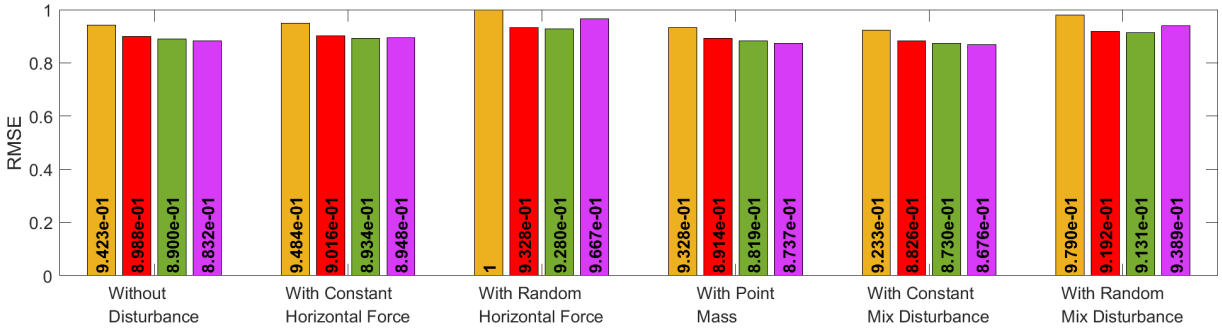



Figure 5.1 – Normalize RMSE of quadrotor position tracking error.

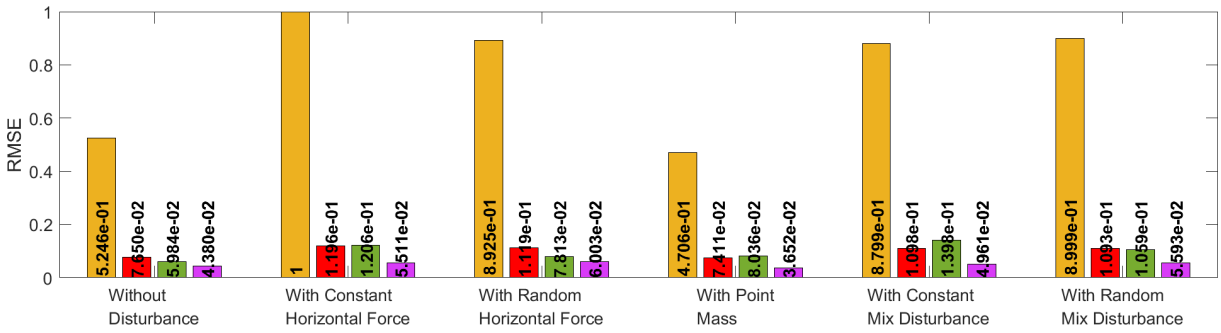



Figure 5.2 – Normalize RMSE of quadrotor rotation tracking error.

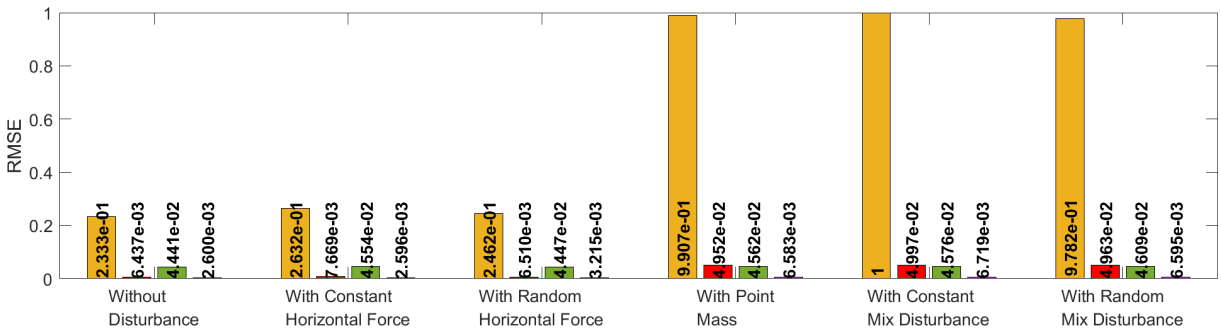



Figure 5.3 – Normalize RMSE of manipulator orientation error.

### 5.1.2 Input force-torque normalize variation

Another method for performance indicator in this study consist in analyzing force or torque variation. The variation is the sum of the absolute different between two consecutive force or torque as in (4.1). These variations are the total force acting on the quadrotor, also joint 1 torque and joint 2 torque of the manipulator arm. The topic on force acting on quadrotor is discuss in Subsection 2.4.2. While in the case of manipulator joint torque, it is the torque of individual joint actuator as discuss in Subsection 2.5.2. Figure 5.4, 5.5 and 5.6 shows the normalize variation for quadrotor total thrust, manipulator joint 1 torque and manipulator joint 2 torque respectively. In terms of the normalized variation of quadrotor total thrust, the adaptive gain super-twisting sliding mode control demonstrates consistently lower total thrust values across the majority of simulated conditions. On the other hand, the adaptive gain first-order sliding mode control registers the highest total thrust. Furthermore, upon examining the normalized variation of manipulator joint 1 torque and joint 2 torque, it is obvious that the adaptive gain super-twisting sliding mode control exhibits the most significant total torque variations when a point mass is introduced at the end of the manipulator arm. Subsequently, the adaptive gain first-order sliding mode control demonstrates the next highest total torque variations for the same simulated condition.

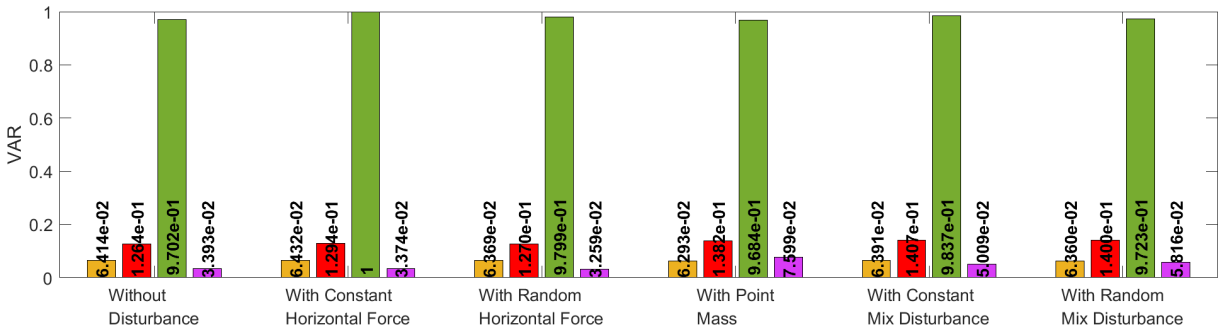



Figure 5.4 – Normalize variation of quadrotor total thrust.

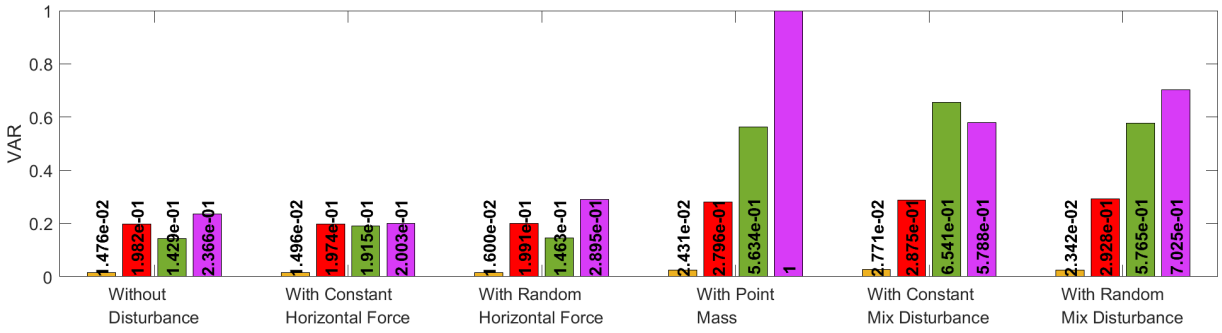



Figure 5.5 – Normalize variation of manipulator joint 1 torque.

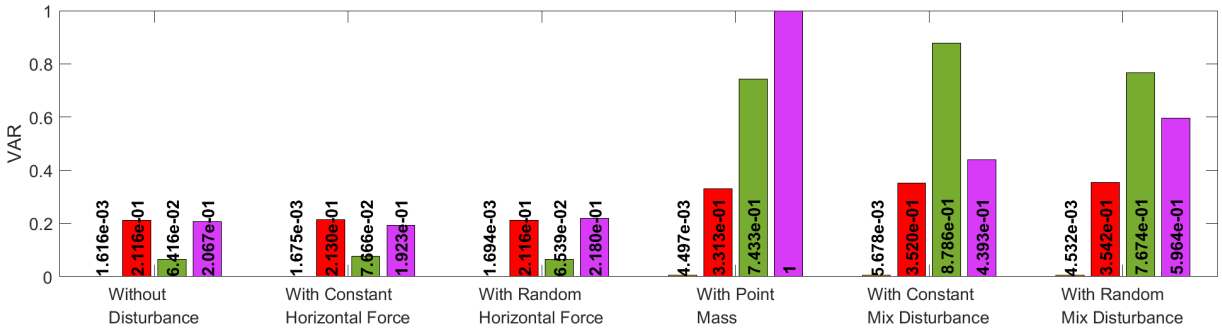



Figure 5.6 – Normalized variation of manipulator joint 2 torque.

Chapter 6

# Conclusions and Perspectives

## Conclusions

In this study the final performance of the two proposed control scheme (that is SMCA and STWCA) is tested for helical trajectory motion with moving manipulator arm. A comparison is made to linear PID control and non-linear SMC. There are 24 simulations conducted which are group into six test configurations. These test configurations are trajectory tracking without any disturbance, with constant horizontal force on quadrotor, with random horizontal force on quadrotor, manipulator arm point mass loading, mix of constant horizontal force with mass loading and mix of random horizontal force with mass loading. From the simulations six key values are examined: the quadrotor position error, the quadrotor rotation error and the manipulator orientation error along with the input force of the quadrotor and the input torque of both manipulator arm joints.

The main challenges in this study is to control the movement of UAV. Quadrotors UAV are under-actuated systems in which only four inputs, that is altitude, roll, pitch and yaw, are used to control motion in six degrees of freedom. The UAV changes position and attitude by modifying the speed of four actuator motors. The four inputs control is achieved by collectively increasing or decreasing the speeds of all four UAV actuator motors while preserving the total thrust. Consequently these four motors of UAV is highly coupled in which a changes in one control result in changes in other control. As a coupled system, the manipulator arm performance is also affected by the stability of its floating base (i.e the UAV). In future a further investigation can be done to find a good balance tuning between the four UAV control (also manipulator arm) which will not exceed the maximum total thrust.

The performance evaluation is based on tracking RMSE and input force (or torque) variation. Table (6.1) shows the average of both tracking RMSE and force variation for all six test configuration being examined. STWCA has the best value of tracking RMSE nevertheless its force (or torque) variation is much higher as opposed to PID and SMC control scheme. SMCA shows a smaller value of tracking RMSE as opposed to PID and SMC but with the highest value of force (or torque) variation. In reference to the two proposed control scheme, it is worth noting that STWCA has the best performance with a balance of best tracking RMSE and reduced force (or torque) variation. This indicates that STWCA combines accuracy and robustness, with

SMCA following as the second-best controller in this aspect.

Table 6.1 – Average of tracking RMSE and input force variation.

(a) Average of tracking RMSE.

| Control Scheme | Quadrotor Position | Quadrotor Rotation | Manipulator Orientation |
|---|---|---|---|
| PID | 0.1406 | 0.1049 | 0.3058 |
| SMC | 0.1332 | 1.352e-02 | 1.398e-02 |
| SMCA | 0.1332 | 1.315e-02 | 2.240e-02 |
| STWCA | 0.1332 | 6.768e-03 | 2.332e-03 |

(b) Average of input force variation.

| Control Scheme | Quadrotor Total Thrust | Manipulator Joint 1 Torque | Manipulator Joint 2 Torque |
|---|---|---|---|
| PID | 46.68 | 2.866 | 0.3215 |
| SMC | 97.82 | 34.40 | 27.33 |
| SMCA | 716.8 | 53.80 | 42.38 |
| STWCA | 34.71 | 71.13 | 43.31 |

## Perspectives

Several points can be considered for future work on the control of UAM, as follows:

— Experimental tests could provide validation to determine whether the simulated results hold true under actual physical conditions.

— Simulation with different types of complex trajectories, such as the Lissajous trajectory, can be employed to further verify the performance of the proposed controllers.

— Testing for different types of external disturbances and introducing noise within the system to mimic more real-world scenarios.

— Investigating alternative methods for computing the desired roll and pitch angles can potentially yield more accurate results.

— Using the proposed control scheme for the $x$- and $y$-position control of the quadrotor.

— To find a good balance tuning between the four UAV control (also manipulator arm) which will not exceed the maximum total thrust (or torque).

— In this study, differentiation was computed using a transfer function; however, it can also be accomplished using sliding mode observers or differentiators.

These suggestions aim to enhance the performance and robustness of future research on UAM control.

Appendix A

# List of Symbols

In this writing the subscript represent the base frame of vector value while the superscript is the reference frame. In the absent of superscript it is reference in the same frame.

## A.1 Reference Frame

| | |
|---|---|
| $\{x, y, z\}$ | main axes in three dimensional geometric space |
| $\{\phi, \theta, \psi\}$ | Euler angle of roll, pitch and yaw |
| $F_W$ | inertial fixed frame or inertial frame |
| $F_B$ | quadrotor body frame at its centre of mass |
| $F_0$ | manipulator arm base frame |
| $F_1$ | manipulator arm link 1 joint frame |
| $F_2$ | manipulator arm link 2 joint frame |
| $F_e$ | manipulator arm end effector frame |
| $\{x_W, y_W, z_W\}$ | main axes of inertial frame |
| $\{x_B, y_B, z_B\}$ | main axes of body frame |
| $\{x_0, y_0, z_0\}$ | main axes of manipulator base frame |
| $\{x_1, y_1, z_1\}$ | main axes of joint 1 frame |
| $\{x_2, y_2, z_2\}$ | main axes of joint 2 frame |
| $\{x_e, y_e, z_e\}$ | main axes of end effector frame |
| $O_W$ | inertial frame origin |
| $O_B$ | body frame origin |
| $O_0$ | manipulator base frame origin |
| $O_1$ | joint 1 frame origin |
| $O_2$ | joint 2 frame origin |
| $O_e$ | end effector frame origin |
| $\boldsymbol{\xi} = [x\ y\ z]^T$ | quadrotor position vector |
| $\boldsymbol{\eta} = [\phi\ \theta\ \psi]^T$ | quadrotor rotation vector |
| $\boldsymbol{\nu} = [u\ v\ w]^T$ | quadrotor body frame linear velocity |
| $\boldsymbol{\omega} = [p\ q\ r]^T$ | quadrotor body frame angular velocity |

| | |
|---|---|
| $\boldsymbol{\xi_1} = [x_1\ y_1\ z_1]^T$ | joint 1 position vector wrt manipulator base frame |
| $\boldsymbol{\xi_2} = [x_2\ y_2\ z_2]^T$ | joint 2 position vector wrt joint 1 frame |
| $\boldsymbol{\xi_e} = [x_e\ y_e\ z_e]^T$ | end effector position vector wrt manipulator base frame |
| $\boldsymbol{q} = [q_1\ q_2]^T$ | manipulator joint position vector |
| $q_1$ | $F_1$ orientation wrt $F_0$ |
| $q_2$ | $F_2$ orientation wrt $F_1$ |

## A.2 Rotation Matrix

| | |
|---|---|
| $\{\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}, \hat{\boldsymbol{z}}\}$ | unit vector in three dimensional geometric space |
| ${}^A\boldsymbol{R}_B = [{}^A\hat{\boldsymbol{x}}_B, {}^A\hat{\boldsymbol{y}}_B, {}^A\hat{\boldsymbol{z}}_B]$ | rotation matrix of unit vectors in frame $B$ relative to frame $A$ |
| $\boldsymbol{V}_A \in \mathbb{R}^3$ | vector A |
| $\boldsymbol{V}_B \in \mathbb{R}^3$ | vector B |
| $\boldsymbol{\xi}_A = [x_a\ y_a\ z_a]^T$ | vector A position vector |
| $\boldsymbol{\xi}_B = [x_b\ y_b\ z_b]^T$ | vector B position vector |
| $\boldsymbol{R_x} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & C\phi & -S\phi \\ 0 & S\phi & C\phi \end{bmatrix}$ | rotation matrix about x-axis |
| $\boldsymbol{R_y} = \begin{bmatrix} C\theta & 0 & S\theta \\ 0 & 1 & 0 \\ -S\theta & 0 & C\theta \end{bmatrix}$ | rotation matrix about y-axis |
| $\boldsymbol{R_z} = \begin{bmatrix} C\psi & -S\psi & 0 \\ S\psi & C\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$ | rotation matrix about z-axis |

## A.3 Quadrotor Modelling

| | |
|---|---|
| ${}^W\boldsymbol{R}_B = \boldsymbol{R_z}(\psi)\,\boldsymbol{R_y}(\theta)\,\boldsymbol{R_x}(\phi)$ | rotation matrix from the body frame to the inertial frame |
| ${}^B\boldsymbol{R}_W$ | rotation matrix from inertial frame to the body frame |
| ${}^W\boldsymbol{T}_B$ | quadrotor transfer matrix from body frame to inertial frame |
| ${}^B\boldsymbol{T}_W$ | quadrotor transfer matrix from inertial frame to body frame |
| $C\psi$ | cosine psi (roll) |
| $C\theta$ | cosine theta (pitch) |

| | |
|---|---|
| $C\phi$ | cosine phi (yaw) |
| $S\psi$ | sine psi (roll) |
| $S\theta$ | sine theta (pitch) |
| $S\phi$ | sine phi (yaw) |
| $T\psi$ | tangent psi (roll) |
| $T\theta$ | tangent theta (pitch) |
| $T\phi$ | tangent phi (yaw) |
| $\dot{\boldsymbol{\xi}} = [\dot{x}\ \dot{y}\ \dot{z}]^T$ | quadrotor linear velocity |
| $\ddot{\boldsymbol{\xi}} = [\ddot{x}\ \ddot{y}\ \ddot{z}]^T$ | quadrotor linear acceleration |
| $\dot{\boldsymbol{\eta}} = [\dot{\phi}\ \dot{\theta}\ \dot{\psi}]^T$ | quadrotor Euler angle rate |
| $\boldsymbol{\omega} = [p\ q\ r]^T$ | quadrotor angular velocity |
| $\dot{\boldsymbol{\omega}} = [\dot{p}\ \dot{q}\ \dot{r}]^T$ | quadrotor angular acceleration |
| $F_i$ | quadrotor upward thrust |
| $M_i$ | quadrotor moment of force of torque |
| $A$ | propeller cross sectional area |
| $r$ | propeller radius |
| $d$ | distance from centre of the propeller to quadrotor centre of mass |
| $C_T$ | thrust aerodynamic coefficients |
| $C_D$ | torque aerodynamic coefficients |
| $C_F$ | propeller thrust coefficients |
| $C_M$ | propeller torque coefficients |
| $m$ | mass |
| $\boldsymbol{g}$ | gravitational force |
| $\boldsymbol{F} = [f_x\ f_y\ f_z]^T$ | force vector |
| $\boldsymbol{\tau} = [\tau_x\ \tau_y\ \tau_z]^T$ | torque vector |
| $U_1 = f_z$ | upward thrust |
| $U_2 = \tau_x$ | rolling torque |
| $U_3 = \tau_y$ | pitching torque |
| $U_4 = \tau_z$ | yawing torque |
| $I$ | inertia matrix |
| $I_{xx}$ | inertia matrix along x-axis |
| $I_{yy}$ | inertia matrix along y-axis |
| $I_{zz}$ | inertia matrix along z-axis |

## A.4 Manipulator Modelling

| | |
|---|---|
| $j$ | manipulator arm link number |
| $\boldsymbol{f}_j$ | three dimensonal force on link $j$ |
| $\boldsymbol{m}_j$ | three dimensonal moment of force or torque on link $j$ |
| $\boldsymbol{f}_{tj}$ | total force acting on link $j$ |
| $\boldsymbol{m}_{tj}$ | total torque acting on link $j$ |
| $\boldsymbol{f}_{ej}$ | force exerted by link $j$ on the environment |
| $\boldsymbol{m}_{ej}$ | torque exerted by link $j$ on the environment |
| $\boldsymbol{L}_j$ | position vector of link $j$ |
| $\boldsymbol{MS}_j$ | vector of the first moment of inertia of link $j$ |
| $\boldsymbol{\Gamma}$ | vector of joint forces or torques |
| $\boldsymbol{q}$ | vector of joint positions |
| $\dot{\boldsymbol{q}}$ | vector of joint velocities |
| $\ddot{\boldsymbol{q}}$ | vector of joint accelerations |
| $\boldsymbol{F}_e$ | vector of forces and moments exerted by the robot on the environment |
| $\boldsymbol{M}(\boldsymbol{q})$ | inertia matrix of the manipulator |
| $\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ | vector of Coriolis and centrifugal torque |
| $\boldsymbol{Q}(\boldsymbol{q})$ | vector of gravity effects |
| ${}^j\boldsymbol{\omega}_j$ | angular velocity of link $j$ |
| ${}^j\dot{\boldsymbol{\omega}}_j$ | angular acceleration of link $j$ |
| ${}^j\boldsymbol{\nu}_j$ | linear velocity of link $j$ |
| ${}^j\dot{\boldsymbol{\nu}}_j$ | linear acceleration of link $j$ |

## A.5 Modelling of Coupled System

| | |
|---|---|
| ${}^0\boldsymbol{R}_W$ | rotation matrix from inertial frame to manipulator base frame |
| ${}^0\boldsymbol{R}_B$ | rotation matrix from body frame to manipulator base frame |
| $\boldsymbol{\omega}_0$ | angular velocity of manipulator base frame |
| $\dot{\boldsymbol{\omega}}_0$ | angular acceleration of manipulator base frame |
| $\dot{\boldsymbol{\nu}}_0$ | linear acceleration of manipulator base frame |
| $\boldsymbol{f}_0 = [f_{0.x}\ f_{0.y}\ f_{0.z}]^T$ | force vector on manipulator base frame |
| $\boldsymbol{m}_0 = [m_{0.x}\ m_{0.y}\ m_{0.z}]^T$ | torque vector on manipulator base frame |
| $\boldsymbol{f}_W = [f_{W.x}\ f_{W.y}\ f_{W.z}]^T$ | reaction force applied to quadrotor COM |
| $\boldsymbol{\tau}_B = [\tau_{B.x}\ \tau_{B.y}\ \tau_{B.z}]^T$ | reaction torque applied to quadrotor COM |

## A.6 Differential Kinematic

| | |
|---|---|
| ${}^{W}\boldsymbol{R}_e$ | rotation matrix from end effector frame to inertial frame |
| ${}^{B}\boldsymbol{R}_e$ | rotation matrix from end effector frame to body frame |
| $\boldsymbol{\xi}_e$ | end effector position vector wrt body frame |
| $\dot{\boldsymbol{\xi}}_e$ | end effector linear velocity wrt body frame |
| $\boldsymbol{\xi}_T$ | end effector position vector wrt inertial frame |
| $\dot{\boldsymbol{\xi}}_T$ | end effector linear velocity wrt inertial frame |
| $\boldsymbol{\eta}_e$ | end effector rotation vector wrt body frame |
| $\dot{\boldsymbol{\eta}}_e$ | end effector Euler angle rate wrt body frame |
| $\boldsymbol{\eta}_T$ | end effector rotation vector wrt inertial frame |
| $\dot{\boldsymbol{\eta}}_T$ | end effector Euler angle rate wrt inertial frame |
| $\boldsymbol{\omega}_e$ | end effector angular velocity wrt body frame |
| $\boldsymbol{\omega}_T$ | end effector angular velocity wrt inertial frame |
| $\boldsymbol{S}(\cdot)$ | $(3 \times 3)$ skew symmetric matrix |
| $\boldsymbol{J}_*(\cdot)$ | Jacobian matrix |
| $\boldsymbol{T}(\boldsymbol{\eta}_*) = \begin{bmatrix} 0 & -S\psi_* & C\psi_* C\theta_* \\ 0 & C\psi_* & S\psi_* C\theta_* \\ 1 & 0 & -S\theta_* \end{bmatrix}$ | transformation matrix of Euler angle rate to angular velocity in a same frame |
| $\boldsymbol{I}_3$ | identity matrix |
| $\boldsymbol{O}_3$ | null matrix |
| $\boldsymbol{\kappa} = \begin{bmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\beta} \end{bmatrix}$ | quadrotor position and rotation vector |
| $\boldsymbol{\alpha} = \begin{bmatrix} \boldsymbol{\xi} \\ \psi \end{bmatrix}$ | quadrotor vector of position and yaw angle |
| $\boldsymbol{\beta} = \begin{bmatrix} \theta \\ \phi \end{bmatrix}$ | quadrotor vector of roll and pitch angle |
| $\boldsymbol{\zeta} = \begin{bmatrix} \boldsymbol{\xi} \\ \psi \\ \boldsymbol{q} \end{bmatrix}$ | vector of quadrotor position, quadrotor yaw angle and manipulator rotation vector |
| $\dot{\boldsymbol{\kappa}} = \begin{bmatrix} \dot{\boldsymbol{\xi}} \\ \dot{\boldsymbol{\eta}} \end{bmatrix}$ | quadrotor linear velocity and Euler rate |
| $\dot{\boldsymbol{\kappa}}_T = \begin{bmatrix} \dot{\boldsymbol{\xi}}_T \\ \dot{\boldsymbol{\eta}}_T \end{bmatrix}$ | end effector linear velocity and Euler rate wrt inertial frame |

## A.7 Control Architecture

| | |
|---|---|
| $\boldsymbol{\xi}_{T.d}$ | desired end effector position vector wrt inertial frame |
| $\dot{\boldsymbol{\xi}}_{T.d}$ | desired end effector linear velocity wrt body frame |
| $\boldsymbol{\eta}_{T.d}$ | desired end effector rotation vector wrt inertial frame |
| $\dot{\boldsymbol{\eta}}_{T.d}$ | desired end effector Euler angle rate wrt inertial frame |
| $\boldsymbol{\xi}_r$ | quadrotor reference position vector wrt inertial frame |
| $\dot{\boldsymbol{\xi}}_r$ | quadrotor reference position linear vlocity wrt inertial frame |
| $\psi_r$ | quadrotor reference yaw rotation vector wrt inertial frame |
| $\dot{\psi}_r$ | quadrotor reference yaw Euler angle rate wrt inertial frame |
| $\boldsymbol{q}_r$ | manipulator reference vector of joint positions |
| $\dot{\boldsymbol{q}}_r$ | manipulator reference vector of joint velocities |
| $\boldsymbol{U}$ | quadrotor input forces or input torques |
| $\boldsymbol{\Gamma}$ | manipulator arm vector of joint input forces or input torques |
| $\boldsymbol{\xi}$ | quadrotor position vector |
| $\boldsymbol{\eta}$ | quadrotor rotation vector |
| $\boldsymbol{q}$ | manipulator joint position vector |

# BIBLIOGRAPHY


[1] E. Robotics, "Strategic research agenda for robotics in europe 2014–2020," *IEEE Robot. Autom. Mag*, vol. 24, p. 171, 2014.

[2] A. Ollero and B. Siciliano, "Introduction," in *Aerial Robotic Manipulation Research, Development and Applications*, A. Ollero and B. Siciliano, Eds. Cham, Switzerland: Springer Nature, 2019, ch. 1, pp. 3–11.

[3] F. Ruggiero, V. Lippiello, and A. Ollero, "Aerial manipulation: A literature review," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1957–1964, 2018.

[4] D. Mellinger, Q. Lindsey, M. Shomin, and V. Kumar, "Design, modeling, estimation and control for aerial grasping and manipulation," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011, pp. 2668–2673.

[5] S. Kim, S. Choi, and H. S. Kim, "Aerial manipulation using a quadrotor with a two dof robotic arm," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013, pp. 4990–4995.

[6] G. Heredia, A. E. Jimenez-Cano, I. Sanchez, D. Llorente, V. Vega, J. Braga, J. A. Acosta, and A. Ollero, "Control of a multirotor outdoor aerial manipulator," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2014, pp. 3417–3422.

[7] G. Heredia, R. Cano, A. Jimenez Cano, and A. Ollero, "Modeling and design of multirotors with multi-joint arms," in *Aerial Robotic Manipulation Research, Development and Applications*, A. Ollero and B. Siciliano, Eds. Cham, Switzerland: Springer Nature, 2019, ch. 2, pp. 15–33.

[8] X. Ding, P. Guo, K. Xu, and Y. Yu, "A review of aerial manipulation of small-scale rotorcraft unmanned robotic systems," *Chinese Journal of Aeronautics*, vol. 32, no. 1, pp. 200 – 214, 2019.

[9] F. Forte, R. Naldi, A. Macchelli, and L. Marconi, "Impedance control of an aerial manipulator," in *2012 American Control Conference (ACC)*, 2012, pp. 3839–3844.

[10] B. Yang, Y. He, J. Han, and G. Liu, "Rotor-flying manipulator: Modeling, analysis, and control," *Mathematical Problems in Engineering*, vol. 2014, May 2014.

[11] P. E. I. Pounds, D. R. Bersak, and A. M. Dollar, "Grasping from the air: Hovering capture and load stability," in *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 2491–2498.

[12] S. Kannan, M. A. Olivares-Mendez, and H. Voos, "Modeling and control of aerial manipulation vehicle with visual sensor," *IFAC Proceedings Volumes*, vol. 46, no. 30, pp. 303 – 309, 2013.

[13] S. Kannan, M. Alma, M. A. Olivares-Mendez, and H. Voos, "Adaptive control of aerial manipulation vehicle," in *2014 IEEE International Conference on Control System, Computing and Engineering (ICCSCE 2014)*, 2014, pp. 273–278.

[14] Y. Stergiopoulos, E. Kontouras, K. Gkountas, K. Giannousakis, and A. Tzes, "Modeling and control aspects of a uav with an attached manipulator," in *2016 24th Mediterranean Conference on Control and Automation (MED)*, June 2016, pp. 653–658.

[15] K. Gkountas, D. Chaikalis, and A. Tzes, "Force control design for a robot manipulator attached to a uav," *IFAC-PapersOnLine*, vol. 51, no. 30, pp. 548–553, 2018.

[16] V. Lippiello and F. Ruggiero, "Cartesian impedance control of a uav with a robotic arm," *IFAC Proceedings Volumes*, vol. 45, no. 22, pp. 704 – 709, 2012.

[17] A. E. Jimenez-Cano, J. Martin, G. Heredia, A. Ollero, and R. Cano, "Control of an aerial robot with multi-link arm for assembly tasks," in *2013 IEEE International Conference on Robotics and Automation*, 2013, pp. 4916–4921.

[18] G. Arleo, F. Caccavale, G. Muscio, and F. Pierri, "Control of quadrotor aerial vehicles equipped with a robotic arm," in *21St mediterranean conference on control and automation*. IEEE, 2013, pp. 1174–1180.

[19] F. Caccavale, G. Giglio, G. Muscio, and F. Pierri, "Adaptive control for uavs equipped with a robotic arm," *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 11 049 – 11 054, 2014.

[20] G. Antonelli and E. Cataldi, "Adaptive control of arm-equipped quadrotors. theory and simulations," in *22nd Mediterranean Conference on Control and Automation*, 2014, pp. 1446–1451.

[21] N. Mimmo, A. Macchelli, R. Naldi, and L. Marconi, "Robust motion control of aerial manipulators," *Annual Reviews in Control*, vol. 49, pp. 230–238, 2020.

[22] N. Bulut, A. Turgut, and K. Aríkan, "Computed torque control of an aerial manipulation system with a quadrotor and a 2-dof robotic arm," 2019.

[23] C. Zhiyuan, L. Yanyang, S. Yanhua, C. Hongyu, W. Bin, H. Mingqi, and Y. Rao, "Fuzzy sliding mode control for rotorcraft aerial manipulator with extended state observer," in *2020 Chinese Automation Congress (CAC)*. IEEE, 2020, pp. 1710–1714.

[24] Y. Chen, W. Zhan, B. He, L. Lin, Z. Miao, X. Yuan, and Y. Wang, "Robust control for unmanned aerial manipulator under disturbances," *Ieee Access*, vol. 8, pp. 129 869–129 877, 2020.

[25] H. Lee and H. J. Kim, "Estimation, control, and planning for autonomous aerial transportation," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 4, pp. 3369–3379, 2016.

[26] R. Jiao, W. Chou, Y. Rong, and M. Dong, "Anti-disturbance control for quadrotor uav manipulator attitude system based on fuzzy adaptive saturation super-twisting sliding mode observer," *Applied Sciences*, vol. 10, no. 11, p. 3719, 2020.

[27] Z. Samadikhoshkho, S. Ghorbani, F. Janabi-Sharifi, and K. Zareinia, "Nonlinear control of aerial manipulation systems," *Aerospace Science and Technology*, vol. 104, p. 105945, 2020.

[28] C. Kuchwa-Dube and J. O. Pedro, "Altitude and attitude tracking of a quadrotor-based aerial manipulator using super twisting sliding mode control," in *Proceedings of the 6th International Conference on Control, Mechatronics and Automation*, 2018, pp. 65–69.

[29] S. Rajappa, C. Masone, H. H. Bülthoff, and P. Stegagno, "Adaptive super twisting controller for a quadrotor uav," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 2971–2977.

[30] R. Akbar and N. Uchiyama, "Adaptive modified super-twisting control for a quadrotor helicopter with a nonlinear sliding surface," in *2017 SICE International Symposium on Control Systems (SICE ISCS)*. IEEE, 2017, pp. 1–6.

[31] H. Castañeda and J. L. Gordillo, "Spatial modeling and robust flight control based on adaptive sliding mode approach for a quadrotor mav," *Journal of Intelligent & Robotic Systems*, vol. 93, pp. 101–111, 2019.

[32] Y. Shtessel, M. Taleb, and F. Plestan, "A novel adaptive-gain supertwisting sliding mode controller: Methodology and application," *Automatica*, vol. 48, no. 5, pp. 759–769, 2012.

[33] C. Kuchwa-Dube and J. O. Pedro, "Quadrotor-based aerial manipulator altitude and attitude tracking using adaptive super-twisting sliding mode control," in *2019 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2019, pp. 144–151.

[34] M. Tognon, "Theory and applications for control and motion planning of aerial robots in physical interaction with particular focus on tethered aerial vehicles," Ph.D. dissertation, Institut National des Sciences Appliquées de Toulouse, Universite de Toulouse, France, July 2019.

[35] W. Khalil and E. Dombre, *Modeling, identification and control of robots.* Butterworth-Heinemann, 2004.

[36] V. Lippiello and F. Ruggiero, "Exploiting redundancy in cartesian impedance control of uavs equipped with a robotic arm," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 3768–3773.

[37] W. Khalil, "Dynamic modeling of robots using recursive newton-euler techniques," in *ICINCO2010*, 2010.

[38] H. T. M. N. ElKholy, "Dynamic modeling and control of a quadrotor using linear and nonlinear approaches," Master's thesis, School of Sciences and Engineering, The American University in Cairo, April 2014.

[39] J. Diebel, "Representing attitude: Euler angles, unit quaternions, and rotation vectors," *Matrix*, vol. 58, no. 15-16, pp. 1–35, 2006.

[40] F. Sabatino, "Quadrotor control: Modeling, nonlinearcontrol design, and simulation," Master's thesis, KTH Electrical Engineering, Stockholm, Sweden, June 2015.

[41] A. N. B. Hussien, "Autopilot design for a quadcopter," Master's thesis, Department of Electrical and Electronics Engineering, University Of Khartoum, Sudan, October 2017.

[42] D. W. Mellinger, "Trajectory generation and control for quadrotors," Ph.D. dissertation, Faculties of the University of Pennsylvania, University of Pennsylvania, Jan 2012.

[43] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics: modelling, planning and control.* Springer Science & Business Media, 2010.

[44] E. Cataldi, G. Muscio, M. A. Trujillo, Y. Rodríguez, F. Pierri, G. Antonelli, F. Caccavale, A. Viguria, S. Chiaverini, and A. Ollero, "Impedance control of an aerial-manipulator: Preliminary results," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).* IEEE, 2016, pp. 3848–3853.

[45] V. I. Utkin, *Sliding modes in control and optimization.* Springer Science & Business Media, 2013.

[46] Y. Shtessel, C. Edwards, L. Fridman, and A. Levant, *Sliding mode control and observation.* Springer, 2014, vol. 10.

[47] E. Tahoumi, "New robust control schemes linking linear and sliding mode approaches," Ph.D. dissertation, École centrale de Nantes, 2019.

[48] S. Bouabdallah and R. Siegwart, "Backstepping and sliding-mode techniques applied to an indoor micro quadrotor," in *Proceedings of the 2005 IEEE international conference on robotics and automation.* IEEE, 2005, pp. 2247–2252.

[49] F. Plestan, Y. Shtessel, V. Bregeault, and A. Poznyak, "New methodologies for adaptive sliding mode control," *International journal of control*, vol. 83, no. 9, pp. 1907–1919, 2010.

[50] J. Burton and A. S. Zinober, "Continuous approximation of variable structure control," *International journal of systems science*, vol. 17, no. 6, pp. 875–885, 1986.

[51] A. Levant, "Sliding order and sliding accuracy in sliding mode control," *International journal of control*, vol. 58, no. 6, pp. 1247–1263, 1993.

[52] M. Taleb and F. Plestan, "Adaptive supertwisting controller with reduced set of parameters," in *2021 European Control Conference (ECC).* IEEE, 2021, pp. 2627–2632.

[53] J. A. Moreno, "On strict lyapunov functions for some non-homogeneous super-twisting algorithms," *Journal of the Franklin Institute*, vol. 351, no. 4, pp. 1902–1919, 2014.

[54] P. R. Kumar, A. K. Behera, and B. Bandyopadhyay, "Robust finite-time tracking of stewart platform: A super-twisting like observer-based forward kinematics solution," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 5, pp. 3776–3785, 2017.

[55] A. Chalanga and F. Plestan, "Finite time stabilization of an uncertain chain of integrators by integral sliding mode approach," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 9613–9618, 2017.

[56] B. Siciliano and O. Khatib, *Springer handbook of robotics 2nd Edition.* springer, 2016.

[57] B. Siciliano, L. Sciavicco, S. Chiaverini, P. Chiacchio, L. Villani, and F. Caccavale, "Jacobian-based algorithms: A bridge between kinematics and control," in *Proceedings of the Special Celebratory Symposium In the honor of Professor Bernie Roth's 70th Birthday.* Citeseer, 2003, pp. 4–35.

[58] S. R. Buss, "Introduction to inverse kinematics with jacobian transpose, pseudoinverse and damped least squares methods," *IEEE Journal of Robotics and Automation*, vol. 17, no. 1-19, p. 16, 2004.

**Titre :** Commande basée modes glissants d'un manipulateur aérien autonome

**Mot clés :** Manipulateur aérien autonome, commande par modes glissants, supertwisting, gain adaptatif

**Résumé :** Ce manuscrit présente une étude sur l'application de commandes par modes glissants adaptatives (premier ordre et supertwisting) à un système robotique. Un manipulateur aérien composé d'un quadrirotor équipé d'un bras manipulateur rigide à deux degrés de liberté est considéré, le travail s'attachant au suivi de trajectoires complexes, en agissant sur l'altitude et l'attitude du quadrotor et le positionnement du bras manipulateur. Un simulateur complet a été développé.
L'enjeu a été d'évaluer la capacité du système à suivre des trajectoires de mouvements hélicoïdaux en présence de perturbations (forces externes) ou d'incertitudes (masse embarquée). Pour cela, des lois de commande basées sur la théorie des modes glissants ont été proposées : la nature robuste de ce type d'approches, couplée à leur caractère adaptatif, a motivé leur usage pour la première fois dans ce cadre applicatif.
Une analyse détaillée des performances de ces nouvelles approches de commande a été faite en simulation, ainsi qu'une comparaison avec des approches plus classiques, notamment avec/sans présence de perturbations externes et avec/sans présence d'un chargement de masse ponctuelle à l'extrémité du bras du manipulateur.

**Title:** Sliding mode based control of an unmanned aerial manipulator

**Keywords:** Unmanned aerial manipulator, sliding mode control, supertwisting, adaptive gain.

**Abstract:** This manuscript presents a study on the application of adaptive sliding mode control (first-order and supertwisting) to a robotic system. An aerial manipulator composed of a quadrotor equipped with a rigid two-degree-of-freedom manipulator arm is considered, the work focusing on the tracking of complex trajectories, by acting on the altitude and attitude of the quadrotor and the positioning of the manipulator arm. A complete simulator has been developed.
The challenge was to assess the system's ability to follow helical motion trajectories in the presence of disturbances (external forces) or uncertainties (onboard mass). To achieve this, control laws based on the theory of sliding modes were proposed: the robust nature of this type of approach, coupled with their adaptive feature, has motivated their use for the first time in this field of application.
A detailed analysis of the performance of these new control approaches has been carried out in simulation, along with a comparison with more conventional approaches, in particular with/without the presence of external disturbances and with/without the presence of point mass loading at the end of the manipulator arm.