RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE

ÉCOLE NATIONALE POLYTECHNIQUE
Département d'Automatique

Final Year Project Report

For the attainment of an Engineer degree in Control Engineering

---

Fractional-Order Control of a Helicopter Flight Simulator (TRMS): Simulation and Experimentation

---

ZERROUGUI Yahya Moundher & DEBACHE Mounsef

Supervised by **Pr. BOUKHETALA Djamel** and **Dr. ELBEY Abdeldjalil**, ENP

Publicly defended on 21/06/2025

Jury members:

President:	Pr.BOUDJEMAA Fares	ENP
Supervisors:	Pr.BOUKHETALA Djamel & Dr.ELBEY Abdeldjalil	ENP
Examiner:	Pr.BOUDANA Djamel	ENP

ENP 2025

Département d'Automatique / ENP, 10 Rue des Frères OUDEK, El Harrach 16200 Algiers, Algeria.

RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE

المدرسة الوطنية المتعددة التقنيات
Ecole Nationale Polytechnique

مخبر التحكم في العمليات التصنيعية
Laboratoire de Commande des Processus

ÉCOLE NATIONALE POLYTECHNIQUE
Département d'Automatique

Final Year Project Report

For the attainment of an Engineer degree in Control Engineering

---

Fractional-Order Control of a Helicopter Flight Simulator (TRMS): Simulation and Experimentation

---

ZERROUGUI Yahya Moundher & DEBACHE Mounsef

Supervised by **Pr. BOUKHETALA Djamel** and **Dr. ELBEY Abdeldjalil**, ENP

Publicly defended on 21/06/2025

Jury members:

President:	Pr.BOUDJEMAA Fares	ENP
Supervisors:	Pr.BOUKHETALA Djamel & Dr.ELBEY Abdeldjalil	ENP
Examiner:	Pr.BOUDANA Djamel	ENP

ENP 2025

Département d'Automatique / ENP, 10 Rue des Frères OUDEK, El Harrach 16200 Algiers, Algeria.

RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE

ÉCOLE NATIONALE POLYTECHNIQUE
Département d'Automatique

Mémoire de projet de fin d'études

En vue de l'obtention du diplôme d'Ingénieur d'état en Automatique

---

Commandes d'Ordre Fractionnaire d'un Simulateur de Vol d'Hélicoptère (TRMS) : Simulation et Expérimentation

---

ZERROUGUI Yahya Moundher & DEBACHE Mounsef

Sous la direction de **Pr. BOUKHETALA Djamel** et **Dr. ELBEY Abdeldjalil**, ENP

Présenté et soutenu publiquement le 21/06/2025

Composition du jury:

Président:	Pr.BOUDJEMAA Fares	ENP
Promoteurs:	Pr.BOUKHETALA Djamel & Dr.ELBEY Abdeldjalil	ENP
Examinateur:	Pr.BOUDANA Djamel	ENP

ENP 2025

Département d'Automatique / ENP, 10 Rue des Frères OUDEK, El Harrach 16200 Alger Algérie.

الملخص:

يستعرض هذا البحث استخدام الحساب الكسري في التحكم بنظام Twin Rotor MIMO System. بعد نمذجة مفصلة لهذا النظام وتقديم لمحة عن أساسيات الحساب الكسري، تم تصميم منظمات PID التقليدية والمنظمات الكسرية (FOPID)، بالإضافة إلى تقنيات التحكم بالانزلاق (سواء التقليدية أو الكسرية).تم تحسين معلمات هذه المنظمات باستخدام خوارزمية التحسين بسرب الجسيمات (PSO). وقد أظهرت نتائج المحاكاة أن تقنيات التحكم الكسري توفر أداءً أفضل مقارنة بالتحكمات التقليدية، خاصة من حيث المتانة وقدرة النظام على رفض الاضطرابات.

الكلمات المفتاحية : الحساب الكسري، نظام TRMS، منظم PID، منظم FOPID، التحكم بالانزلاق، التحكم الكسري، خوارزمية سرب الجسيمات (PSO).

Résumé

Ce mémoire explore l’utilisation du calcul fractionnaire pour la commande du système Twin Rotor MIMO (TRMS). Après une modélisation détaillée du système et un rappel des bases du calcul fractionnaire, des régulateurs PID classiques et fractionnaires (FOPID), ainsi que des commandes par mode de glissement (classique et fractionnaire), sont conçus. Leurs paramètres sont optimisés via l’algorithme PSO. Les simulations montrent que les commandes fractionnaires offrent de meilleures performances que les approches classiques, en particulier en termes de robustesse et de rejet des perturbations.

Mots-clés : calcul fractionnaire, TRMS, régulateur PID, FOPID, commande par mode de glissement, commande fractionnaire, optimisation par essaim particulaire (PSO).

Abstract

This project focuses on designing and optimizing fractional-order controllers for the Twin Rotor MIMO System (TRMS), a nonlinear and strongly coupled platform. Two strategies—FOPID and FOSMC—are developed and tuned using Particle Swarm Optimization (PSO). Simulation results show that fractional-order controllers offer superior performance over classical methods in terms of robustness, disturbance rejection, and handling system nonlinearities.

Keywords: fractional-order calculus, Twin Rotor MIMO System (TRMS), PID controller, FOPID, sliding mode control, fractional-order sliding mode control, Particle Swarm Optimization (PSO).

Dedication

I dedicate this work to my mother, for her unconditional love, patience, and unwavering support. You have been my greatest source of inspiration, and I am deeply grateful for all your sacrifices.
To my father, who has been a model of determination and perseverance. Your sacrifices, unconditional love, and constant support have been the foundation of my success.

To my dear sister Ramla, the pillar of my life and one of my greatest allies.You are unique and precious in your own way. May our love and family bond last forever.

To my beloved aunts Hassina, Wahiba, and Soumeya, whose kindness, wisdom, and heartfelt encouragement have always meant the world to me. To my cousins Imad and Nounou, thank you for the joy, the laughter, and the unforgettable moments we've shared.You each hold a special place in my heart.

To my teammate Mounsef, we have come a long way together.Thank you for sharing this unique journey with me.This thesis is the result of our strong partnership and hard work.

To my dear friends, especially Badro, Youcef, Minou, Haythem, Raid, and Ala,your friendship has been a source of strength, joy, and unwavering support.I am truly honored to have walked this path with you. May our paths continue to cross and our friendship flourish.

I am deeply honored to have you all by my side. Thank you.

Yahya

Dedication

I dedicate this work to my mother, whose unconditional love, patience, and unwavering support have been my greatest source of strength. Your sacrifices and boundless encouragement continue to inspire me every day.

To my father, a true role model of perseverance and determination. Your love and tireless support have laid the foundation for all my achievements.

To my dear brothers Oussama and Aymen, and to my beloved sisters Soumia, Fatima, Hiba, and Wissal you are the pillars of my life and my most trusted allies. Each of you holds a special and irreplaceable place in my heart. May our bond remain strong and enduring.

To the memory of my grandfather baba hassen (Allah yarhmou) whose values and wisdom continue to guide me even in his absence.

To my teammate and friend Yahya we have journeyed far together. I am grateful for your dedication, support, and the commitment we shared throughout this work. This thesis is the fruit of our collaboration and shared effort.

To my cherished friends , particularly Yousri, Joewii,Lakche,Assil,Ayoub, Souheib, Ramy, Zaza, Haithem and Dash your friendship has been a source of strength, laughter, and unwavering support. I am honored to have shared this path with you.

To my team `.csv` *thank you for the energy, creativity, and teamwork that made our experience so memorable.*

Thank you all, from the bottom of my heart.

Mounsef

Acknowledgements

First and foremost, we thank Allah the Almighty for granting us the strength and patience to carry out and complete this work.

*We would like to express our sincere gratitude to our supervisors, **Mr. Djamel Boukhetala** and **Mr. Abdeldjalil ElBey**, for their invaluable guidance and support throughout our journey. Their expertise and dedication were instrumental to our success, and we are deeply grateful for their contribution.*

We extend our heartfelt thanks to the jury members for accepting, with honor, to evaluate and assess our work.

We also wish to express our deep gratitude to all the professors who have accompanied us throughout our academic path.

Their commitment and expertise have played a vital role in shaping our education.

Our thanks also go to the academic and administrative staff of the ENP for their continuous efforts in providing us with a high-quality education.Finally, we wish to thank all those who, directly or indirectly, contributed to the completion of this work.

Yahya , Mounsef

Contents

List of Tables

List of Figures

Acronyms

- **PID** : Proportional Integral Derivative
- **FOPID** : Fractional Order PID
- **TRMS** : Twin Rotor MIMO System
- **MIMO** : Multi-Input Multi-Output
- **LQR** : Linear Quadratic Regulator
- **SMC** : Sliding Mode Control
- **FOSMC** : Fractional Order Sliding Mode Control
- **SISO** : Single-Input Single-Output
- **TITO** : Two-Input Two-Output
- **PSO** : Particle Swarm Optimization
- **ACO** : Ant Colony Optimization
- **GA** : Genetic Algorithm
- **LSTM** : Long Short-Term Memory
- **DC** : Direct Current
- **PWM** : Pulse Width Modulation
- **LTI** : Linear Time-Invariant
- **G-L** : Grünwald-Leitnikov Definition (of fractional derivative)
- **R-L** : Riemann–Liouville Definition (of fractional derivative)
- **ITAE** : Integral of Time-weighted Absolute Error
- **RMSE** : Root Mean Square Error
- **MAE** : Mean Absolute Error
- **IAE** : Integral of Absolute Error
- **ISE** : Integral of Squared Error

General introduction

Motivation

Technological progress, especially in the aeronautics sector, has accelerated the development of advanced control laws. While many control strategies have proven effective on aircraft and helicopters, increasing system complexity often leads to intricate, hard-to-implement controllers. To address this, simulators like the Twin Rotor MIMO System (TRMS) are widely used as experimental platforms.

The TRMS, a nonlinear, coupled, and unstable multivariable system, offers a realistic yet safe environment for testing advanced control techniques before their application to real helicopters.

State of the Art

In recent decades, the growing complexity and performance requirements in industry have driven major advances in automatic control. These improvements were made possible by technological progress especially the emergence of microprocessors and by developments in mathematical tools such as numerical analysis and systems theory.

Among the promising tools in modern control theory is *fractional calculus*, which generalizes classical differentiation to non-integer orders. Initially a theoretical curiosity, it gained practical traction in the 1990s with Oustaloup's CRONE controller and Podlubny's $\mathrm{PI}^{\lambda}\mathrm{D}^{\mu}$ regulator, both exploiting the memory and robustness properties of fractional operators.

Since then, fractional-order control has proven its relevance in various fields, including process control, robotics, and system identification, offering enhanced flexibility and robustness compared to traditional methods.

Several scientific studies have confirmed the benefits of fractional-order systems and contributed to their theoretical and practical development [2, 3, 4, 5]. These references provide a strong foundation for further research in this area.

Objective of this Thesis

Our main objective in this thesis is to develop and validate fractional-order control strategies for the Twin Rotor MIMO System (TRMS), demonstrating how non-integer operators can improve performance compared to classical integer-order approaches. Specifically, we aim to:

- Introduce fractional-order operators into TRMS control laws and quantify their benefits (robustness, disturbance rejection, tuning flexibility) relative to integer-order designs.

- Optimize both PID and fractional PID (FOPID) controllers using Particle Swarm Optimization (PSO) for the decoupled TRMS model, then assess performance degradation if any when applied to the fully coupled dynamics.
- Extend sliding mode control (SMC) to a fractional-order sliding mode controller (FOSMC) for the TRMS, and compare its ability to mitigate coupling effects and reduce chattering versus classical SMC.
- Implement and validate both integer-order and fractional-order controllers on an experimental TRMS.

Thesis Overview

This thesis investigates the application of fractional-order control strategies to the Twin Rotor MIMO System (TRMS), a benchmark platform that mimics helicopter flight dynamics.

Chapter 1 presents the full modeling of the TRMS, including its physical components, and the nonlinear dynamic equations governing the pitch and yaw axes. An open-loop stability analysis reveals the system's intrinsic coupling and instability.

Chapter 2 introduces fractional calculus, detailing the Riemann–Liouville and Caputo definitions, along with their distinctive properties (memory effect, nonlocal behavior). Numerical approximation methods such as Oustaloup's recursive filter are also discussed in the context of control systems.

Chapter 3 focuses on designing classical PID and fractional-order PID (FOPID) controllers, optimized using the Particle Swarm Optimization (PSO) algorithm. The performance of both controllers is compared on decoupled and coupled TRMS models. Results highlight FOPID's superior robustness.

Chapter 4 develops sliding mode control (SMC) and its fractional counterpart (FOSMC). After designing both on the decoupled model* Simulations show that FOSMC significantly reduces chattering and better handles the coupling effects.

Chapter 5 presents the experimental validation of PID and FOPID controllers on a physical TRMS bench. Despite practical constraints (sensor noise, actuator saturation, model uncertainties), the FOPID controller maintains improved performance.

The **general conclusion** emphasizes the benefits of fractional-order controllers (FOPID and FOSMC) in terms of robustness, disturbance rejection, and coupling management. Future work may include online adaptive fractional control strategies.

Chapter 1

Principles and Mathematical Modeling of the Twin Rotor MIMO System

1.1 Introduction

The field of aerospace control presents significant challenges due to the high complexity, strong nonlinearities, and safety-critical nature of flight systems. Real-world aircraft and rotorcraft operate under uncertain environmental conditions, experience strongly coupled dynamics, and require precise control to ensure stability and performance.

While the development of advanced control strategies is essential for aerospace applications, their direct implementation on real aircraft is often impractical and risky. High costs, safety concerns, and the complexity of real-world systems necessitate the use of experimental testbeds and simulation environments to evaluate control strategies before real-world deployment.

To address these challenges, experimental platforms such as the Twin Rotor MIMO System have been developed. The TRMS serves as a laboratory-scale model, designed to mimic the dynamics of a helicopter, providing a controlled environment for testing and validating advanced control techniques. Its strongly coupled pitch and yaw dynamics make it an ideal benchmark for studying multi-input multi-output (MIMO) control strategies.

A comprehensive understanding of the TRMS's behavior requires the development of a mathematical model that accurately describes the relationships between its inputs and outputs. Such a model can be expressed in the form of differential equations, state-space representations, or transfer functions to analyze the system's dynamics. However, achieving an accurate analytical model is often challenging due to complex physical phenomena such as aerodynamics, friction, and coupling effects. Alternatively, a system identification approach can be used, where the model is derived from experimental data to capture the system's behavior.

This chapter begins with an introduction to helicopter flight principles and a presentation of the TRMS. We then provide a detailed modeling of the system, covering both coupled and decoupled models, before concluding with open-loop simulations to validate the derived equations.

1.2 Helicopter Flight Principle

Helicopters achieve flight through the principles of aerodynamics, utilizing rotating blades to generate lift and maneuver in multiple directions. Unlike fixed-wing aircraft, helicopters can hover, take off, and land vertically, as well as move forward, backward, and sideways. Additionally, helicopters possess six degrees of freedom, allowing them to move in three translational axes (forward/backward, left/right, up/down) and three rotational axes (pitch, roll, yaw). This unique capability is enabled by two key components: the main rotor, which provides lift and thrust, and the tail rotor, which counters torque and maintains stability.

Figure 1.1: Single Main Rotor Helicopter[1]

1.2.1 Forces Acting on the Helicopter

Once a helicopter leaves the ground, it is acted upon by four aerodynamic forces; thrust, drag, lift, and weight. Understanding how these forces work and knowing how to control them with the use of power and flight controls are essential to flight.They are defined as follows:

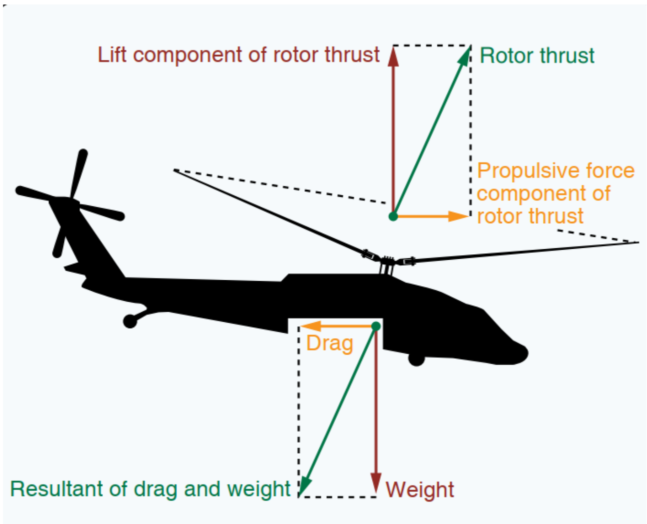


Figure 1.2: Four forces acting on a helicopter in forward flight[1]

- **Lift**: opposes the downward force of weight, is produced by the dynamic effect of the air acting on the airfoil and acts perpendicular to the flightpath through the center of lift.

- **Weight**: the combined load of the aircraft itself, the crew, the fuel, and the cargo or baggage. Weight pulls the aircraft downward because of the force of gravity. It opposes lift and acts vertically downward through the aircraft's center of gravity (CG).

- **Thrust**: the force produced by the power plant/ propeller or rotor. It opposes or overcomes the force of drag. As a general rule, it acts parallel to the longitudinal axis.

- **Drag**: a rearward, retarding force caused by disruption of airflow by the rotor, fuselage, and other protruding objects. Drag opposes thrust and acts rearward parallel to the relative wind.

1.2.2 The main rotor

The main rotor is the primary component responsible for generating lift and controlling the helicopter's movement. It consists of rotating blades with an asymmetrical airfoil shape, which creates aerodynamic lift based on Bernoulli's principle. As the blades rotate, air moves faster over the upper surface (extrados) than the lower surface (intrados), generating a pressure difference that results in an upward lifting force. The amount of lift depends on two key factors the rotor speed (RPM) and the blade angle of attack. The collective pitch control, operated by a lever, adjusts the angle of attack of all blades simultaneously increasing blade pitch results in greater lift and ascent, while decreasing blade pitch reduces lift and causes descent. Additionally, the cyclic pitch control, managed through a joystick-like stick, changes the pitch of individual blades as they rotate, allowing the pilot to tilt the rotor disk in different directions. For instance, pushing the cyclic forward increases the pitch of the blades at the rear while decreasing it at the front, causing the helicopter to move forward. These inputs are transmitted through the swashplate mechanism, which uses tilting rings and control rods to dynamically adjust blade angles. In essence, the main rotor provides both lift and thrust, enabling the helicopter to hover, climb, descend, and move in any direction with precise control.

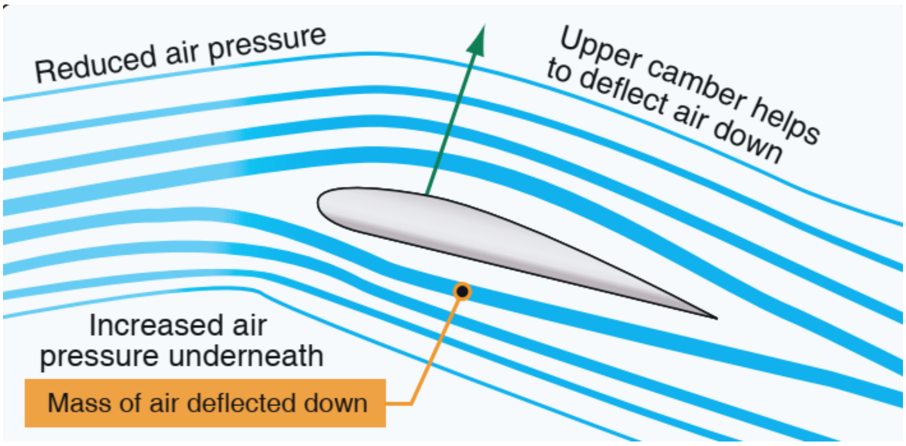


Figure 1.3: An illustration of aerodynamic lift generation, showing the pressure difference and airflow around a rotor blade[1]

1.2.3 The tail rotor

The tail rotor is essential for maintaining stability and controlling the helicopter's yaw (rotation around the vertical axis). According to Newton's Third Law, the rotation of the main rotor generates an opposite reactionary torque that would cause the fuselage to spin uncontrollably without compensation. To counteract this, the tail rotor produces lateral thrust in the opposite direction of the main rotor's torque, stabilizing the aircraft. The pilot controls this thrust using anti-torque pedals, which adjust the blade pitch of the tail rotor. Pressing the right pedal increases thrust, causing the helicopter to rotate right, while pressing the left pedal decreases thrust, making the helicopter turn left. This system provides precise yaw control, allowing the pilot to maintain heading and execute turns smoothly. While the tail rotor does not physically tilt, its thrust variation is perceived as a directional force that influences the helicopter's movement. Ultimately, the tail rotor ensures stability and maneuverability, enabling controlled flight.

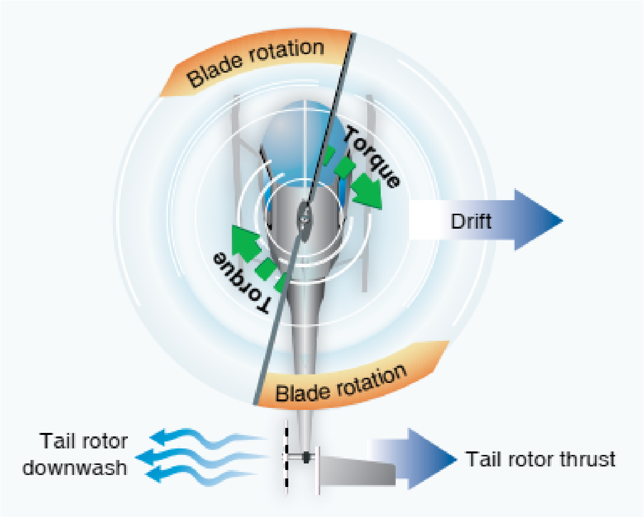


Figure 1.4: A tail rotor is designed to produce thrust in a direction opposite torque. The thrust produced by the tail rotor is sufficient to move the helicopter laterally.[1]

1.3 TRMS Description and Configuration

1.3.1 System description and schematics

The Twin Rotor MIMO System is an experimental platform designed by Feedback Instruments Ltd to facilitate laboratory-based research in control system engineering. It serves as a simplified representation of a helicopter's dynamic behavior, allowing for the safe testing and validation of advanced control algorithms before applying them to actual aerial vehicles.

While the TRMS replicates key aerodynamic and control characteristics of a helicopter, it incorporates several simplifications. Unlike a real helicopter, which maintains a nearly constant rotor speed while adjusting blade pitch to control movement, the TRMS operates by modulating rotor speeds to achieve changes in position and velocity. Additionally, the system is fixed to a central pivot, limiting its motion to pitch (vertical) and yaw (horizontal) rotations.

A key feature of the TRMS is its strong cross-coupling between the rotors, similar to real helicopters. Adjustments in the main rotor, which influences vertical movement, also induce changes in the yaw direction due to aerodynamic interactions. These complex MIMO (Multi-Input Multi-Output) dynamics make the TRMS an ideal testbed for control system experimentation, providing researchers with a platform to explore and refine nonlinear, adaptive, and robust control techniques[6].

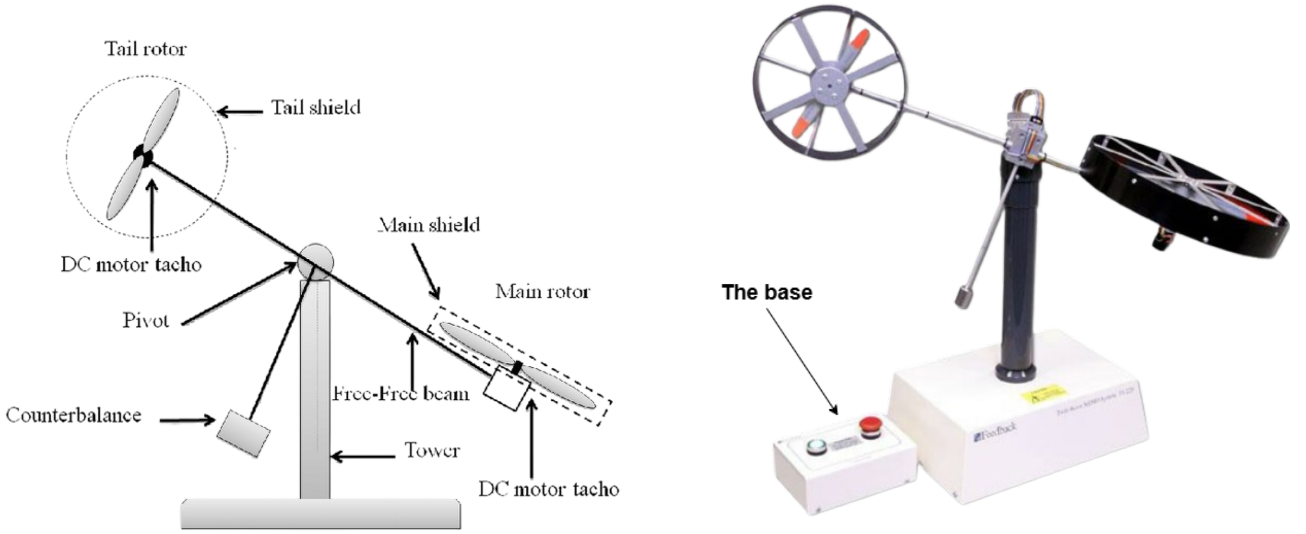


Figure 1.5: Scheme of TRMS' Components

The principle components of the TRMS are as represented in Figure 1.5:

- **The base:** It is the component on which the system rests. It contains electronic circuits for signal adaptation and filtering of both input and output signals.
- **The tower:** Supports the beam structure.
- **The Free-Free beam:** Can freely rotate on its base in both the horizontal and vertical planes.
- **Two DC motor(main and tail):** Attached to the ends of the beam, each consisting of a propeller, a DC rotor, and a safety shield.
- **The counterbalance:**Fixed at the center or at the end of the beam to reduce vibrations, it is also used to simulate different scenarios of weight.
- **The rotor on/off control box:**It allows initializing the TRMS or stopping it using the emergency stop button

Overview of System Dynamics:

The Twin Rotor MIMO System (TRMS) has two degrees of freedom, allowing it to move freely in both the horizontal and vertical planes. The horizontal motion (*azimuth*) ranges from $-2.82\,\text{rad}$ to $+2.82\,\text{rad}$, while the vertical motion (*elevation*) is limited to $-1.05\,\text{rad}$ to $+1.22\,\text{rad}$ like is represented on figure 1.6.

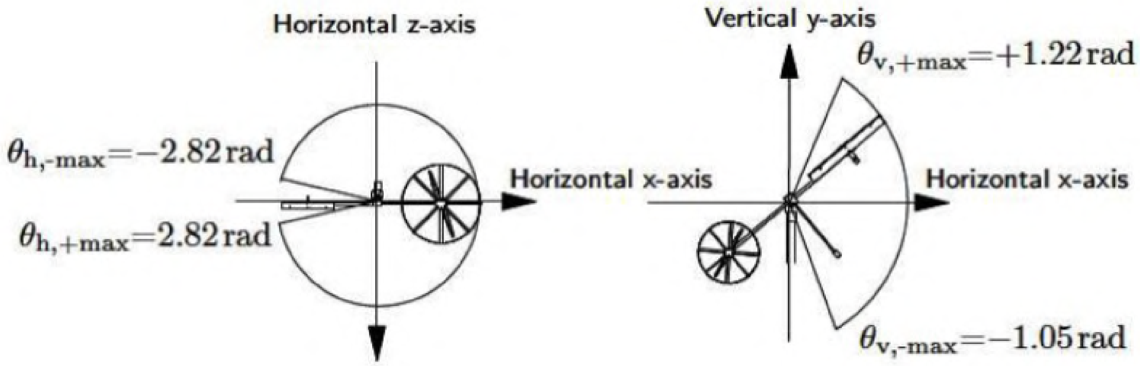


Figure 1.6: Range of Motion of the TRMS

If needed, the movement can be restricted to a single plane using locking screws, enabling experiments that focus solely on one-degree-of-freedom (1-DOF) control. This feature provides flexibility for researchers to study isolated dynamics before addressing the full MIMO complexity of the system.

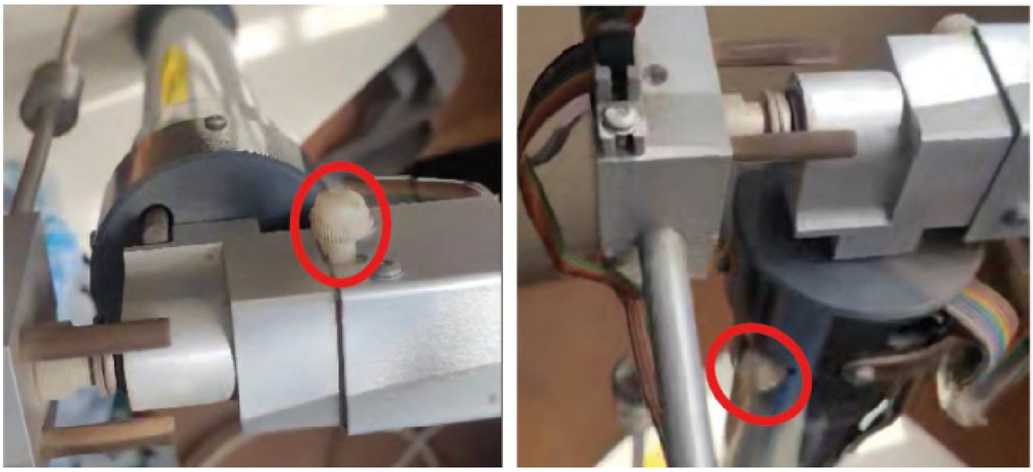


Figure 1.7: Vertical and Horizontal Screws

Since the TRMS operates in two degrees of freedom, it is essential to measure its key state variables to analyze and control its behavior effectively.

The measurable values of the TRMS (representing the system outputs) consist of:

- The horizontal angle α_h, called *azimuth*, which is primarily generated by the propulsive force of the tail rotor.
- The vertical angle α_v, called *elevation*, which is mainly produced by the propulsive force of the main rotor.

The spherical joint allows the beam to rotate simultaneously in both the horizontal and vertical planes, making it a two-degree-of-freedom (2-DOF) system.

Control of the rotors:

The system inputs (i.e., the control commands) are the voltage supplies to the rotors, u_v and u_h, which vary within a range of $[-2.5\,\mathrm{V}, +2.5\,\mathrm{V}]$.

To effectively control the motion of the TRMS, the system relies on DC rotors, which have the advantage that their rotation speed is proportional to the applied voltage. This allows for precise regulation of the pitch and yaw angles by adjusting the input voltage.

The rotor control mechanism is based on a chopper circuit, which is a static converter powered by a DC voltage source. This converter produces a variable average voltage across the rotor terminals, enabling smooth speed control. The chopper is driven by Pulse Width Modulation (PWM) signals, which are generated from the control signal sent by the PC via the data acquisition card (See Figure 1.8).

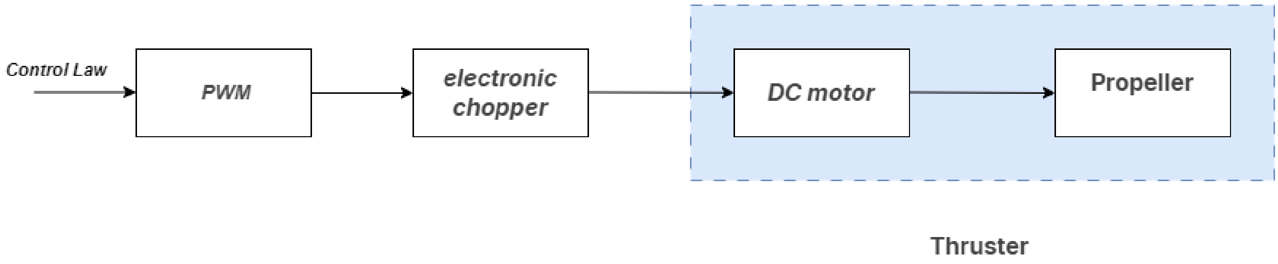


Figure 1.8: Control of the rotors

Measurement and Control Signal Acquisition:

The state of the beam is described by four process variables: the horizontal and vertical angles, measured using position sensors placed at the pivot point, and the two corresponding angular velocities. Tacho-generators measure the angular velocities of the rotors. Unlike a real helicopter, where aerodynamic force is controlled by adjusting the angle of attack, this model keeps the angle of attack fixed, and aerodynamic forces are regulated by varying rotor speeds. Adjustments in voltage levels lead to changes in the propellers' rotational speeds, subsequently altering the beam's position. Optical encoders determine the beam's angular position: two light sources (A and B) emit beams passing through two rings of slits on a disc. These slits are offset to produce rectangular wave signals with a specific phase difference at the output. The sign of this phase difference indicates the direction of rotation.1.9

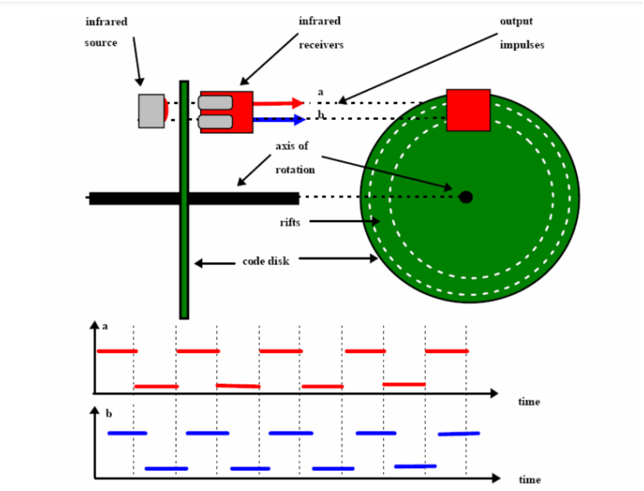


Figure 1.9: Position sensor

The control signal originates from the computer and passes through the D/A converter of the data acquisition board. The output of the D/A converter is connected to a power amplifier, which then drives the DC rotor. Both the power amplifier and encoder interface are housed within the TRMS unit. Additionally, this unit includes two switches: a main power switch and a separate switch to cut off power specifically to the DC rotor.

1.4 TRMS Modeling

The Twin Rotor MIMO System (TRMS) has been modeled using various methodologies in the literature. An analytical modeling approach using fundamental laws of physics, specifically Newton's second law and Euler–Lagrange equations, was applied in [7]. A hybrid modeling method, combining physics-based analytical modeling using Newton's equations and Genetic Algorithms (GA) for parameter estimation, was utilized in [8]. Furthermore, [7] employed a hybrid modeling approach, combining first-principles physics-based nonlinear equations derived using Newton's second law and refined through Genetic Algorithm optimization,Also An ant colony optimization-based approach was proposed for predicting the dynamic behavior of a twin rotor system, demonstrating the effectiveness of nature-inspired algorithms in system modeling [9].Finally, a data-driven approach utilizing a fuzzy inference system based on clustering and Differential Evolution for the identification of a twin rotor system was presented in [10]and A real-time observer based on a Chebyshev neural network was developed and implemented for the control of a twin rotor system, highlighting the potential of neural network-based observers in nonlinear system estimation [11].

In this work, the mathematical modeling of the Twin Rotor MIMO System follows the methodology proposed by [12]. This approach involves deriving nonlinear equations explicitly based on fundamental physical principles, capturing the dynamics associated with gravitational effects, aerodynamic forces, and rotor dynamics. Given that the TRMS is an underactuated nonlinear MIMO system, it requires careful consideration of its complex interactions. The nonlinear equations obtained through this approach are subsequently validated and calibrated through rigorous simulation and experimental testing, ensuring an accurate and reliable mathematical representation of the TRMS dynamics. Since the TRMS possesses two degrees of freedom (pitch and yaw rotations), these dynamics are modeled separately yet simultaneously, reflecting the intricate coupling between the two axes. The validation step through simulation and experiments ensures the robustness and precision of the resulting model.

1.4.1 Vertical subsystem

In this subsystem, the analysis focuses on rotation around the horizontal axis. Several forces act on this subsystem: primarily, the force generated by the rotor, responsible for most of the system dynamics, as well as gravitational forces due to the weight of the various components.Figure 1.10 illustrates these forces.

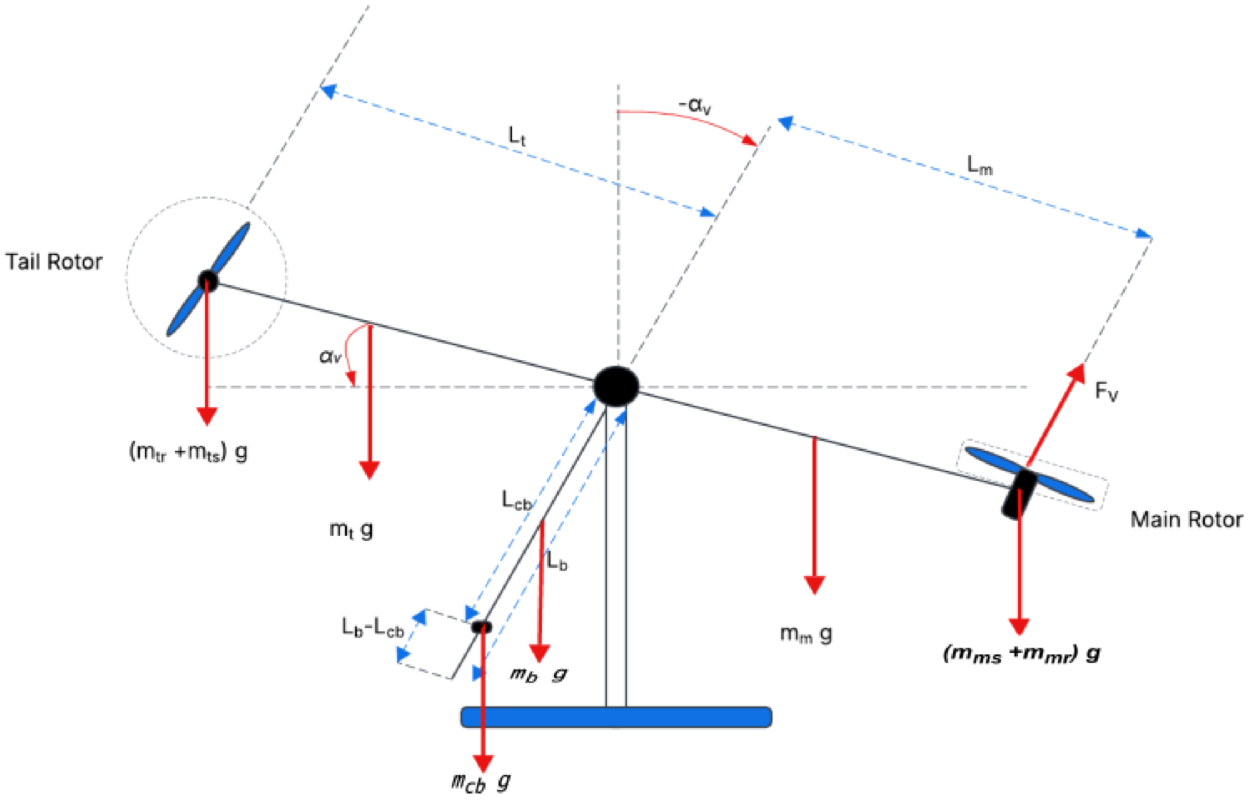


Figure 1.10: vertical subsystem.

The application of Newton's second law gives:

$$M_v = J_v \frac{d^2\alpha_v}{dt^2} \tag{1.1}$$

With:

$$\begin{cases} M_v = \sum_{i=1}^{4} M_{vi} \\ J_v = \sum_{i=1}^{8} J_{vi} \end{cases} \tag{1.2}$$

Where:

- M_v: Moment of the resultant force
- J_v: Sum of moments of inertia about the horizontal axis
- α_v: Vertical angle (angle of elevation)

As mentioned above, we have 4 moments of force, so we have 8 moments of inertia to calculate. We have 8 rigid bodies, so 8 moments. We will start by detailing the calculation of the moments of force and then move on to the calculation of the moments of inertia.

We therefore have the following moments of forces:

- Moment of the gravitational force
- Moment of the aerodynamic force
- Moment of the centrifugal force

- Moment of the friction force

Moment of the Gravitational Force M_{v1}:

Physically, the moment measures the tendency of a force to rotate an object around a point or an axis. Therefore, one can say that the moment is the applied force at the point, multiplied by the lever arm (distance).

$$\vec{M} = \vec{d} \wedge \vec{F}$$

By defining the applied force and measuring the distance between the point where the force is applied and the center of our reference frame, we can calculate the resulting moment. We will do this for the 8 bodies present (the two parts of the arm, the bar and the counterweight, the two rotors, and the two airscrew rotors). For example, for $\vec{F} = m_i g \hat{j}$:

$$\vec{M} = \vec{d} \wedge \vec{F} = \begin{pmatrix} -\frac{l}{2} \\ 0 \\ 0 \end{pmatrix} \wedge \begin{pmatrix} m_i g \sin(\alpha_v) \\ -m_i g \cos(\alpha_v) \\ 0 \end{pmatrix} = \frac{l}{2} m_i g \cos(\alpha_v)$$

We can therefore apply this definition to our system. The moment of the gravitational force is given by the sum of each gravitational moment. We can write:

$$M_{v1} = g\left[(A - B)\cos(\alpha_v) - C\sin(\alpha_v)\right] \tag{1.3}$$

With:

$$\begin{cases} A = \left(\frac{m_t}{2} + m_{tr} + m_{ts}\right) l_t \\ B = \left(\frac{m_m}{2} + m_{mr} + m_{ms}\right) l_m \\ C = \frac{m_b}{2} l_b + m_{cb} l_{cb} \end{cases} \tag{1.4}$$

Where:

m_{mr}: Mass of the main rotor.

m_m: Mass of the main part of the beam.

m_{tr}: Mass of the tail rotor.

m_t: Mass of the tail part of the beam.

m_{cb}: Mass of the counterweight.

m_b: Mass of the beam's counterweight.

m_{ms}: Mass of the main rotor airscrew.

m_{ts}: Mass of the tail rotor airscrew.

l_m: Length of the main part of the beam.

l_t: Length of the tail part of the beam.

l_b: Length of the counterweight beam.

l_{cb}: Distance between the counterweight and the joint.

g: Gravitational acceleration.

Moment of the Centrifugal Force M_{v2}:

When the beam performs a rotation in the horizontal plane, it creates an orthogonal force to the direction of rotation, called the centrifugal force. The expression of this force is given by the following relation:

$$\vec{F} = m_t \Omega_n^2 \vec{r_t} \tag{1.5}$$

$\Omega_h = \frac{d\alpha_h}{dt}$ being the angular velocity of the arm around the vertical axis.

We can therefore write the moment as:

$$\vec{M} = \vec{d} \wedge \vec{F}$$

We can thus write the moment of the centrifugal force as follows:

$$\begin{aligned} M_{v2} &= -\Omega_h^2 (Al_t + Bl_m + c^*) \sin(\alpha_v) \cos(\alpha_v) \\ M_{v2} &= -\frac{1}{2} \Omega_h^2 (Al_t + Bl_m + c^*) \sin(2\alpha_v) \end{aligned} \tag{1.6}$$

With

$$C^* = \frac{m_b l_b^2}{2} + m_{cb} l_{cb}^2 \tag{1.7}$$

Moment of the Aerodynamic Force M_{v3}:

The aerodynamic lift force depends on the angular velocity of the main rotor. The relationship will be determined experimentally using empirical methods (non-linearity, so we will give the expression later in this report). We can therefore write the expression of the force as:

$$F_v = S_f f_v(\omega_m) \tag{1.8}$$

The resulting moment is therefore:

$$M_{v3} = l_m S_f f_v(\omega_m) \tag{1.9}$$

Where: ω_m is the angular velocity of the main rotor, and $f_v(\omega_m)$ is the dependence of the aerodynamic force on the angular velocity of the main rotor.

Moment of the Friction Force M_{v4}:

The frictions here are viscous; they are proportional to the rotational speed around the horizontal axis:

$$M_{v4} = -\Omega_v k_v \tag{1.10}$$

With: $\Omega_v = \frac{d\alpha_v}{dt}$ is the angular velocity of the arm around the horizontal axis, and k_v is the friction constant that we will determine experimentally.

Moment of Inertia:

The moment of inertia allows us to express the distribution of the body's mass during its rotation around an axis. It is calculated using the following formula:

$$J = \int r^2 dm \tag{1.11}$$

The solids that make up our flight simulator are basic geometric shapes (cylinders, circles, rods...), the moments of such figures are known, so we can write:

$$\begin{cases} J_{v1} = m_{mr} l_m^2 \\ J_{v2} = m_m \frac{l_m^2}{3} \\ J_{v3} = m_{tr} l_t^2 \\ J_{v4} = m_b \frac{l_b^2}{3} \\ J_{v5} = m_{cb} l_{cb}^2 \\ J_{v6} = m_t \frac{l_t^2}{3} \\ J_{v7} = m_{ms} l_m^2 + \frac{m_{ms}}{2} r_{ms}^2 \\ J_{v8} = m_{ts} l_t^2 + m_{ts} r_{ts}^2 \end{cases} \tag{1.12}$$

Where:

- r_{ms}: The radius of the main propeller.
- r_{ts} : The radius of the tail propeller.

1.4.2 Horizontal subsystem

Similarly to the vertical plane, the tail rotor is responsible for most of the horizontal dynamics. Since gravity does not have an effect in this plane, aerodynamic force is the primary force causing motion. Additionally, friction in the joints is always present during motion and must be accounted for, as illustrated in 1.11 By always applying Newton's second law, we obtain:

$$M_h = J_h \frac{d^2 \alpha_h}{dt^2} \tag{1.13}$$

With:

$$\begin{cases} M_h = \sum_{i=1}^{2} M_{hi} \\ J_h = \sum_{i=1}^{8} J_{hi} \end{cases} \tag{1.14}$$

Where:

- M_h: Moment of the resultant force in the horizontal plane.
- J_h: The sum of moments of inertia about the vertical axis.
- α_h: The azimuth angle of the beam.

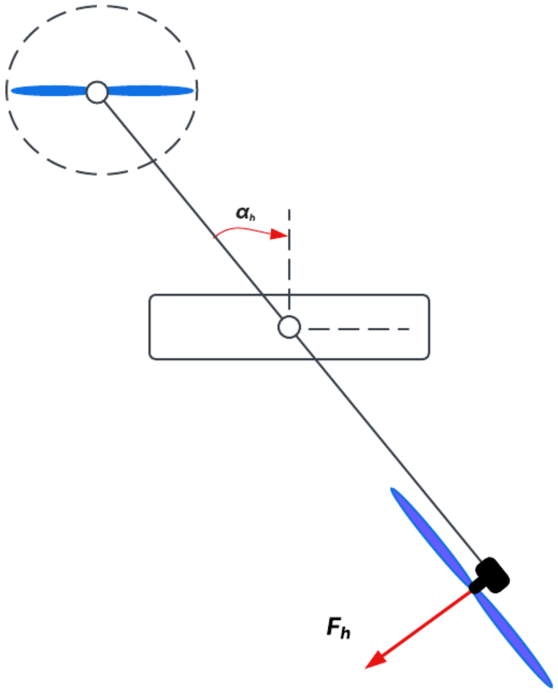


Figure 1.11: horizontal subsystem

Moment of the Aerodynamic Force M_{h1}:

From figure 1.11 which shows the direction of the aerodynamic force, this force depends on the angular velocity of the tail rotor. Therefore, the moment of the aerodynamic force is as follows:

$$M_{h1} = l_t S_f F_h(\omega_t) \cos(\alpha_v) \tag{1.15}$$

Where:

- ω_t: The angular velocity of the tail rotor.
- $F_h(\omega_t)$: The dependence of the aerodynamic force on the angular velocity of the tail rotor.

Moment of the Friction Force M_{h2}:

We still consider viscous-type friction, so the general form of the moment of the friction force is:

$$M_{h2} = -\Omega_h k_h \tag{1.16}$$

with:

- $\Omega_h = \frac{d\alpha_h}{dt}$: Angular velocity of the beam around the vertical axis.
- k_h: Friction constant to be determined experimentally.

Moment of Inertia J_h:

The moments of inertia about the vertical axis are:

$$\begin{cases} J_{h1} = \frac{m_m}{3}(l_m \cos\alpha_v)^2 \\ J_{h2} = \frac{m_t}{3}(l_t \cos\alpha_v)^2 \\ J_{h3} = \frac{m_b}{3}(l_b \sin\alpha_v)^2 \\ J_{h4} = m_{tr}(l_t \cos\alpha_v)^2 \\ J_{h5} = m_{mr}(l_m \cos\alpha_v)^2 \\ J_{h6} = m_{cb}(l_{cb} \sin\alpha_v)^2 \\ J_{h7} = m_{ms}r_{ms}^2 + m_{ms}(l_m \cos\alpha_v)^2 \\ J_{h8} = \frac{m_{ts}}{2}r_{ts}^2 + m_{ts}(l_t \cos\alpha_v)^2 \end{cases} \tag{1.17}$$

Or in compact form:

$$J_h(\alpha_v) = D\cos^2\alpha_v + E\sin^2\alpha_v + F \tag{1.18}$$

With:

$$\begin{cases} D = \left(\frac{m_m}{3} + m_{mr} + m_{ms}\right) l_m^2 + \left(\frac{m_t}{3} + m_{tr} + m_{ts}\right) l_t^2 \\ E = \frac{m_b}{3} l_b^2 + m_{cb} l_{cb}^2 \\ F = m_{ms} r_{ms}^2 + \frac{m_{ts}}{2} r_{ts}^2 \end{cases}$$

Gyrosopic Effect:

When a helicopter is rotated around an axis, it generates several phenomena such as the gyroscope effect. Additionally, an object in rotation possesses a kinetic moment proportional to its inertia and angular velocity, which is expressed by the formula $S = I\omega$. Koenig's theorem links the kinetic moment and the sum of moments as follows:

$$\frac{dS}{dt} = \sum_{i=1}^{n} M(\vec{F_i})$$

In a helicopter, the rotor (mass) rotates around a main axis, just like in a gyroscope. In case of system perturbation, it will try to compensate by performing a rotation around a horizontal axis parallel to the applied force. Thus, any force applied outside the center of mass of the system will cause a resulting perpendicular movement. The following relations are derived:

$$\begin{cases} S_v = J_v\Omega_v - J_{tr}\omega_t \\ S_h = J_h(\alpha_v)\Omega_h - J_{mr}\omega_m cos(\alpha_v) \end{cases}$$

With:

- $J_{tr}\omega_t$: is the kinetic moment generated by the tail rotor to counteract the gyroscopic effect.

The kinetic moment $J_{mr}\omega_m\alpha_v$ is generated by the main rotor to counteract the gyroscopic effect due to the rotation of the tail rotor. By replacing all the results found, we can summarize the dynamics of our system as follows:

$$\begin{cases}\frac{dS_v}{dt} = l_m S f F_v(\omega_m) - \Omega_h^2 (A l_t + B l_m + C^*) \sin(\alpha_v)\cos(\alpha_v) - \Omega_v k_v \\ +g[(A-B)\cos(\alpha_v) - C\sin(\alpha_v)] \\ \Omega_v = \frac{d\alpha_v}{dt} \\ S_v = J_v \Omega_v - J_{tr}\omega_t \\ \frac{dS_h}{dt} = l_t S f F_h(w_t)\cos(\alpha_v) - \Omega_h K_h \\ \Omega_h = \frac{d\alpha_h}{dt} \\ S_h = J_h(\alpha_v)\Omega_h - J_{mr}\omega_m cos(\alpha_v)\end{cases} \tag{1.19}$$

1.4.3 Dynamics of Propellers

The Twin Rotor MIMO System (TRMS) is equipped with two propellers, each consisting of a directe current motor and a rotor airscrew. Although the two rotors are identical, they have different mechanical charges.

We consider the simple model of an MCC with an external charge:

$$I\dot{\omega} = \frac{k_i}{R}(u - k_b\omega) - T_L \tag{1.20}$$

With:

- ω: Angular velocity of the rotor (rad/s).
- u: Command voltage (V).
- I: Moment of inertia.
- R: Resistance of the armature.
- k_b: Constant of the EMF.
- k_i: Torque constant.
- T_L: Resistant torque generated by the load.

The load T_L encompasses the mechanical and aerodynamic frictions resulting from the rotation of the rotors. Since these frictions are complex to model, a new variable U_{vv} for the vertical axis and U_{hh} for the horizontal axis has been introduced, and two non-linear functions have been experimentally determined to characterize their behavior. As a result, the subsystems of the propellers become first-order systems. of first order with a nonlinear output function as follows [6]:

$$\begin{cases}\frac{du_{vv}}{dt} = \frac{1}{T_{mr}}(-u_{vv} + k_{mr}u_v) \\ \frac{du_{hh}}{dt} = \frac{1}{T_{tr}}(-u_{hh} + k_{tr}u_h)\end{cases} \tag{1.21}$$

With:

$$\begin{cases}\omega_m = p_v(u_{vv}) \\ \omega_t = p_h(u_{hh})\end{cases}$$

Where:

- T_{mr}: The time constant of the main rotor.
- T_{tr}: The time constant of the tail rotor.
- k_{mr}: The static gain of the main rotor.
- k_{tr}: The static gain of the tail rotor.

This can be illustrated in the following figure1.12

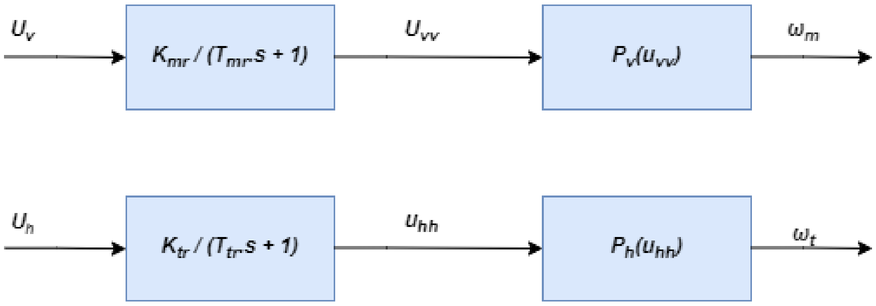


Figure 1.12: Bloc diagram of propellers

1.4.4 Nonlinearities and Model Parameters:

We will now establish the various nonlinear equations related to the thrusters and aerodynamic forces. The characteristics of the aerodynamic forces can be expressed in polynomial form as follows:

$$\begin{cases} F_v(\omega_m) = -3.48 \cdot 10^{-12}\omega_m^5 + 1.09 \cdot 10^{-9}\omega_m^4 + 4.123 \cdot 10^{-6}\omega_m^3 - 1.632 \cdot 10^{-4}\omega_m^2 + 9.544 \cdot 10^{-2}\omega_m \\ F_h(\omega_t) = -3 \cdot 10^{-14}\omega_t^5 - 1.595 \cdot 10^{-11}\omega_t^4 + 2.511 \cdot 10^{-7}\omega_t^3 - 1.808 \cdot 10^{-4}\omega_t^2 + 0.0801\omega_t \end{cases} \tag{1.22}$$

$$\begin{cases} p_v(u_{vv}) = \omega_m = 90.99u_{vv}^6 + 599.73u_{vv}^5 - 129.26u_{vv}^4 - 1238.64u_{vv}^3 + 63.45u_{vv}^2 + 1283.41u_{vv} \\ p_h(u_{hh}) = \omega_t = 2020u_{hh}^5 - 194.69u_{hh}^4 - 4283.15u_{hh}^3 + 262.27u_{hh}^2 + 3796.83u_{hh} \end{cases} \tag{1.23}$$

After establishing the equations governing our system, we must now define the parameter values of our system in order to simulate it.

Parameter	Numerical Value	Unit
m_{mr}	0.228	kg
m_m	0.0145	kg
m_{tr}	0.206	kg
m_t	0.0155	kg
m_{cb}	0.068	kg
m_b	0.022	kg
m_{ms}	0.225	kg
m_{ts}	0.165	kg
l_m	0.24	m
l_t	0.25	m
l_b	0.26	m
l_{cb}	0.13	m
g	9.806	m/s^2
J_v	0.055448	kg·m^2
J_{tr}	$2.65 \cdot 10^{-5}$	kg·m^2
J_{mr}	$1.6543 \cdot 10^{-5}$	kg·m^2
r_{ms}	0.155	m
r_{ts}	0.10	m

Table 1.1: TRMS Parameters.

We can also summarize the table of time constants and various gains present in our system:

Parameter	Numerical Value	Unit
k_v	0.00545371	V·s/rad
k_h	0.0095	N·m/V
T_{mr}	1.432	s
T_{tr}	0.3842	s
k_{mr}	1	–
k_{tr}	1	–
S_f	$8.4332 \cdot 10^{-4}$	N·m·s/rad

Table 1.2: Motor Parameters.

1.5 State Space Representation

One of the representations that allows for a complete description of our system and enables analysis of stability, controllability, and observability is the state space representation, which is given as follows:

$$\begin{cases} \dot{X} = F(X,U) \\ Y = h(X) \end{cases}$$

We must therefore define a state vector **X**, an input vector **U**, and an output vector **Y**. We have chosen the following vectors:

- Input vector: $U = [u_v \;\; u_h]^T$
- State vector: $X = [x_1 \;\; x_2 \;\; x_3 \;\; x_4 \;\; x_5 \;\; x_6]^T = [\alpha_v \;\; S_v \;\; u_{vv} \;\; \alpha_h \;\; S_h \;\; u_{hh}]^T$
- Output vector: $Y = [\alpha_v \;\; \alpha_h]^T$

Using our mathematical model 1.19, we can provide the state representation of our simulator as follows:

$$\begin{cases} \dot{x}_1 = \dfrac{x_2 + J_{tr} P_h(x_6)}{J_v} \\ \dot{x}_2 = l_m S_f F_v(P_v(x_3)) - k_v \left[\dfrac{x_2 + J_{tr} P_h(x_6)}{J_v}\right] + g\left[(A-B)\cos(x_1) - C\sin(x_1)\right] \\ \qquad - \dfrac{1}{2}\left[\dfrac{x_5 + J_{mr} P_v(x_3)\cos(x_1)}{J_h(x_1)}\right]^2 (Al_t + Bl_m + C^*)\sin(2x_1) \\ \dot{x}_3 = \dfrac{1}{T_{mr}}(-x_3 + K_{mr} u_v) \\ \dot{x}_4 = \dfrac{x_5 + J_{mr} P_v(x_3)\cos(x_1)}{J_h(x_1)} \\ \dot{x}_5 = l_f S_f F_h(P_h(x_6))\cos(x_1) - k_h \left[\dfrac{x_5 + J_{mr} P_v(x_3)\cos(x_1)}{J_h(x_1)}\right] \\ \dot{x}_6 = \dfrac{1}{T_{tr}}(-x_6 + K_{tr} u_h) \end{cases} \tag{1.24}$$

1.5.1 Decoupling of the State Model

The state model 1.24 is nonlinear and strongly coupled. However, by restricting the motion of the rod to a single plane, we reduce the degrees of freedom from 2 to 1. This allows us to obtain two state representations for our two subsystems.

Vertical Subsystem:

In this first case, we fix the azimuth angle $\alpha_h = \alpha_{h0}$, so $U = u_v$ (the input voltage to the tail rotor motor is zero), and $y = \alpha_v$. The state vector becomes $X_v = [\alpha_v \;\; S_v \;\; u_{vv}]^T$. We can therefore write the first decoupled state model as follows:

$$\begin{cases} \dot{x}_{1v} = \frac{x_{2v}}{J_v} \\ \dot{x}_{2v} = g[(A-B)\cos(x_{1v}) - C\sin(x_{1v})] + l_m S_f F_v P_v(x_{3v}) - k_v \frac{x_{2v}}{J_v} \\ \dot{x}_{3v} = \frac{1}{T_{mr}}(-x_{3v} + k_{mr} u_v) \end{cases} \tag{1.25}$$

Horizontal Subsystem:

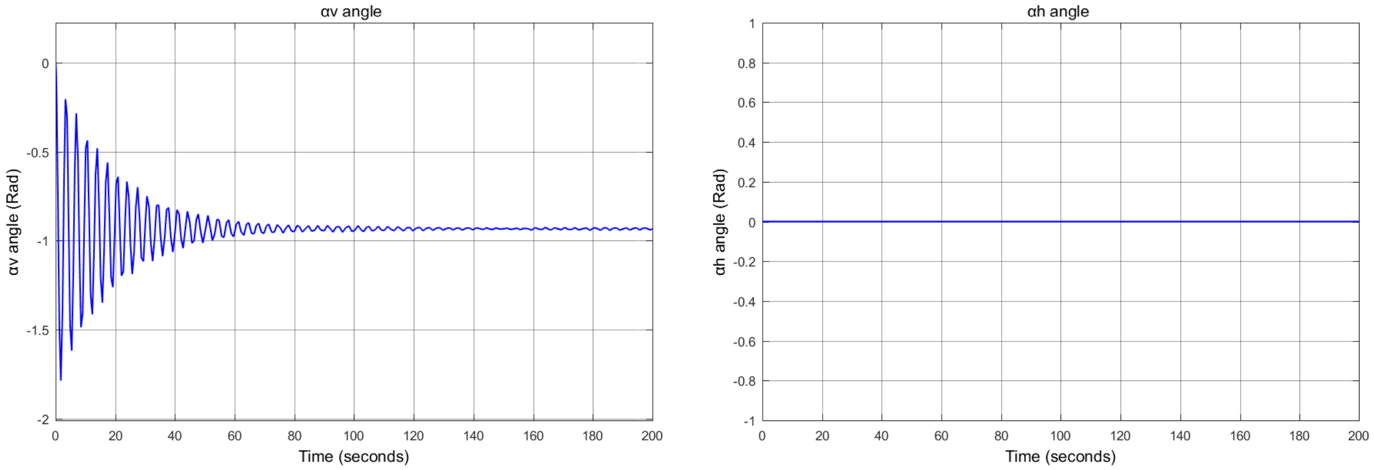
In this second case, we fix the elevation angle $\alpha_v = \alpha_{v0}$, so $U = u_h$ (the input voltage to the main rotor motor is zero), and $y = \alpha_h$. The state vector becomes $X_h = [\alpha_h \ \ S_h \ \ u_{hh}]^T$. We can therefore write the second decoupled state model as follows:

$$\begin{cases} \dot{x}_{1h} = \frac{x_{2h}}{J_h(\alpha_{v0})} \\ \dot{x}_{2h} = l_t S_f F_h P_h(x_{3h}) - \frac{k_h}{J_h(\alpha_{v0})} x_{2h} \\ \dot{x}_{3h} = \frac{1}{T_{tr}}(-x_{3h} + k_{tr} u_h) \end{cases} \tag{1.26}$$

1.5.2 Open-Loop Simulation

We will now perform some open-loop simulations on our system. We will plot the free responses and step responses of the two subsystems as well as the coupled system:

Free Response:



(a) Vertical subsystem

(b) Horizontal subsystem

Figure 1.13: Free response of the decoupled subsystems

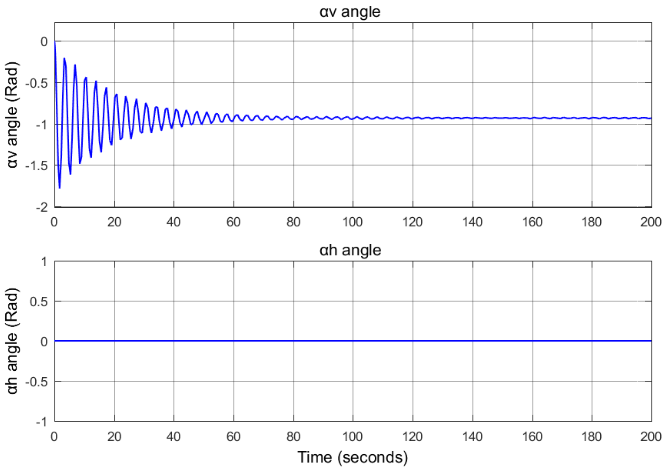
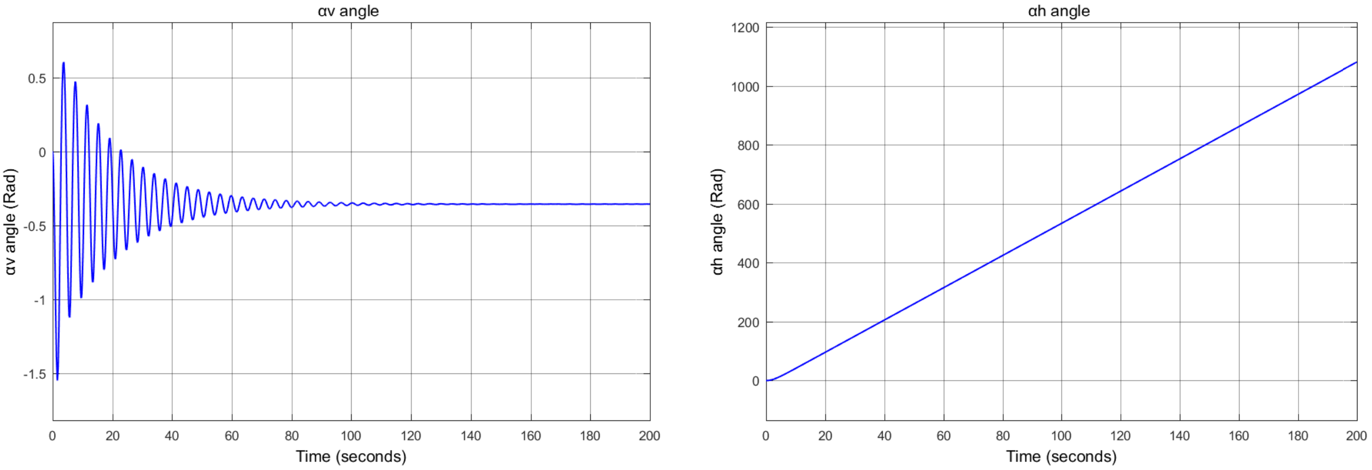


Figure 1.14: Free response of the coupled system

Step Response ($u = 0.5V$):



(a) Vertical subsystem (b) Horizontal subsystem

Figure 1.15: Step response of the decoupled subsystems ($u = 0.5V$)

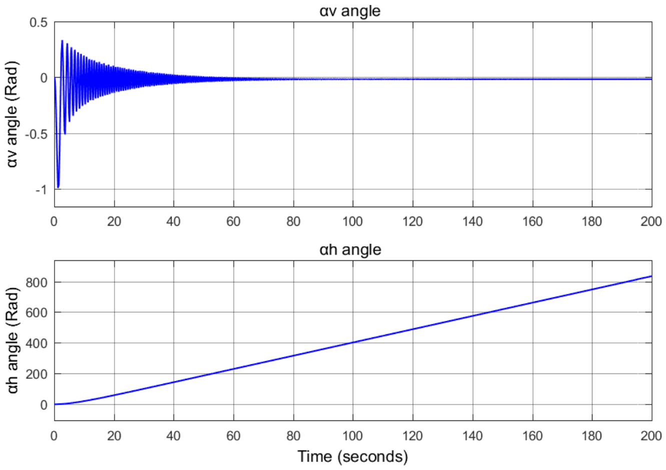


Figure 1.16: Step response of the coupled system ($u = 0.5V$)

1.5.3 Comments and Observations

1. **Vertical Subsystem**
   - We notice that the system stabilizes at an equilibrium point $\alpha_{v0} = -0.9\,\text{rad/sec}$, which results from friction forces.
   - The damped oscillatory response is due to the gravitational moment.
   - This vertical subsystem is stable for $|U_v| < 1.3\,\text{V}$.
2. **Horizontal Subsystem**
   - The horizontal subsystem remains in its initial state as long as it is not excited.
   - This subsystem diverges even for small excitations.
   - When the horizontal subsystem is excited, the vertical subsystem converges to a new equilibrium point.

1.6 Conclusion

In this chapter, we studied the TRMS system. We began by understanding the operation of a helicopter and how the TRMS simulator works.

We then established the mathematical model of our system by providing the differential equations governing its operation as well as the nonlinear state representation.

Finally, we performed some open-loop simulations to observe how our simulator behaves in the absence of a controller.

Chapter 2

fractional order calculus

2.1 Introduction

Fractional-order calculus is the field of mathematics that deals with the study and application of integrals and derivatives of arbitrary order. It is considered an ancient concept. Fractional-order calculus was introduced 300 years ago, and the question of fractional derivatives was first addressed in 1695 by Leibniz in a letter to L'Hôpital. When L'Hôpital asked him what the half-order derivative of the function x(t) with respect to the variable t might be, Leibniz replied that it led to an apparent paradox, from which useful consequences would one day be drawn. This event is now regarded as the birth of fractional calculus. Many mathematicians such as N.H. Abel, M. Caputo, L. Euler, J. Fourier, A.K. Grünwald, J. Hadamard, G.H. Hardy, O. Heaviside, P.S. Laplace, G.W. Leibniz, J. Liouville, B. Riemann, M. Riesz, and H. Weyl contributed to its development up until the middle of the last century [13]. However, fractional-order calculus is considered a new and interesting field of research, as it has only been the subject of extensive research for just over three decades. The first book dedicated to fractional-order calculus was published in 1974 by K.B. Oldham and J. Spanier [14], following collaborative work initiated in 1968. Today, the use of fractional-order calculus and its applications is rapidly growing in various fields.

Nevertheless, fractional-order calculus has been applied to almost all areas of science, engineering, and mathematics. Fractional-order calculus has had a profound impact on nearly every field, including viscoelasticity and rheology, electrical engineering, electrochemistry, biology, biophysics and biotechnology, signal and image processing, mechanics, mechatronics, physics, and especially in the field of control theory [15].

In this chapter, we will begin by exploring the application areas of fractional-order calculus, providing definitions related to fractional calculus and fractional-order operators, their main properties, and also the Laplace transforms of fractional-order derivatives and integrals. We will particularly present some approximation methods for fractional-order operators.

2.2 Application Areas of Fractional-Order Calculus

Over the past two decades, fractional-order calculus has been applied to almost every field of science, engineering, and mathematics. Fractional-order calculus has made a profound impact in nearly all domains,electrical engineering, electrochemistry, biology, biophysics and biotechnology, signal and image processing, mechanics, mechatronics, physics, and especially in the field of control theory. Although some mathematical issues remain unresolved, most challenges have been overcome, and most major mathematical problems in the field of fractional-order calculus have been addressed to the point where several mathematical tools for integer-order and fractional-order calculus are now identical [15].

In Control Systems Many authors have focused on non-integer order calculus and have used control laws that incorporate fractional derivatives. Podlubny[5, 16], and Caponetto et al. [17] demonstrated that the best approach to ensure effective control of fractional-order systems is the use of fractional-order controllers. They propose a generalization of traditional PID controllers. Mbodje and Montseny [18] and Matsuda and Fuji (1993) successfully applied fractional-order control laws to distributed parameter systems.

However, we cannot discuss fractional-order control without addressing the CRONE approach, introduced by Oustaloup (1991). The CRONE control is the work of a group of researchers at the Laboratoire de l'Intégration du Matériau au Système (IMS) in Bordeaux, under the direction of Alain Oustaloup. This approach has been the subject of numerous publications and several books. The CRONE methodology enables the synthesis of robust dynamic output feedback controls in the frequency domain for linear time-invariant (LTI) systems, whether uncertain, single-input single-output (SISO), or multi-input multi-output (MIMO). The performance it achieves can be explained by the minimally pessimistic consideration of uncertainties in the controlled systems, as well as the effectiveness of the tuning parameters used. This control strategy has been applied to automotive suspension systems, which also bear the same name. However, it is worth noting that the implemented compensator is of integer order and physically synthesized using classical components. The non-integer qualifier is still justified because the behavior of an authentic non-integer operator can be approximated with arbitrarily small error over a given frequency band.

In Electrical Engineering Fractional-order systems are particularly interesting for finely representing and reducing the order of devices whose operation relies on the diffusion of a quantity (field, temperature, etc.). The link between diffusion and non-integer-order modeling is analyzed in detail in [5, 16].

In Robotics Path Planning, Trajectory Optimization, and Beyond Fractional calculus has emerged as a transformative tool in robotics, particularly in path planning, trajectory optimization, and system modeling. For path planning, Likun Li et al. [19] introduced a fractional-order enhanced ant colony optimization (ACO) method tailored for car-like mobile robots operating in narrow or large environments. By integrating fractional-order kinematic models and dynamic angle constraints, their approach generates smoother, more efficient paths while adapting to real-time environmental changes. In trajectory control, Yixiao Ding et al. [3] developed a fractional-order impedance controller for robot manipulators, leveraging fractional derivatives to model damping forces with greater accuracy. This method outperforms traditional integer-order controllers in step response and disturbance rejection, ensuring precise trajectory tracking even under uncertainties.

For modeling and forecasting, Bhukya Ramadevi et al.[2] proposed a hybrid fractional-order neural network to address wind power prediction challenges. Combining long short-term memory (LSTM) networks with fractional-order layers, their model mitigates data gaps in wind speed and direction, enhancing forecasting reliability for renewable energy applications. Meanwhile, Kishore Bingi et al. [4] reviewed fractional-order modeling of robotic manipulators , demonstrating its superiority in capturing complex dynamics, such as viscoelasticity in actuators and memory-driven behaviors, over conventional integer-order models.

2.3 Fractional-order operators

Fractional calculus is a generalization of integration and differentiation to the fundamental operator of non-integer order $t_0 D_t^{\alpha}$, where t_0 and t are the limits of the operation. The continuous differential and integral operator is defined as [16]:

$$t_0 D_t^{\alpha} = \begin{cases} \frac{d^{\alpha}}{dt^{\alpha}} & \alpha > 0 \\ 1 & \alpha = 0 \\ \int_{t_0}^{t} (dt)^{\alpha} & \alpha < 0 \end{cases} \tag{2.1}$$

Where $\alpha \in \mathbb{R}$ is the order of the operation. Several mathematical definitions exist for fractional integration and differentiation.

2.3.1 Riemann-Liouville (R-L) Definition

Definition 1: Let $\alpha \in \mathbb{R}^{+}$ and $t_0 \in \mathbb{R}$, and let f be a locally integrable function defined on $[t_0, +\infty[$. The integral of order α of f with lower bound t is defined by [16]:

$${}_{t_0}^{RL}I^{\alpha} f(t) = \frac{1}{\Gamma(\alpha)} \int_{t_0}^{t} (t-\tau)^{\alpha-1} f(\tau) d\tau \tag{2.2}$$

where $t \geq t_0$ and Γ is the Euler Gamma function defined by:

$$\Gamma(\alpha) = \int_{0}^{\infty} e^{-y} y^{\alpha-1} dy \tag{2.3}$$

Definition 2: Let $\alpha \in \mathbb{R}^{+}$, $t_0 \in \mathbb{R}$, and f be a locally derivable function defined on $[t_0, +\infty[$. The derivative of order α of f with lower bound t is defined by:

$${}_{t_0}^{RL}D_t^{\alpha} f(t) = \frac{1}{\Gamma(n-\beta)} \frac{d^n}{dt^n} \int_{t_0}^{t} (t-\tau)^{n-\beta-1} f(\tau) d\tau \tag{2.4}$$

where n is an integer such that $n-1 < \beta < n$.

This definition can also be derived from Equation 2.2 as follows:

$${}_{t_0}^{RL}D_t^{\alpha} f(t) = \frac{d^n}{dt^n} \left\{ I^{(n-\alpha)} f(t) \right\} \tag{2.5}$$

2.3.2 Caputo Definition

At the end of the 1960s, within the framework of his work on dissipation in a linear viscoelastic material, Caputo introduced another definition of fractional differentiation [16] as follows:

$$ {}_{t_0}^{C}D_t^{\alpha} f(t) = I^{n-\alpha} D^n f(t) = \frac{1}{\Gamma(n-\alpha)} \int_{t_0}^{t} \frac{f^{(n)}(\tau)}{(t-\tau)^{\alpha-n+1}} d\tau \tag{2.6} $$

This definition can also be formulated in terms of the Riemann-Liouville definition as follows:

$$ {}_{t_0}^{C}D_t^{\alpha} f(t) = {}_{t_0}^{RL}D_t^{\alpha} f(t) + \sum_{k=0}^{n-1} \frac{t^{k-\alpha}}{\Gamma(k-\alpha+1)} f^{(k)}(0^+) \tag{2.7} $$

To illustrate the practical computation of the Caputo derivative, Figure 2.1 demonstrates its application to the function $f(t) = t^2$, highlighting the integral-based formulation and its dependence on the order α

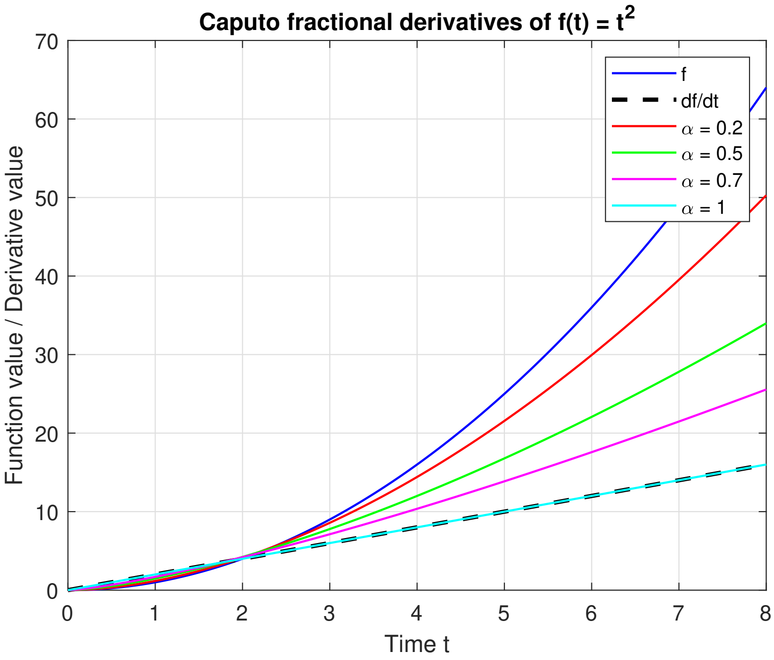


Figure 2.1: Fractional derivative using caputo method

2.3.3 Definition of Grünwald-Leitnikov (G-L)

The fractional derivative for $\alpha > 0$ of the G-L is given by [16]:

$$ {}_{t_0}^{GL}D_t^{\alpha} f(t) = \lim_{h \to 0} \frac{1}{h^{\alpha}} \sum_{k=0}^{\left[\frac{t-t_0}{h}\right]} (-1)^k \binom{\alpha}{k} f(t-k.h) \tag{2.8} $$

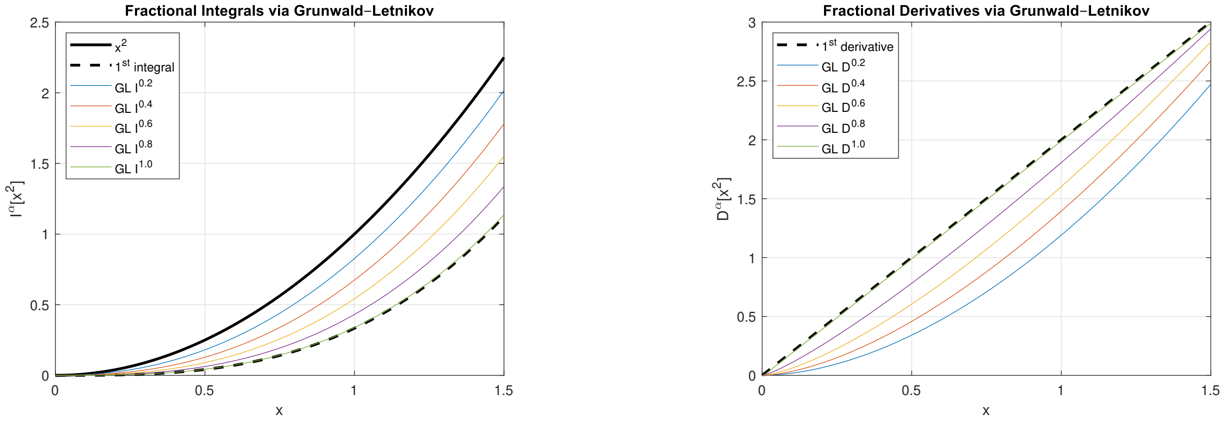
where $[\cdot]$ denotes the integer part of a real number, h is the sampling period, and the coefficients $\binom{\alpha}{k}$ are given by:

$$\binom{\alpha}{k} = \frac{\Gamma(\alpha+1)}{\Gamma(k+1)\cdot\Gamma(\alpha-k+1)} \tag{2.9}$$

The definition of Grünwald-Leitnikov for fractional integration is formulated as follows [15]:

$${}^{GL}_{t_0}I^{\alpha}_{t} f(t) = {}^{GL}_{t_0}D^{-\alpha}_{t} f(t) = \lim_{h\to 0} h^{\alpha} \sum_{k=0}^{\left[\frac{t-t_0}{h}\right]} (-1)^k \binom{-\alpha}{k} f(t-k.h) \tag{2.10}$$

To illustrate the discrete approximation framework of the Letnikov-Grunwald definition, Figures 2.2a and 2.2b demonstrate its application to the fractional integral and derivative of $f(t) = t^2$



(a) Fractional Integral

(b) Fractional Derivative

Figure 2.2: Comparison of fractional integral and derivative of $f(t) = t^2$ using the Letnikov-Grunwald definition.

2.4 Properties of Fractional Derivatives

- If $f(z)$ is an analytic function of z, then its fractional derivative $D^{\alpha}f(z)$ is also an analytic function of both z and α.
- For $\alpha = n$, where n is an integer, the operator $D^{\alpha}f(z)$ yields the classical derivative of order n.
- For $\alpha = 0$, the operator acts as the identity:

$$D^0 f(z) = f(z)$$

- Fractional-order differentiation and integration are linear operations:

$$D^m\left\{af(z) + bg(z)\right\} = aD^m\left\{f(z)\right\} + bD^m\left\{g(z)\right\}$$

- **Additive law (semigroup property):**

$$ {}_{t_0}D_t^{\alpha}\left[{}_{t_0}D_t^{\beta}f(t)\right] = {}_{t_0}D_t^{\beta}\left[{}_{t_0}D_t^{\alpha}f(t)\right] = {}_{t_0}D_t^{\alpha+\beta}f(t), \tag{2.11} $$

  valid when $\Re(\alpha) > 0$, $\Re(\beta) > 0$, and under certain regularity conditions on $f(t)$.

- **Commutation with integer-order derivatives:**

$$ \frac{d^n}{dt^n}\left({}_{t_0}D_t^{\alpha}f(t)\right) = {}_{t_0}D_t^{\alpha}\left(\frac{d^n f(t)}{dt^n}\right) = {}_{t_0}D_t^{\alpha+n}f(t), \tag{2.12} $$

  provided that $f^{(k)}(a) = 0$ for all $k = 0, 1, \dots, n-1$.

- **Successive fractional derivatives:**

$$ {}_{t_0}D_t^{\alpha}f(t) = {}_{t_0}D_t^{\alpha_1}\cdots{}_{t_0}D_t^{\alpha_n}f(t), \quad \text{where } \alpha = \sum_{i=1}^{n}\alpha_i, \quad 0 < \alpha_i < 1 $$

2.4.1 Properties of the Gamma Function

- $\Gamma(z)$ is defined and analytic in the region $\Re(z) > 0$.
- $\Gamma(n+1) = n!$ for all integers $n \geq 0$.
- **Functional equation:**

$$ \Gamma(z+1) = z\Gamma(z) $$

- This property, together with the previous one, characterizes the factorial function. Hence, $\Gamma(z)$ generalizes $n!$ to complex numbers z. Some authors denote this by $\Gamma(z+1) = z!$.
- $\Gamma(z)$ can be analytically continued to a meromorphic function on the whole complex plane, with simple poles at $0, -1, -2, \dots$. The residues at those poles are:

$$ \operatorname{Res}(\Gamma, -m) = \frac{(-1)^m}{m!} $$

- **Infinite product representation:**

$$ \Gamma(z) = \left[z e^{\gamma z} \prod_{n=1}^{\infty}\left(1 + \frac{z}{n}\right) e^{-z/n} \right]^{-1} $$

  where $\gamma \approx 0.577$ is the Euler–Mascheroni constant:

$$ \gamma = \lim_{n\to\infty}\left(\sum_{k=1}^{n}\frac{1}{k} - \log n\right) $$

- **Euler's reflection formula:**

$$ \Gamma(z)\Gamma(1-z) = \frac{\pi}{\sin(\pi z)} $$

This identity, together with the infinite product representation, leads to a product formula for $\sin(\pi z)$.

- **Stirling's formula (asymptotic approximation for large $|z|$, $\Re(z) > 0$):**

$$\Gamma(z+1) \approx \sqrt{2\pi}\, z^{z+1/2} e^{-z}$$

In particular:

$$n! \approx \sqrt{2\pi}\, n^{n+1/2} e^{-n}$$

- **Legendre's duplication formula:**

$$2^{2z-1}\Gamma(z)\Gamma(z+1/2) = \sqrt{\pi}\,\Gamma(2z)$$

2.5 Laplace Transform of Fractional-Order Operators

2.5.1 Laplace Transform of the Fractional Integral

The Laplace transform of the Riemann-Liouville fractional integral of order $\alpha > 0$, defined by Equation 2.2 and written as a convolution product of functions $g(t) = \frac{t^{\alpha-1}}{\Gamma(\alpha)}$ and $f(t)$ [15]:

$$I^{\alpha} f(t) = D^{-\alpha} f(t) = \frac{1}{\Gamma(\alpha)} \int_0^t (t-\tau)^{\alpha-1} f(\tau) d\tau \tag{2.13}$$

The Laplace transform of the function $t^{\alpha-1}$ is given by [16]:

$$G(s) = L\{t^{\alpha-1}\} = \frac{\Gamma(\alpha)}{s^{\alpha}}$$

Thus, the Laplace transform of the Riemann-Liouville fractional integral is:

$$L\{I^{\alpha}(f(t))\} = s^{-\alpha} F(s) \tag{2.14}$$

In the same way, the Laplace transform of the fractional integral defined by Grünwald-Leitnikov or Caputo is also given by Equation 2.10.

2.5.2 Laplace Transform of the Fractional Derivative

Definition of Riemann-Liouville:

$$L\{D^{\alpha} f(t)\} = s^{\alpha} F(s) - \sum_{k=0}^{n-1} s^k \left[D^{n-k-1} f(t)\right]_{t=0} \tag{2.15}$$

where $(n-1) < \alpha < n$; however, the practical applicability of the Laplace transform of the Riemann-Liouville derivative is limited due to the lack of physical interpretation of the fractional-order derivatives at $t = 0$.

Definition of Caputo:

$$L\{D^{\alpha} f(t)\} = s^{\alpha} F(s) - \sum_{k=0}^{n-1} s^{k-\alpha} f^{(k)}(0) \tag{2.16}$$

With $\alpha > 0$. The main advantage of the Caputo definition over the Riemann-Liouville definition is that it allows for the consideration of conventional initial conditions that are easy to interpret. Furthermore, the derivative of a constant in the Caputo sense is zero, whereas the derivative of a constant in the Riemann-Liouville sense is an unbounded function at $t = 0$.

Definition of Grünwald-Leitnikov [16]

$$L\{D^{\alpha} f(t)\} = s^{\alpha} F(s) \tag{2.17}$$

2.6 Approximation of Fractional-Order Operators

There exist several techniques for approximating fractional-order operators using rational functions. The available approximations in the frequency domain are referred to as analog approximations or frequency-domain approximations [20]. Among these methods, one can find:

- The Oustaloup method.

- The Carlson method.

- The Charef method.

2.6.1 Oustaloup Method

This method [20] is based on approximating a function of the form:

$$H(s) = s^{\alpha}, \alpha \in \mathbb{R} \tag{2.18}$$

by a rational function:

$$\widehat{H}(s) = C \prod_{k=-N}^{N} \frac{1 + \frac{s}{\omega'_k}}{1 + \frac{s}{\omega_k}} \tag{2.19}$$

Using the following set of formulas:

$$\omega'_0 = \alpha^{-0.5} \omega_u, \quad \omega_0 = \alpha^{0.5} \omega_u, \quad \frac{\omega'_{k+1}}{\omega'_k} = \frac{\omega_{k+1}}{\omega_k} = \alpha\eta > 1$$

$$\frac{\omega_k'}{\omega_k} = \eta > 0; \quad N = \frac{\log(\omega_N/\omega_0)}{\log(\alpha\eta)}; \quad \mu = \frac{\log \alpha}{\log(\alpha\eta)}, \eta = \frac{\omega_h}{\omega_l}$$

where ω_u is the unit frequency gain and ω_c is the central frequency of a geometrically distributed frequency band around it. Let $\omega_u = \sqrt{\omega_h \omega_l}$, where ω_h and ω_l are the high and low frequencies, respectively.

Approximation Order Comparison The transfer function approximations for $s^{0.5}$ using Oustaloup's method with orders $N = 2, 4, 6$ demonstrate increasing complexity:

$$\mathrm{H1} = \frac{2.512s^5 + 98.83s^4 + 531.7s^3 + 442.3s^2 + 56.87s + 1}{s^5 + 56.87s^4 + 442.3s^3 + 531.7s^2 + 98.83s + 2.512}$$

$$\begin{aligned} \mathrm{H2} = {} & \frac{2.512s^9 + 212.2s^8 + 4737s^7 + 3.469\times10^4 s^6 + 8.844\times10^4 s^5}{s^9 + 103.7s^8 + 2840s^7 + 2.552\times10^4 s^6 + 7.983\times10^4 s^5} \\ & \frac{+7.983\times10^4 s^4 + 2.552\times10^4 s^3 + 2840s^2 + 103.7s + 1}{+8.844\times10^4 s^4 + 3.469\times10^4 s^3 + 4737s^2 + 212.2s + 2.512} \end{aligned}$$

$$\mathrm{H3} = \frac{2.512s^7 + 155.9s^6 + 2045s^5 + 6806s^4 + 5967s^3 + 1378s^2 + 80.73s + 1}{s^7 + 80.73s^6 + 1378s^5 + 5967s^4 + 6806s^3 + 2045s^2 + 155.9s + 2.512}$$

where higher N yields better approximation accuracy at the cost of higher-order dynamics in the frequency range $[\omega_L, \omega_H] = [0.1, 1000]$ rad/s.

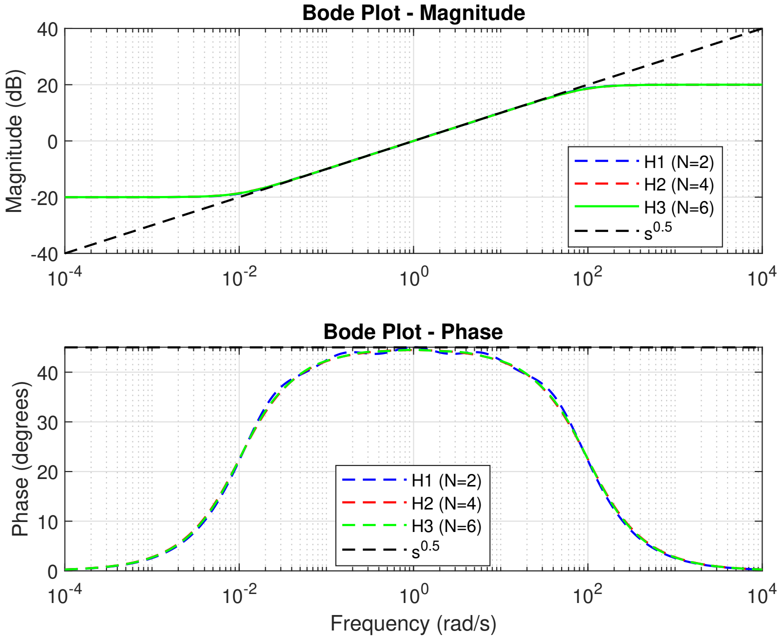


Figure 2.3: Effect of Approximation Order N

Bandwidth Impact on Approximation Accuracy The transfer function approximations for $s^{0.5}$ using Oustaloup's method with fixed order $N = 4$ and varying bandwidths demonstrate sensitivity to the chosen frequency range $[\omega_L, \omega_H]$. For $N = 4$, the approximations take the form:

$$\mathrm{H1} = \frac{\begin{array}{l}3.162s^9 + 53.25s^8 + 333.9s^7 + 1013s^6 + 1628s^5 \\ \quad + 1432s^4 + 690.1s^3 + 176.1s^2 + 21.75s + 1\end{array}}{\begin{array}{l}s^9 + 21.75s^8 + 176.1s^7 + 690.1s^6 + 1432s^5 \\ \quad + 1628s^4 + 1013s^3 + 333.9s^2 + 53.25s + 3.162\end{array}}$$

$$\mathrm{H2} = \frac{\begin{array}{l}10s^9 + 724.5s^8 + 1.387 \times 10^4 s^7 + 8.715 \times 10^4 s^6 + 1.905 \times 10^5 s^5 \\ \quad + 1.475 \times 10^5 s^4 + 4.045 \times 10^4 s^3 + 3860s^2 + 120.8s + 1\end{array}}{\begin{array}{l}s^9 + 120.8s^8 + 3860s^7 + 4.045 \times 10^4 s^6 + 1.475 \times 10^5 s^5 \\ \quad + 1.905 \times 10^5 s^4 + 8.715 \times 10^4 s^3 + 1.387 \times 10^4 s^2 + 724.5s + 10\end{array}}$$

$$\mathrm{H3} = \frac{\begin{array}{l}31.62s^9 + 1.275 \times 10^4 s^8 + 9.106 \times 10^5 s^7 + 1.35 \times 10^7 s^6 + 4.278 \times 10^7 s^5 \\ \quad + 2.915 \times 10^7 s^4 + 4.269 \times 10^6 s^3 + 1.337 \times 10^5 s^2 + 868.4s + 1\end{array}}{\begin{array}{l}s^9 + 868.4s^8 + 1.337 \times 10^5 s^7 + 4.269 \times 10^6 s^6 + 2.915 \times 10^7 s^5 \\ \quad + 4.278 \times 10^7 s^4 + 1.35 \times 10^7 s^3 + 9.106 \times 10^5 s^2 + 1.275 \times 10^4 s + 31.62\end{array}}$$

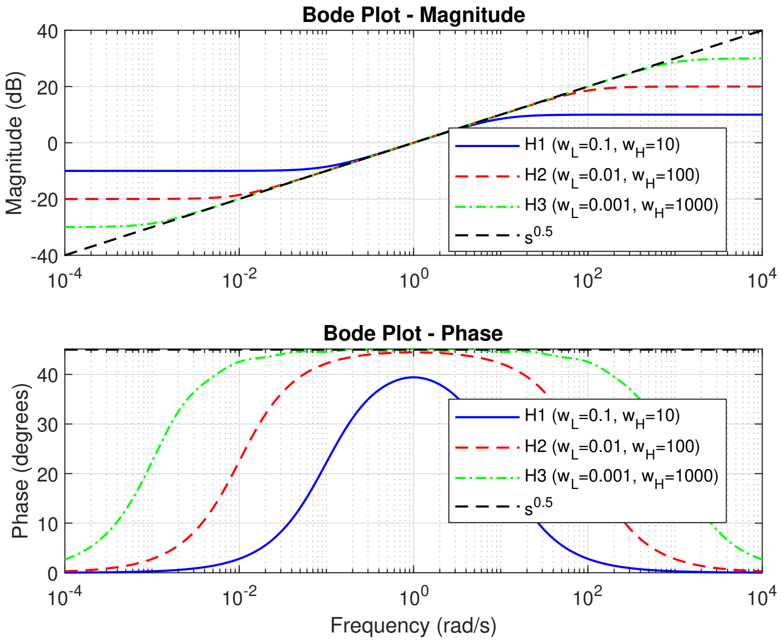


Figure 2.4: Effect of Bandwidth $[\omega_L, \omega_H]$ on $s^{0.5}$ Approximation with Fixed Order $N=4$.

2.6.2 Carlson Method

This approximation is based on Continued Fraction Expansion and Newton iterative process. In the method, fractional operator is defined as

$$H(s) = s^{\alpha} \tag{2.20}$$

The iterative Newton method will generate a sequence of approximations $H_i(s)$ starting with the initial value $H_0(s) = 1$ [5]. The rational function approximating $H(s)$ is obtained in the following form:

$$H_i(s) = H_{i-1}(s)\frac{\left(\frac{1}{\alpha} - 1\right)(H_{i-1}(s))^{1/\alpha} + \left(\frac{1}{\alpha} + 1\right)s}{\left(\frac{1}{\alpha} + 1\right)(H_{i-1}(s))^{1/\alpha} + \left(\frac{1}{\alpha} - 1\right)s} \tag{2.21}$$

The fractional order differentiator has the general transfer function $H(s) = s^{\alpha}$, where $0 < \alpha < 1$, which has to be approximated to an integer order differentiator model $H(s)$. The transfer functions for $S^{0.1}$, $S^{0.2}$, and $S^{0.5}$ obtained by substituting α as 0.1, 0.2, and 0.5 respectively in equation 2.21 for $i = 1$, are:

$$H_{0.1}(s) = s^{0.1} = \frac{11s + 9}{9s + 11} \tag{2.22}$$

$$H_{0.2}(s) = s^{0.2} = \frac{3s + 2}{2s + 3} \tag{2.23}$$

$$H_{0.5}(s) = s^{0.5} = \frac{3s + 1}{s + 3} \tag{2.24}$$

The models of equations. 2.22,2.23,2.24 obtained by first iteration of the Newton process, are of first order.

$1/\alpha$ can have only integer values, hence it is not possible to obtain the approximated transfer function of S^{α} directly for all other values of α.

For example: for $S^{0.3}$ we use $S^{0.1}S^{0.2}$. The approximated transfer function for

$$H_{0.3}(s) = s^{0.3} = s^{0.1}s^{0.2} = \frac{(11s + 9)(3s + 2)}{(9s + 11)(2s + 3)} = \frac{33s^2 + 49s + 18}{18s^2 + 49s + 33} \tag{2.25}$$

For $S^{0.8}$ and $S^{0.9}$, the approximations used are

$$S^{0.8} = S^{0.5}S^{0.2}S^{0.1} \quad \text{and} \quad S^{0.9} = S^{0.5}S^{0.2}S^{0.2} \tag{2.26}$$

		$G(s)$
$s^{0.1}$	$i=1$	$1.2222\left(\frac{s+0.8182}{s+1.222}\right)$
	$i=2$	$1.4938 \cdot \frac{(s+9.697)(s+1.002)(s+0.8182)(s+0.8781)(s^2+2s+1.005)}{(s^2+1.985s+1.006)(s^2+1.933s+1.012)(s^2+1.614s+1.054)} \cdot \frac{(s+12.81)(s+1.222)(s+0.998)(s+0.1031)(s^2+1.991s+0.9955)}{(s^2+1.972s+0.9938)(s^2+1.91s+0.9879)(s^2+1.53s+0.9487)}$
$s^{0.2}$	$i=1$	$1.5\left(\frac{s+0.6667}{s+1.5}\right)$
	$i=2$	$2.25 \cdot \frac{(s+8.733)(s+0.6667)(s+0.06545)}{(s^2+2.019s+1.038)(s^2+1.7455s+1.123)} \cdot \frac{(s+15.28)(s+1.5)(s+0.1145)}{(s^2+1.945s+0.963)(s^2+1.553s+0.8903)}$
$s^{0.3}=s^{0.2}s^{0.1}$		$1.8333\left(\frac{(s+0.6667)(s+0.8182)}{(s+1.5)(s+1.222)}\right)$
$s^{0.4}=s^{0.2}s^{0.2}$		$2.25\left(\frac{s+0.6667}{s+1.5}\right)^2$
$s^{0.5}$		$3\left(\frac{s+0.3333}{s+3}\right)$
$s^{0.6}=s^{0.5}s^{0.1}$		$3.6667\left(\frac{(s+0.3333)(s+0.8182)}{(s+3)(s+1.222)}\right)$
$s^{0.7}=s^{0.2}s^{0.5}$		$4.5\left(\frac{(s+0.3333)(s+0.6667)}{(s+3)(s+1.5)}\right)$
$s^{0.8}=s^{0.5}s^{0.2}s^{0.1}$		$5.5\left(\frac{(s+0.3333)(s+0.6667)(s+0.8182)}{(s+3)(s+1.5)(s+1.222)}\right)$
$s^{0.9}=s^{0.5}s^{0.2}s^{0.2}$		$6.75\left(\frac{(s+0.3333)(s+0.6667)^2}{(s+3)(s+1.5)^2}\right)$

Table 2.1: Approximated Transfer Functions for Various Fractional Orders

Approximation Order Comparison The transfer function approximations for $s^{0.5}$ using Carlson's method show progressively more complex rational forms when implementing successive iterations of the approximation algorithm:

$$
\begin{aligned}
\text{first iteration} & \\
\text{H1} &= \frac{3s+1}{s+3}, \\
\text{seconde iteration} & \\
\text{H2} &= \frac{9s^4+84s^3+126s^2+36s+1}{s^4+36s^3+126s^2+84s+9}, \\
\text{third iteration} & \\
\text{H3} =& \frac{27s^{13}+2925s^{12}+80730s^{11}+888030s^{10}}{s^{13}+351s^{12}+17550s^{11}+296010s^{10}} \\
& \frac{+4.687\times10^6s^9+1.304\times10^7s^8+2.006\times10^7s^7}{+2.22\times10^6s^9+8.436\times10^6s^8+1.738\times10^7s^7} \\
& \frac{+1.738\times10^7s^6+8.436\times10^6s^5+2.22\times10^6s^4}{+2.006\times10^7s^6+1.304\times10^7s^5+4.687\times10^6s^4} \\
& \frac{+296010s^3+17550s^2+351s+1}{+888030s^3+80730s^2+2925s+27}
\end{aligned}
$$

where higher-order approximations provide better accuracy at the cost of increased computational complexity.

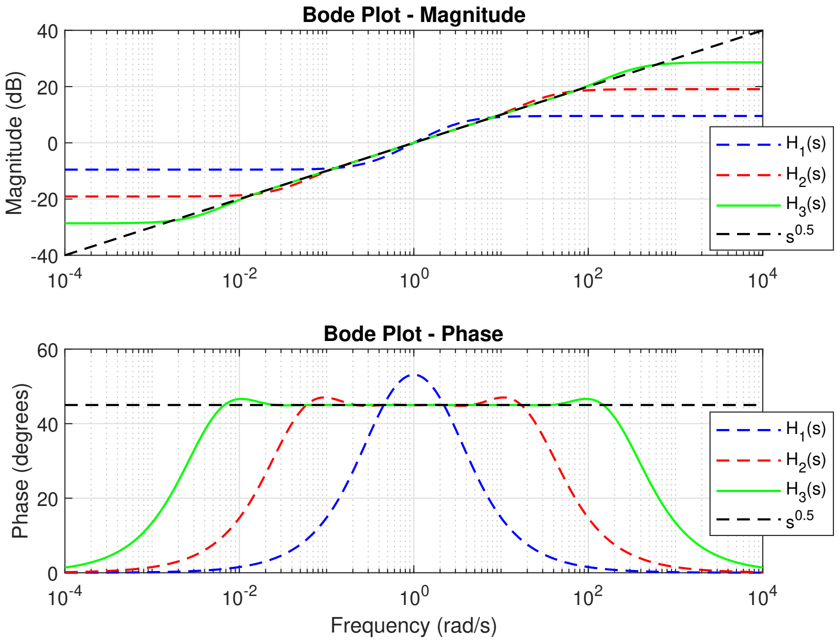


Figure 2.5: Response of f-o differentiator $s^{0.5}$ on applying Carlson method

2.6.3 Charef Method (Singularity Function)

The approximation method will differ depending on the order of the fractional transfer function to be approximated, whether it is first or second order [21]:

Fractional-Order System of First Order

For a first-order fractional system, the modeling is in the form:

$$H(s) = \frac{1}{(1+\frac{s}{p_T})^{\beta}} \tag{2.27}$$

Equation 2.27 can be written in the form :

$$H(s) = \frac{1}{(1+\frac{s}{p_T})^{\beta}} = \lim_{N\to\infty} \prod_{i=0}^{N-1} \frac{(1+\frac{s}{z_i})}{(1+\frac{s}{p_i})} \tag{2.28}$$

Where $(N+1)$ is the total number of singularities that can be determined by the frequency band of the system. Equation 2.20 can be reduced to a finite number N, and the approximation becomes:

$$H(s) = \frac{1}{(1+\frac{s}{p_T})^{\alpha}} \approx \frac{\prod_{i=0}^{N-1}(1+\frac{s}{z_i})}{\prod_{i=0}^{N}(1+\frac{s}{p_i})} \tag{2.29}$$

The poles and zeros of the singularity function can be obtained as follows:

$$\begin{cases} p_i = (ab)^i p_0 & i = 1, 2, 3, \ldots, N \\ z_i = (ab)^i a p_0 & i = 1, 2, 3, \ldots, N-1 \end{cases} \tag{2.30}$$

with

$$\begin{aligned} p_0 &= p_T 10^{\frac{\epsilon_p}{20\beta}} \\ a &= 10^{\frac{\epsilon_p}{10(1-\beta)}} \\ b &= 10^{\frac{\epsilon_p}{\log(ab)}} \\ \beta &= \frac{\log(a)}{\log(ab)} \end{aligned}$$

ϵ_P is the tolerated error in dB.

Fractional-Order System of Second Order

A second-order system is described by the equation:

$$G(s) = \frac{1}{\left(\frac{s^2}{\omega_n^2} + 2\xi\frac{s}{\omega_n} + 1\right)^{\beta}} \tag{2.31}$$

where β is a positive real number such that $0 < \beta < 1$. Two cases can be distinguished:

- **Case** $0 < \beta < 0.5$

One can express the function 2.31 as follows:

$$G_e(s) = \frac{\left(\frac{s}{\omega_n} + 1\right)\left(\frac{s}{\omega_n + 1}\right)^{\eta}}{\left(\frac{s^2}{\omega_n^2} + 2\xi\frac{s}{\omega_n} + 1\right)} \tag{2.32}$$

With $\alpha = \zeta^{\beta}$ and $n = 1 - 2\beta$, this can also be approximated by the function:

$$G_e(s) \approx \frac{\left(\frac{s}{\omega_n} + 1\right)\prod_{i=1}^{N-1}\left(1 + \frac{s}{z_i}\right)}{\left(\frac{s^2}{\omega_n^2} + 2\xi\frac{s}{\omega_n} + 1\right)\prod_{i=1}^{N}\left(1 + \frac{s}{p_i}\right)} \tag{2.33}$$

The singularities (poles p_i and zeros z_i) are given by the following formulas:

$$\begin{cases} p_i = (ab)^{i-1} a z_1 & i = 1, 2, 3, \ldots, N \\ z_i = (ab)^{i-1} z_1 & i = 2, 3, \ldots, N-1 \end{cases} \tag{2.34}$$

With

$$\begin{cases} z_1 = \omega_n \sqrt{b} \\ a = 10^{\frac{\varepsilon_p}{10(1-\eta)}} \\ b = 10^{\frac{\varepsilon_p}{10\eta}} \\ \eta = \frac{\log(a)}{\log(ab)} \end{cases} \tag{2.35}$$

ε_P is the tolerated error in dB.

The approximation order N is calculated by fixing the frequency band of operation, specified by $N_{\max}$ such that: $p_{n-1} < \omega_{\max} < p_n$, which leads to the following value:

$$N = \left\lfloor \frac{\log\left(\frac{\omega_{\max}}{p_i}\right)}{\log(ab)} + 1 \right\rfloor + 1 \tag{2.36}$$

The coefficients a_m and b_m are calculated from the singularities p_i and z_i such that α and ω_i

- For $0.5 < \beta < 1$

The approximation function is given as follows:

$$G_E(s) = \frac{\left(\frac{s}{\omega_n} + 1\right)\left(\frac{s}{\omega_n+1}\right)^{\eta}}{\left(\frac{s^2}{\omega_n^2} + 2\xi\frac{s}{\omega_n} + 1\right)} \tag{2.37}$$

Where

$$\begin{aligned} p_i &= (ab)^{i-1} p_1 \quad i = 1, 2, 3, \ldots, N \\ z_i &= (ab)^{i-1} a p_1 \quad i = 2, 3, \ldots, N-1 \\ p_1 &= \omega_n \sqrt{\frac{\varepsilon_p}{b}} \\ a &= 10^{\frac{\varepsilon_p}{10(1-\eta)}} \\ b &= 10^{\frac{\varepsilon_p}{10\eta}} \\ \eta &= \frac{\log(a)}{\log(ab)} \end{aligned}$$

ε_P is the error in dB.

2.7 Representations of Fractional-Order Systems

The classification diagram of linear and time-invariant fractional-order systems is given by the following Figure2.6:

There are several ways to represent fractional-order systems, such as representation by differential equations, recurrence equations, transfer functions, and state-space representations.

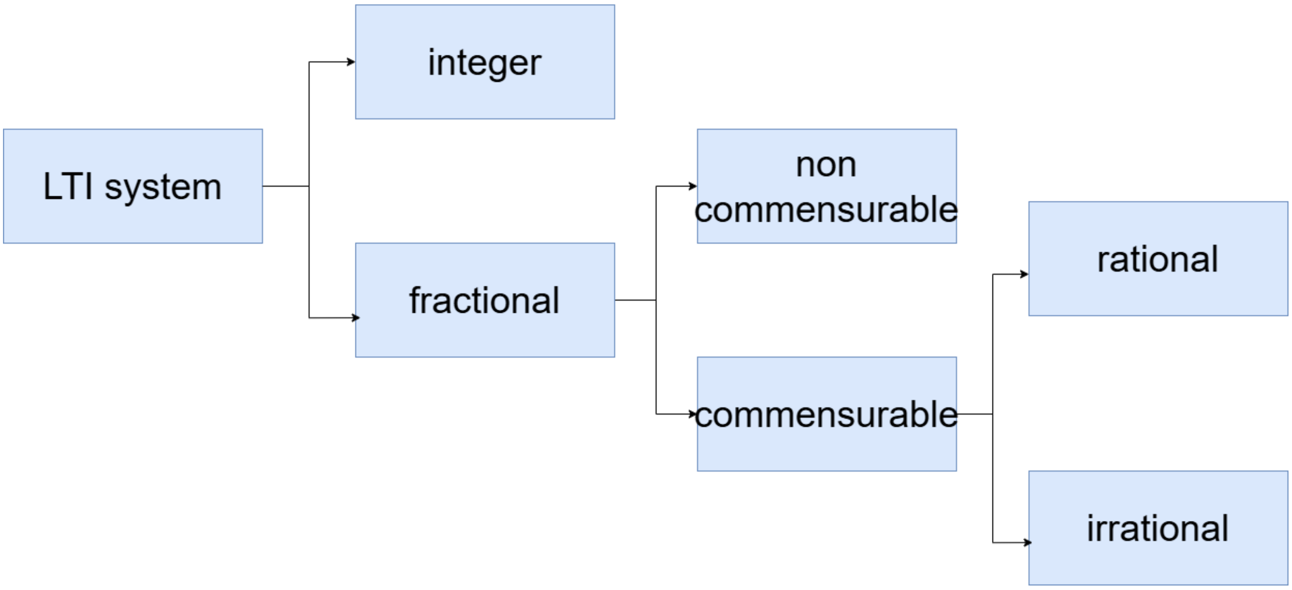


Figure 2.6: Classification of LTI systems: integer-order vs. fractional-order

2.7.1 Representation by Differential Equations

A fractional-order linear system is, by definition, a system described by a fractional differential equation of the form [16]:

$$a_n D^{\alpha_n} y(t) + a_{n-1} D^{\alpha_{n-1}} y(t) + \ldots + a_0 D^{\alpha_0} y(t) = b_0 D^{\beta_m} u(t) + b_{m-1} D^{\beta_{m-1}} u(t) + \ldots + b_0 D^{\beta_0} u(t) \tag{2.38}$$

Where $u(t)$ and $y(t)$ are respectively the input and output of the fractional system, α_i and β_i are the orders of the derivatives, which are real numbers. When the orders α_i and β_j are multiples of the same real number α $(0 < \alpha < 1)$, such that $\alpha_i = i \cdot \alpha$ $(0 < i < n)$ and $\beta_j = j \cdot \alpha$ $(0 < j < m)$, the fractional system is said to be commensurable. Then, the fractional differential equation 2.38 becomes:

$$\sum_{i=0}^{a} a_i D^{\alpha_i} y(t) = \sum_{j=0}^{m} b_j D^{\beta_j} u(t) \tag{2.39}$$

2.7.2 Representation by Transfer Function

The Laplace transform of the differential equation 2.38 allows determining the transfer function of the fractional-order system as follows:

$$F(s) = \frac{Y(s)}{U(s)} = \frac{\sum_{j=0}^{m} b_j s^{\beta_j}}{1 + \sum_{i=1}^{n} a_i s^{\alpha_i}} \tag{2.40}$$

In the case of commensurate-order fractional systems α of Equation 2.40, this transfer function is written in the form:

$$F(s) = \frac{\sum_{j=0}^{m} b_j s^{\alpha_j}}{1 + \sum_{i=1}^{n} a_i s^{\alpha_i}} \tag{2.41}$$

2.7.3 Representation by State Space of a Fractional-Order System

The state representation of a continuous-time fractional-order system is written in the form:

$$\begin{cases} D^{\alpha}x(t) = A.x(t) + B.u(t) \\ y(t) = C.x(t) + D.u(t) \end{cases} \tag{2.42}$$

Where $u(t)$ is the input vector, $y(t)$ is the output vector, $x(t)$ is the state vector, and α is the order of derivation $(0 < \alpha < 1)$. The matrices A, B, C, and D are of appropriate dimensions with constant coefficients, called the state matrix, input matrix, output matrix, and direct transmission matrix, respectively.

2.7.4 Controllability, Observability, and Stability

Theorem 1 The fractional order system of Equation 2.42 is controllable if and only if the controllability matrix given by

$$M_{\text{com}} = \begin{bmatrix} B & AB & \dots & A^{L-1}B \end{bmatrix} \tag{2.43}$$

is invertible; where L is the number of state variables .

Theorem 2 The fractional order system of Equation 2.42 is observable if and only if the observability matrix given by:

$$M_{\text{ob}} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{L-1} \end{bmatrix} \tag{2.44}$$

is invertible; where L is the number of state variables .

Theorem 3 The fractional-order system 2.42 is stable (in the sense that bounded inputs will always produce bounded outputs) if and only if all the roots (complexes) r of its denominator satisfy:

$$|\arg(r)| > \frac{\pi}{2}\alpha \tag{2.45}$$

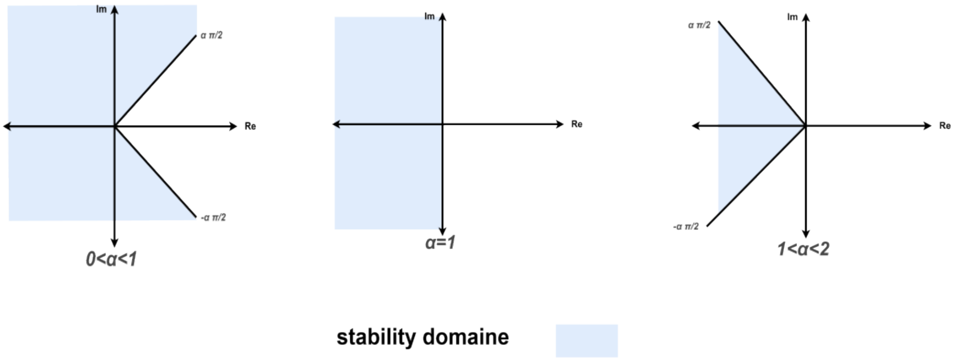


Figure 2.7: Stability Region in the s^{α}-Plane for a Fractional-Order System

2.8 Conclusion

This chapter is an introduction to the basic elements of fractional calculus. We have presented some essential concepts regarding fractional-order control systems necessary for understanding our work on the fractional-order control of dynamic systems.

After introducing non-integer order processes, we presented the definitions of fractional-order operators and fractional-order systems. The characteristics and representation and approximation methods of these systems are also studied.

Chapter 3

TRMS Control: Optimized PID and FOPID Using PSO

3.1 Introduction

PID (Proportional-Integral-Derivative) controllers have been a fundamental tool in control engineering for over a century. Originating in the early 20th century, PID controllers gained widespread adoption due to their simplicity, reliability, and effectiveness in regulating a wide range of systems, from industrial processes to aerospace applications. The PID structure, with its three terms proportional, integral, and derivative provides a balanced approach to achieving desired system performance by minimizing errors, reducing overshoot, and improving stability. PID controllers remain a go-to solution for many control problems due to their adaptability and ease of tuning [22].

In recent years, advancements in control theory have led to the development of more sophisticated controllers, such as the Fractional-Order PID (FOPID). The FOPID controller extends the traditional PID framework by introducing fractional-order calculus [23], which allows for non-integer orders in the integral and derivative terms. This added flexibility enables finer tuning and better performance, particularly for complex, nonlinear, or highly coupled systems. FOPID controllers have shown promise in applications requiring high precision and robustness, such as robotics, energy systems, and aerospace engineering.

Building on these developments, this chapter focuses on the application of both PID and FOPID controllers to the TRMS (Twin Rotor MIMO System), a multivariable system known for its nonlinear dynamics and strong coupling between its two rotors. The TRMS serves as an excellent testbed for evaluating control strategies, as it mimics the behavior of real-world systems like helicopters and drones, where coupling effects and external disturbances are significant challenges.

The primary objective of this work is to design, optimize, and compare the performance of PID and FOPID controllers for the TRMS. Using the Particle Swarm Optimization (PSO) algorithm [24], the parameters of both controllers are tuned to minimize control error and enhance system performance. The controllers are first tested on a decoupled version of the TRMS, where rotor interactions are neglected, and then validated on the coupled system, where these interactions are treated as disturbances. This approach allows us to evaluate the robustness and effectiveness of each controller in handling real-world complexities.

The chapter is structured as follows: First, the design and optimization of the PID and FOPID controllers are presented, with a focus on the PSO-based tuning process [25]. Next, the performance of both controllers is analyzed through simulations, comparing their ability to achieve stability, precision, and disturbance rejection. Finally, the results are discussed, highlighting the advantages of FOPID in achieving superior control for complex, coupled systems like the TRMS.

3.2 Particle Swarm Optimization (PSO)

Particle Swarm Optimization (PSO) is a population-based optimization algorithm inspired by the social behavior of birds flocking or fish schooling. It is widely used for solving complex optimization problems due to its simplicity, efficiency, and ability to find global optima in high-dimensional search spaces.

3.2.1 PSO Algorithm Steps

The PSO algorithm operates as follows:

Algorithm 1 Particle Swarm Optimization (PSO) Algorithm

```
Initialize parameters: swarm size $n$, cognitive coefficient $c_1$, social coefficient $c_2$, inertia weight $\omega$, position and velocity bounds, and maximum number of iterations.
Initialize positions and velocities of $n$ particles randomly.
for each particle $i$ in the swarm do
    Evaluate the objective function $f$ at the current position.
    Set personal best $P^i_{\text{best}}$ = current position.
end for
Set global best $G_{\text{best}}$ = best position among all particles.
while termination condition not met (e.g., max iterations or convergence) do
    for each particle $i$ in the swarm do
        Update velocity:
            $v_i \leftarrow \omega v_i + c_1 r_1 (P^i_{\text{best}} - x_i) + c_2 r_2 (G_{\text{best}} - x_i)$
        where $r_1, r_2 \sim \mathcal{U}(0, 1)$ are random numbers.
        Update position:
            $x_i \leftarrow x_i + v_i$
        Evaluate objective function $f$ at new position.
        Update personal best $P^i_{\text{best}}$ if current position is better.
    end for
    Update global best $G_{\text{best}}$ if any particle has a better solution.
end while
return Global best solution $G_{\text{best}}$.
```

3.2.2 Explanation of PSO Components

The PSO algorithm initializes a swarm of particles, where each particle represents a potential solution to the optimization problem. Each particle has a position and a velocity in the search space, which are updated iteratively based on the following principles [**?**]:

- **Personal Best ($P_{\mathbf{best}}$)**: Each particle remembers the best position it has encountered so far.
- **Global Best ($G_{\mathbf{best}}$)**: The swarm shares the best position found by any particle in the entire population.
- **Velocity Update**: The velocity of each particle is updated based on its current velocity, the distance to its personal best, and the distance to the global best.
- **Position Update**: The position of each particle is updated using its new velocity.

Mathematically, the updates for the personal best and global best are given by:

$$P_{\text{best},i}^{k+1} = \begin{cases} P_{\text{best},i}^{k}, & \text{if } f(x_i^{k+1}) \geq f(P_{\text{best},i}^{k}), \\ x_i^{k+1}, & \text{otherwise.} \end{cases} \tag{3.1}$$

$$G_{\text{best}}^{k+1} = \arg \min_{1 \leq i \leq n} f(P_{\text{best},i}^{k+1}). \tag{3.2}$$

where:

- $P_{\text{best}i}^{k}$: The personal best of particle i at iteration k.
- G_{best}^{k+1}: The global best at iteration $k+1$.
- $f(x)$: The objective function (fitness function) to be minimized.

3.2.3 Stopping Criteria

The stopping criterion for PSO can be:

- A fixed number of iterations.
- A threshold on the fitness value.
- When the velocities of the particles approach zero (indicating convergence).

3.2.4 Challenges and Improvements

While PSO is a powerful and versatile optimization algorithm, it faces some challenges in practical applications. A notable issue is the potential **divergence of particles** from the defined search space, leading to suboptimal solutions or algorithm instability. To mitigate these problems, several enhancements to the standard PSO have been proposed in the literature:

- **Velocity Clamping:** This technique limits the maximum allowable velocity ($V_{\max}$) of each particle, ensuring that the particles remain within the feasible region of the search space. By capping the velocity, particles are prevented from "exploding" out of bounds and the algorithm maintains a stable exploration behavior [26, 27].
- **Constriction Coefficient (χ):** A more theoretically grounded approach introduces a **constriction factor** into the velocity update equation to improve convergence stability and control the swarm dynamics. The velocity update with constriction is defined as [27]:

$$v_{i,j}^{k+1} = \chi \left[v_{i,j}^{k} + c_1 r_{1i,j}^{k} (P_{\text{best}j}^{k} - x_{i,j}^{k}) + c_2 r_{2i,j}^{k} (G_{\text{best}j}^{k} - x_{i,j}^{k}) \right], \quad j \in \{1, \ldots, D\} \tag{3.3}$$

where:

- $\chi = \frac{2}{|2-\phi-\sqrt{\phi^2-4\phi}|}$, with $\phi = \phi_1 + \phi_2 > 4$ [28].
- A common choice is $\phi_1 = \phi_2 = 2.05$, leading to $\chi \approx 0.7298844$ [27].

These enhancements strike a balance between **exploration** (diversifying the search across the space) and **exploitation** (refining solutions in promising areas), which is crucial for ensuring robust convergence and avoiding premature stagnation.

3.2.5 Application to PID and FOPID Tuning

In this study, PSO is applied to optimize the parameters of **PID** and **FOPID** controllers for the **Twin Rotor MIMO System (TRMS)**. The objective is to fine-tune controller parameters to achieve the desired system performance by minimizing a predefined cost function, such as the **Integral of Squared Error (ISE)** or other performance indices.

The optimized controller parameters, obtained through the PSO algorithm, allow the TRMS to meet stability and performance criteria, ensuring improved response characteristics such as reduced overshoot, faster settling time, and better disturbance rejection.

3.3 PID Control

The Proportional-Integral-Derivative (PID) controller is one of the most widely used control strategies in engineering, renowned for its simplicity, effectiveness, and versatility. Its structure comprises three key components: the proportional (P), integral (I), and derivative (D) actions, each playing a distinct role in shaping the system's response.

The output of a PID controller, $u(t)$, is calculated using the following equation:

$$u(t) = K_p \cdot e(t) + K_i \int e(t)\, dt + K_d \frac{de(t)}{dt} \tag{3.4}$$

Where:

- $u(t)$: The control signal sent to the system.
- $e(t)$: The error signal, defined as the difference between the desired reference $y_r(t)$ and the system output $y(t)$, i.e., $e(t) = y_r(t) - y(t)$.
- K_p: The proportional gain, which adjusts the controller's response to the current error.
- K_i: The integral gain, which addresses accumulated past errors to eliminate steady-state offsets.
- K_d: The derivative gain, which predicts future error trends based on the rate of change of the error.

Role of Each Term

- Proportional Term ($K_p \cdot e(t)$): This term provides an immediate response proportional to the current error. It reduces steady-state error and improves response speed, but excessive gain can lead to overshoot or instability.
- Integral Term ($K_i \int e(t)\,dt$): This term addresses accumulated past errors, eliminating steady-state offsets and ensuring long-term accuracy. However, high integral gains can cause sluggish responses or oscillations.
- Derivative Term ($K_d \frac{de(t)}{dt}$): This term predicts future error trends based on the rate of change of the error. It enhances stability and reduces overshoot but can amplify noise if not properly tuned.

In this section, we focus on the design and implementation of a PID controller for the TRMS. The primary objective is to stabilize the system and achieve key performance metrics, such as fast response times, minimal overshoot, and robustness to disturbances. The PID controller's parameters K_p, K_i, and K_d are optimized using the Particle Swarm Optimization (PSO) algorithm to ensure optimal performance.

3.3.1 PID Control of the TRMS

The main objective of this section is to stabilize and track the angular positions α_v (vertical) and α_h (horizontal) of the Twin Rotor MIMO System (TRMS). To achieve this, we implement a PID controller for each subsystem, one for the vertical axis and one for the horizontal axis.

The control laws governing these two PID controllers are defined as follows:

$$u_v(t) = K_{pv} e_v(t) + K_{iv} \int e_v(t) dt + K_{dv} \frac{de_v(t)}{dt} \tag{3.5}$$

$$u_h(t) = K_{ph} e_h(t) + K_{ih} \int e_h(t) dt + K_{dh} \frac{de_h(t)}{dt} \tag{3.6}$$

where:

- $e_v(t) = \alpha_{vr}(t) - \alpha_v(t)$ represents the tracking error for the vertical angle.

- $e_h(t) = \alpha_{hr}(t) - \alpha_h(t)$ represents the tracking error for the horizontal angle.
- K_{pv}, K_{iv}, K_{dv} are the proportional, integral, and derivative gains for the vertical subsystem.
- K_{ph}, K_{ih}, K_{dh} are the proportional, integral, and derivative gains for the horizontal subsystem.

The **tuning of these controller gains** is performed using the Particle Swarm Optimization (PSO) algorithm, which estimates the optimal values of the parameters to ensure effective control of the TRMS.

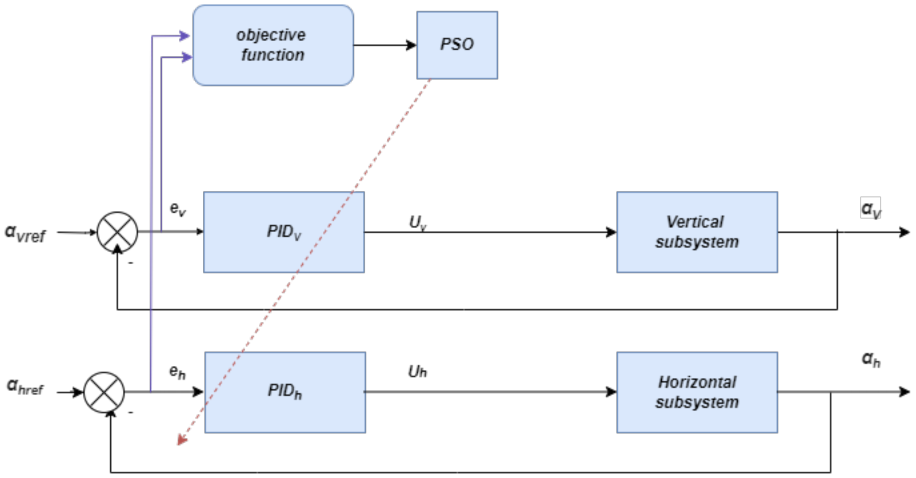


Figure 3.1: Block diagram of the PID controller optimized using PSO

To optimize the six PID gains $(K_{ph}, K_{ih}, K_{dh}, K_{pv}, K_{iv}, K_{dv})$, we define the following PSO parameters:

- Optimization vector:
$$K = \{K_{ph}, K_{ih}, K_{dh}, K_{pv}, K_{iv}, K_{dv}\}$$
- Number of particles: $n = 50$
- Number of iterations: $m = 100$
- Lower and upper bounds of the PID gains:

$$K(i) \in [0, 15], \quad i \in \{1, \ldots, 6\}$$

Optimization Criterion

The PSO algorithm minimizes the following fitness function, which evaluates the cumulative tracking error over the entire simulation period:

$$f(m) = \sum_{j=1}^{N} (e_h(j)^2 + e_v(j)^2) \tag{3.7}$$

where N is the total number of samples, defined as:

$$N = \frac{\text{simulation time}}{\text{integration step}} \tag{3.8}$$

This function ensures that the optimization process minimizes the overall control error, leading to improved tracking accuracy and system stability.

The PID gains obtained through Particle Swarm Optimization (PSO) are presented in **Table 3.1**.

Table 3.1: Optimized PID Gains for TRMS using PSO

Subsystem	**Proportional Gain (K_p)**	**Integral Gain (K_i)**	**Derivative Gain (K_d)**
Vertical (α_v)	2.0395	7.83	6.556
Horizontal (α_h)	0.31145	0.086634	0.29503

3.3.2 Simulation Results

This part presents the results obtained from implementing the PSO-optimized PID controllers for the TRMS. Various tests were conducted to assess their tracking accuracy, stability, and robustness under different conditions.

Figures 3.2 and 3.3 present the closed-loop simulation results of the optimized PID control for the Twin Rotor MIMO System. These results demonstrate the effectiveness of the PID controllers in stabilizing and tracking the desired angular positions while respecting the saturation constraints for motors protection.

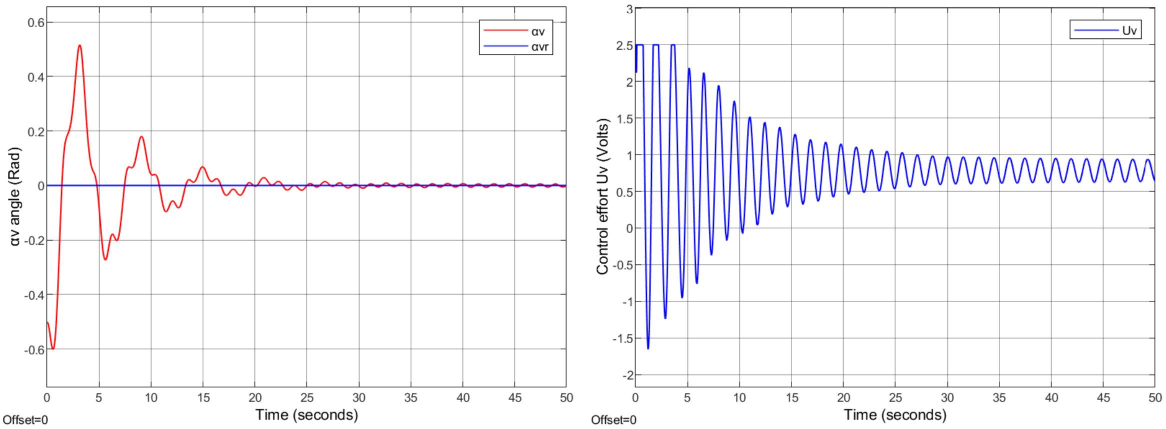


Figure 3.2: Simulation results of the optimized PID control for the vertical subsystem

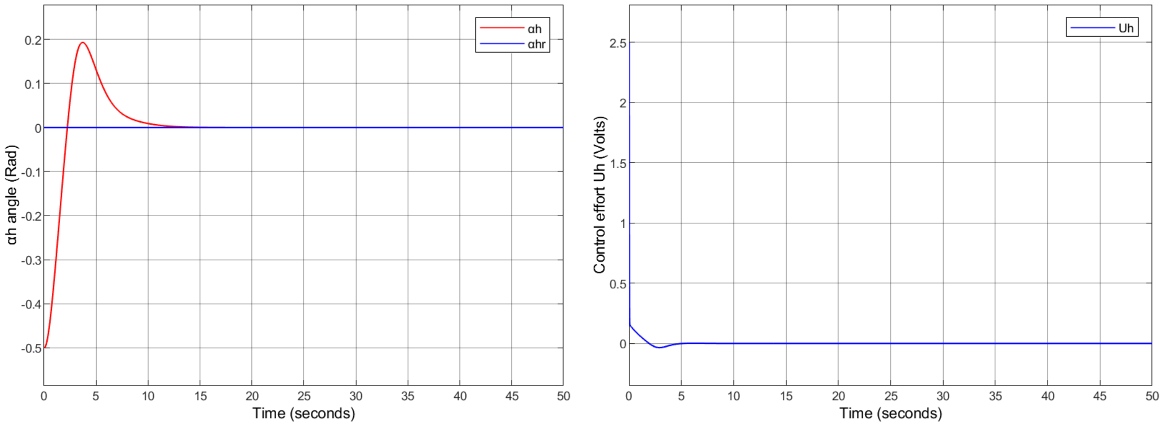


Figure 3.3: Simulation results of the optimized PID control for the horizontal subsystem

Observations from the simulation Results Using PID Control

Vertical Subsystem Response (α_v)

- The system starts with an initial condition of $\alpha_v = -0.5$ radians and converges to the desired setpoint of zero radian.
- The response exhibits a significant overshoot (0.5 radian) and high-frequency oscillations in the transient phase
- The vertical angle stabilizes at zero radian, demonstrating the stability of the PID controller.

Horizontal Subsystem Response (α_h)

- The system starts with an initial condition of $\alpha_h = -0.5$ radians and quickly converges to the desired setpoint of zero radian.
- The system exhibits a small overshoot (0.2 radians) but quickly settles.
- The horizontal angle stabilizes at zero radian, demonstrating the stability of the PID controller.

Control Effort and Saturation Constraints

- The control efforts (U_v and U_h) for both subsystems are constrained within the saturation limits ($-2.5 \leq U_v, U_h \leq 2.5$) to protect the motors and ensure safe operation.
- Initially, the control efforts are high to compensate for the initial deviations ($\alpha_v = -0.5$ and $\alpha_h = -0.5$) and bring the system back to the desired setpoints.
- As the system stabilizes, the control efforts decrease, showing that the PID controllers effectively reduce the errors without excessive control inputs.

3.3.3 Robustness Testing of the PID Control

To evaluate the robustness of the optimized PID control, we conducted tests by modifying key parameters of the Twin Rotor MIMO System TRMS. Specifically, **we changed the weight and the position** of the counterweight. These modifications simulate real-world variations and disturbances, allowing us to assess the controller's ability to maintain stability and performance under different conditions.

Simulation Results

Changing the position of the counterweight

We modified the position of the counterweight by 7 cm. The results are shown in Figures 3.4 and 3.5)

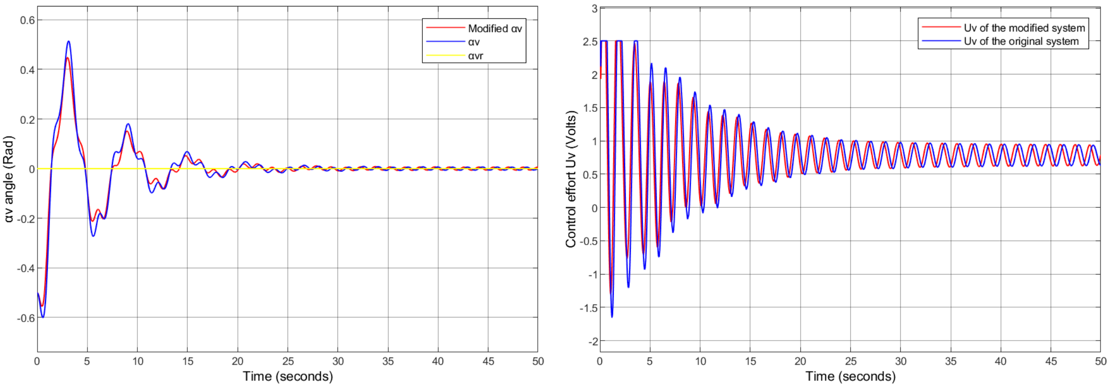


Figure 3.4: Modified Vertical Subsystem (lcb) Stabilization Using PID.

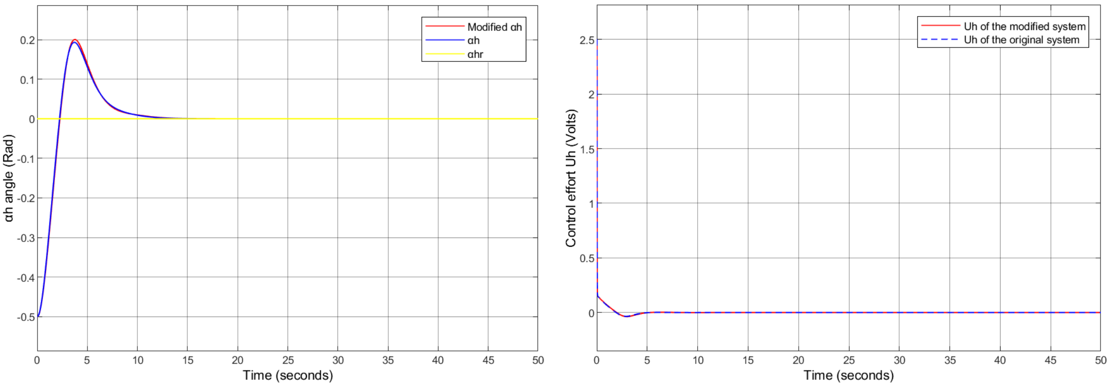


Figure 3.5: Modified Horizontal Subsystem (lcb) Stabilization Using PID.

Changing the mass of the counterweight

We modified the mass of the counterweight by adding 20g. The results are shown in Figures (3.6 and 3.7)

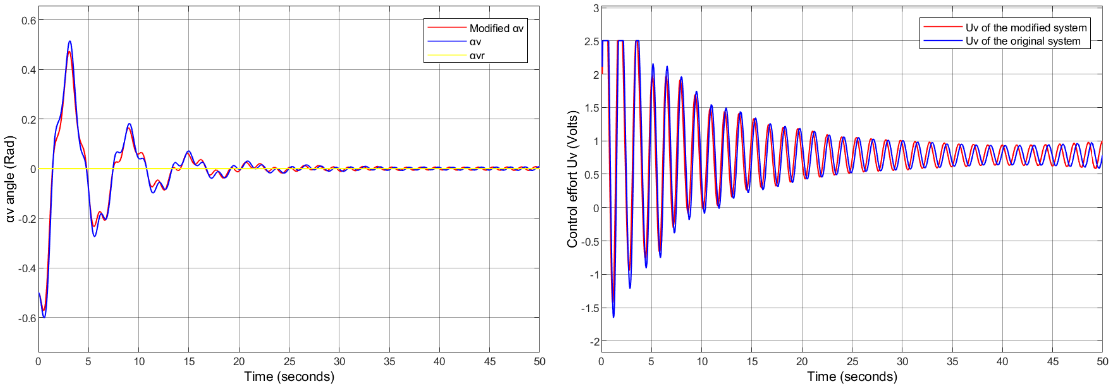


Figure 3.6: Modified Vertical Subsystem (mcb) Stabilization Using PID

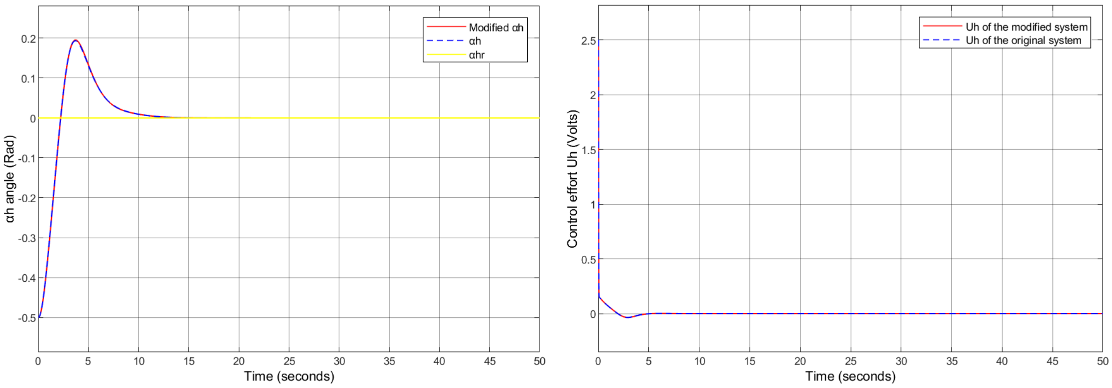


Figure 3.7: Modified Horizontal Subsystem (mcb) Stabilization Using PID

Observations on the Robustness Test

During this robustness tests, we first altered the position of the counterweight, then modified its mass. In both scenarios, we observed similar results

Vertical Subsystem Response

- The vertical angle (α_v) converges to the reference in both modified systems
- Demonstrates the controller's adaptive capability to dynamic parameter changes

Horizontal Subsystem Response

- Stable trajectory tracking with minimal oscillations (α_h)
- Rapid convergence to reference in all configurations.

3.4 FOPID Control

The Fractional-Order PI$^\lambda$D$^\mu$ (also known as PI$^\lambda$D$^\mu$ controller) was introduced by [29] as a generalization of the classical PID controller. Unlike conventional PID controllers, this approach uses an integrator of real order λ and a differentiator of real order μ. The transfer function of the Fractional-Order Controller (FOC) in the Laplace domain is given as follows [29], [30]):

$$C(s) = \frac{U(s)}{E(s)} = K_p + K_i s^{-\lambda} + K_d s^{\mu}, \quad (0 < \lambda, \mu < 2). \tag{3.9}$$

where:

- K_p is the proportional gain,
- K_i is the integration constant,
- K_d is the differentiation constant.

As illustrated in Figure 3.8, the internal structure of the fractional-order controller consists of a parallel connection of the proportional, integral, and derivative components [31].

The fractional differential equation is given by:

$$u(t) = K_p e(t) + K_i\, D_t^{-\lambda} e(t) + K_d\, D_t^{\mu} e(t), \tag{3.10}$$

or its discrete transfer function form:

$$C(z) = \frac{U(z)}{E(z)} = K_p + \frac{K_i}{(\omega(z^{-1}))^{\lambda}} + K_d(\omega(z^{-1}))^{\mu}, \tag{3.11}$$

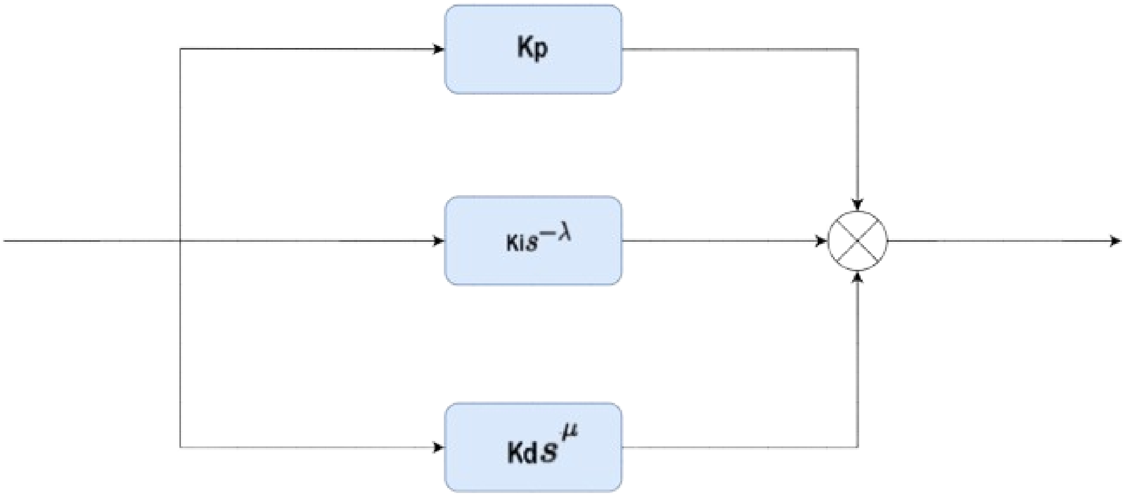


Figure 3.8: General Structure of a PI$^\lambda$D$^\mu$ Controller.

where $\omega(z^{-1})$ denotes the discrete operator, expressed as a function of the complex variable z or the shift operator z^{-1}.

Taking $\lambda = 1$ and $\mu = 1$, we obtain a classical PID controller. If $\lambda = 0$ and $K_i = 0$, we obtain a PD$^\mu$ controller, etc.

All these types of controllers are particular cases of the fractional-order controller, which is more flexible and provides an opportunity to better adjust the dynamical properties of the fractional-order control system.

It can be expected that the PI$^\lambda$D$^\mu$ controller (3.9) may enhance the system's control performance due to more tuning knobs introduced, which is intuitively illustrated in Figure 3.9.

Figure 3.9: PID controller: from points to plane.

3.4.1 Controller Tuning Methods for FOPID Controllers

The tuning of the PI$^\lambda$D$^\mu$ controller parameters is determined according to the given requirements. These requirements are, for example, the damping ratio, the steady-state error (e_{ss}), dynamical properties, etc. One of the methods being developed is the method of dominant roots [32], [33], based on the given stability measure and the damping ratio of the closed control loop. Assume that the desired dominant roots are a pair of complex conjugate roots as follows:

$$s_{1,2} = -\sigma \pm j\omega_d, \tag{3.12}$$

designed for the damping ratio ζ and natural frequency ω_n. The damping constant (stability measure) is $\sigma = \zeta\omega_n$ and the damped natural frequency of oscillation is:

$$\omega_d = \omega_n\sqrt{1-\zeta^2}. \tag{3.13}$$

The design of parameters K_p, T_i, λ, T_d and μ can be computed numerically from the characteristic equation. More specifically, for a simple plant model $P(s)$, this can be done by solving:

$$\min_{K_p,T_i,\lambda,T_d,\mu} ||C(s)P(s)+1||_s = -\sigma \pm j\omega_d. \tag{3.14}$$

Another possible way to obtain the controller parameters is using the tuning formula, based on gain A_m and phase Φ_m margins specifications for crossover frequency ω_{cg}. Gain and phase margins have always served as important measures of robustness. The equations that define the phase margin and the gain crossover frequency are expressed as [34], [35]:

$$|C(j\omega_{cg})P(\omega_{cg})|_{\text{dB}} = 0 \text{ dB}, \tag{3.15}$$

$$\arg(C(j\omega_{cg})P(\omega_{cg})) = -\pi + \Phi_m. \tag{3.16}$$

The above equations are often used also for the so-called auto-tuning techniques. For instance, the relay auto-tuning process has been widely used in industrial applications and it was already modified for fractional-order controllers [34].

Other numerical rules for controller tuning might use specifications like Integral of Absolute Error (IAE), Integral of Time multiplied Absolute Error (ITAE), Integral of Time multiplied Squared Error (ITSE), etc., which are calculated numerically [36]. These criteria are particularly suitable for use with computational intelligence-based optimization methods like Evolutionary and Swarm-based optimizations.

$$IAE = \int_0^\infty |e(t)|dt = \int_0^\infty |r(t)-y(t)|dt \tag{3.17}$$

$$ITAE = \int_0^\infty t|e(t)|dt \tag{3.18}$$

$$ITSE = \int_0^\infty te^2(t)dt \tag{3.19}$$

where $r(t)$ is the desired set-point of the system, and $y(t)$ is the actual output of the system. These criteria are related to the set-point tracking performance of the system.

- In Equation (3.17), the penalty applies only to the absolute error.

- In Equation (3.18) and Equation (3.19), the penalty also includes a time-dependent term.
- As time increases, the penalty grows larger, encouraging faster error correction.

Some other specifications for FO controller tuning, which can be numerically calculated, are as follows [37]:

- The values of the gain crossover frequency ω_{gc} and phase margin ϕ_m.
- Phase flatness at ω_{gc}, i.e.,

$$\left.\frac{d\angle G(s)}{ds}\right|_{s=j\omega_{gc}} = 0. \tag{3.20}$$

- High-frequency noise rejection,

$$\frac{C(j\omega)G(j\omega)}{1+C(j\omega)G(j\omega)} \leq H, \quad \forall\omega \geq \omega_h. \tag{3.21}$$

- Output disturbance rejection at low frequencies,

$$\frac{1}{1+C(j\omega)G(j\omega)} \leq N, \quad \forall\omega \leq \omega_l. \tag{3.22}$$

3.4.2 FOPID Control for the TRMS

For our TRMS system, the outputs of the fractional-order FOPID controllers are described, for each subsystem, by the following control laws:

$$u_v(t) = K_{p_v} e_v(t) + K_{i_v} D_t^{-\lambda_v} e_v(t) + K_{d_v} D_t^{\mu_v} e_v(t), \quad e_v(t) = \alpha_{vr}(t) - \alpha_v(t). \tag{3.23}$$

$$u_h(t) = k_{p_h} e_h(t) + k_{i_h} D_t^{-\lambda_h} e_h(t) + k_{d_h} D_t^{\mu_h} e_h(t), \quad e_h(t) = \alpha_{hr}(t) - \alpha_h(t). \tag{3.24}$$

As in the PID control case, with four additional parameters to estimate, the parameter vector becomes:

$$K = \left[k_{p_h}, k_{i_h}, k_{d_h}, k_{p_v}, k_{i_v}, k_{d_v}, \lambda_h, \lambda_v, \mu_h, \mu_v\right].$$

The search space is limited by:

$$0 \leq K_{P_h}, K_{i_h}, K_{d_h}, K_{p_v}, K_{i_v}, K_{d_v} \leq 15,$$

$$0 \leq \lambda_h, \lambda_v, \mu_h, \mu_v \leq 2.$$

The number of particles, the number of iterations, and the cost function remain unchanged. Furthermore, based on the results obtained for PID control.

The gains estimated by the PSO algorithm are presented in the following table (Table 3.2).

Table 3.2: PSO-Optimized FOPID Gains

Parameter	**Vertical Subsystem**	**Horizontal Subsystem**
K_p	7.4353	0.897
K_i	1.7997	0.1
K_d	3.515	0.6121
λ	0.81822	0.2
μ	1.4147	0.99031

3.4.3 Simulation Results

This section presents the results obtained from the implementation of PSO-optimized FOPID controllers for the Twin Rotor MIMO System (TRMS). A series of experiments were conducted to evaluate the performance of the proposed control strategy in terms of tracking accuracy, stability, and robustness under various operating conditions and disturbances. Figures(3.12) and (3.13) present the closed-loop simulation results of the optimized FOPID control for the TRMS.

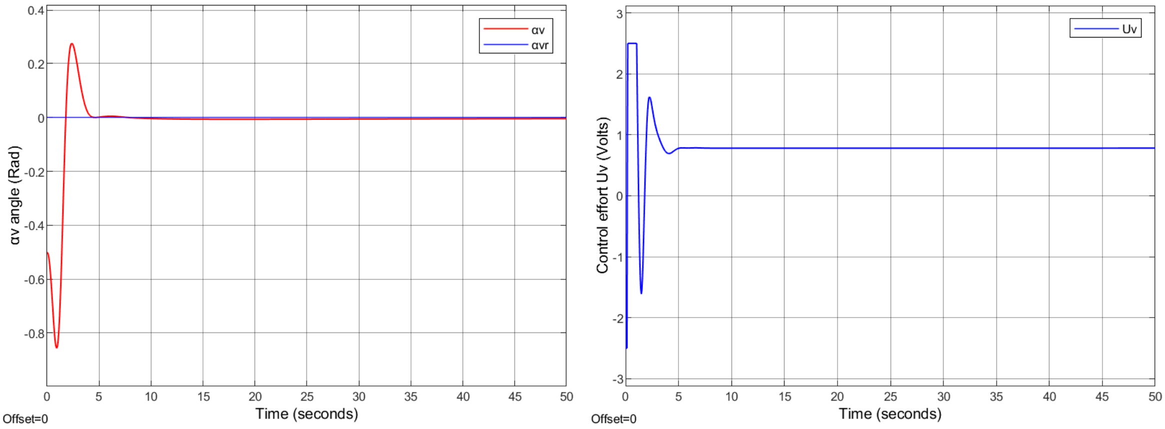


Figure 3.10: simulation results of the optimized FOPID control for the vertical subsystem

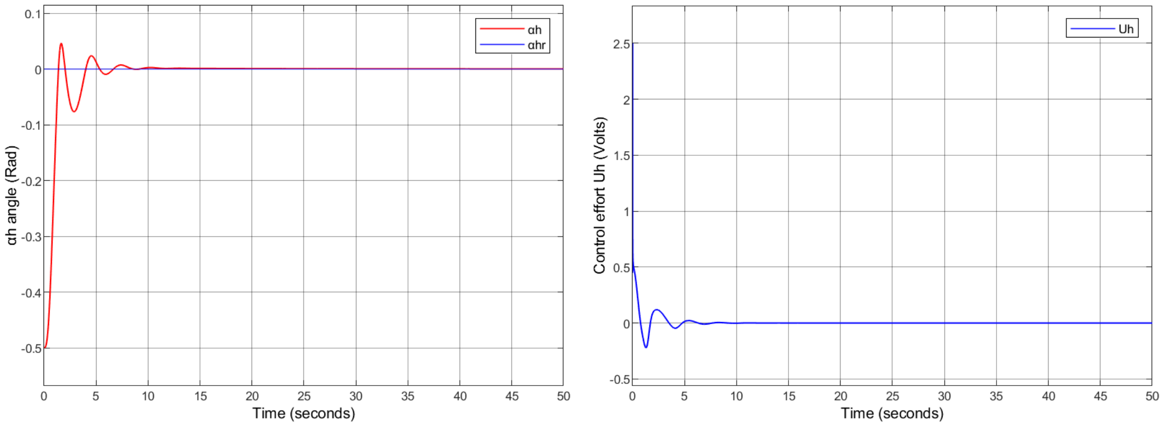


Figure 3.11: simulation results of the optimized FOPID control for the horizontal subsystem

Observations from the simulation Results Using FOPID Control

Vertical Subsystem Response (α_v):

- The system starts with an initial condition of $\alpha_v = -0.5$ radians and quickly reaches the desired setpoint of zero (rad).
- The overshoot is significantly reduced compared to PID control, indicating improved damping characteristics.
- The transient response is smoother, with faster settling time and minimal oscillations.

Horizontal Subsystem Response (α_h):

- The system starts at $\alpha_h = -0.5$ radians and rapidly converges to zero (rad).
- Overshoot is almost negligible, with a very smooth transition towards the reference.
- The response demonstrates higher accuracy and reduced steady-state error, confirming the efficiency of FOPID tuning.

Control Effort and System Stability:

- The control efforts U_v and U_h remain well within the saturation limits ($-2.5 \leq U_v, U_h \leq 2.5$), preventing excessive actuator strain.
- The required control input is less aggressive, leading to energy-efficient operation.
- FOPID reduces high-frequency oscillations, leading to smoother and more stable control action.

3.4.4 Robustness Testing of the FOPID Control

To test the robustness of our FOPID controller, we apply the control strategy obtained in the modified systems defined earlier.

Changing the position of the counterweight

We modified the position of the counterweight by 7 cm. The results are shown in Figures 3.12 and 3.13)

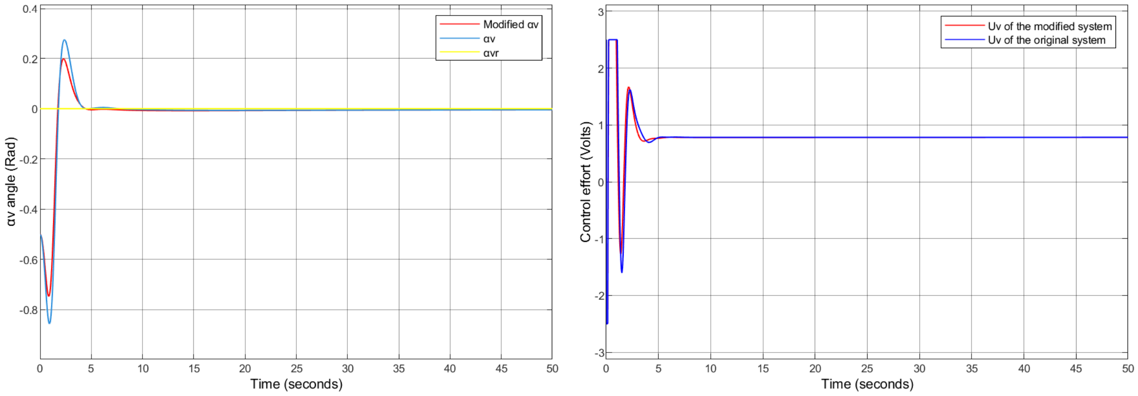


Figure 3.12: Modified Vertical Subsystem (lcb) Stabilization Using FOPID

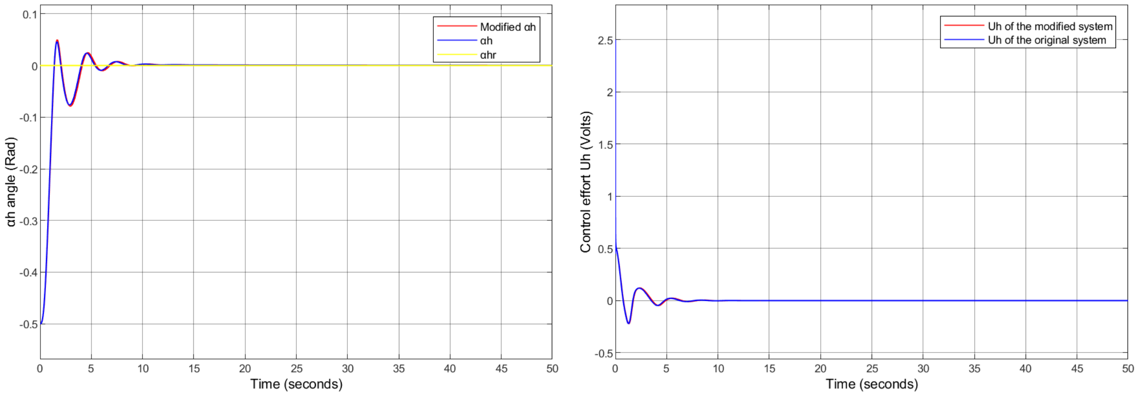


Figure 3.13: Modified Horizontal Subsystem (lcb) Stabilization Using FOPID

Changing the mass of the counterweight

We modified the mass of the counterweight by adding 20g. The results are shown in Figures (3.14 and 3.15)

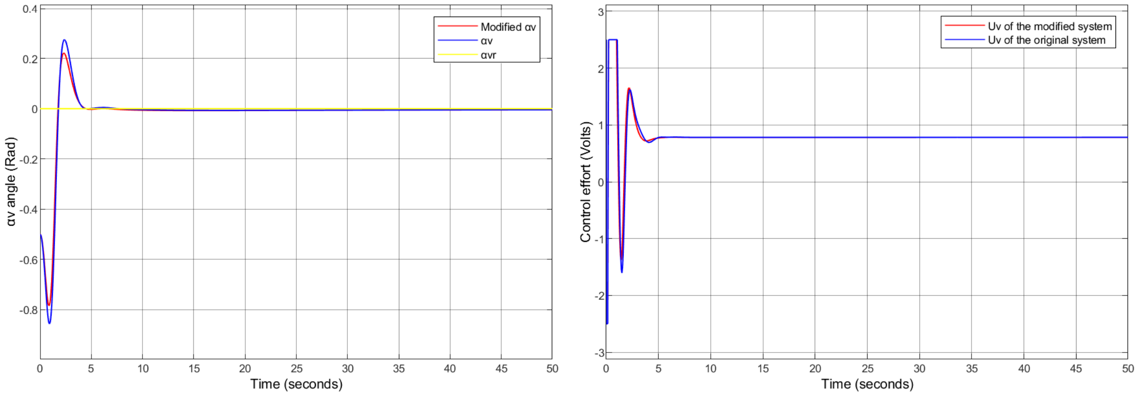


Figure 3.14: Modified Vertical Subsystem (mcb) Stabilization Using FOPID

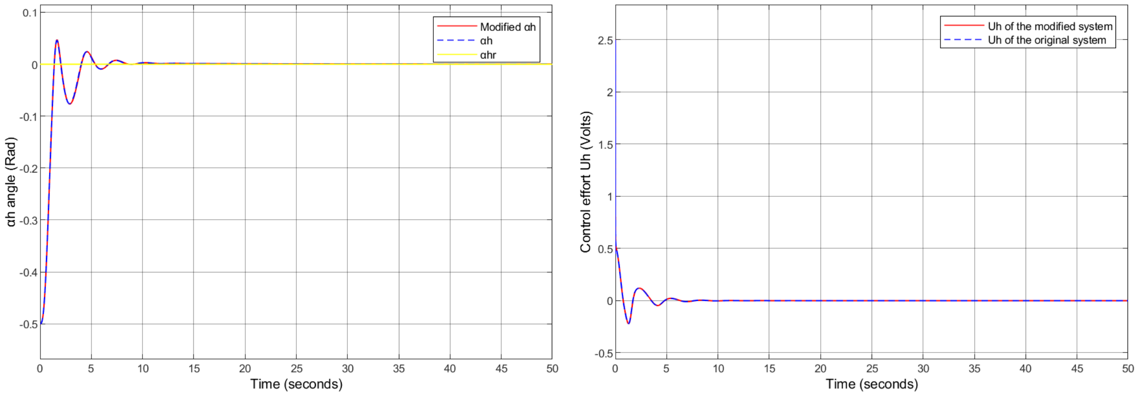


Figure 3.15: Modified Horizontal Subsystem (mcb) Stabilization Using FOPID

Observations on the Robustness Test

During the robustness tests, we first altered the position of the counterweight, then modified its mass. In both scenarios, we observed similar results, highlighting the robustness of the control strategy.

Vertical Subsystem Response

- The vertical angle (α_v) of the modified system also converges to zero radian.
- This confirms the controller's ability to adapt to changes in the system dynamics.

Horizontal Subsystem Response

- The horizontal angle (α_h) of the modified system exhibits similar performance, maintaining stability despite the modifications.
- The response features minimal oscillations and a fast convergence to zero radian.

3.5 Comparative Study Of PID and FOPID Control

In this section, we will conduct a comparison and qualitative analysis of the two controllers, PID and FOPID. The various simulations will cover several aspects, such as stabilization, reference tracking, and rejection of external disturbances.

This analysis will provide valuable insights into the advantages and limitations of each controller, helping to guide design and optimization decisions for the control system.

We begin our comparative study with a tracking test, where the input is changed to a sinusoidal signal of the form:

$$0.6\sin(0.2t).$$

3.5.1 Simulation Results

The following results are obtained (see Figure 5.7 for the vertical subsystem and Figure 5.8 for the horizontal subsystem):

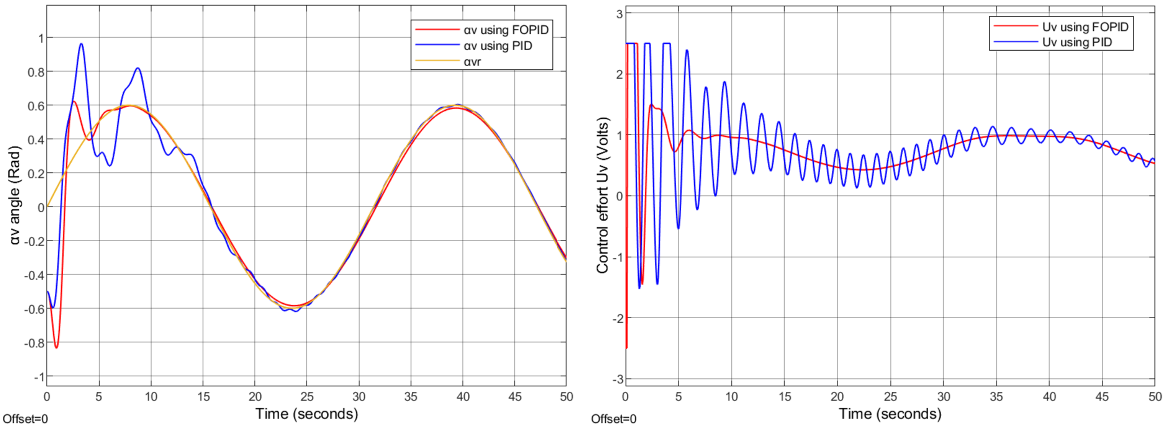


Figure 3.16: Response of the Vertical Subsystem for Both Controllers to a Sinusoidal Trajectory

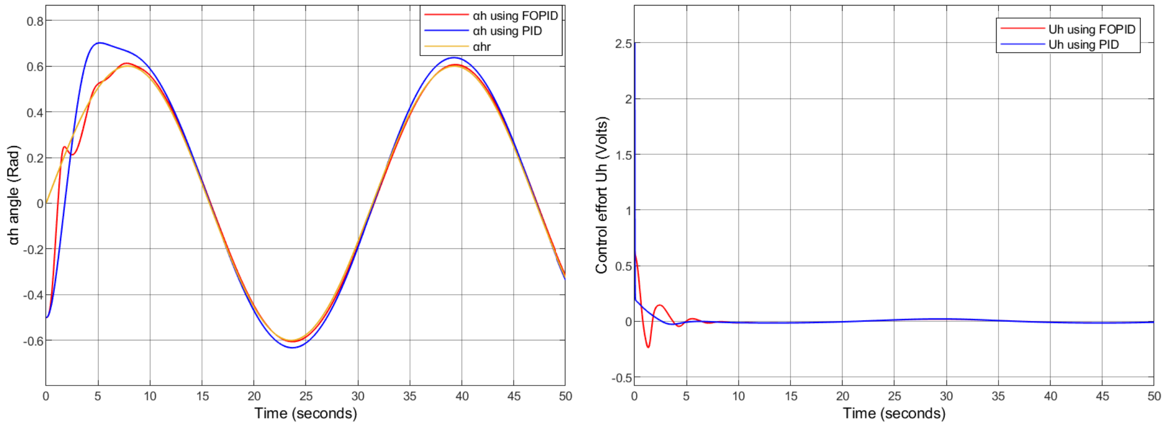


Figure 3.17: Response of the Horizontal Subsystem for Both Controllers to a Sinusoidal Trajectory

Interpretation of the Results

Vertical Subsystem Response (α_v)

- The FOPID controller exhibits better reference tracking, reducing initial oscillations compared to the PID controller.
- The PID controller presents a higher overshoot and transient oscillations, which can negatively impact system stability.
- Both controllers ultimately follow the sinusoidal reference signal, but FOPID achieves smoother tracking with reduced oscillatory behavior.

Horizontal Subsystem Response (α_h)

- The FOPID controller results in a smoother trajectory with minimal overshoot, while the PID controller introduces more aggressive oscillations.
- The PID controller lags slightly behind the reference during transitions, whereas FOPID provides a more precise and synchronized response.
- Overall, FOPID outperforms PID in terms of stability and reference tracking accuracy.

Control Effort Analysis

- The control effort (U_v and U_h) with the PID controller is more oscillatory, requiring higher energy consumption to maintain reference tracking.
- The FOPID controller provides a more stable and optimized control effort, reducing unnecessary fluctuations and improving efficiency.

3.5.2 disturbance rejection Analysis

To evaluate the disturbance rejection capability of both controllers, we introduce an external disturbance in the form of two impulse signals applied at $T = 0s$ and $T = 30s$ with an amplitude of 1. This disturbance simulates sudden external forces or unexpected system variations, as shown in Figure 3.18.

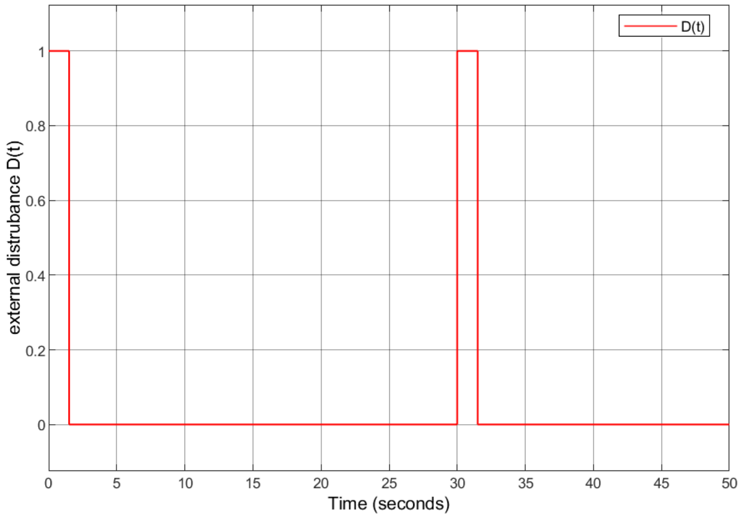


Figure 3.18: impulse disturbances at $T = 0s$ and $T = 30s$

Simulation Results

We get the following Results (figures 3.19 and 3.20)

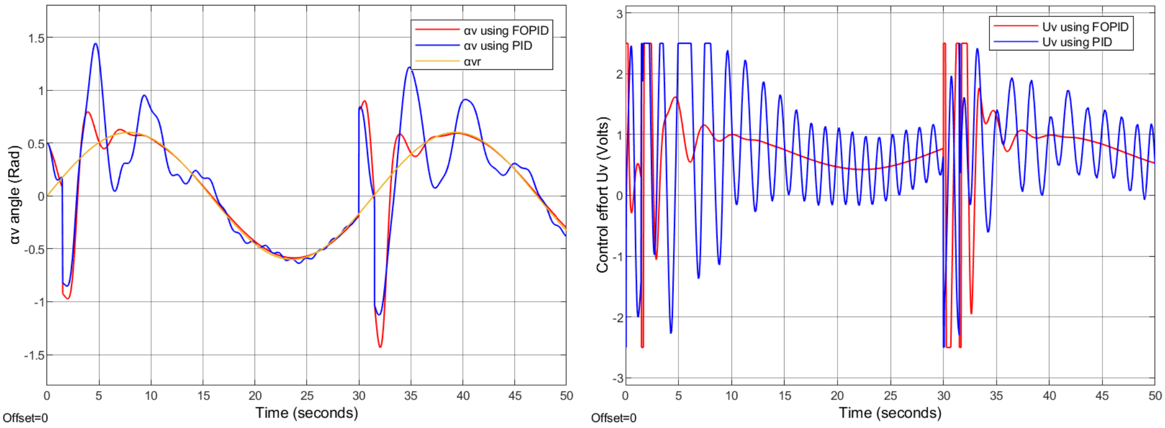


Figure 3.19: Response of the Vertical Subsystem for Both Controllers

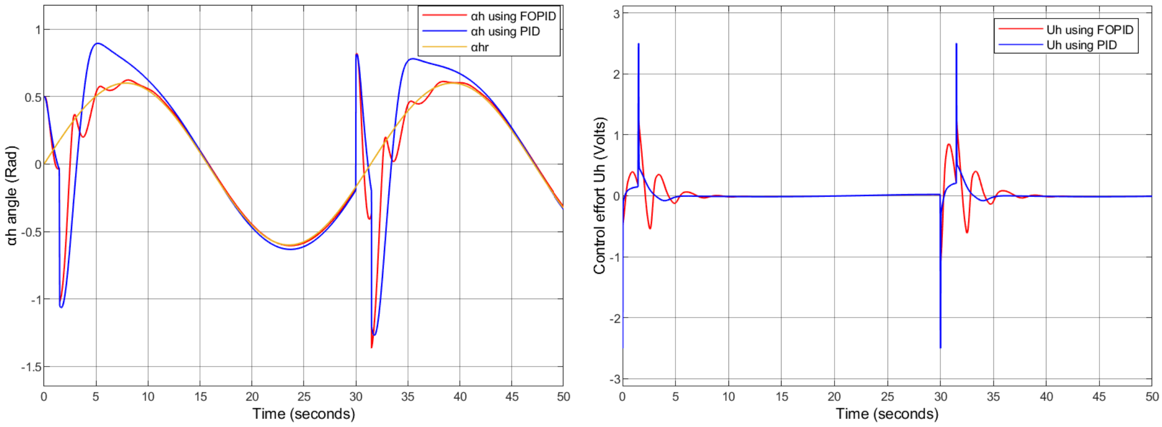


Figure 3.20: Response of the Horizontal Subsystem for Both Controllers

Discussion on Disturbance Rejection

Vertical Subsystem Response

- The vertical subsystem shows a significant transient deviation following the impulse disturbances.
- The FOPID controller exhibits improved damping, reducing oscillations and stabilizing faster compared to the PID controller.
- The PID controller produces higher amplitude oscillations, indicating a weaker disturbance rejection capability.

Horizontal Subsystem Response

- The horizontal subsystem is less affected by disturbances compared to the vertical subsystem.
- The FOPID controller allows for a smoother response with minimal overshoot and faster stabilization.
- The PID controller exhibits a more pronounced oscillatory behavior before stabilizing.

Control Effort Analysis

- The control effort under the FOPID controller is more stable and produces lower high-frequency oscillations compared to the PID controller.
- The PID controller generates a higher control effort with more fluctuations in response to disturbances.

These observations confirm that the FOPID controller provides better disturbance rejection and stability, especially in handling sudden impulse disturbances, compared to the conventional PID controller.

3.5.3 Applying the Controllers to the Coupled System

The optimized PID and FOPID controllers are now applied to the coupled Twin Rotor MIMO System. Unlike previous tests where each subsystem was considered independently, the coupling effects between the vertical and horizontal subsystems are now taken into account.

The coupling between the two subsystems is treated as an external disturbance affecting both dynamics. This allows us to evaluate the controllers ability to maintain stability, and ensure reference tracking.

The controllers are now tested with a sawtooth signal as the input, with an amplitude of 0.6 and a frequency of 0.15 rad/sec. This type of signal presents a new challenge, as it requires the system to track sharp variations while maintaining stability.

By introducing this non-smooth periodic input, we evaluate how well the PID and FOPID controllers adapt to sudden changes in direction. The goal is to observe their ability to minimize tracking errors, reduce overshoot.

Simulation Results

The following results are obtained (Figures 3.21 and 3.22):

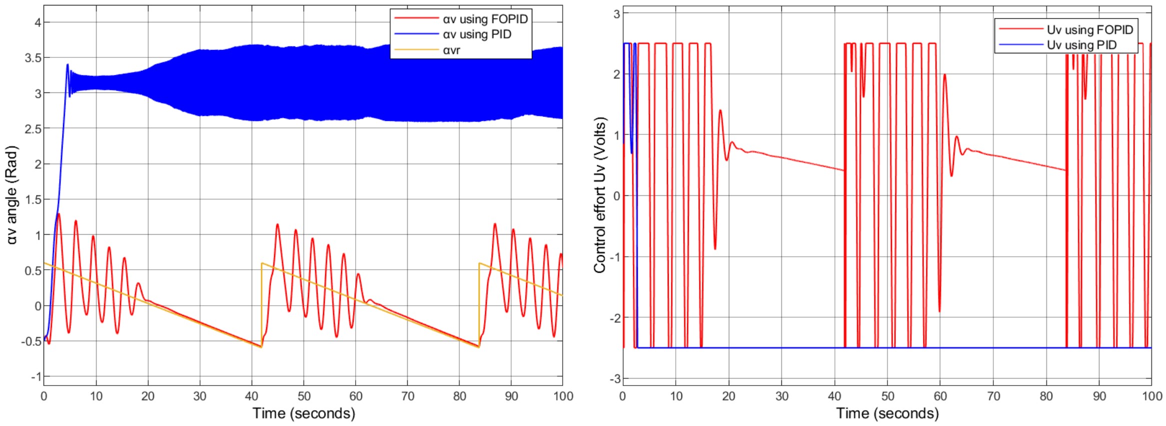


Figure 3.21: Vertical Response of the Coupled System: PID vs. FOPID

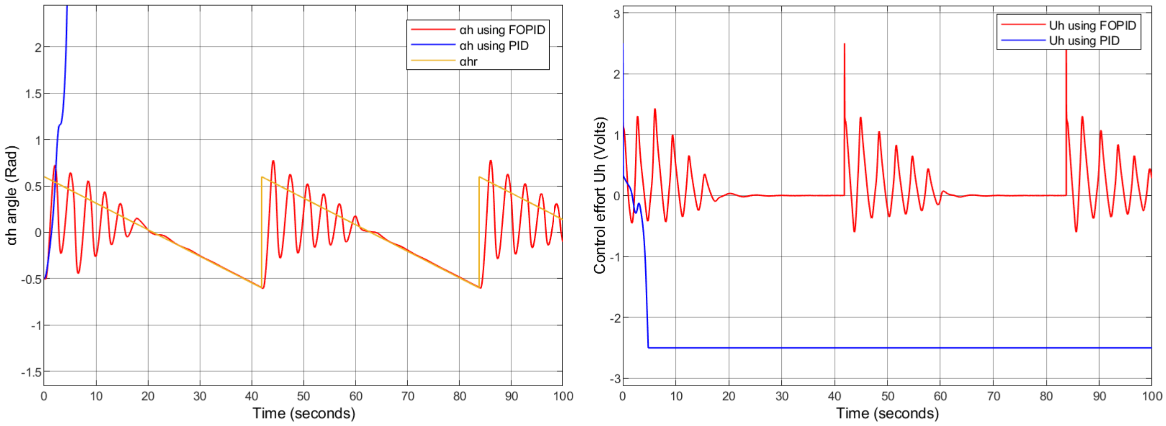


Figure 3.22: Horizontal Response of the Coupled System: PID vs. FOPID

Coupled System Response Analysis

Vertical Subsystem Response

1. **FOPID Controller**:
    - Accurately tracks the sawtooth reference signal (α_v) with minimal overshoot.
    - Maintains a smooth and stable response throughout the simulation.
    - Effectively rejects disturbances from the coupled horizontal motion.
    - Generates a low and steady control effort, indicating efficient energy usage.
2. **PID Controller**:
    - Fails to properly track the sawtooth reference signal, especially during abrupt changes.
    - Produces high-frequency oscillations in α_v, reducing performance.
    - Struggles with disturbance rejection due to the coupling effect.
    - Control effort is highly oscillatory and reaches saturation, risking actuator limitations.

Horizontal Subsystem Response

1. **FOPID Controller**:
    - Maintains stability in the horizontal angle α_h despite coupling.
    - Closely follows the reference trajectory with low steady-state error.
    - Exhibits good disturbance rejection and robustness.
    - Control effort remains low and smooth across the simulation.
2. **PID Controller**:
    - System becomes unstable with the horizontal angle diverging over time.
    - Cannot handle coupling disturbances, leading to continuous deviation from the reference.
    - Control effort is erratic and lacks regulation.
    - Performance degrades rapidly, indicating poor robustness.

These observations confirm that the FOPID controller significantly outperforms the PID controller in handling the coupled TRMS system, offering better stability, disturbance rejection, and tracking performance.

3.6 Conclusion

In this chapter, we developed and optimized PID and FOPID controllers for the Twin Rotor MIMO System using the Particle Swarm Optimization algorithm.

A detailed comparative analysis was carried out to evaluate both controllers based on the closed-loop behavior of the TRMS. The results clearly show that the FOPID controller delivers superior performance compared to the classical PID controller, offering better handling of nonlinearities and coupling, smoother responses, reduced overshoot, faster convergence, and stronger robustness against modeling errors and external disturbances.

Chapter 4

SMC and FOSMC Control Approaches for the Twin Rotor MIMO System

4.1 Introduction

Sliding Mode Control (SMC) is a nonlinear control technique known for its accuracy, robustness, and ease of tuning and implementation. The core idea behind SMC is to guide the system's states toward a specific surface in the state space, called the sliding surface. Once the system reaches this surface, the control law ensures that the states stay close to it. This process involves two main steps: first, designing a sliding surface that meets the desired performance goals, and second, creating a control law that ensures the system states are drawn to this surface [38], [39].

SMC has two key advantages. First, the system's behavior can be tailored by carefully choosing the sliding function. Second, the system becomes highly resistant to certain types of uncertainties, such as changes in model parameters, external disturbances, and nonlinearities. This makes SMC especially useful for controlling nonlinear systems that face external disturbances or significant model uncertainties.

To further improve performance, Fractional-Order Sliding Mode Control (FOSMC) introduces fractional calculus into the control design. This advanced approach builds on the strengths of SMC while offering better handling of complex dynamics and increased robustness, making it a valuable tool for modern control challenges.

This chapter introduces the fundamental concepts of Sliding Mode Control and its core principles. We begin by explaining the basics of SMC, including the design and selection of sliding surfaces. Next, we derive the control law using SMC and then apply Fractional-Order Sliding Mode Control to enhance the control design. Finally, we compare the simulation results of both methods, highlighting their differences and performance characteristics.

4.2 Fundamentals of Sliding-Mode Control Design

Sliding Mode control uses output feedback and a fast switching control action to steer the system's behavior. Ideally, this switching happens infinitely quickly, driving the system toward a specific subspace called the sliding surface or manifold. Once the system reaches this surface, the control law keeps it close to the manifold, ensuring the desired performance.[40]

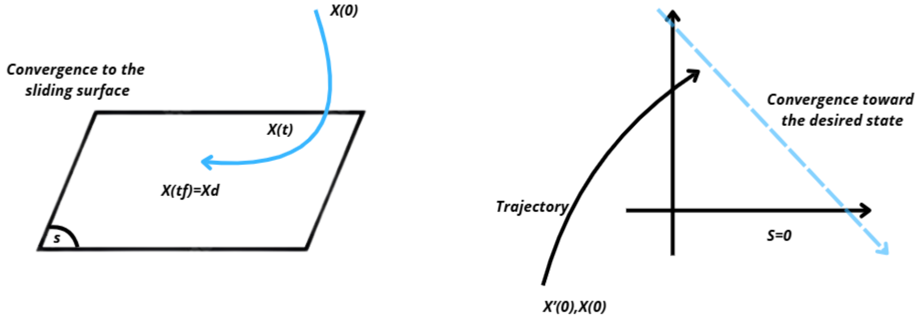


Figure 4.1: The fundamental principle of sliding mode

4.2.1 Design and Synthesis of Sliding Mode Control

Consider the nonlinear system:

$$\begin{cases} \dot{x} = f(t,x) + g(t,x)u \\ y = h(x) \end{cases} \tag{4.1}$$

where:

- x represents the state vector,
- u is the control input,
- $f(t,x)$ and $g(t,x)$ are nonlinear functions of the system dynamics,
- y is the system output.

Choice of the Sliding Surface

The design of the sliding surface is crucial and depends on the control objective. Slotine introduced a commonly used scalar sliding function [41] :

$$S(x) = \left(\frac{d}{dt} + \lambda\right)^{r-1} e(x) \tag{4.2}$$

where:

- $e(x) = x_d - x$ is the tracking error (difference between desired and actual state),
- λ is a positive constant influencing the convergence rate,
- r is the relative degree of the system.

This formulation ensures exponential convergence of the tracking error, meaning that:

$$\lim_{t \to +\infty} e = 0.$$

Once the system reaches $S(x) = 0$, it follows the desired trajectory.

Sliding Mode Existence Condition

To ensure that the system reaches and remains on the sliding surface, the attractivity condition must be satisfied [42] :

$$\begin{cases} \dot{S}(x) > 0 & \text{if } S(x) < 0, \\ \dot{S}(x) < 0 & \text{if } S(x) > 0. \end{cases} \tag{4.3}$$

This can be rewritten as a single condition:

$$S(x)\dot{S}(x) < 0. \tag{4.4}$$

This guarantees that the trajectory is attracted towards the sliding surface.

Control Law Design

The control input u must ensure that the system reaches and stays on the sliding surface, even in the presence of disturbances and uncertainties. The control law consists of two components :

$$u = u_{eq} + u_{att}. \tag{4.5}$$

- Equivalent Control (u_{eq}): Ensures that the system follows the desired dynamics once in sliding mode.
- Attracting Control (u_{att}): Forces the system towards the sliding surface.

Equivalent Control

The equivalent control is derived by imposing the invariance condition:

$$\begin{cases} S(x) = 0 \\ \dot{S}(x) = 0 \end{cases} \tag{4.6}$$

This leads to:

$$u_{eq} = -\left(\frac{\partial S}{\partial x} g(x,t)\right)^{-1} \left[\frac{\partial S}{\partial x} f(x,t)\right]. \tag{4.7}$$

Attracting Control

The second term ensures the attractivity condition:

$$u_{att} = -K \cdot \operatorname{sign}(S), \tag{4.8}$$

where K is a positive gain ensuring fast convergence.

Thus, the total control law is:

$$u = -\left(\frac{\partial S}{\partial x} g(x,t)\right)^{-1} \left[\frac{\partial S}{\partial x} f(x,t)\right] - K \cdot \operatorname{sign}(S). \tag{4.9}$$

4.2.2 The Chattering Phenomenon

A perfect sliding mode requires infinite switching frequency, which is impractical due to actuator limitations. In real applications, chattering occurs, leading to high-frequency oscillations around the sliding surface. This degrades performance and may cause instability, [42].

Causes of Chattering

- Time delays in measurement and control execution.
- High-frequency switching of the control signal.
- Modeling imperfections and actuator constraints.

Chattering Reduction Techniques

Several methods have been proposed to mitigate chattering while maintaining robustness:

- Boundary Layer Method (Saturation Function)

  Instead of using the discontinuous sign function, we use a continuous approximation:

$$\operatorname{sat}(S,\delta) = \begin{cases} \operatorname{sign}(S), & \text{if } |S| \geq \delta, \\ \frac{S}{\delta}, & \text{if } |S| < \delta. \end{cases} \tag{4.10}$$

  This allows smooth convergence to the sliding surface.

- Pseudo-Sign Function

Another smooth approximation is:

$$v(S,\delta) = \frac{S}{|S|+\delta}. \tag{4.11}$$

- Arctangent Function

  A more gradual transition can be achieved using:

$$v(S,\delta) = \frac{2}{\pi}\arctan\left(\frac{S}{\delta}\right). \tag{4.12}$$

- Hyperbolic Tangent Function

  This provides another smooth approximation:

$$v(S,\delta) = \tanh\left(\frac{S}{\delta}\right). \tag{4.13}$$

Each of these methods introduces a small trade-off between robustness and chattering reduction. The choice depends on the application and the desired balance between control accuracy and actuator constraints.

4.3 Sliding Mode Control of the TRMS

4.3.1 Selection of the Sliding Surface

The design of the sliding surface is a fundamental step in the development of a sliding mode control strategy for the Twin Rotor MIMO System [43]. We adopt Slotine's sliding surface, which is defined as:

$$S(x) = \left(\frac{d}{dt}+\lambda\right)^{r-1} e(x) \tag{4.14}$$

For the TRMS, we define two sliding surfaces for each subsystem:

$$\begin{cases} S_v = \left(\frac{d}{dt}+\lambda_v\right)^{r_v-1} e_v \\ S_h = \left(\frac{d}{dt}+\lambda_h\right)^{r_h-1} e_h \end{cases} \tag{4.15}$$

where r_v and r_h are the relative degrees of the vertical and horizontal subsystems, respectively.

Relative Degree of the Vertical Subsystem

The vertical subsystem is modeled by the state-space representation:

$$\begin{cases} \dot{x}_1 = A_v x_2 \\ \dot{x}_2 = F_v(x_3) - G_v(x_1) - B_v x_2 \\ \dot{x}_3 = -C_v x_3 + D_v u_v \\ y = x_1 \end{cases} \tag{4.16}$$

where:

$$A_v = \frac{1}{J_v}, \quad B_v = \frac{k_v}{J_v}, \quad C_v = \frac{1}{T_{mr}}, \quad D_v = \frac{k_{mr}}{T_{mr}}$$

And

$$F_v(x_3) = l_m S f F_v(P_v(x_3)), \quad G_v(x_1) = g((A-B)\cos(x_1) - C\sin(x_1)).$$

By differentiating y until the control term u_v appears:

$$\begin{cases} \dot{y} \quad = \quad \dot{x}_1 = A_v x_2 \\ \ddot{y} \quad = \quad A_v \dot{x}_2 = A_v(F_v(x_3) - G_v(x_1) - B_v x_2) \\ \dddot{y} \quad = \quad A_v \left(\frac{\partial F_v(x_3)}{\partial x_3} \dot{x}_3 - \frac{\partial G_v(x_1)}{\partial x_1} \dot{x}_1 - B_v \dot{x}_2 \right) \\ \quad = A_v \frac{\partial F_v(x_3)}{\partial x_3} (-C_v x_3 + D_v u_v) - A_v^2 \frac{\partial G_v(x_1)}{\partial x_1} x_2 - A_v B_v \dot{x}_2 \end{cases} \tag{4.17}$$

Since the control input u_v appears in the third derivative of y, the relative degree of the vertical subsystem is $r_v = 3$.

Relative Degree of the Horizontal Subsystem

Similarly, the horizontal subsystem is given by:

$$\begin{cases} \dot{x}_1 = A_h x_2 \\ \dot{x}_2 = F_h(x_3) - B_h x_2 \\ \dot{x}_3 = -C_h x_3 + D_h u_h \\ y = x_1 \end{cases} \tag{4.18}$$

where:

$$A_h = \frac{1}{J_h(\alpha_{v0})}, \quad B_h = \frac{k_h}{J_h(\alpha_{v0})}, \quad C_h = \frac{1}{T_{tr}}, \quad D_h = \frac{k_{tr}}{T_{tr}}, \quad F_h(x_3) = l_t S f F_h(P_h(x_3)).$$

Differentiating y until u_h appears:

$$\begin{cases} \dot{y} & = & \dot{x}_1 = A_h x_2 \\ \ddot{y} & = & A_h \dot{x}_2 = A_h(F_h(x_3) - B_h x_2) \\ \dddot{y} & = & A_h \left(\frac{\partial F_h(x_3)}{\partial x_3} \dot{x}_3 - B_h \dot{x}_2 \right) \\ & & = A_h \frac{\partial F_h(x_3)}{\partial x_3}(-C_h x_3 + D_h u_h) - A_h B_h \dot{x}_2 \end{cases} \tag{4.19}$$

Since u_h appears in the third derivative of y, the relative degree of the horizontal subsystem is $r_h = 3$.

Since both subsystems have a relative degree of 3, the sliding surfaces are:

$$\begin{cases} S_v = \ddot{e}_v + 2\lambda_v \dot{e}_v + \lambda_v^2 e_v \\ S_h = \ddot{e}_h + 2\lambda_h \dot{e}_h + \lambda_h^2 e_h \end{cases} \tag{4.20}$$

4.3.2 Control Law Calculation

Computation of u_v

The total control law consists of:

$$u_v = u_{v,eq} + u_{v,att}.$$

To derive the equivalent control $u_{v,eq}$, we start from:

$$\dot{S}_v = \lambda_v^2 \dot{e}_v + 2\lambda_v \ddot{e}_v + \dddot{e}_v = 0. \tag{4.21}$$

where the tracking error is defined as:

$$e_v = y - y_d = x_1 - x_1^d. \tag{4.22}$$

Expanding $\dot{S}_v$, we obtain:

$$\begin{aligned} \dot{S}_v &= \lambda_v^2 \dot{e}_v + 2\lambda_v \ddot{e}_v + \dddot{x}_1 - \dddot{x}_1^d \\ &= \lambda_v^2 \dot{e}_v + 2\lambda_v \ddot{e}_v - \dddot{x}_1^d + A_v \frac{\partial F_v(x_3)}{\partial x_3}(-C_v x_3 + D_v u_v) - A_v^2 \frac{\partial G_v(x_1)}{\partial x_1} x_2 - A_v B_v \dot{x}_2 \end{aligned} \tag{4.23}$$

To satisfy $\dot{S}_v = 0$, the equivalent control $u_{v,eq}$ is:

$$\begin{aligned} u_{v,eq} = &- \frac{1}{A_v D_v \frac{\partial F_v(x_3)}{\partial x_3}} \Bigg[\lambda_v^2 \dot{e}_v + 2\lambda_v \ddot{e}_v - \dddot{x}_1^d \\ &- A_v C_v \frac{\partial F_v(x_3)}{\partial x_3} x_3 - A_v^2 \frac{\partial G_v(x_1)}{\partial x_1} x_2 - A_v B_v \dot{x}_2 \Bigg]. \end{aligned} \tag{4.24}$$

To ensure the attractiveness condition $S_v \dot{S}_v < 0$, we add the term $u_{v,att}$, leading to the total control law:

$$\begin{aligned} u_v = -\frac{1}{A_v D_v \frac{\partial F_v(x_3)}{\partial x_3}} &\Bigg[\lambda_v^2 \dot{e}_v + 2\lambda_v \ddot{e}_v - \dddot{x}_1^d - A_v C_v \frac{\partial F_v(x_3)}{\partial x_3} x_3 \\ &- A_v^2 \frac{\partial G_v(x_1)}{\partial x_1} x_2 - A_v B_v \dot{x}_2 + K_v \operatorname{sign}(S_v) \Bigg]. \end{aligned} \tag{4.25}$$

By substituting this control law into $S_v \dot{S}_v$, we easily obtain:

$$S_v \dot{S}_v = -S_v K_v \operatorname{sign}(S_v) < 0,$$

Computation of u_h

Similar to the derivation of u_v, the control law for u_h is given by:

$$u_h = u_{h,eq} + u_{h,att}.$$

To determine $u_{h,eq}$, we start by defining the tracking error:

$$e_h = y - y_d = x_1 - x_1^d.$$

By differentiating S_h, we obtain:

$$\begin{aligned} \dot{S}_h &= \lambda_h^2 \dot{e}_h + 2\lambda_h \ddot{e}_h + \dddot{e}_h \\ &= \lambda_h^2 \dot{e}_h + 2\lambda_h \ddot{e}_h + \dddot{x}_1 - \dddot{x}_1^d \\ &= \lambda_h^2 \dot{e}_h + 2\lambda_h \ddot{e}_h - \dddot{x}_1^d + A_h \frac{\partial F_h(x_3)}{\partial x_3} (-C_h x_3 + D_h u_h) - A_h B_h \dot{x}_2 \end{aligned} \tag{4.26}$$

Setting $\dot{S}_h = 0$ leads to the equivalent control:

$$u_{h,eq} = -\frac{1}{A_h D_h \frac{\partial F_h(x_3)}{\partial x_3}} \left[\lambda_h^2 \dot{e}_h + 2\lambda_h \ddot{e}_h - \dddot{x}_1^d - A_h C_h \frac{\partial F_h(x_3)}{\partial x_3} x_3 - A_h B_h \dot{x}_2 \right]. \tag{4.27}$$

To ensure the attractivity condition $S_h\dot{S}_h < 0$, we add the term $u_{h,att}$ to the equivalent control $u_{h,eq}$. The total control law for u_h is then given by:

$$u_h = -\frac{1}{A_h D_h \frac{\partial F_h(x_3)}{\partial x_3}}\left[\lambda_h^2 \dot{e}_h + 2\lambda_h \ddot{e}_h - \dddot{x}_1^d - A_h C_h \frac{\partial F_h(x_3)}{\partial x_3} x_3 - A_h B_h \dot{x}_2 + K_h \operatorname{sign}(S_h)\right]. \quad (4.28)$$

The convergence condition can be verified as:

$$S_h\dot{S}_h = -S_h K_h \operatorname{sign}(S_h) < 0.$$

To avoid the chattering phenomenon, the sign function in the control law can be replaced with the hyperbolic tangent function. The updated control laws for u_v and u_h become:

$$\begin{cases} u_v = -\frac{1}{A_v D_v \frac{\partial F_v(x_3)}{\partial x_3}}\left[\lambda_v^2 \dot{e}_v + 2\lambda_v \ddot{e}_v - \dddot{x}_1^d - A_v C_v \frac{\partial F_v(x_3)}{\partial x_3} x_3 - A_v^2 \frac{\partial G_v(x_1)}{\partial x_1} x_2 - A_v B_v \dot{x}_2 + K_v \tanh(S_v)\right] \\ u_h = -\frac{1}{A_h D_h \frac{\partial F_h(x_3)}{\partial x_3}}\left[\lambda_h^2 \dot{e}_h + 2\lambda_h \ddot{e}_h - \dddot{x}_1^d - A_h C_h \frac{\partial F_h(x_3)}{\partial x_3} x_3 - A_h B_h \dot{x}_2 + K_h \tanh(S_h)\right] \end{cases}$$

4.3.3 Simulation Results

This section presents the simulation results of the Twin Rotor MIMO System under Sliding Mode Control (SMC). The objective is to stabilize the system around the equilibrium point $(\alpha_v, \alpha_h) = (0, 0)$ rad, starting from an initial condition of $(-0.5, -0.5)$ rad. To determine the optimal control parameters $[K_h, K_v, \lambda_h, \lambda_v]$, we employ the Particle Swarm Optimization (PSO) algorithm introduced in the previous chapter.

By applying PSO, the estimated values of the parameters are obtained and summarized in Table 4.1.

Table 4.1: Optimized SMC parameters using PSO

Parameter	K	λ
Vertical Subsystem	4.0127	2.4736
Horizontal Subsystem	1.685	1.8975

Initially, the sign function is incorporated into the attracting control term to examine the chattering phenomenon (figure 4.2), a common issue in sliding mode control. Once the chattering effect is observed and, the sign function is replaced with the tanh function to mitigate its impact while preserving control robustness (figure 4.3).

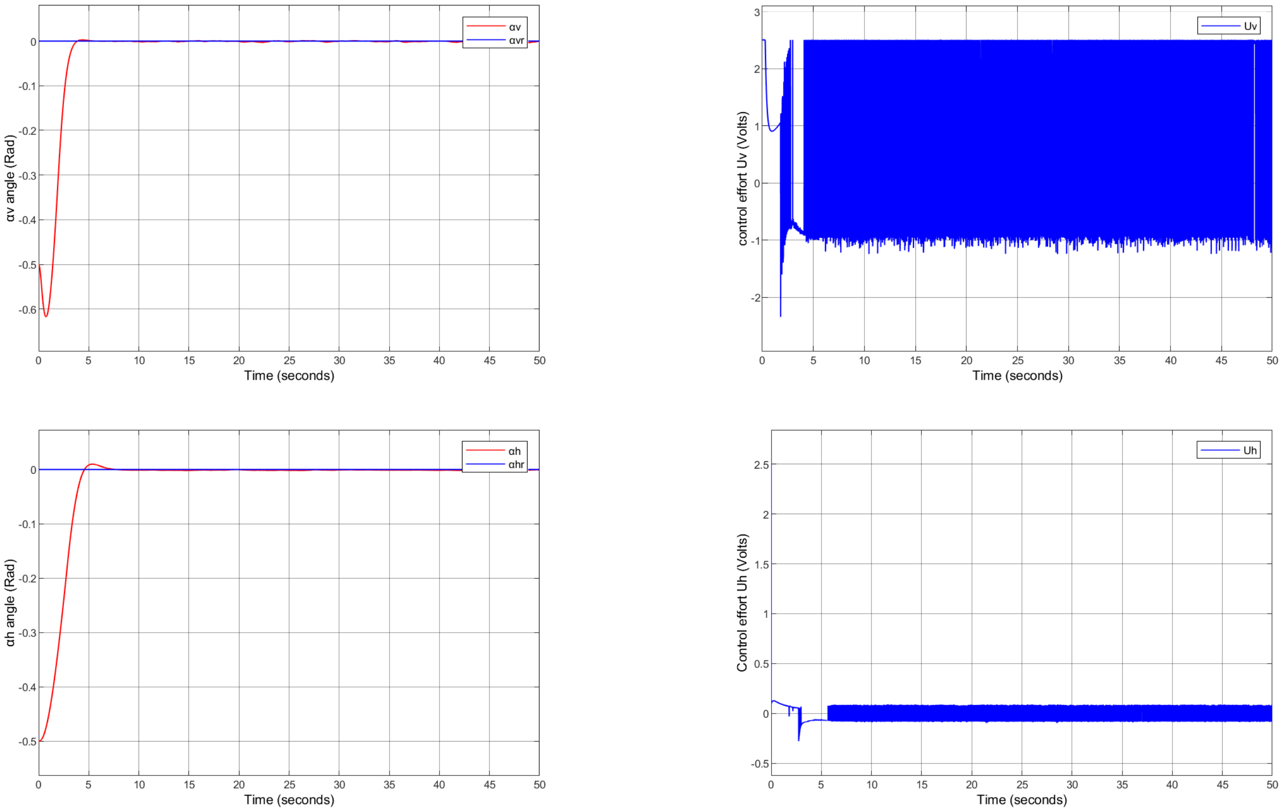


Figure 4.2: Simulation with an Attracting Function: $-K\,\text{sign}(S)$

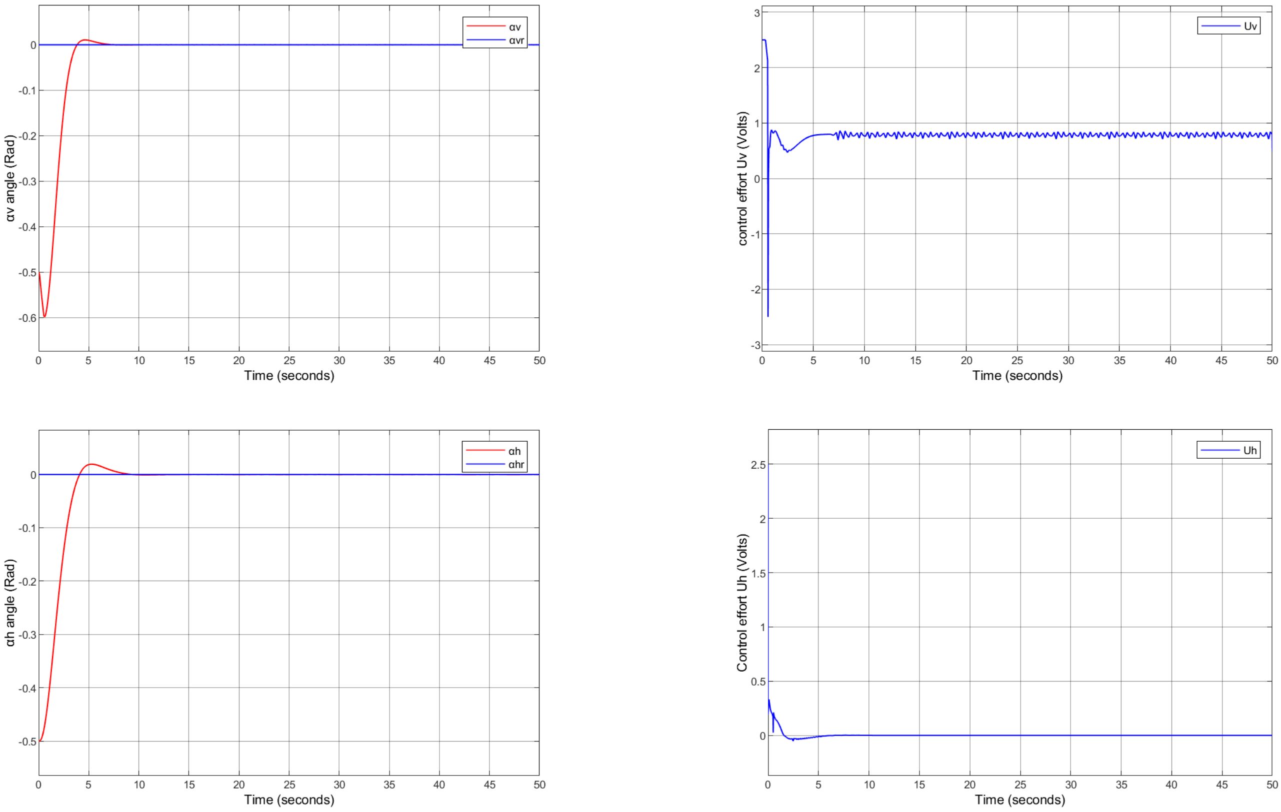


Figure 4.3: Simulation with an Attracting Function: $-K\tanh(S)$

Observations from the Results

From the simulation results, the following observations can be made:

Sign Function Results (figure 4.2):

- The system angles (α_v, α_h) successfully converge to their reference values.
- The control signals (u_v, u_h) exhibit high-frequency oscillations ($-2.5 \leq U_v, U_h \leq 2.5$).
- Chattering is clearly visible in the control efforts, leading to large variations in the control signal.

Tanh Function Results (figure 4.3):

- The system angles (α_v, α_h) still converge to the reference without deviation.
- The control signals (u_v, u_h) are significantly smoother compared to the sign function case.
- Chattering is greatly reduced, making the control signal more stable.

Comparison:

- The sign function introduces high-frequency oscillations in the control effort.
- The tanh function leads to a smoother control effort while maintaining system stability.

4.3.4 Robustness Testing of the SMC Control

To evaluate the robustness of the SMC control, we conducted tests by modifying key parameters of the Twin Rotor MIMO System. Specifically, **we changed the weight and the position** of the counterweight.

Changing the position of the counterweight

We modified the position of the counterweight by 7 cm. The results are shown in Figures (4.4 and 4.5)

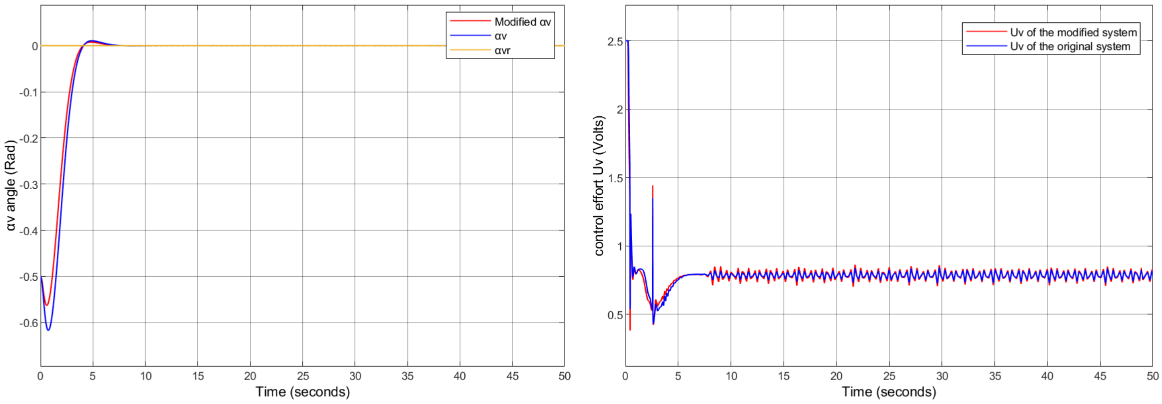


Figure 4.4: Modified Vertical Subsystem (lcb) Stabilization Using SMC

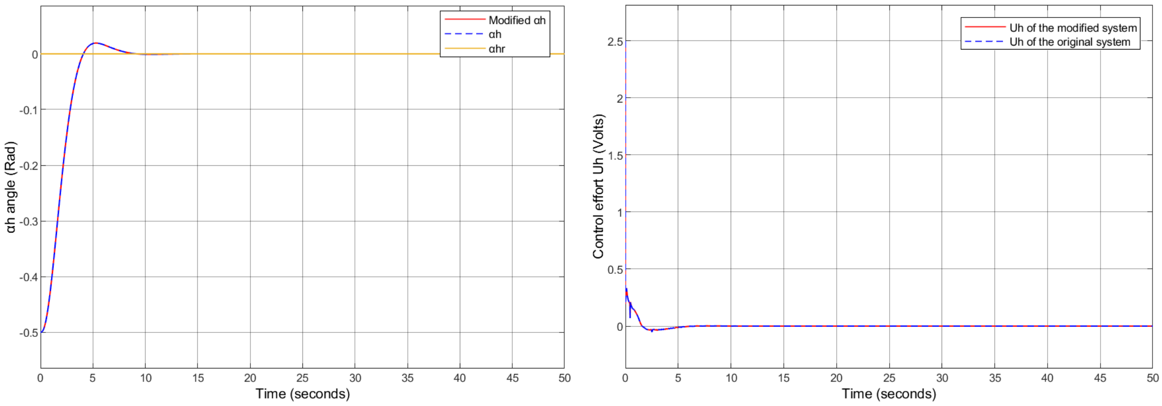


Figure 4.5: Modified Horizontal Subsystem (lcb) Stabilization Using SMC

Changing the mass of the counterweight

We modified the mass of the counterweight by adding 20g. The results are shown in Figures (4.6 and 4.7)

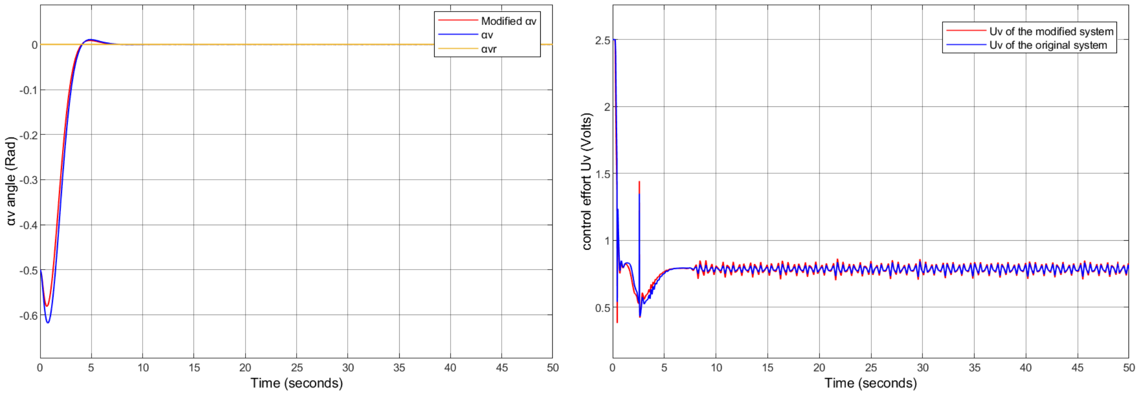


Figure 4.6: Modified Vertical Subsystem (mcb) Stabilization Using SMC

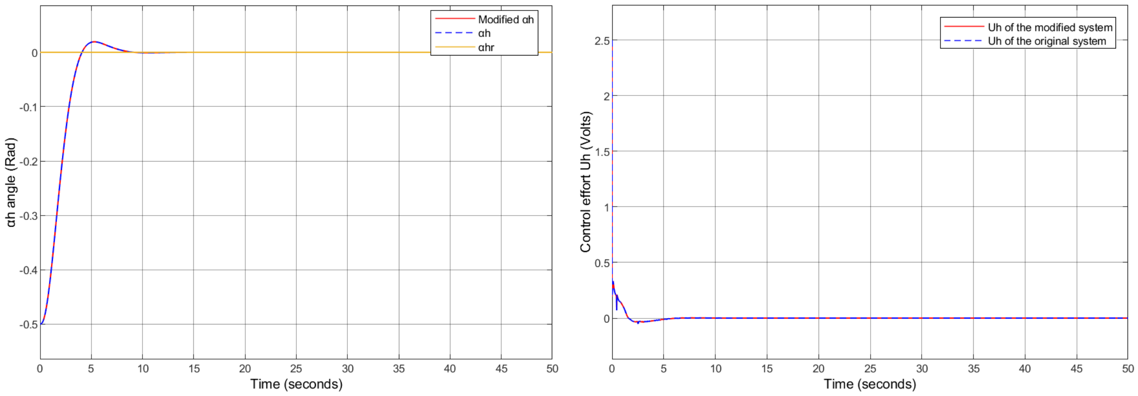


Figure 4.7: Modified Horizontal Subsystem (mcb) Stabilization Using SMC

Observations on the Robustness Test

During the robustness tests, we first altered the position of the counterweight, then modified its mass. In both scenarios, we observed similar results, highlighting the robustness of the control strategy.

Vertical Subsystem Response

- The vertical angle (α_v) of the modified system successfully converges to its reference value, and there is no significant difference between it and the vertical angle (α_v) of the original system.
- This confirms the controller's robustness and ability to adapt to variations in system parameters while maintaining stability.
- The control effort Uv shows nearly identical voltage profiles between the modified and original systems, indicating equivalent energy requirements for vertical stabilization.

Horizontal Subsystem Response

- The horizontal angles (α_h) are perfectly superposable, demonstrating identical dynamic responses and tracking performance in the horizontal orientation.
- The response remains stable, demonstrating the robustness of the control strategy despite the changes in system parameters.
- The control effort U_h shows no difference between the original system and the two modified systems.

4.4 Fractional-Order Sliding Mode Control for the TRMS System

To enhance the performance already achieved by the previous control strategy, Slotine's formula is extended using fractional-order derivatives [44]. This leads to the development of a Fractional-Order Sliding Mode Control (FOSMC), which builds upon conventional Sliding Mode Control

(SMC) by incorporating fractional calculus. This integration increases both the robustness and precision of the control system, particularly for the Twin Rotor MIMO System.

4.4.1 Fractional Sliding Surfaces Development

Building upon conventional sliding mode control, we develop fractional-order sliding surfaces for both TRMS axes using Slotine's formulation with fractional derivatives [44] :

$$\begin{cases} S_v = \left(D_t^{\alpha_1} + \lambda_v\right)^{r_v - 1} e_v \\ S_h = \left(D_t^{\alpha_2} + \lambda_h\right)^{r_h - 1} e_h \end{cases} \tag{4.29}$$

Given the relative degrees $r_v = r_h = 3$, these surfaces simplify to:

$$\begin{cases} S_v = D_t^{2\alpha_1} e_v + 2\lambda_v D_t^{\alpha_1} e_v + \lambda_v^2 e_v \\ S_h = D_t^{2\alpha_2} e_h + 2\lambda_h D_t^{\alpha_2} e_h + \lambda_h^2 e_h \end{cases} \tag{4.30}$$

Applying the switching property from Eq. (2.12), we obtain the implementable forms:

$$\begin{cases} S_v = D_t^{2(\alpha_1 - 1)} \ddot{e}_v + 2\lambda_v D_t^{\alpha_1 - 1} \dot{e}_v + \lambda_v^2 e_v \\ S_h = D_t^{2(\alpha_2 - 1)} \ddot{e}_h + 2\lambda_h D_t^{\alpha_2 - 1} \dot{e}_h + \lambda_h^2 e_h \end{cases} \tag{4.31}$$

4.4.2 Control Laws Derivation

Vertical Control Law (u_v)

The equivalent control is derived from $\dot{S}_v = 0$:

$$\dot{S}_v = D_t^{2(\alpha_1 - 1)} \dddot{e}_v + 2\lambda_v D_t^{\alpha_1 - 1} \ddot{e}_v + \lambda_v^2 \dot{e}_v = 0 \tag{4.32}$$

To solve the equation $\dot{S} = 0$ and determine the equivalent control $u_{v_{eq}}$, we multiply both sides of the equation by the fractional derivative operator $D_t^{2(1-\alpha_1)}$:

$$D_t^{2(1-\alpha_1)} \dot{S}_v = 0$$

Applying this operation to the original equation:

$$D_t^{2(1-\alpha_1)} \left[D_t^{2(\alpha_1 - 1)} \dddot{e}_v + 2\lambda_v D_t^{\alpha_1 - 1} \ddot{e}_v + \lambda_v^2 \dot{e}_v \right] = 0 \tag{4.33}$$

Using the semi-group property of fractional derivatives (2.11):

$$D_t^{2(1-\alpha_1)} D_t^{2(\alpha_1 - 1)} = D_t^0 = \text{Identity operator}$$

This simplifies to:

$$A_v \frac{\partial F_v(x_3)}{\partial x_3}\Big[-C_v x_3 + D_v u_{v_{eq}}\Big] - A_v^2 \frac{\partial G_v(x_1)}{\partial x_1} x_2 - A_v B_v \dot{x}_2 - \dddot{x}_{1d} + 2\lambda_v D_t^{1-\alpha_1} \ddot{e}_v + \lambda_v^2 D_t^{2(1-\alpha_1)} \dot{e}_v = 0 \tag{4.34}$$

Equivalent Control Solution

Solving for $u_{v_{eq}}$ yields:

$$u_{v_{eq}} = \frac{-1}{A_v D_v \frac{\partial F_v}{\partial x_3}} \Bigg[- C_v A_v \frac{\partial F_v}{\partial x_3} x_3 - A_v^2 \frac{\partial G_v}{\partial x_1} x_2 - A_v B_v \dot{x}_2 - \dddot{x}_{1d} + 2\lambda_v D_t^{1-\alpha_1} \ddot{e}_v + \lambda_v^2 D_t^{2(1-\alpha_1)} \dot{e}_v \Bigg] \tag{4.35}$$

Total Control Law with Attractive Term

To ensure convergence to the sliding surface S_v, the Lyapunov stability condition must be satisfied:

$$S_v \dot{S}_v < 0$$

The total control input u_v is designed as:

$$u_v = \frac{-1}{A_v D_v \dfrac{\partial F_v(x_3)}{\partial x_3}} \Big[- C_v A_v \frac{\partial F_v(x_3)}{\partial x_3} x_3 - A_v^2 \frac{\partial G_v(x_1)}{\partial x_1} x_2 - A_v B_v \dot{x}_2 - \dddot{x}_{1d} \\ + 2\lambda_v D_t^{1-\alpha_1} \ddot{e}_v + \lambda_v^2 D_t^{2(1-\alpha_1)} \dot{e}_v + K_v D_t^{2(1-\alpha_1)} \tanh(S_v) \Big] \tag{4.36}$$

This control law always satisfies the convergence condition:

$$S_v \dot{S}_v = -S_v K_v \tanh(S_v) < 0$$

Horizontal Control Law (u_h)

Similarly, the horizontal control is derived as: The equivalent control is derived from The equivalent control is derived from $\dot{S}_h = 0$:

$$\dot{S}_h = D_t^{2(\alpha_2 - 1)} \dddot{e}_h + 2\lambda_h D_t^{\alpha_2 - 1} \ddot{e}_h + \lambda_h^2 \dot{e}_h = 0 \tag{4.37}$$

To solve $\dot{S}_h = 0$ and determine $u_{h_{eq}}$, we multiply both sides by $D_t^{2(1-\alpha_2)}$:

$$D_t^{2(1-\alpha_2)} \dot{S}_h = 0$$

Applying the operation to the original equation:

$$D_t^{2(1-\alpha_2)} \left[D_t^{2(\alpha_2 - 1)} \dddot{e}_h + 2\lambda_h D_t^{\alpha_2 - 1} \ddot{e}_h + \lambda_h^2 \dot{e}_h \right] = 0 \tag{4.38}$$

Using the semi-group property of fractional derivatives ((2.11)):

$$D_t^{2(1-\alpha_2)} D_t^{2(\alpha_2 - 1)} = D_t^0 = \text{Identity operator}$$

This simplifies to:

$$A_h \frac{\partial F_h(x_3)}{\partial x_3} \left[-C_h x_3 + D_h u_{h_{eq}} \right] - A_h B_h \dot{x}_2 - \dddot{x}_{1d} + 2\lambda_h D_t^{1-\alpha_2} \ddot{e}_h + \lambda_h^2 D_t^{2(1-\alpha_2)} \dot{e}_h = 0 \tag{4.39}$$

Equivalent Control Solution

Solving for $u_{h_{eq}}$ yields:

$$u_{h_{eq}} = \frac{-1}{A_h D_h \dfrac{\partial F_h(x_3)}{\partial x_3}} \Bigg[- C_h A_h \frac{\partial F_h(x_3)}{\partial x_3} x_3 - A_h B_h \dot{x}_2 - \dddot{x}_{1d} + 2\lambda_h D_t^{1-\alpha_2} \ddot{e}_h + \lambda_h^2 D_t^{2(1-\alpha_2)} \dot{e}_h \Bigg] \tag{4.40}$$

Total Control Law with Attractive Term

We must finally verify the convergence condition:

$$S_h \dot{S}_h < 0$$

To do so, we define the total control u_h as follows:

$$u_h = \frac{-1}{A_h D_h \dfrac{\partial F_h(x_3)}{\partial x_3}} \Bigg[- C_h A_h \frac{\partial F_h(x_3)}{\partial x_3} x_3 - A_h B_h \dot{x}_2 - x_{1d} + 2\lambda_h D_t^{1-\alpha_2} \ddot{e}_h + \lambda_h^2 D_t^{2(1-\alpha_2)} \dot{e}_h + K_h D_t^{2(1-\alpha_2)} \tanh(S_h) \Bigg] \tag{4.41}$$

This control law always satisfies the convergence condition:

$$S_h \dot{S}_h = -S_h K_h \tanh(S_h) < 0$$

4.4.3 Simulation Results

This section presents the simulation results of the Twin Rotor MIMO System under **Fractional Order Sliding Mode Control**.

The control objective is to stabilize the system around the equilibrium point $(\alpha_v, \alpha_h) = (0, 0)$ rad, starting from an initial condition of $(-0.5, -0.5)$ rad.

The sliding surface dynamics are defined by fractional orders α_1 and α_2. The control law incorporates these fractional orders to enhance robustness and reduce chattering effects. To determine the optimal control parameters $[K_h, K_v, \lambda_h, \lambda_v, \alpha_1, \alpha_2]$, we use the Particle Swarm Optimization (PSO) algorithm, as introduced in the previous chapter.

By applying PSO, the optimal values of the FOSMC parameters are obtained and summarized in Table 4.2.

Table 4.2: Optimized FOSMC parameters

Parameter	K	λ	α
Vertical Subsystem	3.6852	2.4806	0.9624
Horizontal Subsystem	2.013	2.0263	0.9501

Figures(4.8) and (4.9) present the closed-loop simulation results of the FOSMC control for the TRMS.

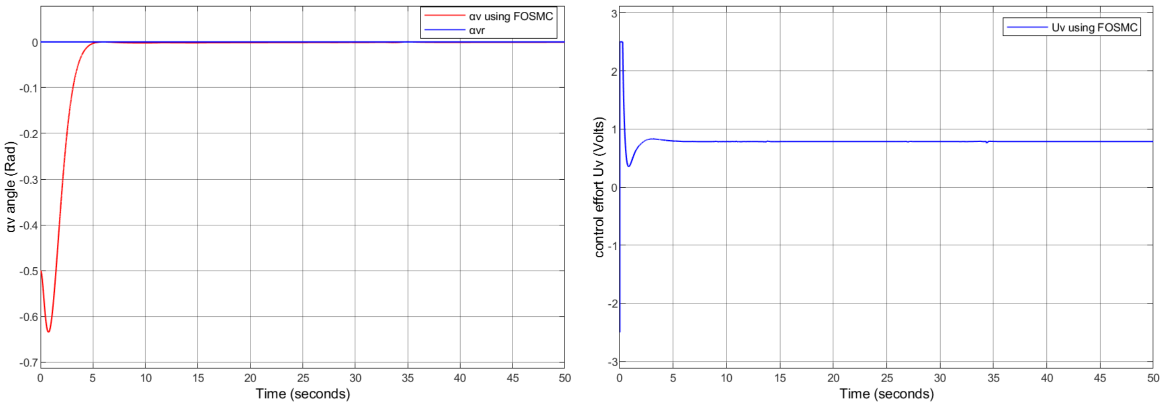


Figure 4.8: simulation results of FOSMC control for the vertical subsystem

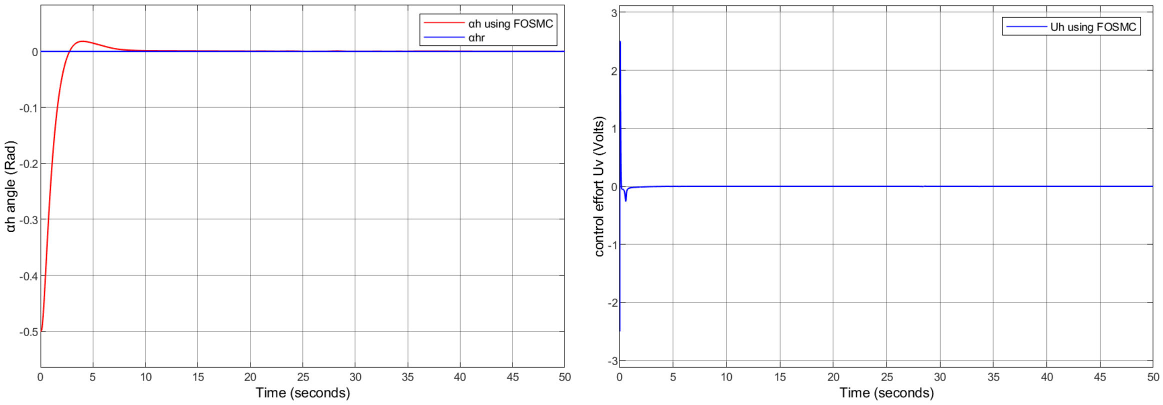


Figure 4.9: simulation results of FOSMC control for the horizontal subsystem

Observations from Simulation Results

Vertical Subsystem Response (α_v)

- The system starts from an initial condition of $\alpha_v = -0.5$ radians and quickly converges to the desired setpoint at 0 radians.
- The overshoot is negligible, reflecting enhanced damping and precision in tracking.
- The transient response is smooth, with a fast settling time and no noticeable oscillations.
- The control effort u_v is bounded and smooth, with chattering effectively eliminated.

Horizontal Subsystem Response (α_h)

- Starting from $\alpha_h = -0.5$ radians, the system rapidly reaches the equilibrium point with a clean, monotonic trajectory.
- The overshoot is very low, and the system shows excellent stability and accuracy.
- The settling time is short, and the response is free from abrupt variations.
- The control input u_h is smooth and continuous.

4.4.4 Robustness Testing of the FOSMC Control

To evaluate the robustness of the FOSMC control, we conducted tests by modifying key parameters of the Twin Rotor MIMO System (TRMS). Specifically, **we changed the weight and the position** of the counterweight.

Changing the position of the counterweight

We modified the position of the counterweight by 7 cm. The results are shown in Figures (4.10 and 4.11)

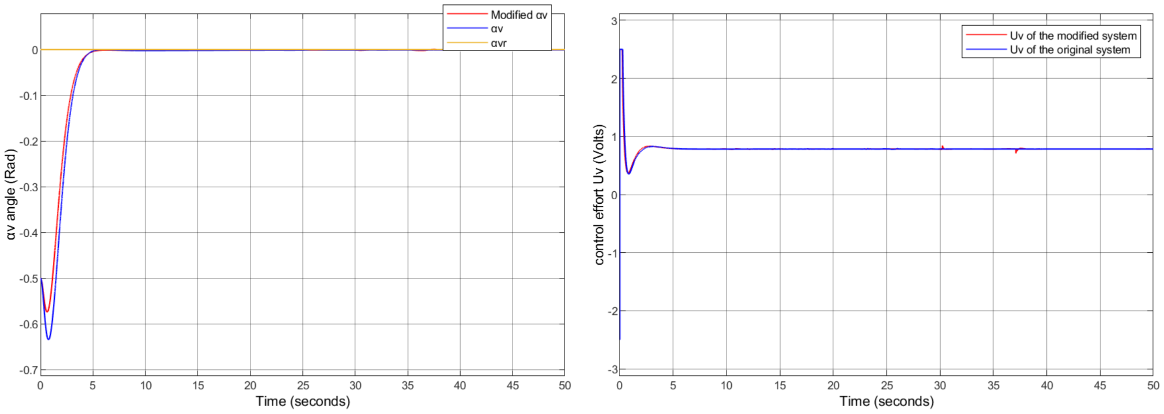


Figure 4.10: Modified Vertical Subsystem (lcb) Stabilization Using FOSMC

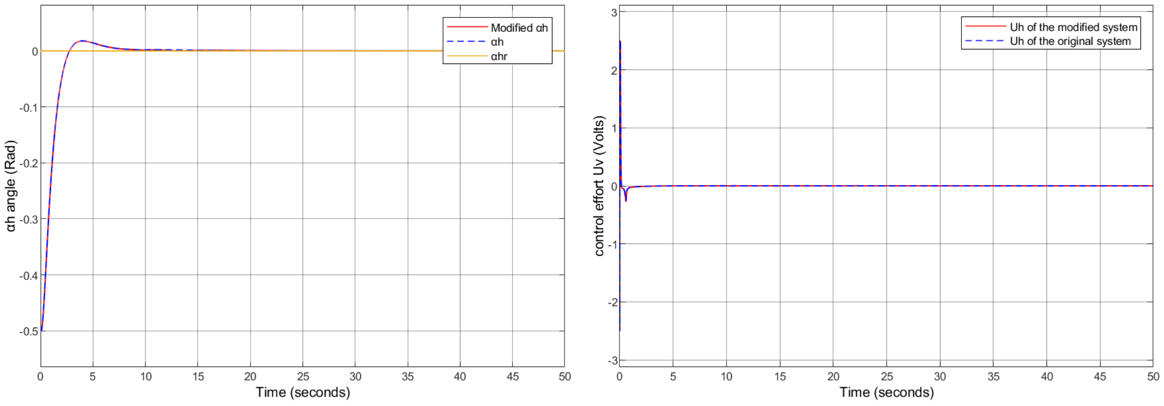


Figure 4.11: Modified Horizontal Subsystem (lcb) Stabilization Using FOSMC

Changing the mass of the counterweight

We modified the mass of the counterweight by adding 20g. The results are shown in Figures (4.12 and 4.13)

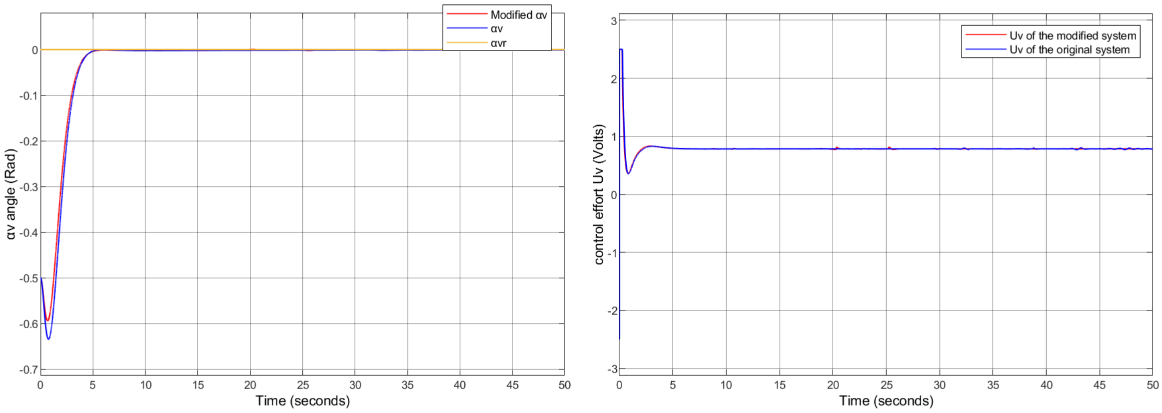


Figure 4.12: Modified Vertical Subsystem (mcb) Stabilization Using FOSMC

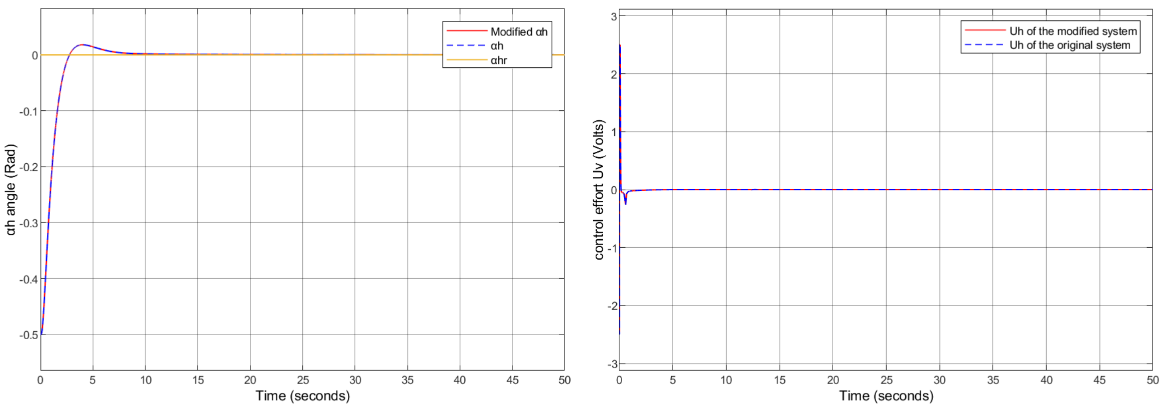


Figure 4.13: Modified Horizontal Subsystem (mcb) Stabilization Using FOSMC

Observations on the Robustness Tests

During the robustness tests, we first altered the position of the counterweight, then modified its mass. In both scenarios, we observed similar results, highlighting the resilience of the control strategy.

Vertical Subsystem Response

- Both modified systems show successful convergence of α_v to reference values
- No significant difference observed versus the original system's α_v response
- The control effort Uv shows nearly identical voltage profiles between the modified and original systems, indicating equivalent energy requirements for vertical stabilization.

Horizontal Subsystem Response

- Horizontal angles (α_h) are perfectly superposable across all systems
- Identical dynamic responses in original and modified configurations
- Control signals U_h show no measurable differences

4.5 Comparative Study Of SMC and FOSMC Control

In this section, we will conduct a comparison and qualitative analysis of the two controllers, SMC and FOSMC. The various simulations will cover several aspects, such as reference tracking, and rejection of external disturbances.

We begin our comparative study with a tracking test, where the input is changed to a square signal of the form:

$$0.4 \cdot \text{square}(0.2t)$$

4.5.1 Simulation Results

The following results are obtained (Figure 4.14 for the vertical subsystem and Figure 4.15 for the horizontal subsystem):

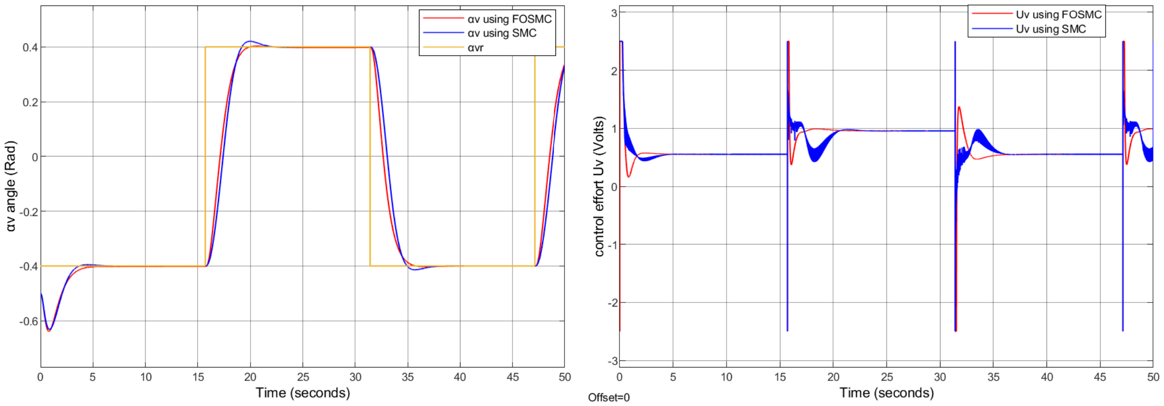


Figure 4.14: Response of the Vertical Subsystem for Both Controllers to a Square Trajectory

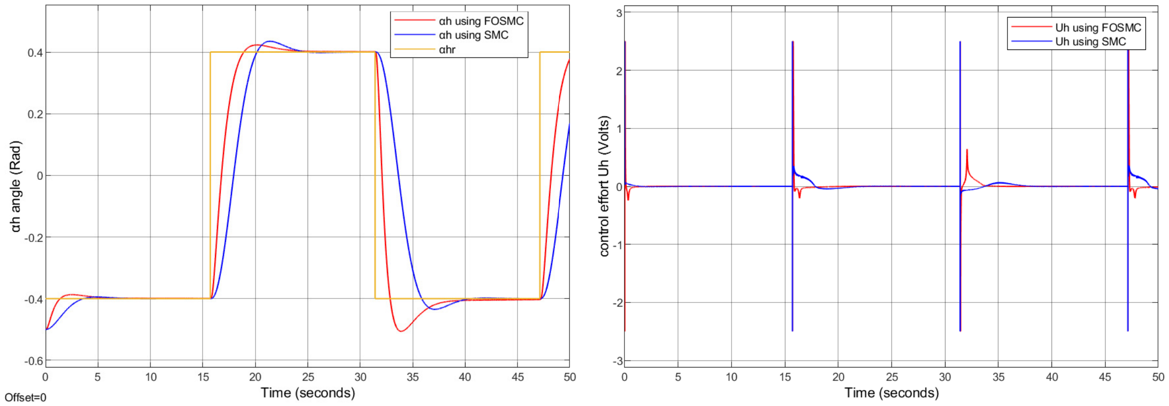


Figure 4.15: Response of the Horizontal Subsystem for Both Controllers to a Square Trajectory

Interpretation of the Results

Vertical Subsystem Response

- The FOSMC controller achieves better reference tracking of the square signal, with reduced overshoot compared to the standard SMC.
- Despite both controllers stabilizing around the desired reference,

- The control signal under SMC exhibits high-frequency chattering, particularly at transition points of the square signal.
- The control signal under FOSMC significantly reduces sharp peaks, resulting in a smoother control effort and enhanced energy efficiency.

Horizontal Subsystem Response

- the FOSMC controller exhibits faster response and better alignment with the square reference
- While SMC demonstrates acceptable reference tracking.
- The control input U_h is smooth and continuous for both controllers.

4.5.2 disturbance rejection Analysis

To evaluate the disturbance rejection capability of both controllers, we introduce an external disturbance in the form of two impulse signals applied at $T = 0s$ and $T = 30s$ with an amplitude of 0.3. This disturbance simulates sudden external forces or unexpected system variations, as shown in Figure 4.16.

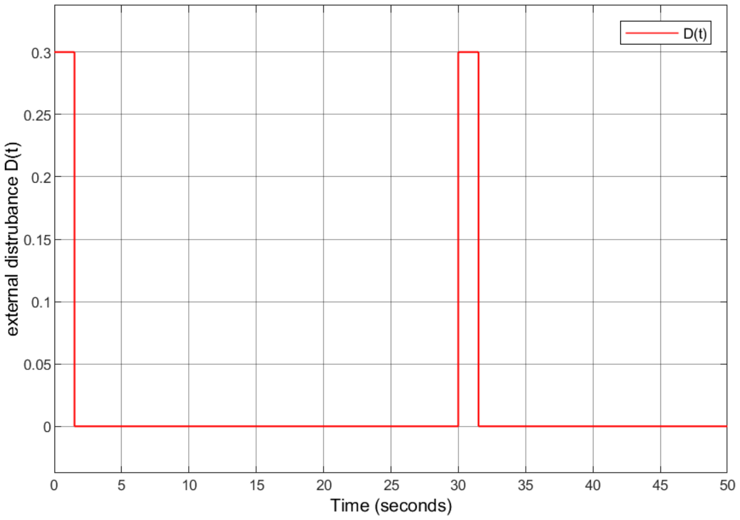


Figure 4.16: impulse disturbances at $T = 0s$ and $T = 30s$.

Simulation Results

We get the following Results (figures 4.17 and 4.18)

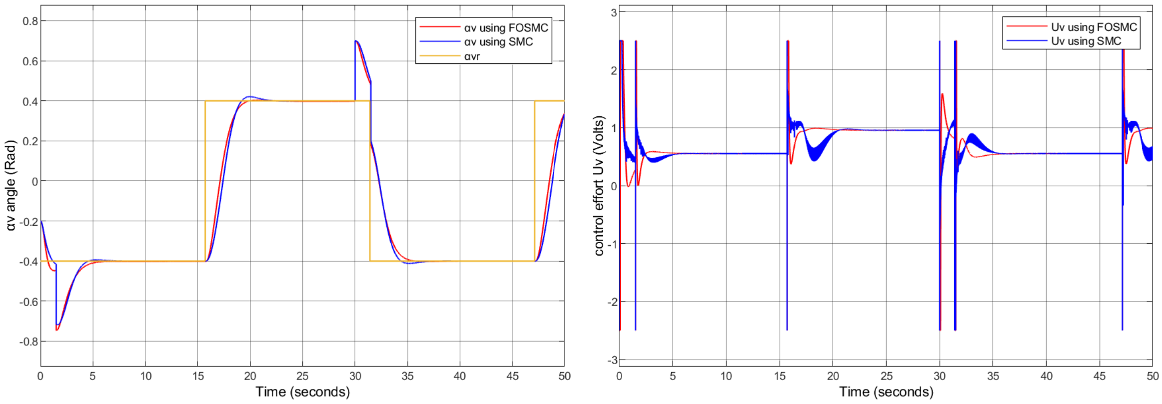


Figure 4.17: Response of the Vertical Subsystem for SMC and FOSMC

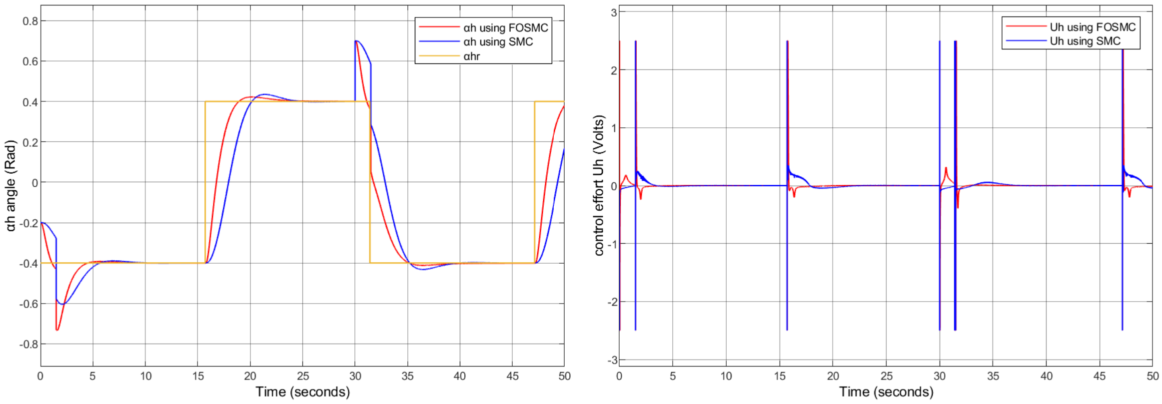


Figure 4.18: Response of the Horizontal Subsystem for SMC and FOSMC

Discussion on Disturbance Rejection

Vertical Subsystem Response

- Both controllers show noticeable deviation after impulse disturbances
- FOSMC demonstrates smoother recovery with minimal oscillations
- The control signal under SMC exhibits high-frequency, particularly at transition points of the square signal.

Horizontal Subsystem Response

- The FOSMC controller offers better synchronization with the square reference trajectory, achieving faster and more stable performance
- SMC demonstrates acceptable reference tracking
- The control input U_h is smooth and continuous for both controllers

4.5.3 Applying the Controllers to the Coupled System

The SMC and FOSMC controllers are now applied to the coupled Twin Rotor MIMO System. Unlike previous tests where each subsystem was considered independently, the coupling effects between the vertical and horizontal subsystems are now taken into account.

The coupling between the two subsystems is treated as an external disturbance affecting both dynamics. This allows us to evaluate the controllers ability to maintain stability, and ensure reference tracking.

The controllers are now tested with a square signal as the input, with an amplitude of 0.2 and a frequency of 0.15 rad/sec.

Simulation Results

The following results are obtained (Figures 4.19 and 4.20):

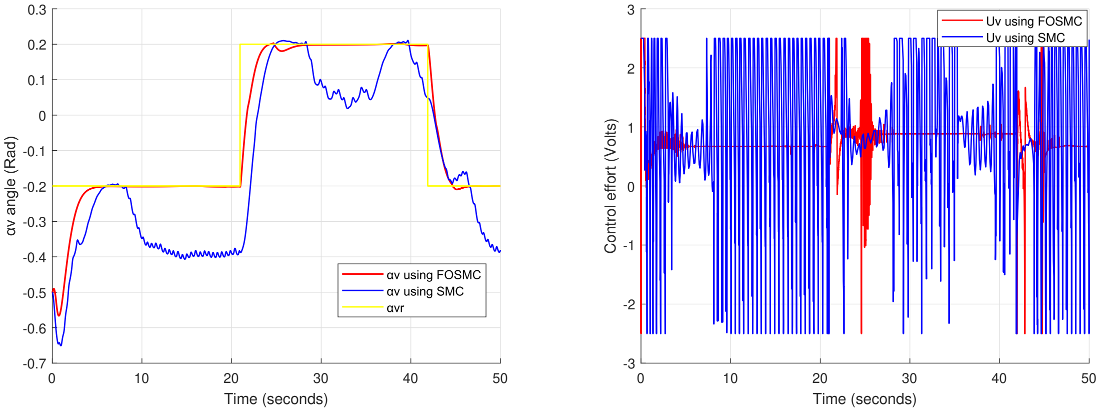


Figure 4.19: Vertical Response of the Coupled System: SMC vs. FOSMC

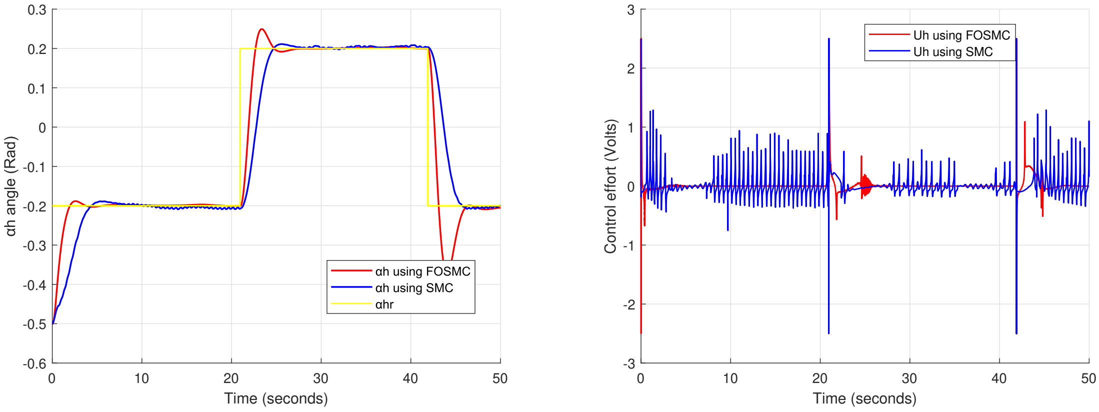


Figure 4.20: Horizontal Response of the Coupled System: SMC vs. FOSMC

Coupled System Response Analysis

Vertical Subsystem Response

1. **FOSMC Controller**:
    - Provides accurate tracking of the square reference input with minimal steady-state error.
    - Ensures a smooth and stable response with fast settling time.
    - The control signal U_v is nearly chatter-free, indicating reduced switching activity.
    - Maintains performance despite the coupling effect from the horizontal subsystem.
2. **SMC Controller**:
    - Exhibits tracking oscillations and notable steady-state error.
    - The control effort U_v shows strong chattering throughout the simulation.
    - Aggressive switching leads to inefficient energy usage and potential actuator wear.
    - Susceptible to disturbances from horizontal subsystem dynamics.

Horizontal Subsystem Response

1. **FOSMC Controller**:
    - Achieves accurate reference tracking with minor overshoot.
    - Control signal U_h is smooth and stable with minimal chattering.
    - Demonstrates robustness against coupling effects from the vertical axis.
    - Maintains efficient control performance during signal transitions.
2. **SMC Controller**:
    - Shows overshoot and slower convergence to the reference.
    - The control effort U_h includes high-frequency oscillations and chattering.
    - Response is sensitive to inter-axis disturbances, reducing stability.
    - Less energy-efficient due to excessive control effort.

These observations confirm that the FOSMC controller significantly outperforms the SMC controller in handling the coupled TRMS system, offering better stability, disturbance rejection, and tracking performance.

4.6 Conclusion

This comparative study between SMC and FOSMC for the Twin Rotor MIMO System reveals clear advantages of the fractional-order approach. While both controllers achieve reference tracking, FOSMC demonstrates superior performance with smoother control signals, faster response, and minimal chattering compared to conventional SMC.

The FOSMC controller shows exceptional disturbance rejection capabilities and maintains stability despite coupling effects between subsystems. In contrast, SMC exhibits noticeable oscillations and energy-inefficient chattering,

Chapter 5

Experimental TRMS Control through PID and FOPID

Experiments are essential to validate the observations and conclusions drawn from theoretical studies. Establishing a connection between theory and its practical implementation is crucial for understanding and applying control strategies in engineering systems. Therefore, a series of experiments were conducted on the real Twin Rotor MIMO System (TRMS) to assess the effectiveness of the PID and FOPID controllers in stabilizing the system at the origin, in tracking desired reference trajectories, and in ensuring robustness against disturbances and model uncertainties.

The parameters of the PID and FOPID controllers were obtained through simulation studies based on the nonlinear mathematical model of the TRMS, as discussed in Chapter 3. In this experimental validation, we focus on three aspects of controller performance:

- **Stabilization**: Bringing the system to the equilibrium point (origin) from an initial condition.
- **Trajectory Tracking**: Following a time-varying reference trajectory, such as a sinusoidal signal, to evaluate the controller's tracking capability.
- **Robustness**: Testing the system's ability to maintain stability and performance in the presence of external disturbances and model uncertainties.

The designed PID and FOPID controllers were implemented on the physical TRMS, and the experimental results are analyzed to assess their practical effectiveness under different conditions.

5.1 Pre-Experiment Configuration

Before proceeding to the different experiments, let us outline some key instructions to ensure that the data exported from the experiments accurately reflects the actual system behavior.

First, we used MATLAB 6.1 to create the pid and fopid structure as a Simulink file, which required specific settings adjustments. To ensure smooth execution of the program, follow these steps:

1. Open the settings window of the Simulink file and:
    - Set the numerical solver to `ode1` (Euler method) with a fixed step size of 0.001.
    - Deselect all selected options in the `I/O Workspace` tab.
    - Verify that the target is set to `rtwinfeedback.tlc` in the `Realtime Workshop` tab.
2. In the `Tools` → `Drag` menu, set the control panel mode to `External`.
3. Use the built-in Microsoft C++ compiler to build the Simulink file by pressing the `Ctrl+B` shortcut.

Once the software setup is complete, it is ready to be transmitted to the hardware. For this, the TRMS was held and maintained at $\psi = \varphi = 0$ rad, and the Simulink file was connected to the system for experimental testing. the actual system by tapping at the following key: the reason that the connection was done in this specific position is that the sensor (i.e. position encoders) needed to be initialized at zero so that it best reflects the behavior of the physical plant. Now, the system can be let go of and the simulation is started by pressing the start button and clicking on the run command of the system simultaneously. All that rest now is opening the scope and observing the closed-loop behavior. This process is the same and needs to be repeated for every single experiment, except for building the Simulink file which is done only when it is edited.

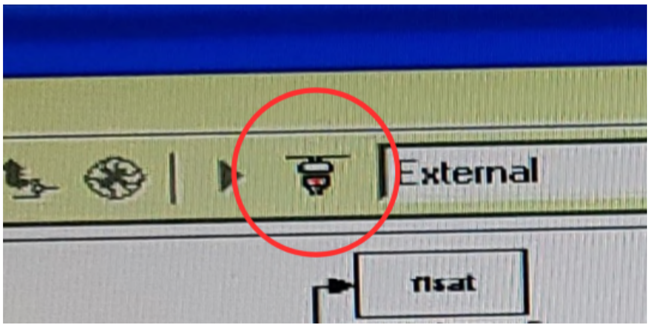


Figure 5.1: TRMS Connection button using Matlab 6.1

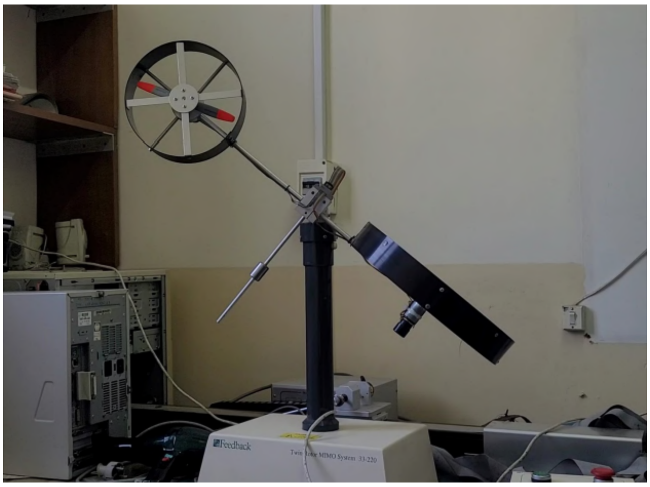


Figure 5.2: Real TRMS Experimentation

5.2 Experimental Scenarios

This section provides an overview of the experimental procedures conducted on the Twin Rotor MIMO system to evaluate the performance of PID and FOPID controllers. The experiments aim to validate the controllers under different scenarios, both in the decoupled and coupled configurations.

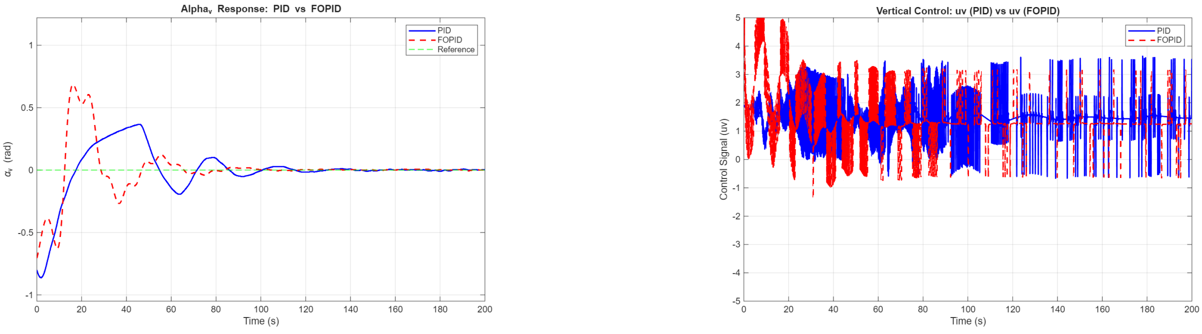
The first test focuses on stabilizing the decoupled system around a zero reference using PID and FOPID controllers. A step input signal is applied, setting the desired horizontal and vertical angles to zero. The stabilization performance is evaluated using metrics such as Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Overshoot, Steady-State Error, Settling Time, Integral of Absolute Error (IAE), and Integral of Squared Error (ISE).

The second test involves trajectory tracking, where the system follows a sinusoidal reference signal of amplitude 0.2 and frequency 0.07 Hz on both the horizontal and vertical angles. Tracking accuracy is assessed based on the correlation between the reference and output signals, as well as the phase delay.

A robustness test is also performed by manually introducing a wind disturbance to assess the controllers' ability to reject external perturbations. The system's deviation and recovery time under these conditions are analyzed.

Finally, the same experimental scenarios are repeated for the coupled system configuration, where the interaction between horizontal and vertical dynamics introduces additional challenges such as cross-axis effects and nonlinearities.

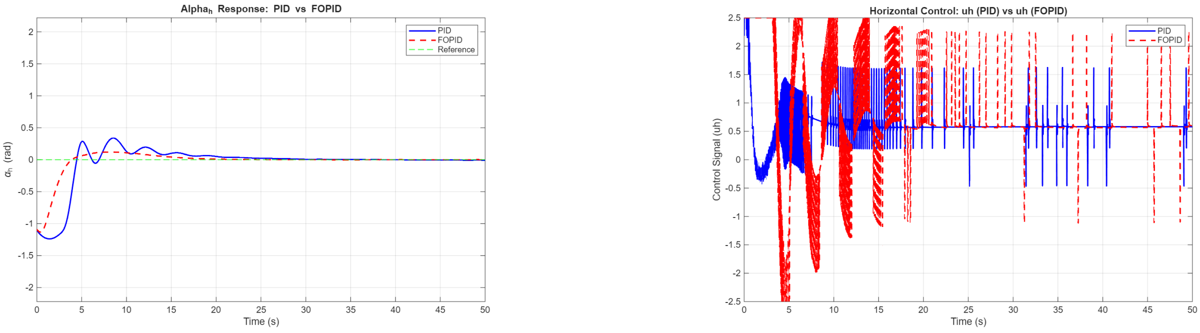
5.3 Decoupled System stabilization using PID and FOPID Controllers



(a) Vertical angular response α_v

(b) Vertical control signal U_v

Figure 5.3: Response of the Vertical Subsystem for PID and FOPID Controllers.



(a) Horizontal angular response α_h

(b) Horizontal control signal U_h

Figure 5.4: Response of the Horizontal Subsystem for PID and FOPID Controllers.

Table 5.1: Performance Metrics for Vertical (Pitch) and Horizontal (Yaw) Angles

vertical angle Metrics

Metric	PID	FOPID
RMSE	0.2790	0.2281
MAE	0.2067	0.1729
Overshoot	2.7978	2.4651
Steady-State Error	0.1232	0.1230
Settling Time (s)	5.4300	3.2500
IAE	10.3321	8.6413
ISE	3.8878	2.5988

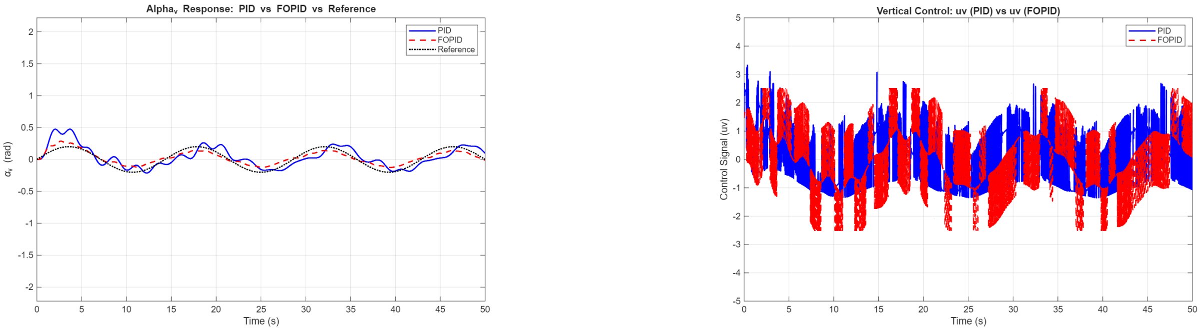
Horizontal angle Metrics

Metric	PID	FOPID
RMSE	0.4331	0.2308
MAE	0.2847	0.1696
Overshoot	2.3703	3.1962
Steady-State Error	0.1539	0.1268
Settling Time (s)	4.7100	5.7200
IAE	14.2267	8.4775
ISE	9.3687	2.6593

Observations: The results clearly highlight the superiority of the FOPID controller over the classical PID in most performance metrics for both pitch and yaw angles. In particular, the FOPID significantly reduces the Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Integral Absolute Error (IAE), and Integral Square Error (ISE) in both axes—indicating better tracking accuracy and disturbance rejection. For the vertical (pitch) angle, FOPID also improves the settling time and slightly lowers the overshoot compared to PID.

However, in the horizontal (yaw) axis, while FOPID offers better tracking performance (lower RMSE, MAE, IAE, ISE), it shows a slightly higher overshoot and longer settling time, which may reflect a trade-off between speed and stability in that direction. Overall, the FOPID demonstrates more robust and efficient control behavior, particularly in reducing cumulative errors.

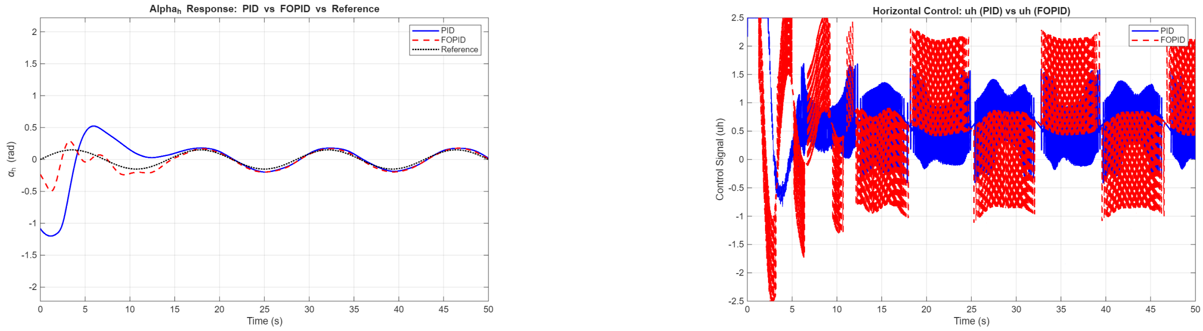
5.4 decoupled System trajectory tracking using PID and FOPID Controllers



(a) Vertical angular response α_v

(b) Control signal U_v

Figure 5.5: Response of the Vertical Subsystem for PID and FOPID Controllers to a Sinusoidal Trajectory.



(a) Horizontal angular response α_h

(b) Control signal U_h

Figure 5.6: Response of the Horizontal Subsystem for PID and FOPID Controllers to a Sinusoidal Trajectory.

Table 5.2: Correlation and Phase Delay Metrics for Vertical (Pitch) and Horizontal (Yaw) Angles

Vertical Angle (Pitch) Metrics

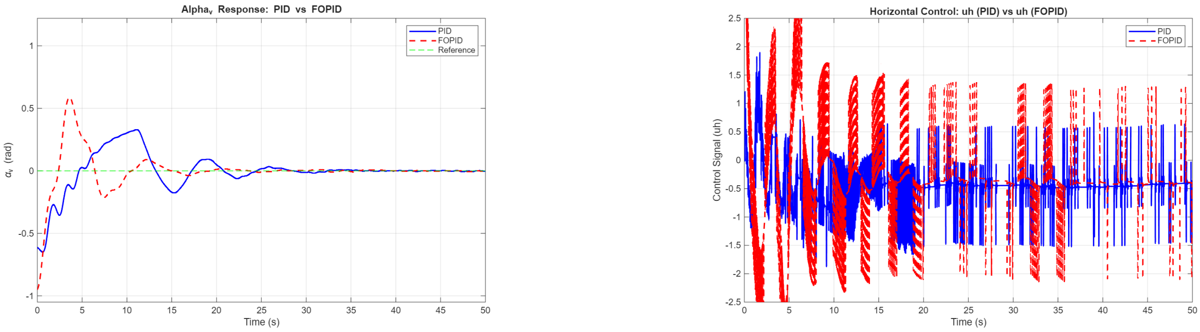
Metric	PID	FOPID
Correlation	0.7710	0.9353
Phase Delay (s)	0.9100	-0.1000

Horizontal Angle (Yaw) Metrics

Metric	PID	FOPID
Correlation	0.1162	0.7771
Phase Delay (s)	-38.3000	0.4700

Observations: The correlation and phase delay metrics demonstrate that the FOPID controller offers superior tracking performance compared to the classical PID. It ensures better signal alignment and responsiveness in both pitch and yaw axes, with notably improved correlation and reduced phase lag—especially in the yaw axis, where the PID fails to track effectively.

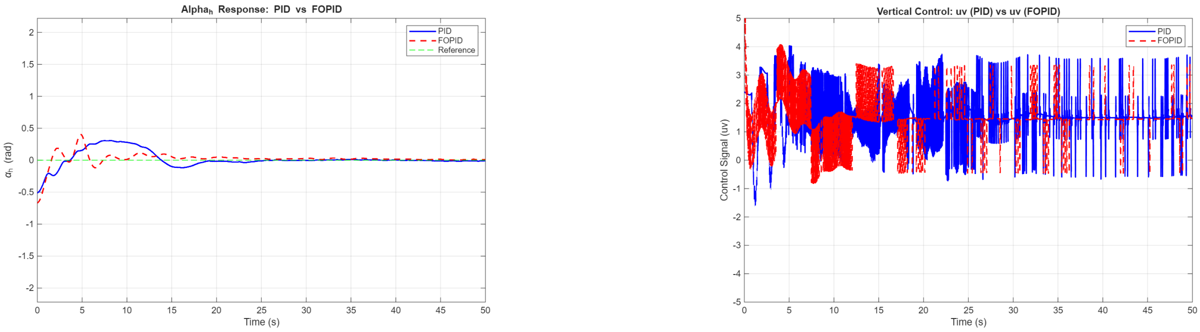
5.5 Coupled System Stabilization using PID and FOPID Controllers



(a) Vertical angular response α_v

(b) Control signal U_v

Figure 5.7: Response and control signal of the Vertical Subsystem for PID and FOPID Controllers



(a) Horizontal angular response α_h

(b) Control signal U_h

Figure 5.8: Response and control signal of the Horizontal Subsystem

Table 5.3: Performance Metrics for Vertical (Pitch) and Horizontal (Yaw) Angles

Vertical Angle (Pitch) Metrics

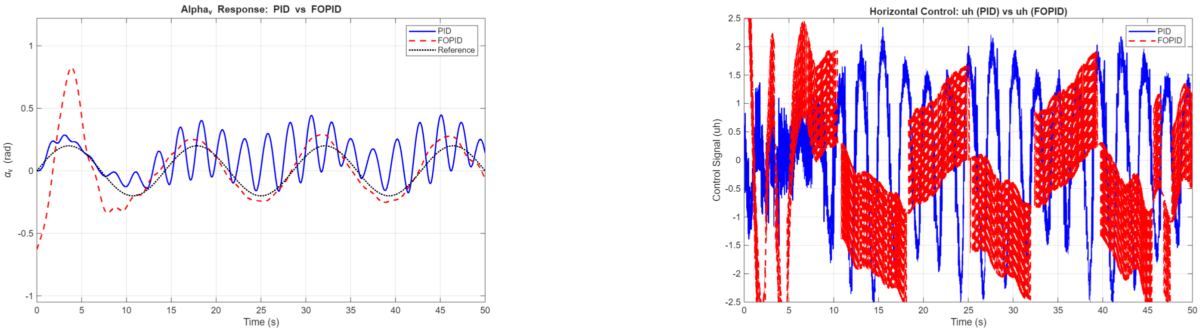
Metric	PID	FOPID
RMSE	0.2402	0.1930
MAE	0.1894	0.1529
Overshoot	1.9305	2.6373
Steady-State Error	0.1229	0.1236
Settling Time (s)	5.8900	2.6700
IAE	9.4667	7.6383
ISE	2.8821	1.8577

Horizontal Angle (Yaw) Metrics

Metric	PID	FOPID
RMSE	0.2171	0.1695
MAE	0.1814	0.1423
Overshoot	1.5346	2.3585
Steady-State Error	0.1299	0.1199
Settling Time (s)	4.9700	1.9200
IAE	9.0657	7.1124
ISE	2.3549	1.4343

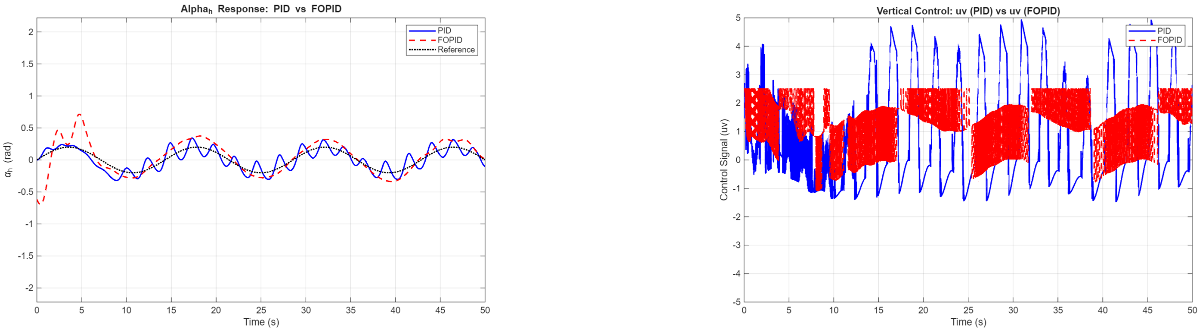
Observations: The FOPID controller demonstrates superior overall performance compared to PID, with lower tracking errors (RMSE, MAE), faster settling times, and reduced cumulative errors (IAE, ISE) on both pitch and yaw axes. However, this improved responsiveness comes at the cost of a slightly higher overshoot—notably 2.64% vs. 1.93% for pitch, and 2.36% vs. 1.53% for yaw. While the overshoot is slightly increased, it remains within acceptable bounds and is outweighed by the gains in speed and accuracy, making FOPID a more effective controller for stabilization.

5.6 Coupled System Trajectory Tracking using PID and FOPID Controllers



(a) Vertical angular response α_v (b) Control signal U_v

Figure 5.9: PID and FOPID Vertical Subsystem Response and Control Signal to Sinusoidal Trajectory in Coupled System.



(a) Horizontal angular response α_h (b) Control signal U_h

Figure 5.10: PID and FOPID Horizontal Subsystem Response and Control Signal to Sinusoidal Trajectory in Coupled System.

Table 5.4: Correlation and Phase Delay Metrics for Vertical (Pitch) and Horizontal (Yaw) Angles

Vertical Angle (Pitch) Metrics

Metric	PID	FOPID
Correlation	0.4880	0.8075
Phase Delay (s)	-0.2500	-0.1900

Horizontal Angle (Yaw) Metrics

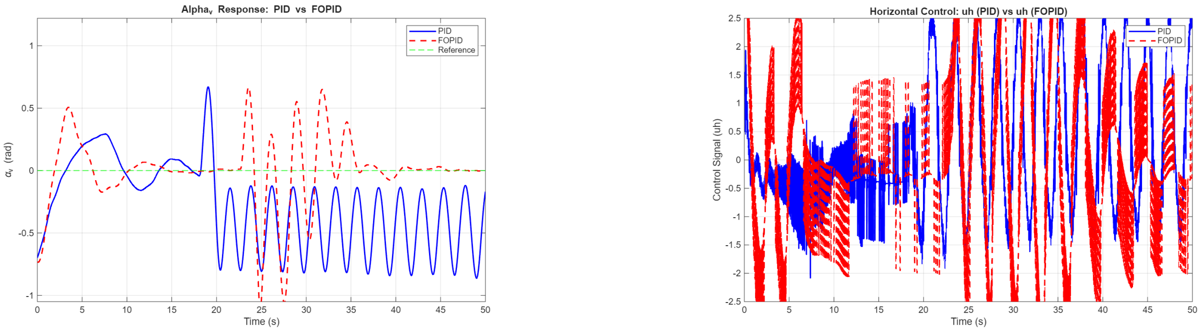
Metric	PID	FOPID
Correlation	0.8303	0.8620
Phase Delay (s)	-0.3800	0.2000

Observation: The tracking performance in the coupled TRMS system confirms the advantage of the FOPID controller over the classical PID. As shown in Figures5.9 and 5.10, FOPID provides smoother and more accurate responses in both vertical and horizontal subsystems when following a sinusoidal reference, with reduced phase lag and more stable control signals.

The metrics in Table5.4 reinforce this observation. For the vertical (pitch) axis, FOPID improves the correlation from 0.4880 to 0.8075 and slightly reduces the phase delay (0.19s vs. 0.25s). For the horizontal (yaw) axis, both controllers perform well, but FOPID maintains a higher correlation (0.8620 vs. 0.8303) and achieves a positive phase delay (0.20s), indicating a faster and more synchronized response.

Overall, FOPID enhances trajectory tracking accuracy and reduces lag in both axes under coupled dynamics.

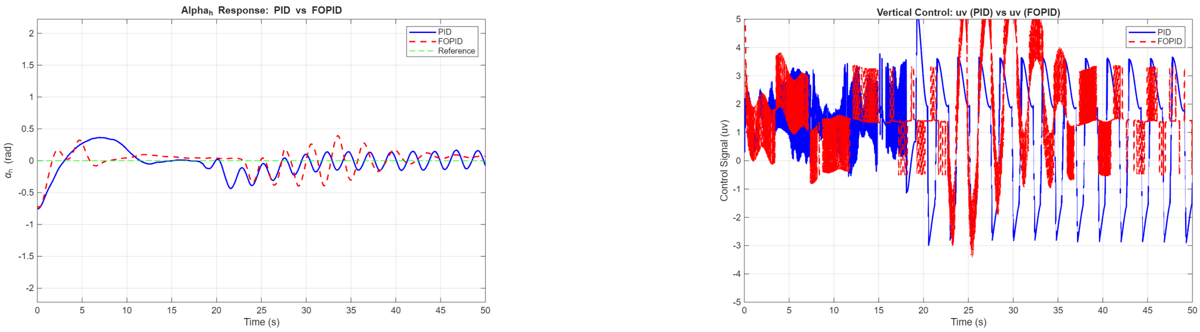
5.7 Coupled System Stabilization using PID and FOPID Controllers under Perturbation



(a) Vertical angular response α_v

(b) Control signal U_v

Figure 5.11: PID and FOPID Vertical Subsystem Response and Control Signal in Coupled System under Perturbation.



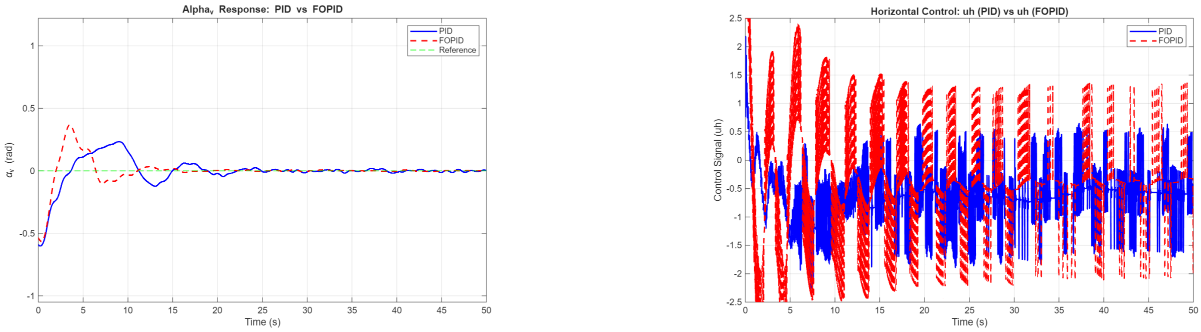
(a) Horizontal angular response α_h

(b) Control signal U_h

Figure 5.12: PID and FOPID Horizontal Subsystem Response and Control Signal in Coupled System under Perturbation.

Observation: The results show that the PID controller becomes unstable under perturbation, leading to a limit cycle and persistent oscillations, particularly in the vertical subsystem. In contrast, the FOPID controller effectively damps the disturbances, providing a more stable response and smoother control signal in both axes. It thus offers greater robustness, better disturbance rejection, and enhanced stability, confirming its relevance for strongly nonlinear coupled systems subjected to real-world perturbations.

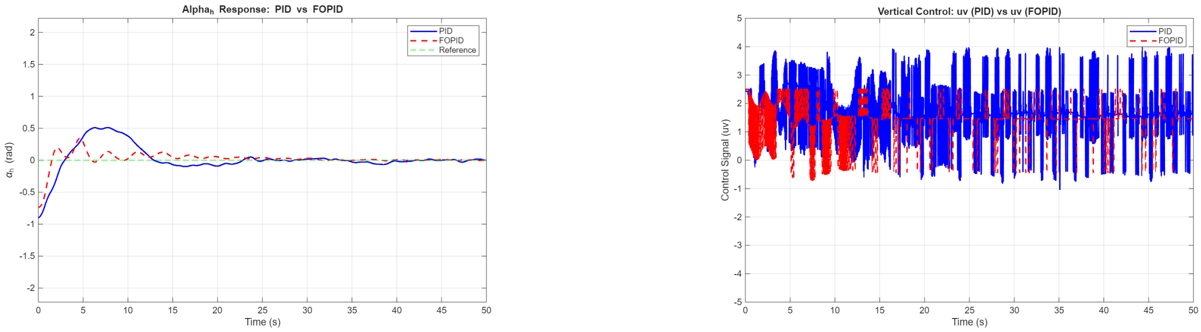
5.8 Coupled System Stabilization using PID and FOPID Controllers under Parameter Variation



(a) Vertical angular response α_h

(b) Control signal U_v

Figure 5.13: PID and FOPID Vertical Subsystem Response and Control Signal in Coupled System under Parameter Variation.



(a) Horizontal angular response α_h

(b) Control signal U_h

Figure 5.14: PID and FOPID Horizontal Subsystem Response and Control Signal in Coupled System under Parameter Variation.

Observation: Both controllers perform well under parameter variation, ensuring system stability. However, the FOPID controller shows slightly better transient behavior and smoother control signals, offering a small but consistent advantage over the classical PID.

5.9 Conclusion

his chapter presented a comprehensive experimental validation of PID and FOPID controllers applied to the Twin Rotor MIMO System (TRMS). Through a series of tests conducted on the real system including stabilization, trajectory tracking, and robustness assessment under perturbations and parameter variations—we evaluated the practical effectiveness of both controllers in handling complex, nonlinear, and coupled dynamics.

The experimental results confirm that while the classical PID controller achieves acceptable performance in many scenarios, it exhibits clear limitations under challenging conditions such as external disturbances and system uncertainties. In contrast, the FOPID controller consistently demonstrates superior performance across all tested scenarios. It provides more accurate stabilization, better trajectory tracking, faster convergence, smoother control actions, and enhanced robustness—especially in the presence of perturbations and parameter deviations.

Overall, the FOPID controller proves to be a more reliable and efficient solution for real-world control of nonlinear and coupled systems like the TRMS. These findings reinforce the theoretical advantages of fractional-order control and validate its practical applicability in modern control engineering.

General conclusion

The work presented in this thesis focuses on the modeling and control of the Twin Rotor MIMO System (TRMS), with the implementation of fractional-order control strategies and the use of Particle Swarm Optimization (PSO) for tuning.

Fractional calculus extends the integral-differential operator to any positive real order, offering a broad range of applications and several advantages across various scientific fields—particularly in control theory, which is the main focus of this study. As such, we introduced the fundamental properties and tools necessary to transition from classical integer-order control theory to its fractional-order counterpart.

The application part of this work includes two main control strategies: PID control and Sliding Mode Control (SMC). For each strategy, two versions of the controller were synthesized—one based on classical integer-order design, and the other on fractional-order design. Due to the nonlinear nature of the TRMS, the PSO algorithm was employed to determine optimal parameters for all controllers. This optimization included PID gains, the sliding surface constants for SMC, and the fractional orders

and

, yielding the best possible performance for the proposed control laws.

For the PID and FOPID controllers, a detailed comparative study was conducted. The results demonstrated the superiority of the fractional-order approach in several aspects: improved stability, better oscillation damping, enhanced robustness to modeling errors, and strong rejection of external disturbances.

Regarding Sliding Mode Control, we contributed to the development of a fractional-order sliding mode law, which showed, in most cases, higher performance than its classical counterpart. While both SMC controllers exhibited strong robustness to model uncertainties—often exceeding that of PID and FOPID—the fractional-order SMC proved to be more effective in trajectory tracking, especially under strong disturbances. In contrast, the classical SMC tended to diverge under such conditions.

Throughout this study, fractional-order controllers have consistently demonstrated their effectiveness in controlling complex nonlinear systems like the TRMS. They also offer increased design flexibility, particularly due to their smoother input signals, which are better suited to real-world actuators. Furthermore, these controllers have shown excellent performance across multiple evaluation criteria.

A practical validation of the PID and FOPID controllers was carried out on the physical TRMS platform. Experimental results confirmed the findings from simulation studies: the FOPID controller consistently achieved better stabilization, faster convergence, smoother control efforts, and superior tracking performance compared to the classical PID. These results demonstrate the practical feasibility and benefits of fractional-order control in real-time embedded systems.

Future perspectives:

- Extending other classical control methods (e.g., LQR, H-infinity, adaptive control) to their fractional-order counterparts.
- Exploring alternative sliding surface definitions to further enhance the performance and flexibility of sliding mode control strategies.
- Implementing both classical Sliding Mode Control (SMC) and its fractional-order version (FOSMC) on the real TRMS platform to validate their practical robustness against disturbances and coupling effect.

In conclusion, this work confirms that fractional-order controllers are not only theoretically promising but also experimentally viable, offering robust, flexible, and high-performance solutions for nonlinear and coupled system control.

Bibliography

[1] Federal Aviation Administration. *Helicopter Flying Handbook*. U.S. Department of Transportation, Federal Aviation Administration, 2019. Chapter 2: Aerodynamics of Flight.

[2] Bhukya Ramadevi, Venkata Ramana Kasi, and Kishore Bingi. Hybrid lstm-based fractional-order neural network for jeju island's wind farm power forecasting. *Fractal and Fractional*, 8(3), 2024.

[3] Yixiao Ding, Xiaolian Liu, Pengchong Chen, Xin Luo, and Ying Luo. Fractional-order impedance control for robot manipulator. *Fractal and Fractional*, 6(11), 2022.

[4] Kishore Bingi, B Rajanarayan Prusty, and Abhaya Pal Singh. A review on fractional-order modelling and control of robotic manipulators. *Fractal and Fractional*, 7(1), 2023.

[5] I. Podlubny. Fractional-order systems and pi/sup /spl lambda//d/sup /spl mu//-controllers. *IEEE Transactions on Automatic Control*, 44(1):208–214, 1999.

[6] Feedback Instruments Ltd. *Twin Rotor MIMO System Control Experiment*. Feedback Instruments Ltd., Crowborough, East Sussex, UK, 1998.

[7] David Mohammed Ezekiel, Ravi Samikannu, and Matsebe Oduetse. Modelling of the twin rotor mimo system (trms) using the first principles approach. In *2020 International Conference on Computer Communication and Informatics (ICCCI)*, pages 1–7, 2020.

[8] Bidyadhar Subudhi and Debashisha Jena. Nonlinear system identification of a twin rotor mimo system. In *TENCON 2009 - 2009 IEEE Region 10 Conference*, pages 1–6, 2009.

[9] S. F. Toha, S. Julai, and M. O. Tokhi. Ant colony based model prediction of a twin rotor system. *Procedia Engineering*, 41:1135–1144.

[10] L. D. S. Coelho, M. W. Pessoa, V. C. Mariani, and G. Reynoso-Meza. Fuzzy inference system approach using clustering and differential evolution optimization applied to identification of a twin rotor system. In *IFAC-PapersOnLine*, volume 50, pages 13102–13107. Elsevier, 2017.

[11] F. A. Shaik, S. Purwar, and B. Pratap. Real-time implementation of chebyshev neural network observer for twin rotor control system. *Expert Systems with Applications*, 38(10):13043–13049, 2011.

[12] Nghia Nguyen-Huu. Modelling, simulation, and calibration of twin rotor mimo system. 06 2007.

[13] Rudolf Gorenflo and Francesco Mainardi. *Fractional calculus: integral and differential equations of fractional order, in: A. Carpinteri, F. Mainardi (Eds.), Fractals and Fractional Calculus in Continuum Mechanics*. 01 1997.

[14] K. B. Oldham and J. Spanier. *The Fractional Calculus.* Academic Press, New York, NY, USA, 1974.

[15] Jocelyn Sabatier, Om Agrawal, and José Tenreiro Machado. *Advances in fractional calculus: theoretical developments and applications in physics and Engineering.* 06 2007.

[16] I. Podlubny. *Fractional Differential Equations.* Academic Press, San Diego, 1999.

[17] Riccardo Caponetto, Luigi Fortuna, and D. Porto. Parameter tuning of a non integer order pid. 07 2002.

[18] C. Casenave and G. Montseny. Introduction to diffusive representation. *IFAC Proceedings Volumes*, 43(21):370–377, 2010. 4th IFAC Symposium on System Structure and Control.

[19] Likun Li, Liyu Jiang, Wenzhang Tu, Liquan Jiang, and Ruhan He. Smooth and efficient path planning for car-like mobile robot using improved ant colony optimization in narrow and large-size scenes. *Fractal and Fractional*, 8(3), 2024.

[20] Farshad Merrikh-Bayat. Rules for selecting the parameters of oustaloup recursive approximation for the simulation of linear feedback systems containing pid controller. *Communications in Nonlinear Science and Numerical Simulation*, 17(4):1852–1861, 2012.

[21] A. Charef, H.H. Sun, Y.Y. Tsao, and B. Onaral. Fractal system as represented by singularity function. *IEEE Transactions on Automatic Control*, 37(9):1465–1470, 1992.

[22] Karl Johan Åström and Tore Hägglund. *Advanced PID Control.* ISA – The Instrumentation, Systems, and Automation Society, 2006.

[23] C. A. Monje, Y. Chen, B. M. Vinagre, D. Xue, and V. Feliu. *Fractional-order Systems and Controls: Fundamentals and Applications.* Springer, 2010.

[24] James Kennedy and Russell Eberhart. Particle swarm optimization. In *Proceedings of ICNN'95 - International Conference on Neural Networks*, volume 4, pages 1942–1948. IEEE, 1995.

[25] Mehdi J. Marie, Ghaida A. Al-Suhail, and Wisam A. Latif. Pso-based optimal pid controller for twin rotor mimo system. *International Journal of Computers & Technology*, 14(5):5719–5726, March 2015. Dept. of Electrical & Computer Engineering, University of Basrah, Iraq and Al-Tahady State Company, Ministry of Industry & Minerals, Baghdad, Iraq.

[26] Tareq M. Shami, Ayman A. El-Saleh, Mohammed Alswaitti, Qasem Al-Tashi, Mhd Amen Summakieh, and Seyedali Mirjalili. Particle swarm optimization: A comprehensive survey. *IEEE Access*, 10:23681–23726, January 2022. Received November 26, 2021, accepted January 10, 2022, date of publication January 13, 2022, date of current version January 27, 2022.

[27] Riccardo Poli, James Kennedy, and Tim Blackwell. Particle swarm optimization: An overview. *Swarm Intelligence*, 1(1):33–57, August 2007. Received December 19, 2006; accepted May 10, 2007; published online August 1, 2007.

[28] Daniel Bratton and James Kennedy. Defining a standard for particle swarm optimization. In *Proceedings of the 2007 IEEE Swarm Intelligence Symposium (SIS 2007)*, pages 120–127, Honolulu, HI, USA, 2007. IEEE.

[29] Pham Luu Trung Duong and Moonyong Lee. Pid controller design for fractional order systems with random parameters. In *2011 International Symposium on Advanced Control of Industrial Processes (ADCONIP)*, pages 559–564, 2011.

[30] I. Podlubny, Ivo Petráš, Blas Vinagre, Paul O'Leary, and L̆ Dorčák. Analogue realizations of fractional-order controllers. *Nonlinear Dynamics*, 29:281–296, 07 2002.

[31] Richard C. Dorf and Robert H. Bishop. *Modern Control Systems.* Addison-Wesley, New York, 6 edition, 1990.

[32] Igor Petráš. The fractional-order controllers: Methods for their synthesis and application. *Journal of Electrical Engineering*, 50:284–288, 1999.

[33] Igor Petráš and Ľubomír Dorčák. Fractional-order control systems: Modelling and simulation. *Fractional Calculus and Applied Analysis*, 6(3):205–232, 2003.

[34] C. A. Monje, B. M. Vinagre, V. Feliu, and Y. Q. Chen. Tuning and auto-tuning of fractional order controllers for industry application. *Control Engineering Practice*, 16(7):798–812, 2008.

[35] Blas Vinagre, I. Podlubny, A. Hernandez, and Vicente Feliu. On realization of fractional-order controllers. *Proc. of Conference Internationale Fracophone d'Automatique*, 7:945–950, 01 2000.

[36] S. Das, I. Pan, S. Das, and A. Gupta. A novel fractional order fuzzy pid controller and its optimal time domain tuning based on integral performance indices. *Engineering Applications of Artificial Intelligence*, 25(2):430–442, 2012.

[37] Duarte Valério and Jenny da Costa. Introduction to single-input, single-output fractional control. *Control Theory Applications, IET*, 5:1033–1057, 05 2011.

[38] V. I. Utkin. Variable structure systems with sliding modes. *IEEE Transactions on Automatic Control*, 22(2):212–222, 1977.

[39] J. Y. Hung, W. Gao, and J. C. Hung. Variable structure control: A survey. *IEEE Transactions on Industrial Electronics*, 40(1):2–22, February 1993.

[40] H. Sira-Ramirez. Periodic sliding motions. *IEEE Transactions on Automatic Control*, 33(12):1191–1193, 1988. Manuscript received July 13, 1987; revised Nov. 16, 1987 and Mar. 3, 1988.

[41] J.-J. E. Slotine, J. Hedrick, and E. Misawa. Nonlinear state estimation using sliding observers. In *Proceedings of the 25th IEEE Conference on Decision and Control (CDC)*, pages 332–339, Athens, Greece, 1986. IEEE.

[42] V. I. Utkin. *Sliding Modes in Control and Optimization.* Communication and Control Engineering. Springer-Verlag, Berlin, Germany, 1992.

[43] Farah Faris, Abdelkrim Moussaoui, Boukhetala Djamel, and Tadjine Mohammed. Design and real-time implementation of a decentralized sliding mode controller for twin rotor multi-input multi-output system. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 231(1):3–13, 2017.

[44] N. Bouarroudj, D. Boukhetala, and F. Boudjema. Sliding-mode controller based on fractional order calculus for a class of nonlinear systems. *International Journal of Electrical and Computer Engineering*, 6(4):2239–2248, 2016.

Business Model Canvas

1. Customer Segments

- Aerospace R&D departments (e.g., drones, helicopters)
- Robotics and automation companies
- Academic institutions and research labs
- Industrial control integrators
- Control system hardware vendors

2. Value Proposition

- Robust fractional-order controllers (FOPID, FOSMC)
- Improved disturbance rejection and adaptability
- Simulation and real-time deployment on TRMS
- Ready-to-integrate modules for education and R&D

3. Channels

- Direct industry partnerships
- Academic and industrial conferences
- Online distribution platform (API/Software)
- Pilot projects and demos

4. Customer Relationships

- Co-development and integration support
- Technical maintenance contracts
- Training and certification programs
- Free trials for academic use

5. Revenue Streams

- Software library licensing
- Consulting and engineering services
- Sale of TRMS-based platforms
- Training and workshops

6. Key Resources

- Proprietary FOPID/FOSMC algorithms
- TRMS demo platform
- Expertise in fractional control and optimization
- Research and industrial network

7. Key Activities

- Control algorithm development
- Hardware/software integration
- Simulation and experimentation
- Marketing and customer support

8. Key Partnerships

- National Polytechnic School (ENP)
- Aerospace research centers
- TRMS manufacturers
- Incubators and innovation hubs

9. Cost Structure

- R&D and prototyping
- Hardware maintenance and testing
- Personnel and consulting
- Marketing and outreach