République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
École Nationale Polytechnique

Département d'Automatique

in collaboration with the research center Inria de Saclay

Final project thesis
in view of obtaining the State Engineering Diploma in Automatic control

---

Finite time estimators for state and unknown inputs estimation for a class of nonlinear systems

---

FOUNAS Sidahmed and **TOUATI Samy Ali**

Under the supervision of **Dr. Messaoud CHAKIR**
and **Dr. Taous Meriem LALEG-KIRATI**

Publicly presented and defended on 25/06/2023 in front of the jury composed of :

President	Mr O. STIHI	ENP
Examinator	Pr M. TADJINE	ENP
Promoter	Dr M. CHAKIR	ENP
Co-promoter	Dr T.M. LALEG-KIRATI	INRIA SACLAY
Guest	Dr Z. BELKHATIR	UNIVERSITY OF SOUTHAMPTON

ENP 2023

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
École Nationale Polytechnique

Département d'Automatique

in collaboration with the research center Inria de Saclay

Final project thesis
in view of obtaining the State Engineering Diploma in Automatic control

Finite time estimators for state and unknown inputs estimation for a class of nonlinear systems

FOUNAS Sidahmed and **TOUATI Samy Ali**

Under the supervision of **Dr. Messaoud CHAKIR**
and **Dr. Taous Meriem LALEG-KIRATI**

Publicly presented and defended on 25/06/2023 in front of the jury composed of :

President	Mr O. STIHI	ENP
Examinator	Pr M. TADJINE	ENP
Promoter	Dr M. CHAKIR	ENP
Co-promoter	Dr T.M. LALEG-KIRATI	INRIA SACLAY
Guest	Dr Z. BELKHATIR	UNIVERSITY OF SOUTHAMPTON

ENP 2023

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
École Nationale Polytechnique

Département d'Automatique

En collaboration avec le centre de recherche Inria de Saclay

Mémoire de projet de fin d'études
en vue de l'obtention du diplôme d'Ingénieur d'État en Automatique

Estimateurs en temps fini pour l'estimation d'états et des entrées inconnues d'une classe de systèmes non linéaires

FOUNAS Sidahmed et **TOUATI Samy Ali**

Sous la supervision de **Dr. Messaoud CHAKIR**
et **Dr. Taous Meriem LALEG-KIRATI**

Présenté et soutenue publiquement le 25/06/2023 devant le jury composé de :

Président	Mr O. STIHI	ENP
Examinateur	Pr M. TADJINE	ENP
Promoteur	Dr M. CHAKIR	ENP
Co-promoteur	Dr T.M. LALEG-KIRATI	INRIA SACLAY
Invitée	Dr Z. BELKHATIR	UNIVERSITY OF SOUTHAMPTON

ENP 2023

ملخص

في هذا العمل ، سنقدم نهجين للتقدير الغير المقارب : الطريقة القائمة على الدوال المعدلة للتقدير والمراقبين ذو التقارب الزمني المحدد. سنشرح المبادئ الأساسية لهذه ونوضح أدائها من خلال أمثلة. سنركز على المراقبين القائمون على الدوال المعدلة لاكتساب فهم عميق للطريقة والتعرف عليها. سنقترح تمديد هذه المراقبين إلى فئة جديدة من الأنظمة الغير الخطية. كتطبيق للتمديد المقترح، سنرتكز على الإشكالية تم تقديمها مؤخرًا, التحكم في موقع الدرون في بيئة متنقلة. سنتناول أولاً جزء التحكم من الإشكالية ثم ننتقل إلى جزء التقدير. من أجل إظهار أداء المراقب القائم على الدوال المعدلة، سنقارنه بمراقبين آخرين يستخدمان على نطاق واسع في الأدبيات.

كلمات دالة: الاستقرار في الوقت المحدود ، المراقبون ، الاستقرار في الوقت المحدد ، الاستقرار في الوقت مُثَبَّت ، الطريقة القائمة على الدوال المعدلة ، الدرون.

Résumé

Dans ce travail, nous présenterons deux approches non asymptotiques d'estimation : la méthode basée sur les fonctions modulatrices et les observateurs à convergence en temps prescrit. Nous expliquerons les principes de bases de ces dernières et illustrerons leurs performances à travers des exemples illustratifs. Nous nous attarderons sur les observateurs basés sur les fonctions modulatrices pour acquérir une compréhension profonde de la méthode et se familiariser avec. Nous proposerons une extension de cet observateur a une nouvelle classe de systèmes non linéaires. À titre d'application de l'extension proposée, nous nous intéresserons à une problématique introduite récemment : La commande de la position d'un drone dans un environnement mobile. Nous traiterons dans un premier temps la partie contrôle de la problématique pour ensuite passer à la partie estimation. Pour démontrer les performances de l'observateur basé sur les fonctions modulatrices, nous le comparerons avec deux autres observateurs largement utilisés dans la littérature.

Mots clés: Stabilité en temps fini, observateurs, stabilité en temps prescrit, stabilité en temps fixe, méthode basée sur les fonctions modulatrices, drone.

Abstract

In this work, we'll present two non-asymptotic approaches for estimation: the modulating function-based method for estimation and observers with prescribed-time convergence. We'll explain the basic concepts of the latter and illustrate their performance through illustrative examples. We will dwell on observers based on modulating functions to gain a deep understanding of the method and become familiar with it. We will propose then an extension of the observer to a new class of nonlinear systems. As an application of the proposed extension, we will focus on a recently introduced problem: Controlling the position of a drone in a mobile environment. Initially, we'll deal with the Control part of the problem, before moving on to the Estimation part. In order to demonstrate the performance of the observer based on the modulating functions, we'll compare it with two other observers widely used in the literature.

Keywords: Finite time stability, observers, prescribed time stability, fixed-time stability, modulating functions based method, drone.

Dedication

“ *To our dear parents, our families, and our friends.* ”

- Samy & Sid Ahmed

Acknowledgements

Above all, we are grateful to Allah, the Almighty, for having given us this opportunity and granted us the ability and patience to carry out this work.

We would like to thank all those who contributed in any way to the realization of this graduation project, because on our own, we probably wouldn't have been able to do much.

We would like to thank our supervisors, Pr. Laleg-Kirati and Dr. Chakir, first of all for their continuous support throughout our final year project, for their wise guidance and advice, as well as their generous encouragement. Our gratitude extends to all the teachers in the Automatic Control Department and at the Ecole Nationale Polytechnique in general, for all the knowledge they have imparted to us during our five years of study.

Our sincere thanks also to the jury members who honored us by evaluating our work.

Finally, we would like to express our deepest gratitude to our dear parents and respective families, may God bless them, for their continuous support, encouragement and unconditional commitment throughout the realization of this graduation project.

Contents

ListofTables

ListofFigures

List of Abbreviations

AS	Asymptotically stable.
EKF	Extended kalman filter.
EKF-UI	Extended kalman filter with unknown inputs.
ES	Exponentially stable.
GAS	Globally asymptotically stable.
GES	Globally exponentially stable.
GFxTs	Globally fixed time stable.
GUA	Globally uniformly attractive.
GUAS	Globally uniformly asymptotically stable.
GUFTS	Globally uniformly finite time stable.
GUS	Globally uniformly stable.
LMI	Linear matrix inequalities.
LPDF	Locally positive definite function.
LS	Lyapunov stable.
MF	Modulating functions.
MFBM	Modulating functions based method.
MIMO	Multi input multi output.
NDF	Negative definite function.
NSDF	Negative semi-definite function.
ODE	Ordinary differential equation.
PDE	Partial differential equation.
PDF	Positive definite function.
PSDF	Positive semi-definite function.
SISO	Single input single output.
UA	Uniformly attractive.
UAS	Uniformly asymptotically stable.
UAV	Unmanned aerial vehicle.
US	Uniformly stable.

General introduction

"*Measure what is measurable, and make measurable what is not so*"
Galileo Galilei, 1564- 1642, Italy.

Precise knowledge of a system's model parameters and states is crucial to solving many control theory problems. To achieve this, sensors can be used for each variable we wish to know about, providing dynamics independent of those of the observed system. However, the sensors must be capable of acquiring measurements faster than the fastest dynamics of the system. For example, to take a sharp picture of a moving vehicle, the exposure time needs to be very short. However, there are a number of reasons why it may not be possible to use sensors exclusively. Firstly, sensors can be expensive. In addition, some sensors may be too slow compared to the dynamics of the variable to be measured, as in the case of a camera-based perception system requiring sophisticated image processing. Finally, in some applications, there may be no sensor available to measure the desired physical quantity, or even the variable we wish to know has no physical significance. Fortunately, there are ways of obtaining this information indirectly, using mathematical estimation techniques and state observers. Essentially, this involves recreating the system's internal behavior from external measurements.

By definition, a state observer is an algorithm designed to allow the convergence of state estimation dynamics towards their real values. The first observer was introduced by Kalman [36], and the concept was generalized by Luenberger [47] for deterministic linear systems. From this point onwards, observers and estimators have been widely developed and studied. The concept has been generalized to different classes of nonlinear systems, and numerous applications have since emerged [26] [38] [71] [9][84]. However, these observers are designed to work in systems where all inputs are known and measurable. Yet in many practical cases, the system behavior is affected by disturbances or non-measurable inputs. These disturbances can come from a variety of sources, such as external forces, environmental factors or unmodeled dynamics. When unknown inputs are present, conventional state observers cannot provide an accurate estimate of the system's state. This is where state observers with unknown inputs come into play. These observers are designed to estimate the state of the system taking into account the unknown inputs, thus providing a better estimate of the overall state [56] [34] [51]. These observers are also used for fault detection by generating residuals [59] [20]. Today, observers and estimators are considered as digital sensors and are used in various applications.

The earliest estimators were designed for asymptotic convergence only. This means that, in theory, it would take an infinite amount of time for the estimation error to be resolved, which is sufficient for many applications. However, the evolution of technology has shown the limits of such asymptotic convergence in many applications, and thus demonstrated the need to achieve state estimation within a specific lapse of time. The problem of observing systems within specified time limits referred to as non-asymptotic observation presents an intriguing challenge in both control theory [25] [4] and practical applications, for example in [63] where they formalized the problem of finite time chaos synchronization to secure data transmission as a nonlinear finite-time observer design which relies on the homogeneity properties of nonlinear systems. Finite-time observation offers a straightforward approach to achieve the separation principle, which allows the design and analysis of control and observation algorithms independently. Moreover, in numerous control systems problems, the transition processes are strongly restricted in time. For instance, walking robots for which each step has to be obviously achieved in a finite time [6][65]. This research field is currently booming.

Several observers that converge in finite time have been proposed [50] [85],[46], but without any doubt one of the most interesting techniques for non-asymptotic estimation must be the modulating functions based method for estimation, first introduced in [77] for parameter identification of ordinary differential equations, it has recently been extended to estimate both the state and the unknown input in real time for a restricted class of nonlinear systems [87]. This method transforms the complex non-asymptotic estimation problem into a linear algebraic system. It also allows getting rid of the initial conditions, which are often unknown. It has also provided good robustness properties, which is important in presence of noisy data [75].

Objectives

The first objective of this thesis is to contribute to improving the observation techniques based on modulating functions by proposing an extension of the current methods to a new, more general class of nonlinear systems. This extension makes it possible to estimate both the states and the unknown inputs of these systems.

The second objective of this work is to take a look into observers that ensure the stabilization of the estimation error within a finite time prescribed by the user in advance, independently of initial conditions and system's parameters, as well as analyze their performance.

Our third objective on one hand is to illustrate our extension of the modulating functions based observer through an example, and on the other hand, contribute to a problem recently tackled for the first time in [49] which deals with the position control of a drone flying in a moving environment by combining the proposed modulating functions based estimator with a sliding mode controller in a closed-loop to ensure tracking of the desired reference in this moving environment. Numerical results will be presented using noisy measurements to demonstrate the efficiency and robustness of the proposed method. In addition, the performance of the proposed method will be compared to the Extended Kalman filter with unknown input (EKF-UI) and to a sliding mode observer, in order to highlight its suitability and advantage over these classical approaches. We will also contribute to the problem of UAV's (unmanned aerial vehicle) position control in a non-inertial reference frame by implementing a new controller, namely, the backstepping controller and do a quick comparison with the sliding mode controller already developped in [49].

Description of the work

This report consists of six chapters, in addition to the general introduction and conclusion. Chapter 1 begins with a review of the concepts and definitions of classical stability according to Lyapunov, as well as existing theorems for studying its properties. We then turn to the subject of finite-time stability in its various forms, highlighting the distinction between classical asymptotic convergence and non-asymptotic convergence.

Chapter 2 is devoted to a literature review of dynamical observers. We will start by examining the notion of the system's observability, then review asymptotically converging observers for linear and nonlinear systems, as well as the non-asymptotically converging observers existing in the literature.

Chapter 3 focuses on estimation using the modulating function approach. First, we introduce the modulating function approach. We'll start by defining what a modulating function is, highlighting its properties, then present the different types of modulating functions that exist and explain the estimation approach, which we'll illustrate with a comprehensive example. Once we've familiarized ourselves with the principle, we'll present algorithms based on this approach for parameter identification, as well as for the estimation of unknown states and inputs. Finally, we'll present our contribution, which consists in extending the unknown-input observer based on modulating functions to a new, more general form of nonlinear systems.

In Chapter 4, we'll look at a specific type of finite-time observer, namely the prescribed-time observer. We begin by establishing the necessary preliminaries, then present the principles on which this observer is built. Next, we describe the method for designing this observer for a certain class of linear and nonlinear systems, as proposed in [33] and [5] respectively.

In Chapter 5, we tackle the problem of controlling the relative position of a drone in a moving environment as an illustrative example of our extension of the observer based on the modulating functions. We'll start by presenting the drone model expressed in the non-inertial reference frame (moving environment), as developed in [49]. We'll then focus on the "control" part of the problem. We will design a controller based on sliding modes, as already realized in [49], as well as a controller based on the Backstepping method. Finally, we'll compare the performance of the two controllers.

The Chapter 6 will focus on the "estimation" part of the problem of controlling the relative position of a UAV in an inertial reference frame. We will combine our extension of the modulating function-based observer with the sliding mode controller designed in the previous chapter. To evaluate the effectiveness of this approach, we will compare the performance of our method with that of the EKF-UI and the super twisting sliding mode observer.

A general conclusion will summarize the work carried out, the main results obtained as well as some potential perspectives and future work.

To show the relevance of the work conducted in this thesis, the main contributions are summarized in the following:

- An extension of the MFBM observer for state and unknown inputs to a new class of broader multi output nonlinear systems is developed.
- Improve the robustness of the estimation of unknown inputs.
- A backstepping controller is implemented to control the relative position of an UAV in a non-inertial frame of reference.
- A combination of the extended MFBM observer and a sliding mode controller is implemented to the drone problem.
- A comparative study is conducted between the extended MFBM observer, EKF-UI and the super twisting sliding mode controller in the context of the drones relative position control.

Chapter 1

Stability of dynamical systems

1.1 Introduction

Stability analysis plays an essential role in the design of control laws and observers for systems. It is a subject of great importance to researchers in the field of control theory. The literature offers numerous notions of stability, depending on the nature of the system under study, the desired performance and the environment in which the system operates. Stability in the Lyapunov sense is one of the best-known forms of stability, introduced in 1892 by Alexander Mikhailovïch Lyapunov [48]. However, there are other notions of stability that are used to solve specific practical cases, such as input-state stability (ISS) [37] and practical stability [42]. All these notions generally deal with asymptotic convergence. However, some applications require convergence in finite time, for example in the case of a walking robot where each step must be completed in a given time [66]. In recent years, this type of stability has attracted more attention from researchers, leading to the development of several approaches based on this notion. Finite-time stability means that the solutions of the dynamical system reach the equilibrium point exactly within a finite time. This type of stability, sometimes called "non-asymptotic stability" to avoid confusion, should not be confused with finite-time stability, which refers to stability over a finite time interval, as described in [43]

We begin this chapter with a reminder of the notions and definitions of stability in the Lyapunov sense. Next, we introduce finite-time stability under all of its variants, with the aim of giving a clear and concise understanding of the difference between standard asymptotic convergence and non-asymptotic convergence, along with the mathematical and practical implications of both concepts.

1.2 Stability in the sense of Lyapunov

The definitions below [37] [55] are given by considering non-autonomous systems governed by the following differential equation system :

$$\begin{cases} \dot{x} = f(t, x(t)), \\ x(t_0) = x_0. \end{cases} \tag{1.1}$$

Hypothesis 1 : The function $f : \mathbb{R}_{\geqslant 0} \times \mathbb{R}^n \to \mathbb{R}^n$ is continuous in t, locally Lipschitzian in x on the domain of study $D \subset \mathbb{R}^n$ containing the origin guaranteeing the unicity of the solution of the system (1.1).

The basic concept when studying the stability of any system is the concept of the equilibrium point.

Definition 1.2.1 :*(Equilibrium point)* The point x_e is said to be an *equilibrium point* for the system (1.1) if x_e verifies :

$$f(t, x_e) = 0.$$

Remark 1.2.1 : For the remainder of this document, we will give all the definitions and theorems for the case where the equilibrium point corresponds to the origin of $\mathbb{R}^n$ without

(a) Unstable equilibrium point (b) Stable equilibrium point

Figure 1.1: Illustration of the intuitive definition of stability [55].

loss of generality.

Definition 1.2.2 :*(Intuitive notion of stability)* If the system is initially "slightly" disturbed from its equilibrium point and the system remains "close" to this equilibrium point. In this case, the equilibrium point is said to be *stable*.

So it's the system's ability to stay close to the equilibrium point in the presence of external disturbances. Figure (1.1) visualizes this concept: if the ball in Figure (1.1.(a)) is pushed even slightly, it will move indefinitely away from its equilibrium point, it's an *unstable equilibrium point*, whereas the ball in the Figure (1.1.(b)) will always remain close to its equilibrium position, it's a *stable equilibrium point*.

Definition 1.2.3 :*(Stability in the sense of Lyapunov)* The origin is said to be *Lyapunov stable* (LS) if for each $\epsilon > 0$ and $t_0 \geqslant 0$ there exists $\delta = \delta(t_0, \epsilon) > 0$ such that :

$$||x_0|| < \delta \Rightarrow ||x(t)|| < \epsilon, \forall t \geqslant 0,$$

i.e. for any $t \geqslant t_0$, the trajectory of the system resulting from a small perturbation around the origin is always bounded by ϵ and remains close to the origin.

Remark 1.2.2 : The origin is said to be unstable in the Lyapunov sense if it is not stable in the sense of *Definition 1.2.3*.

Definition 1.2.4 :*(Attraction point)* The origin is said to be :

- *attractive equilibrium point* if there exists $r > 0$ such that :

$$||x_0|| < r \Rightarrow \lim_{t \to +\infty} ||x(t)|| = 0,$$

- *globally attractive equilibrium point* if :

$$\forall ||x_0|| \in \mathbb{R}^n \Rightarrow \lim_{t \to +\infty} ||x(t)|| = 0.$$

In the case of local attractivity, we call the region around the origin defined by r : *Region of attractivity*, if the state of the system is changed while remaining inside this region, it will return to the origin after a certain time, even if it's after an infinite time. In the case of global attractivity, the whole space $\mathbb{R}^n$ is considered as *Region of attractivity*.

Definition 1.2.5:*(Asymptotic stability)* The origin is said to be :

- *asymptotically stable equilibrium point* (AS), if it is stable and attractive.
- *globally asymptotically stable equilibrium point* (GAS), if it is stable and globally attractive.

Definition 1.2.6 :*(Uniform Boundedness)* The solutions of the system (1.1) are said *uniformly bounded*, if : $\exists a \geqslant 0$ and an increasing function $c :]0, a[\to \mathbb{R}$ such that $\forall \alpha \in]0, a[$,

$$||x_0|| < \alpha \Rightarrow ||x(t)|| < c(\alpha),\ \forall t \geqslant t_0,$$

The solutions are said *globally uniformly bounded*, if the previous property holds for $a = +\infty$.

Definition 1.2.7 :*(Uniform stability)* The origin is said to be a :

- *uniformly stable equilibrium point* (US) if : $\forall \epsilon > 0, \exists \delta = \delta(\epsilon) > 0$, such that $\forall t + 0 \geqslant 0$,
$$||x_0|| < \delta \Rightarrow ||x(t)|| < \epsilon,\ \forall t \geqslant t_0 \geqslant 0.$$
- *globally uniformly stable equilibrium point* (GUS), if it is uniformly stable and the solutions of the system are globally uniformly bounded.

Definition 1.2.8 :*(Uniform attractivity)* The origin $x = 0$ is said to be :

- *uniformly attractive equilibrium point* (UA), if : $\exists c > 0 / \forall ||x_0|| < c, \forall \epsilon > 0, \exists T := T(\epsilon, c)$ such that
$$||x(t)|| < \epsilon,\ \forall t \geqslant T + t_0.$$
- *globally uniformly attractive equilibrium point* (GUA), if :$\forall c > 0, \forall \epsilon > 0, \exists T := T(\epsilon, c)$ such that
$$||x(t)|| < \epsilon,\ \forall t \geqslant T + t_0.$$

Definition 1.2.9 :*(Uniform asymptotic stability)* The origin is said to be :

- *uniformly asymptotically stable equilibrium point* (UAS), if it is uniformly stable and uniformly attractive.
- *globally uniformly asymptotically stable equilibrium point* (GUAS), if it is globally uniformly stable and globally uniformly attractive.

Definition 1.2.10*(Exponential stability)* The origin is said to be :

- *exponentially stable equilibrium point* (ES), if there exists $\alpha > 0$ such that, for every $\epsilon > 0$, there exists $\rho(\epsilon)$ such that :

$$||x_0|| \leqslant \rho \Rightarrow ||x(t)|| < \epsilon ||x_0|| e^{-\alpha(t-t_0)},\ \forall t \geqslant t_0 \geqslant 0.$$

- *globally exponentially stable equilibrium point* (GES), if there exists $\alpha > 0$ such that, for every $\epsilon > 0$, we have :

$$\forall ||x_0|| \in \mathbb{R}^n \Rightarrow ||x(t)|| < \epsilon ||x_0|| e^{-\alpha(t-t_0)},\ \forall t \geqslant t_0 \geqslant 0.$$

The exponential stability introduces a new concept to stability, namely the *speed of convergence*. In fact, the ES induces that the system converges towards the origin faster than an exponential, α is called in this case the *rate of convergence*.

Remark 1.2.3 : The Exponential stability implies asymptotic stability, which in turn implies Lyapunov stability.

It may be noted that the above definitions of stability have a number of disadvantages :

- Explicitly calculating each solution of the system (1.1) corresponding to each initial condition is very difficult.
- These definitions are expressed in mathematical terms that are not easy to verify. (finding $\epsilon, \delta, c, \rho$, etc.)

Fortunately, there is a method for analyzing stability without having to integrate differential equations, further details can be found in A.

1.3 Finite time stability

Definition 1.3.1 :*(Finite-time stability)*[12][58][70] The origin of the system (1.1) is said to be *globally uniformly finite time stable* (GUFTS) if it is US and attractive in finite time, i.e there exists a locally bounded function $T : \mathbb{R}^n \rightarrow \mathbb{R}_{\geqslant 0} \cup 0$ such that $x(t, t_0, x_0) = 0$ for every $t \geqslant t_0 + T(x_0)$, where $x(t, t_0, x_0)$ is a solution of (1.1) with $x_0 \in \mathbb{R}^n$. The T function is called *settling time function* of the system (1.1).

Definition 1.3.2 :*(Fixed-time stability)*[67] The origin of the system (1.1) is said to be *globally fixed-time stable* (GFxTS) if it is GUFTS and the settling time function T is globally bounded by a positive real $T_{max} > 0$, i.e

$$T(x_0) \leqslant T_{max},\ \forall x_0 \in \mathbb{R}^n.$$

Remark 1.3.1 : the choice of T_{max} is not unique, for example, if $0 < T(x_0) \leqslant T_{max}$, we can also write $0 < T(x_0) \leqslant \lambda T_{max}$ with $\lambda \geqslant 1$. This drives us to define a set containing all the upper bounds of the settling time function.

Definition 1.3.3 :*(Settling time Set)*[81] Suppose that the origin of (1.1) is GFxTs. The set of all upper bounds of the settling time function is defined as follows

$$\mathcal{T} = \{T_{max} \in \mathbb{R}_{\geqslant 0} : T(x_0) \leqslant T_{max},\ \forall x_0 \in \mathbb{R}^n\}.$$

We also define the minimum upper bound of the settling time function of (1.1) by :

$$T_f = min\mathcal{T} = sup_{x_0 \in \mathbb{R}^n} T(x_0).$$

For various applications such as state estimation, dynamic optimization and fault detection, it would be preferable for the system (1.1) to stabilize within a time $T_c \in \mathcal{T}$, defined a priori as a function of the system parameters (denoted by ρ) $T_c = T_c(\rho)$. This desirable property motivates the definition of the stability in predefined time.

Definition 1.3.4 :*(Predefined time Stability).*[22] Denote by ρ the system parameters of (1.1) and let $T_c(\rho) > 0$ a design parameter. The origin of the system is said to be :

- *Globally weakly predefined-time stable*, if it is globally fixed time stable and if the settling time function satisfies

$$T(x_0) \leqslant T_c, \ \forall x_0 \in \mathbb{R}^n,$$

  In this case, T_c is called *weak predefined time.*

- *Globally strongly predefined-time stable*, if it is globally fixed time stable and if the settling time function satisfies

$$sup_{x_0 \in \mathbb{R}^n} \ T(x_0) = T_c, \ \forall x_0 \in \mathbb{R}^n,$$

  In this case, T_c is called *strong predefined time.*

Definition 1.3.5 :*(Prescribed-time stability).*[22] The origin of the system (1.1) is said to be *globally prescribed-time stable* if it is GFxTS and if every nonzero trajectory reaches the origin at exactly a desired user defined finite constant T_p after t_0, i.e :

$$T(x_0) = T_p, \ \forall x_0 \in \mathbb{R}^n.$$

Remark 1.3.1 : Based on the Definitions presented above, we can distinguish several finite-time stability criteria :

- *Finite-time convergence* refers to the situation where the settling time function depends on the initial conditions of the system(1.1).
- *Fixed-time convergence* refers to the situation where the settling time function does not depend on the system's initial conditions, and is bounded.
- *Predefined-time convergence* refers to the situation where the settling time function is bounded, but depends on the parameters of the control system.
- *Prescribed-time convergence* refers to the situation where the user can establish a priori the desired convergence time, which is independent of initial conditions and control system parameters.

The figure 1.2 summarizes the types of convergence presented in this chapter.

Example 1.3.1 :[62] Consider the following scalar systems such as $\lceil x \rfloor^k = |x|^k sign(x)$

$$f_1(x) = -k_0 \lceil x \rfloor^{a_0}, \tag{1.2}$$

$$f_2(x) = -k_0 \lceil x \rfloor^{a_0}(1 + arctan(|x|)), \tag{1.3}$$

$$f_3(x) = -(k_0 \lceil x \rfloor^{a_0} + K_\infty \phi_{a_\infty})(1 + \psi(x)), \tag{1.4}$$

$$f_4(x) = -k_0 \lceil x \rfloor^{a_0}(1 + k_1 |x|^{\frac{a_\infty - a_0}{k_2}})^{k_2}. \tag{1.5}$$

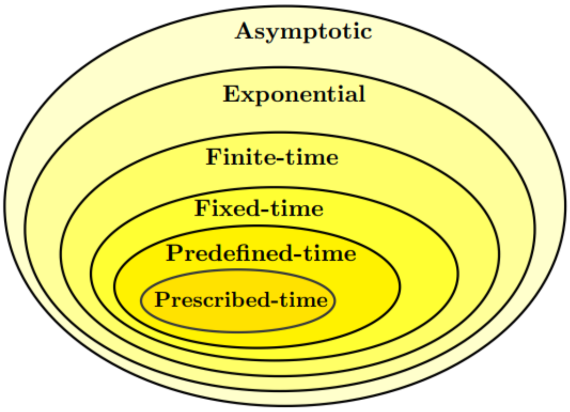

Figure 1.2: Types of convergence [5].

f	**Stability**	$T(x_0))$
$f_1(1.2)$	**GUFTS**	$= \frac{\lvert x_0\rvert^{1-a_0}}{k_0(1-a_0)}$
$f_2(1.3)$	**GUFTS**	$\leqslant \frac{\lvert x_0\rvert^{1-a_0}}{k_0(1-a_0)}$
$f_3(1.4)$	**GFxTS**	$\leqslant \frac{\pi csc(\frac{\pi(1-a_0)}{a_\infty - a_0})}{(a_\infty - a_0)k_0^{\frac{a_\infty - 1}{a_\infty - a_0}} k_\infty^{\frac{1-a_0}{a_\infty - a_0}}}$
$f_4(1.5)$	**GFxTS**	$\leqslant \frac{\pi csc(\frac{\pi(1-a_0)}{a_\infty - a_0})}{(a_\infty - a_0)k_0 k_1^{\frac{k_2(1-a_0)}{a_\infty - a_0}}}$

Table 1.1: Examples of functions f and their properties [62]

with $\psi(x)$ a positive function, k_0 and K_∞ are positive real numbers and $a_0 \in]0, 1[$, $a_{\infty > 1}$.

The table (1.1) presents the properties of each system with respect to the type of stability and the corresponding settling time function.
A logical extension of the previous theorems would be the application of Lyapunov functions, a sufficient condition for finite-time stability can be found in appendix B.

1.4 Conclusion

The aim of this chapter was to highlight the difference between classical asymptotic stability, as commonly known, and finite-time stability. For this purpose, we presented a reminder of classical stability in the Lyapunov sense of dynamical systems and then introduced the various concepts of non-asymptotic stability and characterized each of them, namely finite-time stability, fixed-time stability, predefined-time stability and prescribed-time stability. Clearly, it is far more interesting to achieve finite-time convergence (ideally prescribed-time) than to achieve the strongest form of asymptotic stability, namely exponential stability.

Chapter 2

Literature review on observers for dynamical systems

2.1 Introduction

When it is not possible to measure one or more component of the state vector directly, it is necessary to develop a program for estimating them, called an observer. The synthesis of the observer is based on the measurements, the knowledge of the inputs and the model of the system. However, observer synthesis is only possible if the system is observable.

The aim of this chapter is to introduce the problem of observing dynamical systems. First, we will define the notion of observability, and then present a method for studying the observability of nonlinear and linear systems. Next, we will review the state of the art regarding the different approaches to synthesis observers, for both asymptotic and non-asymptotic convergence, applied to linear and non-linear systems.

2.2 Observability

From the previous introduction, we can identify three basic concepts: the direct measurement of the state vector, the indirect evaluation of these components and the possibility of carrying out this indirect evaluation. The following definitions are given [18].

Definition 2.1.1 : The ability to measure a physical quantity is called the *mesurability.*

Definition 2.2.2 : A system's *observability* is the ability to evaluate all the quantities making up the state vector uniquely from measurements made on the system.

Definition 2.2.3 : The method for estimating the components of the state vector is called *observer.*

In what follows, a more exact definition of observability is presented, along with a method for investigating the latter for controlled systems of the form

$$\begin{cases} \dot{x}(t) = f(x(t), u(t)), \\ y(t) = h(x(t), u(t)), \end{cases} \tag{2.1}$$

with $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ and $h : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^p$ or n, m and p represents the dimension of the state vector, inputs, and outputs respectively.

Definition 2.2.4:*(Distinguishability)*[14] Let $y_0(t)$ and $y_1(t)$ be two output signals generated by applying the input signal $u(t)$ to the system (2.1) with initial conditions x_0 and x_1 respectively. We say that x_0 and x_1 are *distinguishable* if

$$y(t, x_0, u(t)) \neq y(t, x_1, u(t)), \forall t \geqslant 0, \text{ for all } u.$$

Otherwise, we say that x_0 and x_1 are *indistinguishable.*

Definition 2.2.5 :*(Observability)*[14] The system (2.1) is said to be *observable in* x_0 if x_0 is distinguishable from any $x \in \mathbb{R}^n$. Furthermore, the system (2.1) is said to be *observable* if $\forall x_0 \in \mathbb{R}^n$, x_0 is distinguishable.

We can study the observability of a given linear or non-linear system following the method given in C .

2.3 State of the art of observers

2.3.1 Linear systems

Luenberger proposed in [47] a simple solution for estimating the state of linear systems in a deterministic framework, while Kalman in [36] did so in a stochastic framework. In both cases, we consider the following dynamic model of the linear system

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) + \omega(t), \\ y(t) = Cx(t) + \nu(t), \end{cases} \tag{2.2}$$

where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$ and $y(t) \in \mathbb{R}^p$. A, B and C are matrices of appropriate dimensions. $\omega(t)$ and $\nu(t)$ are two gaussian white noises with a mean equals to 0 and respective covariances Q and R, they are assumed to be uncorrelated.

2.3.1.1 Luenberger observer

The basis of Luenberger's theory of observation lies mainly on the pole placement method . Let's consider the system (2.2) and assume it's observable, the noises ω and ν are zero.

The aim is to implement an asymptotic observer such that the estimate of x denoted $\hat{x}$ converges to x. In other words, set up a dynamic function $\hat{x}$ of the observable output y, to ensure $(\hat{x}(t) - x(t)) \to 0$ when $t \to \infty$. The basic principle is to take an exact copy of the dynamical system we wish to observe and add a correction term that takes into account the observation error. In the case of a simple Luenberger observer, the correction term is simply a gain L multiplied by the estimation error.

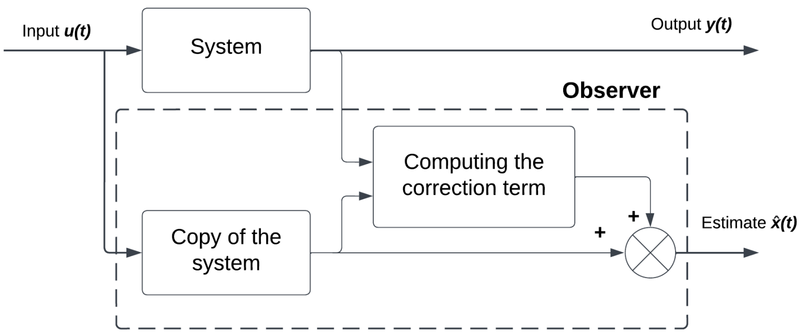

Figure 2.1: Explanatory diagram of how observers work in general

Definition 1.3.1 : A Luenberger observer $\hat{x}$ of x is a solution of the system of the form

$$\dot{\hat{x}} = \underbrace{A\hat{x}(t) + Bu(t)}_{\text{a}} + \underbrace{L(C\hat{x}(t) - y(t))}_{\text{b}}, \tag{2.3}$$

with $L \in \mathbb{R}^{n \times p}$ is called the gain matrix, chosen such that

$$\forall x_0, \hat{x}_0 \in \mathbb{R}^n, \quad \hat{x}(t) - x(t) \to 0 \text{ when } t \to \infty.$$

Part (a) of the system (2.3) corresponds to the observed system dynamics, while part (b) is the corrective term.

Let $e(t)$ be the estimation error given by $e(t) = \hat{x}(t) - x(t)$, the dynamics of the prediction error $\dot{e}(t)$ is given by the following equation

$$\dot{e}(t) = (A + LC)e(t), \tag{2.4}$$

therefore, $e(t) \to 0$ when $t \to +\infty$, $\forall e_0 \in \mathbb{R}^p$, if and only if the matrix $A + LC$ is *Hurwitz*. In order to design an asymptotic observer, we need to find a gain matrix L that makes (A + LC) Hurwitz.

2.3.1.2 Kalman filter

In a deterministic environment (ω and ν are zero), Luenberger's approach uses linear pole placement techniques to choose the gain L so that the (ALC) matrix is Herwitz. The Kalman approach, on the other hand, provides an optimal estimate in the sense of minimum variance of the estimation error in a stochastic environment, based on the results of optimal control by solving a Riccati equation and making use of the statistical properties of noise.

Consider the system (2.2), the corresponding Kalman estimator is of the form

$$\dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + K(t)(C\hat{x}(t) - y(t)). \tag{2.5}$$

The dynamics of the observation error $\dot{e}$ is written as

$$\dot{e}(t) = (A + K(t)C)e(t) + K(t)\nu(t) - \omega(t). \tag{2.6}$$

The gain of the estimator is written as follows

$$K(t) = P(t)C^T R^{-1}, \tag{2.7}$$

where $P(t)$ is the symmetrical positive-definite solution of the Riccati equation

$$\dot{P}(t) = AP(t) + P(t)A^T - P(t)C^T R^{-1} CP(t) + Q. \tag{2.8}$$

The Q and R matrices are constants used to model the noise affecting the system. The Q matrix represents the noise covariance of the process, capturing fluctuations and errors in the evolution of the system over time; a higher value of the Q matrix indicates greater uncertainty in the prediction of the system state. The R matrix represents the covariance of the noise in the measurements, i.e. the uncertainty associated with the actual measurements from the sensors. A higher value in the R matrix indicates greater uncertainty in the measurements.

The gain $K(t)$ then minimizes the covariance matrix of the estimation error $P(t) = E(e(t)e(t)^T)$.

2.3.2 Non-linear systems

For linear systems, the Kalman and Luenberger observers have given satisfactory results. The Kalman filter is suitable for stochastic systems, as it minimizes the covariance matrix of the estimation error, while the Luenberger observer is suitable for deterministic linear systems.

However, state observation for non-linear systems is more complex, and there is currently no universal method for designing observers. Existing approaches consist either of an extension of linear algorithms, or of specific nonlinear algorithms. In the first case, the extension is based on linearization of the model around an operating point. In the case of specific non-linear algorithms, work on this subject has given rise to numerous estimation algorithms. In the remainder of this chapter, we'll look at a few of these approaches.

2.3.2.1 Extended observers

It is possible to apply certain linear techniques to nonlinear systems by calculating the observer gain from the linearized model around an operating point. This is actually the case for the extended kalman filter and the extended Luenberger observer.

A- Extended kalman filter (EKF)

The extended Kalman filter is a widely studied and popular state estimation method for nonlinear dynamical systems. This approach involves applying the equations of the standard Kalman filter to a linearized nonlinear model using the first-order Taylor formula. The EKF has proven its effectiveness in many types of nonlinear processes.

However, it should be noted that the stability and convergence proofs of the extended Kalman filter cannot be generalized to all nonlinear systems. Moreover, the convergence of this estimator is local, meaning that it is valid in a restricted neighborhood of the true state. Analyzing the convergence of the extended Kalman filter remains an open problem, and is the subject of a great deal of research.

A great deal of research has been devoted to analyzing the convergence of the extended Kalman filter for nonlinear systems, and this has led to numerous publications and books on the subject [16][30].

Consider the following nonlinear system

$$\begin{cases} \dot{x} = f(x,u) + \omega(t), \\ y = h(x,u) + \nu(t). \end{cases}$$

The corresponding EKF is given by

$$\begin{aligned} \dot{x} &= f(\hat{x},u) - P(t)H(\hat{x},u)R^{-1}(h(\hat{x},u) - y), \\ \dot{P}(t) &= F(\hat{x},u)P(t) + P(t)F(\hat{x},u)^T - P(t)H(\hat{x},u)^T R^{-1} H(\hat{x},u)P(t) + Q, \end{aligned}$$

with

$$F(\hat{x}, u) = \frac{\partial f}{\partial x}(\hat{x}, u),$$
$$H(\hat{x}, u) = \frac{\partial h}{\partial x}(\hat{x}, u).$$

B- Extended Luenberger observer

The extended Luenberger observer can be used for nonlinear systems using a linearized model, either with constant gain via pole placement, or with state-dependent gain via coordinate changes. However, these approaches have their limitations and are not always applicable to all types of nonlinear systems. Coordinate change methods often require the integration of nonlinear partial differential equations, which is difficult to achieve and often necessitates the use of approximate solutions.

2.3.2.2 Change of base

This is a method that involves a change of coordinates to transform a non-linear system into a linear one, making it easier to estimate the state of the system. Once the system has been transformed, a Luenberger-type observer can be used to estimate the state of the transformed system. Using the inverse coordinate change, it is then possible to estimate the state of the original system. This technique is commonly used in the design of observers for non-linear systems, as it allows applying linear estimation approaches, which are easier to analyze and control. However, it should be noted that this approach is highly sensitive to modeling errors.

2.3.2.3 High-gain observers

High-gain observers are often used for Lipschitzian systems. They take their name from the fact that the observer's gain is chosen to be large enough to compensate for the system's non-linearity. This approach transforms a non-linear system into a linear one, making it easier to estimate the state of the system [26] [38] [71]. High-gain observers are particularly useful for non-linear systems where equations of state are unavailable or difficult to obtain. However, choosing the optimum gain can be tricky, as too high of a gain can cause unwanted oscillations or instabilities in the system.

2.3.3 Finite-time observers

2.3.3.1 Sliding mode observers

The idea behind this type of finite-time observer is to stabilize the estimation error by making it converge towards zero on a given surface. This surface is chosen in such a way that when the error reaches this surface, it becomes zero. The method used for this employs the *sign* function, which can be seen as the use of an infinite gain to crush the non-linearity in the error dynamics and establish the convergence of the finite-time observer error [9][84].

We consider the linear system (C.2), the basic principle of synthesis of such an observer is similar to that of a Luenberger observer in the sense that both approaches use the same structure, the difference lies in the choice of the correction term, in fact, the sliding mode observer uses instead the discontinuous function *sign*, so the observer takes the following form

$$\dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + Lsign(y(t) - C\hat{x}(t)), \tag{2.9}$$

with $(y(t) - C\hat{x}(t)) = 0$ the sliding surface where we want to bring the error dynamics. The gain matrix is a parameter to be determined using the method described in [23]. An extension of this method, allowing the error to converge to zero in a predefined way (convergence time independent of initial estimation error and system parameters), can be found in [69].

The main problem for this observer is the chattering phenomenon. It is a zigzag motion along the sliding surface. This phenomenon comes intrinsically with the discontinuity of the sign function. One way to remedy this is to smooth the sign function and replace it with a sigmoid. [80] [86] However, this may alter its robustness and the finite-time convergence property of this observer in the presence of perturbations [88].

2.3.3.2 Global finite-time observer

The problem of finite-time observers has been given less consideration, as it requires non-smooth techniques. Indeed, there are several methods that achieve convergence in finite time, but some of these are not continuous, such as sliding-mode observers.

One of the observers that can converge in finite time while being smooth and continuous is the global finite-time observer with large gain. Perruquetti et al initially constructed a finite-time observer for systems in observability canonical form in [64], the proof of finite-time stability being based on homogeneity properties. Subsequently, a semi-global finite-time observer was proposed by Shen and Xia in [76] based on the observer proposed by Perruquetti et al. In [54] an extension of this observer was presented, making it global.

Consider the following system

$$\begin{cases} \dot{x}_1 = x_2 + \Sigma_{j=1}^{m} g_{1,j}(x_1)u_j, \\ \dot{x}_2 = x_3 + \Sigma_{j=1}^{m} g_{2,j}(x_1, x_2)u_j, \\ \vdots \\ \dot{x}_{n-1} = x_n + \Sigma_{j=1}^{m} g_{n-1,j}(x)u_j, \\ \dot{x}_n = \phi(x) + \Sigma_{j=1}^{m} g_{n,j}(x)u_j, \\ y = x_1. \end{cases} \tag{2.10}$$

Consider the input of the bounded system $u = (u_1, \ldots, u_m)$ and assume that the functions $(g_{i,j})_{1,j\leqslant n}$ and ϕ are globally Lipschitz. The system (2.11) admits the following finite-time global observer

$$\begin{cases} \dot{\hat{x}}_1 = \hat{x}_2 + \Sigma_{j=1}^{m} g_{1,j}(\hat{x}_1)u_j + k_1([e_1]^{\alpha_1} + \rho e_1), \\ \dot{\hat{x}}_2 = \hat{x}_3 + \Sigma_{j=1}^{m} g_{2,j}(\hat{x}_1, \hat{x}_2)u_j + k_2([e_1]^{\alpha_2} + \rho e_1), \\ \vdots \\ \dot{\hat{x}}_n = \phi(\hat{x}) + \Sigma_{j=1}^{m} g_{n,j}(\hat{x})u_j + k_n([e_1]^{\alpha_n} + \rho e_1). \end{cases} \tag{2.11}$$

Details of the selection of the power α_i and the gains k_i and ρ can be found in [54].

2.4 Conclusion

In this chapter, we addressed the problem of observation, where it was pointed out that in most practical cases, measuring the entire state vector of systems is either physically or economically impossible. To overcome this limitation, observers were introduced, but they can only be implemented if the system of interest is observable. The first observers developed were dedicated to linear systems, such as the Kalman observer and the Luenberger observer. Later, they were adapted to non-linear systems using the linearization principle. Other types of observer have also been developed, each with its own properties, strengths and weaknesses.

The choice of which observer to implement depends on a number of factors, such as the specific characteristics of the system, the shape of the model, the objectives of the observation and the constraints of the application. For example, if the model of the system we wish to observe is highly simplified (with unmodeled dynamics or modeling errors), or if the system operates in a noisy environment, it is clear that the sliding mode observer, renowned for its robustness, would be clearly preferable compared to a basis change observer, which is sensitive to modeling errors. Recently, another type of observers has attracted the attention of researchers, namely finite-time convergence observers. However, the majority of these observers use non-smooth techniques or terms tending towards infinity. In the next chapter, we'll look at a finite-time estimation method that doesn't require any non-smooth or infinity-tending terms.

Chapter 3

Modulating function based method for estimation

3.1 Introduction

In this chapter, we explore the modulating function-based method (MFBM) for estimating and observing systems. The introduction of this approach dates back to the 1950s, when Shinbrot first used it in [77] and [78] for the identification of parameters of ordinary differential equations (ODEs). The method was subsequently extended to the identification of constant and variable parameters in the space of partial differential equations (PDEs) [61] [27].

Further studies then focused on the analysis of the characteristics of the modulating functions (MFs) [82] [73] [72]. More recently, the MFs approach has been adapted for real-time identification of ODE's parameters [19]. The MFBM has also been extended to state estimation and estimation of unknown inputs for ODEs and PDEs [35] [7]. In addition, it has been used to identify the parameters and fractional derivatives of fractional ODEs and PDEs [10] [45].
More recently, MFBM has been applied to certain classes of nonlinear systems, where finite-time convergence has been proposed in [87] and [39]. An integral-type observer has also been introduced in [1]. These MF based estimators offer several advantages. They offer good robustness properties thanks to the integration action. In addition, they are easy to implement and do not require knowledge of initial conditions.

In this chapter, we begin by providing fundamental definitions of modulating functions and their properties. Next, we present the different types of modulating functions found in the literature. We go on to introduce the principle of the MFBM for estimation, presenting the basic MFBM procedure, which will be illustrated with a comprehensive example. Once we've familiarized ourselves with the method, we'll present a generalization of the application of MFBM to the estimation of both constant and time-varying parameters of ordinary differential equations (ODEs) and the estimation of the state vector and unknown inputs of nonlinear systems (UIO). Finally, we present our contribution to this method, which consists of an extension of the previously illustrated UIO to a new, more general form of nonlinear system with multiple outputs.
This chapter intends to provide an in-depth understanding of the concepts and methods related to MFBM for estimation as well as to presenting our contribution to the problem.

3.2 Modulating functions

3.2.1 Definitions & Properties

Definition 3.2.1 : *(Classic modulating function)* [2] A function $\phi(t) \neq 0$ defined on an interval $[a, b]$ is called a *modulating function* (MF) of order k ($k \in \mathbb{N}^*$) if it satisfies :

(P_1) : $\phi(t) \in \mathcal{C}^k([a, b])$,

(P_2) : $\phi^{(i)}(a) = \phi^{(i)}(b) = 0, \quad i = 0, 1, \ldots, k-1,$
where i represents the derivation order.

Definition 3.2.2 : *(Generalized modulating function)* [68] Let $[a, b] \subset \mathbb{R}$, $l, k \in \mathbb{N}$ with $k \leqslant l$, and ϕ a function that satisfies the following properties :

(P_1) : $\phi(t) \in \mathcal{C}^{l+1}([a,b])$,

(P_2) : $\phi^{(i)}(a) = 0, \quad i = 0, 1, \ldots, l$,

(P_3) : $\phi^{(i)}(b) = 0, \quad i = 0, 1, \ldots, k$,
and $\phi^{k+1}(b) \neq 0$. si $k < l$.

Then, ϕ is said to be a modulating function of order (l, k) over $[a, b]$.
- If ϕ satisfies only the properties (P_1) and (P_2) for all $k \in \mathbb{N}$, then ϕ is said to be a modulating function of order $(l, -1)$ on $[a, b]$.
- If $l = k$, then ϕ is a classic modulating function of order l on $[a, b]$.

Definition 3.2.3 : *(Modulating operator)* [2] Consider the modulating function $\phi(t) \in \mathcal{C}^k([a,b])$. The associated modulating operator applied to an integrable signal $y : [a, b] \subset \mathbb{R}^+ \to \mathbb{R}$ is given by the following scalar product on the interval $I = [a, b]$:

$$\langle \phi, y \rangle_I = \int_a^b \phi(t) y(t)\, dt,$$

Property 3.2.1 :[75] Let $\phi(t)$ be a modulating function of order k defined on the interval $[0, T]$ and $f(t)$ a continuously differentiable function of order l ($f \in \mathcal{C}^l([0,T])$), so the product $\phi(t)f(t)$ is also a modulating function of order p on $[0, T]$, such that :

$$p = min(k, l), \tag{3.1}$$

Before explaining the MFBM principle, we need to mention the following Lemma, which represents the basic tool that makes this estimation method possible : *Generalized integration by parts*.

Lemma 3.2.1 : *(Generalized integration by parts)* [8] Let $f, g \in \mathcal{C}^n(\mathbb{R})$, where $n \in \mathbb{N}^*$. So for any interval $[a, b] \subset \mathbb{R}$, we have :

$$\int_a^b f(t) y^{(n)}(t) dt = (-1)^n \int_a^b f^{(n)}(t) g(t) dt + \underbrace{\sum_{k=0}^{n-1} (-1)^k \left[f^{(k)}(t) g^{(n-1-k)}(t) \right]_{t=a}^{t=b}}_{\text{S}}. \tag{3.2}$$

If we take the case where the function f is a classic modulating function, considering the property (P_2) of *Definition 3.2.1*, the term S in (3.2) is equal to 0. So by combining *Lemma 2.2.1.1* and *Definition 2.2.1*, we obtain the following property :

Property 3.2.2 : Given a measurable signal $y : [a, b] \subset \mathbb{R}^+ \to \mathbb{R}$ and ϕ a modulating function of order $n \geqslant i$ defined on $[a, b]$, then :

$$\langle \phi, y^{(i)} \rangle_I = \int_a^b \phi(t) y^{(i)}(t)\, dt = (-1)^i \int_a^b \phi^{(i)}(t) y(t) dt \tag{3.3}$$

The modulating operator allows the derivative of the measured signal to be shifted to the modulating function, which is known analytically, thus avoiding any numerical instability due to differentiation of the measured signal.

3.2.2 Examples of modulating functions

Several types of modulating functions have been proposed in the literature. Here are some of the most commonly found

3.2.2.1 Sinusoidal modulating functions

This type of modulating function was introduced in the early 1950s and was the first to be proposed by Shinbrot in [77][78], used to identify the parameters of ODEs. Sinusoidal modulating functions are given by the following general form :

$$\phi_i(t) = \sin^i(\omega_n t), \qquad \forall t \in [0, T] \text{ et } \forall i \in \mathbb{N}^*, \tag{3.4}$$

With $\omega_n = \frac{n\pi}{T}$, ou $T \in \mathbb{R}_{>0}$, $n \in \mathbb{N}^*$et $\phi_i \in \mathcal{C}^k([0, T])$, $k \geqslant i$.

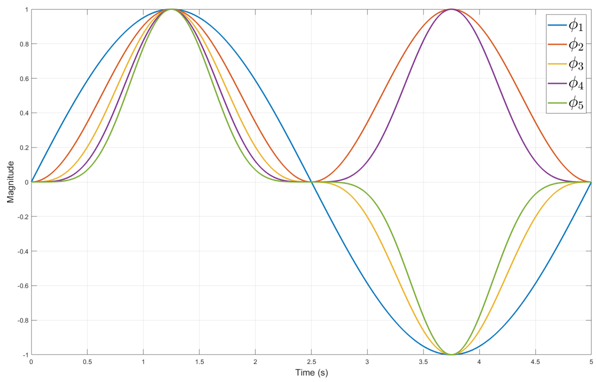

Figure 3.1: Sinusoidal modulating functions for $n = 2$ and $i = 1, \ldots, 5$

3.2.2.2 Jacobi (Polynomial) modulating functions

This type of modulating function is a combination of Jacobi polynomial functions which cancel at the boundary of the defined interval. They are expressed by [31] :

$$\phi_i(t) = t^{i_1}(T - t)^{i_2}, \qquad \forall t \in [0, T] \text{ and } \forall i_1, i_2 \in \mathbb{N}^*, \tag{3.5}$$

where $T \in \mathbb{R}_+^*$ and $\phi_i \in \mathcal{C}^k([0, T])$ with $i_1, i_2 \geqslant k$.
Polynomial modulating functions can reach values that are too large, which can cause numerical problems during implementation. This is why, in practice, we normalize these functions before using them, by dividing them by their norms, we can use for example the Euclidean norm.

3.2.2.3 Poisson's modulating functions

This type of modulating function is introduced in [74] in order to identify the parameters of a continuous lumped linear system, they are defined by :

$$\phi_i(t) = \frac{(T - t)^i}{i!} e^{[-q(T-t)]}, \qquad \forall t \in [0, T] \text{ et } \forall i \in \mathbb{N}^*, \tag{3.6}$$

where $T \in \mathbb{R}_+^*$ and $\phi_i \in \mathcal{C}^k([0, T])$, T must be large enough and q represents the degree of freedom.

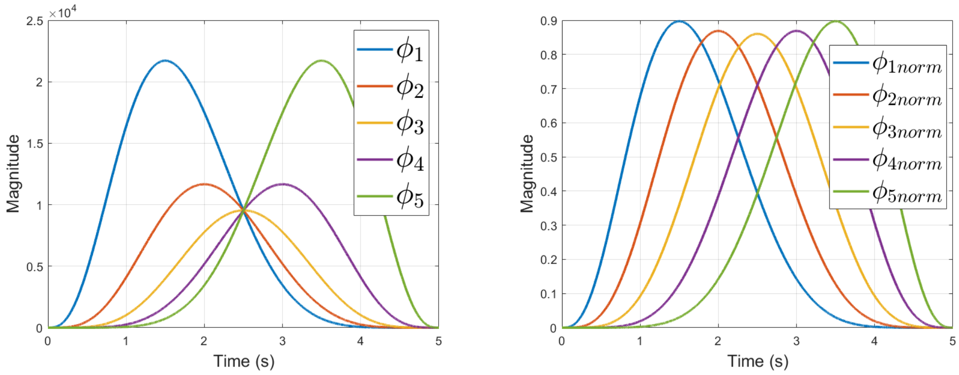

Figure 3.2: Polynomial modulating functions and normalized polynomial modulating functions for $i = 1, \ldots, 5$

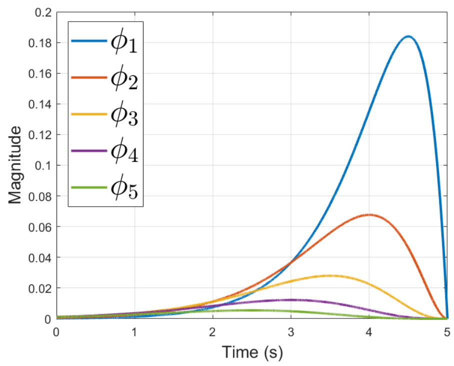

Figure 3.3: Poisson's modulating functions for $q = 2$ et $i = 1, \ldots, 5$

3.2.2.4 Other types

The table (3.1) summarizes some of the least common modulating functions and their properties.

Modulating functions	$\phi_i(t)$	Interval	Remarks
Fourier [60]	$= e^{-j\alpha i}(e^{-j\frac{2\pi}{T}t} - 1)^K$	$[0, T]$	$\phi_i \in \mathcal{C}^n([0,T])$ and $n \geqslant i$
Hermite [82]	$= (-1)^i e^{-\frac{r^2}{2}} \frac{H_i}{\sqrt{2\pi}}$, $r = t - \frac{T}{2}$, H_i is the i^{th} hermit function.	$[-\frac{L}{2}, \frac{L}{2}]$	L is sufficiently large.
Hartley [3]	$= \Sigma_{j=0}^{n}(-1)^j \begin{pmatrix} n \\ j \end{pmatrix} cas((n+i-j)\frac{2\pi}{T}t)$ with $cas(x) = cos(x) + sin(x)$	$[0, T]$	$\phi_i \in \mathcal{C}^n([0,T])$ and $n \geqslant i$

Table 3.1: Examples of modulating functions with their properties

3.3 Principle of the method based on modulating functions for estimation

3.3.1 Standard procedure

The modulating function method is a non-asymptotic estimation approach where the main idea is to transform the problem of estimating a differential equation into an algebraic problem of solving a set of linear equations. Thanks to the properties of modulating functions, this estimation approach is considered to be very fast and easy to implement. In addition, the method is robust to noise. The MFBM steps can be summarized in the following diagram.

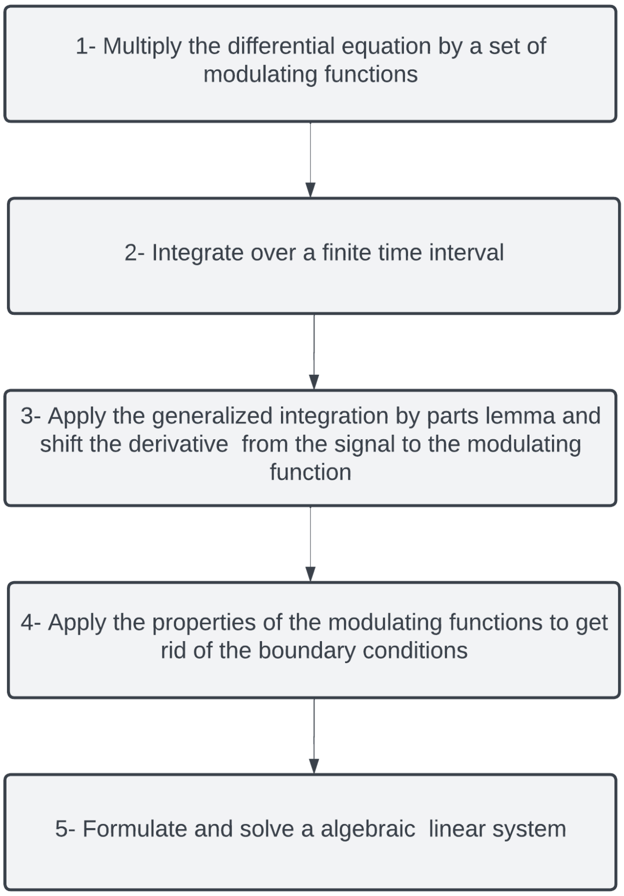

Figure 3.4: Estimation procedure using the MFBM

To better illustrate and understand the principle of this approach, let's consider the following example :

3.3.2 Illustrative example

Suppose we measure a noisy sinusoidal signal of the form :

$$y(t) = x(t) + \nu(t) = A \sin(\omega t + \phi) + \nu(t), \tag{3.7}$$

We want to estimate the parameters of the sinusoidal signal (A, ω and ϕ) and why not even the sinusoidal signal in itself and its derivative (x and $\dot{x}$) given that we have access to only the noisy signal y, where $\nu(t)$ represents the measurement noise.

Estimation of the frequency ω :
We begin by estimating the frequency ω of the sinusoidal signal, knowing that any sinusoidal signal is governed by the following second-order differential equation :

$$\ddot{x}(t) + \omega^2 x(t) = 0. \tag{3.8}$$

Knowing that $\phi_{i,j}(t)$ represents a modulating function of order (i, j). We proceed as follows :

- **Step 1 :** Multiply the differential equation (3.8) by a modulating function of order $(1, 1)$ over a finite time interval $[t - T, t]$:

$$\phi_{1,1}(\tau)\ddot{x}(\tau) + \omega^2 g_{1,1}(\tau)x(\tau) = 0, \quad \forall \tau \in [t - T, t], \tag{3.9}$$

- **Step 2 :** Integrate the equation (3.9) over the interval $[t - T, t]$:

$$\int_{t-T}^{t} \phi_{1,1}(\tau)\ddot{x}(\tau)d\tau + \omega^2 \int_{t-T}^{t} g_{1,1}(\tau)x(\tau)d\tau = 0, \tag{3.10}$$

- **Step 3 :** Apply the generalized integration by parts *Lemma* 3.2.1 in order to shift the derivative of the measured signal to the modulating function :

$$\int_{t-T}^{t} \phi_{1,1}^{(2)}(\tau)x(\tau)d\tau + \omega^2 \int_{t-T}^{t} \phi_{1,1}(\tau)x(\tau)d\tau + \Big[\phi_{1,1}(\tau)\dot{x}(\tau)\Big]_{\tau=t-T}^{\tau=t} - \Big[\dot{\phi}_{1,1}(\tau)x(\tau)\Big]_{\tau=t-T}^{\tau=t} = 0, \tag{3.11}$$

- **Step 4 :** Apply the properties of MFs to remove the boundary values $\phi_{1,1}(t-T) = \phi_{1,1}(t) = \dot{\phi}_{1,1}(t-T) = \dot{\phi}_{1,1}(t) = 0$:

$$\int_{t-T}^{t} \phi_{1,1}^{(2)}(\tau)x(\tau)d\tau + \omega^2 \int_{t-T}^{t} \phi_{1,1}(\tau)x(\tau)d\tau = 0, \tag{3.12}$$

- **Step 5 :** Replace x by the measured signal y and express $\hat{\omega}$:

$$\hat{\omega} = \sqrt{\frac{\int_{t-T}^{t} \phi_{1,1}^{(2)}(\tau)y(\tau)d\tau}{\int_{t-T}^{t} \phi_{1,1}(\tau)y(\tau)d\tau}}. \tag{3.13}$$

The equation (3.13) is very simple to calculate and on top of that, we've been able to transfer the derivative from the measured noisy signal to the modulating function, which we know analytically, this ensures a numerically more stable algorithm.

Estimation of the sinusoidal signal x :
The procedure is similar, only this time we're going to multiply the equation (3.8) by a modulating function of order $(1, 0)$. Steps 1 and 2 are identical to the previous ones, so we'll start from step 3.

- **Step 3 :** Shift the derivative from the measured signal to the modulating function :

$$\int_{t-T}^{t} \phi_{1,0}^{(2)}(\tau)x(\tau)d\tau+\omega^2 \int_{t-T}^{t} \phi_{1,0}(\tau)x(\tau)d\tau+\Big[\phi_{1,0}(\tau)\dot{x}(\tau)\Big]_{\tau=t-T}^{\tau=t}-\Big[\dot{\phi}_{1,0}(\tau)x(\tau)\Big]_{\tau=t-T}^{\tau=t}=0, \tag{3.14}$$

- **Step 4 :** Apply the properties of modulating functions of order $(1,0)$ to eliminate unwanted terms $\phi_{1,0}(t-T)=\dot{\phi}_{1,0}(t-T)=\phi_{1,0}(t)=0$ and $\dot{\phi}_{1,0}(t)\neq 0$

$$\int_{t-T}^{t} \phi_{1,0}^{(2)}(\tau)x(\tau)d\tau+\omega^2 \int_{t-T}^{t} \phi_{1,0}(\tau)x(\tau)d\tau-\dot{\phi}_{1,0}(t)x(t)=0, \tag{3.15}$$

  The order of the modulating function must be chosen according to our needs. Here, since we're estimating $x(t)$, we need to preserve the right-hand boundary conditions of x by choosing a modulating function of order 0 on the right.

- **Step 5 :** Replace x by the measured signal y and express $\hat{x}$:

$$\hat{x}(t)=\frac{1}{\dot{\phi}_{1,0}(t)}\left(\int_{t-T}^{t} \phi_{1,0}^{(2)}(\tau)y(\tau)d\tau+\omega^2 \int_{t-T}^{t} \phi_{1,0}(\tau)y(\tau)d\tau\right). \tag{3.16}$$

Considering the modulating operator, which is simply an integration and it behaves as a filter, the equation (3.16) allows us to estimate the filtered signal $x(t)$. As the integration horizon increases, the ability of noise attenuation also increases.

Estimation of the derivative $\dot{x}$:
We start again from step 3, but this time with the aim of preserving in the equation the derivative $\dot{x}$ at the right limit. To do so, we choose a modulating function of order $(1,-1)$.

- **Step 3 :** Shift the derivative from the measured signal to the modulating function :

$$\int_{t-T}^{t} \phi_{1,-1}^{(2)}(\tau)x(\tau)d\tau+\omega^2 \int_{t-T}^{t} \phi_{1,-1}(\tau)x(\tau)d\tau+\Big[\phi_{1,-1}(\tau)\dot{x}(\tau)\Big]_{\tau=t-T}^{\tau=t}-\Big[\dot{\phi}_{1,-1}(\tau)x(\tau)\Big]_{\tau=t-T}^{\tau=t}=0, \tag{3.17}$$

- **Step 4 :** Apply the properties of modulating functions of order $(1,-1)$ to eliminate unwanted terms $\phi_{1,-1}(t-T)=\dot{\phi}_{1,-1}(t-T)=0$ and $\phi_{1,-1}(t)\neq 0$,

$$\int_{t-T}^{t} \phi_{1,-1}^{(2)}(\tau)x(\tau)d\tau+\omega^2 \int_{t-T}^{t} \phi_{1,-1}(\tau)x(\tau)d\tau+\phi_{1,-1}(t)\dot{x}(t)-\dot{\phi}_{1,-1}(t)x(t)=0, \tag{3.18}$$

- **Step 5 :** Replace x by the measured signal y and express $\hat{\dot{x}}$:

$$\hat{\dot{x}}(t)=\frac{-1}{\phi_{1,-1}(t)}\left(\int_{t-T}^{t} \phi_{1,-1}^{(2)}(\tau)y(\tau)d\tau+\omega^2 \int_{t-T}^{t} \phi_{1,-1}(\tau)x(\tau)d\tau-\dot{\phi}_{1,-1}(t)y(t)\right). \tag{3.19}$$

Estimation of the amplitude and the phase :
Knowing how to compute $\omega, x(t)$ and $\dot{x}(t)$ (equations (3.13),(3.16) and (3.19)), we can

easily deduce the amplitude and phase of the sinusoidal signal from the following trigonometric equations :

$$\begin{cases} \hat{A} = \sqrt{\hat{x}^2(t) + \frac{\hat{\dot{x}}^2(t)}{\hat{\omega}^2}}, \\ \hat{\phi} = arctan\left(\hat{\omega}\frac{\hat{x}(t)}{\hat{\dot{x}}(t)}\right) - \hat{\omega}t. \end{cases} \tag{3.20}$$

Numerical implementation :
The following noisy sinusoidal signal is measured :

$$y(t) = x(t) + \nu(t) = \sin(t + \frac{\pi}{2}) + \nu(t). \tag{3.21}$$

where ν is a normally distributed random noise with variance equal to 0.1 and a mean of zero.
The measured signal y and the actual signal are shown in figure (3.5)
By using the equations (3.13),(3.16)(3.19) and (3.20) and by selecting the following normalized modulating functions (Euclidean norm) :

$$\phi_{i,j}(\tau) = \frac{\bar{\phi}_{i,j}(\tau)}{||\bar{\phi}_{i,j}(\tau)||_2}, \qquad \text{With :} \qquad \bar{\phi}_{i,j}(\tau) = (\tau - t + T)^{(i+1)}(t-\tau)^{(j+1)}, \ \forall \tau \in [t-T, t], \tag{3.22}$$

We find :

$$\begin{cases} \hat{\omega} = 1.0083, \\ \hat{A} = 0.9985, \\ \hat{\phi} = 1.5587. \end{cases}$$

The figure (3.6) shows the estimated $x(t)$ and $\dot{x}(t)$ and their real values.

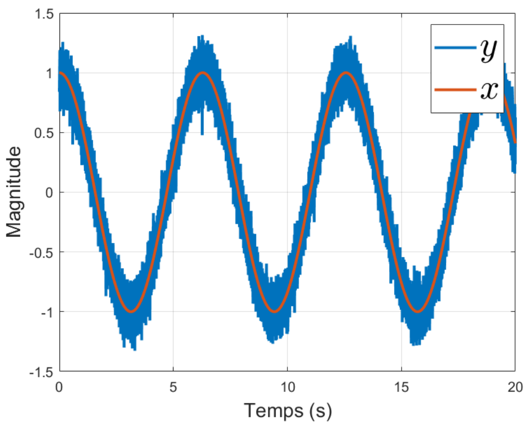

Figure 3.5: Real signal x and measured noisy signal y.

The estimates are almost identical to the real values, despite the presence of significant noise estimated at 12% of the signal x_1, thanks to the integration, which is known for its noise attenuation effects. So, using the properties of the MFs, we were able to bypass the problem of deriving a measured signal and transform it into an algebraic problem by transferring the derivation of the signal to the MF whose analytical expression is known, thus ensuring the stability of the method.

The relative errors of the sinusoidal signal estimation and its derivative are estimated at 0.17% and 0.19% respectively.

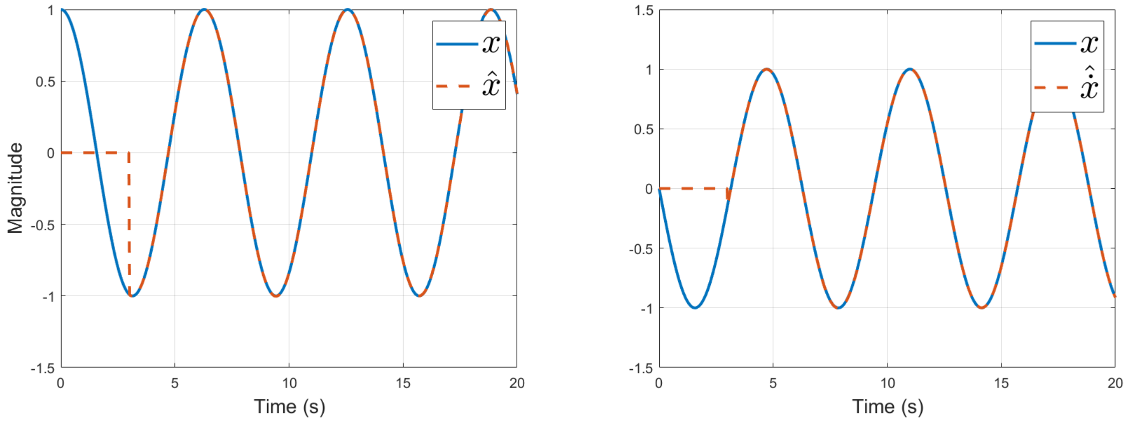

Figure 3.6: Real signals x and $\dot{x}$ and their estimates

3.3.3 The advantages of the modulating functions based estimation method

The method based on modulating functions handles a number of recurrent problems in estimation, which offers considerable advantages :

- No need to know the system's initial conditions, thanks to the properties of modulating functions on the boundaries.
- Shift the derivatives from the signal to the modulating functions that are known analytically.
- Transforms the estimation problem into a simple linear algebraic system that can be solved either by an inverse transformation (for square systems) or by least squares (for any rectangular system), as we'll see later in this chapter.
- The method is robust to external disturbances induced on the measurements, thanks to integration, which has the effect of attenuating noise.

In the remainder of this chapter, we present generalizations of this method to systems of well-defined form for identification and observation purposes, and introduce the associated algorithms.

3.4 Parameter identification by the MFBM

3.4.1 Constant parameters

Consider the system governed by the differential equation of the following form [29] :

$$\sum_{i=0}^{n} a_i \frac{d^i y(t)}{dt^i} = \sum_{i=0}^{m} b_i \frac{d^i u(t)}{dt^i}, \quad n > m \tag{3.23}$$

with $y(t)$ the system output, $u(t)$ the input and a_i and b_i the system parameters we want to identify. They are assumed to be constant in $\mathbb{R}$. Without loss of generality, we also assume that $a_0 = 1$.

- **Step 1 :** Multiply the equation (3.23) by a modulating function $\phi(t) \in \mathcal{C}^{m+n+1}([0,T])$:

$$\sum_{i=0}^{n} a_i \phi(\tau) \frac{d^i y(\tau)}{d\tau^i} = \sum_{i=0}^{m} b_i \phi(\tau) \frac{d^i u(\tau)}{d\tau^i}. \tag{3.24}$$

- **Step 2 :** Integrate the equation (3.24) over the interval $I = [0,T]$:

$$\sum_{i=0}^{n} a_i \int_0^T \phi(\tau) \frac{d^i y(\tau)}{d\tau^i} d\tau = \sum_{i=0}^{m} b_i \int_0^T \phi(\tau) \frac{d^i u(\tau)}{d\tau^i} d\tau. \tag{3.25}$$

- **Step 3 :** Apply property (3.3), the equation (3.25) becomes :

$$\sum_{i=0}^{n} (-1)^i a_i \int_0^T \frac{d^i \phi(\tau)}{d\tau^i} d\tau = \sum_{i=0}^{m} (-1)^i b_i \int_0^T \frac{d^i \phi(\tau)(\tau)}{d\tau^i} u(\tau) d\tau. \tag{3.26}$$

  We assume that y and u are known, we note :

$$\langle \phi^{(i)}, y \rangle_I = \int_0^T \phi^{(i)}(\tau) y(\tau)\, d\tau, \qquad \langle \phi^{(i)}, u \rangle_I = \int_0^T \phi^{(i)}(\tau) u(\tau)\, d\tau, \tag{3.27}$$

  Using numerical integration methods, $\langle \phi^{(i)}, y \rangle_I$ and $\langle \phi^{(i)}, u \rangle_I$ can be calculated.

- **Step 4 :** Rewrite the equation (3.26) and replace with (3.27). (3.26) becomes :

$$\langle \phi, y \rangle_I + \sum_{i=1}^{n} (-1)^i a_i \langle \phi^{(i)}, y \rangle_I = \sum_{i=0}^{m} (-1)^i b_i \langle \phi^{(i)}, u \rangle_I, \tag{3.28}$$

$$\sum_{i=0}^{m} (-1)^i b_i \langle \phi^{(i)}, u \rangle_I - \sum_{i=1}^{n} (-1)^i a_i \langle \phi^{(i)}, y \rangle_I = \langle \phi, y \rangle_I. \tag{3.29}$$

  The equation (3.29) can be rewritten in the following vector form :

$$\begin{bmatrix} \langle \phi, u \rangle_I & \dots & (-1)^m \langle \phi^{(m)}, u \rangle_I & \langle \phi^{(1)}, y \rangle_I & \dots & (-1)^{(n+1)} \langle \phi^{(n)}, u \rangle_I \end{bmatrix} \begin{bmatrix} b_0 \\ \vdots \\ b_m \\ a_1 \\ \vdots \\ a_n \end{bmatrix} = \langle \phi, y \rangle_I, \tag{3.30}$$

  The equation (3.30) is an algebraic system with $n+m+1$ unknown coefficients. To solve it, we need at least $n+m+1$ different modulating functions, to generate at least $n+m+1$ different equations.

- **Step 5 :** Construct from the equation (3.30) an algebraic system of at least $(n+m+1)$ equations, giving the following algebraic system :

$$A(T)\hat{\theta} = B(T), \tag{3.31}$$

where $A(T) \in \mathbb{R}^{(M)\times(m+n+1)}$, $B(t) \in \mathbb{R}^{M}, \theta \in \mathbb{R}^{(m+n+1)}$ and M is the number of modulating functions used.

With :

$$A(T) = \begin{bmatrix} \langle \phi_1, u\rangle_I & \dots & (-1)^m \langle \phi_1^{(m)}, u\rangle_I & \langle \phi_1^{(1)}, y\rangle_I & \dots & (-1)^{(n+1)} \langle \phi_1^{(n)}, u\rangle_I \\ \langle \phi_2, u\rangle_I & \dots & (-1)^m \langle \phi_2^{(m)}, u\rangle_I & \langle \phi_2^{(1)}, y\rangle_I & \dots & (-1)^{(n+1)} \langle \phi_2^{(n)}, u\rangle_I \\ \vdots & \dots & \vdots & \vdots & \dots & \vdots \\ \langle \phi_M, u\rangle_I & \dots & (-1)^m \langle \phi_M^{(m)}, u\rangle_I & \langle \phi_1^{(1)}, y\rangle_I & \dots & (-1)^{(n+1)} \langle \phi_M^{(n)}, u\rangle_I, \end{bmatrix} \tag{3.32}$$

and :

$$B(T) = \begin{bmatrix} \langle \phi_1, y\rangle_I \\ \langle \phi_2, y\rangle_I \\ \vdots \\ \langle \phi_M, y\rangle_I \end{bmatrix} \quad , \quad \hat{\theta} = \begin{bmatrix} b_0 \\ \vdots \\ b_m \\ a_1 \\ \vdots \\ a_n \end{bmatrix}. \tag{3.33}$$

If we take $M = m + n + 1$, the algebraic system (3.31) becomes square. Then, if $A(T)$ is invertible, the parameter estimates of the system (3.23) are given by :

$$\hat{\theta} = A^{-1}(T)B(T). \tag{3.34}$$

If we take $M > m+n+1$, in this case the algebraic system (3.31) is solved by least squares, the estimation of the system parameters (3.23) is given by :

$$\hat{\theta} = (A^T(T)A(T))^{-1}A^T(T)B(T). \tag{3.35}$$

3.4.2 Time-varying parameters

MFBM can also be used to estimate the parameters of a time-varying system. To do so, the parameters $a_i(t), b_i(t)$ are projected onto a known basis of functions called *basis functions*. Now, we'll have to estimate the coefficients of the chosen bases, which reduces the problem of estimating time-varying parameters to a problem of estimating constant parameters again.

We project the parameters of the system (3.23) a_i and b_i into the space generated by a set of basis functions $\{\beta_j(t)\}_{j=1}^{N}$, $N \in \mathbb{N}$ such that

$$\begin{cases} a_i(t) = \Sigma_{j=1}^{N_a} \alpha_{i,j}^a \beta_j(t) & i = 1, 2, \dots, n, \\ b_i(t) = \Sigma_{j=1}^{N_b} \alpha_{i,j}^b \beta_j(t) & i = 0, 1, \dots, m, \end{cases} \tag{3.36}$$

where the coefficients $\alpha_{i,j}^a$ and $\alpha_{i,j}^b$ are the projection parameters to be estimated and N_a and N_b are the numbers of basis function used for a_i and b_i respectively.

Substituting the equation (3.36) into (3.23), we follow the same steps above to obtain the following algebraic system for estimating the projection parameters

$$A(T) = \begin{bmatrix} A_1 & A_2 \end{bmatrix}, \tag{3.37}$$

where

$$A_1 = \left[\begin{array}{ccc|c|ccc} \langle g_{1,1}, u\rangle_I & \dots & \langle g_{1,N_b}, u\rangle_I & \dots & (-1)^{(m)}\langle g_{1,1}^{(m)}, u\rangle_I & \dots & (-1)^{(m)}\langle g_{1,N_b}^{(m)}, u\rangle_I \\ \langle g_{2,1}, u\rangle_I & \dots & \langle g_{2,N_b}, u\rangle_I & \dots & (-1)^{(m)}\langle g_{2,1}^{(m)}, u\rangle_I & \dots & (-1)^{(m)}\langle g_{2,N_b}^{(m)}, u\rangle_I \\ \vdots & \dots & \vdots & \dots & \vdots & \dots & \vdots \\ \langle g_{M,1}, u\rangle_I & \dots & \langle g_{M,N_b}, u\rangle_I & \dots & (-1)^{(m)}\langle g_{M,1}^{(m)}, u\rangle_I & \dots & (-1)^{(m)}\langle g_{M,N_b}^{(m)}, u\rangle_I \end{array}\right], \tag{3.38}$$

$$A_2 = \left[\begin{array}{ccc|c|ccc} \langle g_{1,1}^{(1)}, y\rangle_I & \dots & \langle g_{1,N_a}^{(1)}, y\rangle_I & \dots & (-1)^{(n+1)}\langle g_{1,1}^{(n)}, y\rangle_I & \dots & (-1)^{(n+1)}\langle g_{1,N_a}^{(n)}, y\rangle_I \\ \langle g_{2,1}^{(1)}, y\rangle_I & \dots & \langle g_{2,N_a}^{(1)}, y\rangle_I & \dots & (-1)^{(n+1)}\langle g_{2,1}^{(n)}, y\rangle_I & \dots & (-1)^{(n+1)}\langle g_{2,N_a}^{(n)}, y\rangle_I \\ \vdots & \dots & \vdots & \dots & \vdots & \dots & \vdots \\ \langle g_{M,1}^{(1)}, y\rangle_I & \dots & \langle g_{M,N_a}^{(1)}, y\rangle_I & \dots & (-1)^{(n+1)}\langle g_{M,1}^{(n)}, y\rangle_I & \dots & (-1)^{(n+1)}\langle g_{M,N_a}^{(n)}, y\rangle_I \end{array}\right], \tag{3.39}$$

$$g_{i,j} = \phi_i(t)\beta_j(t), \tag{3.40}$$

and

$$B(T) = \begin{bmatrix} \langle \phi_1, y\rangle_I \\ \langle \phi_2, y\rangle_I \\ \vdots \\ \langle \phi_M, y\rangle_I \end{bmatrix}, \tag{3.41}$$

$$\hat{\theta} = \left[\begin{array}{ccc|c|ccc|ccc|c|ccc} \alpha_{0,1}^b & \dots & \alpha_{0,N_b}^b & \dots & \alpha_{m,1}^b & \dots & \alpha_{m,N_b}^b & \alpha_{1,1}^a & \dots & \alpha_{1,N_a}^a & \dots & \alpha_{n,1}^a & \dots & \alpha_{n,N_a}^a \end{array}\right]^T, \tag{3.42}$$

$g_{i,j}$ is a modulating function according to the *Property* 3.2.1 (3.1) and $A(T) \in \mathbb{R}^{(M)\times(N_a\times n+N_b\times(m+1))}$, $B(t) \in \mathbb{R}^M$,$\theta \in \mathbb{R}^{(N_a\times n+N_b\times(m+1))}$ and M is the number of modulating functions.

3.4.2.1 Numerical example

Consider the simple system governed by the following time-varying first-order differential equation :

$$a(t)y^{(1)}(t) + y(t) = 0, \quad t \in [0, 10], \tag{3.43}$$

with $a(t)$ a real time-varying parameter given by :

$$a(t) = 0.5 + t. \tag{3.44}$$

The system is simulated for the initial condition $y(0) = 3$. The parameter $a(t)$ is estimated in the case of a noiseless environment and in the case of a noisy environment by adding to the measurements a normally distributed random noise of variance equal to 0.1.

We decompose $a(t)$ into 3 polynomial bases, i.e. :

$$a(t) = \sum_{i=0}^{3} \alpha_i t^i. \tag{3.45}$$

From the equations (3.37-3.42), we obtain the following algebraic system :

$$\begin{bmatrix} \langle g_{1,1}^{(1)}, y\rangle_I & \langle g_{1,2}^{(1)}, y\rangle_I & \langle g_{1,3}^{(1)}, y\rangle_I \\ \vdots & \vdots & \vdots \\ \langle g_{M,1}^{(1)}, y\rangle_I & \langle g_{M,2}^{(1)}, y\rangle_I & \langle g_{M,3}^{(1)}, y\rangle_I \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix} = \begin{bmatrix} \langle \phi_1, y\rangle_I \\ \vdots \\ \langle \phi_M, y\rangle_I \end{bmatrix}, \tag{3.46}$$

where M is the number of modulating functions and :

$$g_{i,j} = \phi_i(t) \times t^{(i-1)}. \tag{3.47}$$

The modulating functions used in the simulations are polynomial modulating functions given by :

$$\phi_i(t) = \frac{\bar{\phi}_i(t)}{||\bar{\phi}_i(t)||_2}, \qquad \text{Avec :} \qquad \bar{\phi}_i(t) = t^{(2+i-1)}(10-t)^{(2+i-1)}, \tag{3.48}$$

for $i = 1, \ldots, M$.

The figure (3.7) represents the measured signal without and with noise, while the figure (3.8) represents the estimate of $a(t)$ in the case of an environment without and with noise.

We have also simulated the absolute errors of estimation for different numbers of modulating functions in the case of a noise-free and noise-affected environment represented by the figures (3.9) and (3.10) respectively.

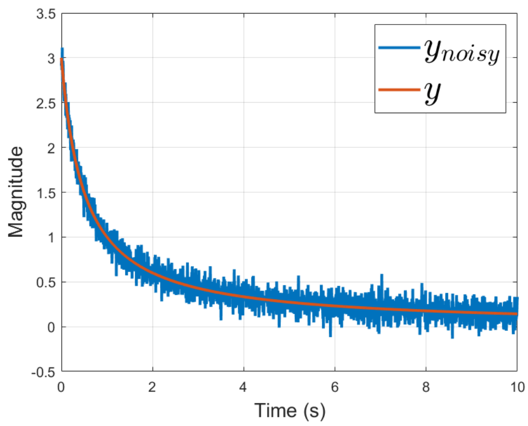

Figure 3.7: Measured signal with and without noise

From the figures, we can see that in both cases (with and without noise) the MFBM provided very satisfactory results, almost identical to the true values despite the presence of noise, and this is due to the integral which has the effect of attenuating the noise. We can also see that increasing the number of modulating functions doesn't necessarily produce better results. Consequently, sticking to a small number of modulating functions is preferable for ease of calculation and simplicity.

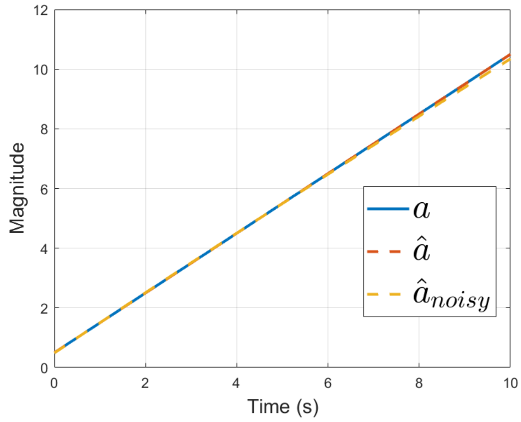

Figure 3.8: Estimation of the parameter $a(t)$ in a noisy and noiseless environment for $M = N = 3$

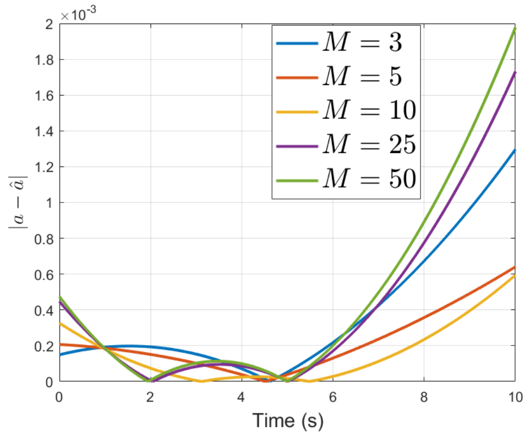

Figure 3.9: Absolute estimation error in a noise-free environment for different numbers of modulating functions

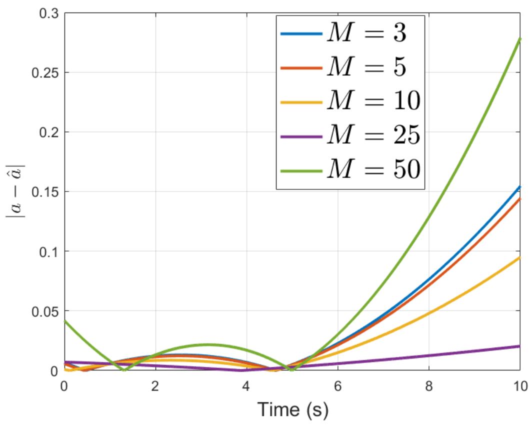

Figure 3.10: Absolute estimation error in a noise-free environment for different numbers of modulating functions

3.5 State and unknown input estimation using MFBM

The method based on modulating functions can also be used to estimate the states of a system, as seen in the sinusoid example, it can also be used to estimate unknown inputs. In what follows, we'll present a general observer algorithm based on the MFBM associated with nonlinear systems of a well-defined form.

3.5.1 Offline estimation

Let's consider the system governed by the following differential equation [87]

$$\begin{cases} \frac{d^n x}{dt^n} = f(x,u) + g(t,x,u), \\ y = x. \end{cases} \tag{3.49}$$

Represented in state space in the following canonical triangular form :

$$\begin{cases} \dot{x} = \begin{bmatrix} \hat{x}_1 \\ \vdots \\ \hat{x}_{n-1} \\ \hat{x}_n \end{bmatrix} = \begin{bmatrix} x_2 \\ \vdots \\ x_n \\ f(x,u) + g(t,x,u) \end{bmatrix}, \\ y = x, \end{cases} \tag{3.50}$$

where $x \in \mathbb{R}^n$ is the system state vector, $u \in \mathbb{R}^m (m \in \mathbb{N}^*)$ is the bounded input vector, $y \in \mathbb{R}$ is the measured output. $f(x,u)$ is a known nonlinear function and $g(t,x,u)$ is an unknown bounded nonlinear function which includes model uncertainty and disturbances. f and g are assumed to be continuous and locally Lipschitz.
The system (3.50) has a canonical structure that decouples the unknown parameters of the nonlinear function from the unmeasured states. As a result, both estimates can be reconstructed under the assumption of constant excitation persistence $u(t)$ and $y(t)$.

MFBM estimates states and disturbance in two main phases :

- **Step 1 :** Having the output measurements $y(t)$, we estimate the states $x_2, \ldots, x_n$ using the $(n-1)$ first equations of the system (3.50) for $t \in [0,T]$.
- **Step 2 :** In the last equation of the system (3.50), the inputs $u(t)$ and the output $y(t)$ are substituted by the estimation of the states $\hat{x}_2, \ldots, \hat{x}_n$, in order to estimate the disturbance g.

3.5.1.1 State estimation

MFBM estimation for triangular systems offers the advantage that each state estimate is independent of the rest of the states and depends only on the measured output. y.

- **Step 1 :** Multiply the $(n-1)$ first equations of (3.50) by a modulating function ϕ

$$\begin{bmatrix} \phi(t)\dot{x}_1 \\ \vdots \\ \phi(t)\ \dot{x}_{n-1} \end{bmatrix} = \begin{bmatrix} \phi(t)x_2 \\ \vdots \\ \phi(t)x_n \end{bmatrix}. \tag{3.51}$$

We replace the first-order derivatives of the states $(\dot{x}_2, \ldots, \dot{x}_{n-1})$ by the higher-order derivatives of the measured output y

$$\begin{bmatrix} \phi(t)\dot{y} \\ \vdots \\ \phi(t)y^{(n-1)} \end{bmatrix} = \begin{bmatrix} \phi(t)x_2 \\ \vdots \\ \phi(t)x_n \end{bmatrix}. \tag{3.52}$$

- **Step 2 :** Apply the modulating operator to the system (3.52) over the interval $I = [0, T]$

$$\begin{bmatrix} \langle \phi, \dot{y} \rangle_I \\ \vdots \\ \langle \phi, \ y^{(n-1)} \rangle_I \end{bmatrix} = \begin{bmatrix} \langle \phi, x_2 \rangle_I \\ \vdots \\ \langle \phi, x_n \rangle_I \end{bmatrix}. \tag{3.53}$$

- **Step 3 :** Apply the property (3.3), the equation (3.54) becomes :

$$\begin{bmatrix} \langle \phi, x_2 \rangle_I \\ \vdots \\ \langle \phi, x_n \rangle_I \end{bmatrix} = \begin{bmatrix} -\langle \dot{\phi}, y \rangle_I \\ \vdots \\ (-1)^{(n-1)} \langle \phi^{(n-1)}, y \rangle_I \end{bmatrix}. \tag{3.54}$$

Since x_j, $j = 2, \dots, n$ are time-varying, we decompose each into a space spanned by a set of unknown coefficients a_i^j and by known basis functions $\alpha_i(t)$ such that :

$$\hat{x}_j(t) = \sum_{i=1}^{N} \hat{a}_i^j \alpha_i(t). \tag{3.55}$$

- **Step 4 :** Substitute by (3.55) in (3.54), we find :

$$\langle \phi, \hat{x}_j \rangle_I = (-1)^{(j-1)} \langle \phi^{(j-1)}, y \rangle_I = \sum_{i=1}^{N} \hat{a}_i^j \langle \phi, \alpha_i \rangle_I. \tag{3.56}$$

The equation (3.56) can be rewritten in the following vector form :

$$\begin{bmatrix} \langle \phi, \alpha_1 \rangle_I & \dots & \langle \phi, \alpha_N \rangle_I \end{bmatrix} \begin{bmatrix} \hat{a}_1^j \\ \vdots \\ \hat{a}_N^j \end{bmatrix} = (-1)^{(j-1)} \langle \phi^{(j-1)}, y \rangle_I. \tag{3.57}$$

The equation (3.57) is an algebraic system with N unknown coefficients. To solve it, we need at least N different modulating functions of order $l \geqslant j$, to generate at least N different equations.

- **Step 5 :** Construct from the equation (3.57) an algebraic system of at least N equations, we obtain :

$$A_j(T) \hat{\theta}_j = B_j(T). \tag{3.58}$$

With :

$$A_j(T) = \begin{bmatrix} \langle \phi_1, \alpha_1 \rangle_I & \dots & \langle \phi_1, \alpha_N \rangle_I \\ \vdots & \dots & \vdots \\ \langle \phi_M, \alpha_1 \rangle_I & \dots & \langle \phi_M, \alpha_N \rangle_I \end{bmatrix}, \tag{3.59}$$

$$B_j(T) = (-1)^{(j-1)} \begin{bmatrix} \langle \phi_1^{(j-1)}, y \rangle_I \\ \vdots \\ \langle \phi_M^{(j-1)}, y \rangle_I \end{bmatrix} \quad , \quad \hat{\theta}_j = \begin{bmatrix} \hat{a}_1^j \\ \vdots \\ \hat{a}_N^j \end{bmatrix}. \tag{3.60}$$

3.5.1.2 Disturbance estimation

Having estimated the states x_j, $j = 2, \ldots, n$ in addition to having measured the inputs $u(t)$ and the output y we can now proceed to estimate the unknown input $d(t)$. Consider the ($n-$ th) equation of (3.50)

$$\dot{x}_n = f(x,u) + d(t). \tag{3.61}$$

The unknown input is therefore expressed by :

$$d(t) = \dot{x}_n - f(x,u). \tag{3.62}$$

The steps involved in estimating the disturbance are as follows :

- **Step 1 :** Multiply the equation (3.62) by a modulating function ϕ :

$$\phi(t)d(t) = \phi(t)\dot{x}_n - \phi(t)f(x,u) = \phi(t)y^{(n)} - \phi(t)f(x,u). \tag{3.63}$$

- **Step 2 :** Apply the modulating operator to equation (3.63) over the interval $I = [0,T]$

$$\langle \phi, d \rangle_I = \langle \phi, y^{(n)} \rangle_I - \langle \phi, f(x,u) \rangle_I. \tag{3.64}$$

- **Step 3 :** According to the property (3.3), the equation (3.64) becomes :

$$\langle \phi, d \rangle_I = (-1)^{(n)} \langle \phi^{(n)}, y \rangle_I - \langle \phi, f(x,u) \rangle_I. \tag{3.65}$$

The disturbance is time varying, so it can be decomposed in the space spanned by a set of unknown coefficients $\hat{b}_i$ and G known basis functions $\beta_i(t)$ chosen such that :

$$\hat{d}(t) = \sum_{i=1}^{G} \hat{b}_i \beta_i(t). \tag{3.66}$$

- **Step 4 :** Substitute with (3.66) in (3.65) and x by $\hat{x}$, we find :

$$\sum_{i=1}^{G} \hat{b}_i \langle \phi, \beta_i \rangle_I = (-1)^{(n)} \langle \phi^{(n)}, y \rangle_I - \langle \phi, f(\hat{x},u) \rangle_I. \tag{3.67}$$

The equation (3.67) can be rewritten in the following vector form :

$$\begin{bmatrix} \langle \phi, \beta_1 \rangle_I & \ldots & \langle \phi, \beta_G \rangle_I \end{bmatrix} \begin{bmatrix} \hat{b}_1 \\ \vdots \\ \hat{b}_G \end{bmatrix} = (-1)^{(n)} \langle \phi^{(n)}, y \rangle_I - \langle \phi, f(\hat{x},u) \rangle_I. \tag{3.68}$$

To solve (3.68), we need at least G different equations, for which we use $W \geqslant G$ different modulating functions of order $l \geqslant n$.

- **Step 5 :** From the equation (3.68), we generate an algebraic system of at least W equations and we find :

$$A_d(T)\hat{\theta}_d = B_d(T), \tag{3.69}$$

with :

$$A_d(T) = \begin{bmatrix} \langle \phi_1, \beta_1 \rangle_I & \dots & \langle \phi_1, \beta_G \rangle_I \\ \vdots & \dots & \vdots \\ \langle \phi_W \beta_1 \rangle_I & \dots & \langle \phi_W, \beta_G \rangle_I \end{bmatrix}, \tag{3.70}$$

and

$$B_d(t) = \begin{bmatrix} (-1)^{(n)} \langle \phi_1^{(n)}, y \rangle_I - \langle \phi_1, f(\hat{x}, u) \rangle_I \\ \vdots \\ (-1)^{(n)} \langle \phi_W^{(n)}, y \rangle_I - \langle \phi_W, f(\hat{x}, u) \rangle_I \end{bmatrix} \quad , \quad \hat{\theta}_d = \begin{bmatrix} b_1 \\ \vdots \\ b_G \end{bmatrix}. \tag{3.71}$$

3.5.2 Online Estimation

Up to now, we have presented MFBM in its offline form, i.e. the experimental signals are first measured so that the desired variables (parameters, states or disturbances) can then be estimated. However, in the majority of control problems (e.g. state feedback control, adaptive control, fault detection ... and so on), the estimation must take place in real time. Estimation must be carried out in real time (online). For this purpose, the sliding window strategy was proposed in [75].

The principle of this approach is to apply the MFBM previously seen on a short time window $[t - \tau, t]$ which at each clock top, progresses by one step allowing the estimation of the variable of interest on the current time window, so the estimate of the variable of interest at each instant is the last estimate on the window. More details are given below.

Let I_{h_k} be the sliding integration window and h its width. At each clock top, the integration window progresses by one sampling step, such that $I_{h_k} = [t_{l_k}, t_{u_k}]$ where t_{l_k} and t_{u_k} are the lower and upper bounds of the window respectively. Modulating functions of order l must therefore satisfy the following requirements :

$$(P_1) \ : \ \phi(t) \in \mathcal{C}^l[t_{l_k}, t_{u_k}]),$$

$$(P_2) \ : \ \phi^{(i)}(t_{l_k}) = \phi^{(i)}(t_{u_k}) = 0, \qquad i = 0, 1, \dots, l-1.$$

At each new sampling cycle, we substitute I by I_{h_k} updated in the system (3.58-3.60) and then the system (3.69-3.71), enabling us to calculate the estimates. $\hat{x}_j(\tau)$ for $j = 2, \dots, n$ and $\hat{d(\tau)}$ respectively, for $\tau \in I_{h_k} = [t_{l_k}, t_{u_k}] = [t - h, t]$, simply replace by t to retrieve the instantaneous estimates which can now be used for control calculation or fault detection. The procedure is described in greater detail in the algorithm (1) and figure (3.11).

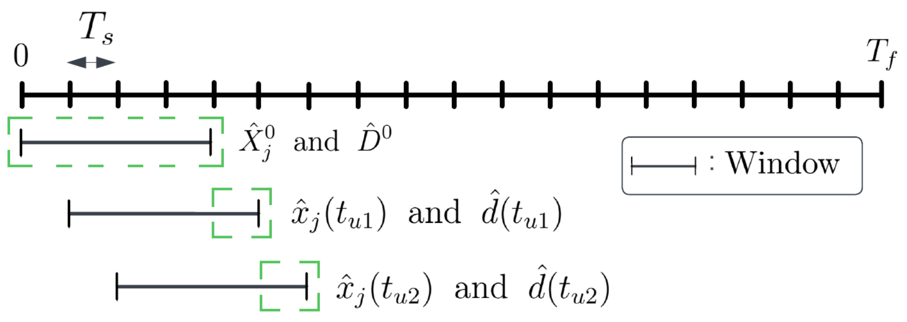

Figure 3.11: Schematic diagram of the algorithm (1)

Algorithm 1: Online estimation of states and unknown input by MFBM

Input: T_f : final time, h : width of the window, T_s : sampling step.
Initialization: $I_{h_1} = [0, h]$, $k = 1$;
for $j = 2$ until n **do :**
- Estimate $\hat{x}_j$ over I_{h_1} by solving the algebraic system (3.58-3.60) for $I = I_{h_1}$; $\hat{X}_j^0 = \hat{x}_j$;

End
- Estimate $\hat{d}$ over I_{h_1} by solving the algebraic system (3.69-3.71) for $I = I_{h_1}$; $\hat{D}^0 = \hat{d}$;

while $h + kT_s \leqslant T$ **do** :
$I_{h_k} = [kT_s, h + kT_s] = [t_{l_k}, t_{u_k}]$;
for $j = 2$ until n **do :**
- Estimate $\hat{x}_j$ over I_{h_k} by solving the algebraic system (3.58-3.60) for $I = I_{h_k}$;
- Update $\hat{X}_j^k = \left[\hat{X}_j^{k-1} \quad \hat{x}_j(t_{u_k})\right]$;

End
- Estimate $\hat{d}$ over I_{h_k} by solving the algebraic system (3.69-3.71) for $I = I_{h_k}$;
- Update $\hat{D}^k = \left[\hat{D}^{k-1} \quad \hat{d}(t_{u_k})\right]$;

k=k+1;
End while
Output: $\hat{D}$ and $\hat{X}_j$ for $j = 2, \ldots, n$.

Remark 3.3.1 :Real-time estimation cannot start before the time $t = h$, which represents a dead time with respect to the control, where we have no information on the system states and the unknown input, so the control inputs cannot be calculated during this time. Depending on the knowledge of the system behavior, it is possible to predefine the control during this dead time or computing it based on the measurements only.

Remark 3.3.2 : This strategy is also valid for MFBM parameter estimation (adaptive control).

3.5.3 Numerical example

Consider the example of an oscillating system governed by the following ordinary differential equation [87] [83]:

$$m\ddot{z} + g(t, z, \dot{z}) + K_s z = F(t), \tag{3.72}$$

with z the mass displacement, $\dot{z}$ the displacement velocity and $\ddot{z}$ the acceleration, $F(t)$ represents the external force applied to the mass, $g(t, z, \dot{z})$ represents the nonlinear dissipative force, K_s is the spring stiffness constant and m represents the body mass.
Let the state vector $x = \begin{bmatrix} z \\ \dot{z} \end{bmatrix}$. The measured output of the system is $y = z$ and the input of the system is the force $u(t) = F(t)$, the system is represented in the state space by

$$\begin{cases} \dot{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ -\frac{K_s}{m}x_1 + \frac{1}{m}u - \frac{1}{m}g(t,x) \end{bmatrix}, \\ y = x_1, \end{cases} \tag{3.73}$$

where $g(t, x)$ is the unknown non-linear part of the model, assumed to be the unknown input to the system. In this example, we take :

$$g(t, x) = \frac{2\delta x_1^2}{m(1 + x_1^2)} x_2(1 + 0.1\sin(\pi t)), \tag{3.74}$$

where δ is the depreciation factor.

The objective is to estimate the state x_2 and the nonlinear part $g(t, x)$ using the MFBM by the offline and online approachs. For each approach, the parameters are given in what follows.
Offline approach :
The type of modulating functions chosen are polynomial modulating functions for both the estimation of the state x_2 and the unknown input $g(t, x)$, respectively given by :

$$\phi_i(t) = \frac{\bar{\phi}_i(t)}{||\bar{\phi}_i(t)||_2}, \quad \text{Avec :} \quad \bar{\phi}_i(t) = t^{(p_x+M+1-i)}(T - t)^{(p_x+i)}, \ i = 1, 2, \ldots, M, \tag{3.75}$$

$$\phi_i(t) = \frac{\bar{\phi}_i(t)}{||\bar{\phi}_i(t)||_2}, \quad \text{Avec :} \quad \bar{\phi}_i(t) = t^{(p_d+W+1-i)}(T - t)^{(p_d+i)}, \ i = 1, 2, \ldots, W, \tag{3.76}$$

where T is the final simulation time, $p_x, p_d \in \mathbb{N}^*$ are degrees of freedom, M and W are the number of modulating functions used for estimating the state and the unknown input respectively. We decompose x_2 and $g(t, x)$ into N and G polynomial basis functions respectively.

The parameters chosen for offline estimation are given in the following table (3.2) :

N	M	G	W	p_x	p_d
35	35	30	30	1	1

Table 3.2: Offline observer parameters

Online approach :
We use the same modulating functions as above, adapting them so that they can be used for the online estimation of x_2 and $g(t,x)$ respectively.

$$\bar{\phi}_i(\tau - t + h) = (\tau - t + h)^{(p_x+M+1-i)}(t-\tau)^{(p_x+i)},\ i = 1, 2, \ldots, M, \tag{3.77}$$

$$\bar{\phi}_i(\tau - t + h) = (\tau - t + h)^{(p_d+W+1-i)}(t-\tau)^{(p_d+i)},\ i = 1, 2, \ldots, M, \tag{3.78}$$

where h is the width of the sliding window.

Remark 3.3.3 : When estimating in an online setting, since we are estimating over small intervals, the approximation of the signals by the basis functions is better and requires a smaller number of basis, so we can reduce the number of basis functions as well as the number of modulating functions.

Remark3.3.4 : The number of samples used to calculate the integrals is much smaller in the case of online estimation than the offline setting, which can cause the A_j and A_d matrices to be ill-conditioned. To remedy this, we need to add a regularization step before inverting the matrices. In this example, we used Thikonov regularization [44].

The parameters chosen for the online estimation are given in the following table (3.3) :

h	N	M	G	W	p_x	p_d
0.5s	4	4	5	5	3	5

Table 3.3: Online observer parameters

Simulations were performed taking $u(t) = 0.5\sin(\frac{2\pi}{5})$ and setting $m = 5, K_s = 1$ and $\delta = 2$. The measurements are assumed to be noisy, so a random normally distributed noise of 5% relative to the real signal without noise was added to the output. Figure (3.12) shows the real oscillator output signal and the noisy measured output signal.

Figure (3.13) shows the results of the offline estimation of the state x_2 and the unknown input $g(t,x)$. It can be seen that the reconstructed velocity $\hat{x}_2$ obtained using the MFBM corresponds perfectly to the actual value of x_2. Regarding the estimation of the disturbance term $\hat{g}(t,x)$, the observer also provides a good estimate, although slightly less accurate, particularly at the limits of the time interval. The relative error of the offline velocity estimate is 0.13%, while the relative error of the unknown input estimate is 1.25%.

The figure (3.14) shows that the observer is able to estimate the velocity x_2 as well as the unknown non-linear term $g(t,x)$ with better accuracy using the sliding integration window. The inaccuracies at the boundaries have disappeared, and the relative error of the estimation of the velocity and the unknown term by the online approach are respectively equal to 0.02% and 0.06%.

The online estimation approach offers more accurate results than the offline approach, as it allows for segmented signal estimation. This segmentation makes it easier to approximate the signals using the basis functions. In contrast, in the offline approach, the entire signal is reconstructed using the basis functions, which requires a larger number of these functions. This can lead to less accurate results, especially when the time interval is long.

By using online setting for estimation, more accurate estimates can be obtained thanks to a better adaptation of the basis functions to each specific segment of the signal.

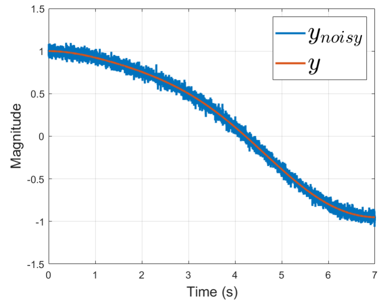

Figure 3.12: System output signal

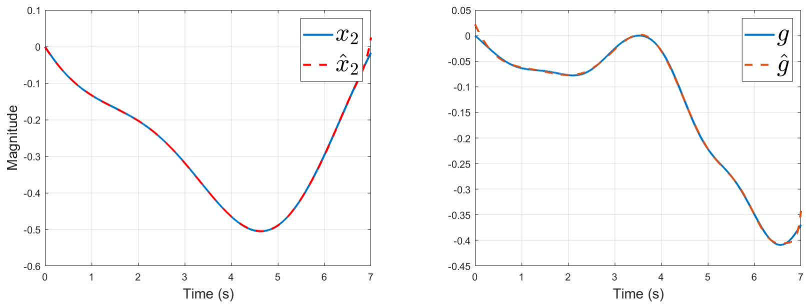

Figure 3.13: Offline estimation of state x_2 and unknown input g.

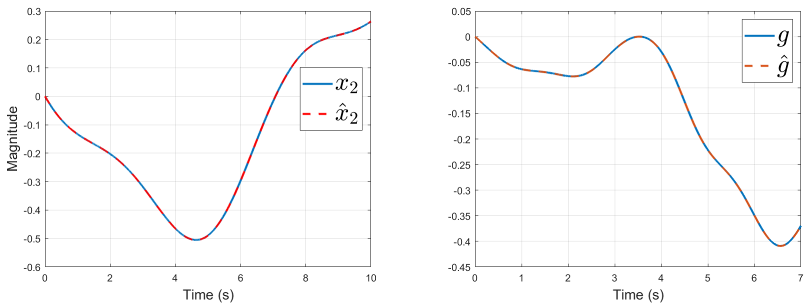

Figure 3.14: Online estimation of state x_2 and unknown input g.

3.6 Contribution - Extension of the modulating function estimation method

3.6.1 Problem formulation

The method presented in section 3.5 can only estimate the states of a nonlinear system in a simple triangular canonical form. However, many other systems have a more complex structure (e.g. the drone model presented in the next chapter). We therefore, propose to extend this method to more general models. In the following, we consider the nonlinear block triangular system, which is composed of r blocks of canonical triangular systems, governed by the following system of differential equations :

$$\begin{cases} \frac{d^{n_1}x_1}{dt^{n_1}} = f_1(x_1, \ldots, x_r, u) + d_1(t, x_1, \ldots, x_r, u), \\ \frac{d^{n_2}x_2}{dt^{n_2}} = f_2(x_1, \ldots, x_r, u) + d_2(t, x_1, \ldots, x_r, u), \\ \vdots \\ \frac{d^{n_r}x_r}{dt^{n_r}} = f_r(x_1, \ldots, x_r, u) + d_r(t, x_1, \ldots, x_r, u), \\ y = (x_1, x_2, \ldots, x_r)^T. \end{cases} \tag{3.79}$$

Represented in the state space as :

$$\begin{cases} \dot{x} = \begin{bmatrix} \dot{x}_{1,1} \\ \vdots \\ \dot{x}_{n_1-1,1} \\ \dot{x}_{n_1,1} \\ \dot{x}_{1,2} \\ \vdots \\ \dot{x}_{n_2-1,2} \\ \dot{x}_{n_2,2} \\ \vdots \\ \vdots \\ x_{1,r} \\ \vdots \\ x_{n_r-1,r} \\ x_{n_r,r} \end{bmatrix} = \begin{bmatrix} x_{2,1} \\ \vdots \\ x_{n_1,1} \\ f_1(x,u) + d_1(t,x,u) \\ x_{2,2} \\ \vdots \\ x_{n_2,2} \\ f_2(x,u) + d_2(t,x,u) \\ \vdots \\ \vdots \\ x_{2,r} \\ \vdots \\ x_{n_r,r} \\ f_r(x,u) + d_r(t,x,u) \end{bmatrix}, \\ y = [x_{1,1}, x_{1,2}, \ldots, x_{1,r}]^T, \end{cases} \tag{3.80}$$

where $x \in \mathbb{R}^{\sum_{i=1}^{r} n_i}$ is the system state vector, $u \in \mathbb{R}^m (m \in \mathbb{N}^*)$ is the bounded input signal, $y \in \mathbb{R}^r$ is the measured output.
The f_j terms are known non-linear functions and are locally Lipschitz, while d_j are unknown inputs and are assumed to be bounded, $\forall j = 1, 2, \ldots, r$. They can represent a modeling error, a time-varying disturbance, or both.

In this section, we present the method we have developed for a more general class of nonlinear systems, namely block-triangular nonlinear systems.

Having a system with the structure described in (3.80), the state estimation $x_{k,j}$, for $k = 2, ..., n_j$, $j = 1, ..., r$ is decoupled from the estimation of the unknown inputs $d_j(t)$. We therefore propose a two-stage algorithm :

- **Step 1 :** Having output measurements $y(t)$, we estimate the states $x_{k,j}$, for $k = 2, ..., n_j - 1$, $j = 1, ..., r$ using the first $(n_j - 1)$ equations of each system block (3.80) for $t \in [0, T]$.
- **Step 2 :** In the last equation of each system block(3.80), we substitute by the states estimations $x_{k,j}$, for $k = 2, ..., n_j - 1$, $j = 1, ..., r$, the inputs $u(t)$ and the outputs $y(t)$ in order to estimate disturbances d_j, $j = 1, ..., r$.

3.6.2 States estimation

The first step in the proposed modulating function-based estimation approach is to estimate the $x_{k,j}$ states, for $k = 2, ..., n_j$, $j = 1, ..., r$, using the $n_j - 1$ first equations of each block of the triangular system, and its corresponding measured output $y_j(t)$. The estimation of the $x_{k,j}$ states is given by the following proposition.

Remark 3.6.1 : In what follows, we'll assume that the number of modulating functions used is equal to the number of basis functions, which means we'll have to solve square algebraic systems.

Proposition 3.6.1 :

Let $y_j = x_{1,j}$, for all $j = 1, .., r$ be the measured outputs of system (3.80) and $\{\phi_i\}_{i=1}^{i=N}$ be a set of modulating functions of order $l \geqslant k$ satisfying (P1) and (P2), $\forall k = 2, ..n_j$, $j = 1, ..., r$. Then an estimate of the state $x_{k,j}$, is given as follows: $\forall k = 2, ..., n_j$, $j = 1, .., r$

$$\hat{x}_{k,j} = \sum_{i=1}^{N} \hat{a}^j_{k,i} \alpha^j_{k,i}(t), \tag{3.81}$$

where $N \in \mathbb{N}$, $\alpha^j_{k,i}(t)$ are chosen basis functions and $\hat{a}^j_{k,i}$ represent their corresponding unknown coefficients and are given by the following closed-form solution :

$$\begin{bmatrix} \hat{a}^j_{k,1} \\ \vdots \\ \hat{a}^j_{k,N} \end{bmatrix} = (-1)^{(k-1)} \Phi_k \begin{bmatrix} \langle \phi_1^{(k-1)}, x_{1,j} \rangle_I \\ \vdots \\ \langle \phi_N^{(k-1)}, x_{1,j} \rangle_I \end{bmatrix}, \tag{3.82}$$

where Φ_k is the $N \times N$ squared matrix defined by :

$$\Phi_k = \begin{bmatrix} \langle \phi_1, \alpha^j_{k,1}(t) \rangle_I & \dots & \langle \phi_1, \alpha^j_{k,N}(t) \rangle_I \\ \vdots & \ddots & \vdots \\ \langle \phi_N, \alpha^j_{k,1}(t) \rangle_I & \dots & \langle \phi_N, \alpha^j_{k,N}(t) \rangle_I \end{bmatrix}. \tag{3.83}$$

Proof :

considering the first $n_j - 1$ equations of each block of triangular system j, for $j = 1, ..., r$ of system (3.80)

$$\begin{bmatrix} \dot{x}_{1,j} \\ \vdots \\ \dot{x}_{n_j-1,j} \end{bmatrix} = \begin{bmatrix} x_{2,j} \\ \vdots \\ x_{n_j,j} \end{bmatrix}. \tag{3.84}$$

Given that each state $x_{k,j}$, $\forall k = 2, ..., n_j, j = 1, ..., r$ is the k^{th} derivative of $x_{1,j}$ ($i \in 2, \dots, n-1$), equation (3.84) can be re-written as

$$\begin{bmatrix} x_{2,j} \\ \vdots \\ x_{n_j,j} \end{bmatrix} = \begin{bmatrix} \dot{x}_{1,j} \\ \vdots \\ x^{(n_j-1)}_{1,j} \end{bmatrix}. \tag{3.85}$$

Applying the operator $\langle \phi, . \rangle$ on both sides

$$\begin{bmatrix} \langle \phi, x_{2,j} \rangle_I \\ \vdots \\ \langle \phi, x_{n_j,j} \rangle_I \end{bmatrix} = \begin{bmatrix} \langle \phi, \dot{x}_{1,j} \rangle_I \\ \vdots \\ \langle \phi, x^{(1)}_{n_j-1,j} \rangle_I \end{bmatrix}. \tag{3.86}$$

Using Property 3.2.1, one obtains

$$\begin{bmatrix} \langle \phi, x_{2,j} \rangle_I \\ \vdots \\ \langle \phi, x_{n_j,j} \rangle_I \end{bmatrix} = \begin{bmatrix} -\langle \dot{\phi}, x_{1,j} \rangle_I \\ \vdots \\ (-1)^{(n_j-1)} \langle \phi^{(n_j-1)}, x_{1,j} \rangle_I \end{bmatrix}. \tag{3.87}$$

The states are time-varying, so they can be decomposed in space spanned by a set of unknown coefficients $a^j_{k,i}$ and N chosen basis functions $\alpha^j_{k,i}(t)$ as follows :

$$\hat{x}_{k,j} = \sum_{i=1}^{N} \hat{a}^j_{k,i} \alpha^j_{k,i}(t). \tag{3.88}$$

replacing the state $\hat{x}_{k,j}$ by its decomposition, (3.87) becomes :

$$\sum_{i=1}^{N} \hat{a}^j_{k,i} \langle \phi, \alpha^j_{k,i}(t) \rangle_I = -\langle \phi^{(k-1)}, x_{1,j} \rangle_I. \tag{3.89}$$

which can be written in vector notation :

$$\begin{bmatrix} \langle \phi, \alpha^j_{k,1}(t) \rangle_I & \dots & \langle \phi, \alpha^j_{k,N}(t) \rangle_I \end{bmatrix} \begin{bmatrix} \hat{a}^j_{k,1} \\ \vdots \\ \hat{a}^j_{k,N} \end{bmatrix} \tag{3.90}$$

$$= (-1)^{(k-1)} \langle \phi^{(k-1)}, x_{1,j} \rangle_I.$$

This is an algebraic system with N unknown coefficients $\hat{a}^j_{k,i}$. In order to solve it, we need N different modulating functions which leads to N different equations. The final equation is :

$$\begin{bmatrix} \hat{a}^j_{k,1} \\ \vdots \\ \hat{a}^j_{k,N} \end{bmatrix} = (-1)^{(k-1)} \Phi_k \begin{bmatrix} \langle \phi_1^{(k-1)}, x_{1,j} \rangle_I \\ \vdots \\ \langle \phi_N^{(k-1)}, x_{1,j} \rangle_I \end{bmatrix}, \tag{3.91}$$

where Φ_k is the $N \times N$ square matrix defined by :

$$\Phi_k = \begin{bmatrix} \langle \phi_1, \alpha^j_{k,1}(t) \rangle_I & \dots & \langle \phi_1, \alpha^j_{k,N}(t) \rangle_I \\ \vdots & \ddots & \vdots \\ \langle \phi_N, \alpha^j_{k,1}(t) \rangle_I & \dots & \langle \phi_N, \alpha^j_{k,N}(t) \rangle_I \end{bmatrix}. \tag{3.92}$$

3.6.3 Unknown input estimation

Given the states estimates $\hat{x}_{k,j}$, $\forall k = 2, ..., n_j$, $j = 1, ..., r$ and the measured input $u(t)$ and output $y_j(t) = x_{1,j}(t)$, we can proceed with estimating the unknown input $d_j(t)$. The estimation of the unknown inputs of system (3.80) is presented in the following proposition.

Proposition 3.6.2 :

Let $u(t)$ and $y_j(t) = x_{1,j}(t)$, $\forall j = 1,..,r$ be the measured input and outputs of system (3.80), respectively, and $\hat{x}_{k,j}$ the estimated states for $k = 2, ..., n_j$, $j = 1, ..., r$. Let $\{\phi_i\}_{i=1}^{i=G}$ be a set of modulating functions of order $l \geqslant n_j$ satisfying (P1) and (P2) $\forall j = 1, ..., r$. Then an estimate of the unknown input $d_j(t)$, $\forall j = 1,..,r$ is given by

$$\hat{d}_j(t, x, u) = \sum_{i=1}^{G} \hat{b}_i^j \beta_i^j(t), \tag{3.93}$$

where $G \in \mathbb{N}$, $\beta_i^j(t)$ are the known basis functions and $\hat{b}_i^j$ are unknown coefficients given by the following closed-form solution :

Solution 1 :

$$\begin{bmatrix} \hat{b}_1^j \\ \vdots \\ \hat{b}_G^j \end{bmatrix} = -\Phi_{ui} \begin{bmatrix} \langle \dot{\phi}_1, \hat{x}_{n_j,j} \rangle_I + \langle \phi_1, f_j(\hat{x}, u) \rangle_I \\ \vdots \\ \langle \dot{\phi}_G, \hat{x}_{n_j,j} \rangle_I + \langle \phi_G, f_j(\hat{x}, u) \rangle_I \end{bmatrix}, \tag{3.94}$$

Solution 2 :

$$\begin{bmatrix} \hat{b}_1^j \\ \vdots \\ \hat{b}_G^j \end{bmatrix} = (-1)^{(n_j)} \times \Phi_{ui} \begin{bmatrix} \langle \phi_1^{(n_j)}, x_{1,j} \rangle_I + \langle \phi_1, f_j(\hat{x}, u) \rangle_I \\ \vdots \\ \langle \phi_G^{(n_j)}, x_{1,j} \rangle_I + \langle \phi_G, f_j(\hat{x}, u) \rangle_I \end{bmatrix}, \tag{3.95}$$

where Φ_{ui} of both solutions is the $G \times G$ square matrix defined by

$$\Phi_{ui} = \begin{bmatrix} \langle \phi_1, \beta_1^j(t) \rangle_I & \dots & \langle \phi_1, \beta_G^j(t) \rangle_I \\ \vdots & \ddots & \vdots \\ \langle \phi_G, \beta_1^j(t) \rangle_I & \dots & \langle \phi_G, \beta_G^j(t) \rangle_I \end{bmatrix}. \tag{3.96}$$

Proof :

Considering the $(n_j - th)$ equation of each block j of triangular systems, for $j = 1, ..., r$:

$$\dot{x}_{n_j,j} = f_j(x, u) + d_j(t, x, u), \tag{3.97}$$

which is equivalent to :

$$d_j(t, x, u) = \dot{x}_{n_j,j} - f_j(x, u). \tag{3.98}$$

Applying the operator $\langle \phi, . \rangle$ on both sides :

$$\langle \phi, d_j(t, x, u) \rangle = \langle \phi, \dot{x}_{n_j,j} \rangle_I - \langle \phi, f_j(x, u) \rangle_I. \tag{3.99}$$

At this point we have two choices ; we can either replace $x_{n_j,j}$ by the output derivative $x_{1,j}^{(n_j-1)}$ or use its estimate $\hat{x}_{n_j,j}$. The second case is better when the measurements are noisy because the numerical computation of derivatives of noisy signals generates numerical unstability. And the signal $\hat{x}_{n_j,j}$ obtained by the modulating function estimator in Proposition 3.6.2 is less corrupted than $x_{1,j}$ due to the integration. Therefore, the noise will be filtered twice with the next integration and be more robust than if we used $x_{1,j}^{(n_j-1)}$.

choice 1 :
Using Property 3.2.1 and substituting x by $\hat{x}$ and $x_{n_j,j}$ by $\hat{x}_{n_j,j}$, equation 3.99 becomes :

$$\langle \phi, d_j(t,x,u) \rangle = -\langle \dot{\phi}, \hat{x}_{n_j,j} \rangle_I - \langle \phi, f_j(\hat{x},u) \rangle_I. \tag{3.100}$$

The unknown input $d_j(t)$ is time-varying, so it can be decomposed in the space spanned by a set of unknown coefficients $\hat{b}_i^j$ and G chosen basis functions $\beta_i^j(t)$ as follows :

$$\hat{d}_j(t,x,u) = \sum_{i=1}^{G} \hat{b}_i^j \beta_i^j(t). \tag{3.101}$$

Substituting $\hat{d}_j(t)$ by its decomposition in (3.107) :

$$\sum_{i=1}^{G} \hat{b}_i^j \langle \phi, \beta_i^j(t) \rangle_I = -\langle \dot{\phi}, \hat{x}_{n_j,j} \rangle_I - \langle \phi, f_j(\hat{x},u) \rangle_I, \tag{3.102}$$

Which can be written in vector notation :

$$\begin{bmatrix} \langle \phi, \beta_1^j(t) \rangle_I & \dots & \langle \phi, \beta_G^j(t) \rangle_I \end{bmatrix} \begin{bmatrix} \hat{b}_1^j \\ \vdots \\ \hat{b}_G^j \end{bmatrix} \tag{3.103}$$

$$= -\langle \dot{\phi}, \hat{x}_{n_j,j} \rangle_I - \langle \phi, f_j(\hat{x},u) \rangle_I.$$

Equation (3.109) is an algebraic system with G unknown coefficient $\hat{b}_i^j$. In order to solve for $\hat{b}_i^j$, $j = 1, ..., n$, we need G different modulating functions in order to obtain G different equations. Finally the coefficients $\hat{b}_i^j$ are obtained as follows

$$\begin{bmatrix} \hat{b}_1^j \\ \vdots \\ \hat{b}_G^j \end{bmatrix} = -\Phi_{ui} \begin{bmatrix} \langle \dot{\phi}_1, \hat{x}_{n_j,j} \rangle_I + \langle \phi_1, f_j(\hat{x},u) \rangle_I \\ \vdots \\ \langle \dot{\phi}_G, \hat{x}_{n_j,j} \rangle_I + \langle \phi_G, f_j(\hat{x},u) \rangle_I \end{bmatrix}, \tag{3.104}$$

where Φ_{ui} is the $G \times G$ square matrix given by :

$$\Phi_{ui} = \begin{bmatrix} \langle \phi_1, \beta_1^j(t) \rangle_I & \dots & \langle \phi_1, \beta_G^j(t) \rangle_I \\ \vdots & \ddots & \vdots \\ \langle \phi_G, \beta_1^j(t) \rangle_I & \dots & \langle \phi_G, \beta_G^j(t) \rangle_I \end{bmatrix}. \tag{3.105}$$

choice 2 :
Using Property 3.2.1 and substituting x by $\hat{x}$ and $x_{n_j,j}$ by $x_{1,j}^{(n_j-1)}$, equation 3.99 becomes

$$\langle \phi, d_j(t,x,u) \rangle = \langle \phi, x_{1,j}^{(n_j)} \rangle_I - \langle \phi, f_j(\hat{x},u) \rangle_I. \tag{3.106}$$

The unknown input $d_j(t)$ is time-varying, so it can be decomposed in the space spanned by a set of unknown coefficients $\hat{b}_i^j$ and G chosen basis functions $\beta_i^j(t)$ as follows :

$$\hat{d}_j(t,x,u) = \sum_{i=1}^{G} \hat{b}_i^j \beta_i^j(t). \tag{3.107}$$

Substituting $\hat{d}_j(t)$ by its decomposition in (3.107) :

$$\sum_{i=1}^{G} \hat{b}_i^j \langle \phi, \beta_i^j(t) \rangle_I = (-1)^{(n_j)} \langle \phi^{(n_j)}, x_{1,j} \rangle_I - \langle \phi, f_j(\hat{x}, u) \rangle_I, \tag{3.108}$$

Which can be written in vector notation :

$$\begin{bmatrix} \langle \phi, \beta_1^j(t) \rangle_I & \dots & \langle \phi, \beta_G^j(t) \rangle_I \end{bmatrix} \begin{bmatrix} \hat{b}_1^j \\ \vdots \\ \hat{b}_G^j \end{bmatrix} \tag{3.109}$$

$$= (-1)^{(n_j)} \langle \phi^{(n_j)}, x_{1,j} \rangle_I - \langle \phi, f_j(\hat{x}, u) \rangle_I.$$

Equation (3.109) is an algebraic system with G unknown coefficient $\hat{b}_i^j$. In order to solve for $\hat{b}_i^j$, $j = 1, ..., n$, we need G different modulating functions in order to obtain G different equations. Finally the coefficients $\hat{b}_i^j$ are obtained as follows :

$$\begin{bmatrix} \hat{b}_1^j \\ \vdots \\ \hat{b}_G^j \end{bmatrix} = (-1)^{(n_j)} \times \Phi_{ui} \begin{bmatrix} \langle \phi_1^{(n_j)}, x_{1,j} \rangle_I + \langle \phi_1, f_j(\hat{x}, u) \rangle_I \\ \vdots \\ \langle \phi_G^{(n_j)}, x_{1,j} \rangle_I + \langle \phi_G, f_j(\hat{x}, u) \rangle_I \end{bmatrix}, \tag{3.110}$$

where Φ_{ui} is the $G \times G$ square matrix given by :

$$\Phi_{ui} = \begin{bmatrix} \langle \phi_1, \beta_1^j(t) \rangle_I & \dots & \langle \phi_1, \beta_G^j(t) \rangle_I \\ \vdots & \ddots & \vdots \\ \langle \phi_G, \beta_1^j(t) \rangle_I & \dots & \langle \phi_G, \beta_G^j(t) \rangle_I \end{bmatrix}. \tag{3.111}$$

3.7 Conclusion

The aim of this chapter was to provide a detailed understanding of the modulating function method and its principles. This method offers a promising approach to estimating the parameters, states and unknown inputs of finite-time dynamical systems, by transforming the estimation problem into a simple algebraic linear system. It enables both constant and time-varying variables to be estimated, using two distinct approaches: the offline approach, which is useful for system parameter identification, and the online approach, which is useful for state estimation in control problems, estimation of unknown inputs in fault detection problems, or even estimation of system parameters in adaptive control problems.

One of the main advantages of this method over other estimation approaches is that it does not require knowledge of the system's initial conditions, thanks to the properties of the modulating functions at the boundaries. In addition, unlike many other finite-time estimation approaches, it does not use any non-smooth or infinity-tending terms. On the other hand, thanks to the modulating operator (integration), the method is robust to external disturbances that can affect measurements.

Finally, we have presented our contribution to observers with unknown inputs via the MFBM, which consists of an extension adapted to a new class of nonlinear systems. In order to illustrate and highlight our proposal, we will compare the observer we have developed with other estimation approaches widely used and studied in the literature in the context of a problem of controlling a drone in a particular environment, which was dealt with for the first time very recently. We'll look at this in the next few chapters.

Chapter 4

Prescribed time observers

4.1 Introduction

In the previous chapter, we focused mainly on finite-time observers based on modulating functions. These observers were algebraic rather than dynamic in nature. In this chapter, we will focus on a recently developed type of observer, which is dynamical and ensures convergence in finite time, to be more specific, convergence in a prescribed time T determined by the user independently of initial conditions and system parameters. This type of observer uses time-varying gains that tend to infinity as time approaches the prescribed convergence time, using an approach based on specific coordinate transformation.

We'll start by providing the necessary preliminaries to fully understand what comes next, then explain the fundamental principle on which this approach is based. Next, we will present the work recently developed in [33] and [5], which respectively propose fixed-time observers with a prescribed time T for linear and nonlinear systems of particular form. The performance of these observers will be illustrated using simple examples. This chapter aims to introduce and provide a general understanding of the fundamental principles on which this type of observer is built.

It is important to note that our original intention was to contribute to this subject by extending this approach to a more general form of systems. However, due to time constraints, this has not been possible. Therefore, for the purposes of this work, we will confine ourselves to presenting recent work and advances concerning this type of observers.

4.2 Preliminaries

In this section, we will introduce some basic concepts on which prescribed-time observers are founded. We will also recall some definitions of fixed-time and prescribed-time stability, slightly modified to align ourselves with the work of Holloway and Krstic, before presenting a formal one.

Suppose that a dynamical system of order n is stable and converges to the origin in a finite time $T(x_0) > 0$, which depends on the initial conditions $x_0 = x(t_0)$, if for all $x_0 \in \mathbb{R}^n$, there exists a time $T > 0$ such that $T(x_0) \leqslant T$, then the system is said to converge in fixed time in T. When this fixed time T can be chosen arbitrarily by the user as a design parameter, in contrast to *Definition 1.3.5* where the prescribed time T represents the exact time at which the system reaches the origin. The definition used by Holloway and Krstic is less restrictive, allowing convergence or stability over a time interval T rather than at a precise instant [33].

4.2.1 Fixed-time scaling functions

Let $\mu_1(t - t_0, T) : [t_0, t_0 + T) \to \mathbb{R}_{>0}$ be a strictly increasing function from 1 when $t = t_0$ to infinity when t tends to $t_0 + T$ given by [33]

$$\mu_1(t - t_0, T) := \frac{T}{T + t_0 - t}, \tag{4.1}$$

with $T > 0$ a parameter to be fixed. we also define the strictly decreasing function from 1 when $t = t_0$ to zero when t tends to $t_0 + T$ given by [33] :

$$\nu(t-t_0,T) := \mu_1^{-1}(t-t_0,T) := \frac{T+t_0-t}{T}. \tag{4.2}$$

For a system of order n, the scaling function and its inverse are defined by [33] :

$$\mu(t-t_0,T) := \mu_1(t-t_0,T)^{n+m} = \frac{T^{n+m}}{(T+t_0-t)^{n+m}}, \tag{4.3}$$

$$\nu(t-t_0,T) := \nu_1(t-t_0,T)^{n+m} = \frac{(T+t_0-t)^{n+m}}{T^{n+m}}, \tag{4.4}$$

or $m \in \mathbb{N}_{>0}$ is a degree of freedom set by the user and has an influence on the speed of divergence to infinity and zero of μ and ν respectively.

Figure (4.1) show the scaling functions μ and ν for different values of the degree of freedom m by setting $n = 1$, $T = 5$ and $t_0 = 0$.

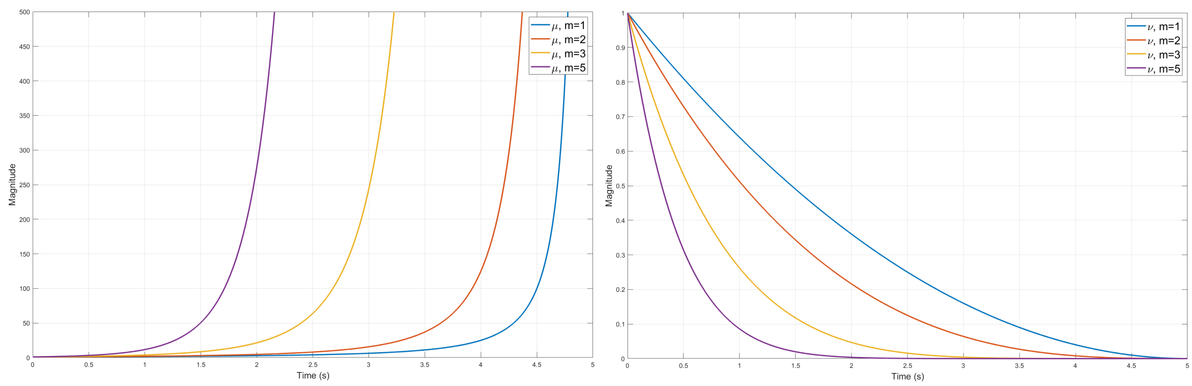

Figure 4.1: Scaling functions $\mu(t_0 - t, T)$ and $\nu(t_0 - t, T)$ with $T = 5$, $n = 1$ and different values of m

Now that we've introduced the various basic concepts, we present the formal definition proposed in [33] on which their work on prescribed-time observers is based, It's given as follows

Definition 6.2.1 : [33] A dynamical system $\dot{x} = f(x,t)$ is said to be globally uniformly asymptotically fixed-time stable (GUAFxS) in a time T, if there exists a function β of class $\mathcal{KL}$ such that for all $t \in [t_0, t_0 + T)$.

$$|x(t)| \leqslant \beta(|x_0|, \mu_1(t-t_0,T) - 1), \tag{4.5}$$

with $\mu_1(t-t_0,T)$ defined by (4.1). $\mu_1(t-t_0,T) - 1 = \frac{t-t_0}{T+t_0-t}$ is a function which at t_0 is equal to 0 and grows strictly to infinity when $t \to t_0 + T$, since β is a function of class $\mathcal{KL}$ so β tends to zero when t tends to $t_0 + T$.

Remark 6.2.2 : Since T is a constant independent of initial conditions and is freely chosen by the user, a system verifying (4.5) can be considered to have convergence properties in prescribed time T.

4.3 Prescribed-time observers for linear systems

Halloway and Krstic developed a prescribed-time state observer with time-varying gains that tend to infinity when time tends towards the prescribed convergence time, and demonstrated that this observer is GUAFxS according to *Definition 6.2.1* in a prescribed time defined apriori by the user, independently of initial conditions and system parameters. Their work focused on SISO linear systems written in the following form :

$$\begin{cases} \dot{x}(t) = Ax(t) - ay(y) + \begin{bmatrix} 0_{(r-1)\times 1} \\ b \end{bmatrix} u(t), \\ y(t) = Cx(t), \end{cases} \tag{4.6}$$

where

$$A = \begin{bmatrix} 0 & I_{n-1} \\ \vdots & \\ 0 & 0 \end{bmatrix}, \quad a = \begin{bmatrix} a_{n-1} \\ \vdots \\ a_0 \end{bmatrix}, \quad b = \begin{bmatrix} b_p \\ \vdots \\ b_0 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0_{1\times(n-1)} \end{bmatrix}, \tag{4.7}$$

where $x(t) \in \mathbb{R}^n$ is the system state vector, $u(t) \in \mathbb{R}$ is the bounded and known control input, $y(t) \in \mathbb{R}$ the measured system output, r is the relative degree of the system such that $r = n - \rho$ or $n > \rho \geqslant 0$ and the coefficients a_i, $i = 0, \ldots, n-1$ and b_j, $j = 0, \ldots, \rho$ are known.

The observer allowing estimation of the system states (4.6) in a prescribed time T is given by [33] :

$$\dot{\hat{x}}(t) = A\hat{x}(t) - ay(t) + \begin{bmatrix} 0_{(r-1)\times 1} \\ b \end{bmatrix} u(t) + \begin{bmatrix} g_1(t-t_0,T) \\ \vdots \\ g_n(t-t_0,T) \end{bmatrix} (y(t) - \hat{x}_1(t)), \tag{4.8}$$

where $g_i(t-t_0,T), i = 1, \ldots, n$, are time-varying gains that depend on the prescribed convergence time T, in the following we present the principle and methodology for designing such observers.

4.3.1 Principle

The observer's estimation errors are defined as $e_i(t) = x_i(t) - \hat{x}_i(t)$, $i = 1, \ldots, n$, and its dynamics is given by

$$\begin{cases} \dot{e}_i(t) = e_{i+1}(t) - g_i(t-t_0,T)e_1(t), \quad i = 1, \ldots, n-1, \\ \dot{e}_n(t) = -g_n(t-t_0,T)e_1(t). \end{cases} \tag{4.9}$$

Consider the following base change :

$$\zeta_i(t) := \mu(t-t_0,T)e_i(t), \quad i = 1, \ldots, n-1, \tag{4.10}$$

or $T > 0$, $t \in [t_0, t_0 + T)$ and $\mu : [t_0, t_0 + T) \to \mathbb{R}_{>0}$ is a strictly increasing function having the property of tending to infinity when t tends to $t_0 + T$ defined by (4.3). The inverse transformation is given by :

$$e_i(t) = \mu(t-t_0,T)^{-1}\zeta(t), \quad i = 1, \ldots, n-1, \tag{4.11}$$

according to the properties of $\mu(t-t_0,T)$ mentioned above, $\mu(t-t_0,T)^{-1}$ is a strictly decreasing function which tends towards 0 when $t \to t_0+T$ defined by (4.4).

If the states $\zeta_i(t)$, $i=1,\ldots,n$ are stable and remain a finite quantity on $[t_0,t_0+T)$ and according to the equation (4.11) and the properties of $\mu^{(-1)}(t-t_0,T)$, the errors $e_i(t)$, $i=1,\ldots,n$ will be forced to tend towards zero as t tends towards t_0+T. This has the advantage of providing a means of achieving fixed-time convergence in a prescribed time T. This is the basic principle on which prescribed-time observers are built.

Consequently, when expressing the errors $e_i(t)$ in the new coordinate base, it will suffice to choose the gains $g_i(t-t_0,T), i=1,\ldots,n$ so as to stabilize the errors expressed in the new base $\zeta_i(t)$, forcing the errors $e_i(t)$ to tend towards zero when $t \to t_0+T$. So, using a particular time-dependent change of basis and choosing the gains $g_i(t-t_0,T)$ in a well-determined way, the system (4.9) will have the properties of being fixed time stable according to *Definition 6.2.1* in a time prescribed by the user T independently of initial conditions and system parameters.

Remark 6.2.2 : The change of variable (4.10) can lead to a more complex expression. In what follows, we present the change of variable proposed in [33], which allows us to transform the system of the estimation errors(4.9) while retaining the interesting property of (4.11) and ensuring that the transformed system is easy to stabilize, allowing fixed-time stabilization in a prescribed time T of the system (4.10).

4.3.2 Prescribed-time coordinate transformation

The change of basis proposed in [33] is actually two successive coordinate transformations, the first transformation $e_i(t) \to \zeta_i(t)$ is used to express the dynamics of the error so that fixed-time stabilization can be achieved in a prescribed time according to the approach described previously, it is given by :

$$\zeta_i(t) := \mu(t-t_0,T)e_i(t), \quad i=1,\ldots,n-1, \tag{4.12}$$

followed by the transformation $\zeta_i(t) \to z_i(t)$ which serves to rewrite the errors transformed into a form that is easy to stabilize, it is defined as follows :

$$z_i(t) = \sum_{j=1}^{n} p^*_{i,j}(\mu_1)\zeta_j(t), \quad i=1,\ldots,n, \tag{4.13}$$

Where the functions $p^*_{i,j}(\mu_1)$ are defined by :

$$p^*_{i,j}(\mu_1) := \bar{p}_{i,j}\mu_1^{i-j}, \quad 1 \leqslant j \leqslant i \leqslant n, \tag{4.14}$$

with $\bar{p}_{i,j}$ being constant coefficients determined by the following set of relations :

$$\bar{p}_{i,i} = 1, \tag{4.15}$$

$$\bar{p}_{i,j} = 0, \quad j > i, \tag{4.16}$$

$$\bar{p}_{n,j-1} = -\frac{2n+m-j}{T}\bar{p}_{n,j}, \quad j=n,n-1,\ldots,2, \tag{4.17}$$

$$\bar{p}_{n,j-1} = -\frac{n+m+i-j}{T}\bar{p}_{i,j} + \bar{p}_{i+1,j}, \quad n-1 \geqslant i \geqslant j \geqslant 2, \tag{4.18}$$

Furthermore, if we take the gains $g_i(t-t_0,T)$ as follows :

$$\begin{cases} g_i = l_i + (\frac{n+m+i-1}{T}\bar{p}_{i,1} - \bar{p}_{i+1,1})\mu_1^i - \sum_{j=1}^{i-1} g_j \bar{p}_{i,j}\mu_1^{i-j}, \quad i = 1,\dots,n-1, \\ g_n = l_n + \frac{2n+m-1}{T}\bar{p}_{n,1}\mu_1^n - \sum_{j=1}^{n-1} g_j \bar{p}_{n,j}\mu_1^{n-j}, \end{cases} \tag{4.19}$$

with $l_i, i = 1,\dots,n$ are constants to be selected. So, considering the transformation expressed in (**??**) and replacing by (4.19), the system (4.9) becomes :

$$\begin{cases} \dot{z}_i(t) = z_{i+1}(t) - l_i z_1(t), \quad i = 1,\dots,n-1, \\ \dot{z}_n(t) = -l_n z_1(t), \end{cases} \tag{4.20}$$

4.3.3 Stabilization

The system (4.20) is easily stabilized, by setting the constant parameters $l_i, i = 1,\dots,n$ so that the polynomial $s^n + l_1 s^{n-1} + \ldots + l_n$ has roots with a negative real part, making the system (4.20) AS, according to the transformation property $e(t) \to z(t)$ the system (4.9) will be in a prescribed time T as defined by *Definition 6.2.1.* Further details and proofs of this method can be found in [33].

Remark 6.3.1 : It has been proved in [33] that although the gains $g_i(t-t_0,T)$ tend to infinity when $t \to t_0 + T$, the correction term $\beta_i(t-t_0,T) = g_i e_1$ remains finite.

4.3.4 Practical aspects

The observer (4.8) is easy to implement. Its construction simply consists in selecting the constants $l_i, i = 1,\dots,n$, to stabilize the system (4.20), then independently choosing the prescribed convergence time T and the design parameter $m \geqslant 1$.

Although the correction terms β_i remain finite, in practice the implementation of such an observer will present numerical limitations when time tends to t_0+T, as the gains g_i tend to infinity. To get around this limitation, two approaches are possible [33]:

- Extend the prescribed convergence time T to a time posterior to t_f
- Monitor the gains g_i in real time during estimation until one of them approaches a threshold g_{max} set by the user, at which point we deactivate the correction terms β_i. Once the injections have been deactivated, the estimation errors will increase again. If necessary, the observer can be reset from this point onwards, considering the current time as a new t_0 and the current estimation error as a new initial condition.

In both cases, we obtain convergence of the estimation error within a neighborhood of the origin, which can be adjusted by the user if necessary.

The performances of this observer are illustrated in the following example.

4.3.5 Illustrative example

Consider the following second-order system :

$$\begin{cases} \dot{x}_1(t) = x_2(t) - a_1 y(t), \\ \dot{x}_2(t) = -a_0 y(t), \\ y(t) = x_1(t), \end{cases} \tag{4.21}$$

where $a_1 = 3$ and $a_0 = 2$. The prescribed-time observer for the system (4.21) is given by :

$$\begin{cases} \dot{\hat{x}}_1(t) = \hat{x}_2(t) - a_1 y(t) + g_1(t - t_0, T)(y(t) - \hat{x}_1(t)), \\ \dot{\hat{x}}_2(t) = -a_0 y(t) + g_2(t - t_0, T)(y(t) - \hat{x}_1(t)). \end{cases} \tag{4.22}$$

The constants $\bar{p}_{i,j}$, are determined according to the directives (4.15)-(4.18), we find :

$$\begin{cases} \bar{p}_{1,1} = \bar{p}_{2,2} = 1, \\ \bar{p}_{1,2} = 0, \\ \bar{p}_{2,1} = -\frac{m+2}{T}. \end{cases} \tag{4.23}$$

Having determined the coefficients $\bar{p}_{i,j}$, we can calculate the time-varying gains g_1 and g_2 following (4.19), we obtain :

$$\begin{cases} g_1(t - t_0, T) = l_1 + 2\frac{m+2}{T}\mu_1, \\ g_2(t - t_0, T) = l_2 + l_1\frac{m+2}{T}\mu_1 + \frac{(m+1)(m+2)}{T^2}\mu_1^2. \end{cases} \tag{4.24}$$

To test the designed observer, we run a simulation with the observer parameters set as $l_1 = 3$, $l_2 = 2$ and $T = 3$ and the system initial conditions set as $x_0 = [2, 1]$. First, we simulate the observer in a deterministic environment without measurement noise.

Deterministic environment :

Figure (4.2) represent state estimates and their errors for three different initial conditions, for $m = 1$, we note that the estimates do indeed converge to the real values at the prescribed instant $T = 3$, independently of the observer's initial conditions.

The figure (4.3) illustrates the time evolution of the correction terms for different initial conditions and for $m = 1$. As Halloway and Krstic proved in [33], the correction terms remain finite and tend towards zero when $t \to t_0 + T$.

Figure (4.4) illustrate state estimates, in this case by varying the parameter m for $x_0 = [-2, -3]^T$, we note that changing m impacts the transient regime of the estimation error in such a way that, by increasing its value, the error tends to approach zero more quickly, but the overshoot is larger. However, the estimates converge to their true values in a prescribed time T independently of m.

Stochastic environment :
In a second step, we consider the presence of noise affecting the measurements of the system output, such as :

$$y(t) = x_1(t) + v(t),$$

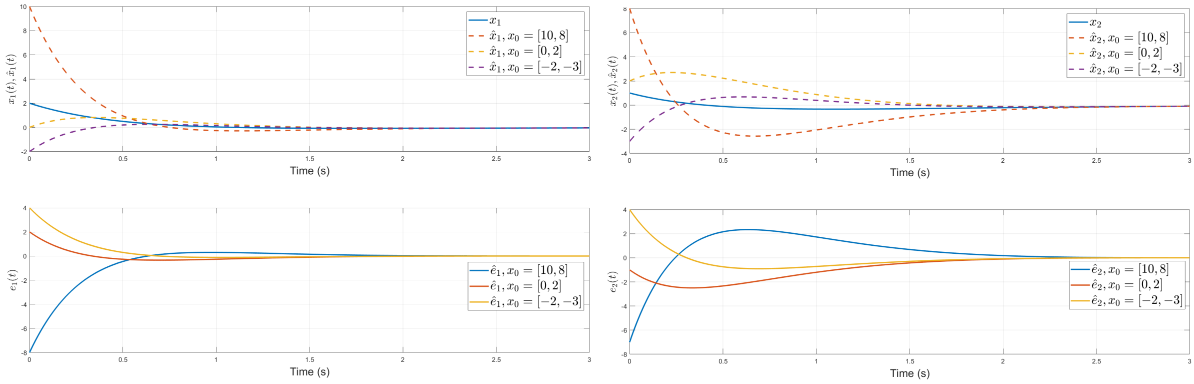

Figure 4.2: State estimation and its error with $T = 3$, $m = 1$ and for different initial observer conditions

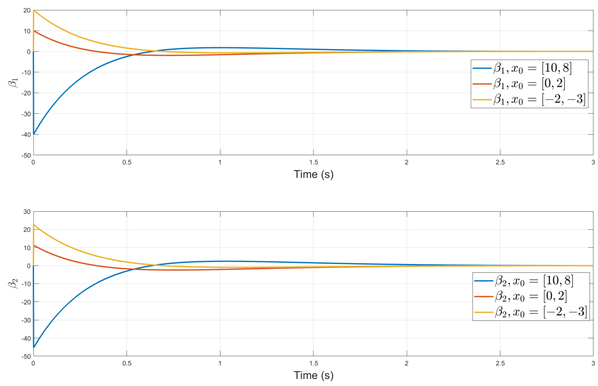

Figure 4.3: Correction terms β_1 and β_2 with $T = 3$, $m = 1$ and for different initial observer conditions

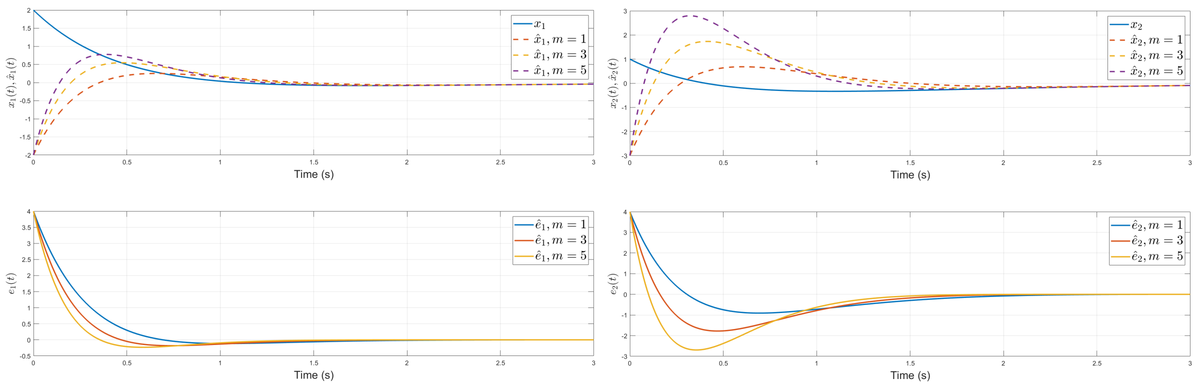

Figure 4.4: State estimation and its error with $T = 3$, $\hat{x} = [-2, -3]^T$ and for different values of m.

where $v(t)$ is a normally distributed random noise estimated at 10% from the real signal x_1. Figure (4.5) represent the evolution of the estimation error of x_1 and x_2 for $x_0 = [-2, -3]^T$ and $m = 1$. We can see that even in the presence of measurement noise, the estimates continue to converge with respect to their initial condition. However, as we start nearing the prescribed time T, the estimates diverge to infinity due to numerical instability, which is accentuated by the measurement noise.

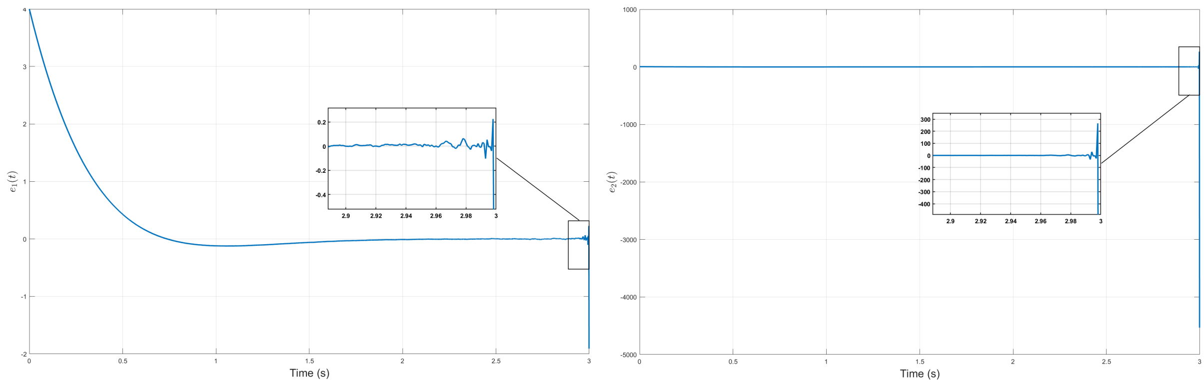

Figure 4.5: State estimation error with $T = 3$, $\hat{x} = [-2, -3]^T$ and $m = 1$ for noisy measurements

4.4 Prescribed-time observers for nonlinear systems

In this section, we will present a recent proposal by the authors of [5], expanding on the work of [33] and [17], which introduces a new extension of prescribed-time observers for nonlinear systems described by

$$\begin{cases} \dot{x}(t) = Ax(t) + f(x(t)), \\ y(t) = Cx(t), \end{cases} \tag{4.25}$$

where

$$A = \begin{bmatrix} 0 & I_{n-1} \\ \vdots & \\ 0 & 0 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0_{1\times(n-1)} \end{bmatrix}. \tag{4.26}$$

with $x(t) \in \mathbb{R}^n$ the system's state vector and $y(t) \in \mathbb{R}$ the system's measured output.

Hypothesis : The functions $f_i : \mathbb{R}^i \to \mathbb{R}$, $i = 1, \dots, n$ are Lipschitzian and satisfy the following property [5]

$$|f_i(x_1, \dots, x_i) - f_i(\bar{x}_1, \dots, \bar{x}_1)| \leqslant \gamma_{f_i} \sum_{j=1}^{i} |x_j - \bar{x}_j|, \tag{4.27}$$

where γ_{f_i} is Lipschitz's constant.

The observer developed in [5] is a modified high-gain observer based on the same change-of-variable principle as above, enabling fixed-time state estimation at a prescribed time T. Details of the observer design are given in the following.

4.4.1 Design

The observer for estimating the states of the system (4.25) in a prescribed time T is given as follows [5] :

$$\dot{\hat{x}} = A\hat{x}(t) + f(\hat{x}(t)) + \Gamma^{-1}K(y(t) - \hat{x}_1(t)), \tag{4.28}$$

where Γ is a scaling matrix given by :

$$\Gamma = \begin{bmatrix} 1/\mu^{1+m} & 0 & \dots & 0 \\ 0 & 1/\mu^{2(1+m)} & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & \dots & \dots & 1/\mu^{n(1+m)} \end{bmatrix}, \tag{4.29}$$

with $m \geqslant 1$ is the degree of freedom.
The dynamics of the estimation error $e(t) = x(t) - \hat{x}(t)$ is expressed by :

$$\dot{e}(t) = Ae(t) + \Delta f - \Gamma^{-1} e_1(t), \tag{4.30}$$

where

$$\Delta f := f(x) - f(\hat{x}). \tag{4.31}$$

As we have already seen, in order for the estimation error to be stable in fixed time at a prescribed time T, we consider the following coordinate transformation

$$z(t) = \Gamma e(t). \tag{4.32}$$

The dynamics of the estimation error expressed in the new base is given as follows

$$\dot{z}(t) = \mu^{1+m}(A - KC)z(t) - (1+m)\frac{\dot{\mu}}{\mu} Dz(t) + \Gamma \Delta f, \tag{4.33}$$

with

$$D = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 2 & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & \dots & \dots & n \end{bmatrix}. \tag{4.34}$$

Remember that due to the nature of the transformation $e(t) \to z(t)$, if $z(t)$ remains a finite quantity, $e(t)$ will be forced to tend towards zero when $t \to t_0 + T$. So in order to ensure that (4.30) is GUAFxS in a prescribed time T, all we need to do is ensure that (4.33) is AS.

Consider the following Lyapunov function [5] :

$$V(z(t)) = z(t)^T P z(t), \tag{4.35}$$

where $P > 0$.
Ensuring that (4.33) is AS is equivalent to forcing the derivative of V to be a NDF on $[t_0, t_0 + T]$, this implies the following theorem

Theorem 6.4.1 : [5] If there exists a matrix $P > 0$, a matrix of adequate dimension $\mathcal{Y}$ and positive constants m, λ_1 and λ_2 ensuring the following conditions :

$$A^T P + PA - C^T \mathcal{Y} - \mathcal{Y}^T C + \lambda_1 I_n \leqslant 0, \tag{4.36}$$

$$D^T P + PD - \lambda_2 I_n \geqslant 0, \tag{4.37}$$

with

$$K = P^{-1}\mathcal{Y}^T = \begin{bmatrix} K_1 & \dots & K_n \end{bmatrix}^T, \tag{4.38}$$

then the dynamics of the error expressed in the new base (4.33) is AS.

Remark 6.4.1 : Constructing a prescribed-time observer for the system (4.25) involves solving the system of linear matrix inequalities (LMI) (4.41)(4.42).
More details on the design of the observer and the proof of *Theorem 6.4.1* can be found in [5]. The performances of this observer are presented in the following example.

4.4.2 Illustrative example

Let's consider the second-order nonlinear system expressed by :

$$\begin{cases} \dot{x}_1(t) = x_2(t) + \cos(x_1(t)x_2(t)), \\ \dot{x}_2(t) = x_2(t)^2 - x_1(t)x_2(t, \\ y(t) = x_1(t). \end{cases} \tag{4.39}$$

the system (4.39) is written in the form described by (4.25) such that

$$A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad f(x(t)) = \begin{bmatrix} f_1(x) \\ f_2(x) \end{bmatrix} = \begin{bmatrix} \cos(x_1(t)x_2(t)) \\ x_2(t)^2 - x_1(t)x_2(t) \end{bmatrix}.$$

The prescribed-time observer associated with the system (4.39) is given by

$$\begin{cases} \dot{\hat{x}}_1(t) = \hat{x}_2(t) + \cos(x_1(t)\hat{x}_2(t)) + \mu^{1+m}(t)K_1(x_1(t) - \hat{x}_1(t)), \\ \dot{\hat{x}}_2(t) = \hat{x}_2(t)^2 - x_1(t)\hat{x}_2(t) + \mu^{2(1+m)}(t)K_2(x_1(t) - \hat{x}_1(t)). \end{cases} \tag{4.40}$$

Using Matlab, we solve the following system of linear matrix inequalities to find the observer's gains K_1 et K_2

$$A^T P + PA - C^T \mathcal{Y} - \mathcal{Y}^T C + \lambda_1 I_2 \leqslant 0, \tag{4.41}$$

$$D^T P + PD - \lambda_2 I_2 \geqslant 0, \tag{4.42}$$

where

$$D = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}, \quad \lambda_1 = 1, \quad \lambda_2 = 0.5, \tag{4.43}$$

$$K = P^{-1}\mathcal{Y}^T = \begin{bmatrix} K_1 & K_2 \end{bmatrix}^T, \tag{4.44}$$

we find

$$K = \begin{bmatrix} 1.45 & 2.7 \end{bmatrix}. \tag{4.45}$$

The convergence time is fixed at $T = 4$ and the parameter $m = 1$. The time origin is taken equal to $t_0 = 0$. The initial conditions of the system and the observer are $x_0 = [0, 1]$ and $\hat{x}_0 = [1, 2]$.

Figures (4.6) and (4.6) illustrate the evolution of estimation errors as well as the estimates of x_1 and x_2 respectively for different values of parameter m. The estimates do converge towards their true values in a time $T = 4$ and this independently of the values of m, having said that, we notice that by increasing the value of m the estimates tend to converge more rapidly.

4.5 Conclusion

In this chapter, we have presented a recently developed observer known as the prescribed-time observer. Its main characteristic is that it ensures the fixed time stability of the estimation error within a predefined time fixed by the user, independently of the observer's initial conditions and the system's parameters. This approach is based on the use of time-varying gains, which tend towards infinity as the prescribed time T is reached, thanks to a specific coordinate change. We have demonstrated the performance of this observer using simple examples, where the results were very satisfying. However, in the presence of measurement noise, as we approach the prescribed time T, estimates diverged due to numerical instabilities, which are exacerbated by measurement noise.

The aim of this chapter was to introduce the fundamental principles on which this observer approach is based, thus opening up new perspectives for future research directions. As a potential example, we can consider a few ideas worth developing, such as extending this approach to more general forms of systems, further analyzing the robustness of these observers and proposing improvements to mitigate the effects of numerical instability and measurement noise. These lines of research open up new prospects for exploring in greater detail the capabilities and limitations of this innovative approach.

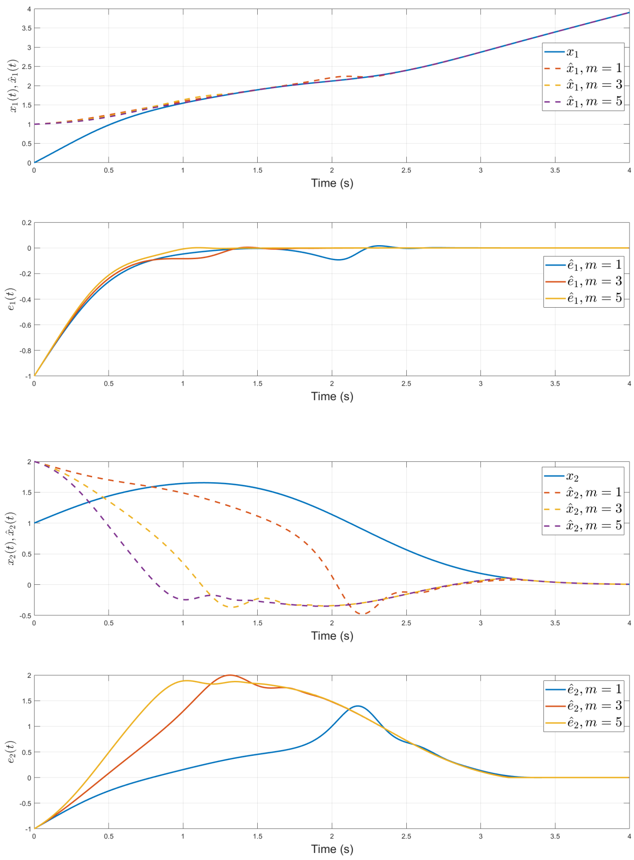

Figure 4.6: Estimation of the state x_1 along with estimation error with $T = 4$, $\hat{x} = [1, 2]^T$ for different values of m

Chapter 5

Application to drone

5.1 Introduction - Problem formulation

Drones have been highly effective in many outdoor applications, including infrastructure monitoring and search and rescue operations. It is questionable whether this effectiveness would be the same in enclosed moving environments, such as vehicles, ships or aircrafts, where GPS is inaccessible. Flying in these environments can present several challenges. The absence of a GPS signal deprives UAVs (unmanned aerial vehicle) of direct access to their absolute position in the geocentric coordinate system, considered an inertial reference frame for these vehicles. On-board sensors such as accelerometers and gyroscopes provide measurements of angles and accelerations relative to this inertial reference frame. In contrast, on-board location sensors provide positions relative to the moving environment, which is a non-inertial reference frame. As a result, the visual odometry of drones is discordant with inertial measurements. Two YouTube videos illustrate this challenge, showing people trying to fly drones inside trucks or elevators [41][40] Human pilots manage to control the drones at very low speeds, but this task becomes increasingly difficult to downright impossible as the environment speed and acceleration increase.

This particular problem has recently been introduced in [49]. No other work has considered this scenario to date. In a first attempt, the authors of [49] proposed a formulation for the problem of UAV flight in a moving environment with translational motion relative to the geostationary reference frame. A model corresponding to this situation was developed, taking into account the acceleration of the environment in which the drone is located. Next, an extended kalman filter with unknown inputs (EKF-UI) was implemented to estimate the drone's linear velocities and the accelerations of the environment. The estimated velocities were then used in a sliding-mode controller, along with the other measured states, to enable the drone to follow the desired trajectory. The results obtained were satisfactory in both simulation and experimentation in an elevator, further confirming their work and validating the effectiveness and reliability of their proposals.

The next two chapters (5 and 6) of our study will be dedicated to this specific problem. In chapter 5, we will present in detail the model developed in [49] and focus on the control aspect of the problem. We will implement two control laws to enable the drone to follow a desired trajectory, namely the sliding mode control already treated by [49] and the backstepping control. The next chapter will focus on the estimation problem.

5.2 Drone dynamical model

We consider the case where the moving environment moves only in translation with respect to the inertial frame of reference. Before a model of the UAV with respect to the non-inertial reference frame can be developed, we must first establish a model with respect to the inertial reference frame. In the following, we'll refer to the inertial and non-inertial reference frames as $R(o, x, y, z)$ and $R'(o', x', y', z')$ respectively.

5.2.1 Dynamical model w.r.t the inertial reference frame

The equations of motion of the UAV in the reference frame R based on the Newton-Euler formalism are given by [49][13]

$$
\begin{cases}
\ddot{x}_{/R} = u_1(c(\phi_{/R})s(\theta_{/R})c(\psi_{/R}) + s(\phi_{/R})s(\theta_{/R})), \\
\ddot{y}_{/R} = u_1(c(\phi_{/R})s(\theta_{/R})s(\psi_{/R}) - s(\phi_{/R})c(\theta_{/R})), \\
\ddot{z}_{/R} = u_1(c(\phi_{/R})c(\theta_{/R})) - g, \\
\ddot{\phi}_{/R} = \frac{I_y - I_z}{I_x}\dot{\theta}_{/R}\dot{\psi}_{/R} + u_2, \\
\ddot{\theta}_{/R} = \frac{I_z - I_x}{I_y}\dot{\phi}_{/R}\dot{\psi}_{/R} + u_3, \\
\ddot{\psi}_{/R} = \frac{I_x - I_y}{I_z}\dot{\phi}_{/R}\dot{\psi}_{/R} + u_4,
\end{cases}
\tag{5.1}
$$

the moments of inertia along each axis are represented by I_x,I_y and I_z and $g = 9.81$ m/s^2 represents the gravitational force. The terms $\phi_{/R}$, $\theta_{/R}$ and $\psi_{/R}$ correspond respectively to the drone's pitch, roll and yaw angles in the reference frame R, while $x_{/R}$, $y_{/R}$ and $z_{/R}$ represent the drone's position in the same reference frame. The variables $u_i,\ i = 1, 2, 3, 4$ designate the drone's control inputs and are defined as follows

$$
\begin{cases}
u_1 = \frac{b}{m}(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2), \\
u_2 = \frac{b}{I_x}(\Omega_4^2 - \Omega_2^2), \\
u_3 = \frac{b}{I_y}(\Omega_3^2 - \Omega_1^2), \\
u_4 = \frac{l}{I_z}(-\Omega_1^2 + \Omega_2^2 - \Omega_3^2 + \Omega_4^2),
\end{cases}
$$

where $a = \frac{I_y - I_z}{I_x}$, $b = \frac{I_z - I_x}{I_y}$,$c = \frac{I_x - I_y}{I_z}$, l is the distance between the rotor and the drone's center of mass and Ω_1, Ω_2, Ω_3 and Ω_4 represent the angular velocities of the rotors.

We define the following state vector: $x_r = (x_{1r}, x_{2r}, x_{3r}, x_{4r}, x_{5r}, x_{6r}, x_{7r}, x_{8r}, x_{9r}, x_{10r}, x_{11r}, x_{12r})^T = (x_{/R}, \dot{x}_{/R}, y_{/R}, \dot{y}_{/R}, z_{/R}, \dot{z}_{/R}, \phi_{/R}, \dot{\phi}_{/R}, \theta_{/R}, \dot{\theta}_{/R}, \psi_{/R}, \dot{\psi}_{/R})^T$, the state space representation of the UAV expressed in the inertial reference frame R is given by

$$
\begin{cases}
\dot{x}_{1r} = x_{2r}, \\
\dot{x}_{2r} = [c(x_{7r})s(x_{9r})c(x_{11r}) + s(x_{7r})s(x_{11r})]u_1, \\
\dot{x}_{3r} = x_{4r}, \\
\dot{x}_{4r} = [c(x_7)s(x_9)s(x_{11}) - s(x_7)c(x_{11})]u_1 - a_y, \\
\dot{x}_{5r} = x_6, \\
\dot{x}_{6r} = [c(x_7)c(x_9)]u_1 - g - a_z, \\
\dot{x}_{7r} = x_8, \\
\dot{x}_{8r} = ax_{10}x_{12} + u_2, \\
\dot{x}_{9r} = x_{10}, \\
\dot{x}_{10r} = bx_8x_{12} + u_3, \\
\dot{x}_{11r} = x_{12}, \\
\dot{x}_{12r} = cx_8x_{10} + u_4.
\end{cases}
$$

5.2.2 Dynamical model w.r.t the non-inertial refrence frame

Since the non-inertial reference frame is assumed to move in translation only, only the drone's linear positions and velocities are affected. Angular positions and angular velocities in the non-inertial reference frame are identical to those in the geostationary reference frame. To better illustrate this idea, let's imagine that the drone is inside a box, as shown in figure (5.1). In this case, the non-inertial frame of reference corresponds to the frame of reference linked to the body of the box. By considering the moving environment as a box, we can encompass all possible scenarios: if we want to fly the drone in an elevator, we can represent it as a box moving along the z axis.

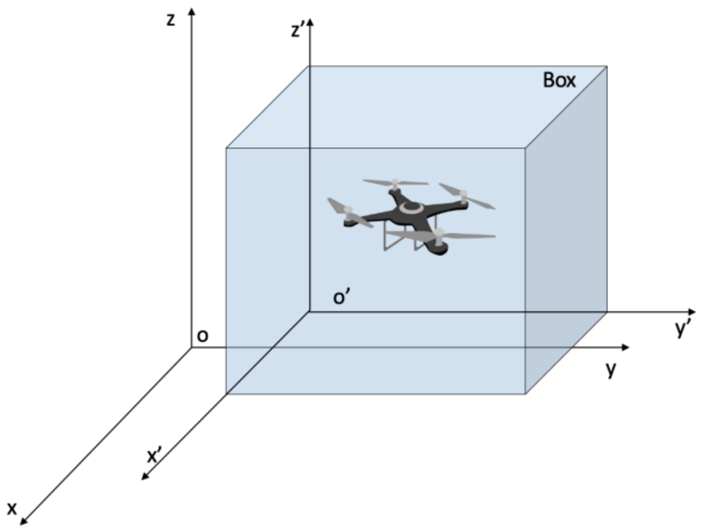

Figure 5.1: Representation of the different refrence frames [[49]

We denote $x_{b/R}$, $y_{b/R}$ and $z_{b/R}$ the position of the fictitious box (the environment) in R, while $\phi_{b/R}$, $\theta_{b/R}$ and $\psi_{b/R}$ represent the orientation of the box in R. Since the box is assumed to move in translation only, we have

$$\begin{bmatrix} \ddot{x}_{b/R} \\ \ddot{y}_{b/R} \\ \ddot{z}_{b/R} \\ \ddot{\phi}_{b/R} \\ \ddot{\theta}_{b/R} \\ \ddot{\psi}_{b/R} \end{bmatrix} = \begin{bmatrix} a_x \\ a_y \\ a_z \\ 0 \\ 0 \\ 0 \end{bmatrix}. \tag{5.2}$$

The UAV's linear accelerations in R are expressed as the sum of the UAV's acceleration in R' and the acceleration of the environment. Consequently, the drone's accelerations relative to the reference frame R' are expressed as follows

$$\begin{bmatrix} \ddot{x}_{/R} \\ \ddot{y}_{/R} \\ \ddot{z}_{/R} \end{bmatrix} = \begin{bmatrix} \ddot{x}_{/R'} + \ddot{x}_{b/R} \\ \ddot{y}_{/R'} + \ddot{y}_{b/R} \\ \ddot{z}_{/R'} + \ddot{z}_{b/R} \end{bmatrix} \rightarrow \begin{bmatrix} \ddot{x}_{/R'} \\ \ddot{y}_{/R'} \\ \ddot{z}_{/R'} \end{bmatrix} = \begin{bmatrix} \ddot{x}_{/R'} - \ddot{x}_{b/R} \\ \ddot{y}_{/R'} - \ddot{y}_{b/R} \\ \ddot{z}_{/R'} - \ddot{z}_{b/R} \end{bmatrix}.$$

By direct measurement, we have access to the relative position of the UAV with respect to the non-inertial reference frame R', as well as to its angular position. So the state representation of the UAV in R' is given by

$$
\begin{cases}
\dot{x}_1 = x_2, \\
\dot{x}_2 = [c(x_7)s(x_9)c(x_{11}) + s(x_7)s(x_{11})]u_1 - a_x, \\
\dot{x}_3 = x_4, \\
\dot{x}_4 = [c(x_7)s(x_9)s(x_{11}) - s(x_7)c(x_{11})]u_1 - a_y, \\
\dot{x}_5 = x_6, \\
\dot{x}_6 = [c(x_7)c(x_9)]u_1 - g - a_z, \\
\dot{x}_7 = x_8, \\
\dot{x}_8 = ax_{10}x_{12} + u_2, \\
\dot{x}_9 = x_{10}, \\
\dot{x}_{10} = bx_8x_{12} + u_3, \\
\dot{x}_{11} = x_{12}, \\
\dot{x}_{12} = cx_8x_{10} + u_4,
\end{cases} \tag{5.3}
$$

$$
y = [x_1 \; x_3 \; x_5 \; x_7 \; x_9 \; x_{11}]^T.
$$

The state vector is given by $x = (x_{/R'}, \dot{x}_{/R'}, y_{/R'}, \dot{y}_{/R'}, z_{/R'}, \dot{z}_{/R'}, \phi_{/R'}, \dot{\phi}_{/R'}, \theta_{/R'}, \dot{\theta}_{/R'}, \psi_{/R'}, \dot{\psi}_{/R'})$. The accelerations of the non-inertial reference frame (a_x,a_y, and a_z) act as a disturbance on the UAV model and can be considered as unknown inputs.

5.3 Control problem

In the system (5.3), it's important to note that angles and their time derivatives don't depend on translation components, whereas translations do depend on angles. This allows us to consider the system (5.3) as consisting of two distinct subsystems : angular rotations and linear translations, as illustrated in Figure (5.2). The output of the angular rotations subsystem then becomes a control input for the translations subsystem.

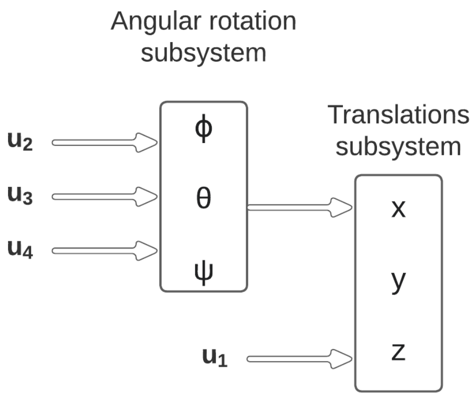

Figure 5.2: Block diagram of the drone model

To ensure efficient control of the entire system, the control scheme is divided into two parts: a position controller and a rotation controller, as shown in Figure (5.3).
In the control scheme, the desired values (x_d, y_d, z_d, ψ_d) are set, the position controller generates the required values (ϕ_d, θ_d) for the rotation controller. The measured quantities

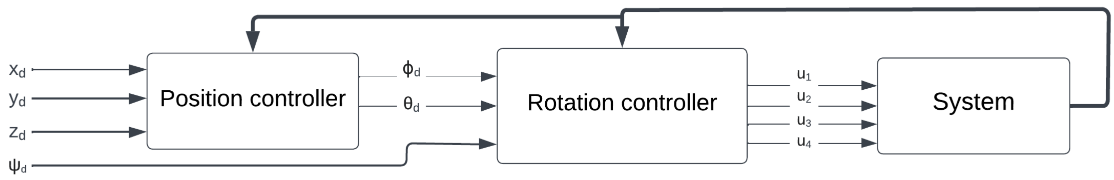

Figure 5.3: UAV control structure

are then fed back to both controllers.

In order to design the control laws, we define u_x and u_y as virtual controls given by

$$\begin{cases} u_x = c(x_7)s(x_9)c(x_{11}) + s(x_7)s(x_{11}), \\ u_y = c(x_7)s(x_9)s(x_{11}) - s(x_7)c(x_{11}). \end{cases} \tag{5.4}$$

The drone model can be rewritten as follows

$$\begin{cases} \dot{x}_1 = x_2, \\ \dot{x}_2 = u_x u_1 - a_x, \\ \dot{x}_3 = x_4, \\ \dot{x}_4 = u_y u_1 - a_y, \\ \dot{x}_5 = x_6, \\ \dot{x}_6 = [c(x_7)c(x_9)]u_1 - g - a_z, \\ \dot{x}_7 = x_8, \\ \dot{x}_8 = a x_{10} x_{12} + u_2, \\ \dot{x}_9 = x_{10}, \\ \dot{x}_{10} = b x_8 x_{12} + u_3, \\ \dot{x}_{11} = x_{12}, \\ \dot{x}_{12} = c x_8 x_{10} + u_4. \end{cases} \tag{5.5}$$

5.3.1 Sliding mode control

5.3.1.1 A reminder about sliding mode control

Sliding mode control is an advanced control technique commonly used in control systems. Its main objective is to enable precise control of dynamical systems that are non-linear and uncertain, using switching control based on state variables. The fundamental idea of this approach is to align the actual dynamics of the system with a desired dynamics, which is defined by a specific surface called the sliding surface. When the state of the system remains on this surface, the system is said to be in a sliding regime. At this stage, the system becomes insensitive to parameter variations, modeling errors and disturbances.

Sliding-mode control is particularly useful for controlling disturbed systems or systems with poorly understood models. Its objective focuses on two essential aspects. Firstly, it involves designing a surface, denoted $S(x)$, so that all system trajectories follow a desired behavior in terms of tracking, control and stability. Next, we need to determine a control or switching law, denoted $u(t, x)$, which attracts all state trajectories to the sliding surface and holds them there.

Consider the following non-linear system

$$\begin{cases} \dot{x} = f(t, x) + g(t, x)u, \\ y = h(x), \end{cases} \tag{5.6}$$

with $x \in \mathbb{R}^n$ is the state vector, $u \in \mathbb{R}^m$ is the control input vector and $y \in \mathbb{R}^p$ is the output vector.
Designing a sliding-mode controller for the system (5.6) generally involves three steps, as described in [15] and [57]. Here's an overview of these steps:

1- Choosing a sliding surface :

Slotine and Li propose in [79], a general form of the sliding surface given by

$$S(x) = (\frac{d}{dt} + \lambda)^{r-1} e(x), \tag{5.7}$$

where $e(x)$ represents the tracking error, λ is a positive constant and r is the relative degree of the system under consideration.

2- Attractivity condition :

The attractivity condition is the condition under which the state trajectory of a system reaches the sliding surface. It is also known as the "sliding surface access condition". There are two types of sliding surface access conditions : the direct approach and the Lyapunov approach. In this context, we'll use the direct approach, which is the oldest and was proposed by Emilyanov and Utkin, it is given by

$$\begin{cases} \dot{S}(x) > 0 \quad \text{When} \quad S(x) < 0, \\ \dot{S}(x) < 0 \quad \text{When} \quad S(x) > 0. \end{cases} \tag{5.8}$$

These two conditions can be rewritten as a single condition, as follows

$$\dot{S}(x) S(x) < 0. \tag{5.9}$$

3- Establishing the control law :

In order to bring the system states to the sliding surface and maintain them there, even in the presence of uncertainties and disturbances, the control u is composed of two terms [57], the control is given by

$$u(t) = u_{att} + u_{eq}(t). \tag{5.10}$$

The first term u_{att}, called the switching term, aims to attract the state trajectories towards the sliding surface. It is designed to ensure fast and accurate convergence of the system to the sliding surface in a finite time, satisfying the attractiveness condition $\dot{S}(x)S(x) < 0$. The simplest and most widely used form is

$$u_{att} = -K \frac{1}{g(t, x)} sign(S). \tag{5.11}$$

The second term, called the correction term, is used to maintain the state trajectories on the sliding surface once they have reached it. Its role is to eliminate the undesirable effects of disturbances and uncertainties, thus ensuring that the system remains on the sliding surface robustly, satisfying the surface invariance condition

$$\begin{cases} S(x) = 0, \\ \dot{S}(x) = 0. \end{cases}$$

This leads to the expression

$$u_{eq}(t) = -[\frac{\partial S}{\partial x} g(t,x)]^{-1}(\frac{\partial S}{\partial x} f(t,x) + \frac{\partial S}{\partial t}). \tag{5.12}$$

By combining these two terms, we obtain a control u that guides the system states towards the sliding surface and maintains them there, even in the presence of disturbed or uncertain conditions:

$$u(t) = -K\frac{1}{g(t,x)} sign(S) - [\frac{\partial S}{\partial x} g(t,x)]^{-1}(\frac{\partial S}{\partial x} f(t,x) + \frac{\partial S}{\partial t}). \tag{5.13}$$

Chattering phenomenon :

In an ideal sliding regime, switching the control at an infinite frequency would be desirable. However, in practice, switching at a finite frequency is realistic, creating a delay between output measurement and control calculation. This delay can cause the system to overshoot the sliding surface before the control can react. As a result, during the sliding regime, control discontinuities can cause high-frequency oscillations in the system's trajectory around the sliding surface. This phenomenon is known as *chattering*. This leads to a deterioration in the system performance, and may even result in instability

To mitigate or eliminate this phenomenon, several solutions have been proposed. One commonly used method is the boundary layer, which replaces the sign function of the control law with a high-gain continuous approximation in the vicinity of the sliding surface, while saturating it outside this vicinity. This allows the sliding regime to extend beyond the sliding surface and into a neighborhood of it. This is referred to as a pseudo-sliding regime. However, this method reduces the robustness of the control. It is parameterized with a constant adjusted to strike a balance between reducing chattering and preserving system robustness. A commonly used approximation to the discontinuous function *sign* is the function *arctan* [15].

5.3.1.2 Sliding mode control design

The sliding surfaces considered are [49]

$$\begin{cases} S_x = \dot{e}_1 + \lambda_1 e_1, \\ S_y = \dot{e}_3 + \lambda_3 e_3, \\ S_z = \dot{e}_5 + \lambda_5 e_5 \\ S_\phi = \dot{e}_7 + \lambda_7 e_7, \\ S_\theta = \dot{e}_9 + \lambda_9 e_9, \\ S_\psi = \dot{e}_{11} + \lambda_{11} e_{11}, \end{cases} \tag{5.14}$$

where $e_i = x_i - x_{id}$, with $i = 1, 3, 5, 7, 9, 11$, are the tracking errors, and λ_i, $i = 1, 3, 5, 7, 9, 11$, are positive constants.

Hypothesis 4.3.1 : The accelerations of the environment are bounded such that

$$|a_x| \leqslant a_1, \quad |a_y| \leqslant a_3, \quad |a_z| \leqslant a_5.$$

Control law of x :

Considering the surface S_x, we present in what follows the steps of the control synthesis. The derivative of the surface S_x is given by

$$\dot{S}_x = \ddot{e}_1 + \lambda_1 \dot{e}_1. \tag{5.15}$$

To ensure system stability and reference tracking, the attractiveness condition $\dot{S}_x S_x \leqslant 0$ must be satisfied.

$$\begin{aligned}
\dot{S}_x S_x &= [\ddot{e}_1 + \lambda_1 \dot{e}_1], \\
&= S_x[\ddot{x}_1 - \ddot{x}_{1d} + \lambda_1(\dot{x}_1 - \dot{x}_{1d})], \\
&= S_x[\dot{x}_2 - \ddot{x}_{1d} + \lambda_1(x_2 - \dot{x}_{1d})], \\
&= S_x[u_x u_1 - a_x - \ddot{x}_{1d} + \lambda_1(x_2 - \dot{x}_{1d})], \\
&= S_x[u_x u_1 - \ddot{x}_{1d} + \lambda_1(x_2 - \dot{x}_{1d})] - S_x a_x, \\
&\leqslant S_x[u_x u_1 - \ddot{x}_{1d} + \lambda_1(x_2 - \dot{x}_{1d})] + |S_x||a_x|.
\end{aligned}$$

Using *Hypothesis 4.3.1*, we find

$$\begin{aligned}
\dot{S}_x S_x &\leqslant S_x[u_x u_1 - \ddot{x}_{1d} + \lambda_1(x_2 - \dot{x}_{1d})] + a_1|S_x|, \\
&\leqslant S_x \underbrace{[u_x u_1 - \ddot{x}_{1d} + \lambda_1(x_2 - \dot{x}_{1d}) + a_1 sign(S_x)]}_{\mathrm{I}}.
\end{aligned}$$

To ensure $\dot{S}_x S_x \leqslant 0$ it is enough to take $I = -k_1 sign(S_x)$, we obtain

$$u_x = \frac{1}{u_1}[-(k_1 + a_1) sign(S_x) + \ddot{x}_{1d} - \lambda_1(x_2 - \dot{x}_{1d})].$$

Following the same steps, we find the control laws of y, z, ϕ, θ and ψ, they are given by

$$\begin{cases}
u_1 = \frac{1}{c(x_7)c(x_9)}[-(k_5 + a_5) sign(S_z) + g + \ddot{x}_{5d} \\
\quad -\lambda_5(\hat{x}_6 - \dot{x}_{5d})], \\
u_2 = -k_7 sign(S_\phi) - a\hat{x}_{10}\hat{x}_{12} + \ddot{x}_{7d} - \lambda_7(\hat{x}_8 - \dot{x}_{7d}), \\
u_3 = -k_9 sign(S_\theta) - b\hat{x}_8\hat{x}_{12} + \ddot{x}_{9d} - \lambda_9(\hat{x}_{10} - \dot{x}_{9d}), \\
u_4 = -k_{11} sign(S_\psi) - c\hat{x}_8\hat{x}_{10} + \ddot{x}_{11d} - \lambda_{11}(\hat{x}_{12} - \dot{x}_{11d}), \\
u_x = \frac{1}{u_1}[-(k_1 + a_1) sign(S_x) + \ddot{x}_{1d} - \lambda_1(\hat{x}_2 - \dot{x}_{1d})], \\
u_y = \frac{1}{u_1}[-(k_3 + a_3) sign(S_y) + \ddot{x}_{3d} - \lambda_3(\hat{x}_4 - \dot{x}_{3d})],
\end{cases} \tag{5.16}$$

where k_i, $i = 1, 3, 5, 7, 9, 11$ are positive constants. u_1 is assumed to never reach zero.

x_{1d}, x_{3d}, x_{5d}, and x_{11d} are the desired trajectories that are defined a priori, while, x_{7d} and x_{9d} are obtained from the virtual control inputs u_x and u_y (5.4) by :

$$\begin{bmatrix} s(x_{7d}) \\ c(x_{7d})s(x_{9d}) \end{bmatrix} = \begin{bmatrix} s(x_{11d}) & -c(x_{11d}) \\ c(x_{11d}) & s(x_{11d}) \end{bmatrix} \begin{bmatrix} u_x \\ u_y \end{bmatrix}.$$

5.3.1.3 Results

In order to validate and test the designd laws, we present in the following figures two scenarios, in the first case the figures (5.4)-(5.6) represent respectively the positions as well as the linear speeds, the angles as well as the angular speeds and the control signals (real and virtual) by considering :

- Non-inertial reference frame accelerations : $a_x = 1m/s^2$, $a_y = 2m/s^2$ and $a_z = 3m/s^2$.
- Desired position and yaw angle : $x_{1d} = 2m$, $x_{3d} = 2.5m$, $x_{5d} = 1m$ and $x_{11d} = 0\ rad$.

As for the second scenario, we simulated the tracking of a circular trajectory, represented by figure (5.7), this time with

- Non-inertial reference frame accelerations : $a_x = \sin(0.5t)m/s^2$, $a_y = 2m\sin(0.5t)/s^2$ and $a_z = 3\sin(0.5t)m/s^2$.
- Desired position and yaw angle : $x_{1d} = 2\sin(t)m$, $x_{3d} = 2\sin(t)m$, $x_{5d} = 6m$ and $x_{11d} = 0\ rad$.

Simulations are performed in the presence of a normally distributed random process noise with a mean equal to 0.001. When implementing the control laws, we replaced $sign(S)$ by $tanh(S/0.1)$, with the UAV initially placed at the origin of the non-inertial reference frame ($x_0 = zeros(12,1)$).

The drone parameters used for the simulations are given in the table (5.1).

In both scenarios, the signals quickly converge to their respective references, enabling the drone to follow the desired trajectory in a moving environment, guaranteeing system stability. Although this technique offers flexibility of manipulation, the choice of the parameters λ_i and associated k_i is tricky, as inappropriate selection can lead to system instability. We note that despite replacing the $sign$ function by a continuous one, the Chattering phenomenon is still present in the u_2 and u_3 control signals, no doubt due to the presence of process noise.

Name	Symbol	Value	Unit
Mass	m	1	Kg
Distance from rotor to center of mass	l	0.24	m
Moment of inertia along x	I_x	8.1×10^{-3}	Kg.m^2
Moment of inertia along y	I_y	8.1×10^{-3}	Kg.m^2
Moment of inertia along z	I_z	14.2×10^{-3}	Kg.m^2

Table 5.1: Simulation parameters[11]

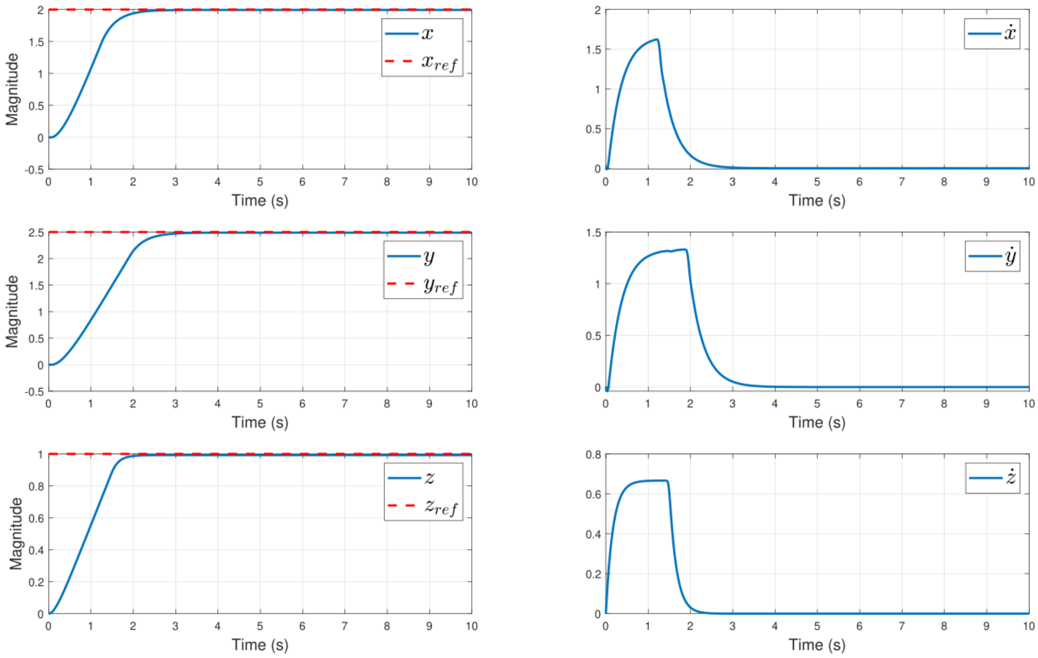

Figure 5.4: Positions and linear velocities (sliding mode control)

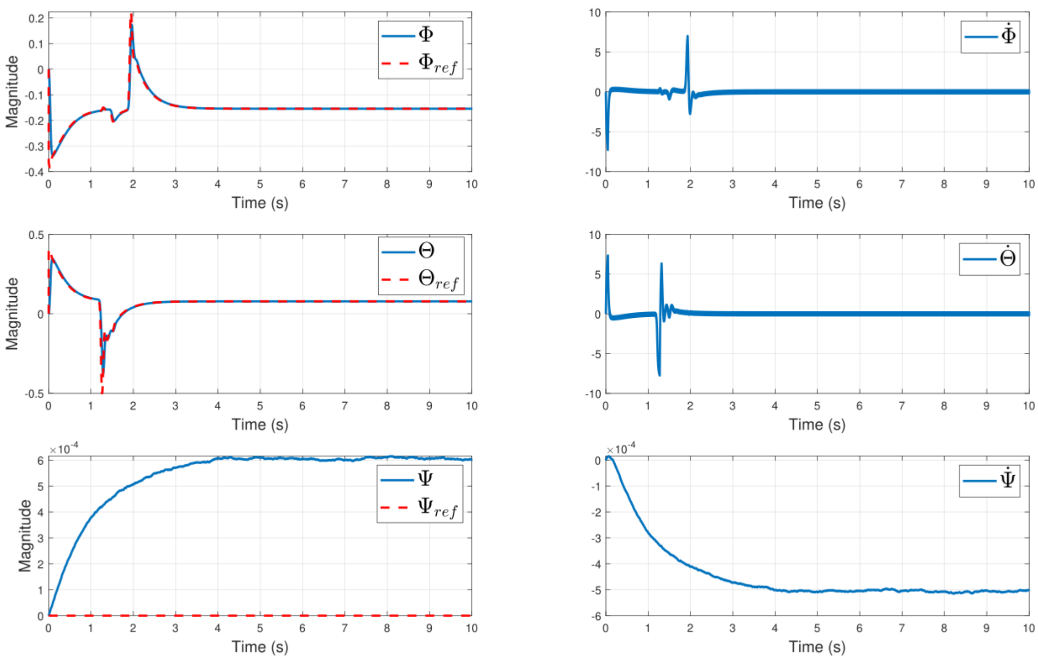

Figure 5.5: Orientation and angular velocities (Sliding mode control)

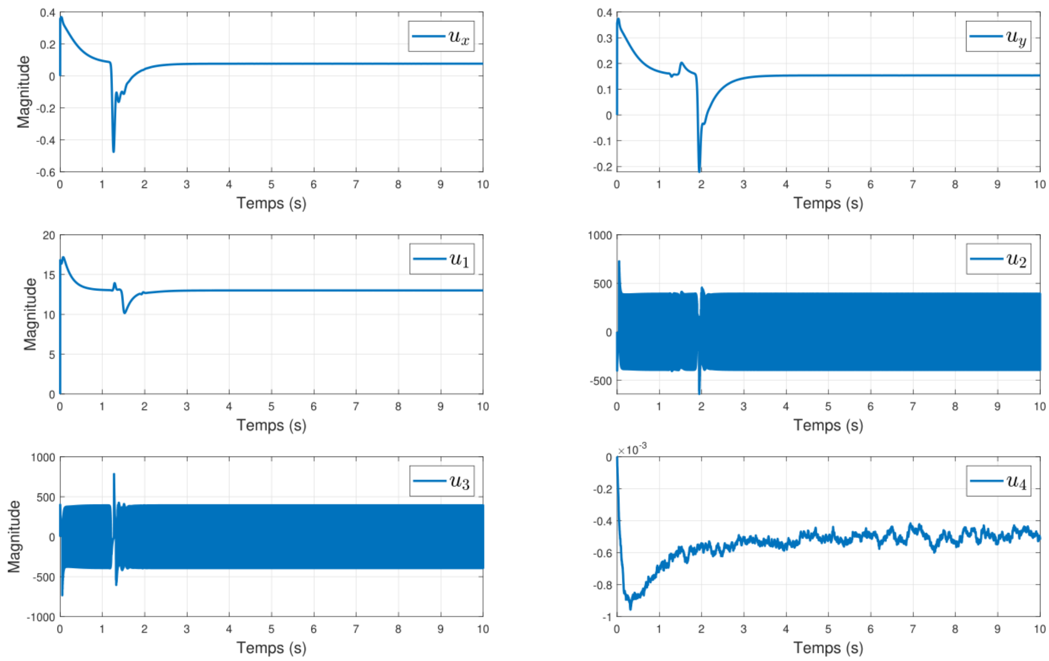

Figure 5.6: Control signals (sliding mode control)

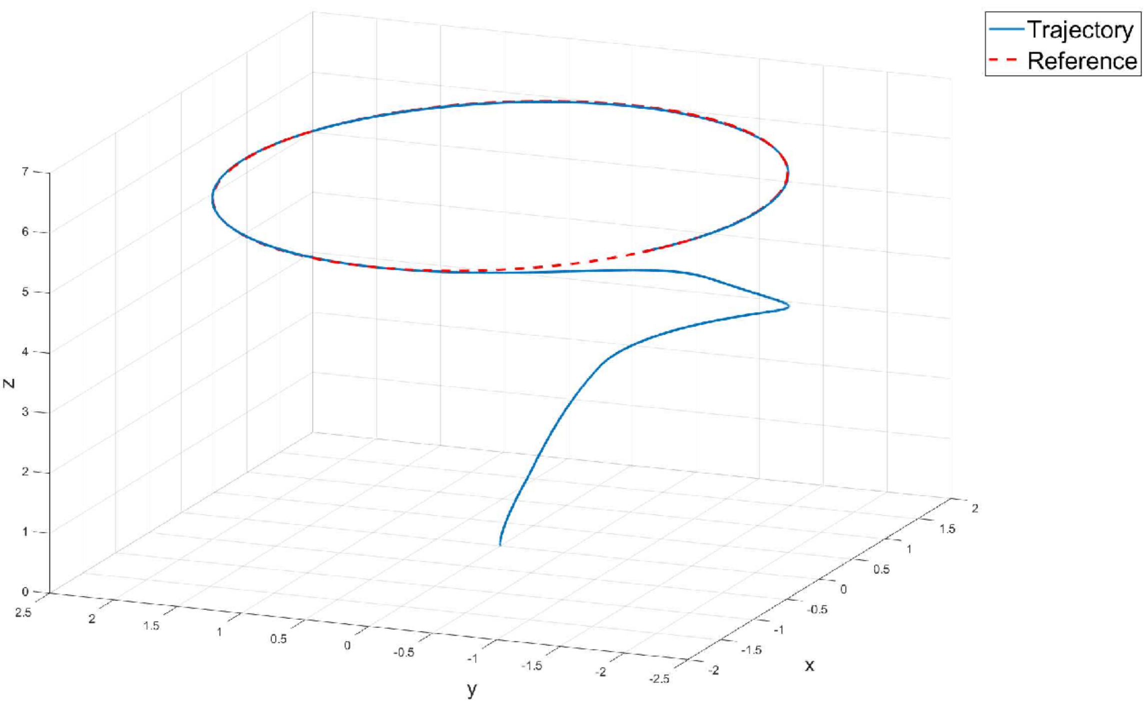

Figure 5.7: Trajectory tracking (sliding mode control)

5.3.2 Backstepping control

Backstepping control is an advanced control method widely used to stabilize and control non-linear dynamical systems, particularly in the field of robotics.

This approach is based on the concept of "backtracking", where the system is divided into several nested subsystems, in descending order. The Lyapunov function is calculated recursively, starting from inside the loop. At each step, the order of the system is increased and the unstabilized part of the previous step is processed. Finally, in the last step, the control law is determined. This control law guarantees the overall stability of the compensated system, ensuring that the system is tracked and regulated at all times.

5.3.2.1 A reminder about Backstepping control

Backstepping is a method that focuses on the progressive construction of the Lyapunov function and the step-by-step design of the control. Unlike many other methods, Backstepping imposes no constraints on the type of non-linearity of the system. However, the system must be formulated in a specific form called "cascade". The equations of such systems are given by

$$\begin{cases} \dot{x}_1 = f_1(x_1) + g_1(x_1)x_2, \\ \dot{x}_2 = f_2(x_1, x_2) + g_2(x_1, x_2)x_3, \\ \vdots \\ \dot{x}_{n-1} = f_{n-1}(x_1, x_2, \dots, x_{n-1}) + g_{n-1}(x_1, x_2, \dots, x_{n-1})x_n, \\ \dot{x}_n = f_n(x_1, x_2, \dots, x_n) + g_n(x_1, x_2, \dots, x_n)u, \\ y = x_1, \end{cases} \tag{5.17}$$

where : $x = \begin{bmatrix} x_1 & x_2 & \dots & x_n \end{bmatrix}^T \in \mathbb{R}^n$, $u \in \mathbb{R}$ and $y \in \mathbb{R}$.

To illustrate the recursive procedure of the Backstepping method, let's assume that the output of the system $y = x_1$, must follow a reference signal y_{ref}. Since the system is of order n, the Backstepping procedure takes place in n successive steps.

Step 1 :
We consider the first equation of the system (5.17) as a separate subsystem where x_2 is considered as an intermediate virtual control. The reference signal for x_1 is denoted

$$y_{ref} = x_{1d} = \alpha_0.$$

The control error and its dynamics are given by

$$\begin{cases} e_1 &= x_1 - \alpha_0, \\ \dot{e}_1 &= \dot{x}_1 - \dot{\alpha}_0, \\ &= f_1(x_1) + g_1(x_1)x_2 - \dot{\alpha}_0. \end{cases} \tag{5.18}$$

We define the Lyapunove function for the system (5.18) as follows

$$V_1 = \frac{1}{2}e_1^2. \tag{5.19}$$

V_1 is PDF on $\mathbb{R}$, its time derivative is given by :

$$\begin{aligned}\dot{V}_1 &= e_1\dot{e}_1,\\ &= e_1[f_1(x_1) + g_1(x_1)x_2 - \dot{\alpha}_0].\end{aligned}$$

A judicious choice of x_2 would make $\dot{V}_1$ NDF and ensure the stability of the system (5.18). To do this, we take

$$x_2 = \alpha_1 = \frac{1}{g_1(x_1)}[-k_1e_1 + \dot{\alpha}_0 - f_1(x_1)], \tag{5.20}$$

with $k_1 > 0$. This leads to

$$\dot{V}_1 = -k_1e_1^2 \leqslant 0. \tag{5.21}$$

Step 2 :
For the system (5.18) to be stable, $x_2 = \alpha_1$ is required, so α_1 is in turn the desired reference of x_2, x_3 is considered to be the virtual control of the x_2 state

$$x_{2d} = \alpha_1.$$

The control error of the variable x_2 and its dynamics are given by :

$$\begin{cases} e_2 &= x_2 - \alpha_1,\\ \dot{e}_2 &= \dot{x}_2 - \dot{\alpha}_1,\\ &= f_2(x_1,x_2) + g_2(x_1,x_2)x_3 - \dot{\alpha}_1.\end{cases} \tag{5.22}$$

We extend the previous Lyapunov function (5.19) to the system (5.22)

$$V_2 = V_1 + \frac{1}{2}e_2^2 = \frac{1}{2}(e_1^2 + e_2^2). \tag{5.23}$$

Its derivative is given by

$$\begin{aligned}\dot{V}_2 &= \dot{V}_1 + e_2\dot{e}_2,\\ &= -k_1e_1^2 + e_2[f_2(x_1,x_2) + g_2(x_1,x_2)x_3 - \dot{\alpha}_1 + e_1g_1(x_1)].\end{aligned}$$

The choice of x_3 that will stabilize the system (5.22) dynamics and make $\dot{V}_2$ NDF is

$$x_3 = \alpha_2 = \frac{1}{g_2(x_1,x_2)}[-k_2x_2 + \dot{\alpha}_1 - f_2(x_1,x_2) - e_1g_1(x_1)], \tag{5.24}$$

with $k_2 > 0$. We then obtain :

$$\dot{V}_2 = k_1e_1^2 - k_2e_2^2 \leqslant 0. \tag{5.25}$$

Step n :
Similarly, the reference to follow at this stage is

$$x_{nd} = \alpha_{n-1}.$$

The tracking error of state x_n and its dynamics are given by

$$\begin{cases} e_n &= x_n - \alpha_{n-1}, \\ \dot{e}_n &= \dot{x}_n - \dot{\alpha}_{n-1}, \\ &= f_n(x_1, x_2, \dots, x_n) + g_n(x_1, x_2, \dots, x_n)u. \end{cases} \tag{5.26}$$

The extended Lyapunov function of the system (5.26) is

$$V_n = V_{n-1} + \frac{1}{2}e_n^2 = \Sigma_{i=1}^{n} e_i^2. \tag{5.27}$$

$\dot{V}_n$ is given by

$$\dot{V}_n = -\Sigma_{i=1}^{n-1} k_i e_i^2 + e_n[f_n + g_n u - \dot{\alpha}_{n-1} + e_{n-1}g_{n-1}].$$

In order for $\dot{V}_n$ to be NDF, we take

$$u = \frac{1}{g_n}, \tag{5.28}$$

with $k_n > 0$. This gives $\dot{V}_n = -\Sigma_{i=1}^{n} k_i e_i^2$ which is NDF on $\mathbb{R}^n$.

The control u guarantees asymptotic convergence of the system output to the desired reference.

5.3.2.2 Backstepping control design

Tracking errors are given by

$$e_i = \begin{cases} x_i - x_{id}, & \text{for} \quad i \in \{1,3,5,7,9,11\}, \\ x_i - \dot{x}_{(i-1)d} + k_{i-1}e_{i-1}, & \text{for} \quad i \in \{2,4,6,8,10,12\}, \end{cases} \tag{5.29}$$

with $k_i > 0$.

The Lyapunov functions are written as

$$V_i = \begin{cases} \frac{1}{2}e_i^2, & \text{for} \quad i \in \{1,3,5,7,9,11\}, \\ \frac{1}{2}(V_{i-1} + e_i^2), & \text{for} \quad i \in \{2,4,6,8,10,12\}. \end{cases} \tag{5.30}$$

We design the Backstepping control of the motion along the x-axis subsystem , which is given by

$$\begin{cases} \dot{x}_1 = x_2, \\ \dot{x}_2 = u_x u_1 - a_x. \end{cases} \tag{5.31}$$

Since the subsystem (5.31) is a second order system, the design of the control law is performed in two steps:

Step 1 :
Let's consider x_2 as a virtual control law to control the state x_1 and make it follow the previously defined reference x_{1d}. The dynamics of the control error of the variable x_1 is given by

$$\dot{e}_1 = x_2 - \dot{x}_{1d}. \tag{5.32}$$

We define the Lyapunov function for the subsystem (5.32) and its derivative as follows

$$\begin{cases} V_1 = \frac{1}{2}e_1^2, \\ \dot{V}_1 = e_1\dot{e}_1 = e_1[x_2 - \dot{x}_{1d}]. \end{cases} \tag{5.33}$$

In order for the subsystem (5.32) to be stable, it is necessary that $\dot{V}_1 \leqslant 0$, this is verified by choosing the virtual control as

$$x_2 = \alpha_1 = \dot{x}_{1d} - k_1e_1, \tag{5.34}$$

in this case $\dot{V}_1 = -k_1e_1^2$, which guarantees convergence of e_1 to the origin.

Step 2 :
In order for x_1 to follow the reference x_{1d}, x_2 must in turn follow α_1. To do this, we take the Lyapunov function

$$\begin{cases} V_2 = \frac{1}{2}(e_1^2 + e_2^2), \\ \dot{V}_2 = e_1\dot{e}_1 + e_2\dot{e}_2 = -k_1e_1^2 + e_2[u_xu_1 - a_x - \dot{\alpha}_1 + e_1]. \end{cases} \tag{5.35}$$

To ensure system stability and reference tracking, $\dot{V}_2 \leqslant 0$ must be satisfied.

$$\begin{aligned} \dot{V}_2 &= -k_1e_1^2 + e_2[u_xu_1 - a_x - \dot{\alpha}_1 + e_1], \\ &= k_1e_1^2 + e_2[u_xu_1 - \dot{\alpha}_1 + e_1] - e_2a_x, \\ &\leqslant k_1e_1^2 + e_2[u_xu_1 - \dot{\alpha}_1 + e_1] + |e_2||a_x|. \end{aligned}$$

Using the *Hypothesis 4.3.1*, we find

$$\begin{aligned} \dot{V}_2 &\leqslant k_1e_1^2 + e_2[u_xu_1 - \dot{\alpha}_1 + e_1] + a_1|e_2|, \\ &\leqslant k_1e_1^2 + e_2[u_xu_1 - \dot{\alpha}_1 + e_1 + a_1sign(e_2)], \end{aligned}$$

by setting

$$u_x = \frac{1}{u_1}[-k_2e_2 + \dot{\alpha}_1 - e_1 - a_1sign(e_2)]. \tag{5.36}$$

We obtain : $\dot{V}_2 \leqslant -k_1e_1^2 - k_2e_2^2 \leqslant 0$.

Following the same steps, we design the control laws u_y, u_1, u_2, u_3 and u_4 :

$$\begin{cases} u_1 = \frac{1}{c(x_7)c(x_9)}(\dot{\alpha}_5 - e_5 + g - a_zsign(e_6) -\text{¿}6e_6), \\ u_2 = \dot{\alpha}_7 - e_7 - ax_{10}x_{12} - k_8e_8, \\ u_3 = \dot{\alpha}_9 - e_9 - ax_8x_{12} - k_{10}e_{10}, \\ u_4 = \dot{\alpha}_{11} - e_{11} - ax_8x_{10} - k_{12}e_{12}, \\ u_x = \frac{1}{u_1}[-k_2e_2 + \dot{\alpha}_1 - e_1 - a_1sign(e_2)], \\ u_y = \frac{1}{u_1}[-k_4e_4 + \dot{\alpha}_3 - e_3 - a_3sign(e_4)]. \end{cases} \tag{5.37}$$

u_1 is assumed to never reach zero.

5.3.2.3 Results

The two previous scenarios are repeated with the same initial conditions and simulation parameters, this time implementing the backstepping control laws given by (5.37). Figure (5.8) shows the positions and linear velocities of the drone, while Figure (5.9) illustrates the angles and their variations. Figure (5.10) shows the control signals (real and virtual). Finally, Figure (5.11) shows the drone's trajectory in space, as well as the reference trajectory.

Backstepping control enables the system outputs to converge towards their reference values, even though a small static error is present concerning the drone's position along the y and z axes. We also note that the pitch and roll angles initially fail to follow their reference values determined by the controller. However, this does not prevent the system from stabilizing and maintaining its reference position while in a moving environment. In Figure (5.11), we can see that backstepping control enables the drone to follow the circular reference trajectory. However, it is noticeable that the control takes longer to reach the reference trajectory than in Figure (5.7) (sliding mode control). It can also be seen that the control u_2 and u_3 signals exhibit strong peaks. This technique is flexible in its handling, but the choice of parameters k_i is difficult, as the wrong selection can lead to the instability of the system. It is therefore crucial to make appropriate choices to guarantee the stability and performance of the controlled system.

5.4 Conclusion

In this chapter, we presented the UAV model in a non-inertial reference frame R', as developed in [49]. Next, we designd two controllers, namely the sliding mode controller and the Backstepping controller. It's worth noting that sliding mode control has already been covered in [49]. We tested them under two scenarios : reaching and maintaining the desired position in the non-inertial reference frame R' and following a circular trajectory in the same frame. Both approaches has given satisfactory results. However, the sliding mode control seems to be the better choice due to its smaller control signals and its reputation for robustness against disturbances and modeling errors (which is essential when implementing on a real system).

However, it is important to note that the implementation of these control laws requires the knowledge of all the system's states at each given time, whereas we only have access to the measured relative position of the UAV and its orientation. This requires the introduction and the implementation of an observer to estimate the other non-measurable states. In the next chapter, we'll look at the estimation problem associated with the drone.

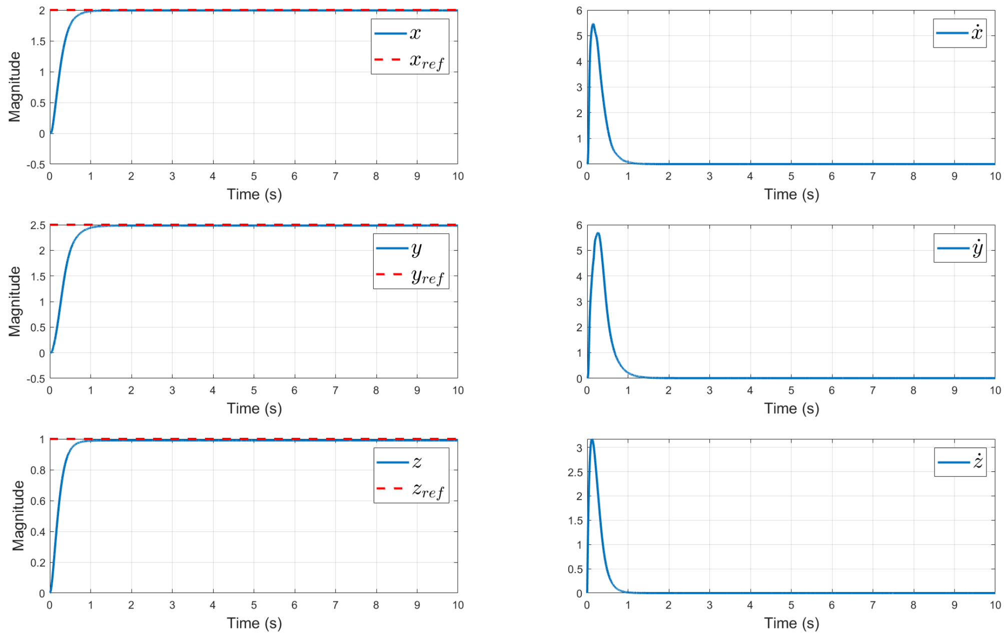

Figure 5.8: Positions and linear velocities (backstepping control)

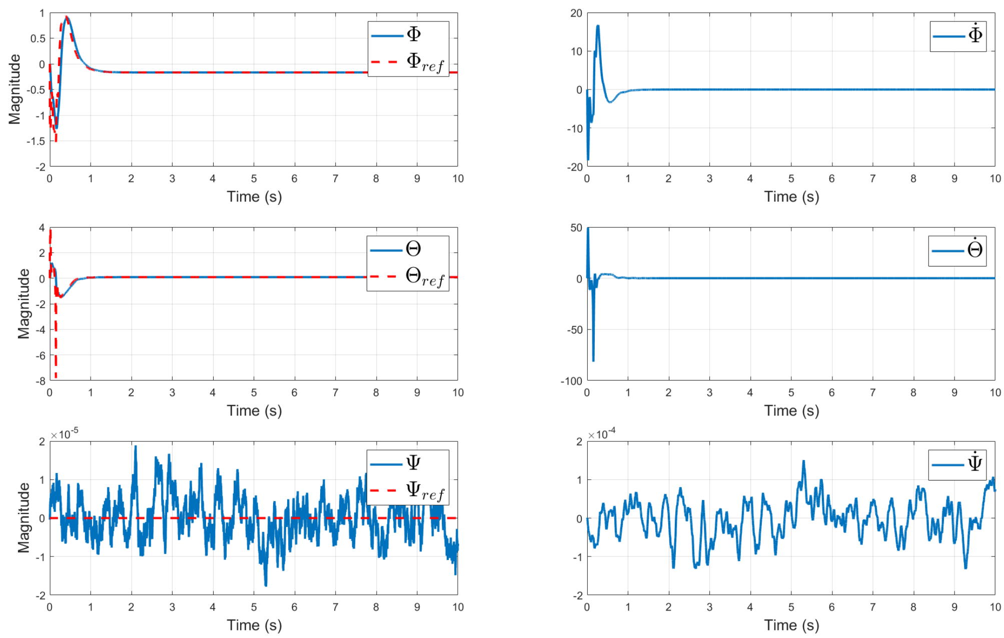

Figure 5.9: Orientation and angular velocities (backstepping control)

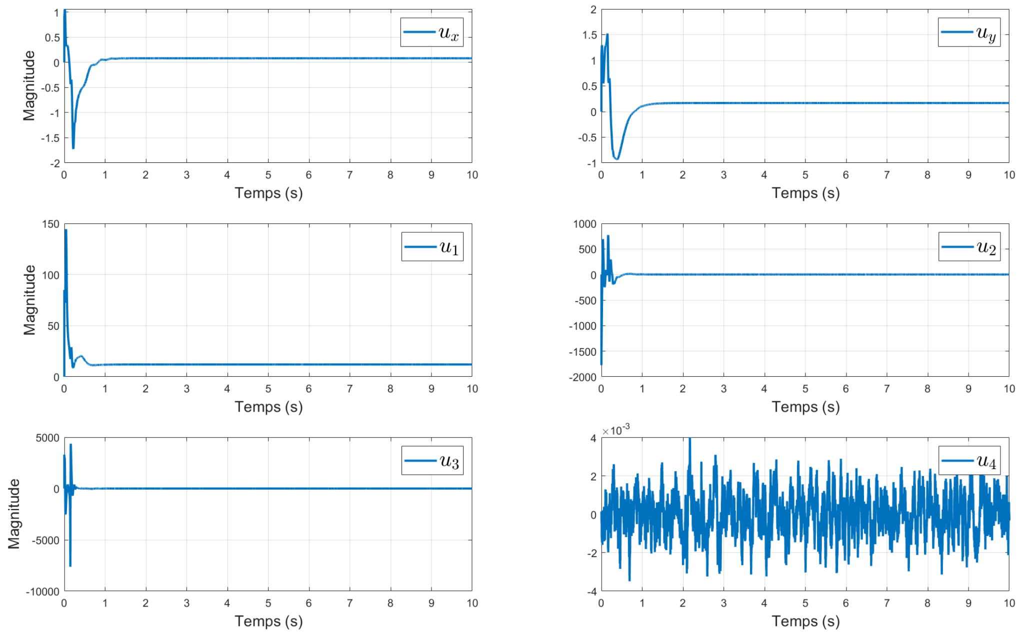

Figure 5.10: Control signals(Backstepping control)

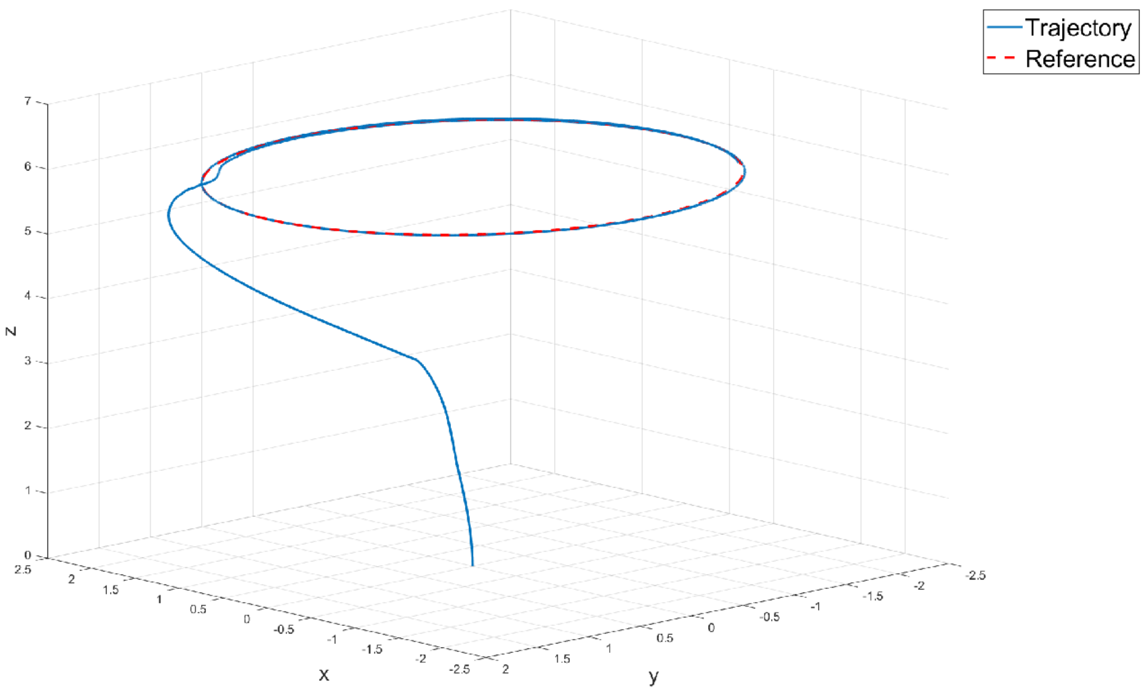

Figure 5.11: Trajectory tracking (backstepping control)

Chapter 6

Estimation of UAV's states and unknown inputs

6.1 Introduction

In this chapter, we'll focus on testing the MF observer for the UAV trajectory tracking problem in a non-inertial frame. We will combine this observer with the sliding mode controller and analyze its performance. We'll also compare it with other, better-known and commonly-used classical observers. The aim is to draw constructive conclusions about the most suitable observer for a given situation.

We will start by implementing the observer developed in section 3.6. Then, we will present the results obtained by combining it with the sliding mode controller described in the previous chapter. Finally, we will compare the modulating functions-based observer with two other observers, namely the unknown input Kalman filter and the super twisting sliding mode observer. This comparison will be based on several qualitative criteria. The objective of this approach is to provide useful information to determine which observer would be most suitable w.r.t each criteria.

6.2 Modulating functions based observer

Note that the drone model (5.3) has the same structure described by the class of nonlinear systems to which we have extended the MF estimator with unknown inputs (3.80), so we can rewrite the drone model in block triangular form, which is composed of $r = 6$ canonical triangular blocks with $n_j = 2, j = 1, 2, ..., r.$, such that

$$\begin{bmatrix} \dot{x}_{1,1} \\ \dot{x}_{2,1} \\ \dot{x}_{1,2} \\ \dot{x}_{2,2} \\ \dot{x}_{1,3} \\ \dot{x}_{2,3} \\ \dot{x}_{1,4} \\ \dot{x}_{2,4} \\ \dot{x}_{1,5} \\ \dot{x}_{2,5} \\ \dot{x}_{1,6} \\ \dot{x}_{2,6} \end{bmatrix} = \begin{bmatrix} x_{2,1} \\ [c(x_{1,4})s(x_{1,5})c(x_{1,6}) + s(x_{1,4})s(x_{1,6})]u_1 - a_x \\ x_{2,2} \\ [c(x_{1,4})s(x_{1,5})s(x_{1,6}) - s(x_{1,4})c(x_{1,6})]u_1 - a_y \\ x_{2,3} \\ [c(x_{1,4})c(x_{1,5})]u_1 - g - a_z \\ x_{2,4} \\ a x_{2,5} x_{2,6} + u_2 \\ x_{2,5} \\ b x_{2,4} x_{2,6} + u_3 \\ x_{2,6} \\ c x_{2,4} x_{2,5} + u_4 \end{bmatrix}, \tag{6.1}$$

where the known nonlinear functions are given by

$$\begin{cases} f_1(x,u) = [c(x_{1,4})s(x_{1,5})c(x_{1,6}) + s(x_{1,4})s(x_{1,6})]u_1, \\ f_2(x,u) = [c(x_{1,4})s(x_{1,5})s(x_{1,6}) - s(x_{1,4})c(x_{1,6})]u_1, \\ f_3(x,u) = [c(x_{1,4})c(x_{1,5})]u_1 - g, \\ f_4(x,u) = a x_{2,5} x_{2,6} + u_2, \\ f_5(x,u) = b x_{2,4} x_{2,6} + u_3, \\ f_6(x,u) = c x_{2,4} x_{2,5} + u_4, \end{cases}$$

and the bounded unknown inputs are

$$\begin{cases} d_1(t) = -a_x, \\ d_2(t) = -a_y, \\ d_3(t) = -a_z, \\ d_i(t) = 0, i = 4, 5, 6. \end{cases}$$

We can therefore easily apply the state estimation and unknown input approach developed in section 3.6. In what follows, we present the results obtained by combining sliding mode control with the unknown input observer we have proposed.

6.2.1 Observer simulations

In order to illustrate the performance of the MF based observer, we present in this section numerical simulations of its implementation to the problem of controlling the relative position of the UAV to the moving environment by combining it with the sliding mode controller designd in the previous chapter (5.16). The results presented below were obtained by setting the position's references as follows: $x_{1d} = 2$, $x_{1d} = 2.5$, $x_{5d} = 1$ and $x_{11d} = 0$. The accelerations of the non-inertial reference frame were taken as $a_x = 1$, $a_y = 2$ and $a_z = 3$. In addition, when implementing the sliding mode controller, we replaced $sign(S)$ by $\tanh(S/0.2)$.

From the figure (6.1), we can see that the drone's position does indeed reach the reference position and maintain it by combining the MFBM observer and sliding mode controller. In what follows, we first look at the state estimates, then present the results for the estimation of the unknown inputs. For the MF based estimator, we obviously observe the system in real time (online approach), using polynomial modulating functions for state and unknown input estimation.

$$\phi_i(\tau - t + h) = \frac{\bar{\phi}_i(\tau - t + h)}{||\bar{\phi}_i(\tau - t + h)||_2},$$

where $||.||_2$ is the Euclidean norm, and $\bar{\phi}_i(\tau - t + h)$ for state estimation and unknown inputs is given by

$$\bar{\phi}_i(\tau - t + h) = (t - \tau)^{(p_x+i)}(\tau - t + h)^{(p_x+M+1-i)} \quad \text{avec} \quad i = 1, 2, \ldots, M,$$

$$\bar{\phi}_j(\tau - t + h) = (t - \tau)^{(p_g+q)}(\tau - t + h)^{(p_g+N+1-q)} \quad \text{avec} \quad i = 1, 2, \ldots, N,$$

with h the width of the sliding integration window, M and N represent the number of modulating functions for estimating the states and the unknown inputs respectively. $p_x, p_g \in \mathbb{N}^*$ are degrees of freedom.
We have chosen the basis functions to be polynomial: $\alpha^j_{k,i}(t) = t^{i-1}$, $\beta^j_q(t) = t^{q-1}$ with $i = 1, 2, ..., M$ and $q = 1, 2, ..., N$ for $k = 2, ..., n_j$, $j = 1, ..., r$.

Note that the Φ_k and Φ_{ui} matrices may present problems of matrix ill-conditioning, so regularization, such as Tikhonov regularization, [44] may be necessary. The parameters used for the estimator are taken as $h = 0.03s$, $M = 3$, $p_x = 4$, $N = 1$, and $p_g = 5$.

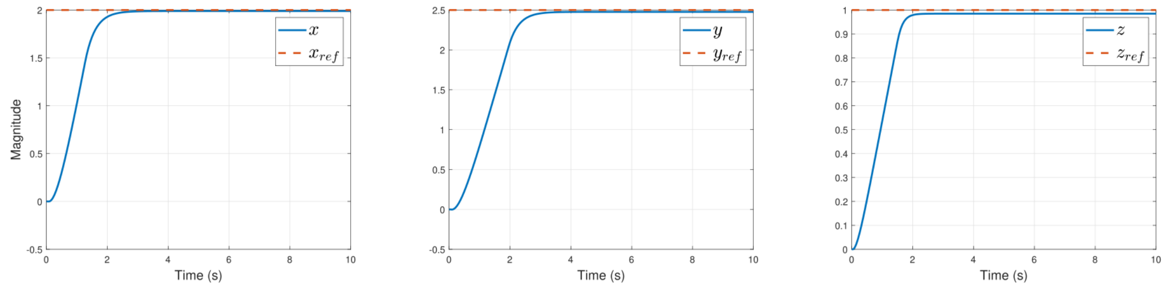

Figure 6.1: Reference tracking using a combination of sliding-mode controller and MF based observer

6.2.1.1 State estimation

Figures (6.2) and (6.3) represent respectively linear and angular velocity estimates by the MF based method in a deterministic environment. These figures show that the observer developed in section 3.6 provides an accurate estimate of linear and angular velocities, without any information on the initial conditions of the system. However, it should be noted that the parameters must be set correctly for the observer to work properly. As for Tikhonov's regularization, it mainly consists of adding a small positive ϵ to improve the conditioning of the matrices to be inverted at each iteration. The larger ϵ gets, the less precise the estimates are, since we are in fact slightly modifying the algebraic system in order to find a numerically stable solution in the vicinity of the real solution, so there's a trade-off between numerical stability and precision.

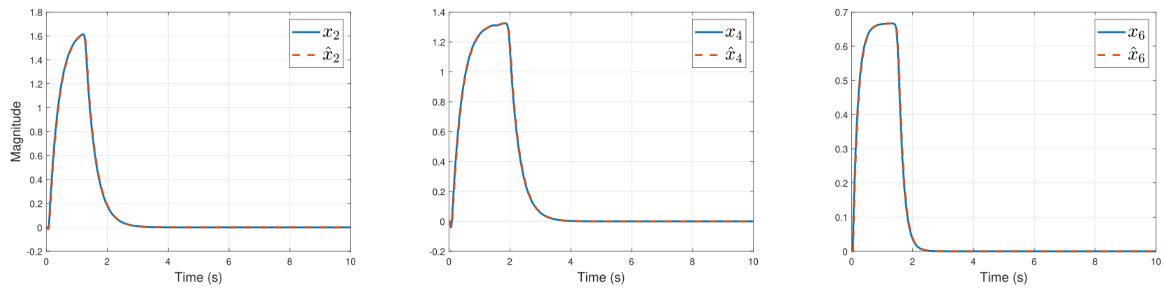

Figure 6.2: MF based method estimates of the linear velocities with their actual values in a closed-loop

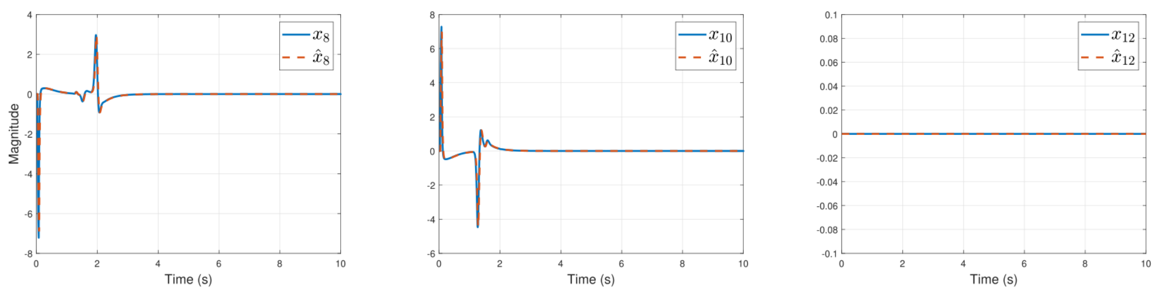

Figure 6.3: MF based method estimates of the angular velocities with their actual values in a closed-loop

Robustness test against measurement noise :

In order to test the robustness of the proposed observer, we add different levels of high-frequency normally distributed Gaussian noise to the y measurements with which we feed the observer, the table (6.1) represents the relative errors of the linear velocity estimates respectively for each measurement noise level in (%).

Estimates of linear velocities and angular velocities in a closed loop (the controller is fed with estimates from the observer) in the presence of gaussian noise of high frequency of 10% are shown in figures (6.4) and (6.5) respectively, while the open-loop estimates (the controller is fed with the real state values) are shown in figures (6.6) and (6.7). It can be seen that even in the presence of high-level measurement noise, the MF based observer was able to provide good state estimation. However, it can be seen that the estimation of angular velocities is not as good as linear velocities when combining the observer with the controller. The presence of measurement noise led to the appearance of the chattering phenomenon, which the observer was unable to completely reconstruct, due to the fact that it is high-frequency, the integration had the effect of smoothing the signal.

Table 6.1: Linear velocities' estimation relative error w.r.t different noise levels

Noise level (%)	$\frac{\lvert x_2-\hat{x}_2\rvert}{\lvert x_2\rvert}\times 100$	$\frac{\lvert x_4-\hat{x}_4\rvert}{\lvert x_4\rvert}\times 100$	$\frac{\lvert x_6-\hat{x}_6\rvert}{\lvert x_6\rvert}\times 100$
0 %	2.52	1.69	3.65
1 %	2.53	1.71	3.67
3 %	2.6	1.75	3.72
5 %	2.65	1.8	3.79
10 %	2.78	1.93	3.91

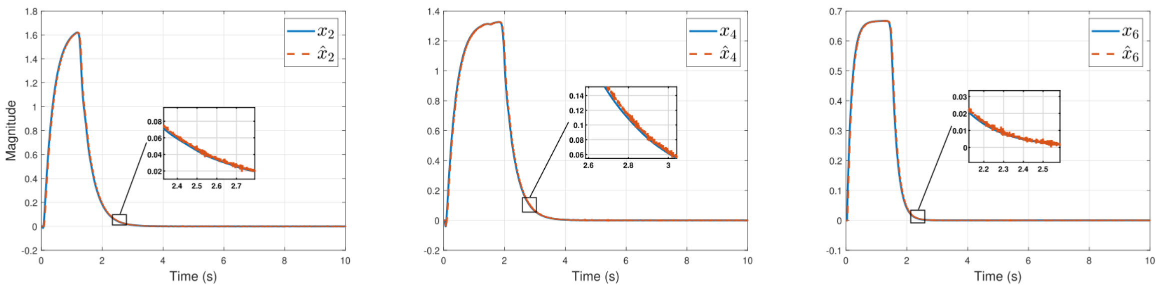

Figure 6.4: MF based method estimates of the linear velocities with their true values in closed-loop with the presence of 5% measurement noise.

6.2.1.2 Unknown inputs estimation

To estimate the unknown inputs, we saw in section 3.6 that two approaches were possible. The first is to replace the states $x_{n_j,j}$ by their previously found estimates $\hat{x}_{n_j,j}$, we replace in (3.99) by

$$\langle \phi, \dot{x}_{n_j,j}\rangle_I = \langle \dot{\phi}, \hat{x}_{n_j,j}\rangle_I$$

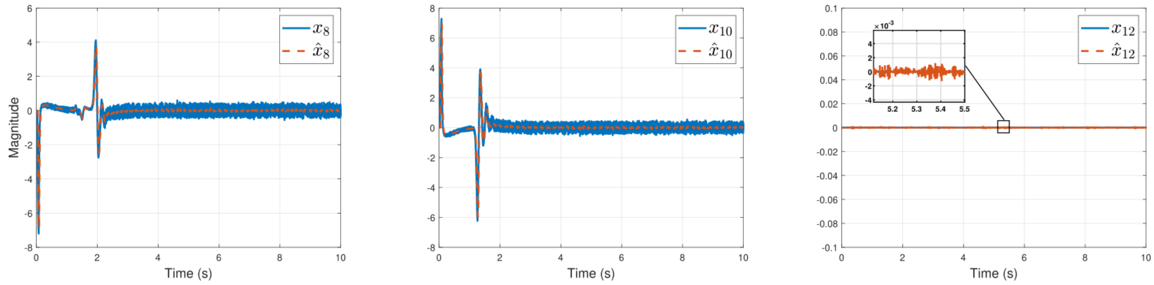

Figure 6.5: MF based method estimates of the angular velocities with their true values in closed-loop with the presence of 5% measurement noise.

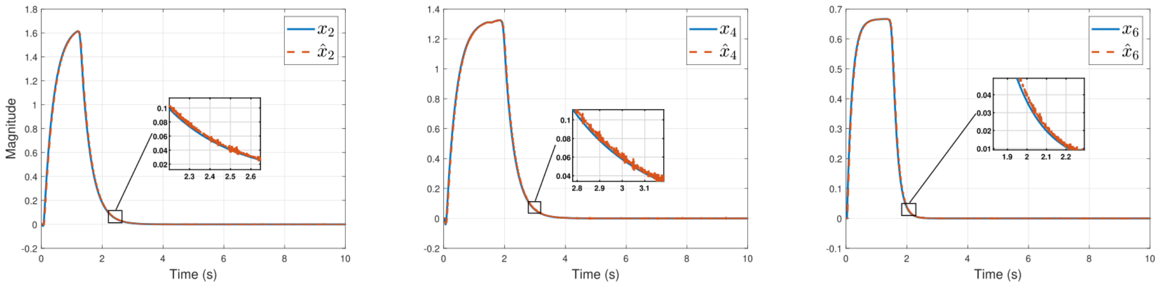

Figure 6.6: MF based method estimates of the linear velocities with their true values in open-loop with the presence of 5% measurement noise.

while the second approach consists of replacing the states $x_{n_j,j}$ by the derivatives of the outputs $x_{1,j}^{(n_j-1)}$, we replace in (3.99) by

$$\langle \phi, \dot{x}_{n_j,j} \rangle_I = \langle \phi, x_{1,j}^{(n_j)} \rangle_I = (-1)^{(n_j)} \langle \phi^{(n_j)}, x_{1,j} \rangle_I$$

So we have two choices. In the case of noisy measurements, the first approach seems to be better than the second, because when estimating the unknown inputs, we replace with state estimates that are less affected by noise (noise is filtered out during state estimation), and noise will be filtered out a second time by integration when estimating the unknown inputs. The above is illustrated by the figures (6.11)-(6.14) where these represent estimates of the acceleration of the environment along the x axis a_x for different levels of measurement noise using the two approaches mentioned above. We can see that choice 1 is more robust than choice 2, thanks to the double integration, which filters the measurement noise twice.

Figures (6.8) and (6.9) represent estimates of the environment's accelerations along the three axes using the first and second approach respectively. We can see that the observer is able to estimate their actual values. However, we can see that there are peaks due to the virtual control signals u_x and u_y and the real control signal u_1, which present sudden and abrupt variations affecting the estimates of the unknown inputs (see figure (6.10)). The peaks of the acceleration estimates a_x, a_y and a_z align perfectly with the sudden variations in u_x, u_y and u_1 respectively.

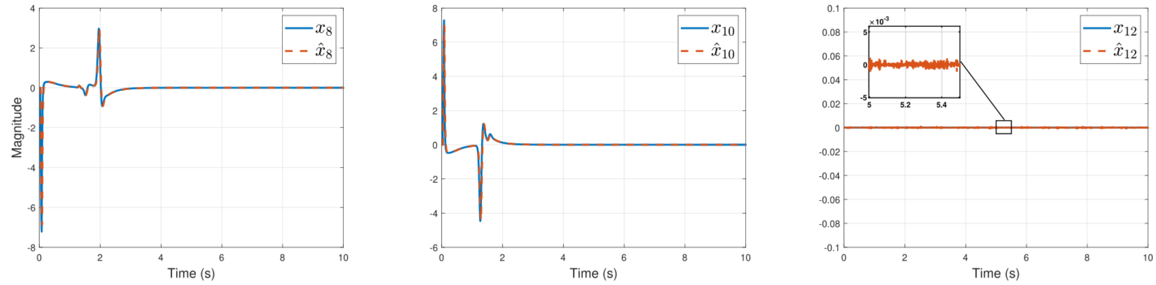

Figure 6.7: MF based method estimates of the angular velocities with their true values in open-loop with the presence of 5% measurement noise.

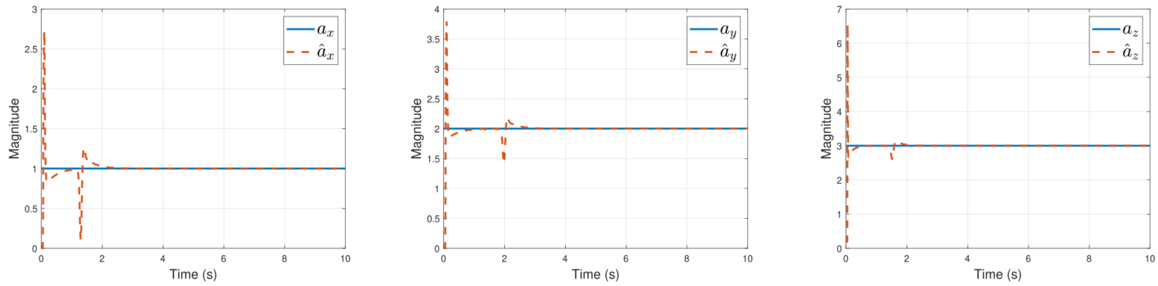

Figure 6.8: MF based method estimates of the environment accelerations (Choice 1) with their actual values

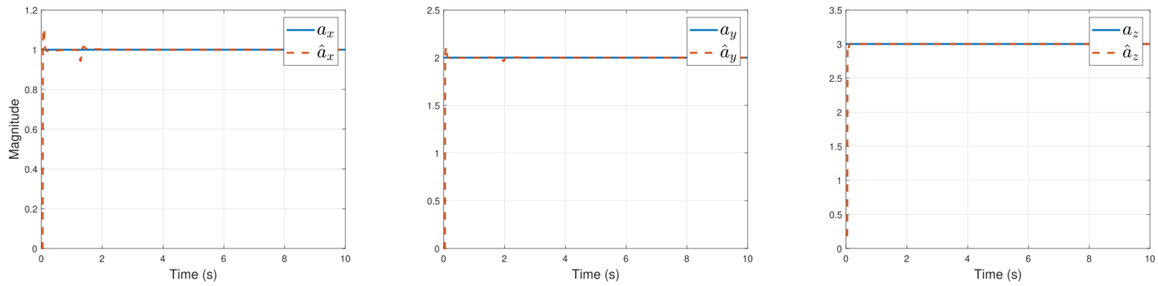

Figure 6.9: MF based method estimates of the environment accelerations (Choice 2) with their actual values

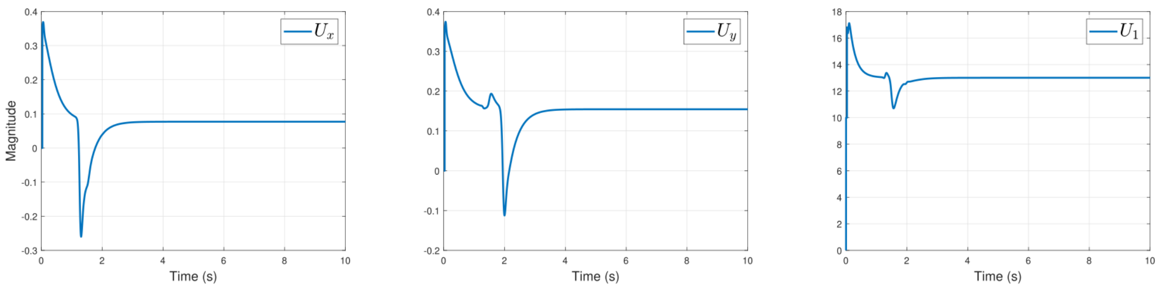

Figure 6.10: Virtual control signals and the real control signal u_1

Choice 1 **Choice 2**

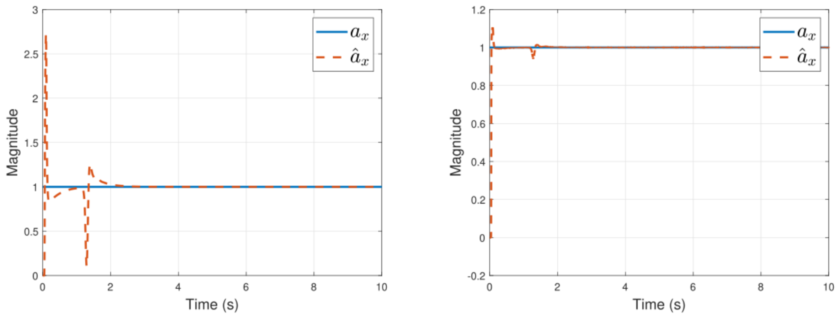

Figure 6.11: Noise level of 1%

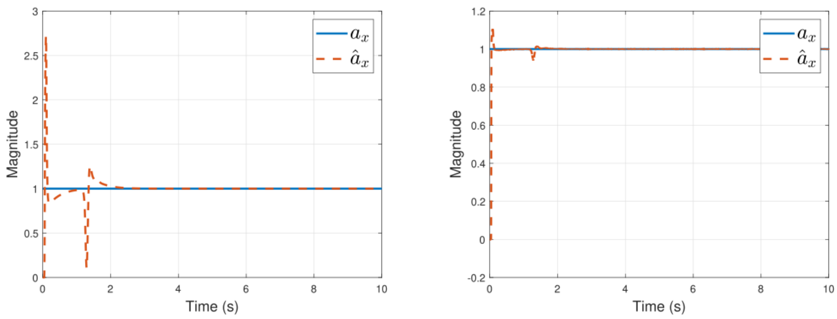

Figure 6.12: Noise level of 3%

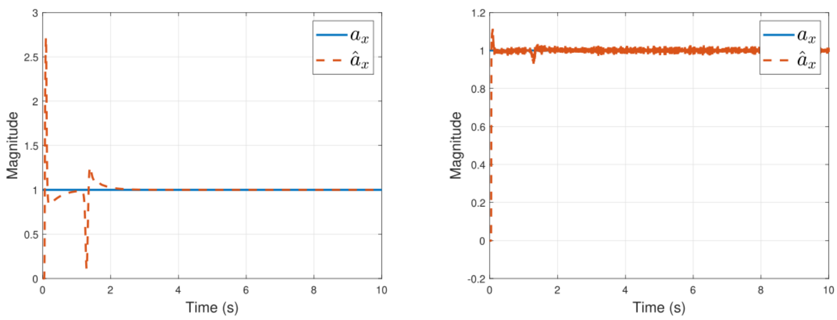

Figure 6.13: Noise level of 5%

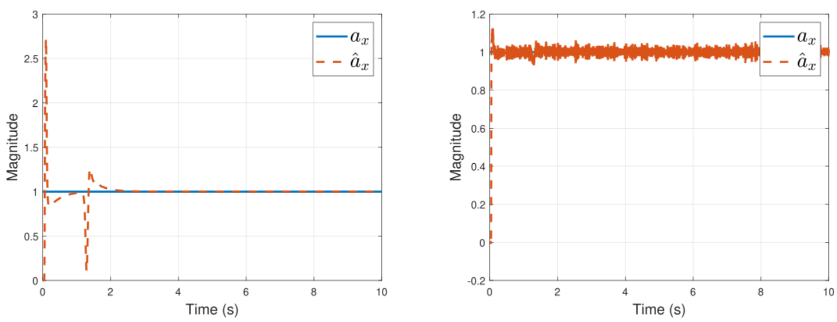

Figure 6.14: Noise level of 10%

6.3 Comparison with other observers

The estimation of states and unknown inputs with modulating functions seems to be giving very satisfactory results. However, to better appreciate the performance of this new method, we propose to carry out a comparison with other, better-known and more widely-used classical observers, and thus come out with some constructive conclusions as to which would be the most suitable observer w.r.t specific criterion

6.3.1 Extended Kalman filter with unknown inputs (EKF-UI)

The Kalman filter is one of the best-known and most widely used observers in automatic control and signal processing. It has the advantage of being both simple and efficient. It can be considered as a reference observer if you want to make comparisons and get an idea of whether a given observer is performing well or not.

The extended Kalman filter is applied according to the following methodology.

Consider the following non-linear system

$$\begin{cases} \dot{x} = f(x, u, w, t), \\ y = h(x, v, t), \end{cases} \tag{6.2}$$

where $x \in \mathbb{R}^n$ is the state vector, $u \in \mathbb{R}^m$ is the control input vector, w and v are gaussian variables with a mean of zero and covariance Q and R respectively. These two terms represent process (internal) noise and measurement noise respectively. The functions f and h are non-linear.

- We start by evaluating the following matrices at the current state estimate:

$$A = \left(\frac{\partial f}{\partial x}\right)_{x=\hat{x}}, \quad L = \left(\frac{\partial f}{\partial w}\right)_{x=\hat{x}}, \quad C = \left(\frac{\partial h}{\partial x}\right)_{x=\hat{x}}, \quad M = \left(\frac{\partial h}{\partial v}\right)_{x=\hat{x}}. \tag{6.3}$$

- We calculate $\tilde{Q}$ and $\tilde{M}$ as follows :

$$\tilde{Q} = LQL^T, \quad \tilde{R} = MRM^T. \tag{6.4}$$

- We can apply the Kalman filter equations give by :

$$\begin{cases} \hat{x}(0) = E[x(0)], \\ P(0) = E[(x(0) - \hat{x}(0))(x(0) - x\hat{(0)})^T], \\ \dot{\hat{x}} = f(\hat{x}, u, w_0, t) + K[y - h(\hat{x}, v_0, t)], \\ K = PC^T\tilde{R}^{-1}, \\ \dot{P} = AP + PA^T + \tilde{Q} - PC^T\tilde{R}^{-1}CP, \end{cases} \tag{6.5}$$

with $w_0 = 0$ and $v_0 = 0$ representing the nominal noise values and $E[.]$ is the mathematical expectation.

This is the basic Kalman filter algorithm, which has been generalized for application to nonlinear systems. There have been subsequent changes and modifications to optimize it, but this form remains the most widely used. The Kalman Filter can also be used to determine unknown inputs, using an augmented state containing the state of the system and the unknown inputs to be estimated in real time. However, the method remains the roughly same, more details are presented through its application to the drone problem.

Implementation to the drone :
The Extended kalman filter with unknown inputs (EKF-UI) has already been implemented as part of the problem of controlling the drone in a non-inertial reference frame in [49], but considering the angular velocities as being measured to directly by the sensors. However, in our case, we have considered that only the drone's orientation and position can be measured, but this doesn't change anything regarding the synthesis of the EKF-UI. In what follows, we present the EKF-UI designd in [49]. The drone model (5.3) can be rewritten in the following form

$$\begin{cases} \dot{x} = A_0 x(t) + B(y)u(t) + f(x) + Ea(t) + w(t), \\ y = C_0 x + v(t), \end{cases} \tag{6.6}$$

where $a(t) = [a_x, a_y, a_z]^T$ is the unknown term, w and v are gaussian variables with a mean of zero and covariance Q and R respectively. The matrices $A, B(y), C$ and E and the vector field $f(x)$ are given by

$$A_0 = \begin{cases} a_{ij} = 1, & j = i+1 \text{ avec } i = 1,3,5,7,9,11, \\ a_{ij} = 0, & \text{sinon.} \end{cases}$$

$$B(y) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ c(x_7)s(x_9)c(x_{11}) + s(x_7)s(x_{11}) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ c(x_7)s(x_9)s(x_{11}) - s(x_7)c(x_{11}) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ c(x_7)c(x_9) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

$$E = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1 \\ \hline & 6\times 3 & \end{bmatrix}, \quad f(x) = \begin{bmatrix} 6\times 3 \\ \hline 0 \\ \mathrm{a}x_{10}x_{12} \\ 0 \\ \mathrm{b}x_8 x_{12} \\ 0 \\ \mathrm{a}x_8 x_{10} \end{bmatrix}.$$

$$C_0 = \left[\begin{array}{cccccc|c} 1 & 0 & 0 & 0 & 0 & 0 & \\ 0 & 0 & 1 & 0 & 0 & 0 & 3\times 6 \\ 0 & 0 & 0 & 0 & 1 & 0 & \\ \hline & & & 6 & & & I_6 \end{array}\right].$$

As explained above, we augment the system, such that the augmented state is given by $\xi(t) = [x(t) \quad a(t)]^T$. The state representation thus becomes :

$$\begin{cases} \dot{\xi} = \begin{pmatrix} A_0 & E \\ 0 & 0 \end{pmatrix} \xi + \begin{pmatrix} B(y) \\ 0 \end{pmatrix} u(t) + \begin{pmatrix} f(x) \\ 0 \end{pmatrix} + \begin{pmatrix} w(t) \\ w_a(t) \end{pmatrix} \\ y = \begin{pmatrix} C_0 & 0_{6\times 3} \end{pmatrix} \xi + v(t) \end{cases} \tag{6.7}$$

where $\omega_a(t)$ is the noise affecting the accelerations. The augmented system can be written in the following compact form

$$\begin{cases} \dot{\xi} = A_\xi \xi + B_\xi(y)u(t) + f_\xi(\xi) + w_\xi(t), \\ y = C_\xi \xi + v(t), \end{cases} \tag{6.8}$$

The EKF-UI associated with the system (6.8) is given by

$$\begin{cases} \dot{\hat{\xi}} = A_\xi \hat{\xi} + B_\xi(y)u(t) + f_\xi(\xi) + K(y - C_\xi \hat{\xi}), \\ K = PC^T R^{-1}, \\ \dot{P} = AP + PA^T + Q - PC^T R^{-1} CP. \end{cases} \tag{6.9}$$

The matrices A,C,$\tilde{Q}$ and $\tilde{R}$ are calculated by 6.3, 6.4 and 6.5. In this case $L = M = 1$, which implies that $\tilde{Q} = Q$ and $\tilde{R} = R$.

6.3.2 Super-twisting sliding-mode observer

The sliding mode observer has the advantage of being a finite-time observer, as well as being robust to modeling error, measurement noise and disturbance noise. It can therefore provide accurate state estimation even under difficult conditions, making it a popular method for applications requiring reliable state estimation. It would also be interesting to compare the MF based observer with another finite-time observer.

Consider a system of the following form :

$$\begin{cases} \dot{x}_1 = x_2, \\ \dot{x}_2 = f(t, x_1, x_2, u) + \varepsilon(t, x_1, x_2, u), \\ y = x_1, \end{cases} \tag{6.10}$$

with f and ε (which is the uncertainty term) are assumed to be uniformly bounded and x_1 is assumed to be measurable.

The super-twisting sliding mode observer associated with the proposed (6.10) system takes the following form [21]

$$\begin{cases} \dot{\hat{x}}_1 = \hat{x}_2 + z_1, \\ \dot{\hat{x}}_2 = f(t, x_1, \hat{x}_2, u) + z_2, \end{cases} \tag{6.11}$$

where $\hat{x_1}$ and $\hat{x_2}$ are the estimated states, and z_1 and z_2 are the correction terms given by :

$$\begin{cases} \hat{z}_1 = \lambda |x_1 - \hat{x}_1|^{1/2} sign(x_1 - \hat{x}_1), \\ \hat{z}_2 = \alpha . sign(x_1 - \hat{x}_1). \end{cases} \tag{6.12}$$

The estimation error is defined by $e = x - \hat{x}$, its dynamics is given by

$$\begin{cases} \dot{e}_1 = e_2 - \lambda |e_1|^{1/2} sign(e_1), \\ \dot{e}_2 = F(t, x_1, x_2, \hat{x}_2, u) - \alpha sign(e_1), \end{cases} \tag{6.13}$$

where $F(t, x_1, x_2, \hat{x}_2, u) = f(t, x_1, x_2, u) - f(t, x_1, \hat{x}_2, u) + \varepsilon(t, x_1, x_2, u)$. If we assume that the system states are bounded, then the existence of a constant f^+ is assured, such that f^+ verifies the following inequality

$$|F(t, x_1, x_2, \hat{x}_2, u)| < f^+, \tag{6.14}$$

for any t, x_1, x_2 and $|\hat{x}_2| \leqslant 2sup|x_2|$. With α and λ satisfying parameters:

$$\alpha > f^+, \tag{6.15}$$

$$\lambda > \sqrt{\frac{2}{\alpha - f^+} \frac{(\alpha + f^+)(1 + p)}{(1 - p)}}, \tag{6.16}$$

where p is a constant chosen such that $0 < p < 1$.

Implementation to the drone :
The drone model is made up of six blocks of the form described in (6.10), with the slight nuance that the functions f for the different blocks are not a function of the two states of the block only, but include other states. For the first three blocks, these states are measured, so their exact values are known, unlike for the last three blocks, where they are considered part of the uncertainty term.

$$\begin{cases} \dot{x}_1 = x_2, \\ \dot{x}_2 = [c(x_7)s(x_9)c(x_{11}) + s(x_7)s(x_{11})]u_1 - a_x, \end{cases}$$

where $f_1(x, u) = [c(x_7)s(x_9)c(x_{11}) + s(x_7)s(x_{11})]u_1$ and $\varepsilon_1(t, x, u) = -a_x$.

$$\begin{cases} \dot{x}_3 = x_4, \\ \dot{x}_4 = [c(x_7)s(x_9)s(x_{11}) - s(x_7)c(x_{11})]u_1 - a_y, \end{cases}$$

where $f_2(x, u) = [c(x_7)s(x_9)s(x_{11}) - s(x_7)c(x_{11})]u_1$ and $\varepsilon_2(t, x, u) = -a_y$.

$$\begin{cases} \dot{x}_5 = x_6, \\ \dot{x}_6 = [c(x_7)c(x_9)]u_1 - g - a_z, \end{cases}$$

where $f_3(x, u) = [c(x_7)c(x_9)]u_1 - g$ and $\varepsilon_3(t, x, u) = -a_z$.

$$\begin{cases} \dot{x}_7 = x_8, \\ \dot{x}_8 = a x_{10} x_{12} + u_2, \end{cases}$$

where $f_4(x, u) = u_2$ and $\varepsilon_4(t, x, u) = a x_{10} x_{12}$.

$$\begin{cases} \dot{x}_9 = x_{10}, \\ \dot{x}_{10} = b x_8 x_{12} + u_3, \end{cases}$$

where $f_5(x,u) = u_3$ and $\varepsilon_5(t,x,u) = bx_8x_{12}$.

$$\begin{cases} \dot{x}_{11} = x_{12}, \\ \dot{x}_{12} = cx_8x_{10} + u_4, \end{cases}$$

where $f_6(x,u) = u_4$ and $\varepsilon_6(t,x,u) = cx_8x_{10}$.

The observer based on the twisting Sliding Mode associated with the drone will therefore be as follows

$$\begin{cases} \dot{\hat{x}}_{2i-1} = \hat{x}_{2i} + z_{2i-1}, \\ \dot{\hat{x}}_{2i} = f_{2i-1}(t, x_{2i-1}, \hat{x}_{2i}, u) + z_{2i}, \end{cases} \Rightarrow \begin{cases} \hat{z}_{2i-1} = \lambda_i |x_{2i-1} - \hat{x}_{2i-1}|^{1/2} sign(x_{2i-1} - \hat{x}_{2i-1}), \\ \hat{z}_{2i} = \alpha_i sign(x_{2i-1} - \hat{x}_{2i-1}), \end{cases} \tag{6.17}$$

avec $i \in 1, 2, \ldots, 6$.

6.3.3 Comparison

The three observers are implemented in closed loop with the sliding mode controller : The MF based observer, the Twisting sliding mode observer and the EKF-UI. The Twisting sliding mode observer, unlike the other two observers, is unable to estimate unknown inputs, but remains robust to disturbances even without any knowledge. about them For this comparison, we will take into account several qualitative criteria such as : convergence speed, numerical stability, robustness, etc.

The system and the MF based observer are simulated under the same conditions (parameters, etc.) as in the previous section. The initial conditions of the EKF-UI are taken as $\hat{\xi}_0 = \begin{bmatrix} 0.5 \times \text{ones}(1,12) & \text{zeros}(1,3) \end{bmatrix}^T$, the covariance matrices of the measurement noise and process noise are taken equal to $Q = \begin{pmatrix} 5I_12 & 0_{12\times3} \\ 0_{3\times12} & 50I_3 \end{pmatrix}$ and $R = 1$. While the initial conditions of the Super twisting sliding mode are fixed at $x_0 = \begin{bmatrix} 1.5 \times \text{ones}(1,12) \end{bmatrix}^T$.

We've added a 5% gaussian noise to the measurements y , in order to test the robustness of the observers to noise affecting the measurements. Figures (6.15), (6.16) and (6.17) represent the evolution of the UAV's linear position by implementing the MF based method, EKF-UI and the super twisting sliding mode observer respectively. It's clear that the MF based method observer gives the best results, and if we compare figure (6.15) with figure ((6.15)), which represents the situation where we know exactly the instantaneous values of each state, we can see that the dynamical performances are very similar, unlike the results obtained with EKF-UI and the super twisting sliding mode, where the dynamical performances were more or less affected, negatively impacting response time.

The figure (6.19) illustrates the estimates obtained from each observer. It can be seen that the estimates converge instantaneously in the case of the MF based observer, followed by the super twisting sliding mode observer, which requires a short period of time to converge. Finally, the estimates obtained by the EKF-UI are the slowest to converge, which explains the deterioration in the system's dynamical performances. It should be noted that the speed of convergence of the last two is very sensitive to the initial conditions of the observer, which can even leads to instability of the observer, whereas the MF based observer does not require knowledge of the initial conditions at all.

From the figure (6.18) illustrating estimates of unknown inputs by MF based method and EKF-UI, the same observation is made, estimates by MF based method converge instantaneously while EKF-UI is much slower, however EKF-UI is numerically stable compared to MFBM. In terms of robustness to measurement noise, all three observers are effective at attenuating it, although we note that the MF based observer is slightly less robust. In fact, the wider the sliding window, the better the observer's ability to attenuate low-frequency noise, but this will make the calculations more cumbersome, so there's a trade-off between computational load and robustness. The table (6.2) below summarizes the qualitative comparison between these three observers according to different criteria.

Table 6.2: Qualitative comparison between the three observers implemented

Criteria	Modulating functions	Twisting Sliding Mode	EKF-UI
Nature	Algebraic	Dynamic	Dynamic
Convergence speed	++	+	-
Dynamical performances	++	+	-
Robustness	+	++	++
Initial conditions sensitivity	++	- -	- -
Numerical stability	-	++	++
Parameter sensitivity	-	-	- -

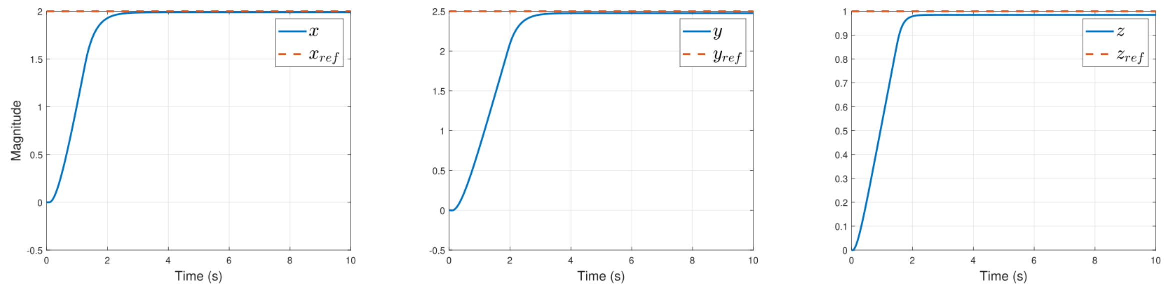

Figure 6.15: Reference tracking using a combination of a sliding-mode controller and the MF based observer

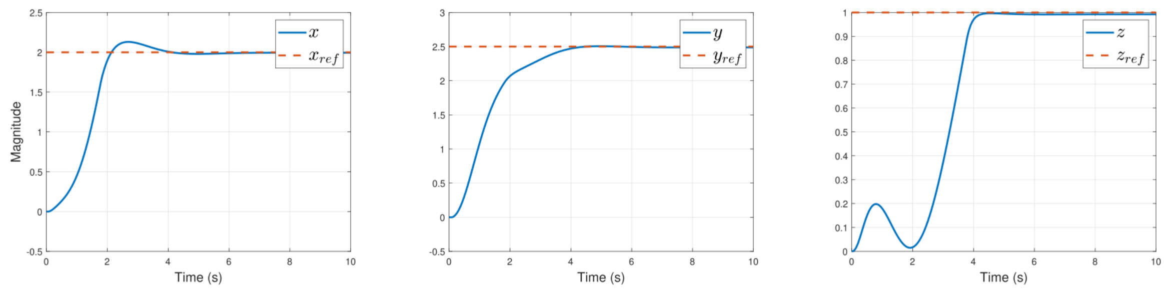

Figure 6.16: Reference tracking using a combination of a sliding-mode controller and the EKF-UI

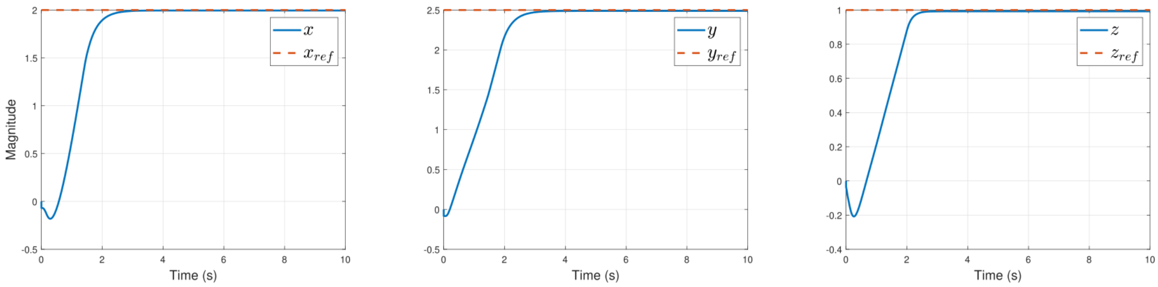

Figure 6.17: Reference tracking using a combination of a sliding-mode controller and the super twisting sliding mode

6.4 Conclusion

In this chapter, we've applied the MF based estimator to the problem of trajectory tracking of an UAV in a moving reference frame. We carried out numerical simulations to estimate the system states and the acceleration of the non-inertial reference frame, using the combination of the MF based observer and a sliding mode controller. We then carried out a qualitative comparison between the MFBM estimator and other estimators widely used in the automatic control and signal processing literature. The aim was to demonstrate the performance of the MF based estimator compared with these other methods, as well as to determine which observer to choose in different situations, taking into account their respective advantages and disadvantages. The comparison revealed that the observer based on modulating functions has significant advantages over the other two observers tested, apart from the numerical instabilities it may present. In particular, the convergence of the estimator is almost instantaneous, and the fact that it does not require knowledge or adjustment of the initial conditions of the estimator thanks to the properties of the modulating functions.

It's important to note that MFBM estimation is non-asymptotic, as it provides an accurate estimate after a finite time. So the method is at least finite time stable, but it could be even more efficient beings fixed time stable or prescribed time stable, but nothing has been proven yet. This represents a very interesting direction for research, we could also think of improving the method by making it numerically stable.

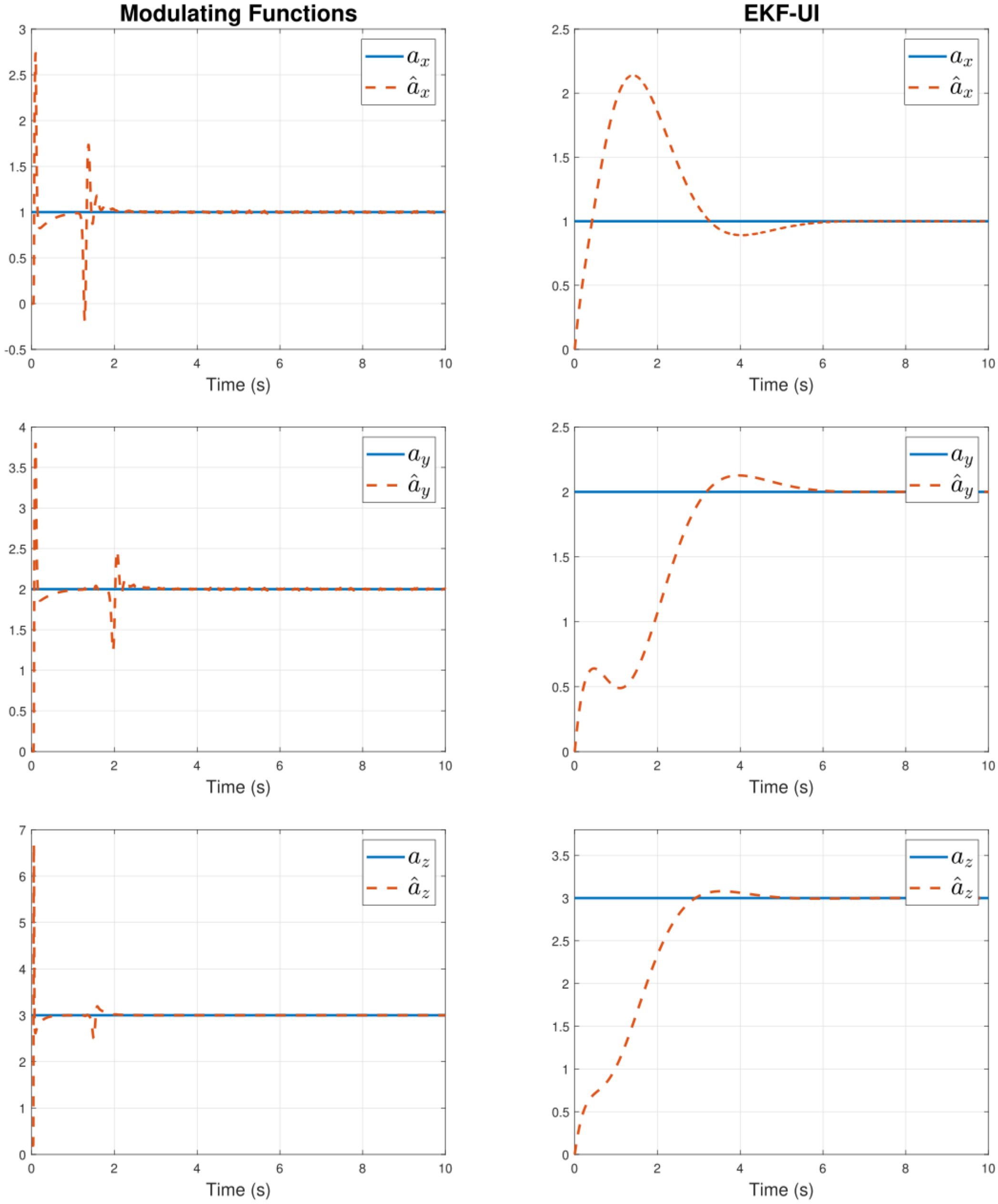

Figure 6.18: Comparison of the acceleration's estimates by the MF based observer and EKF-UI

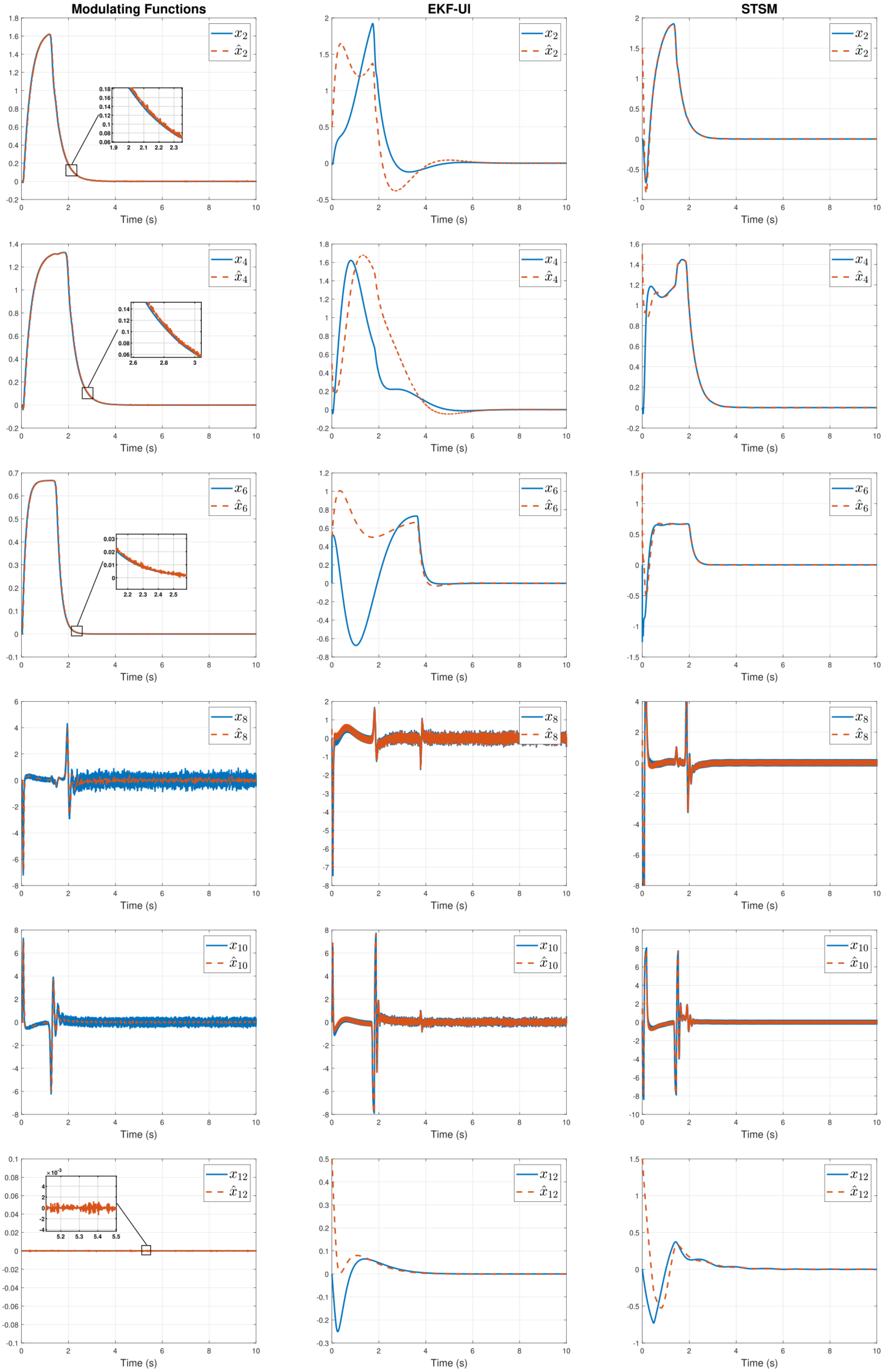

Figure 6.19: Comparison of the linear and angular velocities's estimates by the MF based observer, EKF-EI and the super twisting sliding mode observer

General conclusion

In this project, we were confronted with a concept that was initially unknown to us, namely non-asymptotic stability in the field of observation and estimation. In the first chapter, we were able to appreciate the difference, and even the superiority, of different types of non-asymptotic stability compared with stability in the Lyapunov sense. However, in Chapter 2, we quickly realized why this class of observers has not attracted the same interest from researchers as those with asymptotic convergence. It turns out that to ensure the stability of the estimation error, it is often necessary to resort to non-smooth techniques or to employ terms tending towards infinity, which makes the study and development of these techniques often very complex. Nevertheless, this field has been booming in recent years, and will only increase in popularity with control-theory researchers and engineers in the years to come, thanks to technical advances and the growing need in many applications and fields to ensure convergence in a non-asymptotic fashion.

In this work, we focused on two non-asymptotic observers, one using an algebraic method (MFBM observer) and the other using a specific time-varying base change (prescribed-time observer). However, we focused on the observer based on modulating functions. We introduced it by first defining what a modulating function is and its various properties, on which MFBM is based. To help the reader understand the method, we began by presenting the MFBM in the context of parameter estimation (constant and time-varying) of systems governed by ordinary differential equations, providing numerical examples to illustrate the method's performance. We then moved on to the chore of the matter, presenting the application of MFBM to the estimation of state and unknown inputs of nonlinear triangular canonical systems, both online and offline, accompanied by a numerical example. Throughout this chapter, we have been able to appreciate the advantages that this method brings. We found that, thanks to the properties of the modulating functions, we could estimate the variables of interest (state, unknown inputs and parameters) without having to initialize the estimator, making this observer totally insensitive to initial conditions. Furthermore, we found out that modulating functions allow us to transfer the derivative from an experimental signal that may be subject to measurement noise to the modulating function, whose analytical expression is known. Finally, MFBM is robust to measurement noise. MFBM is a very powerful method of estimation that transforms an estimation problem into a problem of finding the solution of a simple linear algebraic system, without having to resort to non-smooth techniques or terms tending towards infinity.

Having mastered this method, we ended the chapter by presenting our contributions to the MFBM estimation problem, which consists of an extension of the unknown-input observer based on modulating functions to a new class of nonlinear triangular block MIMO systems, and the improvement of the robustness of the estimation of unknown inputs.

We then proceeded to the study of an existing observer that converge in a user-prescribed time T, independently of initial conditions and system parameters, for well-defined linear and nonlinear systems. By illustrating the performance of these observers through simple examples, we found out that this type of observer does indeed converge in a prescribed time. However, it uses time-varying gains that explode to infinity as time approaches the prescribed time T, which considerably limits its practicality. Furthermore, this type of observer is highly sensitive to measurement noise when approaching the prescribed time, due to gains that tend towards infinity. It's also restrained to a specific class of systems

Finally, our work concludes with an application of the proposed extension of the MFBM, we introduce the problem of controlling the drone's position in a moving environment, with access only to relative positions. It is important to note that this problem was recently proposed in [49]. In Chapter 5, we dealt with the control part of the problem where we designed two controllers, namely the sliding mode controller and the backstepping controller. Both commands produced satisfactory results. Knowing that the sliding mode control proved effective in the experiments carried out in [49], we decided to continue with the sliding mode controller. Having designed a controller for the UAV, it is now necessary to implement an observer to estimate the states in order to be able to calculate the control laws in question. For this reason, in Chapter 6, we dealt with the estimation part of the problem. We implemented the extension we proposed in Chapter 3 on the UAV. First, we presented numerical simulations of state estimation and the acceleration of the non-inertial reference frame as an unknown input, combining the MFBM observer and sliding mode controller. We then carried out a sort of qualitative comparison between this estimator and other well-known estimators widely used in the automatic control and signal processing literature. The idea was to illustrate the performance of modulating functions by having a reference and also to know which observer to choose for different situations, knowing the advantages and disadvantages of each observer. The comparison revealed that the observer based on modulating functions has significant advantages over the other two observers tested, apart from the numerical instabilities it may present. In particular, the near-instantaneous convergence of the estimates, and the fact that it does not require knowledge or adjustment of the initial conditions of the estimates, thanks to the properties of the modulating functions, which makes this method very attractive, even when compared with observers that are highly regarded in the literature.

Perspectives

For the continuity of this work, different perspectives can be considered. We propose:

- Validate the combination of the sliding mode controller and the MFBM observers by experiments.
- Improve the numerical stability of the MFBM observer.
- Extend the UAV problem to the broader case where the non-inertial frame has both translation and rotation motion.
- Extend the prescribe time observer to a broader class of systems and improve its robustness.

Bibliography

[1] A. Adil, I. N., and Laleg-Kirati, T. M. Modulated-integral observer design with a virtual output derivative injection term. *IFAC World Congress* (2023).

[2] Abeer, A., TaousMeriem, L., and Dayan, L. A novel approach for parameter and differentiation order estimation for a space fractional advection dispersion equation. *arXiv: Optimization and Control* (2014).

[3] Amit, P., and Unbehauen, H. Identification of a class of nonlinear continuous-time systems using hartley modulating functions. *International Journal of Control 62*, 6 (1995), 1431–1451.

[4] Andrieu, V., Praly, L., and Astolfi, A. Homogeneous approximation, recursive observer design, and output feedback. *SIAM Journal on Control and Optimization 47*, 4 (2008), 1814–1850.

[5] Ania, A. *On Estimation and observer design in nonlinear systems : theory and application.* PhD thesis, Universite Mouloud Mammeri Tizi-Ouzou, 2022.

[6] Aoustin, Y., Chevallereau, C., and Orlov, Y. Finite time stabilization of a perturbed double integrator - part ii: applications to bipedal locomotion. In *49th IEEE Conference on Decision and Control (CDC)* (2010), pp. 3554–3559.

[7] Asiri, S., Elmetennani, S., and Laleg-Kirati, T.-M. Moving-horizon modulating functions-based algorithm for online source estimation in a first order hyperbolic pde. *Journal of Solar Energy Engineering 139* (08 2017).

[8] Bahloul, M. A., and Kirati, T.-M. L. Finite-time joint estimation of the arterial blood flow and the arterial windkessel parameters using modulating functions. *IFAC-PapersOnLine 53*, 2 (2020), 16286–16292. 21st IFAC World Congress.

[9] Barbot, J., Boukhobza, T., and Djemai, M. Sliding mode observer for triangular input form. *IEEE-CDC* (1996).

[10] Belkhatir, Z., and Laleg-Kirati, T.-M. Estimation of multiple point sources for linear fractional order systems using modulating functions. *IEEE Control Systems Letters 2*, 1 (2018), 7–12.

[11] Belkheiri, M., Rabhi, A., Hajjaji, A. E., and Pegard, C. Different linearization control techniques for a quadrotor system. In *CCCA12* (2012), pp. 1–6.

[12] Bhat, S. P., and Bernstein, D. S. Finite-time stability of continuous autonomous systems. *SIAM J. Control. Optim. 38* (2000), 751–766.

[13] Bouabdallah, S., and Siegwart, R. Backstepping and sliding-mode techniques applied to an indoor micro quadrotor. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation* (2005), pp. 2247–2252.

[14] BOULKROUNE, B. *Estimation de letat des systemes non lineaires à temps discret. Application a une station depuration.* PhD thesis, Universite Henri Poincare Nancy 1, 2008.

[15] Bregeault, V. *Quelques contributions à la théorie de la commande par modes glissants.* PhD thesis, Ecole Centrale de Nantes, France, 2010.

[16] Chen, G. *Approximate Kalman filtering.* 1993.

[17] Chen, X., Zhang, X., and Liu, Q. Prescribed-time decentralized regulation of uncertain nonlinear multi-agent systems via output feedback. *Systems Control Letters 137* (2020), 104640.

[18] Cherifa, B. *Stabilisation et Estimation de l'état des systèmes Dynamiques non linéaires et Applications.* PhD thesis, Universite Mouloud MAMMERI Tizi-Ouzou, 2011.

[19] Co, T. B., and Ungarala, S. Batch scheme recursive parameter estimation of continuous-time systems using the modulating functions method. *Automatica 33*, 6 (1997), 1185–1191.

[20] Commault, C., Dion, J., Sename, O., and Motyeian, R. Fault detection and isolation of structured systems. *IFAC Proceedings Volumes 33*, 11 (2000), 1157–1162. 4th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes 2000 (SAFEPROCESS 2000), Budapest, Hungary, 14-16 June 2000.

[21] Davila, J., Fridman, L., and Levant, A. Second-order sliding-mode observer for mechanical systems. *IEEE transactions on automatic control 50*, 11 (2005), 1785–1789.

[22] Djennoune, S., Bettayeb, M., and Al-Saggaf, U. M. Fixed-time convergent sliding-modes-based differentiators. *Communications in Nonlinear Science and Numerical Simulation 104* (2022), 106033.

[23] Drakunov, S., and Utkin, V. Sliding mode observers. tutorial. In *Proceedings of 1995 34th IEEE Conference on Decision and Control* (1995), vol. 4, pp. 3376–3378 vol.4.

[24] Emmanuel, T., and Thomas, H. *Cours dautomatique, Master de Mathématiques.*

[25] Engel, R., and Kreisselmeier, G. A continuous-time observer which converges in finite time. *IEEE Transactions on Automatic Control 47*, 7 (2002), 1202–1204.

[26] Esfandiari, F., and Khalil, H. Observer-based control of uncertain linear systems: Recovering state feedback robustness under matching condition. In *1989 American Control Conference* (1989), pp. 931–936.

[27] Fairman, F. W., and Shen, D. W. C. Parameter identification for a class of distributed systems. *International Journal of Control* (2007), 929940.

[28] Gauthier, J. P., and Bornard, G. Observability for any u(t) of a class of nonlinear systems. *IEEE Trans. Automatic Control 26*, 4 (1994), 922926.

[29] Giuseppe, F., and Loredana, C. A recursive scheme for frequency estimation using the modulating functions method. *Applied Mathematics and Computation 216*, 5 (2010), 1393–1400.

[30] Grewal, M., and Andrews, A. *Kalman filtering : Theory and practice*. Prentice Hall, 1993.

[31] Guo, Q. *Online identification and control of robots using algebraic differentiators*. PhD thesis, Ecole Centrale de Lille, 12 2015.

[32] Haimo, V. T. Finite time controllers. *SIAM Journal on Control and Optimization 24*, 4 (1986), 760–770.

[33] Holloway, J. C. *Prescribed Time Stabilization and Estimation for Linear Systems with Applications in Tactical Missile Guidance*. PhD thesis, TUC San Diego, 2018.

[34] Hou, M., and Muller, P. Design of observers for linear systems with unknown inputs. *IEEE Transactions on Automatic Control 37*, 6 (1992), 871–875.

[35] Jouffroy, J., and Reger, J. Finite-time simultaneous parameter and state estimation using modulating functions. In *2015 IEEE Conference on Control Applications (CCA)* (2015), pp. 394–399.

[36] Kalman.R.E. A new approach to linear filtering and prediction problems. *Transactions of the ASME Journal of Basic Engineering 82* (1960), 3545.

[37] Khalil, H. *Nonlinear Systems*. Pearson Education. Prentice Hall, 2002.

[38] Khalil, H. K. High-gain observers in nonlinear feedback control. In *2008 International Conference on Control, Automation and Systems* (2008), pp. xlvii–lvii.

[39] Korder, K., Noack, M., and Reger, J. Non-asymptotic observer design for nonlinear systems based on linearization. In *2022 IEEE 61st Conference on Decision and Control (CDC)* (2022), pp. 615–621.

[40] Lab, T. A. If you fly a drone in a car, does it move with it? (dangerous in-car flight challenge), Nov. 2017.

[41] Lab, T. A. What happens if you fly a drone in an elevator? real experiment!, Jan. 2019.

[42] Lakshmikantham, V., Leela, S., and Martynyuk, A. A. *Practical Stability of Nonlinear Systems*. WORLD SCIENTIFIC, 1990.

[43] Lam, L., and Weiss, L. Finite time stability with respect to time-varying sets. *Journal of the Franklin Institute 298*, 5 (1975), 415–421.

[44] Lenzen, F., and Scherzer, O. Tikhonov type regularization methods: History and recent progress. *Proceeding Eccomas* (2004).

[45] Liu, D.-Y., and Laleg-Kirati, T.-M. Robust fractional order differentiators using generalized modulating functions method. *Signal Processing 107* (2015), 395–406. Special Issue on ad hoc microphone arrays and wireless acoustic sensor networks Special Issue on Fractional Signal Processing and Applications.

[46] Liu, J., Sun, M., Chen, Z., and Sun, Q. Super-twisting sliding mode control for aircraft at high angle of attack based on finite-time extended state observer. *Nonlinear Dynamics 99* (03 2020).

[47] Luenberger, D. An introduction to observers. *IEEE Transactions on Automatic Control 16*, 6 (1971), 596–602.

[48] LYAPUNOV, A. M. The general problem of the stability of motion. *International Journal of Control 55*, 3 (1992), 531–534.

[49] Marani, Y., Telegenov, K., Feron, E., and Kirati, M.-T. L. Drone reference tracking in a non-inertial frame using sliding mode control based kalman filter with unknown input. In *2022 IEEE Conference on Control Technology and Applications (CCTA)* (2022), pp. 9–16.

[50] Mazenc, F., and Malisoff, M. New finite-time and fast converging observers with a single delay. *IEEE Control Systems Letters 6*, 1561–1566.

[51] Meditch, J. S., and Hostetter, G. Observers for systems with unknown and inaccessible inputs. In *IEEE Conference on Decision and Control* (1973).

[52] Moulay, E. *Une contribution à l'étude de la stabilité en temps fini et de la stabilisation.* PhD thesis, Automatique / Robotique. Ecole Centrale de Lille; Université des Sciences et Technologie de Lille - Lille I, 2005.

[53] Moulay, E., and Perruquetti, W. Finite time stability of nonlinear systems. In *42nd IEEE International Conference on Decision and Control (IEEE Cat. No.03CH37475)* (2003), vol. 4, pp. 3641–3646 vol.4.

[54] Ménard, T., Moulay, E., and Perruquetti, W. Global finite-time observers for non linear systems. In *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference* (2009), pp. 6526–6531.

[55] Müllhaupt, P. *Introduction à l'analyse et à la commande des systèmes non linéaires.* 2009.

[56] Nazari, S. The unknown input observer and its advantages with examples, 2015.

[57] Nedjmi, D. R. Commande hybride avec observation dun uav de type quadrotor. Master's thesis, Ecole Militaire Polytechnique, Alger, Algérie, 2010.

[58] Orlov, Y. Finite time stability and robust control synthesis of uncertain switched systems. *SIAM Journal on Control and Optimization 43*, 4 (2004), 1253–1271.

[59] Patton, R., and Chen, J. Observer-based fault detection and isolation: Robustness and applications. *Control Engineering Practice 5*, 5 (1997), 671–682.

[60] Pearson, A., and Lee, F. Efficient parameter identification for a class of bilinear differential systems. *IFAC Proceedings Volumes 18*, 5 (1985), 161–165.

[61] Perdreauville, F. J., and Goodson, R. E. Identification of systems described by partial differential equations. *J. Basic Eng 88*, 2 (1966), 463468.

[62] Perruquetti, W. Non-asymptotic output feedback of a double integrator: a separation principle, 2022.

[63] Perruquetti, W., Floquet, T., and Moulay, E. Finite-time observers: Application to secure communication. *IEEE Transactions on Automatic Control 53*, 1 (2008), 356–360.

[64] Perruquetti, W., Floquet, T., and Moulay, E. Finite-time observers: Application to secure communication. *IEEE Transactions on Automatic Control 53*, 1 (2008), 356–360.

[65] Plestan, F., Grizzle, J., Westervelt, E., and Abba, G. Stable walking of a 7-dof biped robot. *IEEE Transactions on Robotics and Automation 19*, 4 (2003), 653–668.

[66] Plestan, F., Grizzle, J., Westervelt, E., and Abba, G. Stable walking of a 7-dof biped robot. *IEEE Transactions on Robotics and Automation 19*, 4 (2003), 653–668.

[67] Polyakov, A. Nonlinear feedback design for fixed-time stabilization of linear control systems. *IEEE Transactions on Automatic Control 57*, 8 (2012), 2106–2110.

[68] Preisig, H., and Rippin, D. Theory and application of the modulating function methodi. review and theory of the method and theory of the spline-type modulating functions. *Computers Chemical Engineering 17*, 1 (1993), 1–16.

[69] Riaz, S., Yin, C.-W., Qi, R., Li, B., Ali, S., and Shehzad, K. Design of predefined time convergent sliding mode control for a nonlinear pmlm position system. *Electronics 12*, 4 (2023).

[70] Roxin, E. On finite stability in control systems. *Rendiconti del Circolo Matematico di Palermo 15* (1966), 273–282.

[71] Saberi, A., and Sannuti, P. Observer design for loop transfer recovery and for uncertain dynamical systems. *IEEE Transactions on Automatic Control 35*, 8 (1990), 878–897.

[72] Sadabadi, M., Shafiee, M., and Karrari, M. Parameter estimation of two-dimensional linear differential systems via fourier based modulation function. *IFAC Proceedings Volumes 41*, 2 (2008), 14385–14390. 17th IFAC World Congress.

[73] Saha, D., and Rao, G. P. Identification of distributed parameter systems via multidimensional distributions. *Proc.IEE 127*, 2 (1980), 4550.

[74] SAHA, D. C., RAO, B. B. P., and RAO, G. P. Structure and parameter identification in linear continuous lumped systems the poisson moment functional approach. *International Journal of Control 36*, 3 (1982), 477–491.

[75] Sharefa, M. A. *Modulating Function-Based Method for Parameter and Source Estimation of Partial Differential Equations.* PhD thesis, King Abdullah University of Science and Technology Thuwal, Kingdom of Saudi Arabia, 2017.

[76] Shen, Y., and Xia, X. Semi-global finite-time observers for nonlinear systems. *Automatica 44*, 12 (2008), 3152–3156.

[77] Shinbrot, M. On the analysis of linear and nonlinear dynamical systems from transient-response data. *NACA Technical Notes* (1954).

[78] Shinbrot, M. On the analysis of linear and nonlinear systems. *Trans. ASME* (1957), 547552.

[79] Slotine, J.-J. E. *Applied nonlinear control.* Prentice Hall, 1991.

[80] Sulaiman, M., Patakor, F. A., and Ibrahim, Z. New methodology for chattering suppression of sliding mode control for three-phase induction motor drives.

[81] Sánchez-Torres, J. D., Sanchez, E. N., and Loukianov, A. G. Predefined-time stability of dynamical systems with sliding modes. In *2015 American Control Conference (ACC)* (2015), pp. 5842–5846.

[82] Takaya, K. The use of hermite functions for system identification. *IEEE Transactions on Automatic Control 13*, 4 (1968), 446–447.

[83] Tang, B., and Brennan, M. Comparison of two nonlinear damping mechanisms in a vibration isolator. *A.Journal of Sound and Vibration 332*, 3 (2013), 510520.

[84] Utkin, V. *Sliding modes in control and optimization.* 1992.

[85] Wang, Q., Dong, X., Wang, B., Hua, Y., and Ren, Z. Finite-time observer-based h fault-tolerant output formation tracking control for heterogeneous nonlinear multi-agent systems. *IEEE Transactions on Network Science and Engineering*.

[86] Xu, R., and Zhou, M. Sliding mode control with sigmoid function for the motion tracking control of the piezo-actuated stages. *Electronics Letters 53*, 2 (2017), 75–77.

[87] Yasmine, M., Ibrahima, N., and Taous Meriem, L.-K. Non-asymptotic neural network-based state and disturbance estimation for a class of nonlinear systems using modulating functions. In *2023 American Control Conference* (2023).

[88] Yuri, S., Christopher, E., Leonid, F., and Arie, L. Sliding mode control and observation. *Birkhauser* (2014).

[89] Zemouche, A. *Sur lobservation de létat des systèmes dynamiques non linéaires.* PhD thesis, Université Louis Pasteur Strasbourg I, March 2007.

[90] Zoghlami, N. *Stabilite et stabilisation en temps fini des systemes dynamiques interconnectes et probleme de consensus en temps fini.* PhD thesis, Universite d'Evry Vald'Essonne; Ecole nationale d'ingenieurs de Tunis (Tunisie), 2014.

Appendix A

Lyapunov stability theorems

A.1 Autonomous systems

When the function f of the system (1.1) is not explicitly time-dependent, i.e

$$\begin{cases} \dot{x} = f(x(t)), \\ x(t_0) = x_0, \end{cases} \tag{A.1}$$

the system becomes an *autonomous system.* The behavior of an autonomous system is invariant to the time change. So the solution $x(t)$ depends on x_0 and $t - t_0$ only, independently of t_0. This leads to the following fact :

For autonomous systems, uniform stability (resp. asymptotic uniform stability) is the same as stability (resp. asymptotic stability)..

Before stating the stability theorems, let's recall some useful Definitions.

Definition A.1.1 *(Positive-definite function (PDF))* A continuously differentiable function $V : \mathbb{R}^n \to \mathbb{R}$ is said to be *PDF* in a domain $D \subset \mathbb{R}$ containing the origin, if :

(a) $V(0) = 0$,

(b) $V(x) > 0, x \in D$ and $x \neq 0$,

The function V is said to be *Positive semi-definite (PSDF)*, if the condition (b) is replaced by $V(x) \geqslant 0$.

Theorem A.1.1 *(Lyapunov's theorem)*[37] Let $D \subset \mathbb{R}^n$ containing the origin. If there exists a function $V : D \to \mathbb{R}$ continuously differentiable and positive definite such that :

$$\dot{V}(x) = \frac{\partial V}{\partial x}\frac{dx}{dt} = \frac{\partial V}{\partial x} f(x),$$

is negative semi-definite (NSDF) in D, then the origin of the system (A.1) is LS. Furthermore, if $\dot{V}$ is NDF, then the origin of the system (A.1) is AS.
In addition, if $D = \mathbb{R}^n$ and V is radially unbounded, i.e

$$\lim_{||x|| \to +\infty} V(x) = +\infty,$$

then the origin of the system (A.1) is GAS.

For autonomous systems, when $\dot{V}$ in the above Theorem is only negative semidefinite, the asymptotic stability can still be obtained by applying the following simplified version of LaSalle's Theorem.

Theorem A.1.2 *(LaSalle's invariance principle)*[37] Let $D \subset \mathbb{R}^n$ containing the origin. If there exists a function $V : D \to \mathbb{R}$ continuously differentiable and positive definite such that :

$$\dot{V}(x) = \frac{\partial V}{\partial x} f(x),$$

in D, and let's define the set S as :

$$S = \{x \in D | \dot{V}(x) = 0\},$$

and assume that no solution other than the origin can remain identically in S, then the origin of the system (A.1) is AS. If on top of that, $D = \mathbb{R}^n$ and V is radially unbounded, the origin of the system (A.1) is GAS.

Theorem A.1.3 *(Linear systems)* The following statements are equivalent :

(a) The origin of the linear system

$$\dot{x} = Ax$$

is globally exponentially stable.

(b) All eigenvalues of A have negative real values.

(c) For any symmetric positive-definite matrix Q, there is a unique symmetric positive-definite matrix P which is a solution to the following Lyapunov equation :

$$PA + A^T P = -Q.$$

A.2 Non-autonomous systems

Definition A.2.1 :*(Class $\mathcal{K}$ function)*[37] The continuous function $\alpha : [0, a) \to \mathbb{R}_+$ is said to be of class $\mathcal{K}$ if :

(a) $\alpha(0) = 0$.

(b) α is strictly increasing.

α is said to be of class $\mathcal{K}_\infty$, if $a = +\infty$ and

$$\lim_{x \to +\infty} \alpha(x) = +\infty.$$

Definition A.2.2 :*(Class $\mathcal{KL}$ function)*[37] The continuous function $\beta : [0, a) \times [0, +\infty) \to \mathbb{R}_+$ is said to be of class $\mathcal{KL}$, if for each fixed t, $\beta(x, t)$ belongs to class $\mathcal{K}$ (with respect to x), and for each fixed x, $\beta(x, t)$ is decreasing (with respect to t) and

$$\lim_{t \to +\infty} \beta(x, t) = 0.$$

Definition A.2.3 :*(Locally positive function (LPDF))* A continuous function $V : \mathbb{R}_+ \times D \to \mathbb{R}_+$ is said to be *Locally positive definite function* (LPDF), if there exists a function α of class $\mathcal{K}$ such that :

(a) $V(t,x) \geqslant \alpha(||x||),\ \forall t \geqslant 0 \text{ and } \forall ||x|| \leqslant r \quad \text{with} \quad r > 0,$

(b) $V(t,0) = 0,$

Definition A.2.4 :*(Positive-definite function (PDF))* A continuous function $V : \mathbb{R}_+ \times \mathbb{R}^n \to \mathbb{R}_+$ is said to be *Positive-definite function*(PDF), if there exists a class $\mathcal{K}$ function α such that :

(a) $V(t,x) \geqslant \alpha(||x||),\ \forall t \geqslant 0 \text{ and } \forall x \in (R)^n,$

(b) $V(t,0) = 0.$

Definition A.2.5 *(Decrescent function)* A continuous function $V : \mathbb{R}_+ \times D \to \mathbb{R}_+$ is said to be *locally decrescent*,if there exists a class $\mathcal{K}$ function α such that :

$$V(t,x) \leqslant \alpha(||x||),\ \forall t \geqslant 0,\ \forall x \leqslant r \ \text{ with } \ r > 0$$

V is said *decrescent*, if α is a class $\mathcal{K}_\infty$ function and the previous inequality holds for all $x \in \mathbb{R}^n$.

Theorem A.2.1 :[79] The origin of the system (1.1) is LS, if there exists a locally positive-definite function $V(t,x)$ such that:

$$\dot{V}(t,x) \leqslant 0,\ \forall t \geqslant t_0,\ \forall ||x|| < r \ \text{ with } \ r > 0$$

Theorem A.2.2 :[79] The origin of the system (1.1) is US, if there exists a locally positive-definite function and locally decrescent $V(t,x)$ such that :

$$\dot{V}(t,x) \leqslant 0,\ \forall t \geqslant t_0,\ \forall ||x|| < r \ \text{ with } \ r > 0$$

Theorem A.2.3 :[79] The origin of the system (1.1) is UAS, if there exists a LPDF et locally decrescent $V(t,x)$, such that $-\dot{V}(t,x)$ is LPDF.

Theorem A.2.4 :[79] The origin of the system (1.1) is GUAS, if there is a positive definite decrescent function $V(t,x)$, such that $-\dot{V}(t,x)$ is PDF.

The function V is called a *Lyapunov function* of the system (1.1).

Appendix B

Finite time stability theorems

B.1 Autonomous systems

Haimo proposed in [32] a necessary and sufficient condition for the finite-time stability of the system (A.1) in the case when (A.1) is a scalar system.

Theorem B.1.1 :[32] The origin of the system (A.1) when $n = 1$ is *finite-time stable* if and only if there exists a neighborhood $\mathcal{V}$ from the origin such that for any $x \in \mathcal{V} - \{0\}$

(a) $xf(x) < 0$,

(b) $\int_x^0 \frac{dz}{f(z)} < \infty$.

Example B.1.1[52] Let's consider the following scalar autonomous system :

$$\dot{x} = -\lceil x \rfloor^a, \quad x \in \mathbb{R}, a \in]0, 1[, \tag{B.1}$$

with :

$$\lceil x \rfloor^a = |x|^a sign(x).$$

We have :

$$-x\lceil x \rfloor^a < 0, \quad \text{pour} \quad x \neq 0,$$

and if $x \in \mathbb{R}$, so :

$$\int_x^0 \frac{dz}{f(z)} = \int_x^0 \frac{dz}{-sign(z)|z|^a} = \frac{|x|^{1-a}}{1-a} < \infty.$$

So according to the *Theorem B.1.1*, The origin of the system (B.1) is finite-time stable with a settling time $T(x_0) = \frac{|x_0|^{1-a}}{1-a}$. The explicit solution of the system (B.1) is given by :

$$x(t) = \begin{cases} sign(x_0)(|x_0|^{1-a} - t(1-a))^{\frac{1}{1-a}}) \ \ si \ \ 0 \leqslant t \leqslant \frac{|x_0|^{1-a}}{1-a}, \\ 0 \qquad si \ \ t > \frac{|x_0|^{1-a}}{1-a}. \end{cases} \tag{B.2}$$

Whereas for autonomous systems of higher dimensions, we present the following theorem :

Theorem B.1.2 :[12] If there exists a Lyapunov function $V : D \subset \mathbb{R}^n \to \mathbb{R}$ such as :

(a) V is PDF.

(b) There is a neighborhood of the origin $\mathcal{V} \subset D$ such that

$$\dot{V} + c(V(x))^{\beta} \leqslant 0, x \in \mathcal{V} - \{0\} \text{ with } c > 0 \;\; et \;\; \beta \in]0,1[$$

Then the origin of the system (A.1) is finite-time stable and the settling time function $T(x_0)$ is continuous and bounded as follows :

$$T(x_0) \leqslant \frac{V(x(0))^{1-\beta}}{c(1-\beta)}.$$

Furthermore, if $D = \mathbb{R}^n$ and $\dot{V}$ is positive on $\mathbb{R}^n$, then the origin of the system (A.1) is globally finite-time stable.

Example B.1.2 :[90] Consider the following autonomous system of dimension 2 :

$$\begin{cases} \dot{x}_1 = -\lceil x_1 \rfloor^a - x_1^3 + x_2, \\ \dot{x}_1 = -\lceil x_2 \rfloor^a - x_2^3 - x_1, \end{cases} \tag{B.3}$$

with $a \in]0,1[$. We set :

$$V(x) = \frac{||x||_2^2}{2} = \frac{1}{2}(x_1^2 + x_2^2).$$

We derive and find :

$$\dot{V}(x) = -x_1^4 - x_2^4 - |x_1|^{a+1} - |x_2|^{a+1} \leqslant 0.$$

The Lyapunov function V verifies the following inequality :

$$\dot{V} \leqslant -2^{\frac{a+1}{2}} V^{\frac{a+1}{2}}.$$

So according to *Theorem B.1.2*, The origin of the system (B.3) is globally finite-time stable with a continuous settling time bounded by $T(x_0) \leqslant \frac{2||x_0||_2^{1-a}}{1-a}$.

Remark B.1.1 : There is no Lyapunov function V that satisfies the conditions of *Theorem B.1.2* for a finite-time stable system with a discontinuous settling time function.

In the case where the settling time function of a finite-time stable system is continuous, the following Theorem represents an inverse to the above Theorem :

Theorem B.1.3 :[12] if The origin of the system (A.1) is finite-time stable and its settling time function T is continuous. Let $\mathcal{V}$ be a neighborhood of the origin and $\beta \in]0,1[$. Then, there exists a continuous function $V : \mathcal{V} \to \mathbb{R}$ that verifies

(a) V is positive definite.

(b) $\dot{V}$ is continuous over V and there exists a real $c > 0$ such that

$$\dot{V} + c(V(x))^{\beta} \leqslant 0, x \in \mathcal{V}.$$

B.2 Non-autonomous systems

Theorem B.2.1 :[53] The origin of the system(1.1) is said to be :

- *Finite-time stable*, if there exists a continuously differentiable Lyapunov function $V : \mathbb{R}_{\geqslant 0} \times D \to \mathbb{R}_{\geqslant 0}$ ($D \subset \mathbb{R}$) and $r : \mathbb{R}_{>0} \to \mathbb{R}_{\geqslant 0}$ is continuous PDF such that $r(0) = 0$ verifying the following differential inequality :

$$\dot{V}(t,x) \leqslant -r(V(t,x)),\ \forall (t,x) \in \mathbb{R}_{+} \times D,$$

  such that there exists $\epsilon > 0$ for which :

$$\int_0^{\epsilon} \frac{dz}{r(z)} < +\infty.$$

  The settling time function of the system (1.1) satisfies the following inequality :

$$T_0(t,x) \leqslant \int_0^{V(t,x)} \frac{dz}{r(z)}.$$

- Furthermore, if V is decrescent, then the origin of the system (1.1) is *uniformly finite-time stable.*

- if V is radially unbounded and $D = \mathbb{R}^n$, then the origin of the system (1.1) is *globally finite-time stable.*

Example B.2.1 :[52] Consider the following non-autonomous scalar system :

$$\dot{x} = -(1+t)\lceil x \rfloor^{a},$$

$t \geqslant 0$, $x \in \mathbb{R}$. We choose the following decrescent Lyapunov function :

$$V(x) = x^2,$$

its derivative is given by :

$$\dot{V}(t,x) = -2(1+t)x\lceil x \rfloor^{a} \leqslant -2\lceil x \rfloor^{\frac{a+1}{2}}(x^2) = -r(V(x)),$$

with $r(z) = 2\lceil x \rfloor^{\frac{a+1}{2}}$.

We have already seen in Example 1 that $\int_0^{\epsilon} \frac{dz}{r(z)} = \int_0^{\epsilon} \frac{dz}{sign(z)|z|^a} < +\infty$, therefore, according to *Theorem B.2.1*, The origin of the system (1.1) is uniformly finite-time stable with the settling time function verifying the following inequality :

$$T(t,x_0) \leqslant \frac{4|x_0|^{1-a}}{1-a}.$$

Appendix C

Observability

C.1 Observability of non-linear systems

C.1.1 SISO systems

Let be the system (2.1), we assume that u and y are known and are derivable, we can study the observability of a SISO system ($m = p = 1$) in the following way [89]

- We define y' and u', such that :

$$y' = [y\ \dot{y}\ \ddot{y}\ ...\ y^{(n-1)}],$$
$$u' = [u\ \dot{u}\ \ddot{u}\ ...\ u^{(n-1)}],$$

- $y^{(i)}$ is dependent on x, u, $\ddot{u}$, $\ddot{u}$, ..., u^i for $i^l \leqslant n-1$. This justifies the following equation :

$$y^{(i)} = \psi_i(x, u').$$

- The derivative of $y^{(i)}$ is therefore given by

$$y^{(i+1)} = [\frac{\partial \psi_i(x,u')}{\partial x}]f(x,u) + [\frac{\partial \psi_i(x,u')}{\partial u'}]\frac{du'}{dt},$$

which in turn equals to $\psi_{i+1}(x, u')$ for $i+1 \leqslant n-1$.
- We define the linear operator $\mathcal{M}_f$ by :

$$(\mathcal{M}_f \psi)(x,u') = [\frac{\partial \psi(x,u')}{\partial x}]f(x,u) + [\frac{\partial \psi(x,u')}{\partial u'}]\frac{du'}{dt}.$$

- y' can be expressed as :

$$y' = \omega(x,u') = \begin{bmatrix} h(x,u) \\ (\mathcal{M}_f h)(x,u') \\ \vdots \\ (\mathcal{M}_f^{n-1} h)(x,u') \end{bmatrix}, \quad \text{(C.1)}$$

where $\omega(x.u')$ is called the *observability matrix*.

- If the observability matrix (C.1) is invertible, i.e., there exists ω^{-1} such that

$$x = \omega^{-1}(y', u'),$$

then the corresponding system is observable, since there is a function ω^{-1} that can reconstruct the state vector from the measurements y and the control input u . Furthermore, if the Jacobian of the observability matrix

$$\Omega(x, u') = \frac{\partial \omega(x, u')}{\partial x},$$

is invertible at x_0, then there exists a neighborhood $\mathcal{V}_{x_0}$ of x_0 in which ω is invertible. The corresponding system is therefore locally observable, meaning that x_0 is distinguishable from all points in $\mathcal{V}_{x_0}$.

C.1.2 MIMO systems

In the case of MIMO systems ($y \in \mathbb{R}^p$, $p > 1$), the approach is similar [89] :
- We define : N as a positive integer vector such that :

$$N = \begin{bmatrix} n_1 & n_2 & \dots & n_p \end{bmatrix}^T,$$

verifying $\Sigma_{i=1}^{i=p} n_i = n$.

- Let :

$$y = \begin{bmatrix} y_1 & y_2 & \dots & y_p \end{bmatrix}^T \text{ and } h(x, u) = \begin{bmatrix} h_1(x, u) & h_2(x, u) & \dots & h_p(x, u) \end{bmatrix}^T.$$

- y'_j is written as follows

$$y'_j = \begin{bmatrix} y_j & \dot{y}_j & \dots & y_j^{n_j} \end{bmatrix}^T = \omega_j(x, u') \text{ with } j = 1, 2, \dots, p.$$

where

$$\omega_j(x, u') = \begin{bmatrix} h_j(x, u) \\ (\mathcal{M}_f h_j)(x, u') \\ \vdots \\ (\mathcal{M}_f^{n-1} h_j)(x, u') \end{bmatrix}.$$

- The observability matrix in this case is given by

$$\omega_N(x, u') = \begin{bmatrix} \omega_1(x, u') \\ \omega_2(x, u') \\ \vdots \\ \omega_p(x, u') \end{bmatrix}.$$

- If there exists N such that $\omega_N(x, u')$ is invertible, then the state x can be reconstructed from u' and measurements y and each of their derivatives y_j can be calculated up to the order n_j. The corresponding system is therefore said to be *observable*.

Remark C.1.1 : The literature offers several definitions of the observability for nonlinear systems. For example, in [28], a method using Lie derivatives has been introduced to study the observability of nonlinear affine systems in control.

C.2 Observability of linear systems

Consider the following autonomous linear dynamic system

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y = Cx(t) + Du(t) \end{cases} \tag{C.2}$$

with $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$ and $y(t) \in \mathbb{R}^p$. A, B and C are matrices of appropriate dimensions. It is assumed that $D = 0$ without loss of generality.

The observability of the system (C.2) can be algebraically characterized by the following theorem.

Theorem C.2.1 :[24] The linear system (C.2) is said to be *observable*, if and only if the Kalman observability matrix given by

$$\begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{bmatrix},$$

is of rank n. The pair (A, C) is then said to be observable.

Remark C.2.1 : This theorem is a direct result of the approach introduced in [28] using Lie derivatives.