



*N° d'ordre: 84/D3C/2024*

*N° de série: 04/elect/2024*

THESIS PRESENTED TO OBTAIN THE DIPLOMA OF:

**3rd Cycle Doctorate (LMD) in (Automatique)**

**Speciality: Automatique et Systèmes**

**Theme**

---

**Model-free Control and Algebraic Estimation**

---

Presented by: **Laid SEHILI**

Thesis defended publicly on, **03/12/2024**

In front of the jury composed of:

Pr. Nora MANSOURI	University of Frères Mentouri Constantine 1	President
Pr. Boubekeur BOUKHEZZAR	University of Frères Mentouri Constantine 1	Supervisor
Pr. Said DRID	University of Batna 2	Examiner
Pr. Nabil TAIB	University of Abderrahmane Mira, Béjaïa	Examiner
Dr. Hocine LEHOUCHE	University of Abderrahmane Mira, Béjaïa	Examiner
Pr. Ahmed NEMMOUR	University of Frères Mentouri Constantine 1	Examiner

To my parents and all my teachers,  
Your guidance, support, and wisdom have been the  
cornerstone of my academic journey. From the earliest days  
of primary school to the halls of university, your influence  
has shaped my intellect and character.  
To my parents, your love and sacrifices have been my  
inspiration. To my teachers, your dedication has fueled my  
passion for learning.  
This thesis is a tribute to your collective impact on my life  
and education. With heartfelt gratitude, I dedicate this  
work to you all.

# Acknowledgement

First and foremost, I am profoundly grateful to Allah for granting me the strength, perseverance, and guidance to complete this project successfully.

I extend my deepest gratitude to my supervisor Prof. Boubekeur Boukhezzar for his invaluable guidance, support, and mentorship throughout this research journey. His expertise and encouragement in choosing the topic and navigating its complexities have been instrumental in the completion of this work.

I am immensely thankful to the staff of the University of Constantine 1, especially Prof. N Mansouri and the members of LARC laboratory, for their support and resources provided during this research endeavor.

I would like to express my gratitude to Prof. Cedric Join, the co-inventor of my research topic on model-free control, for his warm welcome and assistance during my internships at CRAN laboratory in France. His collaboration and insights have greatly enriched the development of this research.

To all my PhD mates, I extend my heartfelt thanks for their camaraderie, support, and shared experiences throughout this academic journey. Your friendship and encouragement have made the challenges more manageable and the successes more meaningful.

This achievement is a testament to the collective efforts and contributions of everyone mentioned above, and for that, I am truly grateful.

*Sehili Laid*

## ملخص

تقدم هذه الرسالة تطورات مبتكرة في طرق مراقبة الحالة ضمن إطار التحكم بدون نموذج، مع التركيز على دمج مقدرات المشتقة والجبرية في تنفيذ النموذج شبه المحلي. من خلال استخدام معاملات التوسيع لتايلور من أي ترتيب، يوفر عملية التقدير إطاراً قوياً لتقدير المعلمات، يتم تنفيذه بشكل واضح باستخدام المفاهيم الرياضية التشغيلية والمرشحات. تُجرى تحليل مقارن بين المقدرات المشتقة والجبرية، وتقييم أدائها في مراقبة الحالة من خلال اختبارات على استراتيجيات التحكم في مسار موضع محرك تيار مستمر تحت التشويش وعدم الإستقرار. بالإضافة إلى ذلك، تقدم هذه الرسالة تقنية التحكم بدون نموذج في سياق عنفة الرياح متغيرة السرعة، باستخدام طرق تقديرية جبرية. يمثل هذا التوسيع للتحكم بدون نموذج في مجال الطاقة المتجددة خطوة هامة، مع تطبيقات تشمل تحسين الضعالية في إمتصاص الطاقة في مناطق الرياح ذات السرعات المنخفضة وتوسيعها إلى تحكم متعدد المتغيرات بدون نموذج لمناطق الرياح ذات السرعات العالية من خلال دمج آليات التحكم في الزاوية والعزم. علاوة على ذلك، تؤسس هذه الرسالة دمجاً بين التحكم القائم على الانسيابية والتحكم بدون نموذج، مستفيدة من إمكانياتهما المجتمعة لتحسين التحكم في عنفة الرياح. تهدف هذه الاستراتيجية المتكاملة إلى تحسين القدرة على التكيف والمتانة وفوائد الأداء، مما يشكل مستقبلاً لطرق التحكم في نظم تحويل طاقة الرياح الحديثة. تتم توثيق فعالية وموثوقية استراتيجيات التحكم المقترحة من خلال اختبارات واسعة النطاق باستخدام برنامج ماتلاب ومحاكي عنفات الرياح. يضمن هذا الإجراء التحقق من جاهزية التطبيق في العالم الحقيقي، مما يعزز الثقة في أدائها عبر مجموعة متنوعة من الإعدادات التشغيلية.

**الكلمات المفتاحية:** تحكم خالي النموذج، التقدير الجبري، متحكم تكاملي تناسبى تفاضلي ذكي، تحكم الانسيابية، رد الإضطرابات، عنفة الرياح ذات السرعة والزاوية المتغيرة، محرك التيار المستمر

## Abstract

This thesis presents innovative advancements in state observation methods within the framework of model-free control, focusing on the integration of derivative and algebraic estimators into the ultra-local model implementation. By leveraging Taylor's expansion coefficients of any order, the estimation process offers a robust framework for parameter estimation, discretely implemented using operational mathematical concepts and FIR filters. A comparative analysis between derivative and algebraic estimators is conducted, assessing their performance in state observation through tests on a DC motor's position trajectory control strategy under disturbances and uncertainty. Furthermore, this research

introduces and extensively tests the model-free Control (MFC) technique in the context of variable-speed wind turbines, employing algebraic estimating methods. This expansion of MFC into renewable energy marks a significant milestone, with applications including energy capture optimization in low-speed wind regions and extension to a multi-variable model-free control (MV-MFC) for high-speed wind regions through the integration of pitch and torque control mechanisms. Moreover, this thesis establishes a merger between flatness-based control and model-free control (F-MFC), leveraging their combined potential to enhance wind turbine control. This integrated strategy aims to improve adaptability, robustness, and performance benefits, shaping the future of wind energy conversion system control methods. The efficacy and reliability of the proposed control strategies are validated through extensive testing with MATLAB and the FAST simulator.

**Keywords:** Model-free Control, Algebraic Estimation, Intelligent PID, Flatness control, Disturbance rejection, Variable speed/pitch wind-turbine, DC motors.

## Résumé

Cette thèse présente des avancées innovantes dans les méthodes d'observation de l'état dans le cadre de la commande sans modèle, en mettant l'accent sur l'intégration d'estimateurs dérivés et algébriques dans la mise en œuvre du modèle ultra-local. En utilisant les coefficients d'expansion de Taylor d'ordre quelconque, le processus d'estimation offre un cadre robuste pour l'estimation des paramètres, implémenté discrètement à l'aide de concepts mathématiques opérationnels et de filtres FIR. Une analyse comparative entre les estimateurs dérivés et algébriques est menée, évaluant leur performance dans l'observation de l'état à travers des tests sur la stratégie de commande de trajectoire de position d'un moteur à courant continu sous des perturbations et de l'incertitude. De plus, cette recherche introduit et teste largement la technique de commande sans modèle (CSM) dans le contexte des éoliennes à vitesse variable, en utilisant des méthodes d'estimation algébriques. Cette expansion du CSM dans les énergies renouvelables marque une étape importante, avec des applications comprenant l'optimisation de la capture d'énergie dans les régions de vent à basse vitesse et l'extension à une commande sans modèle multi-variable (MV-CSM) pour les régions de vent à haute vitesse grâce à l'intégration de mécanismes de contrôle de pas et de couple. De plus, cette thèse établit une fusion entre la commande par platitude et la commande sans modèle, exploitant leur potentiel combiné pour améliorer la commande des éoliennes. Cette stratégie intégrée vise à améliorer l'adaptabilité, la robustesse et les avantages de performance, façonnant l'avenir des méthodes de commande des systèmes

de conversion d'énergie éolienne. L'efficacité et la fiabilité des stratégies des commandes proposées sont validées par des tests approfondis avec MATLAB et le simulateur FAST. Cette procédure de validation garantit la préparation au déploiement dans le monde réel, renforçant la confiance dans leurs performances dans divers contextes opérationnels.

**Mots Clée :** Commande sans modèle, Estimation algébrique, PID intelligent, Commande par platitude, Rejet des perturbations, Éolienne à vitesse/angle de pas variable, Moteurs à courant continu.

# Contents

<b>List of Figures</b>	<b>i</b>
<b>List of Tables</b>	<b>iii</b>
<b>General Introduction</b>	<b>1</b>
<b>1 Model-Free and Flatness Control: Introductory Concepts</b>	<b>4</b>
1.1 Introduction . . . . .	4
1.2 Model-free Control . . . . .	5
1.2.1 What is model-free control ? . . . . .	5
1.2.2 Overview of traditional control methods and their limitations. . . . .	6
1.2.3 Model-free control applications . . . . .	9
1.3 Flatness Control . . . . .	11
1.3.1 Fundamentals of Flatness . . . . .	11
1.3.2 Mathematical Formulation . . . . .	12
1.3.3 Literature review of Flatness applications . . . . .	13
1.4 Combination of Flatness with other control strategies . . . . .	14
<b>2 Model-Free and Flatness Control: Introductory Concepts</b>	<b>16</b>
2.1 Introduction . . . . .	16
2.2 Model-free Control . . . . .	17
2.2.1 What is model-free control ? . . . . .	17
2.2.2 Overview of traditional control methods and their limitations. . . . .	18
2.2.3 Model-free control applications . . . . .	21
2.3 Flatness Control . . . . .	23
2.3.1 Fundamentals of Flatness . . . . .	23
2.3.2 Mathematical Formulation . . . . .	24
2.3.3 Literature review of Flatness applications . . . . .	25
2.4 Combination of Flatness with other control strategies . . . . .	26

<b>3</b>	<b>Algebraic and Derivative Estimation</b>	<b>28</b>
3.1	Introduction . . . . .	28
3.2	Derivative Estimation . . . . .	29
3.2.1	1 <sup>st</sup> Order estimation . . . . .	30
3.2.2	2 <sup>nd</sup> Order estimation . . . . .	32
3.2.3	Implementation of the Derivative Estimator in Discrete Form . . . . .	35
3.3	Ultra-local model estimation . . . . .	38
3.3.1	Intelligent PIDs (iPIDs) . . . . .	40
3.3.2	Online estimation of the ultra-local model through algebraic estimation	40
3.3.3	Online estimation of the ultra-local model through derivative estimation . . . . .	42
3.3.4	Discrete implementation of the ultra-local model through Derivative estimation . . . . .	42
3.3.5	Discrete implementation of the ultra-local model through Algebraic estimation . . . . .	44
3.4	Algebraic Derivative Estimation for Position Control of a DC Motor . . . . .	44
3.4.1	DC motor modeling . . . . .	45
3.4.2	Problem formulation . . . . .	46
3.4.3	Model-free Control Design . . . . .	46
3.4.4	Simulations results . . . . .	50
3.5	Conclusion . . . . .	58
<b>4</b>	<b>Algebraic and Derivative Estimation</b>	<b>60</b>
4.1	Introduction . . . . .	60
4.2	Derivative Estimation . . . . .	61
4.2.1	1 <sup>st</sup> Order estimation . . . . .	62
4.2.2	2 <sup>nd</sup> Order estimation . . . . .	64
4.2.3	Implementation of the Derivative Estimator in Discrete Form . . . . .	67
4.3	Ultra-local model estimation . . . . .	70
4.3.1	Intelligent PIDs (iPIDs) . . . . .	72
4.3.2	Online estimation of the ultra-local model through algebraic estimation	72
4.3.3	Online estimation of the ultra-local model through derivative estimation . . . . .	74
4.3.4	Discrete implementation of the ultra-local model through Derivative estimation . . . . .	74
4.3.5	Discrete implementation of the ultra-local model through Algebraic estimation . . . . .	76

4.4 Algebraic Derivative Estimation for Position Control of a DC Motor . . . .	76
4.4.1 DC motor modeling . . . . .	77
4.4.2 Problem formulation . . . . .	78
4.4.3 Model-free Control Design . . . . .	78
4.4.4 Simulations results . . . . .	82
4.5 Conclusion . . . . .	90
<b>General Conclusion</b>	<b>92</b>
<b>General Conclusion</b>	<b>94</b>
<b>Bibliography</b>	<b>96</b>

# List of Figures

1.1	Architecture of MBC theory . . . . .	8
1.2	Architecture of MFC theory . . . . .	9
1.3	The number of annual publications related to MFC since 2009 . . . . .	11
2.1	Architecture of MBC theory . . . . .	20
2.2	Architecture of MFC theory . . . . .	21
2.3	The number of annual publications related to MFC since 2009 . . . . .	23
3.1	MFC scheme . . . . .	47
3.2	parameter's tuning procedure $\beta$ . . . . .	49
3.3	Sinusoidal trajectory - Position tracking employing derivatives and algebraic estimators based on the ultra-local model . . . . .	51
3.4	An estimate of the angular velocity derived from a sinusoidal trajectory $\hat{y} = (\hat{\theta}_m(t))$ . . . . .	52
3.5	An estimate of the unmodeled dynamics for a sinusoidal trajectory . . . . .	52
3.6	Sinusoidal trajectory - Control signal input using algebraic estimator and iPD	53
3.7	Sinusoidal trajectory - Control signal input using derivative estimator and iPD controllers . . . . .	53
3.8	Sinusoidal trajectory - Error tracking with algebraic/derivative estimators and iPD . . . . .	54
3.9	Perturbed Bezier's trajectory - The DC motor' position tracking using iPID with algebraic estimator and considering Coulomb's friction effects . . . . .	54
3.10	Perturbed Bezier's trajectory - Position tracking of the DC motor using iPID with derivative estimator and considering Coulomb's friction effects .	55
3.11	Sinusoidal trajectory: Comparing conventional PD and derivative/algebraic estimators with iPD . . . . .	56
3.12	Sinusoidal trajectory - Error tracking with traditional PD . . . . .	57
3.13	Perturbed Bezier's trajectory - The DC motor position tracking using traditional PID considering Coulomb's friction effects . . . . .	57

4.1	MFC scheme . . . . .	79
4.2	parameter's tuning procedure $\beta$ . . . . .	81
4.3	Sinusoidal trajectory - Position tracking employing derivatives and algebraic estimators based on the ultra-local model . . . . .	83
4.4	An estimate of the angular velocity derived from a sinusoidal trajectory $\hat{y} = (\hat{\theta}_m(t))$ . . . . .	84
4.5	An estimate of the unmodeled dynamics for a sinusoidal trajectory . . . . .	84
4.6	Sinusoidal trajectory - Control signal input using algebraic estimator and iPD . . . . .	85
4.7	Sinusoidal trajectory - Control signal input using derivative estimator and iPD controllers . . . . .	85
4.8	Sinusoidal trajectory - Error tracking with algebraic/derivative estimators and iPD . . . . .	86
4.9	Perturbed Bezier's trajectory - The DC motor' position tracking using iPID with algebraic estimator and considering Coulomb's friction effects . . . . .	86
4.10	Perturbed Bezier's trajectory - Position tracking of the DC motor using iPID with derivative estimator and considering Coulomb's friction effects . . . . .	87
4.11	Sinusoidal trajectory: Comparing conventional PD and derivative/algebraic estimators with iPD . . . . .	88
4.12	Sinusoidal trajectory - Error tracking with traditional PD . . . . .	89
4.13	Perturbed Bezier's trajectory - The DC motor position tracking using traditional PID considering Coulomb's friction effects . . . . .	89

# List of Tables

3.1	DC motor characteristics . . . . .	46
3.2	Sinusoidal trajectory - The efficiency of the control methods according to the ITAE, ISE, and IAE criteria . . . . .	54
3.3	Perturbed Bezier's trajectory - The efficiency of the control methods according to the ITAE, ISE, and IAE criteria . . . . .	55
4.1	DC motor characteristics . . . . .	78
4.2	Sinusoidal trajectory - The efficiency of the control methods according to the ITAE, ISE, and IAE criteria . . . . .	86
4.3	Perturbed Bezier's trajectory - The efficiency of the control methods according to the ITAE, ISE, and IAE criteria . . . . .	87

# General Introduction

The control of dynamic systems has undergone significant advancements in recent years, with the emergence of model-free control techniques offering new perspectives and approaches. This thesis explores the integration of model-free control and algebraic estimation methods, aiming to enhance the performance and robustness of control systems in various engineering applications. Through a combination of theoretical analysis, algorithm development, and experimental validation, this research addresses key challenges in control engineering, particularly focusing on dynamic systems such as DC motors and variable-speed wind turbines.

The literature on model-free control and algebraic estimation provides valuable insights into the theoretical foundations and practical applications of these techniques in control engineering. Previous studies have demonstrated the effectiveness of model-free control approaches in various domains, including robotics, aerospace, and renewable energy systems. However, limited research exists on the integration of model-free control with algebraic estimation methods, particularly in the context of flatness control.

The theoretical framework guiding this research draws upon principles from control theory, optimization, and system identification. Ultra-local model and algebraic estimators serve as fundamental concepts, providing a theoretical basis for the development of novel control strategies without explicit system modeling. Additionally, flatness-based control offers a theoretical framework for representing complex nonlinear systems, complementing the adaptability and robustness of model-free control technique.

The findings of this research are expected to have significant implications for both theory and practice in the field of control engineering. By advancing the understanding and application of model-free control technique combined with algebraic estimation, this study aims to contribute to the development of more robust and adaptive control strategy for a wide range of engineering systems. The proposed approaches in this research have the potential to improve the performance, efficiency, and reliability of dynamic systems, including DC motors and variable-speed wind turbines.

The present study conducts extensive research, attempting to reveal the significant

consequences of robust state observation in strengthening control system stability by detailed comparison analysis and simulation tests. This study examines the design and implementation of model-free controllers, employing the ultra-local model along with derivative and algebraic estimators to address essential features such as discrete realization and parameter tuning using FIR filters.

In the context of DC motor position tracking, algebraic and derivative estimators demonstrate superior performance in robust tracking toward Coulomb friction disturbances and uncertainties, outperforming traditional PID controllers. Transitioning to variable-speed wind turbine systems, model-free controllers emerge as a promising avenue, with the algebraic estimator presenting itself as an alternative to conventional state observers. This not only ensures robust closed-loop performance but also streamlines the control strategy, particularly in scenarios where acquiring accurate dynamical models poses formidable challenges.

Furthermore, the integration of flatness-based model-free control within wind turbine systems underscores exceptional capabilities in optimizing wind energy capture and regulating power generation across varying speeds. This approach significantly contributes to enhancing system robustness, guaranteeing reliable performance even in the face of uncertainties and disturbances.

## Key Contributions

The main contributions of this thesis can be summarized as follow:

**1) Advancement of the ultra-local model-based estimators:** This thesis explores and applies the approaches of the ultra-local model observations in model-free control theory. Using Taylor's expansion parameters for arbitrary orders entails creating ultra-local models based on derivative/algebraic estimators. The study carefully describes the discrete application of these techniques, which include FIR filters and operational calculus.

**2) Validation and extension of model-free control (MFC) approach:** This thesis also introduces and extensively tests the model-free Control (MFC) technique to variable-speed wind turbines based on algebraic estimating. This expands the effective application of MFC to a new discipline of renewable energy.

**3) Optimization of low-speed energy capture:** In this research we apply model-free control based on wind estimation for maximizing energy capture in low-speed wind

regions. This objective aims to improve wind energy transformation systems' efficiency by tailoring control strategies to exploit optimal power coefficients.

**4) Enhancement of high-speed Power quality:** This study extends the model-free control (MFC) to a multi-variable model-free control (MV-MFC) in high-speed wind regions. By integrating pitch and torque control mechanisms, the objective is to sustain satisfactory power quality, ensuring stability and performance even under demanding wind conditions.

**5) Integration of flatness approach with model-free control:** This research establishes a merge between flatness-based control and model-free control (MFC), using their combined potential to improve wind turbine control. This integrated strategy seeks to improve adaptability, robustness, and performance benefits, bringing in the future of cutting-edge wind energy conversion system control methods.

**6) Simplifying system modeling complexity:** Elimination of the difficulties associated with system modeling in wind turbines, facilitating real-time updates devoid of concerns regarding linearity or specific parameter identification. This simplification enables agile adaptation to dynamic environmental conditions, enhancing overall system responsiveness.

## Organization of the thesis:

This thesis is structured into four main chapters:

Chapter 1 provides an introduction to the research topic, outlining the objectives, methodology, and theoretical framework of the study. Chapter 2 focuses on developing and implementing the model-free control technique for controlling the position of a DC motor employing the ultra-local model combined with real-time derivative and algebraic estimators. Chapter 3 explores model-free control approaches to variable-speed wind turbines, addressing challenges related to energy capture and power quality under varying wind conditions. Chapter 4 investigates the integration of flatness-based control with model-free techniques for wind turbine control, examining the benefits and challenges of this combined approach.

# Chapter 1

## Model-Free and Flatness Control: Introductory Concepts

### 1.1 Introduction

In the realm of control systems engineering, the pursuit of precision, adaptability, and robustness has driven the evolution of control methodologies beyond traditional model-based approaches [1]. This journey has led to the emergence of model-free control and flatness control as pivotal paradigms offering innovative solutions to complex control problems across various domains. This chapter delves into the state of the art and introductory concepts of model-free control and flatness control, exploring their theoretical foundations, practical applications, and potential synergies.

Control systems traditionally rely on accurate mathematical models to predict system behavior and design control strategies accordingly [2]. However, the inherent complexity, nonlinearity, and uncertainties present in many real-world systems often challenge the efficacy of model-based approaches. Model-free control offers an alternative paradigm by leveraging data-driven techniques to directly learn control policies from system interactions, without explicit knowledge of system dynamics [3]. Reinforcement learning, adaptive control, and other model-free techniques have garnered significant attention for their ability to tackle complex control tasks in diverse domains such as robotics, autonomous vehicles, and process control [4].

In parallel, flatness control has emerged as a powerful methodology for addressing the trajectory tracking and disturbance rejection challenges inherent in nonlinear systems [5]. The concept of flatness centers around the idea of reparameterizing a nonlinear system such that its state and input trajectories become trivial, enabling straightforward control design and analysis. By exploiting the flatness property, controllers can achieve precise

trajectory tracking and disturbance rejection, even in the presence of nonlinearities and uncertainties [6].

This chapter provides a comprehensive overview of model-free control and flatness control, clarifying their fundamental principles, key techniques, and practical applications. We explore various algorithms and methodologies employed in model-free control, ranging from reinforcement learning to adaptive control, highlighting their advantages and limitations. Additionally, we delve into the theory and approaches underlying flatness control, shedding light on differential flatness, input-output flatness, and their implications for control system design.

Moreover, we discuss the merge of model-free control and flatness control, exploring the potential synergies between these two approaches and their implications for advancing control system performance. In the next chapters, we propose to unlock new avenues for addressing complex control challenges and enhancing system performance by combining the data-driven adaptability of model-free control with the trajectory optimization capabilities of flatness control.

In summary, this chapter serves as a comprehensive guide to the state of the art and introductory concepts of model-free control and flatness control. By understanding the principles and techniques underlying these paradigms, we can navigate the complexities of modern control systems and harness the full potential of data-driven and trajectory-based control methodologies.

## 1.2 Model-free Control

This section overviews model-free control, focusing on its fundamental principles, key concepts, literature review, and applications.

### 1.2.1 What is model-free control ?

Approaches and control techniques can be classified into two categories: control laws that require prior knowledge of a model of the system to be controlled (referred to as Model-based Control) and those that do not require model knowledge (referred to as Model-free Control). Although theoretically, they can provide good performance, implementing control strategies based on model knowledge is challenging. Indeed, the mathematical model resulting from modeling generally does not accurately reflect the dynamics of the system. Furthermore, due to external effects, the system's dynamics can change, which can significantly degrade performance.

We can define model-free control as any type of control that does not require knowledge of the model of the system to be implemented. Still, rather only general information about the system is required, including its class, order, inputs, and outputs. Despite its limitations and the subject of much criticism in the control community, the PID controller, as a simple-to-implement model-free control, is the most widely used controller in the industrial world. Finding another model-free control strategy with higher performance has become one of the concerns of control researchers. In this research, our objective is to explore new approaches that combine model-free control with other techniques.

## 1.2.2 Overview of traditional control methods and their limitations.

Starting from the late 1960s, modern control theory has significantly evolved and expanded. Notable branches, including variable structure control [7], robust control [8], adaptive control [9], system identification [10], optimal control [11], stochastic theory [12], and many other control methods have widespread applications in various fields, including traffic systems, aerospace, and industrial operations. Despite its maturity, modern control theory still presents numerous theoretical and practical challenges, emphasizing the field's continual need for innovation and improvement.

### 1.2.2.1 Identification and Modeling

Model-based control (MBC) or modern control theory began with introducing Kalman's parametric state-space theory in 1960 and applying optimal control techniques [1, 2]. Its application flourished, especially in aerospace, leveraging precise models readily available in the field. Modern control theory encompasses approaches for either linear or nonlinear systems. LQR design [13], Zero-pole assignment [14], and robust control [8] are examples of linear control system design techniques. On the other hand, controller design techniques for nonlinear systems frequently include feedback linearization [15], backstepping [16], and Lyapunov-based designs [17]. Together, these approaches show the common architecture of MBC systems. When using MBC theory in practice, the first step is to model or identify the plant. Then, using the *certainty equivalency principle*, controller design is based on the obtained plant model, assuming that the plant model accurately represents the actual system. Modeling and plant identification are therefore crucial elements of MBC theory.

In order to model a plant from first principles, parameters must be calibrated using measured data, either online or offline. Using identification theory, one can create a plant model within a model set that either fully captures the true system or comes close to it [10].

Regardless of whether modeling is conducted through first principles or data identification, it inherently involves approximating the true system, inevitably incurring some error [18]. Unmodeled dynamics are invariably present in the modeling process, contributing to potential inaccuracies in closed-loop control systems designed using model-based control (MBC) approaches [19].

Efforts have been directed towards enhancing robustness against model errors while retaining the advantages of MBC design. Robust control approach has explored different ways to describe errors of the model in closed-loop system configurations, including additive and multiplicative descriptions, as well as assumptions regarding modeling errors or noises. Nevertheless, the uncertainty characterizations utilized in robust theory design techniques often do not align with those derived from physical-mathematical modeling and identification techniques [10].

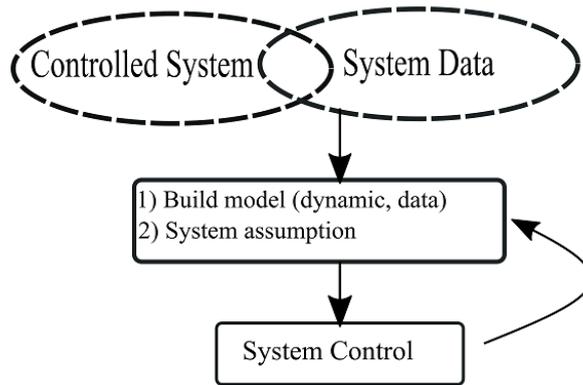
Challenges persist in establishing a perfect control theory, primarily due to the inherent trade-off between unmodeled dynamics and robustness. Achieving high accuracy in plant modeling requires significant effort and cost, and there is no efficient means of producing an accurate model. Additionally, addressing complexities such as parameter variations or structural changes over time poses further challenges. High-order system dynamics may necessitate controller order reduction for practical implementation, highlighting the paradox of striving for accuracy in modeling only to simplify the model for control design [19].

### 1.2.2.2 Model-based control design

The principle of certainty equivalency, a cornerstone of model-based control (MBC) theory, assumes that controllers are effective only if designed based on a plant model within the assumed set. Deviations from this set cause poor performances and system instability due to the sensitivity of closed-loop systems to even minor modeling errors [20].

Even with an accurate model, theoretical analyses covering stability, convergence, and robustness may yield limited insights if there are unfounded presumptions about the system. The MBC theory's architecture, illustrated in Fig. 2.1, underscores the system model's central role and assumptions in controller design and system analysis. However, a discrepancy persists between the model constructed based on assumptions and the actual plant, potentially undermining the effectiveness of MBC approaches if overlooked during design and analysis.

Nonlinear control systems design methods such as Lyapunov-based approaches, backstepping, and feedback linearization rely heavily on accurate plant models. However, the unmodeled dynamics' existence uncertainty can undermine the effectiveness of these



**Figure 1.1:** Architecture of MBC theory

methods. Inaccuracies in the model's structure and dynamic equations can significantly impact system performance since controllers incorporate these details [21]. The significant disparity between the elegant theories of model-based control (MBC) and practical implementation poses a major challenge, leading to various issues in real-world applications. When the model is unavailable, or its assumptions are invalid, it becomes challenging to make significant conclusions. The reliance of MBC on the model underscores its characterization as a "model theory" rather than solely a control theory to some extent.

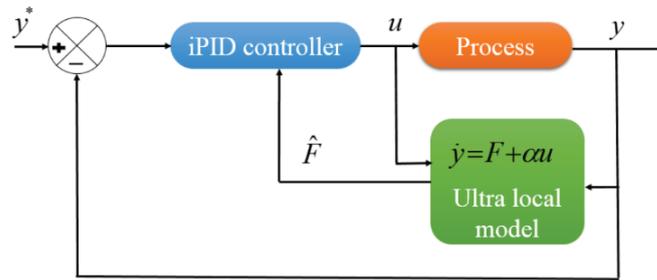
### 1.2.2.3 Model-free control design

This section provides an introduction to "intelligent" PID controllers (iPIDs), which are a counterpart to "model-free control" in a unified manner, considering recent advances [22]. The approach integrates basic differential algebra with outdated analytical techniques. A new online parameter estimation method simplifies identification methods. The significance of iPIDs, particularly iP, is highlighted in dealing with friction. The prevalence of classic PID controllers in industry and their challenges in complex tuning situations are attributed to their connection with iPIDs, as demonstrated through elementary sampling. Chapter 4.1 presents a detailed mathematical explanation of the approach.

Let's review some key theoretical concepts underlying the model-free control approach. Our focus is on systems that have a single control variable  $u$  and a single output variable  $y$  for simplicity's sake. Instead of relying on a complex mathematical model, we adopt an ultra-local model represented in Fig 2.2 as:

$$y^{(\nu)} = F + \alpha u \quad (1.1)$$

where,  $F$  is updated regularly and encompasses both the various potential disturbances and the unappreciated plant parts despite requiring any differentiation,  $\alpha$  is constant, and



**Figure 1.2:** Architecture of MFC theory

$u$  represents the input control signal. The estimation:

- requires only a brief period of time.
- is characterized by algebraic equations that incorporate low-pass filters such as repeated time integrals.
- is robust to appropriately significant noises.

The determined input-output data of the controlled plant is the only factor that influences the controller design in the MFC methodology. Fig. 2.1 illustrates the MFC methodology's architecture. One of this system's features is that it solely uses the measurement of I/O data from the closed-loop systems for both control system analysis and MFC controller design. The plant model vanishes and loses its ability to control the procedure. The closed-loop control system's measurement of I/O data serves as both a standard for control system performance and the origin of controlled problems. More details will be discussed in Chapter 2.

### 1.2.3 Model-free control applications

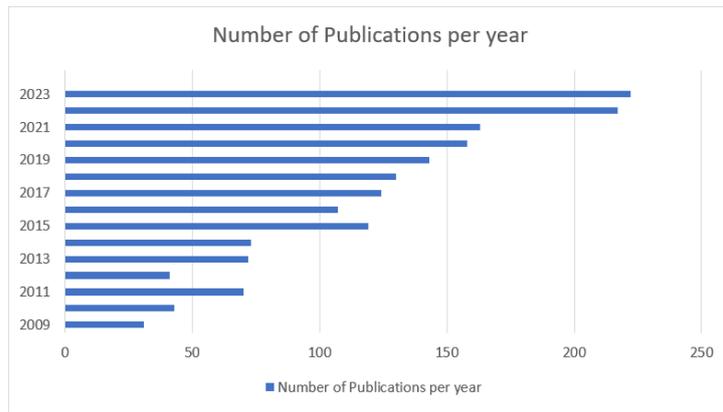
With the advancement of technology and information science, industries such as chemical, metallurgical, electronics, machinery, transportation, and logistics have experienced significant transformations. Those sectors operate on a large scale with complex production technologies and equipment, making modeling processes employing traditional rules or identification methods increasingly challenging. Consequently, conventional model-based control (MBC) approaches have become ineffective in addressing control problems in such industries. Moreover, huge quantities of data are continuously generated by modern industrial processes, and this data contains important details about the states of the equipment and the operations of the processes. Leveraging this data, both online and

offline, for controller design, state prediction, performance evaluation, decision-making, and fault diagnosis becomes crucial, especially without having precise process models. Thus, the development of model-free control theory (MFC) has emerged as a pressing issue in both theory and application domains.

The term "Model-free" has recently gained traction in the control community. While several MFC methods exist, they are often referred to by various identities such as data-driven control, modelless control, data-based control, iterative feedback tuning (IFT)[3], virtual reference feedback tuning (VRFT)[23], active disturbance rejection (ADRC) [24] and iterative learning control (ILC)[25]. Although these terms are used interchangeably, there are subtle differences between them.

Several publications in a wide variety of fields have demonstrated the efficiency and simplicity of MFC approach, we will mention here the most interesting ones.

Among the first tests of MFC was the DC-DC converter system [26]. MFC have been applied to Quadrotor Vehicle Quadrotor Vehicle by Younes [27]. Application reported by Frédéric Lafont includes information on a concrete greenhouse and comparisons with Boolean controllers[28]. A proposed algorithm were introduced based on an integrated structure between the Nonlinear Integral-Backstepping technique (NIB) and the MFC in [29] for a Quadrotor UAV. MFC technique has been also introduced in medical field to solve the problem of regulating glycemia of Type-1 Diabetes(T1D) [30] and the problem of immune therapy [31]. A novel cascade model free adaptive control method (MFAC) with an intelligent PID (i-PID) algorithm is proposed for the attitude adjustment of a multi-input multi-output (MIMO) quadrotor aircraft [32].In order to control the attitude and position of a quadrotor whose model includes parameter variations, uncertainties, and external disturbances, Haoping developed a model-free-based terminal sliding-mode control (MFTSMC) strategy. This method combines a sliding-mode technique with a model-free control approach, allowing for the elimination of tracking error in a finite amount of time [33]. Xinyi Zhang considers a model-free based neural network control with time-delay estimation (TDE-MFNNC) for lower extremity exoskeleton [34]. Raul-Cristian proposes a hybrid controller that consists of a second order data-driven Model-Free Control (MFC), and a Takagi-Sugeno Fuzzy (TSF) logic controller for nonlinear Multi Input-Multi Output (MIMO) twin rotor aerodynamic systems (TRASs) [35]. In [36] Mircea-Bogdan presents a novel model-free trajectory tracking of multiple-input multiple-output (MIMO) systems by the combination of iterative learning control (ILC) and primitives where the optimal trajectory tracking solution is obtained in terms of previously learned solutions to simple tasks called primitives. Kaihui Zhao suggests a robust model-free nonsingular terminal sliding-mode control (MFNTSMC) algorithm based on the ultra-local model to reduce the influence of permanent magnet (PM) demagnetization for PM synchronous motors



**Figure 1.3:** The number of annual publications related to MFC since 2009

(PMSMs) [37]. In [38], model-free fractional-order nonsingular fast terminal sliding mode control (MFF-TSM) based on time delay estimation (TDE) is proposed for the lower-limb robotic exoskeleton in the presence of uncertainties and external disturbance. Ghazally I.Y comprises the model-free based intelligent PID controller, the fuzzy logic controller, the time-delay estimation, and the particle swarm optimization to control the vibration of an active vehicle suspension systems [39]. Recently, Fliess and Cédric introduced MFC as a Service in the industrial internet of Things and machine learning [40, 41]. Figure. 2.3 shows the number of annual publications related to MFC since 2009.

This thesis introduces and validates the Model-free approach to variable speed wind turbine in Chapter 3.

## 1.3 Flatness Control

This section presents the key point of flatness control lies in its ability to ensure that a dynamic system exhibits flat outputs. Flatness control aims to manipulate a system's inputs in such a way that the outputs become simple functions of a few "flat" variables. These flat outputs essentially capture all the relevant information about the system's behavior. This approach offers advantages in terms of simplifying control design, facilitating trajectory planning, and enhancing robustness to disturbances.

### 1.3.1 Fundamentals of Flatness

The concept of flatness emerged from the need to address the challenges posed by nonlinear systems in control engineering. Traditional control techniques struggled with the complexity of nonlinear dynamics, motivating researchers to seek alternative approaches. In the 1980s and early 1990s, researchers were seeking ways to simplify the control of nonlinear systems.

In 1995, a group of researchers published a seminal paper in the International Journal of Control titled "Flatness and Defect of Nonlinear Systems: Introductory Theory and Examples," exploring the concept of "flatness" as a means to achieve this goal. The paper laid out the basic principles of flatness control and provided illustrative examples [5] and marked a significant milestone in the formalization of flatness theory. Another important paper, published in 1999 in the IEEE Transactions on Automatic Control, titled "A Lie-Bäcklund approach to equivalence and flatness of nonlinear systems," further advanced the theory by introducing Lie-Bäcklund transformations for analyzing the equivalence and flatness of nonlinear systems [42].

Flatness control operates on the principle of transforming a nonlinear dynamic system into a form where the outputs are described by flat variables. These flat outputs are chosen to simplify control design and trajectory planning. By exploiting the flatness property, control engineers can design controllers that directly manipulate these flat outputs, thereby simplifying control synthesis and trajectory tracking tasks.

### 1.3.2 Mathematical Formulation

By introducing the concepts of flatness and linearizing output, a novel perspective on the complete linearization issue through dynamic feedback is put forth [15]. This section aims to demonstrate how beneficial such a viewpoint can be for movement tracking [5, 42, 43, 44]. The late 1980s saw the invention of flatness-based control and differential flatness theory. Since then, these concepts have continuously progressed, offering effective answers to complex control and state estimation problems [45].

A differentially flat system is defined as follows: Since  $f$  is a smooth vector field, a system  $\dot{x} = f(x, u)$  with state vector  $x \in R^n$  and input vector  $u \in R^m$  is differentially flat if there is a vector  $y \in R^m$  with the form:

$$y = h(x, u, \dot{u}, \dots, u^{(r)}) \quad (1.2)$$

That is,

$$\begin{aligned} x &= \phi(y, \dot{y}, \dots, y^{(q)}) \\ u &= \alpha(y, \dot{y}, \dots, y^{(q)}) \end{aligned} \quad (1.3)$$

as smooth functions  $h, \phi$ , and  $\alpha$ . Accordingly, the  $m$  algebraic variables  $y_i, i = 1, 2, \dots, m$  describe the new system.  $y = h(x, u, \dot{u}, \dots, u^{(r)})$  was the definition of the flat output provided above. Zero-flat systems are those in which the flat output is solely a function of the state vector  $x$ . Nevertheless, it could be necessary to represent the flat output as a function of the control  $u$  and its derivatives in addition to the state vector  $x$ .

This necessity arises in more complex scenarios, such as when dealing with dynamic feedback linearization or applying dynamic extension techniques. In such cases, the system's state vector is extended by including the control inputs and their derivatives as additional state variables. This extension allows for a more comprehensive and flexible description of the system, accommodating the intricacies of control and state estimation problems. By incorporating these elements, we ensure that the flat output adequately captures the dynamic behavior of the system, facilitating more effective control strategies and solutions.

Equation (2.3) demonstrates that the flat output and its derivatives can be used to express the state vector of the differentially flat system and its control inputs. In the study of differential flatness, the fundamental question is whether there exists a function  $y = h(\cdot)$  given by  $y = h(x, u, \dot{u}, \dots, u^{(r)})$ , such that the state vector of the system  $x$  and the control input  $u$  can be stated as functions of  $y$  and of its derivatives, as in Eq. (2.3). This is given by the differential equations that describe the nonlinear system dynamics  $\dot{x} = f(x, u)$ .

### 1.3.3 Literature review of Flatness applications

Flatness control is a versatile and effective technique with applications spanning diverse domains, showcasing its broad applicability in optimizing system performance and achieving precise control objectives. Its utility is evident in stabilizing and maneuvering quadrotor UAVs for precise trajectory tracking, allowing these unmanned aerial vehicles to navigate through complex environments with ease [46]. Additionally, it enhances the navigation capabilities of autonomous underwater vehicles, enabling them to explore underwater terrain accurately and efficiently [47]. In industrial settings, flatness control facilitates precise manipulation of robotic arms [48], ensuring accurate assembly and manufacturing processes [49]. Moreover, it plays a pivotal role in optimizing wind turbine energy capture, maximizing power generation efficiency while minimizing structural loads [50]. In hybrid energy systems, such as those combining fuel cells and renewable sources, flatness control manages power flow to maximize energy output and system stability [51]. It also controls electric vehicle powertrains, optimizing power distribution for enhanced performance and range [52]. In aerospace applications, flatness control maintains satellite orientation in space missions, ensuring precise positioning and stabilization [53]. Additionally, it regulates industrial processes like chemical reactors and distillation columns, optimizing production efficiency and quality [54]. In aircraft flight dynamics, flatness control enables autonomous navigation and altitude control, ensuring safe and efficient flight operations [55]. Furthermore, it finds applications in smart grid energy management [52], HVAC systems

[54], biomedical systems[56], water treatment plants [57], renewable energy microgrids [58], and traffic management systems [59], contributing to improved efficiency, reliability, and sustainability across various industries and applications.

## 1.4 Combination of Flatness with other control strategies

Combining control techniques can indeed lead to improved system performance and robustness. Here are a few research papers that demonstrate the integration of flatness/model-free control approaches with other control techniques.

The integration of flatness control with various other control techniques offers significant added value across different applications. In traditional control methods, such as adaptive [60] control and proportional-derivative-integral (PID) control [61], combining them with flatness control enhances robustness and adaptability, ensuring precise motion control despite uncertainties. Additionally, integrating flatness control with fuzzy logic [62] or backstepping control [63] enables robust and adaptive regulation of system dynamics, particularly in complex systems like DC motor speed control or hydraulic servo systems. Moreover, combining flatness control with modern techniques like reinforcement learning [64] or model predictive control [65, 66] enhances system performance and autonomy. Reinforcement learning enables the system to learn optimal control policies, while model predictive control anticipates future behavior, both ensuring efficient trajectory tracking and navigation, as seen in applications such as autonomous vehicles or wind turbines. Furthermore, incorporating flatness control with fault detection techniques like sliding mode observers[67] improves system reliability by enabling real-time fault detection and isolation in electromechanical systems. Finally, integrating flatness control with deep reinforcement learning or artificial neural networks [68, 69] enhances adaptability and efficiency, maximizing performance in tasks such as solar tracking or quadcopter trajectory tracking. In essence, these combinations leverage the strengths of each method to achieve superior control performance, stability, and robustness across diverse applications.

This thesis significantly contributes by showcasing the innovative integration of the differential flatness approach with the model-free control technique. Combining flatness-based control with model-free control (MFC) offers a synergistic approach that leverages the structured framework of flatness-based control for system modeling and analysis while harnessing the adaptability and resilience of MFC to uncertainties and disturbances. This integration enhances adaptability, robustness, and performance benefits, as flatness-based control ensures precise system output regulation and decoupling of nonlinear dynamics,

while MFC optimizes control actions in real-time based on online observations. By merging these methodologies, control strategies become more versatile, future-proofed, and capable of achieving superior performance in dynamic and uncertain environments, making them particularly valuable for applications such as wind energy conversion systems. This is deeply discussed in Chapter 4.

# Chapter 2

## Model-Free and Flatness Control: Introductory Concepts

### 2.1 Introduction

In the realm of control systems engineering, the pursuit of precision, adaptability, and robustness has driven the evolution of control methodologies beyond traditional model-based approaches [1]. This journey has led to the emergence of model-free control and flatness control as pivotal paradigms offering innovative solutions to complex control problems across various domains. This chapter delves into the state of the art and introductory concepts of model-free control and flatness control, exploring their theoretical foundations, practical applications, and potential synergies.

Control systems traditionally rely on accurate mathematical models to predict system behavior and design control strategies accordingly [2]. However, the inherent complexity, nonlinearity, and uncertainties present in many real-world systems often challenge the efficacy of model-based approaches. Model-free control offers an alternative paradigm by leveraging data-driven techniques to directly learn control policies from system interactions, without explicit knowledge of system dynamics [3]. Reinforcement learning, adaptive control, and other model-free techniques have garnered significant attention for their ability to tackle complex control tasks in diverse domains such as robotics, autonomous vehicles, and process control [4].

In parallel, flatness control has emerged as a powerful methodology for addressing the trajectory tracking and disturbance rejection challenges inherent in nonlinear systems [5]. The concept of flatness centers around the idea of reparameterizing a nonlinear system such that its state and input trajectories become trivial, enabling straightforward control design and analysis. By exploiting the flatness property, controllers can achieve precise

trajectory tracking and disturbance rejection, even in the presence of nonlinearities and uncertainties [6].

This chapter provides a comprehensive overview of model-free control and flatness control, clarifying their fundamental principles, key techniques, and practical applications. We explore various algorithms and methodologies employed in model-free control, ranging from reinforcement learning to adaptive control, highlighting their advantages and limitations. Additionally, we delve into the theory and approaches underlying flatness control, shedding light on differential flatness, input-output flatness, and their implications for control system design.

Moreover, we discuss the merge of model-free control and flatness control, exploring the potential synergies between these two approaches and their implications for advancing control system performance. In the next chapters, we propose to unlock new avenues for addressing complex control challenges and enhancing system performance by combining the data-driven adaptability of model-free control with the trajectory optimization capabilities of flatness control.

In summary, this chapter serves as a comprehensive guide to the state of the art and introductory concepts of model-free control and flatness control. By understanding the principles and techniques underlying these paradigms, we can navigate the complexities of modern control systems and harness the full potential of data-driven and trajectory-based control methodologies.

## 2.2 Model-free Control

This section overviews model-free control, focusing on its fundamental principles, key concepts, literature review, and applications.

### 2.2.1 What is model-free control ?

Approaches and control techniques can be classified into two categories: control laws that require prior knowledge of a model of the system to be controlled (referred to as Model-based Control) and those that do not require model knowledge (referred to as Model-free Control). Although theoretically, they can provide good performance, implementing control strategies based on model knowledge is challenging. Indeed, the mathematical model resulting from modeling generally does not accurately reflect the dynamics of the system. Furthermore, due to external effects, the system's dynamics can change, which can significantly degrade performance.

We can define model-free control as any type of control that does not require knowledge of the model of the system to be implemented. Still, rather only general information about the system is required, including its class, order, inputs, and outputs. Despite its limitations and the subject of much criticism in the control community, the PID controller, as a simple-to-implement model-free control, is the most widely used controller in the industrial world. Finding another model-free control strategy with higher performance has become one of the concerns of control researchers. In this research, our objective is to explore new approaches that combine model-free control with other techniques.

## 2.2.2 Overview of traditional control methods and their limitations.

Starting from the late 1960s, modern control theory has significantly evolved and expanded. Notable branches, including variable structure control [7], robust control [8], adaptive control [9], system identification [10], optimal control [11], stochastic theory [12], and many other control methods have widespread applications in various fields, including traffic systems, aerospace, and industrial operations. Despite its maturity, modern control theory still presents numerous theoretical and practical challenges, emphasizing the field's continual need for innovation and improvement.

### 2.2.2.1 Identification and Modeling

Model-based control (MBC) or modern control theory began with introducing Kalman's parametric state-space theory in 1960 and applying optimal control techniques [1, 2]. Its application flourished, especially in aerospace, leveraging precise models readily available in the field. Modern control theory encompasses approaches for either linear or nonlinear systems. LQR design [13], Zero-pole assignment [14], and robust control [8] are examples of linear control system design techniques. On the other hand, controller design techniques for nonlinear systems frequently include feedback linearization [15], backstepping [16], and Lyapunov-based designs [17]. Together, these approaches show the common architecture of MBC systems. When using MBC theory in practice, the first step is to model or identify the plant. Then, using the *certainty equivalency principle*, controller design is based on the obtained plant model, assuming that the plant model accurately represents the actual system. Modeling and plant identification are therefore crucial elements of MBC theory.

In order to model a plant from first principles, parameters must be calibrated using measured data, either online or offline. Using identification theory, one can create a plant model within a model set that either fully captures the true system or comes close to it [10].

Regardless of whether modeling is conducted through first principles or data identification, it inherently involves approximating the true system, inevitably incurring some error [18]. Unmodeled dynamics are invariably present in the modeling process, contributing to potential inaccuracies in closed-loop control systems designed using model-based control (MBC) approaches [19].

Efforts have been directed towards enhancing robustness against model errors while retaining the advantages of MBC design. Robust control approach has explored different ways to describe errors of the model in closed-loop system configurations, including additive and multiplicative descriptions, as well as assumptions regarding modeling errors or noises. Nevertheless, the uncertainty characterizations utilized in robust theory design techniques often do not align with those derived from physical-mathematical modeling and identification techniques [10].

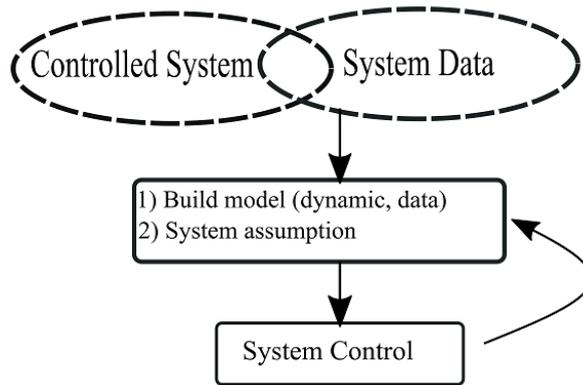
Challenges persist in establishing a perfect control theory, primarily due to the inherent trade-off between unmodeled dynamics and robustness. Achieving high accuracy in plant modeling requires significant effort and cost, and there is no efficient means of producing an accurate model. Additionally, addressing complexities such as parameter variations or structural changes over time poses further challenges. High-order system dynamics may necessitate controller order reduction for practical implementation, highlighting the paradox of striving for accuracy in modeling only to simplify the model for control design [19].

### 2.2.2.2 Model-based control design

The principle of certainty equivalency, a cornerstone of model-based control (MBC) theory, assumes that controllers are effective only if designed based on a plant model within the assumed set. Deviations from this set cause poor performances and system instability due to the sensitivity of closed-loop systems to even minor modeling errors [20].

Even with an accurate model, theoretical analyses covering stability, convergence, and robustness may yield limited insights if there are unfounded presumptions about the system. The MBC theory's architecture, illustrated in Fig. 2.1, underscores the system model's central role and assumptions in controller design and system analysis. However, a discrepancy persists between the model constructed based on assumptions and the actual plant, potentially undermining the effectiveness of MBC approaches if overlooked during design and analysis.

Nonlinear control systems design methods such as Lyapunov-based approaches, backstepping, and feedback linearization rely heavily on accurate plant models. However, the unmodeled dynamics' existence uncertainty can undermine the effectiveness of these



**Figure 2.1:** Architecture of MBC theory

methods. Inaccuracies in the model's structure and dynamic equations can significantly impact system performance since controllers incorporate these details [21]. The significant disparity between the elegant theories of model-based control (MBC) and practical implementation poses a major challenge, leading to various issues in real-world applications. When the model is unavailable, or its assumptions are invalid, it becomes challenging to make significant conclusions. The reliance of MBC on the model underscores its characterization as a "model theory" rather than solely a control theory to some extent.

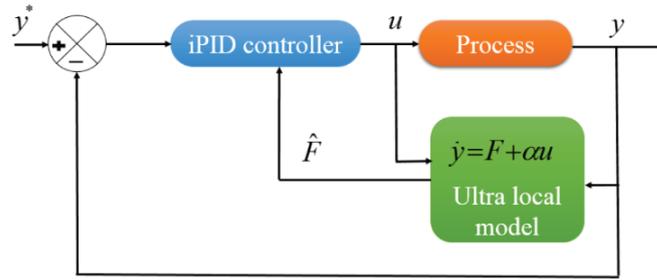
### 2.2.2.3 Model-free control design

This section provides an introduction to "intelligent" PID controllers (iPIDs), which are a counterpart to "model-free control" in a unified manner, considering recent advances [22]. The approach integrates basic differential algebra with outdated analytical techniques. A new online parameter estimation method simplifies identification methods. The significance of iPIDs, particularly iP, is highlighted in dealing with friction. The prevalence of classic PID controllers in industry and their challenges in complex tuning situations are attributed to their connection with iPIDs, as demonstrated through elementary sampling. Chapter 4.1 presents a detailed mathematical explanation of the approach.

Let's review some key theoretical concepts underlying the model-free control approach. Our focus is on systems that have a single control variable  $u$  and a single output variable  $y$  for simplicity's sake. Instead of relying on a complex mathematical model, we adopt an ultra-local model represented in Fig 2.2 as:

$$y^{(\nu)} = F + \alpha u \quad (2.1)$$

where,  $F$  is updated regularly and encompasses both the various potential disturbances and the unappreciated plant parts despite requiring any differentiation,  $\alpha$  is constant, and



**Figure 2.2:** Architecture of MFC theory

$u$  represents the input control signal. The estimation:

- requires only a brief period of time.
- is characterized by algebraic equations that incorporate low-pass filters such as repeated time integrals.
- is robust to appropriately significant noises.

The determined input-output data of the controlled plant is the only factor that influences the controller design in the MFC methodology. Fig. 2.1 illustrates the MFC methodology's architecture. One of this system's features is that it solely uses the measurement of I/O data from the closed-loop systems for both control system analysis and MFC controller design. The plant model vanishes and loses its ability to control the procedure. The closed-loop control system's measurement of I/O data serves as both a standard for control system performance and the origin of controlled problems. More details will be discussed in Chapter 2.

### 2.2.3 Model-free control applications

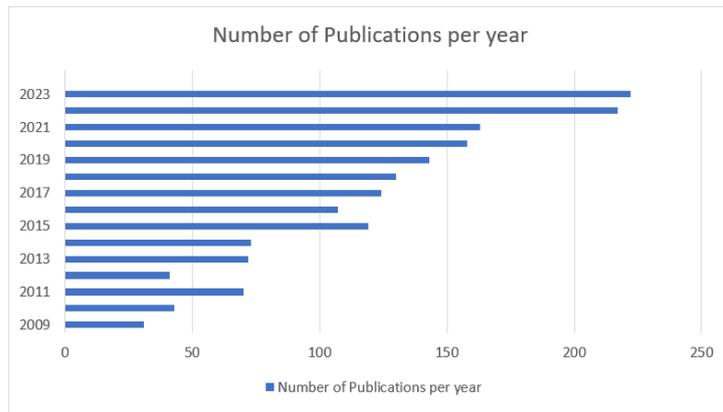
With the advancement of technology and information science, industries such as chemical, metallurgical, electronics, machinery, transportation, and logistics have experienced significant transformations. Those sectors operate on a large scale with complex production technologies and equipment, making modeling processes employing traditional rules or identification methods increasingly challenging. Consequently, conventional model-based control (MBC) approaches have become ineffective in addressing control problems in such industries. Moreover, huge quantities of data are continuously generated by modern industrial processes, and this data contains important details about the states of the equipment and the operations of the processes. Leveraging this data, both online and

offline, for controller design, state prediction, performance evaluation, decision-making, and fault diagnosis becomes crucial, especially without having precise process models. Thus, the development of model-free control theory (MFC) has emerged as a pressing issue in both theory and application domains.

The term "Model-free" has recently gained traction in the control community. While several MFC methods exist, they are often referred to by various identities such as data-driven control, modelless control, data-based control, iterative feedback tuning (IFT)[3], virtual reference feedback tuning (VRFT)[23], active disturbance rejection (ADRC) [24] and iterative learning control (ILC)[25]. Although these terms are used interchangeably, there are subtle differences between them.

Several publications in a wide variety of fields have demonstrated the efficiency and simplicity of MFC approach, we will mention here the most interesting ones.

Among the first tests of MFC was the DC-DC converter system [26]. MFC have been applied to Quadrotor Vehicle Quadrotor Vehicle by Younes [27]. Application reported by Frédéric Lafont includes information on a concrete greenhouse and comparisons with Boolean controllers[28]. A proposed algorithm were introduced based on an integrated structure between the Nonlinear Integral-Backstepping technique (NIB) and the MFC in [29] for a Quadrotor UAV. MFC technique has been also introduced in medical field to solve the problem of regulating glycemia of Type-1 Diabetes(T1D) [30] and the problem of immune therapy [31]. A novel cascade model free adaptive control method (MFAC) with an intelligent PID (i-PID) algorithm is proposed for the attitude adjustment of a multi-input multi-output (MIMO) quadrotor aircraft [32].In order to control the attitude and position of a quadrotor whose model includes parameter variations, uncertainties, and external disturbances, Haoping developed a model-free-based terminal sliding-mode control (MFTSMC) strategy. This method combines a sliding-mode technique with a model-free control approach, allowing for the elimination of tracking error in a finite amount of time [33]. Xinyi Zhang considers a model-free based neural network control with time-delay estimation (TDE-MFNNC) for lower extremity exoskeleton [34]. Raul-Cristian proposes a hybrid controller that consists of a second order data-driven Model-Free Control (MFC), and a Takagi-Sugeno Fuzzy (TSF) logic controller for nonlinear Multi Input-Multi Output (MIMO) twin rotor aerodynamic systems (TRASs) [35]. In [36] Mircea-Bogdan presents a novel model-free trajectory tracking of multiple-input multiple-output (MIMO) systems by the combination of iterative learning control (ILC) and primitives where the optimal trajectory tracking solution is obtained in terms of previously learned solutions to simple tasks called primitives. Kaihui Zhao suggests a robust model-free nonsingular terminal sliding-mode control (MFNTSMC) algorithm based on the ultra-local model to reduce the influence of permanent magnet (PM) demagnetization for PM synchronous motors



**Figure 2.3:** The number of annual publications related to MFC since 2009

(PMSMs) [37]. In [38], model-free fractional-order nonsingular fast terminal sliding mode control (MFF-TSM) based on time delay estimation (TDE) is proposed for the lower-limb robotic exoskeleton in the presence of uncertainties and external disturbance. Ghazally I.Y comprises the model-free based intelligent PID controller, the fuzzy logic controller, the time-delay estimation, and the particle swarm optimization to control the vibration of an active vehicle suspension systems [39]. Recently, Fliess and Cédric introduced MFC as a Service in the industrial internet of Things and machine learning [40, 41]. Figure. 2.3 shows the number of annual publications related to MFC since 2009.

This thesis introduces and validates the Model-free approach to variable speed wind turbine in Chapter 3.

## 2.3 Flatness Control

This section presents the key point of flatness control lies in its ability to ensure that a dynamic system exhibits flat outputs. Flatness control aims to manipulate a system's inputs in such a way that the outputs become simple functions of a few "flat" variables. These flat outputs essentially capture all the relevant information about the system's behavior. This approach offers advantages in terms of simplifying control design, facilitating trajectory planning, and enhancing robustness to disturbances.

### 2.3.1 Fundamentals of Flatness

The concept of flatness emerged from the need to address the challenges posed by nonlinear systems in control engineering. Traditional control techniques struggled with the complexity of nonlinear dynamics, motivating researchers to seek alternative approaches. In the 1980s and early 1990s, researchers were seeking ways to simplify the control of nonlinear systems.

In 1995, a group of researchers published a seminal paper in the International Journal of Control titled "Flatness and Defect of Nonlinear Systems: Introductory Theory and Examples," exploring the concept of "flatness" as a means to achieve this goal. The paper laid out the basic principles of flatness control and provided illustrative examples [5] and marked a significant milestone in the formalization of flatness theory. Another important paper, published in 1999 in the IEEE Transactions on Automatic Control, titled "A Lie-Bäcklund approach to equivalence and flatness of nonlinear systems," further advanced the theory by introducing Lie-Bäcklund transformations for analyzing the equivalence and flatness of nonlinear systems [42].

Flatness control operates on the principle of transforming a nonlinear dynamic system into a form where the outputs are described by flat variables. These flat outputs are chosen to simplify control design and trajectory planning. By exploiting the flatness property, control engineers can design controllers that directly manipulate these flat outputs, thereby simplifying control synthesis and trajectory tracking tasks.

### 2.3.2 Mathematical Formulation

By introducing the concepts of flatness and linearizing output, a novel perspective on the complete linearization issue through dynamic feedback is put forth [15]. This section aims to demonstrate how beneficial such a viewpoint can be for movement tracking [5, 42, 43, 44]. The late 1980s saw the invention of flatness-based control and differential flatness theory. Since then, these concepts have continuously progressed, offering effective answers to complex control and state estimation problems [45].

A differentially flat system is defined as follows: Since  $f$  is a smooth vector field, a system  $\dot{x} = f(x, u)$  with state vector  $x \in R^n$  and input vector  $u \in R^m$  is differentially flat if there is a vector  $y \in R^m$  with the form:

$$y = h(x, u, \dot{u}, \dots, u^{(r)}) \quad (2.2)$$

That is,

$$\begin{aligned} x &= \phi(y, \dot{y}, \dots, y^{(q)}) \\ u &= \alpha(y, \dot{y}, \dots, y^{(q)}) \end{aligned} \quad (2.3)$$

as smooth functions  $h, \phi$ , and  $\alpha$ . Accordingly, the  $m$  algebraic variables  $y_i, i = 1, 2, \dots, m$  describe the new system.  $y = h(x, u, \dot{u}, \dots, u^{(r)})$  was the definition of the flat output provided above. Zero-flat systems are those in which the flat output is solely a function of the state vector  $x$ . Nevertheless, it could be necessary to represent the flat output as a function of the control  $u$  and its derivatives in addition to the state vector  $x$ .

This necessity arises in more complex scenarios, such as when dealing with dynamic feedback linearization or applying dynamic extension techniques. In such cases, the system's state vector is extended by including the control inputs and their derivatives as additional state variables. This extension allows for a more comprehensive and flexible description of the system, accommodating the intricacies of control and state estimation problems. By incorporating these elements, we ensure that the flat output adequately captures the dynamic behavior of the system, facilitating more effective control strategies and solutions.

Equation (2.3) demonstrates that the flat output and its derivatives can be used to express the state vector of the differentially flat system and its control inputs. In the study of differential flatness, the fundamental question is whether there exists a function  $y = h()$  given by  $y = h(x, u, \dot{u}, \dots, u^{(r)})$ , such that the state vector of the system  $x$  and the control input  $u$  can be stated as functions of  $y$  and of its derivatives, as in Eq. (2.3). This is given by the differential equations that describe the nonlinear system dynamics  $\dot{x} = f(x, u)$ .

### 2.3.3 Literature review of Flatness applications

Flatness control is a versatile and effective technique with applications spanning diverse domains, showcasing its broad applicability in optimizing system performance and achieving precise control objectives. Its utility is evident in stabilizing and maneuvering quadrotor UAVs for precise trajectory tracking, allowing these unmanned aerial vehicles to navigate through complex environments with ease [46]. Additionally, it enhances the navigation capabilities of autonomous underwater vehicles, enabling them to explore underwater terrain accurately and efficiently [47]. In industrial settings, flatness control facilitates precise manipulation of robotic arms [48], ensuring accurate assembly and manufacturing processes [49]. Moreover, it plays a pivotal role in optimizing wind turbine energy capture, maximizing power generation efficiency while minimizing structural loads [50]. In hybrid energy systems, such as those combining fuel cells and renewable sources, flatness control manages power flow to maximize energy output and system stability [51]. It also controls electric vehicle powertrains, optimizing power distribution for enhanced performance and range [52]. In aerospace applications, flatness control maintains satellite orientation in space missions, ensuring precise positioning and stabilization [53]. Additionally, it regulates industrial processes like chemical reactors and distillation columns, optimizing production efficiency and quality [54]. In aircraft flight dynamics, flatness control enables autonomous navigation and altitude control, ensuring safe and efficient flight operations [55]. Furthermore, it finds applications in smart grid energy management [52], HVAC systems

[54], biomedical systems[56], water treatment plants [57], renewable energy microgrids [58], and traffic management systems [59], contributing to improved efficiency, reliability, and sustainability across various industries and applications.

## 2.4 Combination of Flatness with other control strategies

Combining control techniques can indeed lead to improved system performance and robustness. Here are a few research papers that demonstrate the integration of flatness/model-free control approaches with other control techniques.

The integration of flatness control with various other control techniques offers significant added value across different applications. In traditional control methods, such as adaptive [60] control and proportional-derivative-integral (PID) control [61], combining them with flatness control enhances robustness and adaptability, ensuring precise motion control despite uncertainties. Additionally, integrating flatness control with fuzzy logic [62] or backstepping control [63] enables robust and adaptive regulation of system dynamics, particularly in complex systems like DC motor speed control or hydraulic servo systems. Moreover, combining flatness control with modern techniques like reinforcement learning [64] or model predictive control [65, 66] enhances system performance and autonomy. Reinforcement learning enables the system to learn optimal control policies, while model predictive control anticipates future behavior, both ensuring efficient trajectory tracking and navigation, as seen in applications such as autonomous vehicles or wind turbines. Furthermore, incorporating flatness control with fault detection techniques like sliding mode observers[67] improves system reliability by enabling real-time fault detection and isolation in electromechanical systems. Finally, integrating flatness control with deep reinforcement learning or artificial neural networks [68, 69] enhances adaptability and efficiency, maximizing performance in tasks such as solar tracking or quadcopter trajectory tracking. In essence, these combinations leverage the strengths of each method to achieve superior control performance, stability, and robustness across diverse applications.

This thesis significantly contributes by showcasing the innovative integration of the differential flatness approach with the model-free control technique. Combining flatness-based control with model-free control (MFC) offers a synergistic approach that leverages the structured framework of flatness-based control for system modeling and analysis while harnessing the adaptability and resilience of MFC to uncertainties and disturbances. This integration enhances adaptability, robustness, and performance benefits, as flatness-based control ensures precise system output regulation and decoupling of nonlinear dynamics,

while MFC optimizes control actions in real-time based on online observations. By merging these methodologies, control strategies become more versatile, future-proofed, and capable of achieving superior performance in dynamic and uncertain environments, making them particularly valuable for applications such as wind energy conversion systems. This is deeply discussed in Chapter 4.

# Chapter 3

## Algebraic and Derivative Estimation

### 3.1 Introduction

In the field of system identification, accurate parameter estimation is crucial for interpreting the dynamics of systems and establishing efficient control strategies. Conventional approaches frequently depend on iterative numerical optimization procedures, which can be computationally demanding and susceptible to noise. Algebraic estimating approaches present an attractive choice as they offer both rapidity and precision in parameter estimation.

In 2004, algebraic estimation techniques were introduced which led to a significant change in the field of system identification. Their approach, centered around polynomial modeling, offered an original take on the subject. Instead of using iterative optimization methods, these techniques approach the parameter identification problem as an algebraic one, allowing for direct estimation of system coefficients using input-output data. This method is especially beneficial for linear systems, as the connection between inputs and outputs can be expressed through polynomial functions [70].

Mboup and Join extended the algebraic framework to compute individual estimators for each coefficient in the polynomial model. Derivative estimation is a technique that overcomes the difficulties associated with simultaneously estimating values, such as the issue of calculating inverses of poorly conditioned matrices. These issues can lead to inaccurate results, especially when there is interference. The proposed derivative estimation computes the derivatives of a system by analyzing each coefficient separately. This approach guarantees that the estimations obtained are robust and reliable [71].

Based on Mboup, Join, and Fliess' approach, the following research has advanced derivative estimation in the algebraic context. According to Mboup, Join, and Fliess [72], sliding-window estimation methods adapt the estimation window size to system dynamics.

This dynamic technique makes derivative estimate algorithms more resilient, especially when system parameters change. Recent advances in machine learning and optimization have integrated algebraic and data-driven methods to improve complex system parameter identification accuracy and efficiency.

This chapter delves into innovative state observation methods within the scope of model-free control. Specifically, it introduces derivative and algebraic estimators for state observation to the ultra-local model implementation. The estimation process utilizes Taylor's expansion coefficients of any order, providing a framework for parameter estimation. The proposed approach is discretely implemented using operational mathematical concepts and FIR filters.

The study conducts a comparative analysis between derivative and algebraic estimators, examining their performance in state observation. Tests are conducted on a DC motor's position trajectory control strategy under disturbances and uncertainty. The ultra-local model employed in this study solely relies on system input and output measurements, making it independent of initial conditions. Furthermore, the proposed design aims to minimize nonlinearities and enhance Coulomb friction robustness, as demonstrated through empirical analysis.

## 3.2 Derivative Estimation

The identification of the parameters of a constricted Taylor expansion of random order is the basis of the derivative estimation approach. To accomplish this, it is necessary to suppose that the output signal  $\tilde{x}(t)$  represents an analytic variable and may therefore be represented using Taylor series that is truncated at  $t = 0$ .

$$\tilde{x}(t) \approx x(t) = \sum_{j=0}^n \frac{x^{(j)}(0)}{j!} t^j \quad (3.1)$$

Identifying these coefficients requires math procedures in a domain similar to the Laplace transform. After converting the expression again to the time domain, the estimation filters are discretized into sliding-window finite impulse response filters for applications in real-time.

A more comprehensive formulation for estimators of arbitrary order can be found in the works [72, 73]. Throughout this thesis, we will offer a detailed explanation of the development of these estimators, aiming to clarify the fundamental concepts of the algebraic technique.

### 3.2.1 1<sup>st</sup> Order estimation

This section presents the filtering and derivation of first-order estimation:

#### 3.2.1.1 Filtering of first order estimation

Let's delve into the initial estimation of the noisy output signal  $\tilde{x}(t)$  near  $t = 0$ :

$$x(t) = a_0 + a_1 t \quad (3.2)$$

The transformation to the Laplace form results in the operational form of Equation (4.2) being expressed as follows:

$$X(s) = \frac{a_0}{s} + \frac{a_1}{s^2} \quad (3.3)$$

By factoring every component of the equation by  $s^2$ , we obtain:

$$s^2 X(s) = sa_0 + a_1 \quad (3.4)$$

This allows us to separate and eliminate  $a_1$  by multiplying both sides with the differentiating operator  $\frac{d}{ds}$ :

$$2sX(s) + s^2 \frac{d}{ds} X(s) = a_0 \quad (3.5)$$

To avoid time derivatives and filtering, the multiplication of both sides by  $\frac{1}{s^3}$  is performed. Each factor involving  $X(s)$  is integrated at least one time:

$$2 \frac{1}{s^2} X(s) + \frac{1}{s} \frac{d}{ds} X(s) = \frac{a_0}{s^3} \quad (3.6)$$

Transforming the right side of the expression back to the time domain:

$$\frac{d}{s^\alpha}, \alpha \geq 1, d \in \mathbb{D} \longleftrightarrow d \frac{t^{\alpha-1}}{(\alpha-1)!} \quad (3.7)$$

By utilizing the Cauchy equation for iterated integration [74], we can derive an expression in closed form to feed the multiplier  $\frac{1}{s^\alpha} \frac{d^n}{ds^n}$  in just one integral. This is possible because differentiation of  $X(s)$  according to  $s$  corresponds to factorization by  $-t$  in the time domain, and left factorization  $X(s)$  by  $\frac{1}{s^\alpha}, \alpha \geq 1$  corresponds to repeated integrals.

$$\frac{1}{s^\alpha} \frac{d^n}{ds^n} X(s) \longleftrightarrow \frac{(-1)^n}{(\alpha-1)!} \int_0^t (t-\tau)^{\alpha-1} \tau^n x(\tau) d\tau \quad (3.8)$$

By applying the above relation, Equation (2.6) can be converted again into the time domain:

$$2 \int_0^t (t - \tau)x(\tau)d\tau - \int_0^t \tau x(\tau)d\tau = a_0 \frac{t^2}{2!} \quad (3.9)$$

By restructuring the formula, we may identify the factor  $a_0$  as:

$$a_0 = \frac{2}{t^2} \int_0^t (2t - 3\tau)x(\tau)d\tau \quad (3.10)$$

By replacing the function  $x(t)$  with  $\tilde{x}(t)$ , Equation (4.10) shall be viewed as being a first-order filtered estimation of the signal. This form of approximation is particularly useful for estimating the signal over short time intervals when  $t > 0$ . Additional information regarding the digital implementation of sliding-window estimators will be presented in the following section.

### 3.2.1.2 Derivation of first-order estimation

Let's analyze the first estimation of the output noisy signal  $\tilde{x}(t)$  at  $t = 0$ , as previously described in  $s$  domain:

$$X(s) = \frac{a_0}{s} + \frac{a_1}{s^2} \quad (3.11)$$

Factoring in each side by  $s$ , we acquire:

$$sX(s) = a_0 + \frac{a_1}{s} \quad (3.12)$$

This allows us to isolate  $a_0$ , which can be removed by factoring with the differentiation operation  $\frac{d}{ds}$ :

$$X(s) + s \frac{d}{ds} X(s) = -\frac{a_1}{s^2} \quad (3.13)$$

Further factoring by  $\frac{1}{s^2}$  aims to eradicate filtering and time derivativion, with each element involving  $X(s)$  being integrated once:

$$\frac{X(s)}{s^2} + \frac{1}{s} \frac{d}{ds} X(s) = -\frac{a_1}{s^4} \quad (3.14)$$

By utilizing the established principles of Cauchy formula and Laplace Transformation

for repetitive integration, we derive:

$$\int_0^t (t - \tau)x(\tau)d\tau - \int_0^t \tau x(\tau)d\tau = -a_1 \frac{t^3}{3!} \quad (3.15)$$

By restructuring the equation, we can find the first estimate of the derivative:

$$a_1 = -\frac{6}{t^3} \int_0^t (t - 2\tau)x(\tau)d\tau \quad (3.16)$$

### 3.2.2 2<sup>nd</sup> Order estimation

This section presents the filtering and derivation of second-order estimation:

#### 3.2.2.1 Filtering of second order estimation

Let us examine the second-order estimation of the noisy signal  $\tilde{x}(t)$  at  $t = 0$ :

$$x(t) = a_0 + a_1 t + a_2 \frac{t^2}{2} \quad (3.17)$$

Employing the Laplace Transform:

$$X(s) = \frac{a_0}{s} + \frac{a_1}{s^2} + \frac{a_2}{s^3} \quad (3.18)$$

Factoring by  $s^3$  to isolate  $a_2$  :

$$s^3 X(s) = s^2 a_0 + s a_1 + a_2 \quad (3.19)$$

Employing  $\frac{d}{ds}$  to eliminate  $a_2$ :

$$3s^2 X(s) + s^3 \frac{d}{ds} X(s) = 2s a_0 + a_1 \quad (3.20)$$

Further application of  $\frac{d}{ds}$  to eliminate  $a_1$ :

$$6sX(s) + 6s^2 \frac{d}{ds} X(s) + s^3 \frac{d^2}{ds^2} X(s) = 2a_0 \quad (3.21)$$

Factoring by  $\frac{1}{s^4}$  to eliminate filtering and time derivatives:

$$6 \frac{1}{s^3} X(s) + 6 \frac{1}{s^2} \frac{d}{ds} X(s) + \frac{1}{s} \frac{d^2}{ds^2} X(s) = 2a_0 \frac{1}{s^4} \quad (3.22)$$

Returning to time domain transformation:

$$3 \int_0^t (t - \tau)^2 x(\tau) d\tau - 6 \int_0^t (t - \tau) x(\tau) d\tau + \int_0^t \tau^2 x(\tau) d\tau = 2a_0 \frac{t^3}{3!} \quad (3.23)$$

The second-order filter is acquired by arranging terms:

$$a_0 = \frac{3}{t^3} \int_0^t (3t^2 - 12t\tau + 10\tau^2) x(\tau) d\tau \quad (3.24)$$

### 3.2.2.2 Derivation of first-order estimation

Let's explore the second-order estimation of the noisy signal  $\tilde{x}(t)$  which has already been described in  $s$  domain:

$$X(s) = \frac{a_0}{s} + \frac{a_1}{s^2} + \frac{a_2}{s^3} \quad (3.25)$$

Factoring by  $s^3$  to isolate  $a_2$  :

$$s^3 X(s) = s^2 a_0 + s a_1 + a_2 \quad (3.26)$$

Utilizing differentiation with relative to  $s$  to exlude  $a_2$ :

$$3s^2 X(s) + s^3 \frac{d}{ds} X(s) = 2s a_0 + a_1 \quad (3.27)$$

Factoring by  $\frac{1}{s}$  to isolate  $a_0$ :

$$3s X(s) + s^2 \frac{d}{ds} X(s) = 2a_0 + \frac{a_1}{s} \quad (3.28)$$

Further differentiation relative to  $s$ , aiming to eliminate  $a_0$ :

$$3X(s) + 5s \frac{d}{ds} X(s) + s^2 \frac{d^2}{ds^2} X(s) = -a_1 \frac{1}{s^2} \quad (3.29)$$

Factoring by  $\frac{1}{s^3}$ :

$$3 \frac{1}{s^3} X(s) + 5 \frac{1}{s^2} \frac{d}{ds} X(s) + \frac{1}{s} \frac{d^2}{ds^2} X(s) = -a_1 \frac{1}{s^5} \quad (3.30)$$

Returning to time domain transformation:

$$\frac{3}{2} \int_0^t (t - \tau)^2 x(\tau) d\tau - 5 \int_0^t (t - \tau) \tau x(\tau) d\tau + \int_0^t \tau^2 x(\tau) d\tau = -a_1 \frac{t^4}{4!} \quad (3.31)$$

By restructuring the equation, we can find the first estimate of the derivative  $a_1$ :

$$a_1 = -\frac{12}{t^4} \int_0^t (3t^2 - 16t\tau + 15\tau^2) x(\tau) d\tau \quad (3.32)$$

### 3.2.2.3 Derivation of second-order estimation

Let's explore the second-order estimation of the noisy signal  $\tilde{x}(t)$  which has already been described in  $s$  domain:

$$X(s) = \frac{a_0}{s} + \frac{a_1}{s^2} + \frac{a_2}{s^3} \quad (3.33)$$

Factoring by  $s^2$  to isolate  $a_1$  :

$$s^2 X(s) = sa_0 + a_1 + \frac{a_2}{s} \quad (3.34)$$

Applying the differential operator  $\frac{d}{ds}$  to eliminate  $a_1$  gives:

$$2sX(s) + s^2 \frac{d}{ds} X(s) = a_0 - \frac{a_2}{s^2} \quad (3.35)$$

Further application of the differential operator  $\frac{d}{ds}$  to eliminate  $a_0$  results in:

$$2X(s) + 4s \frac{d}{ds} X(s) + s^2 \frac{d^2}{ds^2} X(s) = 2a_2 \frac{1}{s^3} \quad (3.36)$$

Using  $\frac{1}{s^3}$  as a scaling factor, we get:

$$2 \frac{1}{s^3} X(s) + 4 \frac{1}{s^2} \frac{d}{ds} X(s) + \frac{1}{s} \frac{d^2}{ds^2} X(s) = 2a_2 \frac{1}{s^6} \quad (3.37)$$

Returning to time domain transformation:

$$\int_0^t (t - \tau)^2 x(\tau) d\tau - 4 \int_0^t (t - \tau) \tau x(\tau) d\tau + \int_0^t \tau^2 x(\tau) d\tau = 2a_2 \frac{t^5}{5!} \quad (3.38)$$

restructuring the equation provides the second-order derivative's estimate:

$$a_2 = \ddot{x}(t) = \frac{60}{t^5} \int_0^t (t^2 - 6t\tau + 6\tau^2) x(\tau) d\tau \quad (3.39)$$

After a series of mathematical operations, we've derived expressions for estimating the second-order derivatives of a noisy signal  $\tilde{x}(t)$  around  $t = 0$ . By manipulating the Laplace Transform representation in the operational domain and performing differential operations, we obtained equations in terms of  $a_0$ ,  $a_1$ , and  $a_2$ , the coefficients of the signal approximation. These equations were then transformed back to the time domain to obtain

practical formulas for estimating the second-order derivative  $\ddot{x}(t)$ . These expressions involve integrals of the signal  $x(t)$  over specific ranges, providing a method to estimate the curvature of the signal at  $t = 0$ . These results can be particularly useful in signal processing and system identification tasks that will be used in estimating the ultra-local model.

### 3.2.3 Implementation of the Derivative Estimator in Discrete Form

The estimators formulated in the preceding section target the estimation of the  $j$ -order derivative of a given noisy signal  $\tilde{x}(t)$ , which is represented by an  $n$ -order Taylor expansion evaluated at  $t = 0$ :

$$\tilde{x}(t) \simeq x(t) = \sum_{j=0}^n x^{(j)}(0) \frac{t^j}{j!} \quad (3.40)$$

The estimations hold true within a limited time frame  $\epsilon$ . In a broader context, the  $j$ -order derivative estimator, where every component associated with  $x(t)$  undergoes at least one integration, is articulated as follows:

$$x^{(j)}(t) \simeq x^{(j)}(0) = a_j = \frac{k}{q(t)} \int_0^t \tilde{P}(t, \tau) x(\tau) d\tau, 0 < t < \epsilon \quad (3.41)$$

This method can be applied within a short time frame of fixed duration  $T$ :

$$a_j = \frac{k}{q(T)} \int_0^T \tilde{Q}(T, \tau) x(\tau) d\tau = \int_0^T \frac{k\tilde{Q}(T, \tau)}{q(T)} x(\tau) d\tau = \int_0^T Q(T, \tau) x(\tau) d\tau \quad (3.42)$$

The equation presented provides an estimate for the parameter  $a_j = x^{(j)}(0)$ , which is accurate only at  $t = 0$ . It relies on a Taylor series expansion centered at  $t = 0$ . While this estimate is exact for noise-free polynomial input signals, it gradually deviates for arbitrary signals as time progresses. This deviation arises from assuming that a polynomial near  $t$  can effectively approximate the input signal. It's crucial to note that this estimator is non-causal, meaning that to estimate the derivative at  $t = 0$ , function values  $x(t)$  for  $t > 0$  are necessary. This necessity prompts an approach adjustment to create a causal estimator at any given time  $t$ . Initially, let's explore the Taylor expansion  $r(\tau)$  centered at  $\tau = 0$  of a random signal  $\tilde{r}(\tau)$ . The coefficients of this expansion may be established using the estimators discussed afterward:

$$\tilde{r}(\tau) \simeq r(\tau) = \sum_{j=0}^n a_j \frac{\tau^j}{j!} \Rightarrow a_j = \int_0^T Q(T, \tau) r(\tau) d\tau \quad (3.43)$$

Taking into account the variable transformations,  $\tau \triangleq t - \delta$  and  $x(\delta) \triangleq r(t - \delta)$  :

$$x(\delta) \triangleq r(t - \delta) = \sum_{j=0}^n a_j \frac{(t - \delta)^j}{j!} \quad (3.44)$$

The preceding equation is a Taylor expansion of  $\tilde{r}(\tau)$  near  $\delta = t$ , as well as a signal transformation of the variable  $\delta$ . It's straightforward to demonstrate that:

$$x^{(j)}(t) = \left. \frac{d^j}{d\delta^j} (r(t - \delta)) \right|_{\delta=t} = (-1)^j a_j \quad (3.45)$$

Rewriting the expression for  $a_j$  with the changed variable :

$$a_j = \int_0^T Q(T, \tau) r(\tau) d\tau = \int_0^T Q(T, \tau) x(t - \tau) d\tau \quad (3.46)$$

The ultimate equation represents a causal estimator computed at a flexible time  $t$ , employing a constant estimation window of duration  $T$ :

$$a_j = x^{(j)}(t) = (-1)^j \int_0^T Q(T, \tau) x(t - \tau) d\tau \quad (3.47)$$

The digital realization of this estimator entails employing the trapezoidal integration approach to construct a digital Finite Impulse Response (FIR) filter. This process involves specifying  $T = T_s N$ , where  $T$  denotes the duration of the estimation window,  $T_s$  signifies a constant sampling interval and  $(N + 1)$  represents the total number of samples. By leveraging this approach, the estimator can be seamlessly integrated into digital systems, providing accurate derivative estimations within a defined time window:

$$a_j = x^{(j)}(t) \simeq (-1)^j \sum_{k=1}^{N+1} \alpha_k Q(T, \tau_k) x(t - \tau_k) \quad (3.48)$$

To minimize computational efficiency, the discretization of  $Q(T, \tau_k)$  is precomputed. Simultaneously, after the trapezoidal integration, we consider the coefficients  $\alpha_k$ :

$$\hat{Q}_{\alpha_k}(T, \tau_k) = \begin{bmatrix} \alpha_1 Q(T, \tau_1) & \alpha_2 Q(T, \tau_2) & \dots & \alpha_{N+1} Q(T, \tau_{N+1}) \end{bmatrix}$$

$$\tau_k = (k - 1)T_s$$

$$\alpha_k = \frac{T_s}{2}, k = 1 \text{ or } k = N + 1$$

$$\alpha_k = T_s, k = 2, 3, \dots, N$$
(3.49)

The vector  $\hat{x}(t - \tau_k)$ , housing the latest  $N + 1$  samples, is arranged vertically. With each sample clock cycle, the entire vector undergoes a shift operation. This process involves discarding the oldest value while incorporating the most recent value  $x(t)$  into the initial position of the vector. By employing this mechanism, the vector continuously maintains the most recent samples, facilitating efficient real-time processing of the input signal:

$$\hat{x}(t - \tau_k) = \begin{bmatrix} x(t) \\ x(t - T_s) \\ \vdots \\ x(t - T_s N) \end{bmatrix}$$
(3.50)

Ultimately, at each sample time, the product of  $\hat{Q}_{\alpha_k}(T, \tau_k)$  and  $\hat{x}(t - \tau_k)$  yields a real-time approximation of  $a_j$ :

$$a_j(t) = x^{(j)}(t) \simeq (-1)^j \sum_{k=1}^{N+1} \alpha_k Q(T, \tau_k) x(t - \tau_k) = (-1)^j \hat{Q}_{\alpha_k}(T, \tau_k) \hat{x}(t - \tau_k)$$
(3.51)

*Remark (1).* **A compromise between accuracy and immunity to noise:**

- The choice of the estimation window  $T$  plays a crucial role in striking a balance between immunity to noise and accuracy in derivative estimation.
- Lower  $T$  values offer advantages such as faster filter response and improved local signal modeling.
- Conversely, larger values of  $T$  enhance noise immunity and provide better high-frequency filtering properties.

*Remark (2).* **Impact of window Size on filter response**

- Smaller  $T$  values contribute to a more agile filter response, allowing it to adapt quickly to rapid changes in the input signal.

- This agility is particularly advantageous in scenarios where capturing transient signal features or responding promptly to signal variations is essential.

*Remark (3).* **Enhanced noise immunity with larger window sizes:**

- Larger  $T$  values are beneficial for improving noise immunity, as they enable the filter to effectively suppress high-frequency noise components.
- By encompassing a broader time window, the filter can integrate information from a more extended temporal range, thereby enhancing its ability to distinguish between signal and noise.

*Remark (4).* **Numerical precision and sampling frequency**

- In addition to window size, numerical precision is a critical concern in derivative estimation algorithms.
- Increasing the sampling frequency is an alternative approach to enhance precision, but it may be limited by hardware constraints, particularly if the system is already operating at maximum frequency.

### 3.3 Ultra-local model estimation

In systems where the system output  $y$  is impacted by the system input  $u$ , governed by a non-linear yet essentially smooth function  $\mathfrak{S}$ , the relationship between these variables is encapsulated in a differentiation equation, as articulated in Equation (4.52) :

$$\mathfrak{S}(t, u, \dot{u}, \dots, u^{(m)}, y, \dot{y}, \dots, y^{(n)}) = 0 \quad (3.52)$$

Here, the equation captures the relationship between time  $t$ , input signal  $u$ , its derivatives up to  $m^{th}$  order, output signal  $y$ , and its derivatives to the  $n^{th}$  order. This formulation provides a comprehensive view of how the system dynamics evolve over time in response to input stimuli [71].

When there exists an index  $i$  within the range  $(0, n)$  for which  $\frac{\partial \mathfrak{S}}{\partial y^{(i)}} \neq 0$ , the absolute function theorem offers a local simplification, revealing the relationship:

$$y^{(i)} = \mathfrak{S}(t, u, \dot{u}, \dots, u^{(m)}, y, \dot{y}, \dots, y^{(i-1)}, y^{(i+1)}, \dots, y^{(n)}) \quad (3.53)$$

This equation unravels the  $i^{th}$  derivative of the output  $y$  as a function of input and output derivatives, providing valuable insights into the intricate dynamics governing the system's behavior.

Fliess's model-free control methodology [75] suggests using an ultra-local model instead of complex nonlinear models, as shown in Equation (4.53). Equation (4.54) demonstrates an ultra-local representation that simplifies system behavior:

$$y^{(\nu)}(t) = F(t) + \beta u(t) \quad (3.54)$$

To optimize transient performance,  $y^{(\nu)}$  represents the  $\nu$ th-order derivative of the system output. On the other hand,  $F$  summarizes the unmodeled dynamics of the system, including unknown perturbations, and is constantly updated in real time using algebraic or derivative-based approaches. The non-physical coefficient  $\beta$  is carefully chosen to match  $y^{(\nu)}$  and  $\beta u$  in terms of magnitude and sign, enhancing closed-loop performance through iterative experimentation.

This simplified representation is an effective approximation of system dynamics over short time intervals, providing a practical basis for control synthesis and eliminating the need for complex nonlinear modeling, consequently streamlining gain tuning processes.

In the closed-loop system, Equation (4.54) incorporates an intelligent PID controller, with  $F(t)$  expressed as:

$$F(t) = y^{(\nu)}(t) - \beta u(t) \quad (3.55)$$

The system control input is formulated as:

$$u = -\frac{F - y_d^{(\nu)} + u_c}{\beta} \quad (3.56)$$

Here,  $y_d^{(\nu)}$  symbolizes the  $\nu^{th}$  derivative of the output desired trajectory, while  $u_c$  represents the causal feedback control input.

Incorporating these equations yields a comprehensive expression for system behavior, ensuring effective tracking and convergence towards the reference trajectory, as delineated in Equations (4.57), (4.58), and (4.59).

$$y^{(\nu)} = F + \beta \left( -\frac{F - y_d^{(\nu)} + u_c}{\beta} \right) = y_d^{(\nu)} - u_c \quad (3.57)$$

Afterwards,

$$y^{(\nu)} - y_d^{(\nu)} + u_c = 0 \quad (3.58)$$

$$e^{(\nu)} + u_c = 0 \quad (3.59)$$

Here,  $e^{(\nu)}$  represents the  $\nu^{th}$  derivative error of the tracking error ( $e = y - y_d$ ). The control signal  $u_c$  is selected to ensure asymptotic tracking, where the output approaches

the desired trajectory as expressed by Equation (4.60),

$$\lim_{t \rightarrow +\infty} e(t) = 0 \quad (3.60)$$

### 3.3.1 Intelligent PIDs (iPIDs)

The distinction between the intelligent and conventional controllers is inspired by the work of D'Andréa Novel [76]. The PID control, i.e.,  $u$ , can be computed based on the error  $e$ . Let's reformulate the control signal in Equation (4.56) using a conventional PID controller:

$$u = -\frac{F - \ddot{y}_d + K_P e + K_I \int e + K_D \dot{e}}{\beta} \quad (3.61)$$

Here, we set  $v = 2$  in Equation (4.54). By substituting Equation (4.54) into Equation (4.61), we define the intelligent proportional-integral-derivative controller (iPID) as:

This equation omits  $F$ , thereby removing uncertainties and unmodeled plant dynamics. Equation (4.61) meets tracking requirements through the careful adjustment of gains  $K_P, K_I, K_D$ , resulting in a stable linear differential equation with constant real coefficients.

Introducing various combinations of controller gains in a proportional-integral-derivative (PID) controller design can result in different types of controllers with unique control characteristics.

- When  $K_P \neq 0, K_I = 0, K_D = 0$ , it results in an intelligent Proportional (iP) controller.
- When  $K_P \neq 0, K_I = 0, K_D \neq 0$ , it leads to an intelligent Proportional-Derivative (iPD) controller.
- When  $K_P \neq 0, K_I \neq 0, K_D \neq 0$ , it forms an intelligent Proportional-Integral-Derivative (iPID) controller.

Each of these configurations provides distinct control characteristics that are tailored to meet specific system requirements.

### 3.3.2 Online estimation of the ultra-local model through algebraic estimation

The process of identifying the parameter  $F$  which is the basis of the ultra-local model entails employing algebraic estimation techniques, a methodology outlined by [77]. Alternatively, numerical differentiation, which involves estimating derivatives, serves as another viable approach outlined in Section 4.2.1.

Let's investigate the implications of Equation (4.54) based on classical operational calculus principles. Here, we replace the parameter  $F$  with  $F_{approx(Alg)}$ , an approximation assumed to be constant over the interval  $[t, t + T]$ . To calculate  $F_{approx(Alg)}$ , we follow the previous systematic approach.

Setting  $v = 1$  in Equation (4.54), we initiate the process by performing a Laplace transformation:

$$sY(s) - y(0) = \frac{F_{approx(Alg)}}{s} + \beta U(s) \quad (3.62)$$

Next, we proceed to differentiate Equation (4.62) with respect to  $s$  to exclude  $y(0)$ :

$$Y(s) - s \frac{d}{ds} Y(s) = -s^{-2} F_{approx(Alg)} + \beta \frac{d}{ds} U(s) \quad (3.63)$$

To further refine our analysis, we factor Equation (4.63) by  $\frac{1}{s^2}$ , aiming to integrate every coefficient once. This action efficiently filters and removes time derivatives, resulting in a low-pass filter that effectively reduces noise:

$$\frac{1}{s^2} Y(s) + \frac{1}{s} \frac{d}{ds} Y(s) = -\frac{1}{s^{-4}} F_{approx(Alg)} + \beta \frac{1}{s^3} \frac{d}{ds} U(s) \quad (3.64)$$

After careful observation, we discover that differentiating  $Y(s)$  in the operational domain corresponds to factorization by  $(-t\hat{a}$  in the time domain. Furthermore, the left factorization of  $Y(s)$  by  $\frac{1}{s^\alpha}$ ,  $\alpha \geq 1$  is consistent with looping integrals. To obtain a closed-form equation, we employ the Cauchy formula for occurring again integration of the operator  $\frac{1}{s^\alpha} \frac{d^n}{ds^n}$ , which results in only one integral:

$$\frac{1}{s^\alpha} \frac{d^n}{ds^n} Y(s) \longleftrightarrow \frac{(-1)^n}{(\alpha - 1)!} \int_0^t (t - \tau)^{\alpha-1} \tau^n y(\tau) d\tau \quad (3.65)$$

Finally, after a series of mathematical manipulations, we derive the approximated time domain equation (4.64) as:

$$F_{approx(Alg)} = \frac{-6}{t^3} \int_0^t (t - 2\tau) y(\tau) d\tau - \frac{6\beta}{t^3} \int_0^t \tau(t - \tau) u(\tau) d\tau \quad (3.66)$$

These sequential steps outline a systematic approach to estimating the parameter  $F_{approx(Alg)}$  using algebraic techniques, allowing for effective analysis and control of dynamic systems.

### 3.3.3 Online estimation of the ultra-local model through derivative estimation

The alternative method to  $F$  parameter estimation in Equation (4.54) involves identifying it through the derivative estimation of the noisy output signal  $y^{(v)}$  in Equation (4.55).

Recall that Equation (4.55) represents the expression for  $F(t)$ , where  $F$  is computed as the difference between the  $v$ -th derivative of the output signal  $y$  and the control input  $u$ .

Choosing  $v = 1$  yields:

$$F(t) = \dot{y}(t) - \beta u(t) \quad (3.67)$$

As discussed in Section 4.2.1.2, the first-order derivative estimation can be obtained through:

$$\dot{y} = a_1 = -\frac{6}{t^3} \int_0^t (t - 2\tau)y(\tau)d\tau \quad (3.68)$$

Thus, the function representing dynamics  $F_{approx(Der)}$  can be determined in real-time using the first-order derivative estimator, as shown in Equation (4.68):

$$F_{approx(Der)} = \dot{y} - \beta u(t) = -\frac{6}{t^3} \int_0^t ((t - 2\tau)y(\tau) - \beta u(t))d\tau \quad (3.69)$$

*Remark.* It's worth noting that if the designer opts to work with a second-order derivative, the ultra model can be estimated following a similar procedure as detailed in Section 4.2.2.3.

### 3.3.4 Discrete implementation of the ultra-local model through Derivative estimation

The procedures outlined in Eqs (4.66) and (4.69) for estimating the function  $F$  will be translated into digital implementations employing an FIR filter given in Section 4.2.3. For alternative implementations utilizing ALIEN filters, refer to [78].

Consider  $\tilde{y}(t)$  from Equation (4.67), approximated for  $\epsilon$  with  $j = 1$  in Equation (4.41):

$$\dot{y} \simeq \dot{y}(0) = a_1 = \frac{k}{q(t)} \int_0^t \tilde{Q}(t, \tau)y(\tau)d\tau; 0 < t < \epsilon \quad (3.70)$$

For a short time fixed-length window  $T$ :

$$\dot{y} = a_1 = \frac{k}{q(T)} \int_0^T \tilde{Q}(T, \tau) y(\tau) d\tau = \int_0^T \frac{k\tilde{Q}(T, \tau)}{q(T)} y(\tau) d\tau = \int_0^T Q(T, \tau) x(\tau) d\tau \quad (3.71)$$

The description estimates  $\dot{y} = a_1$  at  $t = 0$ .

1. Suppose Taylor expansion  $y(\tau)$  close to  $\tau = 0$  for  $\tilde{y}(\tau)$  :

$$\tilde{y}(\tau) \simeq y(\tau) = \sum_{j=0}^n a_1 \tau \Rightarrow a_1 = \int_0^T Q(T, \tau) y(\tau) d\tau \quad (3.72)$$

2. By transforming variables  $\tau \triangleq t - \theta$  and  $y(\theta) \triangleq x(t - \theta)$ :

$$y(\theta) \triangleq x(t - \theta) = \sum_{j=0}^n a_1 (t - \theta) \quad (3.73)$$

Consider this as a Taylor expansion  $\tilde{x}(\theta)$  for  $\theta = t$  with an inverse signal for the parameter  $\theta$ :

$$\dot{y}(t) = \left. \frac{d}{d\theta} (x(t - \theta)) \right|_{\theta=t} = (-1)a_1 \quad (3.74)$$

3. Again by transforming variables  $a_1$ :

$$a_1 = \int_0^T Q(T, \tau) x(\tau) d\tau = \int_0^T Q(T, \tau) y(t - \tau) d\tau \quad (3.75)$$

Thus,

$$\dot{y}(t) = a_1 = (-1) \int_0^T Q(T, \tau) y(t - \tau) d\tau \quad (3.76)$$

Applying Equation (4.48) results in :

$$\dot{y}(t) = a_1 \simeq (-1) \sum_{k=1}^{N+1} \alpha_k Q(T, \tau_k) y(t - \tau_k) \quad (3.77)$$

The representation of  $Q(T, \tau_k)$  in discrete form is theoretically estimated to decrease computational expenses, incorporating coefficients ( $\alpha_k$ ) obtained through trapezoidal integration in Eqs (4.49) and (4.50).

Finally, the real-time update estimation of  $\dot{y}$  is provided by:

$$\dot{y}(t) = a_1(t) \simeq (-1) \sum_{k=1}^{N+1} \alpha_k Q(T, \tau_k) y(t - \tau_k) = (-1) \hat{Q}_{\alpha_k}(T, \tau_k) \hat{y}(t - \tau_k) \quad (3.78)$$

Ultimately, following the estimation of  $\dot{y}$  and referencing equation (4.69), the previous control input is employed to observe the present parameter  $F$ , which can be resolved using this expression:

$$\hat{F}_{k(DER)} = \dot{y}_k - \beta u_{k-1} \quad (3.79)$$

### 3.3.5 Discrete implementation of the ultra-local model through Algebraic estimation

Returning to the estimation method of  $F$  as shown in (4.66), An FIR filter is obviously appropriate for digital implementations. We enhanced the previous technique and incorporated the constrained fixed-length window  $T$  in a reverse integration approach to enable a viable online implementation of the estimator discussed in section 4.3.4.

The current instance,  $F$  is viewed as an independent temporal constant, which is unaffected by the signal during reverse integration. Following additional variable manipulations, equation (4.80) emerges as the definitive formulation for the algebraic estimator of  $F_{(Alg)}$ .

Changing a few factors, the representation of  $F$  in (4.66) becomes the following:

$$F = \frac{-6}{T^3} \int_0^T (t - 2\tau)y(t - \tau)d\tau - \frac{6\beta}{T^3} \int_0^T \tau(T - \tau)u(t - \tau)d\tau \quad (3.80)$$

A good variable selection entails storing the obtained discrete filter coefficients in a vector while accounting for trapezoidal integration coefficients. order to stay out of an algebraic loop, we use the prior value of  $F_{Alg}$  to estimate the present control input. As a result, the determination of  $F_{Alg}$  remains constant, posing no additional challenge in this approach, despite the approximation being made within a short sliding window  $T$ .

## 3.4 Algebraic Derivative Estimation for Position Control of a DC Motor

DC motors, despite their higher operating costs, are widely employed in industry sectors for their remarkable capacity for speed control [79]. Studies on DC motor position control have led to various techniques, adaptable across a wide range to track different load inputs.

Previous research has covered diverse applications such as motion control of disks [80], overhead crane control [81], liquid pumps [82], wheeled mobile manipulators [83], and positioning tables [84], considering Coulomb frictions and unmodeled dynamics. Recently, interest has grown in control techniques leveraging DC motors' high precision and speed. Time response is crucial for efficiency, especially as electromechanical applications shrink. Various control techniques, including integration of discrete-time algorithms [85], sliding mode control [86], classical control with adaptive approaches [87, 88], and adaptive control using bacterial foraging algorithms [89] has been investigated. Researchers have studied disturbance observer-based control [90] and optimization strategies using LQG controllers [91].

This section introduces a model-free control technique for DC motor position tracking, utilizing an ultra-local approach along with the simultaneous use of derivative and algebraic estimators. The methodology relies on FIR filters for effective implementation. Additionally, a straightforward and efficient tuning algorithm for the ultralocal model parameter is presented, along with a comparative study.

### 3.4.1 DC motor modeling

The linear model utilized to describe the behavior of the DC motor encompasses considerations for Coulomb friction effects and unstable disturbance input, as discussed in [92]. The underlying dynamic equation for this model, which is based on Newton's second law, is as follows:

$$kV = J\ddot{\theta}_m + v\dot{\theta}_m + \hat{\Gamma}_c(\dot{\theta}_m) \quad (3.81)$$

The coefficients are as follows:

- $v$ : Signifies the viscous friction coefficient in  $\text{N} \cdot \text{m} \cdot \text{s}$ .
- $V$ : Represents the input voltage supplied to the DC motor system as a control input signal.
- $J$ : Denotes the inertia of the motor's gear in units of  $\text{kg} \cdot \text{m}^2$ .
- $k$ : Stands for the electromechanical factor, measured in  $\text{N} \cdot \text{m}/\text{V}$ .
- $\hat{\Gamma}_c$ : Represents the torque resulting from unknown friction affecting the motor's dynamics.

The non-linear component of friction, perceived as a disturbance, is described by the following equation:

**Table 3.1:** DC motor characteristics

Variable	Value
$k$ (N · m/V)	0.21
$J$ (kg·m <sup>2</sup> )	$6.87 \times 10^{-5}$
$v$ (N · m · s)	$1.041 \times 10^{-3}$
$n$	50

$$\hat{\Gamma}_c = \hat{\Gamma}_{Coul} \text{sign}(\dot{\hat{\theta}}_m) \quad (3.82)$$

$\hat{\Gamma}_{Coul}$  represents the static friction value exceeding the rotational velocity required to initiate rotation around the vertical axis. The terms  $\ddot{\hat{\theta}}_m$ ,  $\dot{\hat{\theta}}_m$ , and  $\hat{\theta}_m$  denote the angular acceleration (rad/s<sup>2</sup>), angular velocity (rad/s), and angular position (rad) of the motor, respectively. Parameter  $n$  denotes the motor gear reduction factor. Therefore,  $\theta_m = \hat{\theta}_m/n$ , where  $\theta_m$  refers to the motor's gear position and  $\hat{\theta}_m$  denotes the motor shaft's position. Additionally,  $\Gamma_c = \hat{\Gamma}_c/n$ , where  $\Gamma_c$  represents the motor gear's Coulomb friction torque.

The simulated DC motor model is the RH-8D-6006-E036AL-SP(N) [88], capable of rotating its shaft within the vertical axis in both left and right directions. Table 4.1 contains the parameter description.

### 3.4.2 Problem formulation

Given the complexities of the DC motor dynamics described by equation (4.81), it's imperative to have a reference path  $\theta_m^*(t)$  that exhibits sufficient smoothness to facilitate trajectory tracking. It's important to consider input signal variations ( $V$ ) and output noise ( $\theta_m(t)$ ). In addition, the system is subject to unknown nonlinear effects caused by model parameter uncertainties or Coulomb friction. Despite these challenges, the feedback controller's main goal is to achieve precise asymptotic tracking of  $\theta_m^*(t)$  using the system output  $\theta_m(t)$ .

### 3.4.3 Model-free Control Design

This section delineates the feedback control technique employed to achieve position tracking for the DC motor. The efficacy of model-free control laws hinges notably on the parameters  $\beta$  and iPID/iPD gains. Consequently, to enhance performance, a tuning methodology is expounded upon in sections 4.4.3.1 and 4.4.3.2 for  $\beta$  and iPD/iPID gains, respectively.

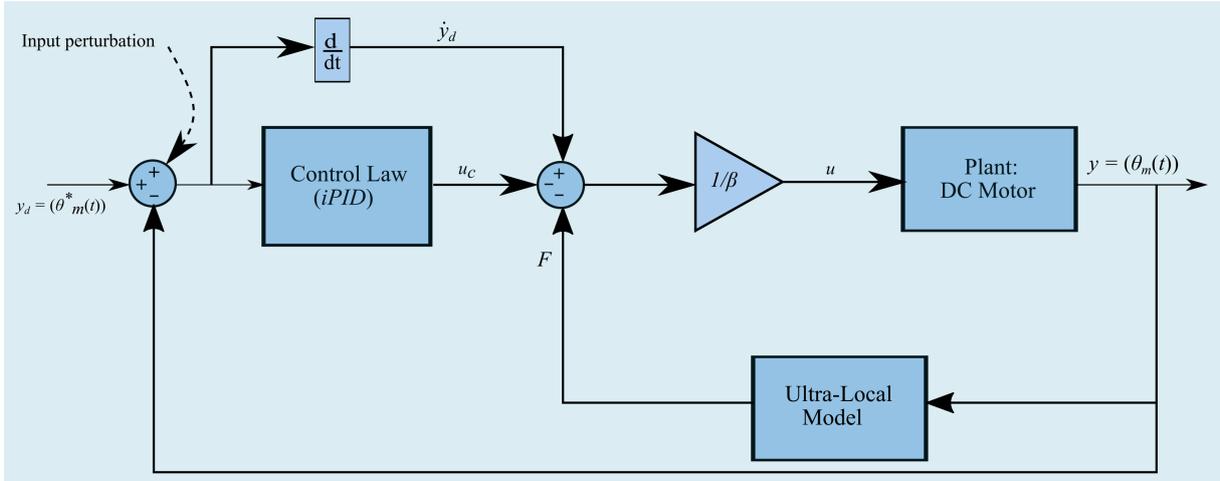


Figure 3.1: MFC scheme

### 3.4.3.1 iPID/iPD gains tuning procedures

In the upcoming simulations, both iPD and iPID controllers are utilized, assuming a constant value of  $v = 1$  in equation (4.54):

$$F(t) = \dot{y}(t) - \beta u(t) \quad (3.83)$$

The closed-loop system incorporates an intelligent PID (i-PID) controller, as illustrated in Fig. ???. It operates according to the equation:

$$u = \frac{-F + y_d - K_I \int e - K_P e - K_D \dot{e}}{\beta} \quad (3.84)$$

Here,  $y_d$  denotes the reference trajectory,  $e = y_d - y$  represents the error tracking, and  $K_P, K_I, K_D$  are the conventional PID tuning gains.

For the iPD controller, the choice of  $K_P = \lambda^2$ ,  $K_I = 0$ , and  $K_D = 2\lambda$ ,  $\lambda \in \mathbb{R}^+$  ensures the closed loop's stability, characterized by two real negative poles at  $-\lambda$ :

$$\ddot{e} + \lambda^2 e + 2\lambda \dot{e} = 0 \quad (3.85)$$

On the other hand, the iPID controller is characterized by  $K_P = 3\lambda^2$ ,  $K_I = \lambda^3$ , and  $K_D = 3\lambda$ ,  $\lambda \in \mathbb{R}^+$ , resulting in three real negative poles within a stable closed loop at  $-\lambda$ :

$$\ddot{e} + \lambda^3 \int e + 3\lambda^2 e + 3\lambda \dot{e} = 0 \quad (3.86)$$

**3.4.3.2 Tuning procedure of  $\beta$** 

According to (4.84), the dynamics of the present output signal can be showed as follows:

$$\dot{y} = \tilde{F}(t) + \tilde{\beta}(t)u \quad (3.87)$$

in which  $\tilde{\beta}(t)$  and  $\tilde{F}(t)$  denote two unspecified time-varying functions. As a result, we can rewrite this equation to demonstrate the dependency on an additional term:

$$F = \tilde{F}(t) + (\tilde{\beta}(t) - \beta)u \quad (3.88)$$

$$u = - \left( \frac{\hat{F} - \dot{y}_d}{\beta} \right) - \mathcal{Q}(e) \quad (3.89)$$

$\mathcal{Q}(e)$  denotes the tracking error correction.

Integrating (4.86) into (4.89) and setting  $\mathcal{Q}(e)$  to 0, thus focusing solely on the feed-forward equation of the control signal, yields:

$$\dot{y} = \frac{\tilde{\beta}(t)}{\beta} \dot{y}_d + \left( \tilde{F}(t) - \frac{\tilde{\beta}(t)}{\beta} \hat{F} \right) \quad (3.90)$$

The sensitivity of  $\hat{F}$  measurement to specific dynamics underscores the criticality of selecting  $\beta$  to closely match the actual system gain and minimize changes in  $F$ .

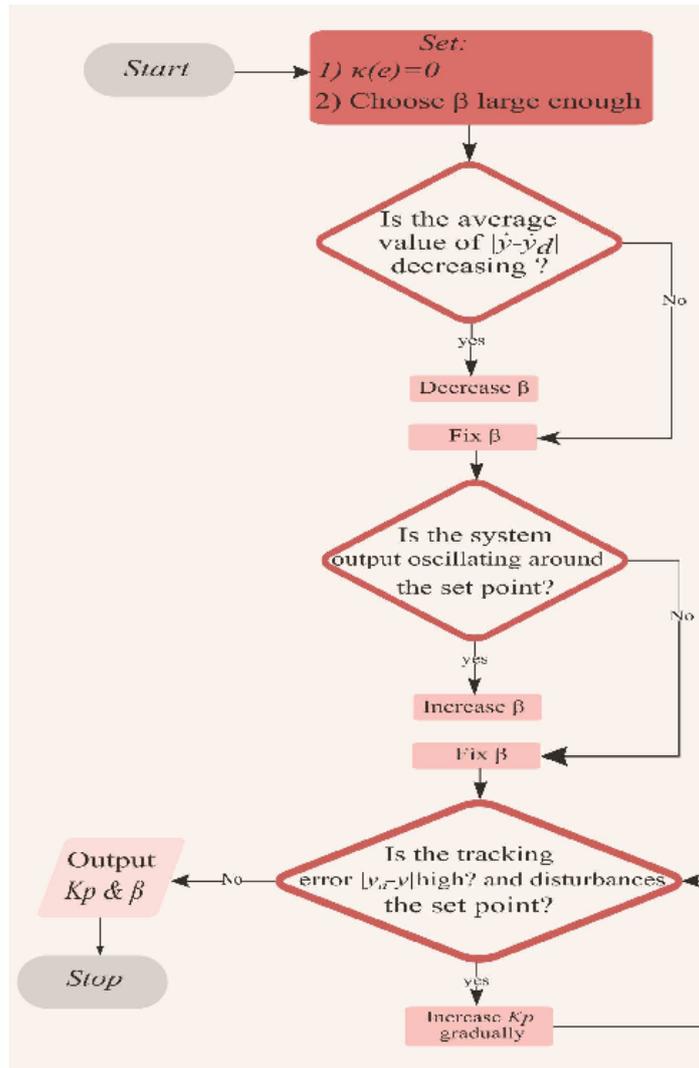


Figure 3.2: parameter's tuning procedure  $\beta$

As the parameter  $\beta$  tends towards infinity, the control input  $u$  outlined in equation (4.89) approaches  $\mathcal{Q}(e)$ . In this limiting scenario, the controller configuration simplifies to resemble that of a conventional PID controller, where the control action is primarily dictated by the error term.

Conversely, when  $\beta$  approaches zero from the positive side ( $\beta \rightarrow 0^+$ ), the control signal  $u$  predominantly relies on the estimated value  $\hat{F}$ . Consequently, tuning  $\beta$  becomes a straightforward process by setting  $\mathcal{Q}(e)$  to zero and observing the system's response to different  $\beta$  values. Initially, selecting a sufficiently high  $\beta$  results in an approximate suppression of the control signal, effectively rendering it close to zero. Subsequently, decreasing the value of  $\beta$  gradually elevates the average control signal until it reaches a level where the condition of the reference closed-loop ( $\dot{y} = \dot{y}_d$ ) is satisfied. This iterative

process allows for the fine-tuning of  $\beta$  to achieve the desired performance of the control system. For a visual representation and better understanding, refer to Figure 4.2, which illustrates the stepwise procedure for tuning the parameter  $\beta$ .

### 3.4.4 Simulations results

Numerical simulations are undertaken to validate the effectiveness of the deployed control strategy in achieving rapid convergence of the tracking error to a confined region near zero. Additionally, the simulations aimed to assess the control law's ability the control effort low and ensure smooth transient responses, even in the presence of inaccuracies in the expert input, such as perturbations stemming from Coulomb friction effects or noisy measurements affecting the system.

The controller gains are selected by positioning poles of the closed-loop appropriately along the real negative axis. Throughout this search, the poles were positioned at  $p = -100$  rad/s, along with the PD and PID gains  $K_P, K_I, K_D$  were calculated using the expressions outlined in (4.85) and (4.86).

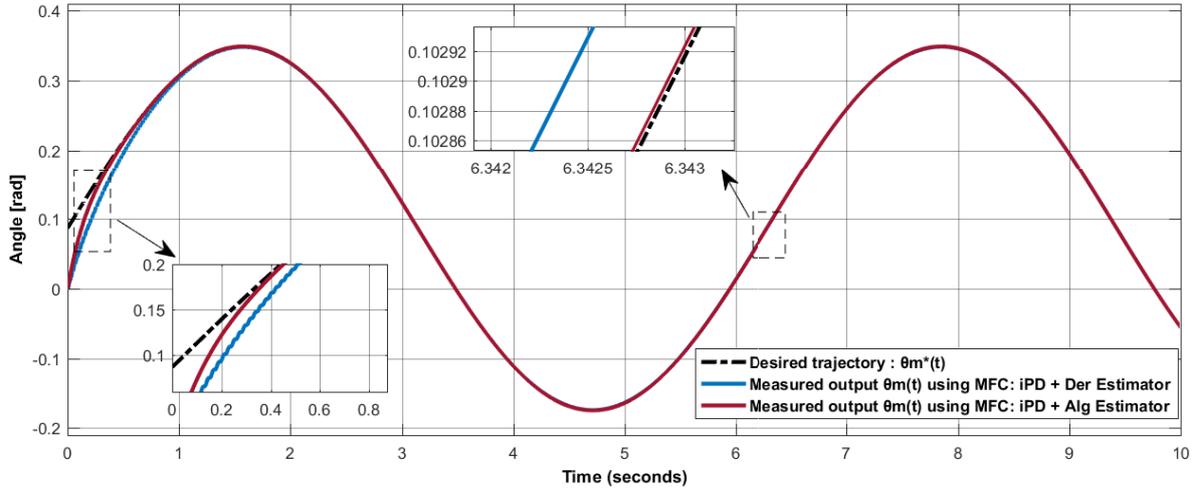
Two different trajectories demonstrate the effectiveness of the model-free control method via the ultra-local model based on derivative/algebraic estimators:

1. Sinusoidal Position Tracking: This trajectory follows a sinusoidal pattern defined by  $\theta_m^* = A \sin(\omega t) + B$ , where  $A = \pi/12$  (rad) represents the amplitude,  $B = \pi/36$  (rad) represents the bias, and  $\omega = 1$  (rad/s) represents the frequency.
2. Nominal Trajectory: This trajectory is defined by an eighth-order Bezier polynomial, smoothly using interpolation from 0 to 1 throughout the period of time  $[t_i, t_f]$ . It can be expressed as:

$$\theta_m^*(t) = \theta_m^*(t_i) + (\theta_m^*(t_f) - \theta_m^*(t_i))\varphi(\tau, t_f, t_i) \quad (3.91)$$

where  $\tau = (t - t_i) / (t_f - t_i)$ .

The study made use of a 10kHz sampling frequency. Equations (4.66) and (4.69) used a fixed sampling time of  $T_s = 10^{-4}$ s, an estimation window of  $T = 0.2$ s, as well as a total of  $N = 2000$  samples both the derivative and algebraic estimation techniques for  $F$ . These parameters were chosen to ensure accurate estimation of the system dynamics while



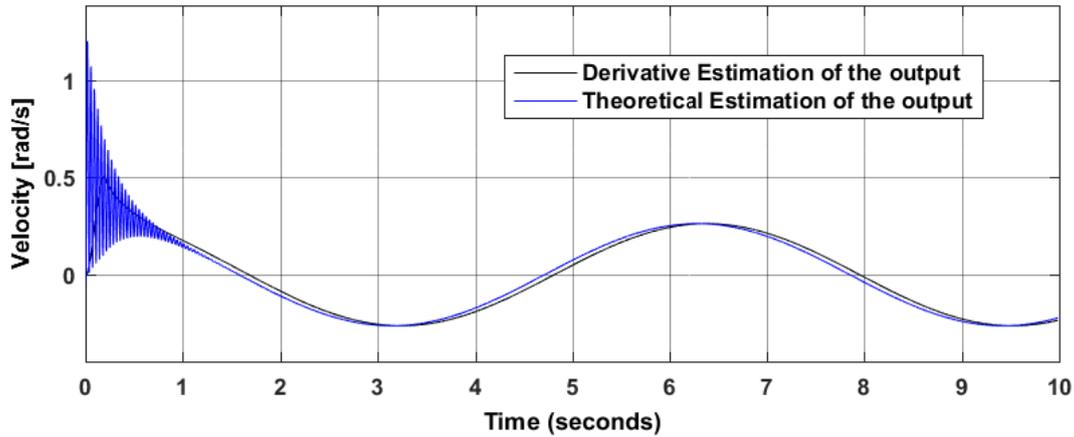
**Figure 3.3:** Sinusoidal trajectory - Position tracking employing derivatives and algebraic estimators based on the ultra-local model

balancing computational efficiency.

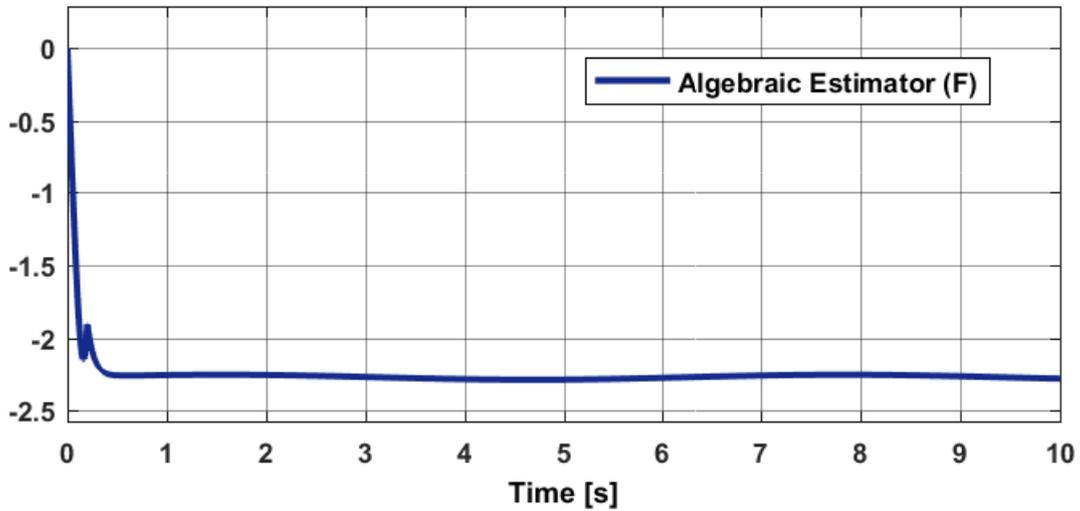
In the forthcoming section, (Der) designates the approximation of parameter  $F$  via angular velocity in Equation (4.54), while (Alg) represents the algebraic approximation of  $F$  in Equation (4.69). The measured output, denoted as ( $y$ ), the reference input  $y_d$ , and the control signal ( $u$ ) are symbolized by  $(\theta_m(t))$ ,  $(\theta_m^*(t))$ , and ( $V$ ), respectively.

The output's first-order derivative  $\dot{y}$ , utilized in Equation (4.66), undergoes estimation using a derivative estimator, evaluated with a theoretical derivator, as illustrated in Fig. 4.4. Various  $\beta$  values were employed for each estimation approach to ensure a desirable response, guided by Fig. 4.3.

The tuning procedure outlined in the preceding section was executed to use the control law for ultra-local model-based estimators. This method commences by setting the proportional gain  $K_P$  to null. Subsequently, a reference trajectory is introduced when the plant is stationary, and open-loop system responses are observed across different  $\beta$  values. Starting with  $\beta_{Alg} = 1000$  and  $\beta_{Der} = 1000$ , their values are systematically decreased until the control signal tends to oscillate, eventually reaching  $\beta_{Alg} = 1$  and  $\beta_{Der} = 45$ . Following this, the proportional gain  $K_P$  is adjusted to ensure a reasonable tracking error. Finally,  $\beta_{Alg}$  and  $\beta_{Der}$  are established at 3 and 100, respectively. This precise process ensures robust and efficient control law application.



**Figure 3.4:** An estimate of the angular velocity derived from a sinusoidal trajectory  $\hat{y} = (\hat{\theta}_m(t))$



**Figure 3.5:** An estimate of the unmodeled dynamics for a sinusoidal trajectory

Figure 4.2 presents the numerical results obtained from the demonstrated controllers, showcasing their effectiveness in maintaining the desired sinusoidal output position despite the presence of various sources of uncertainties. At  $t = 0.01$  s, the online update procedure for both the function  $F$  and the angular velocity begins, as depicted in Figs. 4.4 and 4.5, respectively. Simultaneously, the voltage control effort signal, portrayed in Figs. 4.6 and 4.7, is actively utilized to counteract perturbations stemming from Coulomb friction effects, ensuring precise trajectory tracking and system stability.

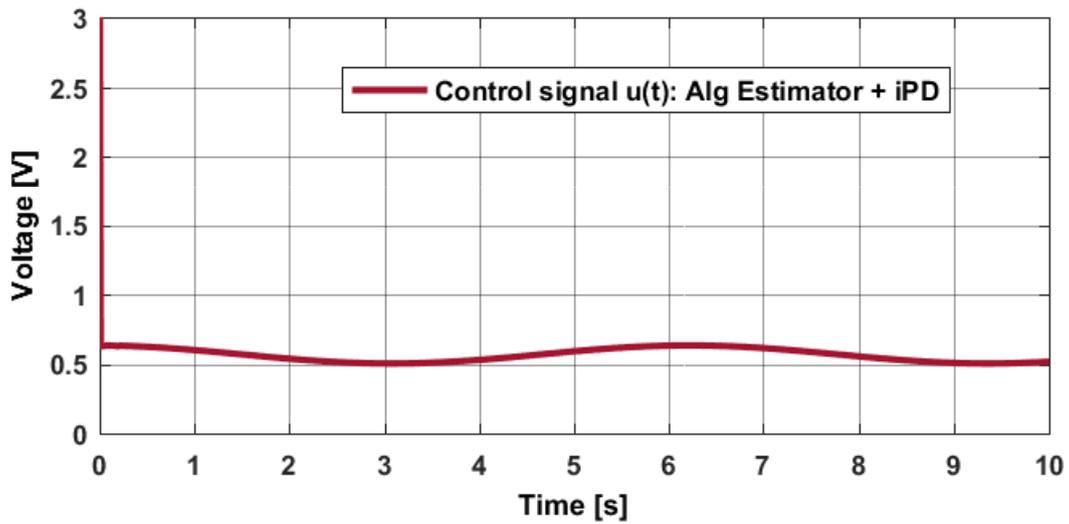


Figure 3.6: Sinusoidal trajectory - Control signal input using algebraic estimator and iPD

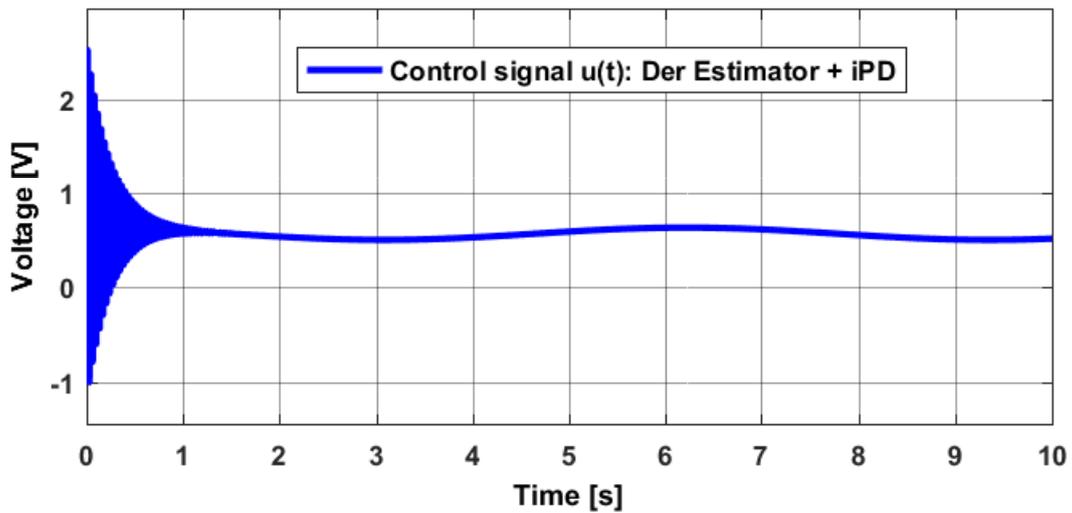
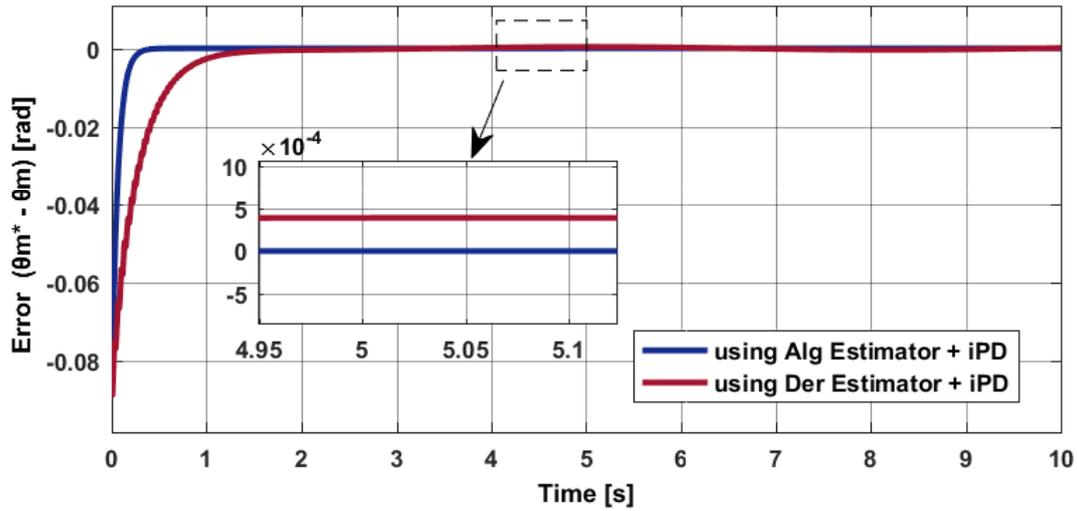


Figure 3.7: Sinusoidal trajectory - Control signal input using derivative estimator and iPD controllers

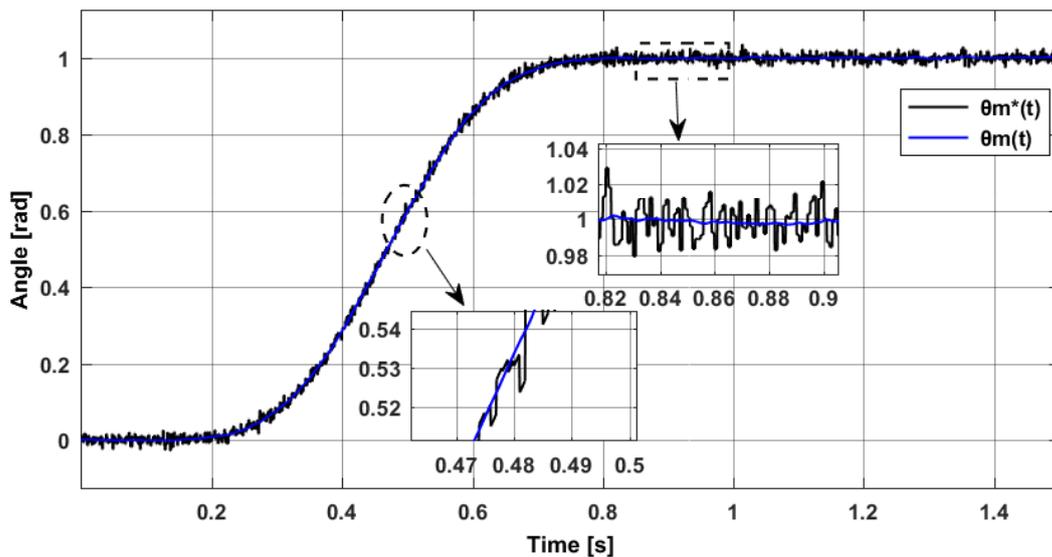
Figure 4.8 illustrates the error in position tracking employing the pair-stated controllers derived from the derivative and algebraic estimators. Through feedback control, the angular position of the DC motor is finely adjusted, minimizing state errors within a short timeframe while compensating for errors in model parameters.

**Table 3.2:** Sinusoidal trajectory - The efficiency of the control methods according to the ITAE, ISE, and IAE criteria

Control technique	ISE	IAE	ITAE
iPD + Derivative Estimator	$0.12 \times 10^{-2}$	0.03	0.03
iPD + Algebraic Estimator	$0.48 \times 10^{-3}$	0.01	0.002
Classical PD	$0.4 \times 10^{-2}$	0.19	0.96



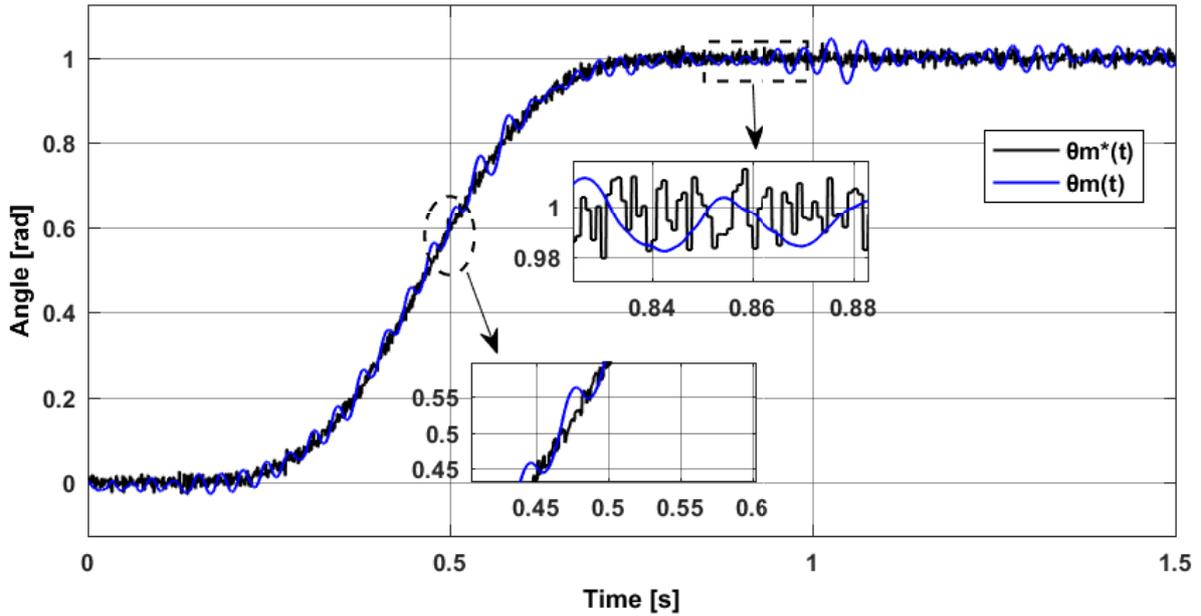
**Figure 3.8:** Sinusoidal trajectory - Error tracking with algebraic/derivative estimators and iPD



**Figure 3.9:** Perturbed Bezier's trajectory - The DC motor's position tracking using iPID with algebraic estimator and considering Coulomb's friction effects

**Table 3.3:** Perturbed Bezier's trajectory - The efficiency of the control methods according to the ITAE, ISE, and IAE criteria

Control technique	ISE	IAE	ITAE
iPD + Derivative Estimator	$0.41 \times 10^{-2}$	0.09	0.21
iPD + Algebraic Estimator	$0.1 \times 10^{-2}$	0.08	0.19
Classical PID	$0.86 \times 10^{-2}$	0.26	0.47

**Figure 3.10:** Perturbed Bezier's trajectory - Position tracking of the DC motor using iPID with derivative estimator and considering Coulomb's friction effects

The investigation findings demonstrate remarkable efficiency in model-free control tracking, particularly enhanced when incorporating an integral component. In comparison, the derivative scheme yielded a noisier control signal, rendering the system less robust. While expanding the estimation window can enhance filtering features, it introduces a delay that may potentially destabilize the closed-loop system.

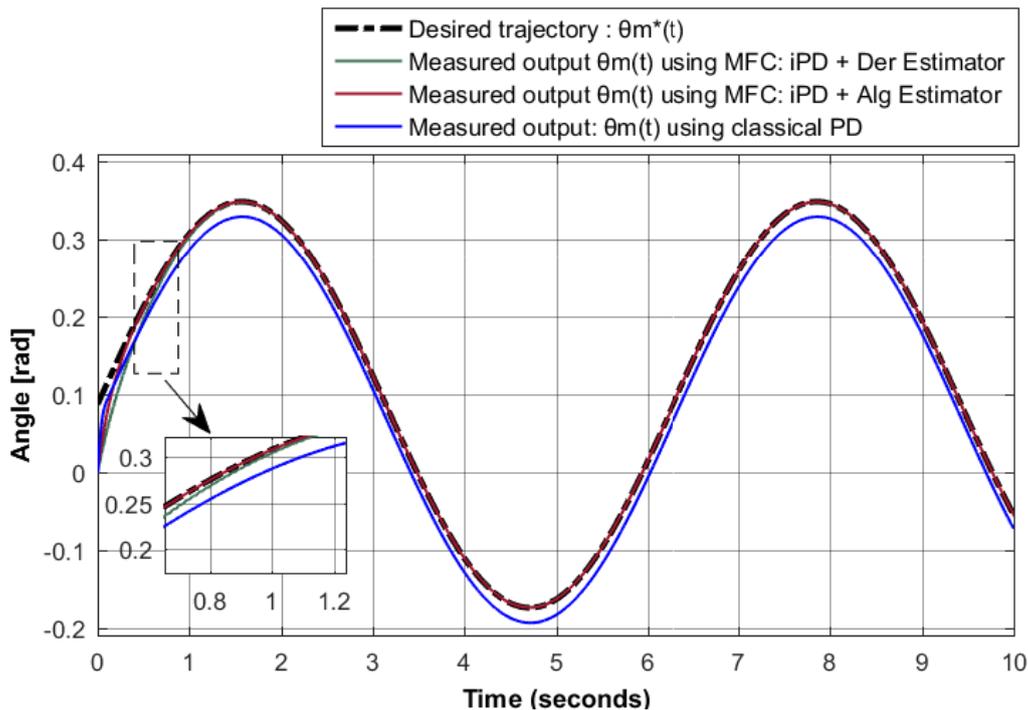
Figures 4.9 and 4.10 present cases where an eighth-order Bezier's polynomial input disturbance signal is applied, generating significant deformations in the input signal. This setup allows for the evaluation of the controller's smoothing effectiveness in cases utilizing derivative and algebraic estimators.

Additionally, the performance evaluation of the control methodologies has been conducted based on integral time absolute tracking error (*ITAE*), integral absolute tracking error (*IAE*), and integral squared tracking error (*ISE*). These metrics are defined as follows:

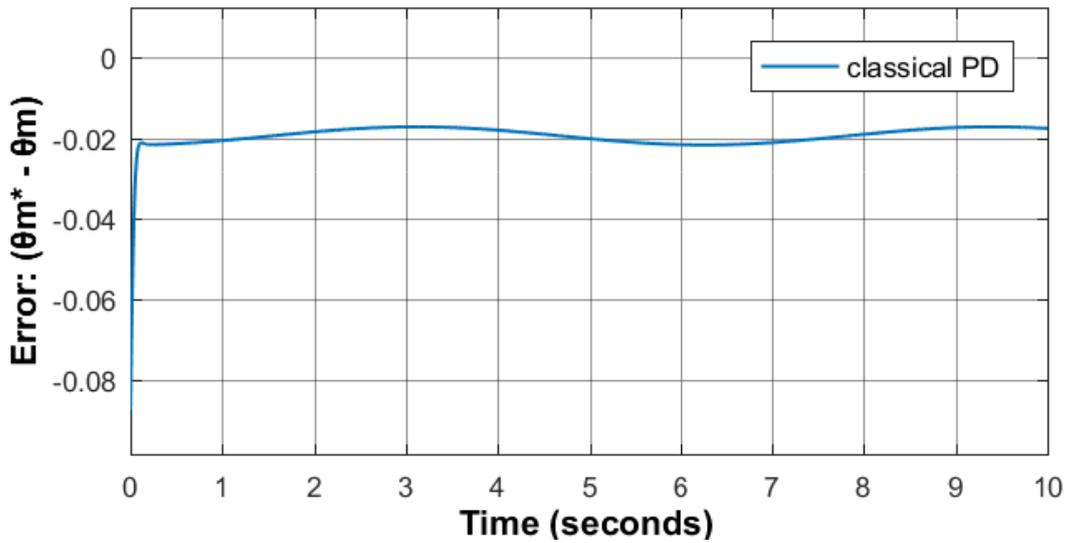
$$\begin{aligned}
ITAE &= \int_{t_i}^{t_f} t |e_{\theta_m}(t)| dt, \\
IAE &= \int_{t_i}^{t_f} |e_{\theta_m}(t)| dt, \\
ISE &= \int_{t_i}^{t_f} e_{\theta_m}^2(t) dt,
\end{aligned} \tag{3.92}$$

where  $t_i = 0$  s and  $t_f = 10$  s represent the initial and final times of the simulation, respectively. While  $ISE$  and  $IAE$  metrics treat all tracking errors equally,  $ITAE$  places significant emphasis on time. Errors that evolve slowly are heavily penalized in the  $ITAE$  standard, while those occurring slightly earlier are disregarded.

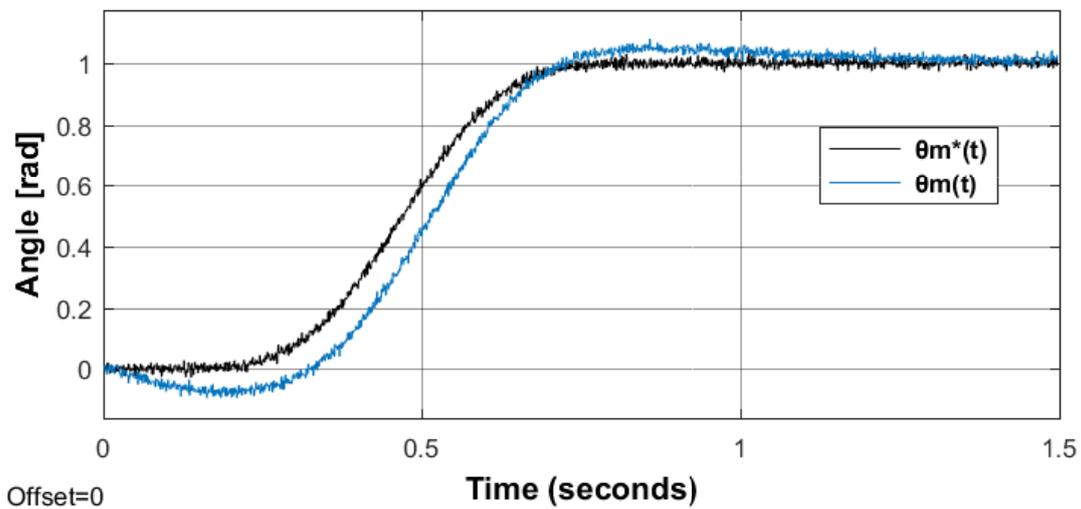
The comparative analysis of the obtained results is presented in Tables 4.2 and 4.3. The findings indicate that the proposed control approach utilizing the ultra-local model based on the algebraic estimator exhibits a slight advantage over the derivative estimator-based model. This suggests that the algebraic estimator may provide better performance in terms of tracking accuracy and robustness.



**Figure 3.11:** Sinusoidal trajectory: Comparing conventional PD and derivative/algebraic estimators with iPD



**Figure 3.12:** Sinusoidal trajectory - Error tracking with traditional PD



**Figure 3.13:** Perturbed Bezier's trajectory - The DC motor position tracking using traditional PID considering Coulomb's friction effects

Finally, a comparison is drawn between the proposed ultra-local model-based algebraic/derivative estimator approaches and a classical PID controller, which remains one of the most commonly employed control strategies. In this comparison, a classical PD controller is employed for tracking the sinusoidal trajectory, while a classical PID controller is utilized for the perturbed Bezier's trajectory. The PD/PID controllers are manually tuned to achieve rapid closed-loop responses with minimal overshoots along the desired trajectory.

The results obtained from this comparison are illustrated in Figures 4.11 and 4.12. In the transient region, it is observed that the iPD controllers track the reference trajectory more rapidly compared to traditional PD controllers. This enhanced performance can be attributed to the incorporation of the  $\dot{y}_r$  factor in the control law, as outlined in Eq. (4.52). Furthermore, the proposed techniques demonstrate accurate and robust tracking results, effectively mitigating disturbances caused by Coulomb friction, as evident in Figs. 4.9, 4.10, and 4.13.

At  $t = 0.25$  s, the PID controller exhibits overshoot, which can be minimized by sacrificing tracking speed. The comparison of performance metrics including ISE, IAE, and ITAE criteria is summarized in Tables 2 and 3. Furthermore, the proposed approaches offer superior tracking performance, particularly in scenarios with rapid trajectory changes or external disturbances. The incorporation of advanced control laws, such as iPD and iPIDs, allows for faster response times, reduced overshoot, and smoother trajectory tracking compared to classical PID controllers. This translates to improved system efficiency, stability, and overall operational performance.

### 3.5 Conclusion

This study investigated and implemented recent state observation techniques for the ultra-local model used in the model-free control approach, employing two distinct methodologies: an ultra-local model-based algebraic estimator and an ultra-local model-based derivative estimator. The estimation process utilized Taylor's expansion coefficients of arbitrary order, with discrete implementation details presented through operational calculus and finite impulse response (FIR) filters.

A comprehensive comparative analysis between algebraic and derivative estimators was conducted, revealing significant insights into their performance in state observation. Various tests were performed on a position trajectory control scheme for a DC motor subjected to disturbances and uncertainties. The ultra-local model design demonstrated independence from initial conditions, requiring only the measurement of the system's input and output values.

Key findings highlighted that the proposed design effectively reduced nonlinearities and enhanced robustness against disturbances arising from Coulomb friction effects. Both algebraic and derivative estimation processes were computed in real-time, ensuring quick and accurate state observation.

The study's outcomes were validated through numerical simulations, which compared the recommended approaches with other control strategies. The simulations confirmed the

feasibility and reliability of the ultra-local model, particularly emphasizing the superior performance of the algebraic method. The robust tracking capability against disturbances and uncertainties surpassed that of classical PID controllers.

Moreover, challenges such as high-frequency sampling and increased computational effort associated with longer filter lengths were acknowledged. These challenges were mitigated by leveraging powerful and cost-effective hardware solutions available in the market.

In addition to demonstrating the efficiency of the closed-loop control estimators, this research also provides valuable insights and simplifies several aspects of the digital implementation of these modern controllers.

In conclusion, the integration of advanced algebraic and derivative estimators into model-free control systems significantly enhanced the control performance of DC motor position tracking. The study provides valuable insights into the practical application of these estimators, paving the way for more resilient and efficient control strategies in DC motor systems.

# Chapter 4

## Algebraic and Derivative Estimation

### 4.1 Introduction

In the field of system identification, accurate parameter estimation is crucial for interpreting the dynamics of systems and establishing efficient control strategies. Conventional approaches frequently depend on iterative numerical optimization procedures, which can be computationally demanding and susceptible to noise. Algebraic estimating approaches present an attractive choice as they offer both rapidity and precision in parameter estimation.

In 2004, algebraic estimation techniques were introduced which led to a significant change in the field of system identification. Their approach, centered around polynomial modeling, offered an original take on the subject. Instead of using iterative optimization methods, these techniques approach the parameter identification problem as an algebraic one, allowing for direct estimation of system coefficients using input-output data. This method is especially beneficial for linear systems, as the connection between inputs and outputs can be expressed through polynomial functions [70].

Mboup and Join extended the algebraic framework to compute individual estimators for each coefficient in the polynomial model. Derivative estimation is a technique that overcomes the difficulties associated with simultaneously estimating values, such as the issue of calculating inverses of poorly conditioned matrices. These issues can lead to inaccurate results, especially when there is interference. The proposed derivative estimation computes the derivatives of a system by analyzing each coefficient separately. This approach guarantees that the estimations obtained are robust and reliable [71].

Based on Mboup, Join, and Fliess' approach, the following research has advanced derivative estimation in the algebraic context. According to Mboup, Join, and Fliess [72], sliding-window estimation methods adapt the estimation window size to system dynamics.

This dynamic technique makes derivative estimate algorithms more resilient, especially when system parameters change. Recent advances in machine learning and optimization have integrated algebraic and data-driven methods to improve complex system parameter identification accuracy and efficiency.

This chapter delves into innovative state observation methods within the scope of model-free control. Specifically, it introduces derivative and algebraic estimators for state observation to the ultra-local model implementation. The estimation process utilizes Taylor's expansion coefficients of any order, providing a framework for parameter estimation. The proposed approach is discretely implemented using operational mathematical concepts and FIR filters.

The study conducts a comparative analysis between derivative and algebraic estimators, examining their performance in state observation. Tests are conducted on a DC motor's position trajectory control strategy under disturbances and uncertainty. The ultra-local model employed in this study solely relies on system input and output measurements, making it independent of initial conditions. Furthermore, the proposed design aims to minimize nonlinearities and enhance Coulomb friction robustness, as demonstrated through empirical analysis.

## 4.2 Derivative Estimation

The identification of the parameters of a constricted Taylor expansion of random order is the basis of the derivative estimation approach. To accomplish this, it is necessary to suppose that the output signal  $\tilde{x}(t)$  represents an analytic variable and may therefore be represented using Taylor series that is truncated at  $t = 0$ .

$$\tilde{x}(t) \approx x(t) = \sum_{j=0}^n \frac{x^{(j)}(0)}{j!} t^j \quad (4.1)$$

Identifying these coefficients requires math procedures in a domain similar to the Laplace transform. After converting the expression again to the time domain, the estimation filters are discretized into sliding-window finite impulse response filters for applications in real-time.

A more comprehensive formulation for estimators of arbitrary order can be found in the works [72, 73]. Throughout this thesis, we will offer a detailed explanation of the development of these estimators, aiming to clarify the fundamental concepts of the algebraic technique.

## 4.2.1 1<sup>st</sup> Order estimation

This section presents the filtering and derivation of first-order estimation:

### 4.2.1.1 Filtering of first order estimation

Let's delve into the initial estimation of the noisy output signal  $\tilde{x}(t)$  near  $t = 0$ :

$$x(t) = a_0 + a_1 t \quad (4.2)$$

The transformation to the Laplace form results in the operational form of Equation (4.2) being expressed as follows:

$$X(s) = \frac{a_0}{s} + \frac{a_1}{s^2} \quad (4.3)$$

By factoring every component of the equation by  $s^2$ , we obtain:

$$s^2 X(s) = sa_0 + a_1 \quad (4.4)$$

This allows us to separate and eliminate  $a_1$  by multiplying both sides with the differentiating operator  $\frac{d}{ds}$ :

$$2sX(s) + s^2 \frac{d}{ds} X(s) = a_0 \quad (4.5)$$

To avoid time derivatives and filtering, the multiplication of both sides by  $\frac{1}{s^3}$  is performed. Each factor involving  $X(s)$  is integrated at least one time:

$$2 \frac{1}{s^2} X(s) + \frac{1}{s} \frac{d}{ds} X(s) = \frac{a_0}{s^3} \quad (4.6)$$

Transforming the right side of the expression back to the time domain:

$$\frac{d}{s^\alpha}, \alpha \geq 1, d \in \mathbb{D} \longleftrightarrow d \frac{t^{\alpha-1}}{(\alpha-1)!} \quad (4.7)$$

By utilizing the Cauchy equation for iterated integration [74], we can derive an expression in closed form to feed the multiplier  $\frac{1}{s^\alpha} \frac{d^n}{ds^n}$  in just one integral. This is possible because differentiation of  $X(s)$  according to  $s$  corresponds to factorization by  $-t$  in the time domain, and left factorization  $X(s)$  by  $\frac{1}{s^\alpha}, \alpha \geq 1$  corresponds to repeated integrals.

$$\frac{1}{s^\alpha} \frac{d^n}{ds^n} X(s) \longleftrightarrow \frac{(-1)^n}{(\alpha-1)!} \int_0^t (t-\tau)^{\alpha-1} \tau^n x(\tau) d\tau \quad (4.8)$$

By applying the above relation, Equation (2.6) can be converted again into the time domain:

$$2 \int_0^t (t - \tau)x(\tau)d\tau - \int_0^t \tau x(\tau)d\tau = a_0 \frac{t^2}{2!} \quad (4.9)$$

By restructuring the formula, we may identify the factor  $a_0$  as:

$$a_0 = \frac{2}{t^2} \int_0^t (2t - 3\tau)x(\tau)d\tau \quad (4.10)$$

By replacing the function  $x(t)$  with  $\tilde{x}(t)$ , Equation (4.10) shall be viewed as being a first-order filtered estimation of the signal. This form of approximation is particularly useful for estimating the signal over short time intervals when  $t > 0$ . Additional information regarding the digital implementation of sliding-window estimators will be presented in the following section.

#### 4.2.1.2 Derivation of first-order estimation

Let's analyze the first estimation of the output noisy signal  $\tilde{x}(t)$  at  $t = 0$ , as previously described in  $s$  domain:

$$X(s) = \frac{a_0}{s} + \frac{a_1}{s^2} \quad (4.11)$$

Factoring in each side by  $s$ , we acquire:

$$sX(s) = a_0 + \frac{a_1}{s} \quad (4.12)$$

This allows us to isolate  $a_0$ , which can be removed by factoring with the differentiation operation  $\frac{d}{ds}$ :

$$X(s) + s \frac{d}{ds} X(s) = -\frac{a_1}{s^2} \quad (4.13)$$

Further factoring by  $\frac{1}{s^2}$  aims to eradicate filtering and time derivativion, with each element involving  $X(s)$  being integrated once:

$$\frac{X(s)}{s^2} + \frac{1}{s} \frac{d}{ds} X(s) = -\frac{a_1}{s^4} \quad (4.14)$$

By utilizing the established principles of Cauchy formula and Laplace Transformation

for repetitive integration, we derive:

$$\int_0^t (t - \tau)x(\tau)d\tau - \int_0^t \tau x(\tau)d\tau = -a_1 \frac{t^3}{3!} \quad (4.15)$$

By restructuring the equation, we can find the first estimate of the derivative:

$$a_1 = -\frac{6}{t^3} \int_0^t (t - 2\tau)x(\tau)d\tau \quad (4.16)$$

## 4.2.2 2<sup>nd</sup> Order estimation

This section presents the filtering and derivation of second-order estimation:

### 4.2.2.1 Filtering of second order estimation

Let us examine the second-order estimation of the noisy signal  $\tilde{x}(t)$  at  $t = 0$ :

$$x(t) = a_0 + a_1 t + a_2 \frac{t^2}{2} \quad (4.17)$$

Employing the Laplace Transform:

$$X(s) = \frac{a_0}{s} + \frac{a_1}{s^2} + \frac{a_2}{s^3} \quad (4.18)$$

Factoring by  $s^3$  to isolate  $a_2$  :

$$s^3 X(s) = s^2 a_0 + s a_1 + a_2 \quad (4.19)$$

Employing  $\frac{d}{ds}$  to eliminate  $a_2$ :

$$3s^2 X(s) + s^3 \frac{d}{ds} X(s) = 2s a_0 + a_1 \quad (4.20)$$

Further application of  $\frac{d}{ds}$  to eliminate  $a_1$ :

$$6s X(s) + 6s^2 \frac{d}{ds} X(s) + s^3 \frac{d^2}{ds^2} X(s) = 2a_0 \quad (4.21)$$

Factoring by  $\frac{1}{s^4}$  to eliminate filtering and time derivatives:

$$6 \frac{1}{s^3} X(s) + 6 \frac{1}{s^2} \frac{d}{ds} X(s) + \frac{1}{s} \frac{d^2}{ds^2} X(s) = 2a_0 \frac{1}{s^4} \quad (4.22)$$

Returning to time domain transformation:

$$3 \int_0^t (t - \tau)^2 x(\tau) d\tau - 6 \int_0^t (t - \tau) x(\tau) d\tau + \int_0^t \tau^2 x(\tau) d\tau = 2a_0 \frac{t^3}{3!} \quad (4.23)$$

The second-order filter is acquired by arranging terms:

$$a_0 = \frac{3}{t^3} \int_0^t (3t^2 - 12t\tau + 10\tau^2) x(\tau) d\tau \quad (4.24)$$

#### 4.2.2.2 Derivation of first-order estimation

Let's explore the second-order estimation of the noisy signal  $\tilde{x}(t)$  which has already been described in  $s$  domain:

$$X(s) = \frac{a_0}{s} + \frac{a_1}{s^2} + \frac{a_2}{s^3} \quad (4.25)$$

Factoring by  $s^3$  to isolate  $a_2$  :

$$s^3 X(s) = s^2 a_0 + s a_1 + a_2 \quad (4.26)$$

Utilizing differentiation with relative to  $s$  to exclude  $a_2$ :

$$3s^2 X(s) + s^3 \frac{d}{ds} X(s) = 2s a_0 + a_1 \quad (4.27)$$

Factoring by  $\frac{1}{s}$  to isolate  $a_0$ :

$$3s X(s) + s^2 \frac{d}{ds} X(s) = 2a_0 + \frac{a_1}{s} \quad (4.28)$$

Further differentiation relative to  $s$ , aiming to eliminate  $a_0$ :

$$3X(s) + 5s \frac{d}{ds} X(s) + s^2 \frac{d^2}{ds^2} X(s) = -a_1 \frac{1}{s^2} \quad (4.29)$$

Factoring by  $\frac{1}{s^3}$ :

$$3 \frac{1}{s^3} X(s) + 5 \frac{1}{s^2} \frac{d}{ds} X(s) + \frac{1}{s} \frac{d^2}{ds^2} X(s) = -a_1 \frac{1}{s^5} \quad (4.30)$$

Returning to time domain transformation:

$$\frac{3}{2} \int_0^t (t - \tau)^2 x(\tau) d\tau - 5 \int_0^t (t - \tau) \tau x(\tau) d\tau + \int_0^t \tau^2 x(\tau) d\tau = -a_1 \frac{t^4}{4!} \quad (4.31)$$

By restructuring the equation, we can find the first estimate of the derivative  $a_1$ :

$$a_1 = -\frac{12}{t^4} \int_0^t (3t^2 - 16t\tau + 15\tau^2) x(\tau) d\tau \quad (4.32)$$

### 4.2.2.3 Derivation of second-order estimation

Let's explore the second-order estimation of the noisy signal  $\tilde{x}(t)$  which has already been described in  $s$  domain:

$$X(s) = \frac{a_0}{s} + \frac{a_1}{s^2} + \frac{a_2}{s^3} \quad (4.33)$$

Factoring by  $s^2$  to isolate  $a_1$  :

$$s^2 X(s) = sa_0 + a_1 + \frac{a_2}{s} \quad (4.34)$$

Applying the differential operator  $\frac{d}{ds}$  to eliminate  $a_1$  gives:

$$2sX(s) + s^2 \frac{d}{ds} X(s) = a_0 - \frac{a_2}{s^2} \quad (4.35)$$

Further application of the differential operator  $\frac{d}{ds}$  to eliminate  $a_0$  results in:

$$2X(s) + 4s \frac{d}{ds} X(s) + s^2 \frac{d^2}{ds^2} X(s) = 2a_2 \frac{1}{s^3} \quad (4.36)$$

Using  $\frac{1}{s^3}$  as a scaling factor, we get:

$$2 \frac{1}{s^3} X(s) + 4 \frac{1}{s^2} \frac{d}{ds} X(s) + \frac{1}{s} \frac{d^2}{ds^2} X(s) = 2a_2 \frac{1}{s^6} \quad (4.37)$$

Returning to time domain transformation:

$$\int_0^t (t - \tau)^2 x(\tau) d\tau - 4 \int_0^t (t - \tau) \tau x(\tau) d\tau + \int_0^t \tau^2 x(\tau) d\tau = 2a_2 \frac{t^5}{5!} \quad (4.38)$$

restructuring the equation provides the second-order derivative's estimate:

$$a_2 = \ddot{x}(t) = \frac{60}{t^5} \int_0^t (t^2 - 6t\tau + 6\tau^2) x(\tau) d\tau \quad (4.39)$$

After a series of mathematical operations, we've derived expressions for estimating the second-order derivatives of a noisy signal  $\tilde{x}(t)$  around  $t = 0$ . By manipulating the Laplace Transform representation in the operational domain and performing differential operations, we obtained equations in terms of  $a_0$ ,  $a_1$ , and  $a_2$ , the coefficients of the signal approximation. These equations were then transformed back to the time domain to obtain

practical formulas for estimating the second-order derivative  $\ddot{x}(t)$ . These expressions involve integrals of the signal  $x(t)$  over specific ranges, providing a method to estimate the curvature of the signal at  $t = 0$ . These results can be particularly useful in signal processing and system identification tasks that will be used in estimating the ultra-local model.

### 4.2.3 Implementation of the Derivative Estimator in Discrete Form

The estimators formulated in the preceding section target the estimation of the  $j$ -order derivative of a given noisy signal  $\tilde{x}(t)$ , which is represented by an  $n$ -order Taylor expansion evaluated at  $t = 0$ :

$$\tilde{x}(t) \simeq x(t) = \sum_{j=0}^n x^{(j)}(0) \frac{t^j}{j!} \quad (4.40)$$

The estimations hold true within a limited time frame  $\epsilon$ . In a broader context, the  $j$ -order derivative estimator, where every component associated with  $x(t)$  undergoes at least one integration, is articulated as follows:

$$x^{(j)}(t) \simeq x^{(j)}(0) = a_j = \frac{k}{q(t)} \int_0^t \tilde{P}(t, \tau) x(\tau) d\tau, 0 < t < \epsilon \quad (4.41)$$

This method can be applied within a short time frame of fixed duration  $T$ :

$$a_j = \frac{k}{q(T)} \int_0^T \tilde{Q}(T, \tau) x(\tau) d\tau = \int_0^T \frac{k\tilde{Q}(T, \tau)}{q(T)} x(\tau) d\tau = \int_0^T Q(T, \tau) x(\tau) d\tau \quad (4.42)$$

The equation presented provides an estimate for the parameter  $a_j = x^{(j)}(0)$ , which is accurate only at  $t = 0$ . It relies on a Taylor series expansion centered at  $t = 0$ . While this estimate is exact for noise-free polynomial input signals, it gradually deviates for arbitrary signals as time progresses. This deviation arises from assuming that a polynomial near  $t$  can effectively approximate the input signal. It's crucial to note that this estimator is non-causal, meaning that to estimate the derivative at  $t = 0$ , function values  $x(t)$  for  $t > 0$  are necessary. This necessity prompts an approach adjustment to create a causal estimator at any given time  $t$ . Initially, let's explore the Taylor expansion  $r(\tau)$  centered at  $\tau = 0$  of a random signal  $\tilde{r}(\tau)$ . The coefficients of this expansion may be established using the estimators discussed afterward:

$$\tilde{r}(\tau) \simeq r(\tau) = \sum_{j=0}^n a_j \frac{\tau^j}{j!} \Rightarrow a_j = \int_0^T Q(T, \tau) r(\tau) d\tau \quad (4.43)$$

Taking into account the variable transformations,  $\tau \triangleq t - \delta$  and  $x(\delta) \triangleq r(t - \delta)$  :

$$x(\delta) \triangleq r(t - \delta) = \sum_{j=0}^n a_j \frac{(t - \delta)^j}{j!} \quad (4.44)$$

The preceding equation is a Taylor expansion of  $\tilde{r}(\tau)$  near  $\delta = t$ , as well as a signal transformation of the variable  $\delta$ . It's straightforward to demonstrate that:

$$x^{(j)}(t) = \left. \frac{d^j}{d\delta^j} (r(t - \delta)) \right|_{\delta=t} = (-1)^j a_j \quad (4.45)$$

Rewriting the expression for  $a_j$  with the changed variable :

$$a_j = \int_0^T Q(T, \tau) r(\tau) d\tau = \int_0^T Q(T, \tau) x(t - \tau) d\tau \quad (4.46)$$

The ultimate equation represents a causal estimator computed at a flexible time  $t$ , employing a constant estimation window of duration  $T$ :

$$a_j = x^{(j)}(t) = (-1)^j \int_0^T Q(T, \tau) x(t - \tau) d\tau \quad (4.47)$$

The digital realization of this estimator entails employing the trapezoidal integration approach to construct a digital Finite Impulse Response (FIR) filter. This process involves specifying  $T = T_s N$ , where  $T$  denotes the duration of the estimation window,  $T_s$  signifies a constant sampling interval and  $(N + 1)$  represents the total number of samples. By leveraging this approach, the estimator can be seamlessly integrated into digital systems, providing accurate derivative estimations within a defined time window:

$$a_j = x^{(j)}(t) \simeq (-1)^j \sum_{k=1}^{N+1} \alpha_k Q(T, \tau_k) x(t - \tau_k) \quad (4.48)$$

To minimize computational efficiency, the discretization of  $Q(T, \tau_k)$  is precomputed. Simultaneously, after the trapezoidal integration, we consider the coefficients  $\alpha_k$ :

$$\begin{aligned} \hat{Q}_{\alpha_k}(T, \tau_k) &= \begin{bmatrix} \alpha_1 Q(T, \tau_1) & \alpha_2 Q(T, \tau_2) & \dots & \alpha_{N+1} Q(T, \tau_{N+1}) \end{bmatrix} \\ \tau_k &= (k-1)T_s \\ \alpha_k &= \frac{T_s}{2}, k = 1 \text{ or } k = N+1 \\ \alpha_k &= T_s, k = 2, 3, \dots, N \end{aligned} \quad (4.49)$$

The vector  $\hat{x}(t - \tau_k)$ , housing the latest  $N + 1$  samples, is arranged vertically. With each sample clock cycle, the entire vector undergoes a shift operation. This process involves discarding the oldest value while incorporating the most recent value  $x(t)$  into the initial position of the vector. By employing this mechanism, the vector continuously maintains the most recent samples, facilitating efficient real-time processing of the input signal:

$$\hat{x}(t - \tau_k) = \begin{bmatrix} x(t) \\ x(t - T_s) \\ \vdots \\ x(t - T_s N) \end{bmatrix} \quad (4.50)$$

Ultimately, at each sample time, the product of  $\hat{Q}_{\alpha_k}(T, \tau_k)$  and  $\hat{x}(t - \tau_k)$  yields a real-time approximation of  $a_j$ :

$$a_j(t) = x^{(j)}(t) \simeq (-1)^j \sum_{k=1}^{N+1} \alpha_k Q(T, \tau_k) x(t - \tau_k) = (-1)^j \hat{Q}_{\alpha_k}(T, \tau_k) \hat{x}(t - \tau_k) \quad (4.51)$$

*Remark (1).* **A compromise between accuracy and immunity to noise:**

- The choice of the estimation window  $T$  plays a crucial role in striking a balance between immunity to noise and accuracy in derivative estimation.
- Lower  $T$  values offer advantages such as faster filter response and improved local signal modeling.
- Conversely, larger values of  $T$  enhance noise immunity and provide better high-frequency filtering properties.

*Remark (2).* **Impact of window Size on filter response**

- Smaller  $T$  values contribute to a more agile filter response, allowing it to adapt quickly to rapid changes in the input signal.

- This agility is particularly advantageous in scenarios where capturing transient signal features or responding promptly to signal variations is essential.

*Remark (3).* **Enhanced noise immunity with larger window sizes:**

- Larger  $T$  values are beneficial for improving noise immunity, as they enable the filter to effectively suppress high-frequency noise components.
- By encompassing a broader time window, the filter can integrate information from a more extended temporal range, thereby enhancing its ability to distinguish between signal and noise.

*Remark (4).* **Numerical precision and sampling frequency**

- In addition to window size, numerical precision is a critical concern in derivative estimation algorithms.
- Increasing the sampling frequency is an alternative approach to enhance precision, but it may be limited by hardware constraints, particularly if the system is already operating at maximum frequency.

### 4.3 Ultra-local model estimation

In systems where the system output  $y$  is impacted by the system input  $u$ , governed by a non-linear yet essentially smooth function  $\mathfrak{S}$ , the relationship between these variables is encapsulated in a differentiation equation, as articulated in Equation (4.52) :

$$\mathfrak{S}(t, u, \dot{u}, \dots, u^{(m)}, y, \dot{y}, \dots, y^{(n)}) = 0 \quad (4.52)$$

Here, the equation captures the relationship between time  $t$ , input signal  $u$ , its derivatives up to  $m^{th}$  order, output signal  $y$ , and its derivatives to the  $n^{th}$  order. This formulation provides a comprehensive view of how the system dynamics evolve over time in response to input stimuli [71].

When there exists an index  $i$  within the range  $(0, n)$  for which  $\frac{\partial \mathfrak{S}}{\partial y^{(i)}} \neq 0$ , the absolute function theorem offers a local simplification, revealing the relationship:

$$y^{(i)} = \mathfrak{S}(t, u, \dot{u}, \dots, u^{(m)}, y, \dot{y}, \dots, y^{(i-1)}, y^{(i+1)}, \dots, y^{(n)}) \quad (4.53)$$

This equation unravels the  $i^{th}$  derivative of the output  $y$  as a function of input and output derivatives, providing valuable insights into the intricate dynamics governing the system's behavior.

Fliess's model-free control methodology [75] suggests using an ultra-local model instead of complex nonlinear models, as shown in Equation (4.53). Equation (4.54) demonstrates an ultra-local representation that simplifies system behavior:

$$y^{(\nu)}(t) = F(t) + \beta u(t) \quad (4.54)$$

To optimize transient performance,  $y^{(\nu)}$  represents the  $\nu$ th-order derivative of the system output. On the other hand,  $F$  summarizes the unmodeled dynamics of the system, including unknown perturbations, and is constantly updated in real time using algebraic or derivative-based approaches. The non-physical coefficient  $\beta$  is carefully chosen to match  $y^{(\nu)}$  and  $\beta u$  in terms of magnitude and sign, enhancing closed-loop performance through iterative experimentation.

This simplified representation is an effective approximation of system dynamics over short time intervals, providing a practical basis for control synthesis and eliminating the need for complex nonlinear modeling, consequently streamlining gain tuning processes.

In the closed-loop system, Equation (4.54) incorporates an intelligent PID controller, with  $F(t)$  expressed as:

$$F(t) = y^{(\nu)}(t) - \beta u(t) \quad (4.55)$$

The system control input is formulated as:

$$u = -\frac{F - y_d^{(\nu)} + u_c}{\beta} \quad (4.56)$$

Here,  $y_d^{(\nu)}$  symbolizes the  $\nu^{th}$  derivative of the output desired trajectory, while  $u_c$  represents the causal feedback control input.

Incorporating these equations yields a comprehensive expression for system behavior, ensuring effective tracking and convergence towards the reference trajectory, as delineated in Equations (4.57), (4.58), and (4.59).

$$y^{(\nu)} = F + \beta \left( -\frac{F - y_d^{(\nu)} + u_c}{\beta} \right) = y_d^{(\nu)} - u_c \quad (4.57)$$

Afterwards,

$$y^{(\nu)} - y_d^{(\nu)} + u_c = 0 \quad (4.58)$$

$$e^{(\nu)} + u_c = 0 \quad (4.59)$$

Here,  $e^{(\nu)}$  represents the  $\nu^{th}$  derivative error of the tracking error ( $e = y - y_d$ ). The control signal  $u_c$  is selected to ensure asymptotic tracking, where the output approaches

the desired trajectory as expressed by Equation (4.60),

$$\lim_{t \rightarrow +\infty} e(t) = 0 \quad (4.60)$$

### 4.3.1 Intelligent PIDs (iPIDs)

The distinction between the intelligent and conventional controllers is inspired by the work of D'Andréa Novel [76]. The PID control, i.e.,  $u$ , can be computed based on the error  $e$ . Let's reformulate the control signal in Equation (4.56) using a conventional PID controller:

$$u = -\frac{F - \ddot{y}_d + K_P e + K_I \int e + K_D \dot{e}}{\beta} \quad (4.61)$$

Here, we set  $v = 2$  in Equation (4.54). By substituting Equation (4.54) into Equation (4.61), we define the intelligent proportional-integral-derivative controller (iPID) as:

This equation omits  $F$ , thereby removing uncertainties and unmodeled plant dynamics. Equation (4.61) meets tracking requirements through the careful adjustment of gains  $K_P, K_I, K_D$ , resulting in a stable linear differential equation with constant real coefficients.

Introducing various combinations of controller gains in a proportional-integral-derivative (PID) controller design can result in different types of controllers with unique control characteristics.

- When  $K_P \neq 0, K_I = 0, K_D = 0$ , it results in an intelligent Proportional (iP) controller.
- When  $K_P \neq 0, K_I = 0, K_D \neq 0$ , it leads to an intelligent Proportional-Derivative (iPD) controller.
- When  $K_P \neq 0, K_I \neq 0, K_D \neq 0$ , it forms an intelligent Proportional-Integral-Derivative (iPID) controller.

Each of these configurations provides distinct control characteristics that are tailored to meet specific system requirements.

### 4.3.2 Online estimation of the ultra-local model through algebraic estimation

The process of identifying the parameter  $F$  which is the basis of the ultra-local model entails employing algebraic estimation techniques, a methodology outlined by [77]. Alternatively, numerical differentiation, which involves estimating derivatives, serves as another viable approach outlined in Section 4.2.1.

Let's investigate the implications of Equation (4.54) based on classical operational calculus principles. Here, we replace the parameter  $F$  with  $F_{approx(Alg)}$ , an approximation assumed to be constant over the interval  $[t, t + T]$ . To calculate  $F_{approx(Alg)}$ , we follow the previous systematic approach.

Setting  $v = 1$  in Equation (4.54), we initiate the process by performing a Laplace transformation:

$$sY(s) - y(0) = \frac{F_{approx(Alg)}}{s} + \beta U(s) \quad (4.62)$$

Next, we proceed to differentiate Equation (4.62) with respect to  $s$  to exclude  $y(0)$ :

$$Y(s) - s \frac{d}{ds} Y(s) = -s^{-2} F_{approx(Alg)} + \beta \frac{d}{ds} U(s) \quad (4.63)$$

To further refine our analysis, we factor Equation (4.63) by  $\frac{1}{s^2}$ , aiming to integrate every coefficient once. This action efficiently filters and removes time derivatives, resulting in a low-pass filter that effectively reduces noise:

$$\frac{1}{s^2} Y(s) + \frac{1}{s} \frac{d}{ds} Y(s) = -\frac{1}{s^{-4}} F_{approx(Alg)} + \beta \frac{1}{s^3} \frac{d}{ds} U(s) \quad (4.64)$$

After careful observation, we discover that differentiating  $Y(s)$  in the operational domain corresponds to factorization by  $(-t\hat{a}$  in the time domain. Furthermore, the left factorization of  $Y(s)$  by  $\frac{1}{s^\alpha}$ ,  $\alpha \geq 1$  is consistent with looping integrals. To obtain a closed-form equation, we employ the Cauchy formula for occurring again integration of the operator  $\frac{1}{s^\alpha} \frac{d^n}{ds^n}$ , which results in only one integral:

$$\frac{1}{s^\alpha} \frac{d^n}{ds^n} Y(s) \longleftrightarrow \frac{(-1)^n}{(\alpha - 1)!} \int_0^t (t - \tau)^{\alpha-1} \tau^n y(\tau) d\tau \quad (4.65)$$

Finally, after a series of mathematical manipulations, we derive the approximated time domain equation (4.64) as:

$$F_{approx(Alg)} = \frac{-6}{t^3} \int_0^t (t - 2\tau) y(\tau) d\tau - \frac{6\beta}{t^3} \int_0^t \tau(t - \tau) u(\tau) d\tau \quad (4.66)$$

These sequential steps outline a systematic approach to estimating the parameter  $F_{approx(Alg)}$  using algebraic techniques, allowing for effective analysis and control of dynamic systems.

### 4.3.3 Online estimation of the ultra-local model through derivative estimation

The alternative method to  $F$  parameter estimation in Equation (4.54) involves identifying it through the derivative estimation of the noisy output signal  $y^{(v)}$  in Equation (4.55).

Recall that Equation (4.55) represents the expression for  $F(t)$ , where  $F$  is computed as the difference between the  $v$ -th derivative of the output signal  $y$  and the control input  $u$ .

Choosing  $v = 1$  yields:

$$F(t) = \dot{y}(t) - \beta u(t) \quad (4.67)$$

As discussed in Section 4.2.1.2, the first-order derivative estimation can be obtained through:

$$\dot{y} = a_1 = -\frac{6}{t^3} \int_0^t (t - 2\tau)y(\tau)d\tau \quad (4.68)$$

Thus, the function representing dynamics  $F_{approx(Der)}$  can be determined in real-time using the first-order derivative estimator, as shown in Equation (4.68):

$$F_{approx(Der)} = \dot{y} - \beta u(t) = -\frac{6}{t^3} \int_0^t ((t - 2\tau)y(\tau) - \beta u(t))d\tau \quad (4.69)$$

*Remark.* It's worth noting that if the designer opts to work with a second-order derivative, the ultra model can be estimated following a similar procedure as detailed in Section 4.2.2.3.

### 4.3.4 Discrete implementation of the ultra-local model through Derivative estimation

The procedures outlined in Eqs (4.66) and (4.69) for estimating the function  $F$  will be translated into digital implementations employing an FIR filter given in Section 4.2.3. For alternative implementations utilizing ALIEN filters, refer to [78].

Consider  $\tilde{y}(t)$  from Equation (4.67), approximated for  $\epsilon$  with  $j = 1$  in Equation (4.41):

$$\dot{y} \simeq \dot{y}(0) = a_1 = \frac{k}{q(t)} \int_0^t \tilde{Q}(t, \tau)y(\tau)d\tau; 0 < t < \epsilon \quad (4.70)$$

For a short time fixed-length window  $T$ :

$$\dot{y} = a_1 = \frac{k}{q(T)} \int_0^T \tilde{Q}(T, \tau) y(\tau) d\tau = \int_0^T \frac{k\tilde{Q}(T, \tau)}{q(T)} y(\tau) d\tau = \int_0^T Q(T, \tau) x(\tau) d\tau \quad (4.71)$$

The description estimates  $\dot{y} = a_1$  at  $t = 0$ .

1. Suppose Taylor expansion  $y(\tau)$  close to  $\tau = 0$  for  $\tilde{y}(\tau)$  :

$$\tilde{y}(\tau) \simeq y(\tau) = \sum_{j=0}^n a_1 \tau \Rightarrow a_1 = \int_0^T Q(T, \tau) y(\tau) d\tau \quad (4.72)$$

2. By transforming variables  $\tau \triangleq t - \theta$  and  $y(\theta) \triangleq x(t - \theta)$ :

$$y(\theta) \triangleq x(t - \theta) = \sum_{j=0}^n a_1 (t - \theta) \quad (4.73)$$

Consider this as a Taylor expansion  $\tilde{x}(\tau)$  for  $\theta = t$  with an inverse signal for the parameter  $\theta$ :

$$\dot{y}(t) = \left. \frac{d}{d\theta} (x(t - \theta)) \right|_{\theta=t} = (-1)a_1 \quad (4.74)$$

3. Again by transforming variables  $a_1$ :

$$a_1 = \int_0^T Q(T, \tau) x(\tau) d\tau = \int_0^T Q(T, \tau) y(t - \tau) d\tau \quad (4.75)$$

Thus,

$$\dot{y}(t) = a_1 = (-1) \int_0^T Q(T, \tau) y(t - \tau) d\tau \quad (4.76)$$

Applying Equation (4.48) results in :

$$\dot{y}(t) = a_1 \simeq (-1) \sum_{k=1}^{N+1} \alpha_k Q(T, \tau_k) y(t - \tau_k) \quad (4.77)$$

The representation of  $Q(T, \tau_k)$  in discrete form is theoretically estimated to decrease computational expenses, incorporating coefficients  $(\alpha_k)$  obtained through trapezoidal integration in Eqs (4.49) and (4.50).

Finally, the real-time update estimation of  $\dot{y}$  is provided by:

$$\dot{y}(t) = a_1(t) \simeq (-1) \sum_{k=1}^{N+1} \alpha_k Q(T, \tau_k) y(t - \tau_k) = (-1) \hat{Q}_{\alpha_k}(T, \tau_k) \hat{y}(t - \tau_k) \quad (4.78)$$

Ultimately, following the estimation of  $\dot{y}$  and referencing equation (4.69), the previous control input is employed to observe the present parameter  $F$ , which can be resolved using this expression:

$$\hat{F}_{k(DER)} = \dot{y}_k - \beta u_{k-1} \quad (4.79)$$

### 4.3.5 Discrete implementation of the ultra-local model through Algebraic estimation

Returning to the estimation method of  $F$  as shown in (4.66), An FIR filter is obviously appropriate for digital implementations. We enhanced the previous technique and incorporated the constrained fixed-length window  $T$  in a reverse integration approach to enable a viable online implementation of the estimator discussed in section 4.3.4.

The current instance,  $F$  is viewed as an independent temporal constant, which is unaffected by the signal during reverse integration. Following additional variable manipulations, equation (4.80) emerges as the definitive formulation for the algebraic estimator of  $F_{(Alg)}$ .

Changing a few factors, the representation of  $F$  in (4.66) becomes the following:

$$F = \frac{-6}{T^3} \int_0^T (t - 2\tau)y(t - \tau)d\tau - \frac{6\beta}{T^3} \int_0^T \tau(T - \tau)u(t - \tau)d\tau \quad (4.80)$$

A good variable selection entails storing the obtained discrete filter coefficients in a vector while accounting for trapezoidal integration coefficients. order to stay out of an algebraic loop, we use the prior value of  $F_{Alg}$  to estimate the present control input. As a result, the determination of  $F_{Alg}$  remains constant, posing no additional challenge in this approach, despite the approximation being made within a short sliding window  $T$ .

## 4.4 Algebraic Derivative Estimation for Position Control of a DC Motor

DC motors, despite their higher operating costs, are widely employed in industry sectors for their remarkable capacity for speed control [79]. Studies on DC motor position control have led to various techniques, adaptable across a wide range to track different load inputs.

Previous research has covered diverse applications such as motion control of disks [80], overhead crane control [81], liquid pumps [82], wheeled mobile manipulators [83], and positioning tables [84], considering Coulomb frictions and unmodeled dynamics. Recently, interest has grown in control techniques leveraging DC motors' high precision and speed. Time response is crucial for efficiency, especially as electromechanical applications shrink. Various control techniques, including integration of discrete-time algorithms [85], sliding mode control [86], classical control with adaptive approaches [87, 88], and adaptive control using bacterial foraging algorithms [89] has been investigated. Researchers have studied disturbance observer-based control [90] and optimization strategies using LQG controllers [91].

This section introduces a model-free control technique for DC motor position tracking, utilizing an ultra-local approach along with the simultaneous use of derivative and algebraic estimators. The methodology relies on FIR filters for effective implementation. Additionally, a straightforward and efficient tuning algorithm for the ultralocal model parameter is presented, along with a comparative study.

#### 4.4.1 DC motor modeling

The linear model utilized to describe the behavior of the DC motor encompasses considerations for Coulomb friction effects and unstable disturbance input, as discussed in [92]. The underlying dynamic equation for this model, which is based on Newton's second law, is as follows:

$$kV = J\ddot{\theta}_m + v\dot{\theta}_m + \hat{\Gamma}_c(\dot{\theta}_m) \quad (4.81)$$

The coefficients are as follows:

- $v$ : Signifies the viscous friction coefficient in  $\text{N} \cdot \text{m} \cdot \text{s}$ .
- $V$ : Represents the input voltage supplied to the DC motor system as a control input signal.
- $J$ : Denotes the inertia of the motor's gear in units of  $\text{kg} \cdot \text{m}^2$ .
- $k$ : Stands for the electromechanical factor, measured in  $\text{N} \cdot \text{m}/\text{V}$ .
- $\hat{\Gamma}_c$ : Represents the torque resulting from unknown friction affecting the motor's dynamics.

The non-linear component of friction, perceived as a disturbance, is described by the following equation:

**Table 4.1:** DC motor characteristics

Variable	Value
$k$ (N · m/V)	0.21
$J$ (kg·m <sup>2</sup> )	$6.87 \times 10^{-5}$
$v$ (N · m · s)	$1.041 \times 10^{-3}$
$n$	50

$$\hat{\Gamma}_c = \hat{\Gamma}_{Coul} \text{sign}(\dot{\hat{\theta}}_m) \quad (4.82)$$

$\hat{\Gamma}_{Coul}$  represents the static friction value exceeding the rotational velocity required to initiate rotation around the vertical axis. The terms  $\ddot{\hat{\theta}}_m$ ,  $\dot{\hat{\theta}}_m$ , and  $\hat{\theta}_m$  denote the angular acceleration (rad/s<sup>2</sup>), angular velocity (rad/s), and angular position (rad) of the motor, respectively. Parameter  $n$  denotes the motor gear reduction factor. Therefore,  $\theta_m = \hat{\theta}_m/n$ , where  $\theta_m$  refers to the motor's gear position and  $\hat{\theta}_m$  denotes the motor shaft's position. Additionally,  $\Gamma_c = \hat{\Gamma}_c/n$ , where  $\Gamma_c$  represents the motor gear's Coulomb friction torque.

The simulated DC motor model is the RH-8D-6006-E036AL-SP(N) [88], capable of rotating its shaft within the vertical axis in both left and right directions. Table 4.1 contains the parameter description.

#### 4.4.2 Problem formulation

Given the complexities of the DC motor dynamics described by equation (4.81), it's imperative to have a reference path  $\theta_m^*(t)$  that exhibits sufficient smoothness to facilitate trajectory tracking. It's important to consider input signal variations ( $V$ ) and output noise ( $\theta_m(t)$ ). In addition, the system is subject to unknown nonlinear effects caused by model parameter uncertainties or Coulomb friction. Despite these challenges, the feedback controller's main goal is to achieve precise asymptotic tracking of  $\theta_m^*(t)$  using the system output  $\theta_m(t)$ .

#### 4.4.3 Model-free Control Design

This section delineates the feedback control technique employed to achieve position tracking for the DC motor. The efficacy of model-free control laws hinges notably on the parameters  $\beta$  and iPID/iPD gains. Consequently, to enhance performance, a tuning methodology is expounded upon in sections 4.4.3.1 and 4.4.3.2 for  $\beta$  and iPD/iPID gains, respectively.

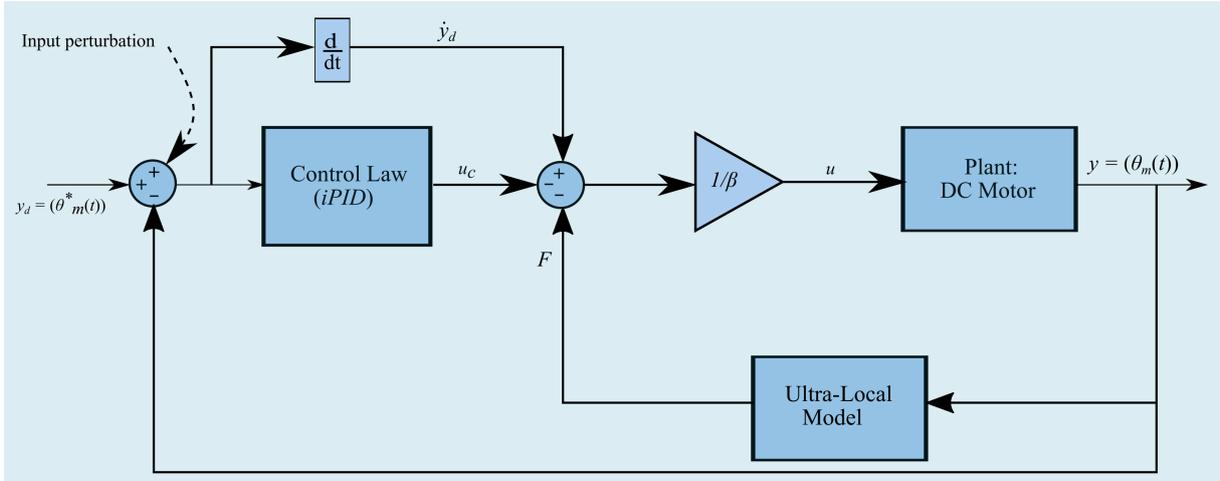


Figure 4.1: MFC scheme

#### 4.4.3.1 iPID/iPD gains tuning procedures

In the upcoming simulations, both iPD and iPID controllers are utilized, assuming a constant value of  $v = 1$  in equation (4.54):

$$F(t) = \dot{y}(t) - \beta u(t) \quad (4.83)$$

The closed-loop system incorporates an intelligent PID (i-PID) controller, as illustrated in Fig. ???. It operates according to the equation:

$$u = \frac{-F + y_d - K_I \int e - K_P e - K_D \dot{e}}{\beta} \quad (4.84)$$

Here,  $y_d$  denotes the reference trajectory,  $e = y_d - y$  represents the error tracking, and  $K_P, K_I, K_D$  are the conventional PID tuning gains.

For the iPD controller, the choice of  $K_P = \lambda^2$ ,  $K_I = 0$ , and  $K_D = 2\lambda$ ,  $\lambda \in \mathbb{R}^+$  ensures the closed loop's stability, characterized by two real negative poles at  $-\lambda$ :

$$\ddot{e} + \lambda^2 e + 2\lambda \dot{e} = 0 \quad (4.85)$$

On the other hand, the iPID controller is characterized by  $K_P = 3\lambda^2$ ,  $K_I = \lambda^3$ , and  $K_D = 3\lambda$ ,  $\lambda \in \mathbb{R}^+$ , resulting in three real negative poles within a stable closed loop at  $-\lambda$ :

$$\ddot{e} + \lambda^3 \int e + 3\lambda^2 e + 3\lambda \dot{e} = 0 \quad (4.86)$$

**4.4.3.2 Tuning procedure of  $\beta$** 

According to (4.84), the dynamics of the present output signal can be showed as follows:

$$\dot{y} = \tilde{F}(t) + \tilde{\beta}(t)u \quad (4.87)$$

in which  $\tilde{\beta}(t)$  and  $\tilde{F}(t)$  denote two unspecified time-varying functions. As a result, we can rewrite this equation to demonstrate the dependency on an additional term:

$$F = \tilde{F}(t) + (\tilde{\beta}(t) - \beta)u \quad (4.88)$$

$$u = - \left( \frac{\hat{F} - \dot{y}_d}{\beta} \right) - \mathcal{Q}(e) \quad (4.89)$$

$\mathcal{Q}(e)$  denotes the tracking error correction.

Integrating (4.86) into (4.89) and setting  $\mathcal{Q}(e)$  to 0, thus focusing solely on the feed-forward equation of the control signal, yields:

$$\dot{y} = \frac{\tilde{\beta}(t)}{\beta} \dot{y}_d + \left( \tilde{F}(t) - \frac{\tilde{\beta}(t)}{\beta} \hat{F} \right) \quad (4.90)$$

The sensitivity of  $\hat{F}$  measurement to specific dynamics underscores the criticality of selecting  $\beta$  to closely match the actual system gain and minimize changes in  $F$ .

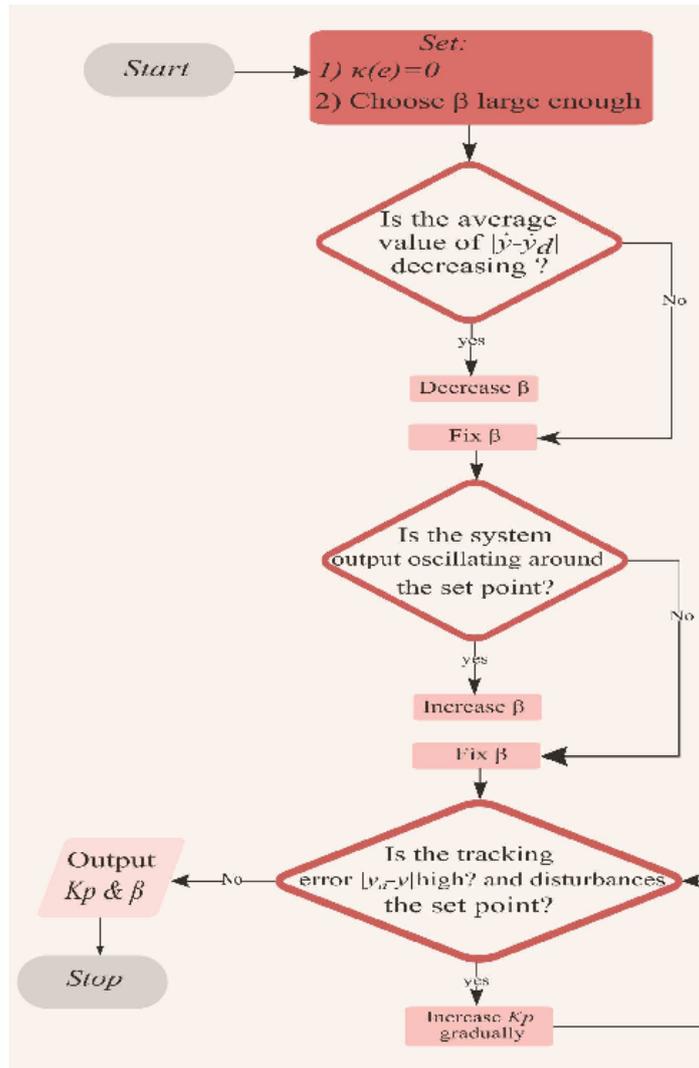


Figure 4.2: parameter's tuning procedure  $\beta$

As the parameter  $\beta$  tends towards infinity, the control input  $u$  outlined in equation (4.89) approaches  $\mathcal{Q}(e)$ . In this limiting scenario, the controller configuration simplifies to resemble that of a conventional PID controller, where the control action is primarily dictated by the error term.

Conversely, when  $\beta$  approaches zero from the positive side ( $\beta \rightarrow 0^+$ ), the control signal  $u$  predominantly relies on the estimated value  $\hat{F}$ . Consequently, tuning  $\beta$  becomes a straightforward process by setting  $\mathcal{Q}(e)$  to zero and observing the system's response to different  $\beta$  values. Initially, selecting a sufficiently high  $\beta$  results in an approximate suppression of the control signal, effectively rendering it close to zero. Subsequently, decreasing the value of  $\beta$  gradually elevates the average control signal until it reaches a level where the condition of the reference closed-loop ( $\dot{y} = \dot{y}_d$ ) is satisfied. This iterative

process allows for the fine-tuning of  $\beta$  to achieve the desired performance of the control system. For a visual representation and better understanding, refer to Figure 4.2, which illustrates the stepwise procedure for tuning the parameter  $\beta$ .

#### 4.4.4 Simulations results

Numerical simulations are undertaken to validate the effectiveness of the deployed control strategy in achieving rapid convergence of the tracking error to a confined region near zero. Additionally, the simulations aimed to assess the control law's ability the control effort low and ensure smooth transient responses, even in the presence of inaccuracies in the expert input, such as perturbations stemming from Coulomb friction effects or noisy measurements affecting the system.

The controller gains are selected by positioning poles of the closed-loop appropriately along the real negative axis. Throughout this search, the poles were positioned at  $p = -100$  rad/s, along with the PD and PID gains  $K_P, K_I, K_D$  were calculated using the expressions outlined in (4.85) and (4.86).

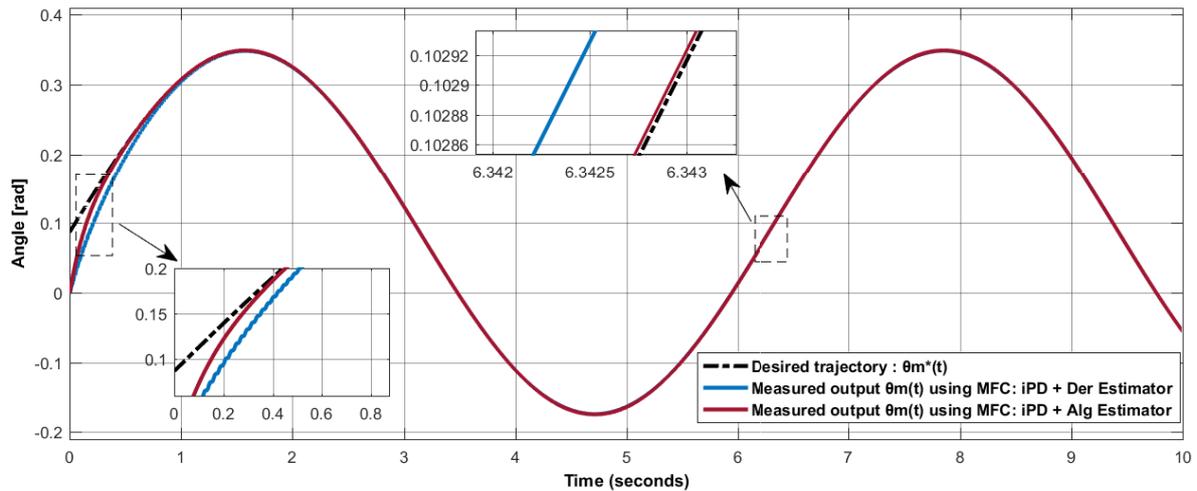
Two different trajectories demonstrate the effectiveness of the model-free control method via the ultra-local model based on derivative/algebraic estimators:

1. Sinusoidal Position Tracking: This trajectory follows a sinusoidal pattern defined by  $\theta_m^* = A \sin(\omega t) + B$ , where  $A = \pi/12$  (rad) represents the amplitude,  $B = \pi/36$  (rad) represents the bias, and  $\omega = 1$  (rad/s) represents the frequency.
2. Nominal Trajectory: This trajectory is defined by an eighth-order Bezier polynomial, smoothly using interpolation from 0 to 1 throughout the period of time  $[t_i, t_f]$ . It can be expressed as:

$$\theta_m^*(t) = \theta_m^*(t_i) + (\theta_m^*(t_f) - \theta_m^*(t_i))\varphi(\tau, t_f, t_i) \quad (4.91)$$

where  $\tau = (t - t_i) / (t_f - t_i)$ .

The study made use of a 10kHz sampling frequency. Equations (4.66) and (4.69) used a fixed sampling time of  $T_s = 10^{-4}$ s, an estimation window of  $T = 0.2$ s, as well as a total of  $N = 2000$  samples both the derivative and algebraic estimation techniques for  $F$ . These parameters were chosen to ensure accurate estimation of the system dynamics while



**Figure 4.3:** Sinusoidal trajectory - Position tracking employing derivatives and algebraic estimators based on the ultra-local model

balancing computational efficiency.

In the forthcoming section, (Der) designates the approximation of parameter  $F$  via angular velocity in Equation (4.54), while (Alg) represents the algebraic approximation of  $F$  in Equation (4.69). The measured output, denoted as ( $y$ ), the reference input  $y_d$ , and the control signal ( $u$ ) are symbolized by  $(\theta_m(t))$ ,  $(\theta_m^*(t))$ , and ( $V$ ), respectively.

The output's first-order derivative  $\dot{y}$ , utilized in Equation (4.66), undergoes estimation using a derivative estimator, evaluated with a theoretical derivator, as illustrated in Fig. 4.4. Various  $\beta$  values were employed for each estimation approach to ensure a desirable response, guided by Fig. 4.3.

The tuning procedure outlined in the preceding section was executed to use the control law for ultra-local model-based estimators. This method commences by setting the proportional gain  $K_P$  to null. Subsequently, a reference trajectory is introduced when the plant is stationary, and open-loop system responses are observed across different  $\beta$  values. Starting with  $\beta_{Alg} = 1000$  and  $\beta_{Der} = 1000$ , their values are systematically decreased until the control signal tends to oscillate, eventually reaching  $\beta_{Alg} = 1$  and  $\beta_{Der} = 45$ . Following this, the proportional gain  $K_P$  is adjusted to ensure a reasonable tracking error. Finally,  $\beta_{Alg}$  and  $\beta_{Der}$  are established at 3 and 100, respectively. This precise process ensures robust and efficient control law application.

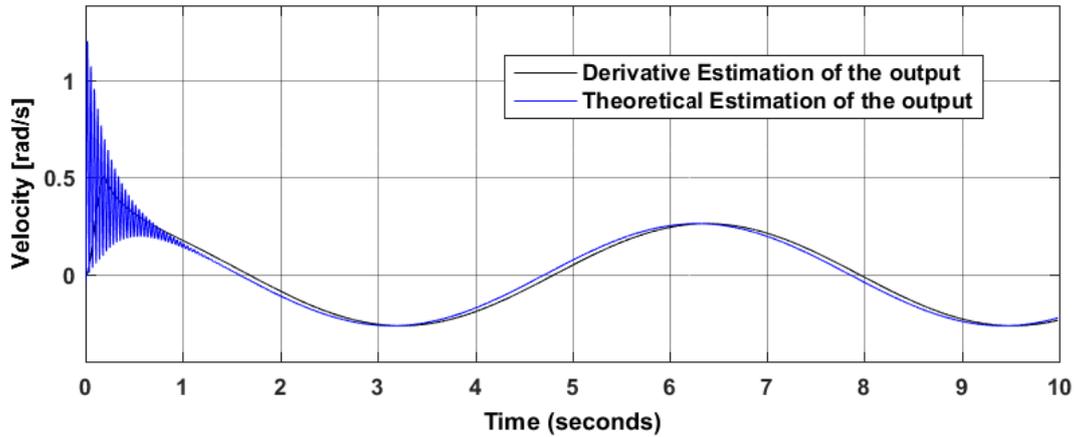


Figure 4.4: An estimate of the angular velocity derived from a sinusoidal trajectory  $\hat{y} = (\hat{\theta}_m(t))$

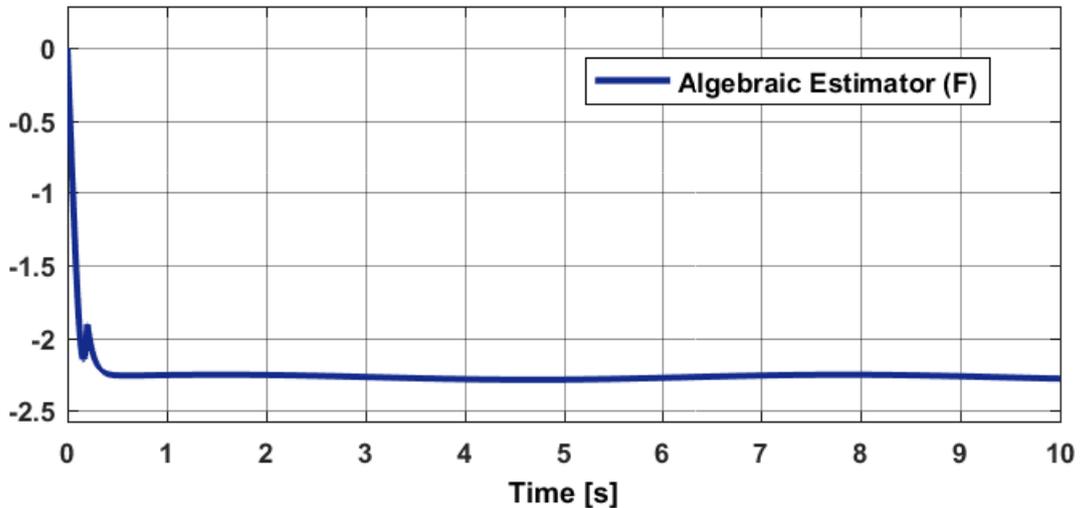


Figure 4.5: An estimate of the unmodeled dynamics for a sinusoidal trajectory

Figure 4.2 presents the numerical results obtained from the demonstrated controllers, showcasing their effectiveness in maintaining the desired sinusoidal output position despite the presence of various sources of uncertainties. At  $t = 0.01$  s, the online update procedure for both the function  $F$  and the angular velocity begins, as depicted in Figs. 4.4 and 4.5, respectively. Simultaneously, the voltage control effort signal, portrayed in Figs. 4.6 and 4.7, is actively utilized to counteract perturbations stemming from Coulomb friction effects, ensuring precise trajectory tracking and system stability.

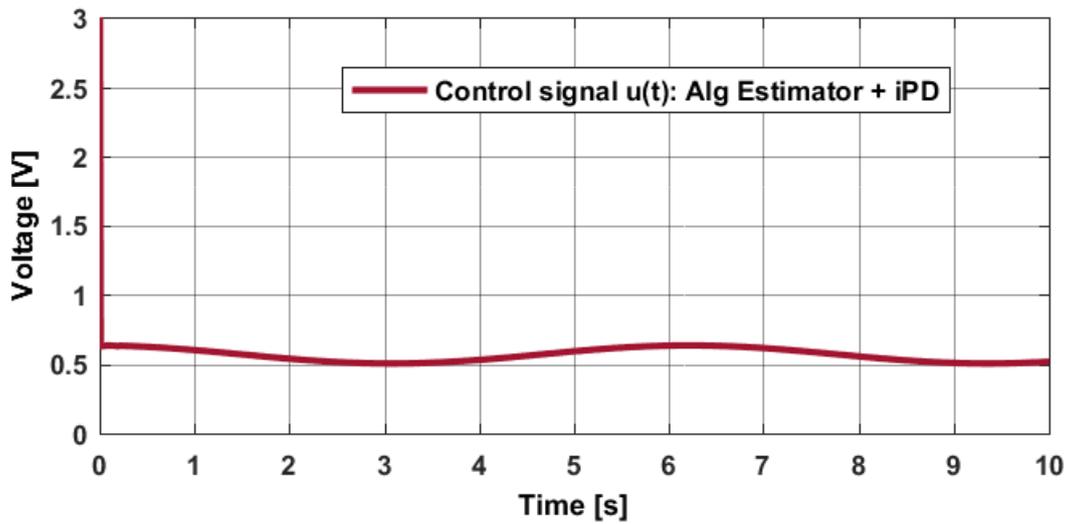


Figure 4.6: Sinusoidal trajectory - Control signal input using algebraic estimator and iPD

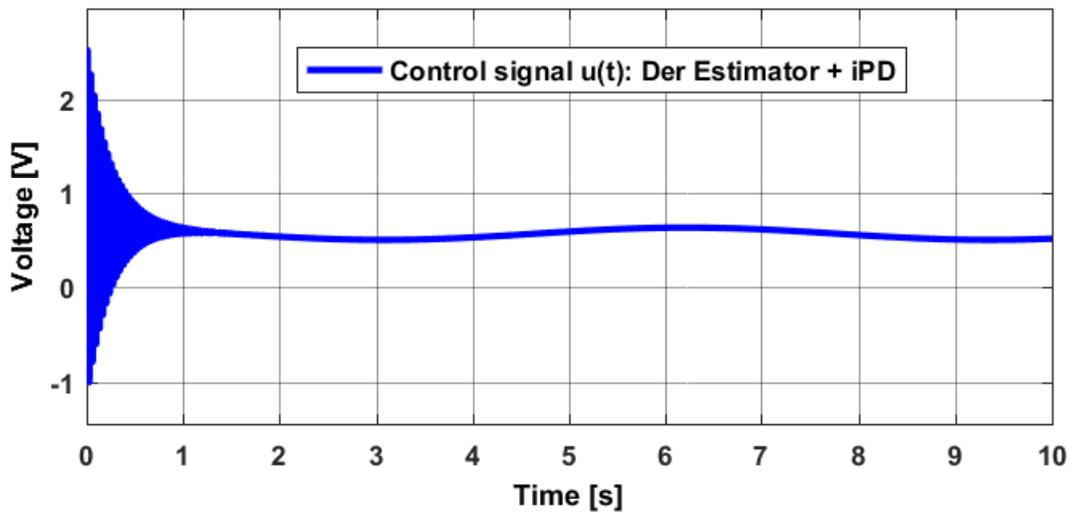
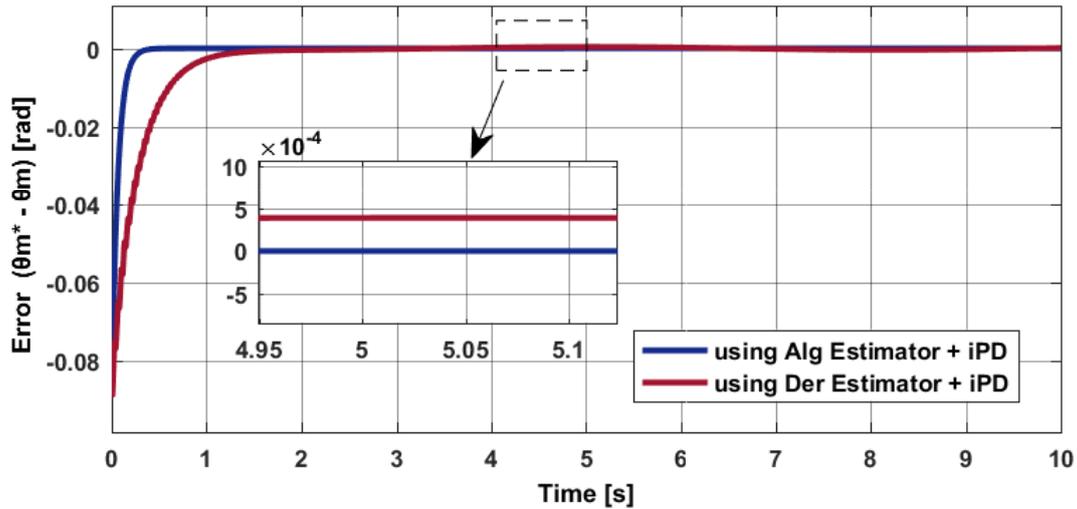


Figure 4.7: Sinusoidal trajectory - Control signal input using derivative estimator and iPD controllers

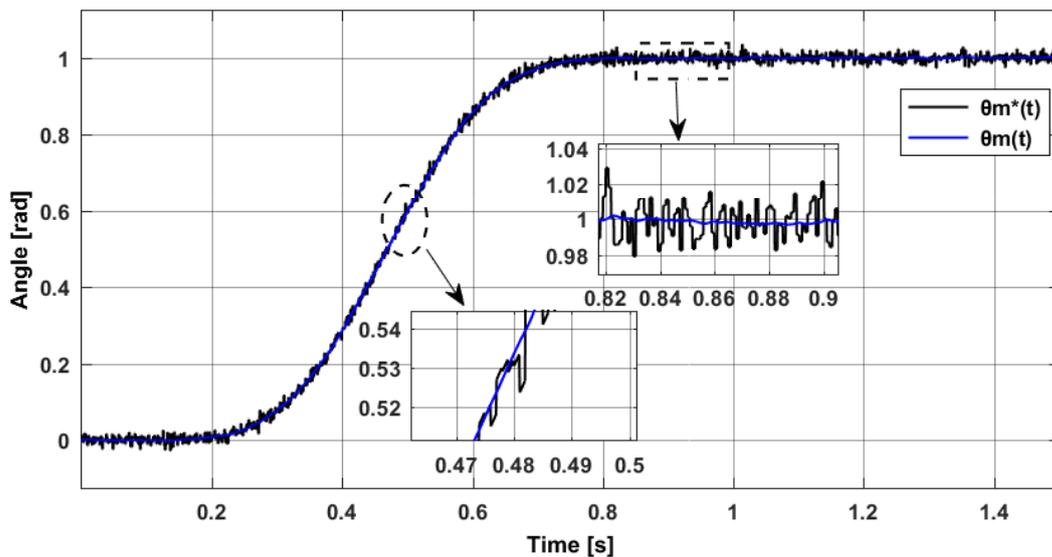
Figure 4.8 illustrates the error in position tracking employing the pair-stated controllers derived from the derivative and algebraic estimators. Through feedback control, the angular position of the DC motor is finely adjusted, minimizing state errors within a short timeframe while compensating for errors in model parameters.

**Table 4.2:** Sinusoidal trajectory - The efficiency of the control methods according to the ITAE, ISE, and IAE criteria

Control technique	ISE	IAE	ITAE
iPD + Derivative Estimator	$0.12 \times 10^{-2}$	0.03	0.03
iPD + Algebraic Estimator	$0.48 \times 10^{-3}$	0.01	0.002
Classical PD	$0.4 \times 10^{-2}$	0.19	0.96



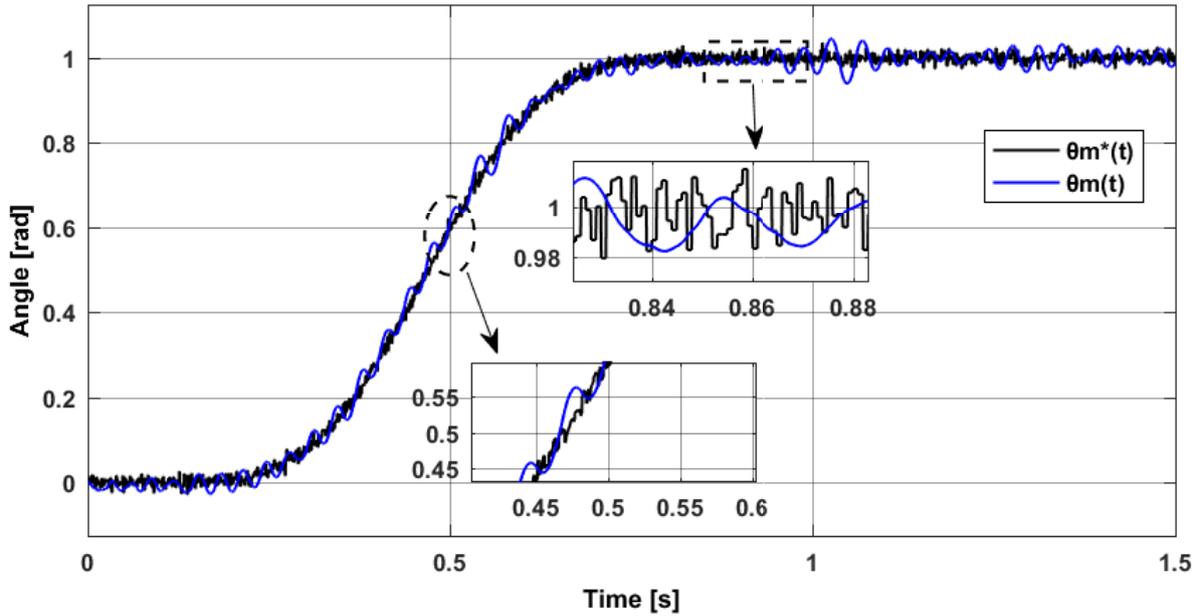
**Figure 4.8:** Sinusoidal trajectory - Error tracking with algebraic/derivative estimators and iPD



**Figure 4.9:** Perturbed Bezier's trajectory - The DC motor's position tracking using iPID with algebraic estimator and considering Coulomb's friction effects

**Table 4.3:** Perturbed Bezier's trajectory - The efficiency of the control methods according to the ITAE, ISE, and IAE criteria

Control technique	ISE	IAE	ITAE
iPD + Derivative Estimator	$0.41 \times 10^{-2}$	0.09	0.21
iPD + Algebraic Estimator	$0.1 \times 10^{-2}$	0.08	0.19
Classical PID	$0.86 \times 10^{-2}$	0.26	0.47

**Figure 4.10:** Perturbed Bezier's trajectory - Position tracking of the DC motor using iPID with derivative estimator and considering Coulomb's friction effects

The investigation findings demonstrate remarkable efficiency in model-free control tracking, particularly enhanced when incorporating an integral component. In comparison, the derivative scheme yielded a noisier control signal, rendering the system less robust. While expanding the estimation window can enhance filtering features, it introduces a delay that may potentially destabilize the closed-loop system.

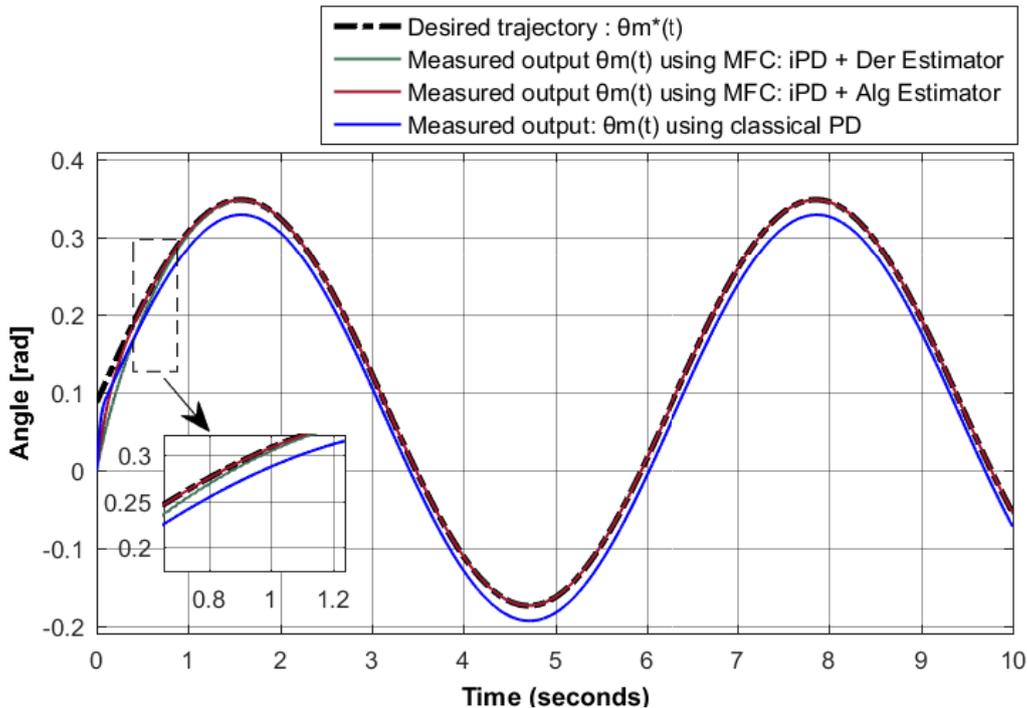
Figures 4.9 and 4.10 present cases where an eighth-order Bezier's polynomial input disturbance signal is applied, generating significant deformations in the input signal. This setup allows for the evaluation of the controller's smoothing effectiveness in cases utilizing derivative and algebraic estimators.

Additionally, the performance evaluation of the control methodologies has been conducted based on integral time absolute tracking error (*ITAE*), integral absolute tracking error (*IAE*), and integral squared tracking error (*ISE*). These metrics are defined as follows:

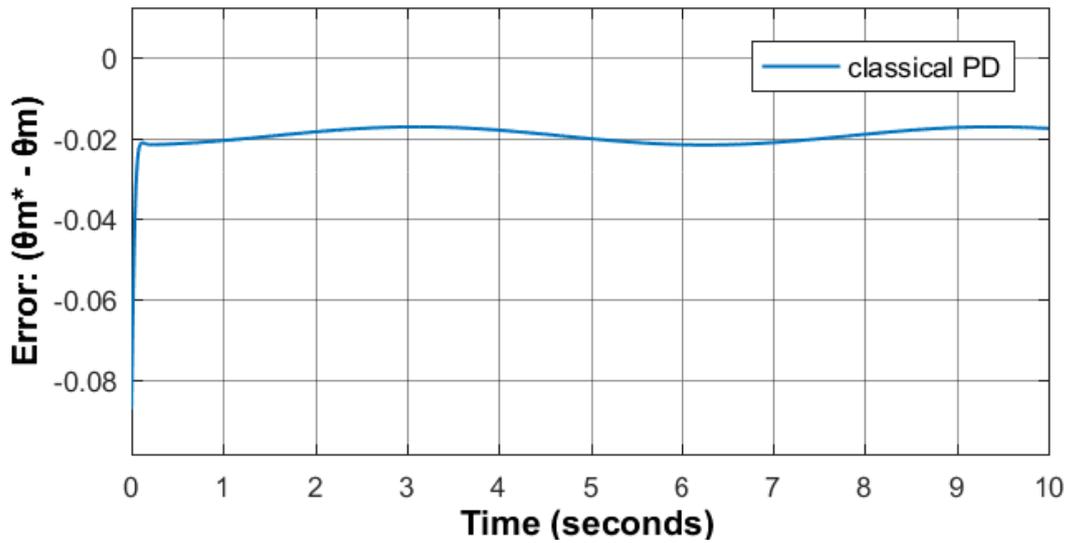
$$\begin{aligned}
ITAE &= \int_{t_i}^{t_f} t |e_{\theta_m}(t)| dt, \\
IAE &= \int_{t_i}^{t_f} |e_{\theta_m}(t)| dt, \\
ISE &= \int_{t_i}^{t_f} e_{\theta_m}^2(t) dt,
\end{aligned} \tag{4.92}$$

where  $t_i = 0$  s and  $t_f = 10$  s represent the initial and final times of the simulation, respectively. While  $ISE$  and  $IAE$  metrics treat all tracking errors equally,  $ITAE$  places significant emphasis on time. Errors that evolve slowly are heavily penalized in the  $ITAE$  standard, while those occurring slightly earlier are disregarded.

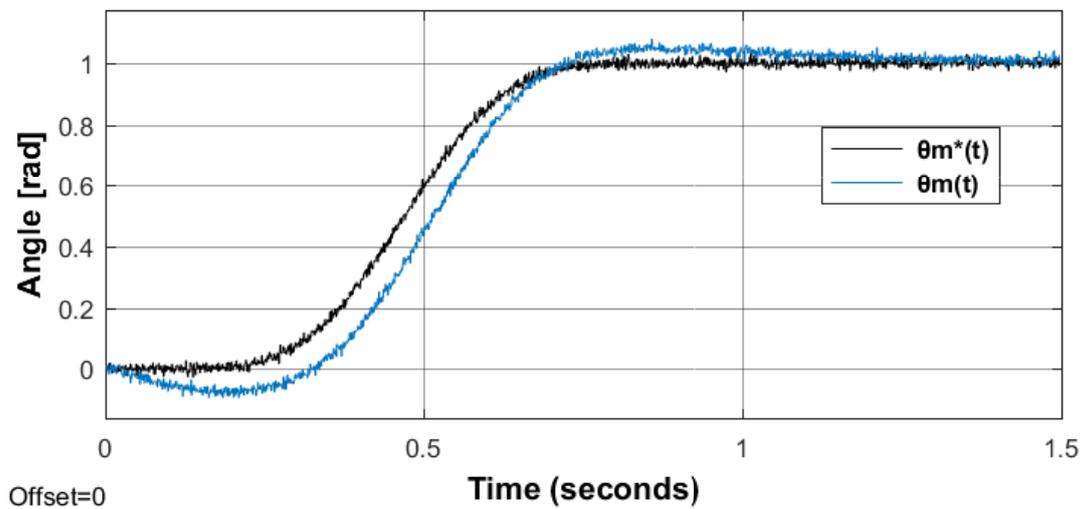
The comparative analysis of the obtained results is presented in Tables 4.2 and 4.3. The findings indicate that the proposed control approach utilizing the ultra-local model based on the algebraic estimator exhibits a slight advantage over the derivative estimator-based model. This suggests that the algebraic estimator may provide better performance in terms of tracking accuracy and robustness.



**Figure 4.11:** Sinusoidal trajectory: Comparing conventional PD and derivative/algebraic estimators with iPD



**Figure 4.12:** Sinusoidal trajectory - Error tracking with traditional PD



**Figure 4.13:** Perturbed Bezier's trajectory - The DC motor position tracking using traditional PID considering Coulomb's friction effects

Finally, a comparison is drawn between the proposed ultra-local model-based algebraic/derivative estimator approaches and a classical PID controller, which remains one of the most commonly employed control strategies. In this comparison, a classical PD controller is employed for tracking the sinusoidal trajectory, while a classical PID controller is utilized for the perturbed Bezier's trajectory. The PD/PID controllers are manually tuned to achieve rapid closed-loop responses with minimal overshoots along the desired trajectory.

The results obtained from this comparison are illustrated in Figures 4.11 and 4.12. In the transient region, it is observed that the iPD controllers track the reference trajectory more rapidly compared to traditional PD controllers. This enhanced performance can be attributed to the incorporation of the  $\dot{y}_r$  factor in the control law, as outlined in Eq. (4.52). Furthermore, the proposed techniques demonstrate accurate and robust tracking results, effectively mitigating disturbances caused by Coulomb friction, as evident in Figs. 4.9, 4.10, and 4.13.

At  $t = 0.25$  s, the PID controller exhibits overshoot, which can be minimized by sacrificing tracking speed. The comparison of performance metrics including ISE, IAE, and ITAE criteria is summarized in Tables 2 and 3. Furthermore, the proposed approaches offer superior tracking performance, particularly in scenarios with rapid trajectory changes or external disturbances. The incorporation of advanced control laws, such as iPD and iPIDs, allows for faster response times, reduced overshoot, and smoother trajectory tracking compared to classical PID controllers. This translates to improved system efficiency, stability, and overall operational performance.

## 4.5 Conclusion

This study investigated and implemented recent state observation techniques for the ultra-local model used in the model-free control approach, employing two distinct methodologies: an ultra-local model-based algebraic estimator and an ultra-local model-based derivative estimator. The estimation process utilized Taylor's expansion coefficients of arbitrary order, with discrete implementation details presented through operational calculus and finite impulse response (FIR) filters.

A comprehensive comparative analysis between algebraic and derivative estimators was conducted, revealing significant insights into their performance in state observation. Various tests were performed on a position trajectory control scheme for a DC motor subjected to disturbances and uncertainties. The ultra-local model design demonstrated independence from initial conditions, requiring only the measurement of the system's input and output values.

Key findings highlighted that the proposed design effectively reduced nonlinearities and enhanced robustness against disturbances arising from Coulomb friction effects. Both algebraic and derivative estimation processes were computed in real-time, ensuring quick and accurate state observation.

The study's outcomes were validated through numerical simulations, which compared the recommended approaches with other control strategies. The simulations confirmed the

feasibility and reliability of the ultra-local model, particularly emphasizing the superior performance of the algebraic method. The robust tracking capability against disturbances and uncertainties surpassed that of classical PID controllers.

Moreover, challenges such as high-frequency sampling and increased computational effort associated with longer filter lengths were acknowledged. These challenges were mitigated by leveraging powerful and cost-effective hardware solutions available in the market.

In addition to demonstrating the efficiency of the closed-loop control estimators, this research also provides valuable insights and simplifies several aspects of the digital implementation of these modern controllers.

In conclusion, the integration of advanced algebraic and derivative estimators into model-free control systems significantly enhanced the control performance of DC motor position tracking. The study provides valuable insights into the practical application of these estimators, paving the way for more resilient and efficient control strategies in DC motor systems.

# General Conclusion

The conclusion of this research journey reveals the profound impact of incorporating algebraic and derivative estimators into model-free control systems, particularly in terms of improving state observation and control strategy for DC motor position tracking and variable-speed wind turbine systems. Through strict comparison analysis and simulation experiments, the critical importance of robust state observation in enhancing control system resilience has been revealed.

In the domain of DC motor position tracking, the utilization of the ultra-local model coupled with algebraic and derivative estimators has yielded promising results, showcasing superior performance, especially with the algebraic method. Notably, robust tracking against Coulomb friction disturbances and uncertainties has surpassed conventional PID controllers, underscoring the efficacy of leveraging algebraic principles for fast and accurate estimation.

Further to variable-speed wind turbine systems, investigation into model-free controllers has revealed exceptional tracking performance, with the algebraic estimator emerging as an attractive alternative to traditional state observers. This not only eliminates the need for model-based observers but also ensures strong closed-loop performance that outperforms classical control techniques, especially when used with the FAST simulator. The practicality and simplicity of model-free control make it a compelling solution for regulating wind turbine systems, particularly in situations where obtaining an accurate dynamical model is difficult.

Moreover, the integration of flatness-based model-free control within wind turbine systems has demonstrated its efficacy across diverse operational scenarios, showcasing exceptional capabilities in optimizing wind energy capture and finely regulating power generation across varying speeds. This approach significantly contributes to enhancing system robustness, ensuring reliable performance even in the face of uncertainties and disturbances.

Finally, the combination of model-free techniques with algebraic and flatness control principles represents an important development in control system design, establishing up an

exciting prospect for improving overall control effectiveness and robustness in engineering applications, particularly wind energy systems. As this research journey comes to an end, it opens the way for further exploration and advancement, setting the edge of control engineering toward greater efficiency, reliability, and sustainability in engineering systems.

# General Conclusion

The conclusion of this research journey reveals the profound impact of incorporating algebraic and derivative estimators into model-free control systems, particularly in terms of improving state observation and control strategy for DC motor position tracking and variable-speed wind turbine systems. Through strict comparison analysis and simulation experiments, the critical importance of robust state observation in enhancing control system resilience has been revealed.

In the domain of DC motor position tracking, the utilization of the ultra-local model coupled with algebraic and derivative estimators has yielded promising results, showcasing superior performance, especially with the algebraic method. Notably, robust tracking against Coulomb friction disturbances and uncertainties has surpassed conventional PID controllers, underscoring the efficacy of leveraging algebraic principles for fast and accurate estimation.

Further to variable-speed wind turbine systems, investigation into model-free controllers has revealed exceptional tracking performance, with the algebraic estimator emerging as an attractive alternative to traditional state observers. This not only eliminates the need for model-based observers but also ensures strong closed-loop performance that outperforms classical control techniques, especially when used with the FAST simulator. The practicality and simplicity of model-free control make it a compelling solution for regulating wind turbine systems, particularly in situations where obtaining an accurate dynamical model is difficult.

Moreover, the integration of flatness-based model-free control within wind turbine systems has demonstrated its efficacy across diverse operational scenarios, showcasing exceptional capabilities in optimizing wind energy capture and finely regulating power generation across varying speeds. This approach significantly contributes to enhancing system robustness, ensuring reliable performance even in the face of uncertainties and disturbances.

Finally, the combination of model-free techniques with algebraic and flatness control principles represents an important development in control system design, establishing

up an exciting prospect for improving overall control effectiveness and robustness in engineering applications, particularly wind energy systems. As this research journey comes to an end, it opens the way for further exploration and advancement, setting the edge of control engineering toward greater efficiency, reliability, and sustainability in engineering systems.

# Bibliography

- [1] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. 1960.
- [2] Rudolf Emil Kalman et al. Contributions to the theory of optimal control. *Bol. soc. mat. mexicana*, 5(2):102–119, 1960.
- [3] Håkan Hjalmarsson. Iterative feedback tuning—an overview. *International journal of adaptive control and signal processing*, 16(5):373–395, 2002.
- [4] Marco A Wiering and Martijn Van Otterlo. Reinforcement learning. *Adaptation, learning, and optimization*, 12(3):729, 2012.
- [5] Michel Fliess, Jean Lévine, Philippe Martin, and Pierre Rouchon. Flatness and defect of non-linear systems: introductory theory and examples. *International journal of control*, 61(6):1327–1361, 1995.
- [6] Ryan R Chladny and Charles R Koch. Flatness-based tracking of an electromechanical variable valve timing actuator with disturbance observer feedforward compensation. *IEEE Transactions on Control Systems Technology*, 16(4):652–663, 2008.
- [7] John Y Hung, Weibing Gao, and James C Hung. Variable structure control: A survey. *IEEE transactions on industrial electronics*, 40(1):2–22, 1993.
- [8] Kemin Zhou and John Comstock Doyle. *Essentials of robust control*, volume 104. Prentice hall Upper Saddle River, NJ, 1998.
- [9] Gang Tao. *Adaptive control design and analysis*, volume 37. John Wiley & Sons, 2003.
- [10] Rolf Isermann and Marco Münchhof. *Identification of dynamic systems: an introduction with applications*, volume 85. Springer, 2011.
- [11] Frank L Lewis, Draguna Vrabić, and Vassilis L Syrmos. *Optimal control*. John Wiley & Sons, 2012.

- [12] Karl J Åström. *Introduction to stochastic control theory*. Courier Corporation, 2012.
- [13] Francesco Borrelli and Tamás Keviczky. Distributed lqr design for identical dynamically decoupled systems. *IEEE Transactions on Automatic Control*, 53(8):1901–1912, 2008.
- [14] Libor Pekař and Radek Matuš. A suboptimal shifting based zero-pole placement method for systems with delays. *International Journal of Control, Automation and Systems*, 16(2):594–608, 2018.
- [15] Bernard Charlet, Jean Lévine, and Riccardo Marino. On dynamic feedback linearization. *Systems & Control Letters*, 13(2):143–151, 1989.
- [16] Herbert G Tanner and Kostas J Kyriakopoulos. Backstepping for nonsmooth systems. *Automatica*, 39(7):1259–1265, 2003.
- [17] Warren E Dixon, Aman Behal, Darren M Dawson, and Siddharth P Nagarkatti. *Nonlinear control of engineering systems: a Lyapunov-based approach*. Springer Science & Business Media, 2003.
- [18] Antonio Sala and Alicia Esparza. Reduced-order controller design via iterative identification and control. *European journal of control*, 9(1):105–117, 2003.
- [19] Brian DO Anderson. Failures of adaptive control theory and their resolution. 2005.
- [20] RE Skelton. Model error concepts in control design. *International Journal of Control*, 49(5):1725–1753, 1989.
- [21] Zhong-Sheng Hou and Zhuo Wang. From model-based control to data-driven control: Survey, classification and perspective. *Information Sciences*, 235:3–35, 2013.
- [22] Michel Fliess and Cédric Join. Model-free control. *International Journal of Control*, 86(12):2228–2252, 2013.
- [23] Marco C Campi, Andrea Lecchini, and Sergio M Savaresi. Virtual reference feedback tuning: a direct method for the design of feedback controllers. *Automatica*, 38(8):1337–1346, 2002.
- [24] Jingqing Han. From pid to active disturbance rejection control. *IEEE transactions on Industrial Electronics*, 56(3):900–906, 2009.
- [25] Douglas A Bristow, Marina Tharayil, and Andrew G Alleyne. A survey of iterative learning control. *IEEE control systems magazine*, 26(3):96–114, 2006.

- [26] Loïc Michel, Cédric Join, Michel Fliess, Pierre Sicard, and Ahmed Chériti. Model-free control of dc/dc converters. In *2010 IEEE 12th Workshop on Control and Modeling for Power Electronics (COMPEL)*, pages 1–8, 2010.
- [27] Younes Al Younes, Ahmad Drak, Hassan Noura, Abdelhamid Rabhi, and Ahmed El Hajjaji. Model-free control of a quadrotor vehicle. In *2014 International conference on unmanned aircraft systems (ICUAS)*, pages 1126–1131. IEEE, 2014.
- [28] Frédéric Lafont, Jean-François Balmat, Nathalie Pessel, and Michel Fliess. A model-free control strategy for an experimental greenhouse with an application to fault accommodation. *Computers and Electronics in Agriculture*, 110:139–149, 2015.
- [29] Younes Al Younes, Ahmad Drak, Hassan Noura, Abdelhamid Rabhi, and Ahmed El Hajjaji. Robust model-free control applied to a quadrotor uav. *Journal of Intelligent & Robotic Systems*, 84(1):37–52, 2016.
- [30] Taghreed Mohammad Ridha and Claude H Moog. Model free control for type-1 diabetes: A fasting-phase study. *IFAC-PapersOnLine*, 48(20):76–81, 2015.
- [31] Ouassim Bara, Michel Fliess, Cédric Join, Judy Day, and Seddik M. Djouadi. Model-free immune therapy: A control approach to acute inflammation. In *2016 European Control Conference (ECC)*, pages 2102–2107, 2016.
- [32] Shida Liu, Zhongsheng Hou, and Jian Zheng. Attitude adjustment of quadrotor aircraft platform via a data-driven model free adaptive control cascaded with intelligent pid. In *2016 Chinese Control and Decision Conference (CCDC)*, pages 4971–4976, 2016.
- [33] Haoping Wang, Xuefei Ye, Yang Tian, Gang Zheng, and Nicolai Christov. Model-free-based terminal smc of quadrotor attitude and position. *IEEE Transactions on Aerospace and Electronic Systems*, 52(5):2519–2528, 2016.
- [34] Xinyi Zhang, Haoping Wang, Yang Tian, Laurent Peyrodie, and Xikun Wang. Model-free based neural network control with time-delay estimation for lower extremity exoskeleton. *Neurocomputing*, 272:178–188, 2018.
- [35] Raul-Cristian Roman, Radu-Emil Precup, and Radu-Codrut David. Second order intelligent proportional-integral fuzzy control of twin rotor aerodynamic systems. *Procedia Computer Science*, 139:372–380, 2018. 6th International Conference on Information Technology and Quantitative Management.

- [36] Mircea-Bogdan Radac, Radu-Emil Precup, and Emil M. Petriu. Model-free primitive-based iterative learning control approach to trajectory tracking of mimo systems with experimental validation. *IEEE Transactions on Neural Networks and Learning Systems*, 26(11):2925–2938, 2015.
- [37] Kaihui Zhao, Tonghuan Yin, Changfan Zhang, Jing He, Xiangfei Li, Yue Chen, Ruirui Zhou, and Aojie Leng. Robust model-free nonsingular terminal sliding mode control for pmsm demagnetization fault. *IEEE Access*, 7:15737–15748, 2019.
- [38] Saim Ahmed, Haoping Wang, and Yang Tian. Model-free control using time delay estimation and fractional-order nonsingular fast terminal sliding mode for uncertain lower-limb exoskeleton. *Journal of Vibration and Control*, 24(22):5273–5290, 2018.
- [39] Ghazally IY Mustafa, HP Wang, and Yang Tian. Vibration control of an active vehicle suspension systems using optimized model-free fuzzy logic controller based on time delay estimation. *Advances in Engineering Software*, 127:141–149, 2019.
- [40] Cédric Join, Michel Fliess, and Frédéric Chaxel. Model-free control as a service in the industrial internet of things: Packet loss and latency issues via preliminary experiments. In *2020 28th Mediterranean Conference on Control and Automation (MED)*, pages 299–306. IEEE, 2020.
- [41] Michel Fliess and Cédric Join. Machine learning and control engineering: The model-free case. In *Proceedings of the Future Technologies Conference*, pages 258–278. Springer, 2020.
- [42] Michel Fliess, Jean Lévine, Philippe Martin, and Pierre Rouchon. A lie-backlund approach to equivalence and flatness of nonlinear systems. *IEEE Transactions on automatic control*, 44(5):922–937, 1999.
- [43] Pierre Rouchon, Michel Fliess, Jean Lévine, and Philippe Martin. Flatness, motion planning and trailer systems. In *Proceedings of 32nd IEEE Conference on Decision and Control*, pages 2700–2705. IEEE, 1993.
- [44] M Fliess, J Lévine, Ph Martin, and P Rouchon. Towards a new differential geometric setting in nonlinear control. *Proc. Internat. Geometric. Coll., Moscow*, 1993.
- [45] Gerasimos G Rigatos. *Nonlinear control and filtering using differential flatness approaches: applications to electromechanical systems*, volume 25. Springer, 2015.

- [46] Koushil Sreenath, Koushil Sreenath, Taeyoung Lee, Taeyoung Lee, Vijay Kumar, Vijay Kumar, and Vijay Kumar. Geometric control and differential flatness of a quadrotor uav with a cable-suspended load. *null*, 2013.
- [47] Arvo Kaldmäe, Arvo Kaldmäe, Ülle Kotta, Ülle Kotta, Christian Meurer, Christian Meurer, Ashutosh Simha, and Ashutosh Simha. Event-based control for differentially flat systems: Application to autonomous underwater vehicle. *IFAC-PapersOnLine*, 2019.
- [48] Ningbo An, Qingyun Wang, and Xiaochuan Zhao. Differential flatness-based distributed control of underactuated robot swarms. *null*, 2023.
- [49] Silvère Bonnabel, Silvere Bonnabel, Xavier Claeys, Claeys Xavier, and Xavier Claeys. The industrial control of tower cranes: An operator-in-the-loop approach [applications in control]. *IEEE Control Systems Magazine*, 2020.
- [50] David Schlipf, David Schlipf, Po-Wen Cheng, and Po Wen Cheng. Flatness-based feedforward control of wind turbines using lidar. *IFAC Proceedings Volumes*, 2014.
- [51] M. Benaouadj, M. Benaouadj, A. Aboubou, A. Aboubou, M. Y. Ayad, M.Y. Ayad, M. Y. Ayad, Mohamed Becherif, M. Becherif, Mohamed Becherif, M. Bahri, M. Bahri, M. Bahri, and M. Bahri. Control algorithm of fuel cell hybrid source based on the flatness propriety. *null*, 2014.
- [52] Igyso Zafeiratou, Igyso Zafeiratou, I. Zafeiratou, Ionela Prodan, Ionela Prodan, Laurent Lefèvre, Laurent Lefèvre, Laurent Lefèvre, Laurent Praly, and Laurent Pietrac. Meshed dc microgrid hierarchical control: A differential flatness approach. *Electric Power Systems Research*, 2020.
- [53] Andrea D’Ambrosio, Andrea D’Ambrosio, Andrea D’Ambrosio, Dario Spiller, Dario Spiller, Fabio Curti, and Fabio Curti. Improved magnetic charged system search optimization algorithm with application to satellite formation flying. *Engineering Applications of Artificial Intelligence*, 2020.
- [54] Juri Belikov, Juri Belikov, Ülle Kotta, Ülle Kotta, Seshadhri Srinivasan, S. Srinivasan, Seshadhri Srinivasan, S. Srinivasan, Arvo Kaldmäe, Arvo Kaldmäe, Kristina Halturina, and K. Halturina. On exact feedback linearization of hvac systems. *null*, 2013.
- [55] Johannes Stephan, Johannes Stephan, Ole Pfeifle, Ole Pfeifle, Stefan Notter, Stefan Notter, Federico Pinchetti, Federico Pinchetti, Walter Fichter, and Walter Fichter.

- Precise tracking of extended three-dimensional dubins paths for fixed-wing aircraft. *Journal of Guidance Control and Dynamics*, 2020.
- [56] Tobias Barth, Tobias Barth, Jens Weber, Jens Weber, O. Woywode, Oliver Woywode, Steffen Bernet, and Steffen Bernet. New flatness-based control of a high-voltage generator for medical x-ray applications. *The Journal of Engineering*, 2018.
- [57] Due Tran Minh, Due Tran Minh, Phuoc Vo Tan, and Phuoc Vo Tan. Flatness-based single tank water level control under actuator constraint. *null*, 2021.
- [58] P.B. Ngancha, P. B Ngancha, K. Kusakana, Kanzumba Kusakana, Elisha Didam Markus, and Elisha Didam Markus. A survey of differential flatness-based control applied to renewable energy sources. *null*, 2017.
- [59] Hirsh Majid, Hirsh Majid, Hassane Abouaïssa, Hassane Abouaïssa, Daniel Jolly, Daniel Jolly, Gildas Morvan, and Gildas Morvan. Real-time dynamic traffic routing using variable structure control. *null*, 2013.
- [60] G Rigatos, Pierluigi Siano, and Ivan Arsie. A differential flatness theory approach to embedded adaptive control of turbocharged diesel engines. *Intelligent Industrial Systems*, 2(4):349–370, 2016.
- [61] Veit Hagenmeyer and Emmanuel Delaleau. Exact feedforward linearization based on differential flatness. *International Journal of Control*, 76(6):537–556, 2003.
- [62] Yingquan Zou, Fei Yan, Xiaomin Wang, and Jiyong Zhang. An efficient fuzzy logic control algorithm for photovoltaic maximum power point tracking under partial shading condition. *Journal of the Franklin Institute*, 357(6):3135–3149, 2020.
- [63] Thomas Meurer and Andreas Kugi. Tracking control for boundary controlled parabolic pdes with varying parameters: Combining backstepping and differential flatness. *Automatica*, 45(5):1182–1194, 2009.
- [64] Xiangyu Kong, Yuanqing Xia, Rui Hu, Min Lin, Zhongqi Sun, and Li Dai. Trajectory tracking control for under-actuated hovercraft using differential flatness and reinforcement learning-based active disturbance rejection control. *Journal of Systems Science and Complexity*, 35(2):502–521, 2022.
- [65] Christoph Kandler, Steven X Ding, Tim Koenings, Nick Weinhold, and Matthias Schultalbers. A differential flatness based model predictive control approach. In *2012 IEEE International Conference on Control Applications*, pages 1411–1416. IEEE, 2012.

- [66] Melissa Greeff and Angela P Schoellig. Flatness-based model predictive control for quadrotor trajectory tracking. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6740–6745. IEEE, 2018.
- [67] Hebertt Sira-Ramírez and Hebertt Sira-Ramírez. Differential flatness and sliding mode control. *Sliding Mode Control: The Delta-Sigma Modulation Approach*, pages 211–252, 2015.
- [68] Qinglong Wang, Jie Sun, Yunjian Hu, Wenqiang Jiang, Xinchun Zhang, and Zhangqi Wang. Deep learning-based flatness prediction via multivariate industrial data for steel strip during tandem cold rolling. *Expert Systems with Applications*, 237:121777, 2024.
- [69] Robin Ferede, Guido de Croon, Christophe De Wagter, and Dario Izzo. End-to-end neural network based optimal quadcopter control. *Robotics and Autonomous Systems*, 172:104588, 2024.
- [70] Michel Fliess and Hebertt Sira-Ramírez. An algebraic framework for linear identification. *ESAIM: Control, Optimisation and Calculus of Variations*, 9:151–168, 2003.
- [71] Mamadou Mboup, Cédric Join, and Michel Fliess. A revised look at numerical differentiation with an application to nonlinear feedback control. In *2007 Mediterranean Conference on Control & Automation*, pages 1–6. IEEE, 2007.
- [72] Mamadou Mboup, Cédric Join, and Michel Fliess. Numerical differentiation with annihilators in noisy environment. *Numerical algorithms*, 50:439–467, 2009.
- [73] Josef Zehetner, Johann Reger, and Martin Horn. A derivative estimation toolbox based on algebraic methods-theory and practice. In *2007 IEEE International Conference on Control Applications*, pages 331–336. IEEE, 2007.
- [74] Marek Kuczma. An introduction to the theory of functional equations and inequalities. (*No Title*), 2009.
- [75] Michel Fliess and Cédric Join. Model-free control. *International Journal of Control*, 86(12):2228–2252, 2013.
- [76] Brigitte d’Andréa Novel, Michel Fliess, Cédric Join, Hugues Mounier, and Bruno Steux. A mathematical explanation via “intelligent” pid controllers of the strange ubiquity of pids. In *18th Mediterranean Conference on Control and Automation, MED’10*, pages 395–400, 2010.

- [77] Michel Fliess, Cédric Join, and Hebertt Sira-Ramirez. Non-linear estimation is easy. *International Journal of Modelling, Identification and Control*, 4(1):12–27, 2008.
- [78] Michel Fliess and Hebertt Sira-Ramirez. An algebraic framework for linear identification. *ESAIM: Control, Optimisation and Calculus of Variations*, 9:151–168, 2003.
- [79] Sergey Edward Lyshevski. Nonlinear control of mechatronic systems with permanent-magnet DC motors. *Mechatronics*, 9(5):539–552, August 1999.
- [80] Y Yavin. Control of the motion of a disk rolling on a plane curve. *Computers & Mathematics with Applications*, 54(11-12):1329–1340, 2007.
- [81] Mahdi Mahfouf, CH Kee, Maysam F Abbod, and Derek A Linkens. Fuzzy logic-based anti-sway control design for overhead cranes. *Neural Computing & Applications*, 9(1):38–43, 2000.
- [82] Veera Chary Mummadi. Steady-state and dynamic performance analysis of pv supplied dc motors fed from intermediate power converter. *Solar Energy Materials and Solar Cells*, 61(4):365–381, 2000.
- [83] Chin Pei Tang, Patrick T Miller, Venkat N Krovi, Ji-Chul Ryu, and Sunil K Agrawal. Differential-flatness-based planning and control of a wheeled mobile manipulator—theory and experiment. *IEEE/ASME Transactions on Mechatronics*, 16(4):768–773, 2010.
- [84] F Wang, X Zhao, D Zhang, Z Ma, and X Jing. Robust and precision control for a directly-driven xy table. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 225(5):1107–1120, 2011.
- [85] Ilyas Eker. Open-loop and closed-loop experimental on-line identification of a three-mass electromechanical system. *Mechatronics*, 14(5):549–565, 2004.
- [86] Ilyas Eker. Sliding mode control with pid sliding surface and experimental application to an electromechanical plant. *ISA transactions*, 45(1):109–118, 2006.
- [87] G Mamani, J Bectedas, V Feliu-Batlle, and H Sira-Ramirez. Open-loop algebraic identification method for a dc motor. In *2007 European Control Conference (ECC)*, pages 3430–3436. IEEE, 2007.

- [88] G Mamani, J Becedas, V Feliu-Batlle, and H Sira-Ramírez. Open-and closed-loop algebraic identification method for adaptive control of dc motors. *International Journal of Adaptive Control and Signal Processing*, 23(12):1097–1103, 2009.
- [89] Bharat Bhushan and Madhusudan Singh. Adaptive control of dc motor using bacterial foraging algorithm. *Applied Soft Computing*, 11(8):4913–4920, 2011.
- [90] Mohamed Guermouche, Sofiane Ahmed Ali, and Nicolas Langlois. Super-twisting algorithm for dc motor position control via disturbance observer. *IFAC-PapersOnLine*, 48(30):43–48, 2015.
- [91] M. A. Aravind, Niranjana Saikumar, and N. S. Dinesh. Optimal position control of a dc motor using lqg with ekf. In *2017 International Conference on Mechanical, System and Control Engineering (ICMSC)*, pages 149–154, 2017.
- [92] Henrik Olsson, Karl Johan Åström, Carlos Canudas De Wit, Magnus Gäfvert, and Pablo Lischinsky. Friction models and friction compensation. *Eur. J. Control*, 4(3):176–195, 1998.