People's Democratic Republic of Algeria

Ministry of Higher Education and Scientific Research

Mohamed Khider University of Biskra

FACULTY OF EXACT SCIENCES

DEPARTEMENT OF MATHEMATICS

Thesis Submitted in Partial Execution of the Requirements of the Degree of

Master in "**Mathematics**"

Option: Partial Differential Equations and Numerical Analysis

Submitted and Defended By

Noura DERBALI

Title :

Chaos Control for a Fractional-Order System via Time Delay Controller.

Examination Committee Members:

Dr.	Fatiha GHEDJEMIS	MKUB	President
Pr.	Tidjani MENACER	MKUB	Supervisor
Dr.	Fatma KACI	MKUB	Examiner

02/06/2025

Contents

List of Figures

Acknowledgments

All praise is due to **Allah**, by whose grace all good deeds are completed, and by whose favor and guidance this work was accomplished. To Him belongs all praise, first and last.

I would like to express my deepest gratitude to my supervisor, **Professor MENACER Tidjani**, for his invaluable support, insightful guidance, and continuous encouragement throughout the course of this project. His expertise and mentorship have been essential to the progress and completion of this work, and it was truly an honor to work under his supervision.

I am also sincerely thankful to the members of the jury, **Dr. Fatiha GHEDJEMIS** and **Dr. Fatma KACI**, for graciously accepting to evaluate my thesis and for their thoughtful comments and constructive feedback.

My heartfelt appreciation extends to **all the professors in our department**, whose dedication to teaching has left a lasting impact on my academic journey. I also wish to thank the administrative staff for their constant support and assistance.

Finally, I am grateful to my family and friends for their unwavering encouragement, patience, and belief in me throughout this journey. Their support has been a source of strength and motivation.

Dedicate

After thanking God for His grace and generosity

To **my father**, my primary hero and supporter, who always said, "Your success is my success, and your happiness is my happiness." I dedicate the fruits of this effort to him, honoring his endless generosity. Thank you, Dad, for your support, patience, and trust the light guiding my path. .

And to my beloved **mother**, my pillar of strength in moments of doubt, who never hesitated to uplift me with her prayers and encouragement. To you, Mom, I dedicate this humble success—a small reflection of your love and sacrifices. Thank you, **my Mommy zouzou**, from the depths of my heart.

To my dear brothers and sisters, thank you for standing by me with your support, prayers, and love that guided me every step of the way especially my sister **Sara**, my dear friend and all I hold dear, and my sister **Bina**, whose prayers were a light strengthening my resolve. And to my dear brother **Saif**, who celebrated my successes as his own and supported me from my very first steps with his patience and belief in me. I dedicate this achievement to him.

To my close friends **Mimi**, **Romaissa** and **Wafa**, who celebrated every achievement and cheered me on at every step your encouragement meant the world to me. This success is as much yours as it is mine, for you were my motivation in moments of doubt. Thank you for always believing in me. Also, to **my civil engineering friends** and **my Discord friends**.

To my dear friends, my inspiring classmates, and those I shared my college days with—**Souhila**, **Ghofrane**, **Maram**, **Khadedja**, **Hadil**, **Sana**, and **Nardjes**—you've become my sisters through your hearts and support. And to my best **PDE** class and all my mathematical colleagues, thank you for making every obstacle overcome with team spirit.

Last but not least. I wanna thank **me**. I wanna thank me for **believing in me**. I wanna thank me for doing all this **hard work**. I wanna thank me for **having no days off**.

Notation

- $\Gamma(.)$: Gamma Euler function.
- $\beta(x, y)$: Beta function.
- $E_{\alpha,\beta}(t)$: The Mittag-Leffler function.
- $\Delta^{\alpha} f(x)$: Grunwald–Letnikov difference of order p of the function $f(x)$.
- $\left(\begin{array}{c}\alpha \\ j\end{array}\right)$: Binomial term.
- $F(s) = L\{f(x); s\}$: Laplace Transform.
- $f(x) * g(x)$: Convolution of the functions $g(x)$ and $f(x)$.
- $G(s)$: Transfer function in Laplace domain.
- $C(I, \mathbb{R}^n)$: Space of continuous functions over the specified domain I.
- BIBO : Bounded-Input Bounded-Output.
- ICs: Initial conditions.
- DDE: Delay Differential Equation.
- FDDE: Fractional Delay Differential Equation.
- LTI: Linear Time-Invariant.
- LTD: Linear Time-Delay.
- FOS: Fractional-Order System.
- PID: Proportional-Integral-Derivative.
- $PI^{\lambda}D^{\mu}$: Fractional order PID.

Introduction

The study of chaos in dynamical systems emerged in the 20th century, and its beginnings are linked to the work of meteorologist Edward Lorenz in 1963 [11], when he observed that a small change in the initial conditions of a weather model leads to large changes in the results, in what later became known as the "butterfly effect." Since then, the concept of chaos has become central to understanding nonlinear systems with complex and sensitive behavior, opening vast new horizons for research across various scientific disciplines.

On the other hand, the roots of fractional calculus can be traced back several centuries, with the concept of non-integer derivatives first introduced in 1695 through a renowned correspondence between Leibniz and l'Hôpital [9]. Nonetheless, the practical applications of fractional derivatives have only gained momentum in recent decades, particularly with the advancement of numerical computation. This mathematical framework has proven to be highly effective in modeling various physical and biological phenomena that are challenging to accurately characterize using classical calculus, as it inherently captures memory effects and the historical behavior of systems.

Regarding time delays in dynamical systems, interest in their effects increased in parallel with advancements in communication and control technologies, particularly during the latter half of the 20th century [18]. Researchers observed that even small delays could induce sudden instability in industrial and communication systems. This insight led to the emergence of Delay Differential Equations (DDEs) as a vibrant field of research, owing to their critical role in modeling systems whose current states depend on their past behavior.

When these three concepts—chaos, fractional calculus, and time delay—are integrated into a single system, the result is a highly complex dynamical model that demands advanced mathematical and control tools [17]. This has led to a recent scientific trend focused on developing adaptive control techniques tailored to the unique characteristics of such systems.

This research aims to contribute to the field by designing a fractional-order time-delayed controller and analyzing the system's stability using the Lyapunov-Razumikhin method, one of the most effective tools for studying time-delayed systems [18]. The main objective of the proposed controller is to suppress and control chaos by guiding the chaotic system towards stable and regular behavior [25], despite the complexities introduced by fractional-order dynamics and time delays. Although chaos is an inherent characteristic of certain system dynamics, controlling it is important for ensuring safe and efficient operational performance. This is particularly evident in applications such as drone systems, which are increasingly used in both military and civilian sectors. However, these systems face significant challenges in unstable environments.

This thesis is divided into four main chapters:

Chapter one builds the theoretical foundation of the research. It begins by introducing the concept of fractional calculus and its historical development, followed by a presentation of key definitions such as those of Riemann–Liouville, Caputo, and Grünwald–Letnikov, along with their properties and corresponding Laplace transforms. The chapter then transitions to the study of stability in fractional-order systems, introduces the fundamental characteristics of chaos, and presents well-known models such as the Rössler and Lotka–Volterra systems. These examples illustrate how fractional derivatives influence the behavior of chaotic systems.

Chapter two addresses fractional delay differential equations (FDDEs). It first defines general delay differential equations (DDEs), then explains how incorporating fractional derivatives leads to more complex formulations. The chapter classifies the types of delays and explores stability analysis using techniques such as the Lyapunov–Razumikhin function. It also investigates bifurcation phenomena and the behaviors arising from variations in delay time.

Chapter three focuses on chaos control in fractional-order systems with time delay. It presents the design of a $PI^{\lambda}D^{\mu}$ controller with a delay component. The chapter discusses the components of the controller, control and observation strategies, and provides well-known numerical examples demonstrating how this controller helps mitigate chaotic behavior through simulations—without a formal mathematical stability analysis.

Chapter four applies the previous results to a real-world system represented by a drone system. The dynamics of the drone system are first modeled in terms of linear and rotational motion, as well as the influence of wind disturbances. Three simulation scenarios are considered: the first without wind, the second with wind and no controller, and the third with the proposed controller activated. These simulations aim to evaluate the controller's performance in reducing chaos and achieving relative stabilization of the drone system.

Chapter 1

Fractional-Order Chaotic Systems

1.1 Fractional-Order

1.1.1 Introduction to Fractional-Order Calculus

Fractional calculus is a natural development of classical calculus, allowing the use of derivatives and integrals of non-integer orders. The importance of this generalization lies in its ability to describe systems that exhibit memory- or state-history-dependent behavior, which is difficult to accurately represent using conventional integer-order models. Common applications include biological processes, and some electrical systems, where fractional calculus provides more accurate tools for representing their complex behavior. [22].

There are several definitions of the fractional derivative, the most prominent of which are **the Riemann-Liouville**, **Caputo**, and **Grünwald-Letnikov** definitions, each of which is suitable for a specific type of application. In this chapter, we review these definitions and their basic properties, then move on to their use in the analysis of chaotic systems of fractional order.

1.1.2 Fractional-Order Integrals and Derivatives

1.1.2.1 Riemann–Liouville Fractional Integral

Definition 1.1.2.1.1. *The left Riemann-Liouville fractional integral ${}_aI_x^\alpha f(x)$ of order $\alpha \in \mathbb{R}^+$ of a function $f(x)$ on the interval $[a, x]$ [22], is defined by*

$$ {}_aI_x^\alpha f(x) = \frac{1}{\Gamma(\alpha)} \int_a^x (x-t)^{\alpha-1} f(t)dt, \qquad x > a. \tag{1.1}$$

1.1.2.2 Riemann–Liouville Fractional Derivative

Definition 1.1.2.2.1. *The left Riemann-Liouville fractional derivative ${}_aD_x^\alpha f(x)$ of order $\alpha \in \mathbb{R}^+$ relies on applying a numerical derivative after performing a fractional integration[22], it is given by*

$$ {}_aD_x^\alpha f(x) = \frac{1}{\Gamma(n-\alpha)} \frac{d^n}{dx^n} \int_a^x (x-t)^{n-\alpha-1} f(t)dt, \qquad n = \lceil \alpha \rceil. \tag{1.2}$$

1.1.2.3 Caputo Fractional Derivative

Definition 1.1.2.3.1. *The left Caputo fractional derivative* ${}_a^C D_x^\alpha f(x)$ *of order* $\alpha \in \mathbb{R}^+$ *is one of the most common definitions in physical and engineering models because it allows the use of initial conditions of the classical form:*

$$ {}_a^C D_x^\alpha f(x) = {}_a D_x^\alpha \left(f(x) - \sum_{k=0}^{n-1} \frac{f^{(k)}(a)}{k!} (x-a)^k \right). \tag{1.3} $$

Where $n = \lceil \alpha \rceil$ *[22], this definition is suitable for numerical analysis and control.*

Remark 1.1.2.1. *There is also the right Riemann-Liouville fractional derivatives and integral and the right Caputo derivatives.*

1.1.2.4 Grünwald-Letnikov Fractional Derivative

Definition 1.1.2.4.1. *Let a function* $f : D \to \mathbb{R}$ *and* $f^{(k)}(x)$ *its derivative order* $k \in \mathbb{N}$ *defined by [5]*

$$ f^{(k)}(x) = \lim_{h \to 0} h^{-k} \Delta_h^k f(x), \qquad \textit{with } h^{-k} \Delta_h^k f(x) = \sum_{j=0}^{k} (-1)^j \begin{pmatrix} k \\ j \end{pmatrix} f(x - jh). $$

Specifically, one may define the fractional backward difference formula of order $\alpha \in \mathbb{R}$*, denoted by* $\Delta_{h,k}^\alpha f(x)$*, by*

$$ \Delta_{h,k}^\alpha f(x) = \sum_{j=0}^{k} (-1)^j \begin{pmatrix} \alpha \\ j \end{pmatrix} f(x - jh). $$

Definition 1.1.2.4.2. *This derivative is based on a generalization of the concept of back-difference, and is widely used in numerical modeling, especially in computer programs, and is given as*

$$ {}_a^{GL} D^\alpha f(x) = \lim h^{-\alpha} \Delta_{h,k}^\alpha f(x). \tag{1.4} $$

Remark 1.1.2.2. *The limit is achieved as* $k \to \infty$ *and* $h \to 0$ *with* $h = \dfrac{x-a}{k}$*, while keeping* $kh = x - a$ *constant value. This shows the Grünwald-Letnikov and Riemann-Liouville fractional derivatives* ***coincide*** *for* $\alpha < 0$.

1.1.3 Laplace Transform Method

The Laplace transform is a powerful mathematical tool used to analyze linear systems. It is also useful when dealing with fractional equations, especially when converting fractional derivatives or integrals to Laplace domain [21].

1.1.3.1 Basic Facts about the Laplace Transform

Another useful property we need is the formula for the Laplace transform of the derivative of an integer order n of the function $f(x)$:

$$L\{f^{(n)};s\} = s^n F(s) - \sum_{k=0}^{n-1} s^{n-k-1} f^{(k)}(0) = s^n F(s) - \sum_{k=0}^{n-1} s^k f^{(n-k-1)}(0). \tag{1.5}$$

1.1.3.2 Laplace Transform of Fractional Integrals

We begin with the Laplace transform of the Riemann-Liouville fractional integral of order $\alpha > 0$ defined by (1.1), which we can write as a convolution of the functions $g(x) = \frac{x^{\alpha-1}}{\Gamma(\alpha)}$ and $f(x)$

$${}_0I_x^\alpha f(x) = {}_0D_x^{-\alpha} f(x) = \frac{1}{\Gamma(\alpha)} \int_0^x (x-t)^{\alpha-1} f(t)dt = g(x) * f(x). \tag{1.6}$$

Therefore, by applying the Laplace transform of convolution, we obtain the Laplace transform of the Riemann-Liouville and the Grünwald-Letnikov fractional integrals:

$$L\{{}_0D_x^{-\alpha} f(x); s\} = s^{-\alpha} F(s). \tag{1.7}$$

1.1.3.3 Laplace Transform of Fractional Derivatives

The Laplace transform of the Riemann-Liouville fractional derivative, which, for our purpose, we express in the form:

$${}_0D_x^\alpha f(x) = g^{(n)}(x). \tag{1.8}$$

Thus

$$g(x) = {}_0D_x^{-(n-\alpha)} f(x) = \frac{1}{\Gamma(\alpha-n)} \int_0^x (x-t)^{\alpha-n-1} f(t)dt, \qquad n-1 \le \alpha < n. \tag{1.9}$$

The use of the formula for the Laplace transform of an integer-order derivative(1.5) leads to

$$L\{g^{(n)}(x); s\} = L\{{}_0D_x^\alpha f(x); s\} = s^n G(s) - \sum_{k=0}^{n-1} s^k g^{(n-k-1)}(0). \tag{1.10}$$

The Laplace transform of the function $g(x)$ is evaluated by (1.7):

$$G(s) = s^{-(n-\alpha)} F(s). \tag{1.11}$$

From the definition of the Riemann-Liouville fractional derivative (1.2) it follows that

$$g^{(n-k-1)}(x) = \frac{d^{n-k-1}}{dt^{n-k-1}} {}_0D_x^{-(n-\alpha)} f(x) = {}_0D^{\alpha-k-1} x f(x). \tag{1.12}$$

Substituting (1.11) and (1.12) into (1.10) we obtain

$$\begin{aligned} L\{{}_0D_x^{\alpha}f(x);s\} =& s^n s^{-(n-\alpha)}F(s) - \sum_{k=0}^{n-1} s^k \left[{}_0D_x^{\alpha-k-1}f(x)\right]_{x=0}, \\ =& s^{\alpha}F(s) - \sum_{k=0}^{n-1} s^k \left[{}_0D_x^{\alpha-k-1}f(x)\right]_{x=0}. \end{aligned} \tag{1.13}$$

The formula for the Laplace transform of the Caputo fractional derivative

$$L\{{}_0^C D_x^{\alpha}f(x);s\} = s^{\alpha}F(s) - \sum_{k=0}^{n-1} s^{\alpha-k-1}f^{(m)}(0). \tag{1.14}$$

For zero initial conditions, the Laplace transform of fractional derivatives of order r (Grünwald-Letnikov, Riemann-Liouville, and Caputo's) reduces to:

$$L\{{}_0D_x^{r}f(x);s\} = s^{r}F(s), \qquad n-1 \leq r < n. \tag{1.15}$$

1.1.4 Fractional Differential Equation

The most popular fractional order derivative has been given by Riemann-Liouville and Caputo, [4] fractional order differential equations involving Riemann-Liouville's fractional order derivative has some practical issues, related to initial value problems, is expressed as

$$\begin{cases} D^{\alpha}y(t) = f(t, y(t)), \\ D^{\alpha-k}y(0) = b_k \qquad (k = 1, ..., n-1), \qquad \lim\limits_{z\to 0^+} I^{n-\alpha}y(z) = b_n, \end{cases} \tag{1.16}$$

where $n-1 < \alpha < n$.

1.1.4.1 Existence and Uniqueness of Solution

The study of the existence and uniqueness of solutions for fractional-order differential equations is fundamental in mathematical analysis.

Theorem 1.1.4.1. *Let $\alpha > 0$, $\alpha \notin \mathbb{N}$ and $n = [\alpha]$, let $K > 0$, $h^* > 0$, and $b_1, ..., b_n \in \mathbb{R}$, define*

$$G := \left\{(t, y) \in \mathbb{R}^2 : 0 \leq t \leq h^*, y \in \mathbb{R} \text{ for } t = 0 \text{ and } \left| t^{n-\alpha} - \sum_{k=1}^{n} \frac{b_k t^{n-k}}{\Gamma(\alpha - k + 1)} \right| < K \right\},$$

let a function $f : G \to \mathbb{R}$ be a continuous, bounded function satisfying a Lipschitz condition with respect to the second variable [16]. That is, there exists a constant $L > 0$ such that for all (t, y_1) and $(t, y_2) \in G$, we have

$$|f(t, y_1) - f(t, y_2)| \leq L|y_1 - y_2|.$$

Then the differential equation of Riemann-Liouville (1.16) has a uniquely defined continuous *solution* $y \in C(0, h]$ *where*

$$h := min\left\{h^*, \tilde{h}, \left(\frac{\Gamma(\alpha+1)K}{M}\right)^{\frac{1}{n}}\right\},$$

with $M = \sup_{(t,z)\in G} |f(t, z)|$ *and* $\tilde{h}$ *being an arbitrary positive number satisfying the constraint*

$$\tilde{h} < \left(\frac{\Gamma(2\alpha - n + 1)}{\Gamma(\alpha - n + 1)L}\right)^{\frac{1}{\alpha}}.$$

The proof of existence and uniqueness follows classical methods similar to those used for first-order equations. It reformulates the initial value problem as a Volterra integral equation and applies an iterative process based on Banach's fixed-point theorem.

1. **Transform the Differential Equation into an Integral Equation**

   Lemma 1.1.4.1. *Assume the hypotheses of Theorem 1.1.4.1 and let* $h > 0$*. The function* $y \in (0, h]$ *is a solution of the differential equation (1.16), if and only if it is a solution of the Volterra* ***integral equation***

   $$y(t) = \sum_{k=1}^{n} \frac{b_k t^{\alpha-k}}{\Gamma(\alpha - k + 1)} + \frac{1}{\Gamma(\alpha)} \int_0^t (t-\tau)^{\alpha-1} f(\tau, y(\tau))d\tau. \tag{1.17}$$

   Proof 1.1.4.1.1. *(See [16], page 79.)*

2. **Prove That the Operator** A **Maps the Function Space into Itself**

   Lemma 1.1.4.2. *Under the assumptions of Theorem 1.1.4.1, the Volterra equation (1.17) possesses a uniquely determined solution* $y \in C(0, h]$.

   Proof 1.1.4.1.2. *We define the set*

   $$B := \left\{y \in C(0, h] : \sup_{0<t\leq h} \left|t^{n-\alpha}y(t) - \sum_{k=1}^{n} \frac{b_k t^{n-k}}{\Gamma(\alpha - k + 1)}\right| \leq K\right\},$$

   and on this set we define the operator A *by*

   $$Ay(t) = \sum_{k=1}^{n} \frac{b_k t^{\alpha-k}}{\Gamma(\alpha - k + 1)} + \frac{1}{\Gamma(\alpha)} \int_0^t (t-\tau)^{\alpha-1} f(\tau, y(\tau))d\tau.$$

   Then we note that, for $y \in B$*,* Ay *is also a continuous function on* $(0, h]$*.Moreover*

   $$\left|t^{n-\alpha}Ay(t) - \sum_{k=1}^{n} \frac{b_k t^{n-k}}{\Gamma(\alpha - k + 1)}\right| \leq \frac{t^n M}{\Gamma(\alpha+1)} \leq K.$$

   For $t \in (0, h]$*, where the last inequality follows from the definition of* h*. This shows that* $Ay \in B$ *if* $y \in B$ *, i.e. the operator* A *maps the set* B *into itself.*

3. **Prove That A is a Contraction (Lipschitz Condition)**
   We introduce a new set

$$\hat{B} = \left\{ y \in C(0,h) : \sup_{0<t\leq h} |t^{n-\alpha} y(t)| < \infty \right\},$$

on which we define a norm $\|\cdot\|_{\hat{B}}$ by

$$\|y\|_{\hat{B}} = \sup_{0<t\leq h} |t^{n-\alpha} y(t)|.$$

Using the definition of A, the Volterra equation can be expressed in a more compact form

$$y = Ay.$$

To prove the desired relation, it is sufficient to show that the operator A has a unique fixed point. For this, we will use the Weissinger fixed-point theorem (Appendix A). We will prove that for any $y, \tilde{y} \in B$,

$$\|A^j y - A^j \tilde{y}\|_{\hat{B}} \leq \left(\frac{Lh^\alpha \Gamma(\alpha - n + 1)}{\Gamma(2\alpha - n + 1)} \right)^j \|\tilde{y} - y\|_{\hat{B}}. \tag{1.18}$$

Proof 1.1.4.1.3. *(See [16], page 81.)*

4. **Apply the Fixed-Point Theorem**

Theorem 1.1.4.2. *(Banach's Fixed-Point)*
Let (U, d) be a non-empty complete metric space, and suppose that $0 \leq \alpha < 1$. Let $A : U \to U$ be a mapping that satisfies the inequality

$$d(Au, Av) \leq \alpha d(u, v)$$

for all $u, v \in U$. Then, A has a unique fixed point u^*. Moreover, for any $u_0 \in U$, the sequence $(A^j u_0)_{j=1}^{\infty}$ converges to this fixed point u^*.

Using induction, we derive relation (1.18). By applying Theorem 1.1.4.2 with $\alpha_j = \gamma^j$, where $\gamma = \left(\frac{Lh^\alpha \Gamma(\alpha - n + 1)}{\Gamma(2\alpha - n + 1)} \right)$, the series $\sum_{j=0}^{\infty} \alpha_j$ converges since $h \leq \tilde{h}$ implies $\gamma < 1$. The fixed-point theorem ensures a unique solution to the integral equation.

1.1.4.2 Solution of the Fractional Differential Equation

The resolution of fractional differential equations relies on a set of complementary approaches, including the numerical methods, the analytical methods, as well as Laplace's method.

- **Solution of the Fractional Differential Equation Using Laplace Transform**
  Consider the following fractional differential equation (1.16), Laplace transform of the

Caputo fractional derivative is given as,

$$L\{{}_0D_t^{\alpha}y(t);s\} = s^{\alpha}Y(s) - \sum_{k=0}^{n-1} s^{\alpha-k-1}{}_0D_t^{k}y(0). \tag{1.19}$$

However, the initial conditions are given in terms of $D^{\alpha-k}y(0)$, for the Caputo derivative, the Laplace transform can be written as:

$$L\{{}_0D_t^{\alpha}y(t);s\} = s^{\alpha}Y(s) - \sum_{k=1}^{n} s^{k-1}{}_0D_t^{\alpha-k}y(0). \tag{1.20}$$

where $D^{\alpha-k}y(0) = b_k$ for $(k = 1, ..., n-1)$.

Taking Laplace transform of (1.16), one can write

$$s^{\alpha}Y(s) - \sum_{k=1}^{n} s^{k-1}b_k = F(s) \quad \Longrightarrow \quad Y(s) = \frac{F(s)}{s^{\alpha}} + \sum_{k=1}^{n} \frac{b_k}{s^{\alpha-k+1}}. \tag{1.21}$$

Using the inverse Laplace transformation, with :

$$L^{-1}\left\{\frac{1}{s^{\beta}}\right\} = \frac{t^{\beta-1}}{\Gamma(\beta)}. \tag{1.22}$$

Using (1.22) in the inverse Laplace transformation of (1.21) can be found as,

$$y(t) = \sum_{k=1}^{n} b_k \frac{t^{\alpha-k}}{\Gamma(\alpha-k+1)} + \frac{1}{\Gamma(\alpha)} \int_0^t (t-\tau)^{\alpha-1} f(\tau, y(\tau))\, d\tau. \tag{1.23}$$

1.2 Chaos in Fractional-Order Systems

1.2.1 Fractional-Order System

A general fractional-order system can be described by a fractional differential equation of the form [9]

$$\begin{aligned} a_nD^{\alpha_n}y(t)+&a_{n-1}D^{\alpha_{n-1}}y(t) + \cdots + a_0D^{\alpha_0}y(t) \\ &= b_mD^{\beta_m}u(t) + b_{m-1}D^{\beta_{m-1}}u(t) + \cdots + b_0D^{\beta_0}u(t). \end{aligned} \tag{1.24}$$

This system illustrates the relationship between input $u(t)$ and output $y(t)$ in a system with fractional dynamics

1.2.1.1 Types of Fractional-Order System

Fractional systems can be classified into two main types according to the nature of the derivation orders:

1 **Incommensurate Systems:** in which different orders are not necessarily related to each other (α_k, and β_k). This classification affects stability analysis and the complexity of solutions.

2 **Commensurate Systems:** The fractional orders are multilples of a common base α, (i.e $\alpha_k = \alpha k$ and $\beta_k = \alpha k, \forall k \in \mathbb{Z}$).

1.2.1.2 Fractional Linear Time-Invariant (LTI) Systems

A fractional LTI system is a dynamical system described by fractional-order differential equations that are linear and time-invariant. The system (1.24) can also be represented in the Laplace domain by a transfer function.
Incommensurate Order: The transfer function is

$$G(s) = \frac{b_m s^{\beta_m} + b_{m-1} s^{\beta_{m-1}} + \cdots + b_0 s^{\beta_0}}{a_n s^{\alpha_n} + a_{n-1} s^{\alpha_{n-1}} + \cdots + a_0 s^{\alpha_0}} = \frac{Q(s^{\beta_k})}{P(s^{\alpha_k})} = \frac{Y(s)}{U(s)}. \tag{1.25}$$

Commensurate Order: For $\alpha_k = \alpha k$ and $\beta_k = \alpha k$, $(0 < \alpha < 1)$, the transfer function is:

$$G(s) = \frac{\sum_{K=0}^{m} b_k (s^\alpha)^k}{\sum_{k=0}^{n} a_k (s^\alpha)^k} = K_0 \frac{Q(s^\alpha)}{P(s^\alpha)}, \qquad \forall k \in \mathbb{Z}, \tag{1.26}$$

the function $G(s)$ becomes a proper rational function in the complex variable s^α which can be expanded in partial fractions of the following form

$$G(s) = K_0 \left[\sum_{i=1}^{n} \frac{A_i}{s^\alpha + \lambda_i} \right], \tag{1.27}$$

where $\lambda_i (i = \overline{1, n})$ are the roots of the $P(s^\alpha) = 0$.
The analytical solution of the system (1.24)

$$y(t) = L^{-1} \left\{ K_0 \left[\sum_{i=1}^{n} \frac{A_i}{s^\alpha + \lambda_i} \right] \right\} = K_0 \sum_{i=1}^{n} A_i t^\alpha E_{\alpha,\alpha}(-\lambda_i t^\alpha). \tag{1.28}$$

Remark 1.2.1.2.1. ***The transfer function*** *is used to transform an ordinary or fractional differential equation into an algebraic equation, typically in the Laplace domain.*

Example 1.2.1.1. *Let's assume we have the following system*

$$\begin{cases} D^{0.5} y_1(t) + 2y_1(t) - y_2(t) = u_1(t), \\ D^{0.5} y_2(t) + 3y_2(t) + y_1(t) = u_2(t). \end{cases}$$

Apply the Laplace transform to the fractional differential equations

$$\begin{cases} (s^{0.5} + 2) Y_1(s) - Y_2(s) = U_1(s), \\ Y_1(s) + (s^{0.5} + 3) Y_2(s) = U_2(s), \end{cases}$$

with $Y_i(s) = L\{y_i(t); s\}$ *and* $U_i(s) = L\{u_i(t); s\}$, $i = 1, 2$.

Write the system in its matrix form

$$\begin{bmatrix} s^{0.5}+2 & -1 \\ 1 & s^{0.5}+3 \end{bmatrix} \begin{bmatrix} Y_1(s) \\ Y_2(s) \end{bmatrix} = \begin{bmatrix} U_1(s) \\ U_2(s) \end{bmatrix}.$$

Use Cramer's rule to solve for $Y_1(s)$ and $Y_2(s)$ in the system

$$Y_1(s) = \frac{U_1(s)(s^{0.5}+3)+U_2(s)}{s+5s^{0.5}+7} \quad \text{and} \quad Y_2(s) = \frac{U_2(s)(s^{0.5}+2)-U_1(s)}{s+5s^{0.5}+7}.$$

Decompose the transfer function into partial fractions

$$s+5s^{0.5}+7 = (s^{0.5}+\lambda_1)(s^{0.5}+\lambda_2) = \left(s^{0.5}+\frac{5-i\sqrt{3}}{2}\right)\left(s^{0.5}+\frac{5+i\sqrt{3}}{2}\right).$$

Using the inverse Laplace transform (1.22), we obtain

$$\begin{cases} y_1(t) = t^{-0.5}\left[A_1 E_{0.5,0.5}\left(\frac{5+i\sqrt{3}}{2}t^{0.5}\right) + A_2 E_{0.5,0.5}\left(\frac{5-i\sqrt{3}}{2}t^{0.5}\right)\right], \\ y_2(t) = t^{-0.5}\left[B_1 E_{0.5,0.5}\left(\frac{5+i\sqrt{3}}{2}t^{0.5}\right) + B_2 E_{0.5,0.5}\left(\frac{5-i\sqrt{3}}{2}t^{0.5}\right)\right]. \end{cases}$$

1.2.1.3 Nonlinear Fractional-Order Systems

In general, an incommensurate fractional-order nonlinear system can be expressed as:

$$\begin{cases} D_t^{\alpha_i} y_i(t) = f_i(y_i(t), t), \\ y_i(0) = c_i, \qquad i = \overline{1,n}. \end{cases} \tag{1.29}$$

where f is nonlinear function $\alpha = [\alpha_1, \alpha_2, ..., \alpha_n]^t$ for $0 < \alpha_i < 2, i = \overline{1,n}$ and $y \in \mathbb{R}^n$.

1.2.1.4 Equilibrium Point

Consider the following fractional-order system

$$D_t^{\alpha_i} y_i = f_i(y_i), \qquad i = \overline{1,n}.$$

The equilibrium points of this system are determined by solving the equation $f(y) = 0$, thus, the point $y^* = (y_1^*, y_2^*, ..., y_n^*)$ is considered an equilibrium point of the system.

1.2.1.5 Stability Analysis of Fractional LTI Systems

In the fractional case, the stability of the system (1.24) is different from the integer case. Let elements of $G(s)$ are transfer functions (1.26) with a common polynomial denominator given by

$$P(s^\alpha) = 0.$$

Definition 1.2.1.5.1. *(Poles of the System) [20]*
In a fractional-order system, the poles of the transfer function determine stability. This is analyzed by examining the denominator $P(s^\alpha)$ and its roots λ_i(the eigenvalues)

$$P(s^\alpha) = \prod_{i=1}^{n}(s^\alpha + \lambda_i),$$

where poles is $s = p_i = (\lambda_i)^{1/\alpha}, \quad i = \overline{1, n}.$

Theorem 1.2.1.1. *(Stability Condition) [20]*
*For a system to be **bounded-input bounded-output** (BIBO) stable, all its poles must lie in the left half of the s-complex plane. This means:*

$$|arg(p_i)| > \frac{\pi}{2}, \quad 1 \leq i \leq n.$$

Using the relation $p_i = (\lambda_i)^{1/\alpha}$ the condition can be rewritten in terms of the eigenvalues as follows:

$$|arg(\lambda_i)| > \alpha\frac{\pi}{2}, \quad 1 \leq i \leq n.$$

If $\alpha = \dfrac{1}{q}$, $q \in \mathbb{Q}$, the stability conditions simplifies to :

$$|arg(\lambda_i)| > \frac{\pi}{2q}, \quad 1 \leq i \leq n.$$

1.2.1.6 Stability Analysis of Nonlinear Systems

- **Stability by Linearization** [13]
  Consider the following fractional-order system

$$D_t^{\alpha_i} y_i = f_i(y_i), \qquad i = \overline{1, n}. \tag{1.30}$$

  Where $y_i^* = (y_1^*, y_2^*, ..., y_n^*)$ are equilibrium points of the system (1.30), to analyze stability, let small perturbations ϵ_i are introduced around equilibrium points:

$$\epsilon_i = y_i - y_i^* \qquad i = \overline{1, n}.$$

  Using Taylor series expansion, the function is approximated around the equilibrium, leading to the Jacobian matrix $J = \left[\dfrac{\partial f_i}{\partial y_i}\right]_{y^*}$. This means

$$f(\epsilon_i + y_i^*) = f(y_i^*) + \left[\frac{\partial f_i}{\partial y_i}\right]_{y^*} \epsilon_i + h(\epsilon_i), \qquad i = \overline{1, n},$$

where h is continuous function such that $h(\epsilon_i) = 0(\|\epsilon_i\|^2)$ and $f(y_i^*) = 0$, so we get

$$f(\epsilon_i + y_i^*) \backsimeq \left[\frac{\partial f_i}{\partial y_i}\right]_{y^*} \epsilon_i, \qquad i = \overline{1, n}.$$

Thus, we obtain

$$D_t^{\alpha} \epsilon = J\epsilon.$$

Now, we can apply the stability condition Theorem 1.2.1.1 to analyze the local stability of equilibrium solutions of the system (1.30).

Remark 1.2.1.6.1. *(Topological Equivalence)*
In the study of dynamical systems, a system is said to be topologically stable if there exists ***a homeomorphism*** *(a continuous, bijective function with a continuous inverse) that maps the nonlinear system to its linearized version.*

1.2.2 Chaos

Chaos theory studies deterministic systems that can exhibit unpredictable behavior due to their extreme sensitivity to initial conditions. This behavior becomes even more complex when modeled using fractal mathematical tools that add memory effects, making it difficult to accurately predict the system's path. The modern aspects of the theory emerged in the 1960s thanks to Edward Lorenz, who in 1963 discovered a chaotic attractor while simulating weather models, revealing what is now known as the butterfly effect—in which small changes lead to large consequences. The theory has become widely applied in various fields such as engineering, artificial intelligence, neuroscience, and finance[10] [11].

1.2.2.1 Characteristics of Chaos

Several key properties define a chaotic system [10]:

- **Sensitivity to Initial Conditions:** A tiny change in the initial conditions of the system can lead to drastically different outcomes over time, like differences in the behavior of the system.
- **The unpredictable:** In chaos theory, unpredictability refers to the inherent difficulty or impossibility of predicting the long-term behavior of a system.
- **Non-linearity:** Chaotic systems are nonlinear, meaning that their evolution cannot be simply predicted by summing up smaller effects, Not all nonlinear systems are chaotic.
- **Parameter Sensitivity:** A small change in system parameters can lead to a drastic change in system behavior.
- **Fractal Structure:** In chaos theory, a fractal structure refers to a complex, self-similar geometric pattern that repeats at different scales. It is characterized by their intricate, repeating shapes.

- **Strange Attractors:** These are complex, fractal-like pattern in chaotic systems, and showing structured yet unpredictable motion within a bounded space.

1.2.3 Examples of Fractional-Order Chaotic Systems

1.2.3.1 Fractional-Order Rössler's System

The fractional Rössler system is one of the most famous examples that enables the study of chaos in systems with memory. This system exhibits rich nonlinear behavior and is distinguished by its ability to produce chaotic attractors with derivative orders less than 1, the effect of the fractional dimension on the complexity of the dynamics. The system is defined as [21]:

$$\begin{cases} D_t^{\alpha_1} x(t) = & -(y(t)+z(t)), \\ D_t^{\alpha_2} y(t) = & x(t)+ay(t), \\ D_t^{\alpha_3} z(t) = & b+z(t)(x(t)-c), \end{cases} \tag{1.31}$$

where $\alpha_1, \alpha_2, \alpha_3$ are derivative orders.

The Rössler system has two equilibrium points :

$$e_1 = \left(\frac{c-\sqrt{c^2-4ab}}{2}, \frac{-c+\sqrt{c^2-4ab}}{2a}, \frac{c-\sqrt{c^2-4ab}}{2a} \right),$$
$$e_2 = \left(\frac{c+\sqrt{c^2-4ab}}{2}, \frac{-c-\sqrt{c^2-4ab}}{2a}, \frac{c+\sqrt{c^2-4ab}}{2a} \right).$$

Typical values are $\alpha_1 = \alpha_2 = \alpha_3 = 0.9$ and for $(a,b,c)^T = (0.4, 0.2, 7.5)^t$ with ICs $(x_0, y_0, z_0)^T = (0.5, 1.5, 0.1)^T$, the system can display a chaotic attractor, and numerical simulations of the Rössler system is depicted in Figure 1.1.

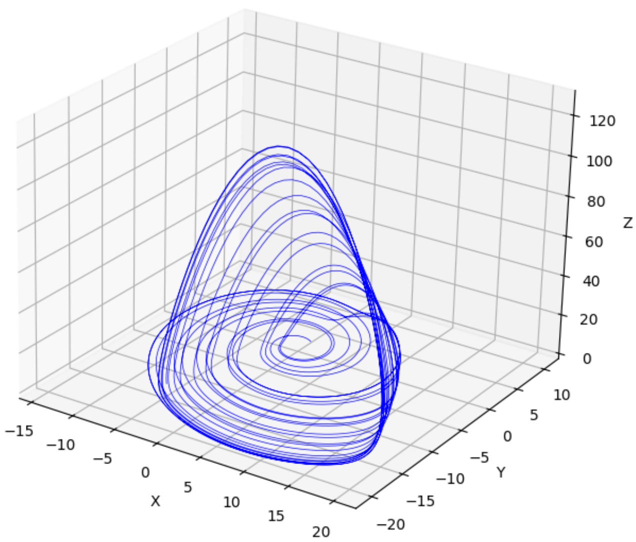


Figure 1.1: 3D Trajectory Plot of Fractional-Order Rössler's System.

1.2.3.2 Fractional-Order Lotka-Volterra System

The Fractional-Order Chaotic Lotka-Volterra System extends the predator-prey model with fractional derivatives, adding memory effects and chaotic dynamics. It models complex interactions in ecosystems, chemical reactions, and economics, describing behaviors like population fluctuations and oscillatory patterns. This makes it a key tool in biology, ecology, and nonlinear dynamics research. The system is defined as: [21]

$$\begin{cases} D_t^{\alpha_1} x(t) = & ax(t) - bx(t)y(t) + ex^2(t) - sz(t)x^2(t), \\ D_t^{\alpha_2} y(t) = & -cy(t) + dx(t)y(t), \\ D_t^{\alpha_3} z(t) = & -pz(t) + sz(t)x^2(t), \end{cases} \tag{1.32}$$

where $\alpha_1, \alpha_2, \alpha_3$ are derivative orders, with a, b, c, d, e, p, s are model parameters

The Lotka-Volterra system has five equilibrium points:

$$e_1 = (0,0,0)^T, \qquad e_2 = \left(-\frac{a}{e}, 0, 0\right)^T, \qquad e_3 = \left(\frac{\sqrt{sp}}{s}, 0, \frac{a + \frac{e\sqrt{sp}}{s}}{\sqrt{sp}}\right)^T,$$

$$e_4 = \left(-\frac{\sqrt{sp}}{s}, 0, -\frac{a - \frac{e\sqrt{sp}}{s}}{\sqrt{sp}}\right)^T, \qquad e_5 = \left(\frac{c}{d}, \frac{da + ec}{db}, 0\right)^T.$$

Typical values are $\alpha_1 = \alpha_2 = \alpha_3 = 0.9$ and for $(a, b, c, d, e, p, s)^T = (1, 1, 1, 1, 2, 3, 2.7)^T$, with ICs $(x_0, y_0, z_0)^T = (1, 1.4, 1)^T$, the system can display a chaotic attractor, and numerical simulations of the Lotka-Volterra system is depicted in Figure 1.2.

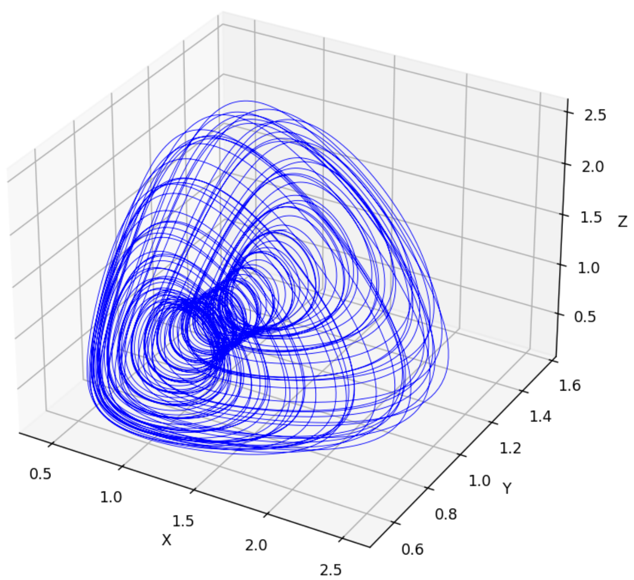


Figure 1.2: 3D Trajectory Plot of Fractional-Order Lotka-Volterra System.

Chapter 2

Fractional Delay Differential Equations

2.1 Delay Differential Equations (DDEs)

2.1.1 Introduction to DDEs

Delay differential equations (DDEs) are essential mathematical models for describing systems whose current state depends not only on current conditions but also on their past evolution. Unlike ordinary differential equations (ODEs), DDEs take into account the effect of time delays, reflecting more realistic behavior in many applications such as biological, economic, and engineering systems[18]. For example, this type of effect can be observed in processes such as reforestation, which takes many years, or in physiological processes such as digestion, which require time before their effects are apparent.

2.1.2 Definitions of DDEs

Definition 2.1.2.1. *In delay differential equations, the time derivative of a function does not depend solely on its value at the present moment, but includes effects from its* ***past*** *behavior, which makes them suitable for describing phenomena that respond slowly or gradually.*

Definition 2.1.2.2. *Let $t_0 \in \mathbb{R}$ and $A > 0$, if Ω is a subset of $[t_0, t_0 + A] \times \mathcal{C}([-\tau, 0], \mathbb{R}^n)$, $f : \Omega \to \mathbb{R}^n$ is a given function that describes the relationship between the current state at time t and its past states y_t[24], the general form given is*

$$\begin{cases} \dot{y}(t) = f(t, y_t), & t \geq t_0, \\ y(t) = \phi(t), & t \in [t_0 - \tau, t_0], \end{cases} \tag{2.1}$$

where $\tau > 0$ is the delay term, and $y(t) \in \mathbb{R}^n$, the history function is given by $y_t(s) = y(t+s)$ for $s \in [-\tau, 0]$ with $\phi(t)$ is the initial function defined on $[t_0 - \tau, t_0]$.

2.1.3 Types of DDE

Delay equations can be classified into several types, depending on their dependence on time history

1. **Ordinary DDE :** These depend on the current state of the function as well as its value at a single point in time in the past. [14].
2. **Multiple DDE:** These consider the values of the function at multiple points in time, adding greater complexity to the system's behavior [14].
3. **Neutral DDE :** These affect not only the function itself but also its derivatives at past times [14].
4. **Variable DDE :** These equations, the delay is not constant but depends on time [24].
5. **Distributed DDE :** These equations, the derivative depends on an integral of the past states over a time interval [19].
6. **Fractional DDE :** Combine time delays and fractional derivatives, allowing a more realistic representation of systems with long-term memory behavior.s [19].

2.1.4 Existence and Uniqueness of Solution

Theorem 2.1.4.1. *(**Existence** [12])*
*Let $t_0 \in \mathbb{R}$ and $A > 0$, if Ω is a subset of $[t_0, t_0 + A] \times \mathcal{C}([-\tau, 0], \mathbb{R}^n)$, and assume that the function f is **continuous** on Ω. If $(t_0, \phi) \in \Omega$, then there **exists** a solution $y(t)$ of the DDE (2.1) that passes through the initial condition (t_0, ϕ).*
*Additionally, if the function $f(t, \phi)$ is **Lipschitz** in ϕ inside a compact set $X \subset [t_0, t_0 + A] \times \mathcal{C}([-\tau, 0], \mathbb{R}^n)$, meaning there exists a constant k such that for any $(t, \phi_i) \in K \subset X$, $i = 1, 2$*

$$|f(t, \phi_1) - f(t, \phi_2)| \leq k\, |\phi_1 - \phi_2|\,.$$

Theorem 2.1.4.2. *(**Uniqueness** [12])*
*Let $t_0 \in \mathbb{R}$ and $A > 0$, if Ω is a subset of $[t_0, t_0 + A] \times \mathcal{C}([-\tau, 0], \mathbb{R}^n)$, $f : \Omega \to \mathbb{R}^n$ is continuous, and $f(t, \phi)$ is **Lipschitz** in ϕ in each compact set in Ω. If $(t_0, \phi) \in \Omega$, then there is a **unique** solution of (2.1) through (t_0, ϕ).*

2.1.5 Linear Time-Delay (LTD) Systems

An LTD system is a dynamical system where the evolution of the system's state depends on its current state and its past states, and this dependence is linear. Mathematically, an LTD system can be described by a linear DDEs of the form

$$y'(t) = Ay(t) + By(t - \tau) + Cu(t), \tag{2.2}$$

where

- $y(t) \in \mathbb{R}^n$ is the state vector at time t, and $y(t-\tau) \in \mathbb{R}^n$ is the delayed state vector at time $t-\tau$, with $u(t) \in \mathbb{R}^m$ is the input vector, where $\tau > 0$ represents the time delay.
- $A \in \mathbb{R}^{n\times n}$ is the system matrix, $B \in \mathbb{R}^{n\times n}$ is the delay matrix, and $C \in \mathbb{R}^{n\times m}$ is the input matrix.

The system (2.2) can also be represented in the Laplace domain by a transfer function.

We obtain

$$sY(s) - y(0) = AY(s) + Be^{-s\tau}Y(s) + CU(s), \tag{2.3}$$

where

- $L\{y'(t); s\} = sY(s) - y(0)$ and $L\{y(t); s\} = Y(s)$.
- $L\{u(t); s\} = U(s)$.
- $L\{y(t-\tau); s\} = L\{y(v); s\} = e^{-s\tau}Y(s) + e^{-s\tau}\int_{-\tau}^{0} y(v)e^{-sv}dv = e^{-s\tau}Y(s)$.

  Remark 2.1.5.1. *Most physical systems and signals are causal, as they respond only after $t = 0$ since causality states that $y(t) = 0$ for $t < 0$, we conclude that*

$$\int_{-\tau}^{0} y(v)e^{-sv}dv = 0.$$

If we assume $y(0) = 0$, the transfer function is:

$$G(s) = \frac{Y(s)}{U(s)} = C\left(sI - A - Be^{-s\tau}\right)^{-1}, \tag{2.4}$$

where $I \in \mathbb{R}^{n\times n}$ is the identity matrix.

Remark 2.1.5.2. *The transfer function $G(s)$ describes the input-output relationship of the system, independent of initial conditions. This is useful for analyzing system behavior (e.g., stability)* ***without needing to consider specific initial states.***

2.1.6 Stability Analysis of LTD Systems

In an LTD System (2.2), stability refers to whether small perturbations in the system remain bounded or grow unbounded over time. One common approach to analyze stability in linear time-delay differential equations (DDEs) is through the transfer function (2.4), which helps assess the system's behavior in the Laplace domain.
To analyze the stability of the system (2.2), we examine the poles of the transfer function $G(s)$

(2.4). Stability is determined by the poles of $G(s)$. The poles are the roots of the denominator of $G(s)$, which correspond to the values of s that satisfy the characteristic equation:

$$det\left(sI - A - Be^{-s\tau}\right) = 0.$$

Theorem 2.1.6.1. *The linear time-delay system (2.2) is stable if and only if all poles s satisfy $Re(s) < 0$.*

The difficulty lies in the fact that the variable appears s within the exponential power $e^{-s\tau}$, making the characteristic equation non-algebraic, and unable to be solved directly using conventional methods. In this case, we have many methods, some of them are theoretical and some of them are numerical, if A and B are non-commutative ($AB \neq BA$), we can do numerical methods like **(Rekasius Substitution method or Kronecker Multiplication method** [3]**)**, and if A and B commute ($AB = BA$), we can do theoretical method like the Lambert W function is used as a tool to analyze the roots of these equations and extract possible values s.

2.1.6.1 Stability by Lambert W Function

Definition 2.1.6.1. *(Lambert W Function)*
The Lambert function, denoted as $W(s)$, is a special function that solves equations of the form:

$$W(s)e^{W(s)} = s.$$

Let an LTD system (2.2), and let its characteristic equation $det(sI - A - Be^{-s\tau}) = 0$, if A and B commute ($AB = BA$) they share the same eigenvectors. This allows simultaneous diagonalization of the matrices, reducing the matrix equation to a set of decoupled scalar equations.

$$det(sI - A - Be^{-s\tau}) = \prod_{i=1}^{n}\left(s - \lambda_i - \beta_i e^{-s\tau}\right) = 0,$$

where A has an eigenvalue λ_i, and B has an eigenvalue β_i. Each term $(s - \lambda_i - \beta_i e^{-s\tau}) = 0$ can then be solved using the Lambert W function, with multiply by τ, such that

$$(s - \lambda_i)\tau = \beta_i \tau e^{-s\tau}.$$

Let $z = (s - \lambda_i)\tau$, we obtain

$$z = \beta_i \tau e^{-(z+\tau\lambda_i)}.$$

Then, the equation simplifies to

$$ze^{z} = \beta_i \tau e^{-\tau\lambda_i}.$$

The solution is given by the Lambert W function

$$z = W(\beta_i \tau e^{-\tau\lambda_i}).$$

Thus, the specific poles s_i are

$$s_i = \lambda_i + \frac{W(\beta_i \tau e^{-\tau\lambda_i})}{\tau}.$$

The Lambert W function is multi-valued, meaning it has infinitely many branches W_k for $k \in \mathbb{Z}$, such that

$$s_{i,k} = \lambda_i + \frac{W_k(\beta_i \tau e^{-\tau\lambda_i})}{\tau}.$$

Theorem 2.1.6.1. *The stability of the system depends on the real parts of the poles $s_{i,k}$*

- *If $Re(s_{i,k}) < 0$ for all i and k, the system is stable.*
- *If $Re(s_{i,k}) > 0$ for all i and k, the system is unstable.*

2.1.7 Stability Analysis of Nonlinear Systems

Consider system (2.1), assuming that the function f is **nonlinear**. The stability analysis of delay differential equations (DDEs) is more challenging than for ordinary differential equations (ODEs), in that the state of the system (2.1) is not just a vector $y(t)$ in $\mathbb{R}^n$ but rather a function y_t that depends on past values $y(t+s)$ for $s \in [-\tau, 0]$. Since traditional Lyapunov functionals require conditions on the entire function history y_t, this makes the analysis more complex, as we do not directly control the relationship between $|y(t)|$ and $|y(t+s)|$. However, **Lyapunov-Razumikhin theorems** simplify this process by focusing on how a function $V(y)$ changes along solutions without requiring conditions on the entire function history.

2.1.7.1 Stability by Lyapunov-Razumikhin Approach

Definition 2.1.7.1. *(Lyapunov Function $V(y)$)*
Let $V(y) : \mathbb{R}^n \to \mathbb{R}$, be a continuously differentiable, positive definite function. The derivative of $V(y)$ along the solutions of DDE is given by:

$$\dot{V}(y(t)) = \frac{\partial V(y)}{\partial y} f(y(t)).$$

This measures how $V(y)$ evolves as the system moves along its trajectory.

Theorem 2.1.7.1. *Suppose $f : \mathbb{R} \times \mathcal{C} \to \mathbb{R}^n$, that the function f maps into a bounded subset of $\mathbb{R}^n$, let u, v, w are continuous, non-decreasing functions satisfying $u(0) = v(0) = w(0) = 0$, and $u(x), v(x), w(x)$ are positives for $x > 0$ [18]. Assume that there is a continuous function $V(y) : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}$ such that*

$$u(|y|) \le V(t, y) \le v(|y|), \qquad t \in \mathbb{R}, \qquad y \in \mathbb{R}^n. \tag{2.5}$$

The stability condition are :

1. *The solution $y = 0$ is **uniformly stable** if*

$$\dot{V}(t, y(t)) \le -w(|y(t)|), \quad \textit{whenever} \quad V(t+s, y(t+s)) \le V(t, y(t)), \quad s \in [-\tau, 0].$$

2. *The solution $y = 0$ is* ***uniformly asymptotically stable*** *if there exists a continuous non-decreasing function $p(x) > 0$ for $x > 0$ such that*

$$\dot{V}(t, y(t)) \leq -w(|y(t)|),$$

whenever

$$V(t+s, y(t+s)) < p(V(t, y(t))), \quad s \in [-\tau, 0].$$

3. *The solution $y = 0$ is* ***globally asymptotically stable*** *if the second condition is satisfied, and $u(x) \to \infty$ as $x \to \infty$.*

2.2 Fractional Delay Differential Equations (FDDEs)

2.2.1 Introduction to FDDEs

Fractional delay differential equation (FDDEs) are a natural extension of conventional time-delay equation, using non-integer (fractional) derivatives to describe the dynamics. This combination of **the long memory** provided by fractional derivatives and **the delayed effect** represented by time delays provides a rich and realistic model for many physical and biological systems [17]. Prominent applications that benefit from these models include disease propagation, control network analysis, and delayed-response financial systems.

2.2.2 Fractional Linear Time-Delay (LTD) System

A fractional LTD system is a dynamical system where the evolution of the system's state depends on its current state and its past states, and this dependence is linear. Additionally, the system incorporates fractional-order dynamics, which introduce memory effects into the system's behavior. Mathematically, a fractional LTD system can be described by a (FDDE) of the form:

$$D^{\alpha} y(t) = Ay(t) + By(t-\tau) + Cu(t), \tag{2.6}$$

where

- D^{α} is the fractional derivative operator of order α (where $n-1 < \alpha < n, \ n \in \mathbb{N}$),
- $y(t) \in \mathbb{R}^n$ is the state vector at time t, and $y(t-\tau) \in \mathbb{R}^n$ is the delayed state vector at time $t-\tau$, with $u(t) \in \mathbb{R}^m$ is the input vector, where $\tau > 0$ represents the time delay,
- $A \in \mathbb{R}^{n \times n}$ is the system matrix, $B \in \mathbb{R}^{n \times n}$ is the delay matrix, and $C \in \mathbb{R}^{n \times m}$ is the input matrix.

The system (2.6) can also be represented in the Laplace domain by a transfer function, so

$$s^{\alpha} Y(s) - \sum_{k=0}^{n-1} s^{\alpha-k-1} y^{(m)}(0) = AY(s) + Be^{-s\tau} Y(s) + CU(s), \tag{2.7}$$

where

- $L\{D^{\alpha}y(t);s\} = s^{\alpha}Y(s) - \sum_{k=0}^{n-1} s^{\alpha-k-1}y^{(m)}(0)$,
- $L\{y(t);s\} = Y(s)$, $L\{u(t);s\} = U(s)$,
- $L\{y(t-\tau);s\} = e^{-s\tau}Y(s)$.

The transfer function is:

$$G(s) = \frac{Y(s)}{U(s)} = C\left(s^{\alpha}I - A - Be^{-s\tau}\right)^{-1}. \tag{2.8}$$

2.2.2.1 Stability Analysis with Matrix Coefficients

To analyze the stability of the system (2.6), we examine the poles of the transfer function $G(s)$ (2.8). Stability is determined by the poles of $G(s)$. The poles are the roots of the denominator of $G(s)$, which correspond to the values of s that satisfy the characteristic equation:

$$det\left(s^{\alpha}I - A - Be^{-s\tau}\right) = 0.$$

If A and B commute, we use a combination between Theorem 1.2.1.1 and Theorem 2.1.6.1 to establish the following stability criterion.

Theorem 2.2.2.1. *(Combination Theorem) [17]*
The system (2.2) is ***asymptotically stable*** *if and only if all characteristic roots s satisfy:*

$$|arg(s_{i,k})| > \frac{\alpha\pi}{2}, \quad i = \overline{1,n}, \quad k \in \mathbb{Z}.$$

If any root violates this condition, the system exhibits ***oscillatory*** *or* ***unstable*** *behavior.*

2.2.3 Nonlinear Fractional Delay Differential Equation

In general, nonlinear fractional delay differential equation can be expressed as:

$$\begin{cases} D^{\alpha}y(t) = Af\big(y(t-\tau)\big) - By(t), \\ y(t) = y_0(t), \quad -\tau \le t \le 0, \end{cases} \tag{2.9}$$

where

- f is nonlinear function with class $\mathcal{C}^1$, with $0 < \alpha \le 1$,
- $A, B \in \mathbb{R}^{n\times n}$ and $y \in \mathbb{R}^n$,
- $\tau > 0$ represents the time delay.

2.2.4 Stability Analysis of Nonlinear FDDE

2.2.4.1 Stability by Linearization

Consider the following nonlinear fractional delay differential equation (2.9), where y^* its equilibrium point, satisfies [7]

$$Af(y^*) - By^* = 0. \tag{2.10}$$

To analyze stability, let small perturbations $\xi = y - y^*$ with $\xi \in \mathbb{R}^n$ is introduced around equilibrium point and $y_\tau = y(t-\tau)$, $\xi_\tau = \xi(t-\tau)$. Using Taylor series expansion, the function is approximated around the equilibrium, this means

$$\begin{aligned} D^\alpha \xi =& Af(y_\tau) - By, \\ =& Af(y^*) + A\xi_\tau f^{'}(y^*) + h(\xi) - B\xi - By^*, \end{aligned}$$

where h is continuous function such that $h(\xi) = 0$ and the function f satisfied (2.10) , so

$$D^\alpha \xi = A\xi_\tau f^{'}(y^*) - B\xi. \tag{2.11}$$

The equation (2.11) can also be represented in the Laplace domain by a transfer function $G(s)$, where the poles are the roots of the denominator of $G(s)$, its satisfied

$$Q(s) = s^\alpha I - Af^{'}(y^*)e^{-s\tau} + B = 0. \tag{2.12}$$

When matrices A and B commute, Theorem 2.2.2.1 provides an exact analytical stability criterion. In non-commutative cases, the analysis typically requires numerical approaches.

The problem when if there are one or more roots on the imaginary axis (i.e. $Re(s) = 0$). This indicates a transition state and the system may be in marginal stability or about to lose stability. Here the value of τ directly affects the stability analysis. It can determine whether the system will remain stable or become unstable and oscillatory, and that with comparison to the values of τ_c (**the critical delay**).

Remark 2.2.4.1.1. *(Critical Delay τ_c)*
It is the smallest delay at which $Re(s) = 0$ and stability changes, often leading to oscillations.

To find the critical delay, let $s = iv$ (**purely imaginary**) and the characteristic equation (2.12) becomes [7]

$$(iv)^\alpha I + B = Af^{'}(y^*)\, e^{-iv\tau_c}. \tag{2.13}$$

Using MATLAB code to find τ_c (**Annex A**).

Finding τ_c opens the door to studying **bifurcations** that may appear due to stability changes, especially when $\tau_c = \tau$, where the system undergoes a transition in its behavior.

2.3 Bifurcation Analysis

2.3.1 Introduction to Bifurcation

In FDDEs, the interaction between memory and delay can trigger dynamic transitions. One key phenomenon is **Hopf bifurcation**, where increasing the delay beyond a critical value shifts the system from steady state to sustained oscillations. This study examines the onset of such behavior and the nature of the resulting periodic solutions.

2.3.2 Limit Cycle in FDDEs

Definition 2.3.2.1. *A limit cycle in an FDDE is defined as a closed trajectory in the phase space, representing a periodic solution of the system. This cycle corresponds to sustained oscillations, emerging due to the combined effects of fractional dynamics and time delay [8].*

2.3.2.1 Classification of Limit Cycles

- **Stable Limit Cycle:** All trajectories in the vicinity of the cycle converge to it as $t \to \infty$.
- **Unstable Limit Cycle:** All trajectories in the vicinity of the cycle diverge from it as $t \to \infty$.
- **Half-stable Limit Cycle:** Some trajectories converge toward the cycle, while others diverge from it as $t \to \infty$.

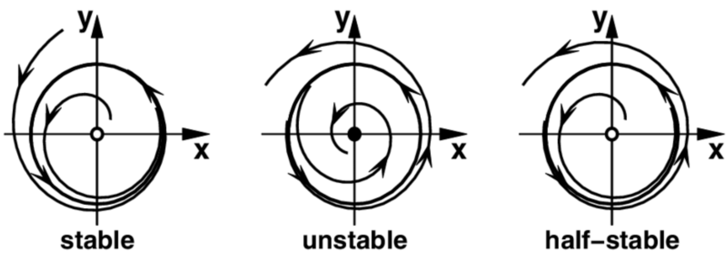


Figure 2.1: Classification of Limit Cycles.

2.3.2.2 Existence of Limit Cycles

In classical delay differential equations (DDEs), the Poincaré-Bendixson theorem is often used to prove the existence of limit cycles [8]. However, in fractional-order systems, due to memory effects, this theorem does not apply directly. Some extensions of the theorem to fractional-order systems exist under specific conditions. But there are some numerical methods arewidely used, of which (**Bifurcation diagrams** to study how solutions change with system parameters).

2.3.3 Bifurcation Diagram

A bifurcation diagram in fractional delay differential equations (FDDEs) is a graphical representation that shows how equilibrium points or periodic solutions of the system change as a parameter (such as the delay τ or the fractional order α) changes. In FDDEs, bifurcations can be more complex due to memory effects and non-locality, leading to non-traditional transitions between stability and oscillations when crossing a critical parameter value.

- **Horizontal Axis:** Represents the control parameter (e.g., time delay τ, fractional order α, or another bifurcation parameter.
- **Vertical Axis:** Represents the system's behavior, such as oscillation amplitudes or steady-state solutions.
- **Stable and Unstable Branches:** Solid lines usually represent stable solutions, while dashed lines indicate unstable solutions.
- **Bifurcation Point:** The critical value τ_c where a qualitative change in system behavior occurs (e.g., transition from stability to oscillations).

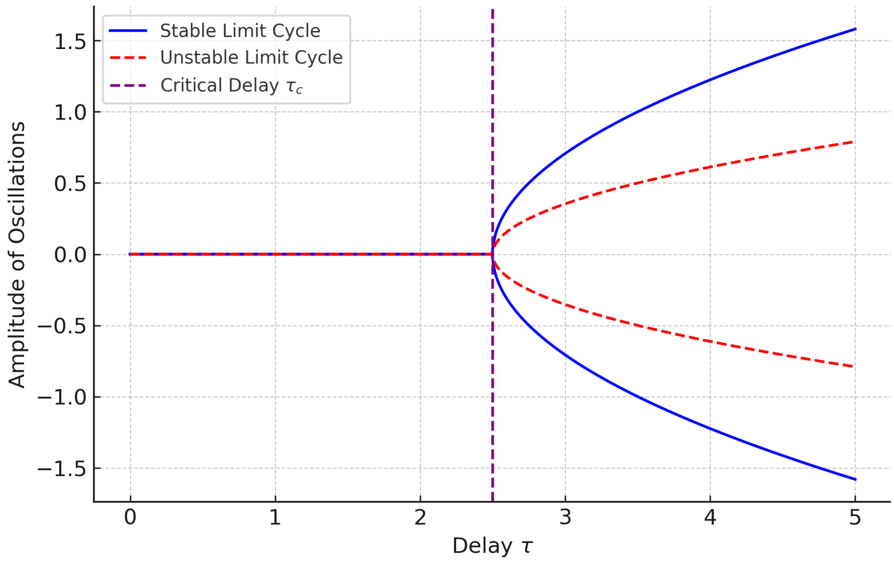


Figure 2.2: Bifurcation Diagram of FDDEs (Hopf Bifurcation).

2.3.4 Hopf Bifurcation in Fractional Delay Differential Equations

A common method to prove the existence of limit cycles in FDDEs is Hopf bifurcation, which occurs when complex conjugate eigenvalues cross the imaginary axis. This happens when the characteristic equation has a root of the form (2.12) a purely imaginary root $\pm iv$ exists and moves to the right half-plane as a parameter (such as τ) changes, this indicates the emergence of a limit cycle. This is when comparing the critical delay τ_c with the delay τ, according to the following rules [7]:

- **if** $\tau < \tau_c$**:** All the solutions converge to the equilibrium point, indicating that the system is **stable**.
- **if** $\tau = \tau_c$**:** The solution starts showing periodic oscillations, marking the onset of **Hopf bifurcation**.
- **if** $\tau > \tau_c$**:** The limit cycle can become **unstable**, leading to complex behaviors, such as chaos or random oscillations can follow.

Chapter 3

Chaos Control for Fractional-Order Systems

3.1 Introduction to Chaos Control for FOSs

Nonlinear fractional systems may exhibit chaotic behavior due to their extreme sensitivity to initial conditions, making control a necessity in practical applications. Controlling chaos in such systems is essential in engineering, medicine, and economics fields, as it helps stabilize dynamics and prevent undesired deviations [11]. Among the most widely used techniques are time-based controllers like **PID** and its fractional derivatives, which are effective in suppressing oscillations and achieving system stability [25].

3.2 Controllability and Observability of Nonlinear Fractional-Order Chaotic Systems

3.2.1 Controllability Analysis

Controllability refers to the ability to drive the system from any initial state $y(0)$ to a desired final state $y(T)$ within a finite time using appropriate inputs $u(t)$. This is typically assessed by linearizing the system around an equilibrium point and analyzing the rank of the controllability matrix. In the local sense, Kalman-based methods are used to determine controllability in fractional systems.

Consider a general nonlinear fractional-order system represented by the following equation

$$\begin{cases} D^{\alpha}y(t) = f(y) + g(y)u(t), & t > 0, \\ y(0) = y_0, \end{cases} \tag{3.1}$$

where

- D^{α} is the fractional Caputo derivative with order $0 < \alpha < 1$,
- $y(t) \in \mathbb{R}^n$ is the state vector, and $u(t) \in \mathbb{R}^m$ is the control input,

- $f(y)$ defines the chaotic system dynamics,
- $g(y)$ represents the control influence on the system.

Remark 3.2.1. *The controllability of a **system cannot be studied without a control signal**, as there would be no means to influence the system's behavior.*

For nonlinear fractional-order chaotic systems, controllability is analyzed using [15]:

3.2.1.1 Local Controllability via Linearization

We linearize the system (3.1) around an equilibrium point y^*, leading to the form

$$D^\alpha \delta y(t) = A\delta y(t) + B\delta u(t),$$

where

- $\delta y(t) := y(t) - y^*$: represents a small perturbation,
- $A = \left[\frac{\partial f}{\partial y}\right]_{y^*}$: Jacobian of $f(y)$ evaluated at y^*,
- $B = g(y^*)$: input matrix evaluated at the equilibrium point.

We then apply the classical Kalman rank condition, which states that the linearized system is locally **controllable** if the controllability matrix $\mathcal{C}$ has full rank

$$rank\begin{bmatrix} B & AB & A^2B & \dots & A^{n-1}B \end{bmatrix} = rank(\mathcal{C}) = n,$$

where n is the system dimension.

3.2.2 Observability Analysis

Observability determines whether the internal state of the system can be reconstructed from output measurements. This is evaluated by analyzing the observability matrix of the linearized system. If the matrix has full rank, the system is locally observable.

3.2.2.1 Local Observability via Linearization

Consider the system (3.1), with single output measurement $y_1(t)$, we linearize the system it around an equilibrium point y^*, leading to the form

$$\begin{cases} D^\alpha \delta y(t) = A\delta y(t) + B\delta u(t), \\ z(t) = y_1(t), \end{cases}$$

where

- $A = \left[\frac{\partial f}{\partial y}\right]_{y^*}$,

- $C = [1 \;\; 0 \;\; 0 \;\; \ldots \;\; 0]$: assume y_1 is measured.

We then apply the rank condition, which states that the linearized system is locally **observable** if the observability matrix $\mathcal{O}$ has full rank [9]

$$rank \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{bmatrix} = rank(\mathcal{O}) = n,$$

where n is the system dimension.

3.3 Time-Delay Controllers in FOSs

In FOS, we have two distinct control approaches: the **delayed feedback controller** ($u(t) = K(y(t-\tau) - y(t))$) and the fractional-order PID ($PI^\lambda D^\mu$). The delayed feedback controller primarily addresses time-delay compensation through a simple difference term, while $PI^\lambda D^\mu$ incorporates fractional calculus operators (I^λ and D^μ) to directly manage the system's inherent fractional-order dynamics. When dealing with FOS, $PI^\lambda D^\mu$ proves superior as it's specifically designed to handle the complex memory-dependent and non-local characteristics of such systems through its fractional-order terms [25].

3.3.1 Classical PID Controller

The PID controller is one of the most widely used control tools. It operates using three components: proportional (P), integral (I), and derivative (D), which together correct the error between the reference and the actual output to reduce it over time [26]. Its transfer function in Laplace domain can be expressed as

$$G_c(s) = \frac{U(s)}{E(s)} = K_p + \frac{K_i}{s} + K_d s.$$

Output $u(t) = L^{-1}\{U(s); t\}$, in the time domain as

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt},$$

where

- $G_c(s)$ is the transfer function of the PID controller in Laplace domain,
- $U(s)$ is the controller output, $e(s)$ is the error signal,
- K_d Derivative gain, which anticipates future error by evaluating the rate of change and thus improves stability,

- K_p Proportional gain, which reacy to the current error,
- K_i Integral gain, which eliminates steady-state error.

3.3.2 $PI^{\lambda}D^{\mu}$ Controller with Time Delay

The fractional-order PID controller with time delay, often referred to as the delayed $PI^{\lambda}D^{\mu}$ controller, is an advanced extension of the classical PID controller. In this controller, both the integration and differentiation are of fractional (non-integer) order, offering increased flexibility in tuning and dynamic response. The controller also incorporates a time delay element, which reflects realistic system behavior in processes where control action is not instantaneous.
The transfer function of the fractional-order PID controller with time delay, in the Laplace domain [9] [22].

$$G_c(s) = \frac{U(s)}{E(s)} = \left(K_p + \frac{K_i}{s^{\lambda}} + K_d s^{\mu}\right) e^{-s\tau}.$$

Output $u(t) = L^{-1}\{U(s); t\}$, in the time domain as

$$u(t) = K_p e(t-\tau) + K_i D^{-\lambda} e(t-\tau) + K_d D^{\mu} e(t-\tau).$$

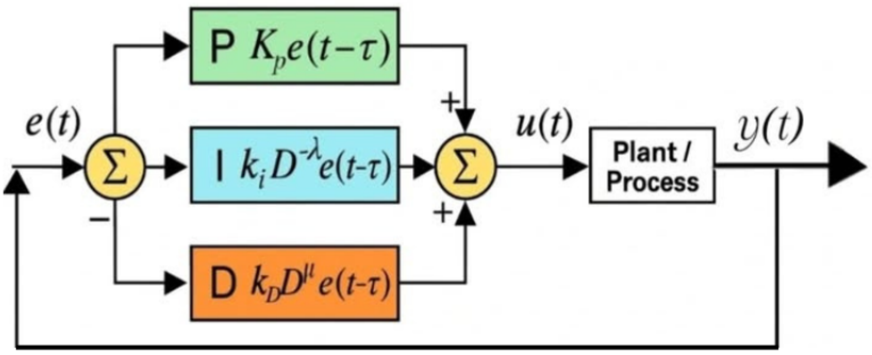


Figure 3.1: $PI^{\lambda}D^{\mu}$ Block Diagram.

Remark 3.3.2.1. *In control theory, $r(t)$ is called the reference signal or setpoint. It's the desired value or trajectory you want the system's output to follow, with $e(t)$ is difference between setpoint $r(t)$ and process variable $y(t)$*

$$e(t) = r(t) - y(t).$$

3.4 Controller Position

Chaotic systems are highly sensitive to initial conditions and exhibit complex, unpredictable behavior. The position of the controller in such systems is critical for effective stabilization or chaos suppression. Below is a structured breakdown of the role, placement, and design of controllers in chaotic systems.

3.4.1 Key Factors that Determine Controller Position

In many control scenarios, the control input is applied to a single state variable, the one that

- Directly influences system behavior,
- Drives the chaos,
- An output variable that we can measure and control directly, such as the angle of rotation or voltage.

3.5 Stability Analysis of Control Systems

We consider a fractional-order delay system under the influence of a delayed $PI^{\lambda}D^{\mu}$ controller

$$\begin{aligned} D^{\alpha}y(t) =& f(y) + g(y)u(t), \\ =& f(y) + g(y)\left[K_p e(t-\tau) + K_i D^{-\lambda} e(t-\tau) + K_d D^{\mu} e(t-\tau)\right], \end{aligned}$$

where D^{α} denotes the Caputo fractional derivative of order $\alpha \in (0,1)$.

3.5.1 Stability Using a Lyapunov Function

Theorem 3.5.1. *Let $y^* \in \mathbb{R}^n$ be an equilibrium point for the fractional-order system (3.1). Suppose there exists a Lyapunov function $V(t, y(t))$ such that*

$$f(y^*) + g(y^*)u^* = 0. \tag{3.2}$$

Define $x(t) = y(t) - y^*$ $(x \approx 0)$, and rewrite the system as

$$D^{\alpha}x(t) = f(x + y^*) + g(x + y^*)u^*. \tag{3.3}$$

Let $V : \mathbb{R}^n \to \mathbb{R}$ be a continuously differentiable [1] satisfying

- *$V(x) > 0$ for all $x \neq 0$,*
- *$V(0) = 0$.*

Then, by estimating the fractional derivative of V the following inequality holds

$$D^{\alpha}V(x(t)) \leq \boldsymbol{\nabla} V(x)^T . D^{\alpha}x(t) < 0 \quad \text{whenever} \quad V(x(s)) \leq V(x(t)), \quad s \in [-\tau, 0].$$

Then the system (3.1) is asymptotically stable in the sense of Lyapunov.

Remark 3.5.1.1. *Estimating the fractional derivative of V, helps avoid computing the fractional derivative of a composite function, which can be challenging. The goal of this estimation is to prove that the Lyapunov function decreases over time.*

3.6 Examples of Control Chaos for FOSs

3.6.1 Fractional-Order Rössler's System

We define fractional Rössler system (1.31) with control $u(t)$, using Caputo fractional derivatives of order $\alpha \in (0, 1]$

$$\begin{cases} D_t^{\alpha_1} x(t) = & -(y(t) + z(t)) + u(t), \\ D_t^{\alpha_2} y(t) = & x(t) + ay(t), \\ D_t^{\alpha_3} z(t) = & b + z(t)(x(t) - c). \end{cases} \tag{3.4}$$

We want to control the chaotic behavior of the fractional-order Rössler system by designing a suitable control law.

Here, we propose a **time-delayed fractional-order PID controller** to stabilize the system around its equilibrium point. The control input is applied to the first state equation of the system, targeting the variable $x(t)$, which plays a key role in the overall dynamics. The objective is to force the state $x(t)$ to follow a desired reference $r(t)$, thus eliminating the chaotic oscillations and ensuring convergence toward a stable behavior.

Defined the controller as:

$$u(t) = K_p e(t - \tau) + K_i D^{-\lambda} e(t - \tau) + K_d D^{\mu} e(t - \tau).$$

Typical values are the same in the system (1.31), with $K_p = K_i = K_d = 1$ and $\lambda = \mu = 0, 9$ with $\tau = 1$, $r(t) = 5.7$ choosing 5.7 because in many chaotic systems (like Rössler or Lorenz), values in that range (5–10) are common in the chaotic regime.

Then numerical simulations of the Rössler system is depicted in Figure 3.2

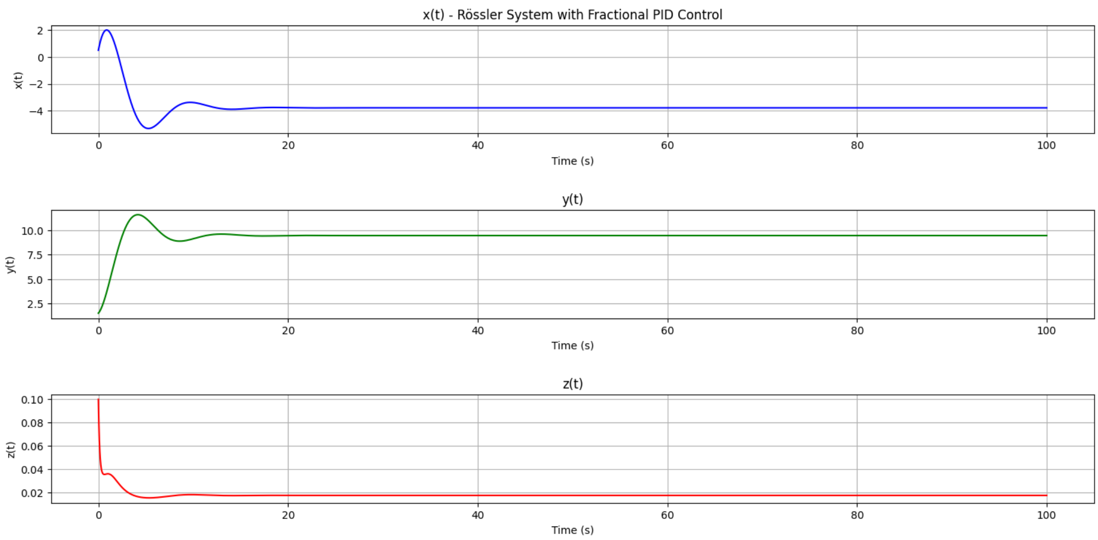


Figure 3.2: Plot of Fractional-Order Rössler System with Time Delay $PI^{\lambda}D^{\mu}$ Controller.

3.6.2 Fractional-Order Lotka-Volterra System

We define fractional Lotka-Volterra system (1.32) with control $u(t)$, using Caputo fractional derivatives of order $\alpha \in (0, 1]$

$$\begin{cases} D_t^{\alpha_1} x(t) = & ax(t) - bx(t)y(t) + ex^2(t) - sz(t)x^2(t) + u(t), \\ D_t^{\alpha_2} y(t) = & -cy(t) + dx(t)y(t), \\ D_t^{\alpha_3} z(t) = & -pz(t) + sz(t)x^2(t). \end{cases} \tag{3.5}$$

we propose a **time-delayed fractional-order PID controller** to stabilize the system around its equilibrium point.

$$u(t) = K_p e(t-\tau) + K_i D^{-\lambda} e(t-\tau) + K_d D^{\mu} e(t-\tau).$$

Typical values are the same in the system (1.32), with $K_p = K_d = 1$ and $= K_i = 0.5$, $\lambda = \mu = 0,8$ with $\tau = 1$, $r(t) = 1.5$ is arbitrary reference.

Then numerical simulations of the Lotka-Volterra system is depicted in Figure 3.3.

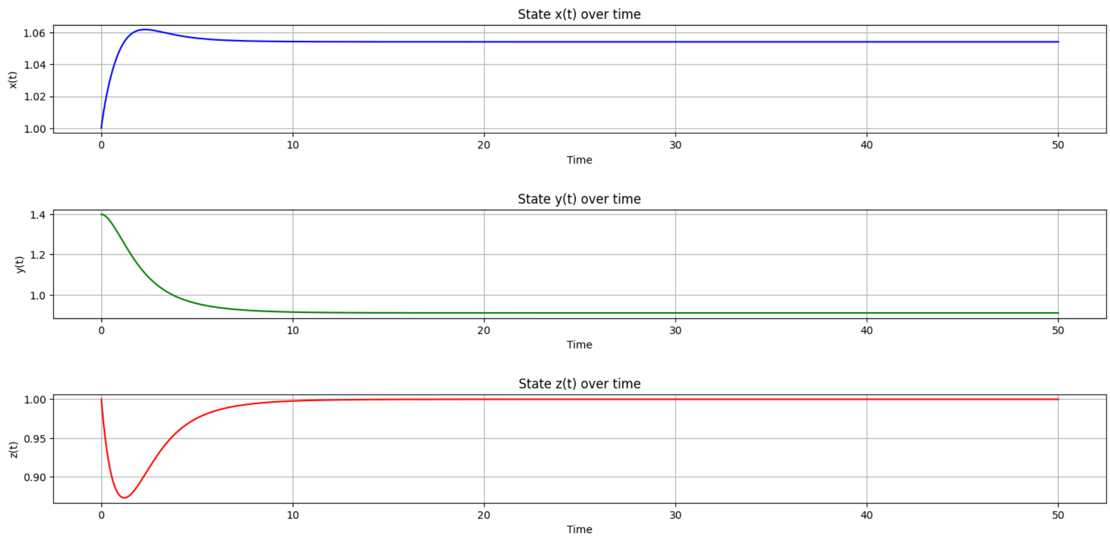


Figure 3.3: Plot of Fractional-Order Lotka-Volterra System with Time Delay $PI^{\lambda}D^{\mu}$ Controller.

Chapter 4

Application

4.1 Chaos Control in Fractional-Order Drone Systems Using Time-Delayed $PI^{\lambda}D^{\mu}$ Controller

Drones, often referred to as unmanned aerial vehicles (UAVs), are autonomous or remotely piloted flight systems that have become essential tools in a variety of fields, including surveillance, agriculture, and scientific research. Due to their complex and nonlinear dynamics, these systems can exhibit **chaotic behavior** under certain conditions, such as sudden wind disturbances or external environmental changes. This chaotic behavior can lead to loss of control, necessitating the development of effective mechanisms to control the movement and ensure stability during flight.

Figure 4.1: Drone

In this work, a fractional $PI^{\lambda}D^{\mu}$ controller integrated with a time delay is adopted to stabilize the chaotic behavior that may appear in the drone model. This controller was specifically designed to address the nonlinear nature of the system, as well as the memory properties characteristic of fractional derivative systems. This design enables a more flexible and accurate response to sudden disturbances, enhancing the drone's stability under non-ideal operating conditions. The study will be conducted through numerical simulations using **MATLAB**.

This research bridges the fields of chaos theory, fractional calculus, and control systems, providing valuable insights into stabilizing chaotic nonlinear systems.

4.2 Full Characteristics of a Drone Flying in the Sky

4.2.1 Degrees of Freedom

A rigid body in 3D space has 6 degrees of freedom:

- **3 Translational**: motion along the x, y, z axes,
- **3 Rotational**: rotation about those axes (roll ϕ, pitch θ, yaw ψ).

For a drone it can

- **Move**: forward/backward, left/right, up/down,
- **Rotate**: roll (left/right tilt), pitch (front/back tilt), yaw (rotate about vertical axis) [28].

4.2.2 Movements of a Drone

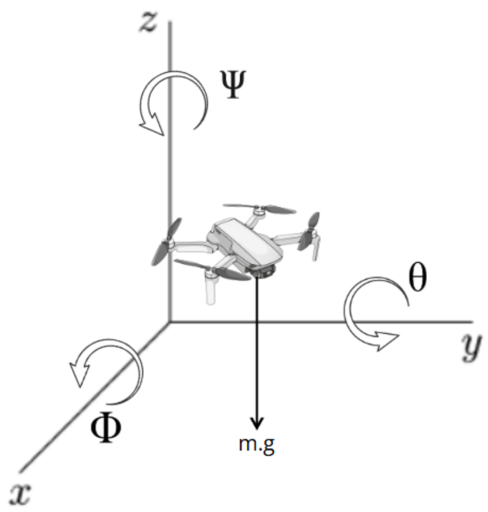


Figure 4.2: Drone in 3D Space.

4.2.2.1 Position Dynamics

The movement of the drone in three-dimensional space [6].

- $x(t), y(t), z(t)$: Represents the **position** of the drone in 3D space.
- $\dot{x}(t), \dot{y}(t), \dot{z}(t)$: Represents the linear **speed** of the drone on the three axes.

Then the Equations are :

$$\begin{cases} D_t^{\alpha_1} x(t) = & \dot{x}(t), \\ D_t^{\alpha_2} y(t) = & \dot{y}(t), \\ D_t^{\alpha_3} z(t) = & \dot{z}(t). \end{cases} \tag{4.1}$$

4.2.2.2 Linear Accelerations

Represents how the thrust generated by propellers transforms into acceleration in 3D space.

- T: Thrust generated by the four propellers,
- m: Mass of the drone,
- d_1, d_2, d_3: Represent disturbances or external influences (wind, air resistance, etc.),
- g: The gravity.

Acceleration in the x-axis:

- Represents the force acting on the forward horizontal axis of the drone,
- Roll ϕ, pitch θ, yaw ψ are controls the direction of thrust on this axis.

Acceleration in the y-axis:

- This axis represents the lateral movement of the aircraft (right/left),
- It also depends on the three angles.

Acceleration in the z-axis:

- This represents the vertical movement (up and down),
- Thrust fights gravity, with $T > m.g$.

Then the Equations are :

$$\begin{cases} D_t^{\alpha_4}\dot{x}(t) = & \dfrac{T}{m}(\cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi) + d_1, \\ D_t^{\alpha_5}\dot{y}(t) = & \dfrac{T}{m}(\cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi) + d_2, \\ D_t^{\alpha_6}\dot{z}(t) = & \dfrac{T}{m}(\cos\phi\cos\theta) - g + d_3. \end{cases} \tag{4.2}$$

4.2.3 Rotation of a Drone

4.2.3.1 Angular Velocity

Represents how the drone's three-point steering angle changes over time as a result of the angular velocities generated by the propellers.

- p, q, r: Angular velocities $\phi(t)$, $\theta(t)$ and $\psi(t)$ about the axis x, y and z-axis.

Then the Equations are :

$$\begin{cases} D_t^{\alpha_7}\phi(t) = & p + \tan\theta\,(q\sin\phi + r\cos\phi)\,, \\ D_t^{\alpha_8}\theta(t) = & q\cos\phi - r\sin\phi, \\ D_t^{\alpha_9}\psi(t) = & \dfrac{q\sin\phi + r\cos\phi}{\cos\phi}. \end{cases} \tag{4.3}$$

4.2.3.2 Angular Accelerations

These equations represent how the angular velocities of the drone change about the three axes x, y, and z, due to torques, physical properties such as moment of inertia, and nonlinear effects.

- I_x, I_y, I_z Moments of inertia along the three axes,
- τ_x, τ_y, τ_z Torques applied around the corresponding axes,
- $\epsilon_1, \epsilon_2, \epsilon_3$ Disturbance or nonlinear damping effects.

$$\begin{cases} D_t^{\alpha_{10}} p(t) = & \frac{1}{I_x}\left[\tau_\phi + (I_y - I_z)qr\right] + \epsilon_1 \cos(\phi.\theta), \\ D_t^{\alpha_{11}} q(t) = & \frac{1}{I_y}\left[\tau_\theta + (I_z - I_x)pr\right] + \epsilon_2 \sin(p.t), \\ D_t^{\alpha_{12}} r(t) = & \frac{1}{I_z}\left[\tau_\psi + (I_x - I_y)pq\right] + \epsilon_3 \sin(qr). \end{cases} \tag{4.4}$$

4.2.4 Nonlinear Fractional-Order System of a Drone

$$D^\alpha X(t) = f(X(t)) + d(t). \tag{4.5}$$

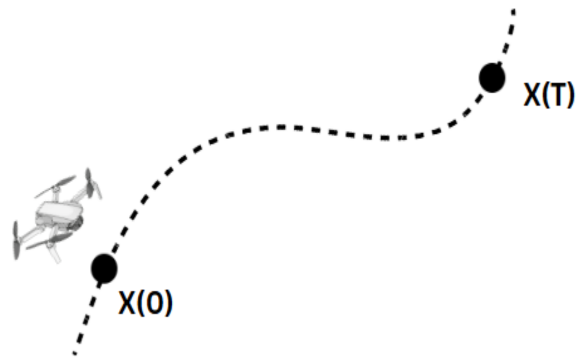
- $X(t) = \left[x, y, z, \dot{x}, \dot{y}, \dot{z}, \phi, \theta, \psi, p, q, r\right]^T$,
- $f(.)$: It is a nonlinear function that represents dynamics.

Thus, the drone system is [6][27]

$$\begin{cases} D_t^{\alpha_1} x(t) = & \dot{x}(t), \\ D_t^{\alpha_2} y(t) = & \dot{y}(t), \\ D_t^{\alpha_3} z(t) = & \dot{z}(t), \\ D_t^{\alpha_4} \dot{x}(t) = & \frac{T}{m}(\cos\phi \sin\theta \cos\psi + \sin\phi \sin\psi) + d_1, \\ D_t^{\alpha_5} \dot{y}(t) = & \frac{T}{m}(\cos\phi \sin\theta \sin\psi - \sin\phi \cos\psi) + d_2, \\ D_t^{\alpha_6} \dot{z}(t) = & \frac{T}{m}(\cos\phi \cos\theta) - g + d_3, \\ D_t^{\alpha_7} \phi(t) = & p + \tan\theta \left(q \sin\phi + r \cos\phi\right), \\ D_t^{\alpha_8} \theta(t) = & q \cos\phi - r \sin\phi, \\ D_t^{\alpha_9} \psi(t) = & \frac{q \sin\phi + r \cos\phi}{\cos\phi}, \\ D_t^{\alpha_{10}} p(t) = & \frac{1}{I_x}\left[\tau_\phi + (I_y - I_z)qr\right] + \epsilon_1 \cos(\phi.\theta), \\ D_t^{\alpha_{11}} q(t) = & \frac{1}{I_y}\left[\tau_\theta + (I_z - I_x)pr\right] + \epsilon_2 \sin(p.t), \\ D_t^{\alpha_{12}} r(t) = & \frac{1}{I_z}\left[\tau_\psi + (I_x - I_y)pq\right] + \epsilon_3 \sin(qr). \end{cases} \tag{4.6}$$

4.3 Effect of Wind on Drone Movement

In this study, we aim to control the drone's trajectory to ensure it reaches $X(T)$, despite external disturbances such as wind. The controller is designed to minimize the error between the actual and desired paths, maintaining system stability during flight.



The simulation was conducted using **MATLAB/Simulink** to model the dynamics of a quadrotor drone. The physical model includes six degrees of freedom: three translational positions (x, y, z) and three rotational angles (ϕ, θ, ψ). In this setup, rotational torques were not introduced as separate input variables. Instead, the simulation relied solely on the vertical thrust forces generated by the four rotors. The resulting torques are computed automatically based on each rotor's position relative to the center.

4.3.1 Phase 1: Natural Stability under Normal Wind Conditions

In this initial phase, the drone operates under nominal atmospheric conditions. The objective is to evaluate the inherent stability of the system in the absence of strong perturbations and without activating any advanced control model.

Wind Profile (Normal Conditions)

- Wind speed: $v_{wind} = 5m/s$,
- Wind direction: Constant or slowly varying within 0 to $\frac{\pi}{4}$ radians,
- Turbulence intensity: Low,
- Disturbance is modeled as an additive low-frequency input to the system dynamics

$$d_{wind} = a\sin(\omega t + \phi), \qquad a < 1,$$

where a is small amplitude it means gentle wind and ω is low frequency it means slow variation.

4.3.1.1 Stability Observation via Simulation

The stability of the system in equation (4.6) is qualitatively evaluated based on its simulated response in MATLAB over the time interval $t \in [0, 10]$, under mild wind conditions.

1. **Equilibrium Point:** A representative hovering equilibrium is assumed as:

$$X^* = \Big[x_0, y_0, z_0, 0, 0, 0, 0, 0, 0, 0, 0, 0\Big]^T,$$

2. **Simulated Behavior:** No explicit numerical methods (e.g., eigenvalue analysis or Stability Condition as Theorem 1.2.1.1) are used. Instead, MATLAB simulations provide insight into the system's response. Under normal wind, the system remains close to equilibrium, indicating bounded and stable behavior.

The resulting trajectories are illustrated in Figure 4.3.

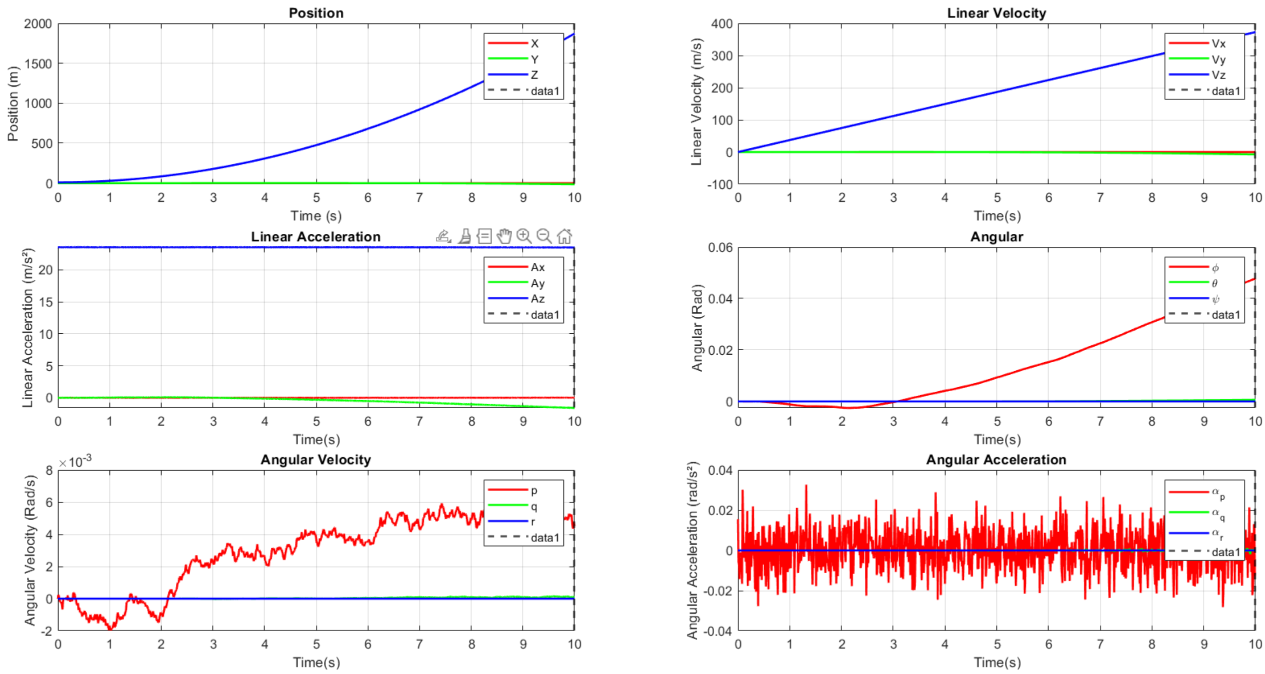
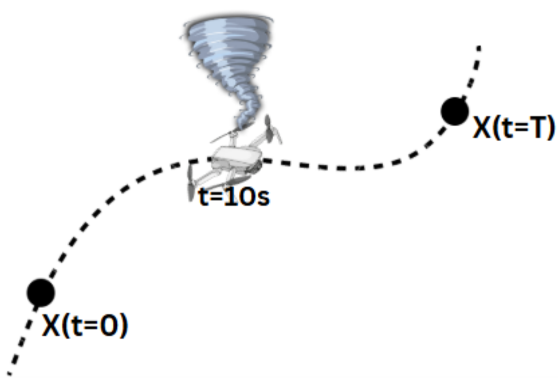


Figure 4.3: Plot of Effect of Wind on Drone Movement and Rotation from $X(t = 0)$ to $X(t \leq 10)$.

Figure 4.3 shows the effect of normal wind on the drone's movement and rotational motion from $t = 0$ to $t = 10$ seconds. The trajectories remain bounded and gradually converge toward the origin, indicating that the system maintains local stability.

4.3.2 Phase 2: Chaos Induced by Strong Wind at $t = 10$ seconds

At $t = 10$ seconds, the drone is suddenly subjected to a **strong wind** disturbance. This external perturbation significantly affects the drone's dynamics, leading to large nonlinear fluctuations in both its translational and rotational states.



The wind intensity is sufficiently high to destabilize the system, driving it from a previously stable regime into a **chaotic state**.

Strong Wind Characteristics:

- Wind speed: $v_{wind} = 10$ to $20m/s$,
- Wind direction: It changes irregularly within the range $\frac{\pi}{4}$ to π radians,
- Turbulence intensity: It is a measure of how much wind speed fluctuates over time,
- The wind disturbance is modeled as an additive high-frequency input injected into the system dynamics

$$d_{wind} = a\sin(\omega t) + noise(t), \qquad a > 1,$$

where the added **noise** introduces random, fast-changing disturbances and turbulence.

The numerical simulation of the system (4.6) under strong wind begins at $t = 10$, and the drone's response is illustrated in Figure 4.4.

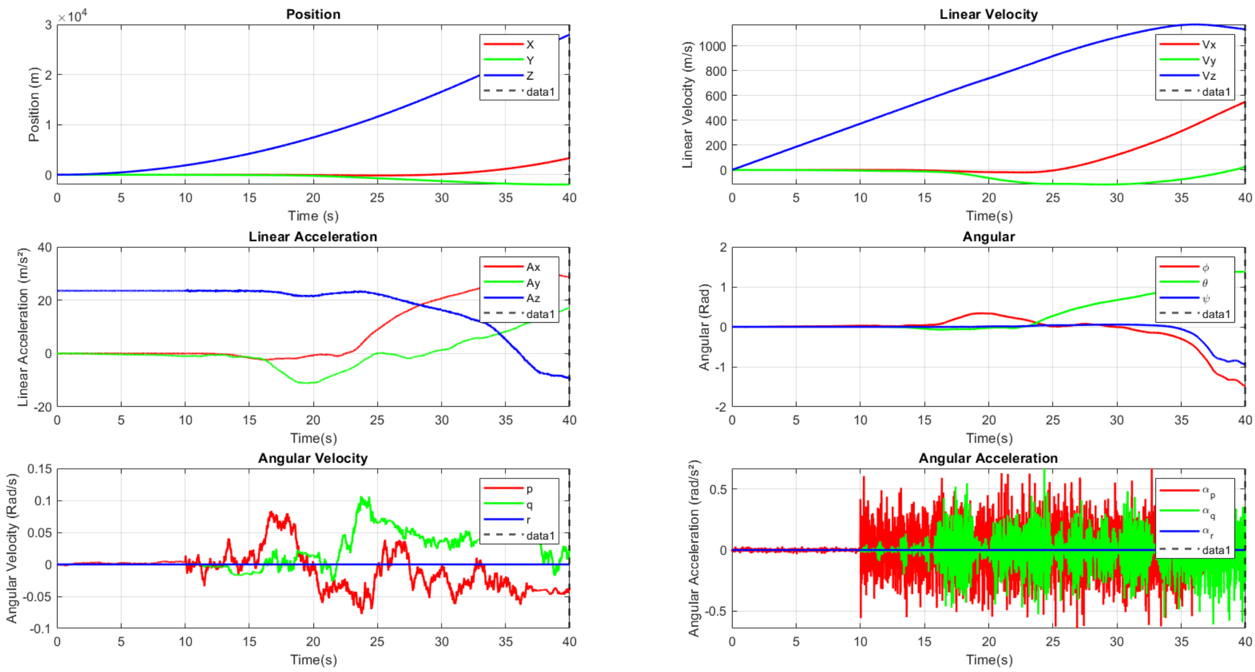


Figure 4.4: Plot of Effect of Strong Wind on Drone Movement and Rotation at $t = 10s$.

Figure 4.4 illustrates the destabilizing impact of strong wind applied at $= 10s$. The system, initially stable, begins to exhibit significant nonlinear oscillations and divergence from equilibrium, indicating the onset of chaotic dynamic.

To further highlight the presence of **chaotic behavior**, the 3D trajectory is shown in Figure 4.5. The intricate, non-repeating path exhibits a **fractal structure**, which is a hallmark of deterministic chaos.

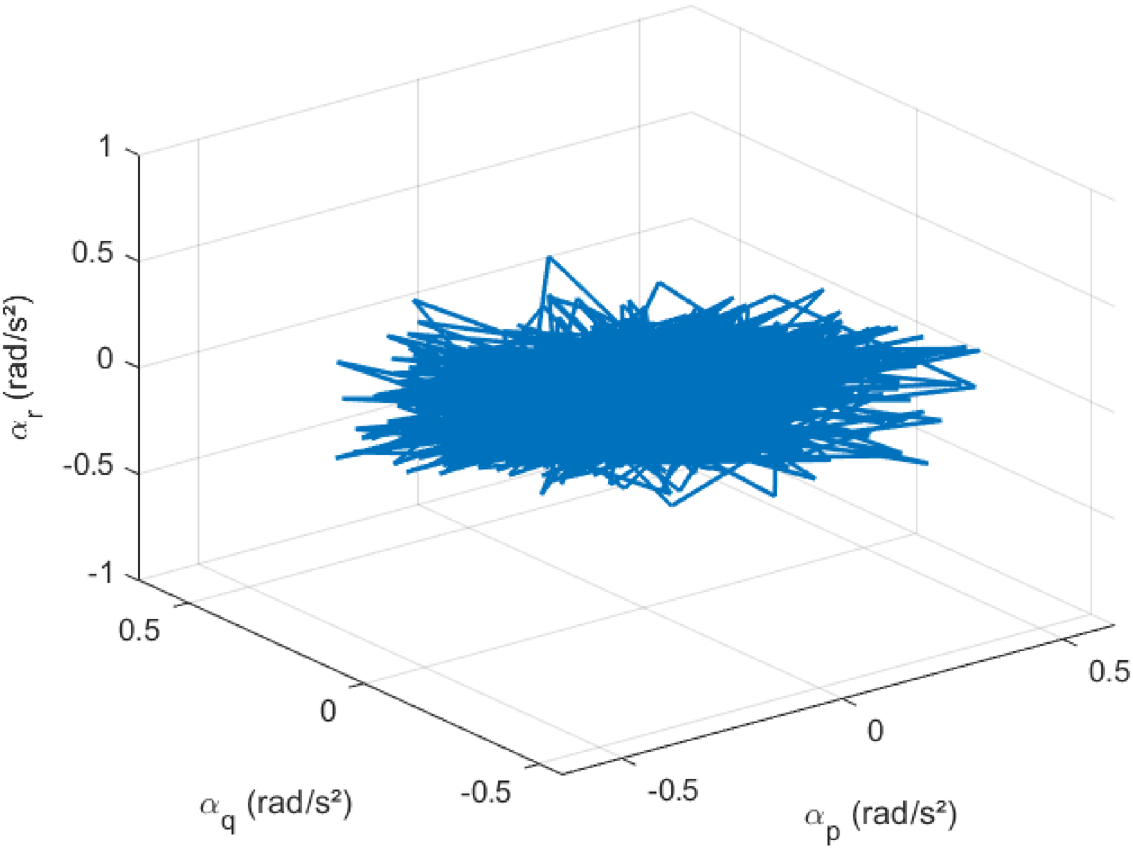
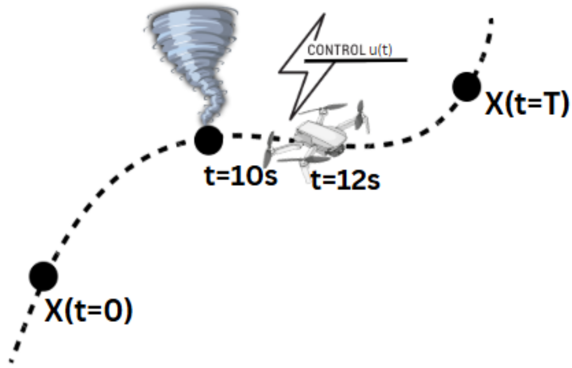


Figure 4.5: 3D Plot of Effect of Strong Wind on Drone Movement and Rotation at $t = 10s$.

4.3.3 Phase 3: Applying Time-Delayed $PI^{\lambda}D^{\mu}$ Controller

In this phase, the objective is to stabilize the drone once it enters a chaotic state caused by strong wind disturbances. To address this, a time-delayed fractional-order $PI^{\lambda}D^{\mu}$ controller is employed. The general nonlinear fractional-order drone dynamics with control input $u(t)$ is described by:



$$D^{\alpha}X(t) = f(X(t)) + Bu(t) + d(t). \tag{4.7}$$

The control input is given by:

$$u(t) = K_p e(t-\tau) + K_i D^{-\lambda} e(t-\tau) + K_d D^{\mu} e(t-\tau). \tag{4.8}$$

Prior to the controller's application, verifying system **controllability** is crucial.

- **Controllability Analysis**

  Linearizing the nonlinear system (4.9) around an equilibrium point X^*, leading to the form

  $$D^{\alpha}\delta X(t) = A\delta X(t) + B\delta u(t).$$

  The matrix B is designed to reflect realistic physical actuation, assuming that control inputs directly influence the drone's **linear accelerations**. Accordingly, the matrices A and B are defined as:

  $$A = \frac{\partial f(X)}{\partial X} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{x_{12}} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_{12}}{\partial x_1} & \cdots & \frac{\partial f_{12}}{\partial x_{12}} \end{bmatrix} \text{ and } B = \begin{bmatrix} 0_{5\times 1} \\ I_{1\times 1} \\ 0_{3\times 1} \\ 0_{3\times 1} \end{bmatrix},$$

  where $u(t) \in \mathbb{R}$.

  For matrix A, we skip the angles as variables and replace them with their value in that case, these matrices A and B must be numerical matrices, not functions of variables.

  Controllability is verified using the Kalman rank condition ($rank(\mathcal{C})$):

  $$rank\begin{bmatrix} B & AB & A^2B & \ldots & A^{11}B \end{bmatrix} = rank(\mathcal{C}),$$

  with $\mathcal{C} \in \mathbb{R}^{12\times 12}$.

  The rank was determined using numerical methods and was found $rank(\mathcal{C}) = 12$
  Thus the drone system is **controllable**.

- **Controller Position**

  According to Newton-Euler dynamics [27], acceleration is directly influenced by applied forces and torques. Thus, the optimal point of control insertion is directly at the acceleration level. Furthermore, the system is subjected to exogenous wind disturbances d_i, which can induce chaotic dynamics.

The complete controlled system is governed by:

$$D^{\alpha}X(t) = f(X(t)) + Bu(t) + d(t). \tag{4.9}$$

$$\begin{cases} D_t^{\alpha_1}x(t) = & \dot{x}(t), \\ D_t^{\alpha_2}y(t) = & \dot{y}(t), \\ D_t^{\alpha_3}z(t) = & \dot{z}(t), \\ D_t^{\alpha_4}\dot{x}(t) = & \dfrac{T}{m}(\cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi) + d_1, \\ D_t^{\alpha_5}\dot{y}(t) = & \dfrac{T}{m}(\cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi) + d_2, \\ D_t^{\alpha_6}\dot{z}(t) = & \dfrac{T}{m}(\cos\phi\cos\theta) - g + d_3 + u(t), \\ D_t^{\alpha_7}\phi(t) = & p + \tan\theta\left(q\sin\phi + r\cos\phi\right), \\ D_t^{\alpha_8}\theta(t) = & q\cos\phi - r\sin\phi, \\ D_t^{\alpha_9}\psi(t) = & \dfrac{q\sin\phi + r\cos\phi}{\cos\phi}, \\ D_t^{\alpha_{10}}p(t) = & \dfrac{1}{I_x}\left[\tau_\phi + (I_y - I_z)qr\right] + \epsilon_1\cos(\phi.\theta), \\ D_t^{\alpha_{11}}q(t) = & \dfrac{1}{I_y}\left[\tau_\theta + (I_z - I_x)pr\right] + \epsilon_2\sin(p.t), \\ D_t^{\alpha_{12}}r(t) = & \dfrac{1}{I_z}\left[\tau_\psi + (I_x - I_y)pq\right] + \epsilon_3\sin(qr). \end{cases} \tag{4.10}$$

- **Stability Analysis of Control Systems**

Following the integration of the time-delayed fractional-order controller into the system (4.10), the next objective is to rigorously examine system stability in the presence of rapid and stochastic wind disturbances. To accomplish this, both the linearization approach and the stability theorem presented in Theorem 3.5.1 are employed.

To examine the stability of the system under these conditions, we use the following Theorem3.5.1 , we also use linearization method.

A numerical simulation showcasing the drone's behavior post-controller activation at is illustrated in Figure 4.6.

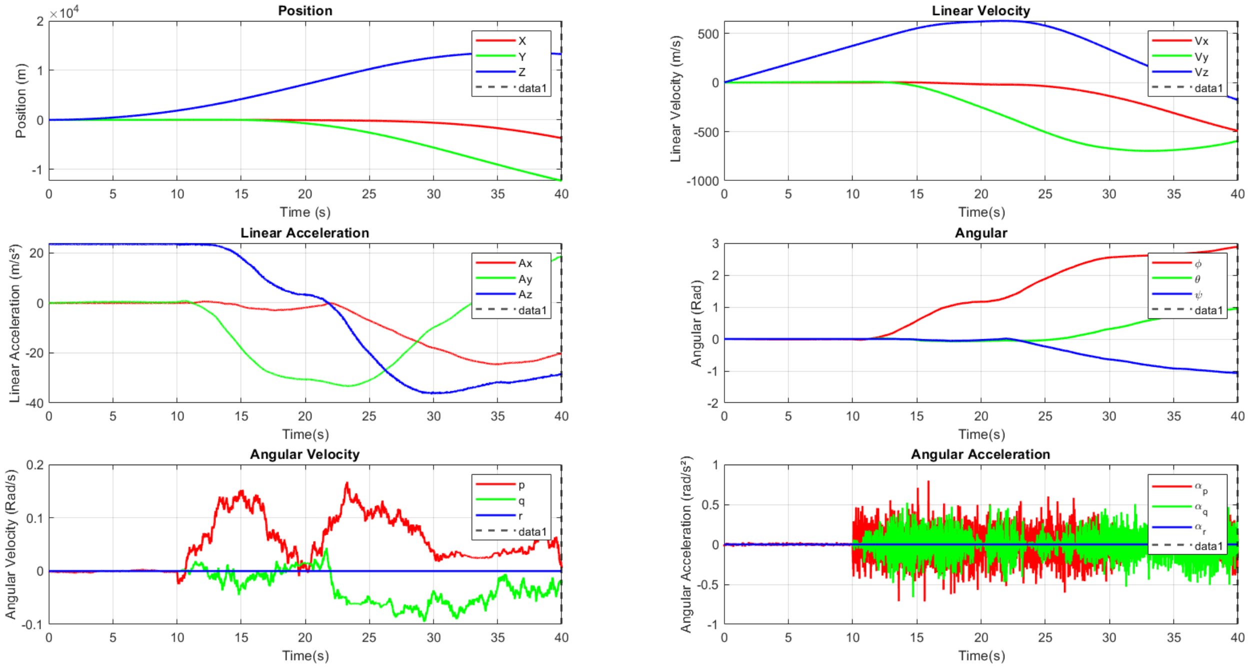


Figure 4.6: Plot of Applying Time-Delayed $PI^{\lambda}D^{\mu}$ Controller on Drone Movement and Rotation at $t = 12s$.

Figure 4.6 demonstrates the effectiveness of the time-delay fractional $PI^{\lambda}D^{\mu}$ controller in stabilize the drone's motion after strong wind turbulence. A noticeable reduction in oscillations particularly in angular velocity and acceleration is observed shortly after the controller is activated, indicating its role in suppressing system disturbances. However, minor residual oscillations persist, likely due to inherent time-delay effects and the system's nonlinear dynamics, highlighting the need for further refinement of the control strategy.

4.4 Corollary

Although the proposed time-delayed fractional $PI^{\lambda}D^{\mu}$ controller has demonstrated promising results in stabilizing chaotic UAV (drone) dynamics, a rigorous analytical proof of stability using a fractional-order **Lyapunov-Razumikhin function** remains an open challenge. The inherent complexity of fractional-order derivatives, especially in the presence of time delay, limits the direct application of classical Lyapunov-based techniques. **This opens an avenue for future research** to develop new stability criteria tailored for such systems. Further exploration may also involve adaptive or learning-based fractional controllers that can self-tune under dynamic environmental conditions.

Conclusion

In this thesis, we have conducted a comprehensive study of **time-delayed fractional chaotic systems**, motivated by the increasing demand for precise models that can accurately capture complex phenomena, particularly those characterized by nonlinear and irregular behavior. Chaos, fractional derivatives, and time delays constitute three fundamental pillars for understanding the intricate dynamics of such systems.

The first chapter presents a historical overview of fractional derivatives, outlining their most significant definitions and properties, followed by an examination of fractional chaotic dynamics through well-established examples. **The second chapter** delves into fractional delay equations, focusing on a qualitative analysis of their stability. In **the third chapter**, we introduce a delayed $PI^{\lambda}D^{\mu}$ controller as a robust mechanism for regulating chaotic system behavior. Finally, **the fourth chapter** demonstrates a practical application by simulating a drone model under three distinct scenarios.

The results obtained indicate that the integration of the proposed controller within the system significantly mitigates chaotic behavior and enhances system responsiveness. Comparative analysis across the three scenarios substantiates the controller's efficacy, particularly in turbulent environments, such as those involving wind disturbances. Nevertheless, these findings are primarily based on **numerical simulations**, **without a rigorous mathematical stability analysis.**

From this perspective, future research directions may include **developing a theoretical stability framework** using Lyapunov-based methods or fractional Laplace transforms, refining controller designs through intelligent algorithms like PSO or GA, and extending the application to multi-input, multi-output systems or more sophisticated real-world environments.

This study underscores the potential of integrating fractional calculus, chaos theory, and delayed control as a promising approach for analyzing and managing complex dynamical systems in the fields of engineering, physics, and robotics.

Finally, the thesis concludes with a general conclusion that summarizes the main findings and discusses future research directions.

Bibliography

[1] Aguila-Camacho, N., Duarte-Mermoud, M. A., & Gallegos, J. A. (2014). Lyapunov functions for fractional order systems. Communications in Nonlinear Science and Numerical Simulation, 19(8), 2951–2957.

[2] Asl, F. M., & Ulsoy, A. G. (2003). Analysis of a system of linear delay differential equations. Journal of Dynamic Systems, Measurement, and Control, 125(2), 215-223.

[3] Armanios, G., & Lind, R. (2017). Efficient Numerical Analysis of Stability of High-Order Systems With a Time Delay. arXiv.

[4] Bandyopadhyay, B., & Kamal, S. (2014). Stabilization and control of fractional order systems: A sliding mode approach. Springer.

[5] Bangti,j. (2021). Fractional differential equations: An approach via fractional derivatives. Springer.

[6] Benmouhoub, I., & Deladji, H. (2021). Etude de stabilité d'un drone quadrirotor [Master's thesis, Université 8 Mai 1945 - Guelma, Faculté des Sciences et de la Technologie, Département d'Electronique et Télécommunications].

[7] Bhalekar, S. B. (2013). Stability analysis of a class of fractional delay differential equations. Pramana – Journal of Physics, 81(2), 215–224

[8] Boukhezzar, B. (2019, April 14). Systèmes Non Linéaires: Chapitre 2 - Plan de phase, Bifurcations [Lecture notes]. Laboratoire Automatique et Robotique de Constantine (LARC), Département d'électronique, Université de Constantine 1.

[9] Caponetto, R., Dongola, G., Fortuna, L., & Petras, I. (2010). Fractional order systems: Modeling and control applications. World Scientific.

[10] Djeddi, K. (2022). Systèmes dynamiques et introduction au chaos Cours et Exercices Corrigés Dr kamel djeddi. ResearchGate

[11] Gleick, J. (1987). Chaos: Making a new science. Viking.

[12] Hale, J. K. (1977). Theory of functional differential equations. Springer-Verlag.

[13] Houmor, T. (2014). Analyse du chaos dans un système d'équations différentielles fractionnaires (Thèse de doctorat). Université Mentouri.

[14] Humphries, T. (2016, January 12–15). Delay differential equations [Lecture]. Continuation Methods in Dynamical Systems, NZMRI Summer School, Raglan, New Zealand.

[15] Jolić, M., Konjik, S., & Mitrović, D. (2022). Control theory for nonlinear fractional dispersive systems. arXiv. https://arxiv.org/abs/2212.12692

[16] Kai, D. (2004). The Analysis of Fractional Differential Equations: An Application-Oriented Exposition Using Differential Operators of Caputo Type. Springer.

[17] Kaslik, E., & Sivasundaram, S. (2012). Analytical and numerical methods for the stability analysis of linear fractional delay differential equations. Journal of Computational and Applied Mathematics, 236(15), 4027-4038.

[18] Kuang, Y. (1993). Delay Differential Equations: With Applications in Population Dynamics. Mathematics in Science and Engineering, Volume 191. Academic Press.

[19] Liu, Y., Zhao, H., & Kang, S. (2022). Existence of oscillatory solutions of fractional differential equations with distributed delays. Journal of Applied Analysis and Computation, 12(2), 807–813.

[20] Monge, C. A., Chen, Y., Vinagre, B. M., Xue, D., & Feliu, V. (2010). Fractional-order systems and controls: Fundamentals and applications. Springer.

[21] Petras, I. (2010). Fractional-order nonlinear systems: Modeling, analysis and simulation. Springer.

[22] Podlubny, I. (1999). Fractional differential equations. Academic Press. (Mathematics in Science and Engineering, Vol. 198).

[23] Shinozaki, H. (2007). Lambert W function approach to stability and stabilization problems for linear time-delay systems. Kyoto Institute of Technology.

[24] Teppstra, C. R. (2016). Delay differential equations (Bachelor's thesis). University of Groningen, Faculty of Mathematics and Natural Sciences. Supervised by Dr. A. E. Steck and Prof. dr. A. J. van der Schaft.

[25] Tavazoei, M. S., & Haeri, M. (2007). Chaos control via a simple fractional-order controller. [Journal Name], [Volume(Issue)], [Page Range]

[26] Zurich Instruments. (2023). Principles of PID controllers [White paper].

[27] https://youtu.be/lAVYDUeqdW4?si=GXCZbN6V0guCauIU

[28] https://youtu.be/pQ24NtnaLl8?si=WiQ1YuBPf7j5FuQa

Annex A: Programs in MATLAB to Calculus τ_C

```
% Setup
alpha;
A ;  % Matrix A
B;     % Matrix B
n
J = eye(n);          % Jacobian matrix at equilibrium point
I = eye(size(A));
v_vals = linspace(0.01, 10, 1000); % Candidate v values
tau_c_list = [];
for v = v_vals
s_alpha = (1i*v)^alpha;
M = s_alpha * I + B;
% Check if A*f' is invertible
if det(A*f_prime) == 0
continue;
end
E = -M / (A * f_prime);
lambda = eig(E); % Get eigenvalues
for k = 1:length(lambda)
phi = angle(lambda(k)); % Eigenvalue angle
tau_c = -phi / v;
if tau_c > 0 && isreal(tau_c)
tau_c_list = [tau_c_list, tau_c];
end end end
% Display the smallest tau_c value
if ~isempty(tau_c_list)
tau_critical = min(tau_c_list);
fprintf('Critical delay tau_c  %.4f\n', tau_critical);
else
disp('No valid tau_c found in the given range.');
end
```

Annex B

- $\Gamma(z) = \int_0^{+\infty} t^{z-1} e^{-t} dt, \quad \text{with } Re(z) > 0.$

- $\beta(x, y) = \int_0^1 t^{x-1}(1-t)^{y-1} dt$, with $x, y \in \mathbb{R}^+$.

- $E_{\alpha,\beta}(t) = \sum_{k=0}^{\infty} \frac{t^k}{\Gamma(\alpha k + \beta)}, \ \alpha, \beta > 0.$

- $\left(\begin{array}{c} \alpha \\ j \end{array} \right) = \frac{\Gamma(\alpha + 1)}{\Gamma(j+1)\Gamma(\alpha - j + 1)}.$

- ${}_aI_x^{\alpha} f(x) = {}_aD_x^{-\alpha} f(x), \qquad x > a.$

- $F(s) = L\{f(x); s\} = \int_0^{\infty} e^{-sx} f(x) dx.$

- $f(x) = L^{-1}\{F(s); x\} = \int_{c-i\infty}^{c+i\infty} e^{sx} F(s) ds, \quad c = Re(s).$

- $f(x) * g(x) = \int_0^{\infty} f(x-t) g(t) dt.$

- $L\{f(x) * g(x); s\} = F(s)G(s).$

- $L^{-1}\{F(s)G(s); x\} = f(x) * g(x).$

- $G(s) = L\{D^{\alpha} f(t); s\} = s^{\alpha} F(s)$ with $f^{(m)}(0) = 0, \quad m = \overline{1, n-1}.$

الملخص

في هذا العمل، تناولنا مشكلة التحكم في الفوضى لنظام ديناميكي كسري، حيث أُعتمد متحكم كسري من نوع $PI^{\lambda}D^{\mu}$ يتضمن تأخيرًا زمنيًا، في حين أن النظام نفسه لا يتضمن تأخيرًا.
بدأنا بعرض المفاهيم الأساسية لحساب التفاضل والتكامل الكسري، ثم قمنا بدراسة معادلات تفاضلية كسرية وتحليل استقرارها باستخدام نظرية Lyapunov–Razumikhin.
وقد تم اختبار المتحكم عددياً على أنظمة فوضوية شهيرة مثل Lotka-Volterra و Rössler.
وفي المرحلة الأخيرة، طُبِّق المتحكم على نموذج لطائرة بدون طيار، وأظهرت نتائج المحاكاة فعاليته في تحسين استقرار النظام في ظل ظروف الرياح.
كلمات مفتاحية : المتحكم، نظام كسري، الفوضى، التأخير.

Abstract

In this work, we addressed the problem of chaos control in a fractional-order dynamical system by employing a fractional $PI^{\lambda}D^{\mu}$ controller that includes a time delay, even though the system itself does not incorporate any delay.
We began by introducing the fundamental concepts of fractional calculus, followed by an analysis of fractional differential equations and their stability using the Lyapunov–Razumikhin theorem.
The controller was then tested numerically on well-known chaotic systems such as Lotka-Volterra and Rössler.
In the final phase, the controller was applied to a drone model, simulation results showed its effectiveness in enhancing system stability under wind disturbances.
Keywords: Controller, Fractional-order system, Chaos, Time delay.

Résumé

Dans ce travail, on a abordé le problème du contrôle du chaos dans un système dynamique d'ordre fractionnaire en utilisant un contrôleur fractionnaire de type $PI^{\lambda}D^{\mu}$ intégrant un délai temporel, bien que le système lui-même ne comporte aucun délai.
on a commencé par présenter les concepts fondamentaux du calcul fractionnaire, puis on a étudié les équations différentielles fractionnaires ainsi que leur stabilité à l'aide du théorème de Lyapunov–Razumikhin.
Le contrôleur a ensuite été testé numériquement sur des systèmes chaotiques bien connus tels que Lotka-Volterra et Rössler.
Enfin, le contrôleur a été appliqué à un modèle de drone. Les résultats de simulation ont démontré son efficacité à améliorer la stabilité du système en présence de perturbations dues au vent.
Mots-clés: Contrôleur, Système d'ordre fractionnaire, Chaos, Délai temporel.