Design and Simulation of a Model Reference Adaptive Control System Employing Reproducing Kernel Hilbert Space for Enhanced Flight Control of a Quadcopter

Brian P. Scurlock

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
in
Mechanical Engineering

Andrew J. Kurdila, Chair
Andrea L'Afflitto, Co-Chair
Dylan P. Losey

May 8th, 2024
Blacksburg, Virginia

Keywords: Model reference adaptive control, Reproducing kernel Hilbert space, Unmanned aerial vehicles
Copyright 2024, Brian P. Scurlock

Design and Simulation of a Model Reference Adaptive Control System Employing Reproducing Kernel Hilbert Space for Enhanced Flight Control of a Quadcopter

Brian P. Scurlock

Abstract

This thesis presents the integration of reproducing kernel Hilbert spaces (RKHSs) into the model reference adaptive control (MRAC) framework to enhance the control systems of quadcopters. Traditional MRAC systems, while robust under predictable conditions, can struggle with the dynamic uncertainties typical in unmanned aerial vehicle (UAV) operations such as wind gusts and payload variations. By incorporating RKHS, we introduce a non-parametric, data-driven approach that significantly enhances system adaptability to in-flight dynamics changes.

The research focuses on the design, simulation, and analysis of an RKHS-enhanced MRAC system applied to quadcopters. Through theoretical developments and simulation results, the thesis demonstrates how RKHS can be used to improve the precision, adaptability, and error handling of MRAC systems, especially in managing the complexities of UAV flight dynamics under various disturbances. The simulations validate the improved performance of the RKHS-MRAC system compared to traditional MRAC, showing finer control over trajectory tracking and adaptive gains.

Further contributions of this work include the exploration of the computational impact and the relationship between the configuration of basis centers and system performance. Detailed analysis reveals that the number and distribution of basis centers critically influence the system's computational efficiency and adaptive capability, demonstrating a significant trade-off between efficiency and performance.

The thesis concludes with potential future research directions, emphasizing the need for further tests and implementations in real-world scenarios to explore the full potential of RKHS in adaptive UAV control, especially in critical applications requiring high precision and reliability. This work lays the groundwork for future explorations into scalable RKHS applications in MRAC systems, aiming to optimize computational resources while maximizing control system performance.

Design and Simulation of a Model Reference Adaptive Control System Employing Reproducing Kernel Hilbert Space for Enhanced Flight Control of a Quadcopter

Brian P. Scurlock

General Audience Abstract

This thesis develops and tests an advanced flight control system for quadcopters, using a technique referred to as reproducing kernel Hilbert space (RKHS) embedded model reference adaptive control (MRAC). Traditional control systems perform well in stable conditions but often falter with environmental challenges such as wind gusts or changes in weight. By integrating RKHS into MRAC, this new controller adapts in real-time, instantly adjusting the drone's operations based on its performance and environmental interactions.

The focus of this research is on the creation, testing, and analysis of this enhanced control system. Results from simulations show that incorporating RKHS into standard MRAC significantly boosts precision, adaptability, and error management, particularly under the complex flight dynamics faced by unmanned aerial vehicles (UAVs) in varied environments. These tests confirm that the RKHS-MRAC system performs better than traditional approaches, especially in maintaining accurate flight paths.

Additionally, this work examines the computational costs and the impact of various RKHS configurations on system performance. The thesis concludes by outlining future research opportunities, stressing the importance of real-world tests to verify the ability of RKHS-embedded MRAC in critical real-world applications where high precision and reliability are essential.

Contents

Chapter 1

Introduction

The integration of Model Reference Adaptive Control (MRAC) within quadcopter control systems represents a significant advancement in the management of dynamic disturbances and operational uncertainties inherent in unmanned aerial vehicle (UAV) navigation. This thesis explores the application of Reproducing Kernel Hilbert Spaces (RKHS) embedded-MRAC systems to enhance the adaptive capabilities of quadcopters facing a variety of environmental conditions and modeling uncertainties.

Traditional control systems, while effective under predictable conditions, often falter when subjected to the unpredictable elements associated with UAV operations, such as wind gusts, air density variations, or mechanical impairments. MRAC systems have proven to guarantee excellent flight performance in the presence of parametric, matched uncertainties and satisfactory performance in the presence of unmatched uncertainties [1, 2]. Indeed, in the presence of parametric matched uncertainties, existing MRAC systems guarantee asymptotic convergence of the trajectory tracking error to zero. Furthermore, in the presence of unmatched uncertainties, existing robust MRAC systems guarantee uniform ultimate boundedness of the trajectory tracking error [3, Ch. 9 and 11].

In this thesis, for the first time, we employ RKHS-embedded MRAC systems to design control systems for quadcopter UAVs. A key advantage of this approach is that it allows compensating for uncertainties and disturbances without relying on any parameterization, as prescribed in classical MRAC by the use of a regressor vector. For these reasons, the adaptability introduced through RKHS within MRAC frameworks promises to address these challenges by leveraging a non-parametric, data-driven approach to control. This

method not only manages functional uncertainties more robustly but also enhances the system's overall responsiveness to in-flight dynamics alterations.

This research advances the state-of-the-art in control systems design for UAVs, particularly in applications requiring precise control under dynamic conditions, a frequent challenge in UAV operations. Enhanced MRAC systems, equipped with RKHS-empowered methodologies, significantly improve the handling of UAVs, especially important in scenarios where physics-based modeling or other forms of system identification, such as liquid payload delivery, is still an open research problem. This capability not only stabilizes the UAV but also prevents spillage during critical tasks such as firefighting or agricultural spraying, where precise deployment of payloads is essential. Furthermore, the adaptive nature of these systems is invaluable for maintaining flight stability and safety, particularly when UAVs may experience structural damage or unexpected payload shifts. Such adaptability is essential across various applications, including emergency response and routine aerial surveillance, highlighting the broad applicability and critical importance of this research in environments where UAV integrity may be compromised by environmental hazards or operational wear and tear.

The objective of this thesis is to demonstrate the capability of RKHS-embedded MRAC in improving the adaptability and robustness of classical MRAC systems, thus enhancing the performance and safety of quadcopters in complex environments. This study meticulously examines how the sophisticated mathematical framework of RKHS can be effectively integrated with MRAC to provide a robust solution to the inherent variability and unpredictability of UAV operations.

This thesis is structured into six chapters, each designed to develop a comprehensive understanding and practical assessment of the integrated RKHS-MRAC framework. Following this introduction, Chapter 2 introduces the dynamics and control framework for quadcopters, laying the foundation for subsequent discussions on control strategies. Chapter 3 introduces the theory of MRAC and applies it to the quadcopter control strategy and dynamics outlined in Chapter 2, establishing a comprehensive understanding of MRAC in the context of UAVs. Chapter 4 delves into the theory of RKHS, explaining its significance and utility in enhancing MRAC systems for better performance and

adaptability. The practical application of RKHS to the quadcopter MRAC systems is thoroughly discussed, linking theoretical aspects to real-world applicability. Chapter 5 presents simulation results, demonstrating the effectiveness of the integrated RKHS-MRAC approach through various tests and comparisons with conventional control systems. The final chapter, Chapter 6, summarizes the study and proposes future research directions, highlighting ongoing challenges and the potential for further enhancements of MRAC systems using RKHS.

Through this structured exploration, this thesis aims to contribute significantly to the field of adaptive control in UAVs, providing a robust theoretical model and practical implementations that showcase the potential of RKHS in complex adaptive systems.

Chapter 2

Control Systems for UAVs

2.1 UAV Control Method Selection and Challenges

Controlling unmanned aerial vehicles (UAVs) poses unique challenges due to their sensitivity to parametric uncertainties, which manifest themselves in challenging scenarios such as payload delivery. The dynamic nature of their flight environment means that UAVs must contend with variable wind gusts, thermal currents, and other atmospheric disturbances, as well as changes in system dynamics due to varying payloads or fuel consumption. Due to these uncertainties and variability, model reference adaptive control (MRAC) presents itself as an optimal control strategy due to its adaptive capabilities, as discussed in [4, 5]. MRAC allows for real time adjustments to the control laws, compensating for uncertainties and ensuring that the UAV maintains its desired trajectory and performance. The adaptability is crucial in managing the external disturbances and internal variations, which are typical for UAV operations.

The use of MRAC is especially well suited for UAVs and aircraft control because it systematically addresses the matched uncertainties that affect the system's behavior through the control input pathway. Furthermore, the control laws within MRAC are designed to track a reference model that represents the ideal system behavior, allowing the actual system to perform closely to this ideal even in the face of unpredictable changes. This results in a robust and high-performance control system that is capable of handling the complex aerodynamics and non-linear dynamics of UAVs. Additionally, the robust nature of some variations of the classical MRAC architecture is beneficial for UAVs, which

are often used in applications that require reliability and precision, even when faced with unexpected circumstances. This robustness ensures that the UAV can maintain operational safety and performance even in the event of unanticipated operational conditions. Moreover, The MRAC framework based on the use of reproducing kernel Hilbert spaces (RKHSs) discussed in Chapter 3 has the potential to further enhance the performance of classical MRAC systems for its ability to account for matched uncertainties that are not parameterized by the user-defined regressor vector.

2.2 Equations of Motions of Multi-Rotor UAVs

In this section, proceeding as in [6, 7] we derive and examine the equations of motion governing multi-rotor UAVs, with a particular focus on quadcopters. This exploration is crucial for the development of advanced control systems aimed at optimizing flight stability, maneuverability, and responsiveness to control inputs. Specifically, we show the application in the case of constant inertial properties with a known center of mass as seen in [8, 9, 10].

2.2.1 Notation

Establishing a consistent notation system and identifying reference frames is essential for formulating the dynamics of UAVs. Let $\mathbb{I}$ denote an orthonormal, inertial, North-East-Down reference frame, and let $\mathbb{J}$ denote an orthonormal reference frame fixed with the UAV and centered in its center of mass. The mass of the UAV, including any payload, be denoted by $m \in \mathbb{R}$, and the acceleration due to gravity by $g \in \mathbb{R}$. The vectors $F \in \mathbb{R}^3$ and $M \in \mathbb{R}^3$ represent the external forces and moments acting on the UAV, respectively. The position $r : [t_0, \infty) \to \mathbb{R}^3$ and velocity $v : [t_0, \infty) \to \mathbb{R}^3$ of the UAV's center of mass are expressed in the inertial frame, while $\omega : [t_0, \infty) \to \mathbb{R}^3$ represents the angular velocity of $\mathbb{J}$ with respect to $\mathbb{I}$. The Euler angles $\phi : [t_0, \infty) \to [0, 2\pi)$, $\theta : [t_0, \infty) \to [0, 2\pi)$, and $\psi : [t_0, \infty) \to [0, 2\pi)$ describe the UAV's orientation in roll, pitch, and yaw, and the principal axes' moments of inertia are given by $I_{xx}, I_{yy}, I_{zz} \in \mathbb{R}$.

The cross-product operation is defined for a vector $a = [a_1, a_2, a_3]^{\mathrm{T}} \in \mathbb{R}^3$ as

$$\mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}, \quad \mathbf{a}\times \triangleq \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}. \tag{2.1}$$

This mathematical operator is essential in characterizing the effects of rotational forces on the UAV.

2.2.2 Kinematics of UAVs

Understanding the kinematics of UAVs involves a comprehensive analysis of their motion in the three-dimensional space. This analysis encompasses the UAV's translational and rotational positions and velocities.

Employing a 3-2-1 rotation sequence, the orientation of $\mathbb{J}$ relative to $\mathbb{I}$ is captured by

$$R(\phi, \theta, \psi) \equiv \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix}, \tag{2.2}$$

The translational kinematic equation for the center of mass of a UAV is defined by

$$\dot{r}^{\mathbb{I}}(t) = R^{\mathrm{T}}(\phi(t), \theta(t), \psi(t))v(t), \qquad r(t_0) = r_0, \qquad t \geq t_0. \tag{2.3}$$

which relates the translational velocity expressed in the reference frame $\mathbb{J}$, to the time derivative of the translational position expressed in the reference frame $\mathbb{I}$.

The rotational equations of motion of a UAV relating the angular velocity of $\mathbb{J}$ with respect to $\mathbb{I}$ [11, Def. 1.9], is given by

$$\begin{bmatrix} \dot{\phi}(t) \\ \dot{\theta}(t) \\ \dot{\psi}(t) \end{bmatrix} = \Gamma_J(\phi(t), \theta(t))\omega(t), \quad \begin{bmatrix} \phi(t_0) \\ \theta(t_0) \\ \psi(t_0) \end{bmatrix} = \begin{bmatrix} \phi_0 \\ \theta_0 \\ \psi_0 \end{bmatrix}, \quad t \geq t_0, \tag{2.4}$$

where the inverse Jacobian matrix is given by

$$\Gamma_J(\phi,\theta) \triangleq \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi\sec\theta & \cos\phi\sec\theta \end{bmatrix}, \quad (\phi,\theta) \in [0,2\pi) \times \left(-\frac{\pi}{2},\frac{\pi}{2}\right). \tag{2.5}$$

While Euler angles offer simplicity, they can also lead to singularities. These singularities occur when the pitch angle θ approaches $\pm\frac{\pi}{2}$, causing a loss of one degree of freedom in the rotation representation.

2.2.3 Dynamic Models

Dynamical models are crucial for understanding and controlling UAVs. They mathematically represent the UAV's motion, taking into account the forces and moments acting upon the vehicle. We distinguish between translational dynamics, describing the UAV's movement through space, and rotational dynamics, outlining changes in the UAV's orientation.

2.2.3.1 Translational Dynamics

The translational dynamics describe the movement of the UAV's center of mass in the inertial reference frame. These dynamics are governed by

$$m\dot{v}^{\mathbb{I}}(t) = -R(\eta(t))\left(u_1(t)e_3 + \frac{1}{2}\rho_{\text{air}}Sc_D\|\dot{r}(t)\|\dot{r}(t)\right) + mge_3, \qquad v^{\text{I}}(t_0) = v_0^{\text{I}}, \tag{2.6}$$

where the control input $u_1 : [t_0,\infty) \to \mathbb{R}$ represents the thrust force applied to the UAV. The vector $e_3 = [0,0,1]^{\text{T}} \in \mathbb{R}^3$ denotes the unit vector pointing in the vertical direction. Parameters $\rho_{\text{air}} > 0$, $S > 0$, and $c_D \in \mathbb{R}$ represent the air density, surface area, and drag coefficient, respectively. The gravitational acceleration $g > 0$ is a scalar.

2.2.3.2 Rotational Dynamics

The rotational dynamics describe how the UAV's orientation changes in response to applied torques. These dynamics are represented by

$$I\dot{\omega}(t) = \tau(t) - \omega^{\times}(t)I\omega(t), \qquad \omega(t_0) = \omega_0, \tag{2.7}$$

where the inertia matrix is denoted by $I \in \mathbb{R}^{3\times3}$ and the torque generated by the UAV is represented by the vector $\tau : [t_0, \infty) \to \mathbb{R}^3$.

2.3 Control System's Framework Overview

In this section, we introduce the foundational dynamic models for our control system and examine both translational and rotational dynamics. The dynamical model will later be reformatted to fit the MRAC problem.

2.3.1 Outer Loop Design

The purpose of the outer loop is to compute the desired roll ϕ_d and pitch θ_d angles, as well as the total thrust u_1 necessary to guide the UAV along the user defined trajectory. The user-defined yaw angle ψ_{user} is essential for these outer loop calculations.

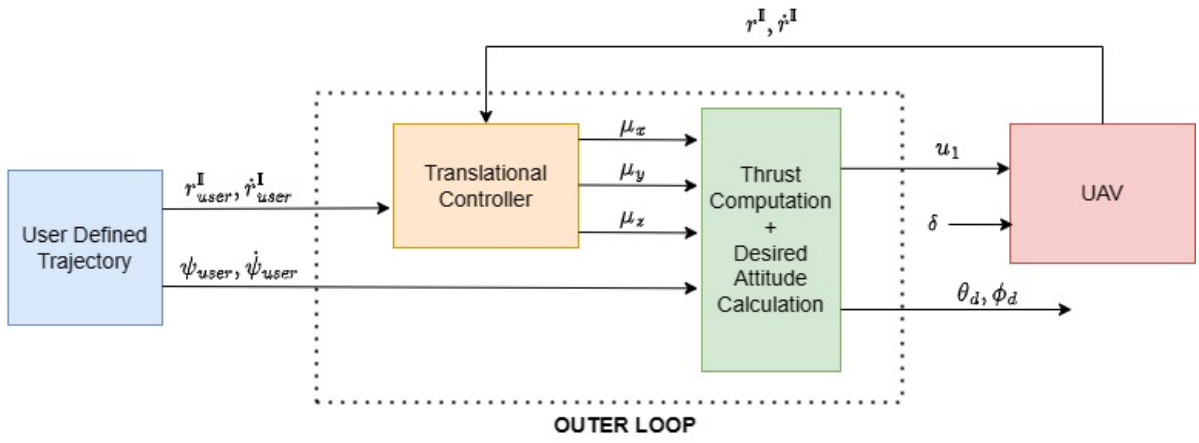

Figure 2.1: Outer loop control system framework

To design the outer loop, we first introduce the virtual translational control input defined by

$$\mu^{\mathbb{I}}(t) = -u_1(t)R(\eta_{\mathrm{d}}(t))e_3 + mge_3, \quad t \geq t_0. \tag{2.8}$$

where $\eta_d(t) \triangleq [\phi_d, \theta_d, \psi_{\text{user}}]^{\mathrm{T}}$ denotes the desired attitude.

From Equation 2.8, it follows that the virtual translational control input in reference frame $\mathbb{I}$ can be expressed component-wise as

$$\mu_x(t) = -u_1(t)\cos\phi(t)\sin\theta(t)\cos\psi_{\text{user}}(t) + \sin\phi(t)\sin\psi_{\text{user}}(t), \quad t \geq t_0, \tag{2.9a}$$

$$\mu_y(t) = -u_1(t)\cos\phi(t)\sin\theta(t)\sin\psi_{\text{user}}(t) - \sin\phi(t)\cos\psi_{\text{user}}(t), \tag{2.9b}$$

$$\mu_z(t) = -u_1(t)\cos\phi(t)\cos\theta(t) + mg. \tag{2.9c}$$

As seen in [7], assuming that $mg - \mu_z(t) > 0$ and $u_1(t) > 0$, implying that no trajectory requires free fall, we can deduce that

$$u_1(t) = \frac{mg - \mu_z(t)}{\cos\phi(t)\cos\theta(t)}, \tag{2.10a}$$

$$\theta_d(t) = \tan^{-1}\left(\frac{-\mu_x(t)\cos\psi_{\text{user}}(t) + \mu_y(t)\sin\psi_{\text{user}}(t)}{mg - \mu_z(t)}\right), \tag{2.10b}$$

$$\phi_d(t) = \sin^{-1}\left(\frac{-\mu_x(t)\sin\psi_{\text{user}}(t) - \mu_y(t)\cos\psi_{\text{user}}(t)}{u_1(t)}\right). \tag{2.10c}$$

The function atan2(y, x) is essential in determining the correct quadrant for the computed angles, ensuring accurate directionality in the UAV's control system. The function of atan2 is defined by

$$\text{atan2}(y,x) \triangleq \begin{cases} \arctan\left(\frac{y}{x}\right), & \text{if } x > 0, \\ \arctan\left(\frac{y}{x}\right) + \pi, & \text{if } x < 0 \text{ and } y \geq 0, \\ \arctan\left(\frac{y}{x}\right) - \pi, & \text{if } x < 0 \text{ and } y < 0, \\ \frac{\pi}{2}, & \text{if } x = 0 \text{ and } y > 0, \\ -\frac{\pi}{2}, & \text{if } x = 0 \text{ and } y < 0, \\ \text{undefined}, & \text{if } x = 0 \text{ and } y = 0, \end{cases} \tag{2.11}$$

The atan2 function addresses the ambiguity of the angle when the ratio of y/x is computed using the standard arctan function, which does not account for the signs of y and x individually and therefore can only determine the angle within the interval $(-\frac{\pi}{2}, \frac{\pi}{2})$.

The following set of equations uses the atan2 function for correctly calculating the desired

pitch and roll angles.

$$u_1(t) = \sqrt{\mu_x^2(t) + \mu_y^2(t) + (mg - \mu_z(t))^2}, \tag{2.12a}$$

$$\theta_d(t) = \operatorname{atan2}\left(-\mu_x(t)\cos\psi_{\text{user}}(t) - \mu_y(t)\sin\psi_{\text{user}}(t), mg - \mu_z(t)\right), \tag{2.12b}$$

$$\phi_d(t) = \operatorname{atan2}\left(\frac{-\mu_x(t)\sin\psi_{\text{user}}(t) + \mu_y(t)\cos\psi_{\text{user}}(t)}{u_1(t)}\right), \sqrt{1 - \left(-\frac{\mu_x(t)\sin\psi_{\text{user}}(t) + \mu_y(t)\cos\psi_{\text{user}}(t)}{u_1(t)}\right)^2}. \tag{2.12c}$$

Equation 2.12 represent the final output of the outer loop. These computed values for desired thrust and attitude angles are then passed to the inner loop for execution, ensuring that the UAV's trajectory is maintained accurately and stable flight is achieved.

2.3.2 Inner Loop Design

In the architecture of UAV control systems, the inner loop is responsible for tracking the desired angles $\phi_d(t)$ and $\theta_d(t)$ calculated by the outer loop, and $\psi_{\text{user}}(t)$ given by the user. This tracking is critical for the UAV's ability to follow the intended translational flight path and react dynamically to navigational commands.

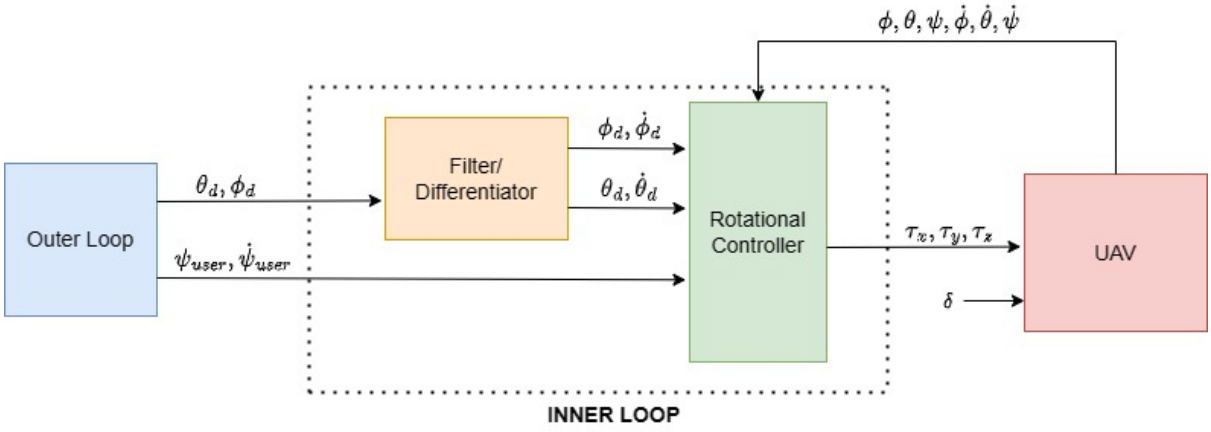

Figure 2.2: Inner loop control system framework

Within the inner loop, a key component is the implementation of second-order low pass filters. Filters not only suppresses undesirable high-frequency noise, enhancing control signal quality, but are also used to calculate the rate of change of the commanded angles, which are essential for the rotational control. The inner loop employs two distinct filters,

namely a second-order low-pass filter for smoothing the commanded angles, and a differentiator filter for determining their rates of change. The second-order low-pass filter is defined by the transfer function

$$H_{\text{filter}}(s) = \frac{\omega_0^2}{s^2 + 2\zeta_0\omega_0 s + \omega_0^2}, \quad s \in \mathbb{C} \setminus S_0. \tag{2.13}$$

In this function, $\omega_0 > 0$ and $\zeta_0 > 0$ are chosen based on the desired response of the filter, where ω_0 is the natural frequency and ζ_0 the damping ratio, and the set S_0 denotes the poles of $H_{\text{filter}}(\cdot)$. This low-pass filter is responsible for smoothing out the high-frequency components in the commanded angles, $\phi_d(t)$ and $\theta_d(t)$, ensuring that the signals fed into the differentiator are free of noise that could lead to erratic behavior.

Subsequent to smoothing, the differentiator filter calculates the derivatives of the desired angles which are required for dynamic control. The transfer function of the second order low pass differntiator is given by

$$G_{\text{diff}}(s) = sH_{\text{filter}}(s) = \frac{\omega^2 s}{s^2 + 2\zeta\omega s + \omega^2}, \quad s \in \mathbb{C} \setminus S_0. \tag{2.14}$$

The state-space representation of this differentiator filter is approximated by

$$G_{\text{diff}}(s) \sim \begin{bmatrix} A_{\text{diff}} & B_{\text{diff}} \\ C_{\text{diff}} & 0 \end{bmatrix} \tag{2.15}$$

where $A_{\text{diff}} \in \mathbb{R}^{2\times 2}$, $B_{\text{diff}} \in \mathbb{R}^2$, and $C_{\text{diff}} \in \mathbb{R}^{1\times 2}$ model the dynamics of the differentiator filter. The derivatives of the smoothed command angles can then be calculated with the state-space equations

$$\dot{x}_{\text{diff}}(t) = A_{\text{diff}}x_{\text{diff}}(t) + B_{\text{diff}}\theta_{\text{cmd}}(t), \quad x_{\text{diff}}(t_0) = 0_2, t \geq t_0, \tag{2.16}$$

$$y_{\text{diff}}(t) = C_{\text{diff}}x_{\text{diff}}(t). \tag{2.17}$$

Where a smoothed command angle is defined by $\theta_{\text{cmd}}(t)$. The system's initial state is set to zero to mirror the UAV's starting condition at rest. From there, the state $x_{\text{diff}}(t)$ evolves based on the dynamics dictated by A_{diff} and B_{diff}, with the output $y_{\text{diff}}(t)$ providing

the derivative rates of the commanded angles, crucial for actuator command generation.

2.4 Conclusions

This chapter has introduced the fundamental aspects of UAV dynamics, focusing on the kinematics and equations of motion that govern multi-rotor UAVs, with particular emphasis on quadcopters. By analyzing these models we have now established a robust foundation for developing advanced control systems tailored to enhance UAV flight stability, maneuverability, and responsiveness to control inputs. The detailed breakdown of notations and the presentation of dynamic equations have provided clear insights into the physical forces and operational parameters that influence UAV behavior. This clarity is crucial for accurately simulating UAV flight characteristics and for creating control strategies that can effectively counteract disturbances while optimizing flight performance. Furthermore, it is shown how the control system employees both an outer and inner control loop to overcome the UAV's underactuation.

Looking ahead, the next chapter will introduce the Model reference adaptive control framework and apply the foundational concepts discussed here to the specific context of MRAC design. This upcoming discussion will not only build on the established groundwork but also expand the scope of potential UAV applications through advanced control methodologies.

Chapter 3

Model Reference Adaptive Control

3.1 Matched Uncertainties of Control Systems

In model reference adaptive control (MRAC), the key concept lies in the adaptive system's ability to adjust its parameters to match a given reference model despite uncertainties or variations in system dynamics. A significant challenge in control theory is dealing with these uncertainties, which may arise due to modeling errors, parameter variations, or disturbances. These uncertainties are generally classified into two categories: matched and unmatched.

Matched uncertainties are those that enter the system dynamics through the same channel as the control input. They can be formally handled within the structure of the control input vector. Consider a generic control system described by the following differential equation

$$\dot{x}(t) = Ax(t) + Bu(t) + \Delta(x,t), \quad x(t_0) = x_0, \quad t \geq t_0, \tag{3.1}$$

where $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times m}$ define the system dynamics, and $\Delta : \mathbb{R}^n \times [t_0, \infty)$ represents the bounded uncertainties affecting the system. For $\Delta(\cdot,\cdot)$ to capture matched uncertainties, it must be reduced to the form

$$\Delta(x,t) = B\delta(x,t), \tag{3.2}$$

where $\delta : \mathbb{R}^n \times [t_0, \infty) \to \mathbb{R}^m$ is some unknown function. Then Equation 3.1 can be rewritten as

$$\dot{x}(t) = Ax(t) + B(u(t) + \delta(x,t)), \quad x(t_0) = x_0, \quad t \geq t_0, \tag{3.3}$$

which demonstrates how the matched uncertainty influences the system in a manner similar to the control input, through the matrix B. To counteract the impact of matched uncertainties, MRAC strategies design the control input, $u(t)$, by using feedback related to the system's state and its performance compared to the predefined reference model. This adaptation involves tuning the adaptive gains, which are parameters that influence the behavior of the control system. The aim of these adjustments is to make the state of the actual system track the state of the reference model, which represents the ideal behavior and performance we wish to achieve. This mechanism ensures that, despite uncertainties, the system behaves in a way that is as close as possible to our desired outcome.

3.2 Classic MRAC

We will now present the standard MRAC equations as seen in [3] beginning with a standard plant model described by the equation

$$\dot{x}(t) = Ax(t) + B\Lambda(u(t) + \Theta^{\mathrm{T}}\Phi(t, x(t)), \quad x(t_0) = x_0, \quad t \geq t_0, \tag{3.4}$$

where $x(t) \in \mathbb{R}^n$ denotes the state vector of the plant, $u(t) \in \mathbb{R}^m$ represents the control input, $A \in \mathbb{R}^{n\times n}$ is an unknown system matrix, and $B \in \mathbb{R}^{n\times m}$ is a known input matrix that, alongside the unknown positive-definite diagonal matrix $\Lambda \in \mathbb{R}^{m\times m}$, ensures that the system is controllable. The system dynamics depend on the unknown function $\Theta^{\mathrm{T}}\Phi(t, x(t))$ which denotes matched uncertainties with $\Theta \in \mathbb{R}^{N\times m}$ being unknown constant coefficients and $\Phi(t, x(t)) \in \mathbb{R}^N$ as the regressor vector composed of basis functions $\phi_i(t, x(t))$, $i = 1, \ldots, N$.

To capture matched uncertainties, we assume that they can be expressed as a linear combination of the basis functions

$$\Phi(t, x) = [\phi_1(t, x, \ldots, \phi_N(t, x]^{\mathrm{T}}, \qquad (t, x) \in [t_0, \infty) \times \mathbb{R}^n, \tag{3.5}$$

The unknown coefficients within the matrix Θ represent the magnitude of these uncertainties. For the existence and uniqueness of the system's solutions, the functions $\phi_i(t, \cdot)$, $i \in \{1, \ldots, N\}$, are assumed to be Lipschitz continuous in x.

The design objective of the adaptive control law is to enable the plant's state $x(\cdot)$ to globally, uniformly, and asymptotically track a reference trajectory $x_{\text{ref}}(t) \in \mathbb{R}^n$, which is generated from the reference model characterized by the equation

$$\dot{x}_{\text{ref}}(t) = A_{\text{ref}} x_{\text{ref}}(t) + B_{\text{ref}} r(t), \quad x_{\text{ref}}(t_0) = x_{\text{ref},0}, \quad t \geq t_0, \tag{3.6}$$

The matrix $A_{\text{ref}} \in \mathbb{R}^{n \times n}$, ensuring the stability of the reference model, while $B_{\text{ref}} \in \mathbb{R}^{n \times m}$ allows for controllability of the system when paired with A_{ref}. The reference input $r(t) \in \mathbb{R}^m$, steers the reference state $x_{\text{ref}}(t)$ along a trajectory that the actual plant state is expected to follow.

The ideal control input is designed to counteract uncertainties and align the plant's output with that of the reference model and is defined by the equation

$$u_{\text{ideal}}(t) = K_x^{\text{T}} x(t) + K_r^{\text{T}} r_{\text{cmd}}(t) - \Theta^T \Phi(t, x(t)), \quad t \geq t_0, \tag{3.7}$$

where $K_x \in \mathbb{R}^{n \times m}$ and $K_r \in \mathbb{R}^{m \times m}$. To ensure that the control inputs effectively emulate the desired dynamics, matching conditions are set to align the plant's dynamics with those of the reference model

$$A_{\text{ref}} = A + B\Lambda K_x^{\text{T}}, \quad B_{\text{ref}} = B\Lambda K_r^{\text{T}} \tag{3.8}$$

These conditions guarantee that the adaptive control law has the potential to match the behavior of the reference model, thereby affirming the controllability of the closed-loop system with respect to the reference input.

The implementation of the ideal control input $u_{\text{ideal}}(t)$ transforms the open-loop plant dynamics into a closed-loop system defined by the equation

$$\dot{x}(t) = (A + B\Lambda K_x^{\top}) x(t) + B\Lambda K_r^{\top} r_{\text{cmd}}(t), \quad x(t_0) = x_0, \quad t \geq t_0, \tag{3.9}$$

which describes how the state vector $x(t)$ evolves over time under the influence of the adaptive control. Here, the term $B\Lambda K_x^\top$ reflects the feedback mechanism's contribution to the system dynamics, and $B\Lambda K_r^\top r_{\text{cmd}}(t)$ represents the command-driven component that guides the system along the desired trajectory.

Similar to the ideal control input, the MRAC control law is defined by the equation

$$u(t) = \hat{K}_x^{\mathrm{T}}(t)x(t) + \hat{K}_r^{\mathrm{T}}(t)r_{\text{cmd}}(t) - \hat{\Theta}^{\mathrm{T}}(t)\Phi(t, x(t)), \quad t \geq t_0. \tag{3.10}$$

The feedforward gain $\hat{K}_x(t) \in \mathbb{R}^{n\times m}$ and the feedback gain $\hat{K}_r(t) \in \mathbb{R}^{m\times m}$ are initially chosen based on a best estimate and are then continuously updated using adaptive laws to obtain global uniform asymptotic tracking of the reference model trajectory. The term $-\hat{\Theta}^{\mathrm{T}}\Phi(t, x(t))$ represents an attempt to cancel the matched uncertainties and similarly, are adjusted using adaptive laws.

By substituting (3.10) into the system dynamics (3.4) and applying the matching conditions (3.8), the plant dynamics is equivalent to

$$\begin{aligned}\dot{x}(t) = A_{\text{ref}}x(t) + BK_r(t)r(t) + B(\hat{K}_x(t) - K_x(t))x(t) \\ + B(\hat{K}_r(t) - K_r(t))r(t) - B(\hat{\Theta}(t) - \Theta(t))^{\mathrm{T}}\Phi(x),\end{aligned} \tag{3.11}$$

where

$$\Delta K_x(t) \triangleq \hat{K}_x(t) - K_x(t), \quad \Delta K_r(t) \triangleq \hat{K}_r(t) - K_r(t), \quad \Delta\Theta(t) \triangleq \hat{\Theta}(t) - \Theta(t), \tag{3.12}$$

represent the error gain.

Discrepancies between the plant's actual path and the reference trajectory are quantified by the tracking error

$$e(t) \triangleq x(t) - x_{\text{ref}}(t), \quad t \geq t_0 \tag{3.13}$$

which serves as a critical performance metric for the adaptive control strategy. The dynamics of the systems tracking error can then be obtained by subtracting (3.6) from (3.11) resulting in the equation

$$\dot{e}(t) = A_{\text{ref}}e(t) + B\Lambda[\Delta K_x^{\mathrm{T}}x(t) + \Delta K_r^{\mathrm{T}}r_{\text{cmd}}(t) + \Delta\Theta^{\mathrm{T}}\Phi(t, x(t))]$$
$$e(t_0) = x_0 - x_{\text{ref},0}, \quad t \geq t_0. \tag{3.14}$$

To adjust the control gains $\hat{K_x}(\cdot)$ and $\hat{K_r}(\cdot)$ and the uncertainty parameters $\hat{\Theta}(\cdot)$, the following adaptive laws are employed

$$\dot{\hat{K}}_x(t) = -\Gamma_x x(t)e^{\mathrm{T}}(t)PB, \qquad \hat{K}_x(t_0) = \hat{K}_{x,0}, \quad t \geq t_0, \tag{3.15a}$$
$$\dot{\hat{K}}_r(t) = -\Gamma_r r(t)e^{\mathrm{T}}(t)PB, \qquad \hat{K}_r(t_0) = \hat{K}_{r,0}, \tag{3.15b}$$
$$\dot{\hat{\Theta}}(t) = \Gamma_\Theta \Phi(t, x(t))e^{\mathrm{T}}(t)PB, \qquad \hat{\Theta}(t_0) = \hat{\Theta}_0. \tag{3.15c}$$

The adaptive rates $\Gamma_x \in \mathbb{R}^{n\times n}$, $\Gamma_r \in \mathbb{R}^{m\times m}$, and $\Gamma_\Theta \in \mathbb{R}^{N\times N}$ are user defined, positive definite, symmetric, and chosen to ensure the convergence of the tracking error $e(\cdot)$ to zero.

The stability of the reference model in the model reference adaptive control framework is verified using the algebraic Lyapunov equation

$$PA_{\text{ref}} + A_{\text{ref}}^{\mathrm{T}}P = -Q, \tag{3.16}$$

where $P \in \mathbb{R}^{n\times n}$ is a symmetric positive-definite matrix, and $Q \in \mathbb{R}^{n\times n}$ is a user defined, symmetric, and positive-definite matrix.

3.3 Robust MRAC

3.3.1 Classical Dead-Zone Modification of MRAC

Model reference adaptive control systems are susceptible to parameter drift in the presence of sustained disturbances or unmodeled dynamics. This phenomenon can lead to undesirable control performance and instability. The classical dead-zone modification in MRAC enhances robustness by creating a region of insensitivity within the adaptation

mechanism. This insensitivity, characterized by a dead-zone, prevents the adaptive gains from responding to small errors or disturbances.

The dead-zone modification effectively stops the adaptation process when the trajectory tracking error is within a predefined threshold ϵ. This modification can be represented by the following adaptive laws

$$\dot{\hat{K}}_x(t) = \begin{cases} -\Gamma_x x(t) e^{\mathrm{T}}(t) PB, & \text{if } \|e(t)\| > \epsilon, \\ 0, & \text{if } \|e(t)\| \leq \epsilon \end{cases}, \qquad \hat{K}_x(t_0) = \hat{K}_{x,0}, \quad t \geq t_0, \tag{3.17a}$$

$$\dot{\hat{K}}_r(t) = \begin{cases} -\Gamma_r r(t) e^{\mathrm{T}}(t) PB, & \text{if } \|e(t)\| > \epsilon, \\ 0, & \text{if } \|e(t)\| \leq \epsilon \end{cases}, \qquad \hat{K}_r(t_0) = \hat{K}_{r,0}, \tag{3.17b}$$

$$\dot{\hat{\Theta}}(t) = \begin{cases} \Gamma_\Theta \Phi(t, x(t)) e^{\mathrm{T}}(t) PB, & \text{if } \|e(t)\| > \epsilon, \\ 0, & \text{if } \|e(t)\| \leq \epsilon \end{cases}, \qquad \hat{\Theta}(t_0) = \hat{\Theta}_0. \tag{3.17c}$$

If the norm of the error $\|e(t)\|$ exceeds ϵ for some $t \geq t_0$, the system perceives this as significant, and the adaptive gains $\hat{K}_x(t)$, $\hat{K}_r(t)$, and $\hat{\Theta}(t)$ are updated to minimize the error. Conversely, when the error is within this dead-zone $\|e(t)\| \leq \epsilon$ for some $t \geq t_0$, the adaptive gains remain static, signifying that the system's current state is sufficiently close to the desired trajectory, and no adjustment is required.

By implementing a dead-zone, the control strategy avoids unnecessary computational effort and prevents adaptive gain drift. The prevention of gain drift is directly linked to preventing numerical instabilities in the computation of the adaptive gains.

3.4 Implementing a Baseline Controller Within MRAC

MRAC systems are designed for precise trajectory tracking, even amidst uncertainties and disturbances, however, their sensitivity can lead to aggressive control behaviors. A baseline controller is introduced to provide stable performance under standard conditions. The adaptive control law is then focused on correcting any deviations arising from the

baseline controller's operation.

The combined control input, $u(t)$, is defined as

$$u(t) = u_{\text{bl}}(t) + u_{\text{MRAC}}(t), \quad t \geq t_0, \tag{3.18}$$

where $u_{\text{bl}}(t) : [t_0, \infty) \to \mathbb{R}^3$ represents the baseline control input, characterizing a predefined, bounded function and $u_{\text{MRAC}}(t) : [t_0, \infty) \to \mathbb{R}^3$ is the adaptive component, created using the MRAC framework. To properly fit the MRAC architecture, with the addition of a baseline, the system can be redefined by

$$\dot{x}(t) = Ax(t) + B\left(u_{\text{MRAC}}(t) + \left[1_m, \Theta^{\text{T}}\right]\begin{bmatrix} u_{\text{bl}}(t) \\ \Phi(t, x(t)) \end{bmatrix}\right), \quad x(t_0) = x_0, \quad t \geq t_0, \tag{3.19}$$

which is in the same form as Equation 3.4 replacing Θ^{T} with $[1_m, \Theta^{\text{T}}]$, and $\Phi(t, x(t))$ with $[u_{\text{bl}}^{\text{T}}(t), \Phi^{\text{T}}(t, x(t))]^{\text{T}}$.

A possible baseline controller is defined by

$$u_{\text{bl}}(t) = u_{\text{ideal}}(t) + \bar{K}_x^{\text{T}} x(t) + \bar{K}_r^{\text{T}} r_{\text{cmd}}(t) - \bar{\Theta}^{\text{T}} \Phi(t, x(t)), \quad t \geq t_0, \tag{3.20}$$

whrere $u_{\text{ideal}}(t) : [t_0, \infty) \to \mathbb{R}^3$ captures the PID control input, while $\bar{K}_x \in \mathbb{R}^{m \times m}$, $\bar{K}_r \in \mathbb{R}^{n \times m}$, and $\bar{\Theta} \in \mathbb{R}^{N \times m}$ are user-defined estimates of the unknown matrices K_x, K_r , and Θ, respectively.

The system dynamics, incorporating the baseline controller within the MRAC structure can then be redefined by the equation

$$\dot{x}(t) = Ax(t) + B\left[u_{\text{MRAC}}(t) + \left[1_m, \bar{K}_x^{\text{T}}, \bar{K}_r^{\text{T}}, (\Theta - \bar{\Theta})^{\text{T}}\right]\begin{bmatrix} u_{\text{ideal}}(t) \\ x(t) \\ r_{\text{cmd}}(t) \\ \Phi(t, x(t)) \end{bmatrix}\right], \tag{3.21}$$
$$x(t_0) = x_0, \quad t \geq t_0.$$

This choice of baseline controller will later be implemented.

3.5 Application of MRAC to the Quadcopter Controller Design

The equations of motion detailed in Section 2.3 will now be manipulated to allow for the application of the MRAC strategy.

3.5.1 Outer Loop System Design Through MRAC

3.5.1.1 MRAC Formulation of Translational Dynamics

First, we will modify the translational dynamics to align with the classical MRAC structure described by (3.4). Substituting the virtual translational control input, as defined in Section 2.3.1 by

$$\mu^{\mathbb{I}}(t) = -u_1(t)R(\eta_d(t))e_3 + mge_3, \quad t \geq t_0,$$

into the translational dynamics expressed in Section 2.2.3 and defined as

$$m\dot{v}^{\mathrm{I}}(t) = -R(\eta(t))\left(u_1(t)e_3 + \frac{1}{2}\rho_{\mathrm{air}}Sc_D\|v(t)\|v(t)\right) + mge_3, \quad v^{\mathbb{I}}(t_0) = v_0^{\mathbb{I}}, \quad t \geq t_0.$$

results in

$$\dot{v}^{\mathbb{I}}(t) = m_3^{-1}\left(\mu(t) - \frac{1}{2}\rho_{\mathrm{air}}Sc_DR(\eta(t))v(t)\|v(t)\|\right), \quad v^{\mathbb{I}}(t_0) = v_0^{\mathbb{I}}, \quad t \geq t_0. \tag{3.22}$$

Defining $\Theta = \frac{1}{2}\rho_{\mathrm{air}}Sc_D$ and $\Phi(t, v(t)) = -R(\eta(t))v(t)\|v(t)\|$, the translational dynamics take the form consistent with (3.4) and can now be described by the equation

$$\dot{v}^{\mathbb{I}}(t) = m_3^{-1}\left(\mu(t) + \Theta^T\Phi(t, v(t))\right), \quad v^{\mathbb{I}}(t_0) = v_0^{\mathbb{I}}, \quad t \geq t_0. \tag{3.23}$$

The equivalent system dynamics, conforming to the classical MRAC structure, and written in matrix form is expressed as

$$\underbrace{\begin{bmatrix} \dot{r}^{\mathbb{I}}(t) \\ \dot{v}^{\mathbb{I}}(t) \end{bmatrix}}_{\dot{x}(t)} = \underbrace{\begin{bmatrix} 0_{3\times3} & I_3 \\ 0_{3\times3} & 0_{3\times3} \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} r^{\mathbb{I}}(t) \\ v^{\mathbb{I}}(t) \end{bmatrix}}_{x(t)} + \underbrace{\begin{bmatrix} 0_{3\times3} \\ 1_3 \end{bmatrix}}_{B} \underbrace{\begin{bmatrix} m^{-1} & 0 & 0 \\ 0 & m^{-1} & 0 \\ 0 & 0 & m^{-1} \end{bmatrix}}_{\Lambda} \big(\mu(t) + \Theta^T \Phi(t, \mathbf{v}(t)),\big)$$

$$v^{\mathbb{I}}(t_0) \;=\; v_0^{\mathbb{I}}, \quad r^{\mathbb{I}}(t_0) \;=\; r_0^{\mathbb{I}}, \quad t \;\geq\; t_0. \quad (3.24)$$

3.5.1.2 Implementing a Baseline Controller – Outer Loop

The baseline controller input for the outer loop is defined as

$$\mu_{\text{ideal}}(t) = -\bar{m}\Big[K_{P,\text{tran}}\left(r^{\mathbb{I}}(t) - r^{\mathbb{I}}_{\text{ref}}(t)\right) + K_{D,\text{tran}}\left(\dot{v}^{\mathbb{I}}(t) - \dot{r}^{\mathbb{I}}_{\text{ref}}(t)\right) - \ddot{r}^{\mathbb{I}}_{\text{ref}}(t)\Big], \quad t \geq t_0, \tag{3.25}$$

where $K_{P,\text{tran}} \in \mathbb{R}^{3\times3}$ and $K_{D,\text{tran}} \in \mathbb{R}^{3\times3}$ denote the proportional and derivative gain matrices in $\mathbb{R}^{3\times3}$, respectively. The scalar $\bar{m}$ represents an estimate of the UAV's mass. The terms $r^{\mathbb{I}}(t)$ and $\dot{v}^{\mathbb{I}}(t)$ indicate the UAV's instantaneous position and velocity in the inertial frame, while $r^{\mathbb{I}}_{\text{ref}}(t)$ and $\dot{r}^{\mathbb{I}}_{\text{ref}}(t)$ correspond to the reference trajectory's position and velocity.

This baseline controler is designed to reduce the UAV's trajectory tracking error, thereby steering the system along its desired path efficiently. The proportional and derivative gains, $K_{P,\text{tran}}$ and $K_{D,\text{tran}}$, enable a dynamic response to positional and velocity deviations, crucial for maintaining the UAV's flight stability and precision.

3.5.1.3 Outer Loop Reference Model

The next step is to select a reference model in the same form as (3.6). The outer loop reference model is responsible for generating the desired position and velocity references, $r_{\text{ref}}(t)$ and $v_{\text{ref}}(t)t \geq t_0$, ensuring that they track the user defined trajectories, $r_{\text{user}}(t)$ and $v_{\text{user}}(t)$, respectively. To achieve this goal we let

$$\begin{bmatrix} \dot{r}^{\mathbb{I}}_{\text{ref}}(t) \\ \dot{v}^{\mathbb{I}}_{\text{ref}}(t) \end{bmatrix} = \begin{bmatrix} 0_{3\times3} & I_3 \\ 0_{3\times3} & 0_{3\times3} \end{bmatrix} \begin{bmatrix} r^{\mathbb{I}}_{\text{ref}}(t) \\ v^{\mathbb{I}}_{\text{ref}}(t) \end{bmatrix} + \begin{bmatrix} 0_{3\times3} \\ m^{-1}I_3 \end{bmatrix} p_{\text{cmd}}(t), \\ \begin{bmatrix} r^{\mathbb{I}}_{\text{ref}}(t_0) \\ v^{\mathbb{I}}_{\text{ref}}(t_0) \end{bmatrix} = \begin{bmatrix} r^{\mathbb{I}}_{\text{ref},0} \\ v^{\mathbb{I}}_{\text{ref},0} \end{bmatrix}, \quad t \geq t_0, \tag{3.26}$$

where $p_{\text{cmd}}(t)$ represents the command input to the reference model, and is defined by the control laws for proportional, derivative, and integral feedback components

$$p_{\text{cmd}}(t) = \tilde{m}\Bigg(- K_{P,\text{tran}}(r^{\mathbb{I}}_{\text{ref}}(t) - r^{\mathbb{I}}_{\text{user}}(t)) - K_{D,\text{tran}}(\dot{r}^{\mathbb{I}}_{\text{ref}}(t) - \dot{r}^{\mathbb{I}}_{\text{user}}(t)) \\ - K_{I,\text{tran}} \int_{t_0}^{t} (r^{\mathbb{I}}_{\text{ref}}(\tau) - r^{\mathbb{I}}_{\text{user}}(\tau))d\tau + \ddot{r}^{\mathbb{I}}_{\text{user}}(t) \Bigg). \tag{3.27}$$

Restructuring (3.26) through the modification of matrix A_{ref} results in

$$\underbrace{\begin{bmatrix} \dot{r}^{\mathbb{I}}_{\text{ref}}(t) \\ \dot{v}^{\mathbb{I}}_{\text{ref}}(t) \end{bmatrix}}_{\dot{x}_{\text{ref}}} = \underbrace{\begin{bmatrix} 0_{3\times3} & I_3 \\ -K_{P,\text{tran}} & -K_{D,\text{tran}} \end{bmatrix}}_{A_{\text{ref}}} \underbrace{\begin{bmatrix} r^{\mathbb{I}}_{\text{ref}}(t) \\ v^{\mathbb{I}}_{\text{ref}}(t) \end{bmatrix}}_{x_{\text{ref}}} + \underbrace{\begin{bmatrix} 0_{3\times3} \\ m^{-1}I_3 \end{bmatrix}}_{B_{\text{ref}}} r_{\text{cmd}}(t), \\ \begin{bmatrix} r^{\mathbb{I}}_{\text{ref}}(t_0) \\ v^{\mathbb{I}}_{\text{ref}}(t_0) \end{bmatrix} = \begin{bmatrix} r^{\mathbb{I}}_{\text{ref},0} \\ v^{\mathbb{I}}_{\text{ref},0} \end{bmatrix}, \quad t \geq t_0, \tag{3.28}$$

where

$$r_{\text{cmd}}(t) = \tilde{m}\Bigg(K_{P,\text{tran}} r^{\mathbb{I}}_{\text{user}}(t) + K_{D,\text{tran}} \dot{r}^{\mathbb{I}}_{\text{user}}(t) \\ - K_{I,\text{tran}} \int_{t_0}^{t} \left(r^{\mathbb{I}}_{\text{ref}}(\tau) - r^{\mathbb{I}}_{\text{user}}(\tau) \right) d\tau + \ddot{r}^{\mathbb{I}}_{\text{user}}(t) \Bigg). \tag{3.29}$$

Equations (3.28) and (3.29) ensure with proper tuning of gains, that the reference model will track the user defined trajectories $r_{\text{user}}(t)$ and $v_{\text{user}}(t)$.

3.5.2 Inner Loop System Design Through MRAC

3.5.2.1 MRAC Formulation of Rotational Dynamics

We will now restructure the rotational dynamics, as previously done with the translational dynamics in Section 3.5.1. Unlike the translational dynamics, however, the relationship between the kinematics and dynamics equations is not described by a simple integrator since the attitude vector $\eta(t)$ is not defined by the integral of angular velocity $\omega(t)$. First, the rotational dynamics as defined by (2.7), can be rewritten in the form

$$\dot{\omega}(t) = I^{-1}[\tau(t) - \omega^{\times}(t)I\omega(t)], \quad \omega(t_0) = \omega_0, \quad t \geq t_0. \tag{3.30}$$

Assuming that the inertia matrix I is diagonal, the term $\omega^{\times}(t)I\omega(t)$ in (3.30), where $\omega = [\omega_x, \omega_y, \omega_z]^{\mathrm{T}}$, can be expanded as follows

$$\omega^{\times} I\omega = \begin{bmatrix} (I_z - I_y) & 0 & 0 \\ 0 & (I_x - I_z) & 0 \\ 0 & 0 & (I_y - I_x) \end{bmatrix} \begin{bmatrix} \omega_z\omega_y \\ \omega_z\omega_x \\ \omega_y\omega_x \end{bmatrix}.$$

This equation captures the gyroscopic effect, where the inertia differences along each axis contribute to the angular acceleration.

The UAV's rotational dynamics can then be restructured into the MRAC form as

$$\underbrace{\begin{bmatrix} \dot{\omega}_x(t) \\ \dot{\omega}_y(t) \\ \dot{\omega}_z(t) \end{bmatrix}}_{\dot{\boldsymbol{x}}(t)} = \underbrace{\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} \omega_x(t) \\ \omega_y(t) \\ \omega_z(t) \end{bmatrix}}_{\boldsymbol{x}(t)} +$$

$$\underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{B} \underbrace{\begin{bmatrix} \frac{1}{I_x} & 0 & 0 \\ 0 & \frac{1}{I_y} & 0 \\ 0 & 0 & \frac{1}{I_z} \end{bmatrix}}_{\Lambda} \left[\tau(t) - \underbrace{\begin{bmatrix} I_y - I_z & 0 & 0 \\ 0 & I_z - I_x & 0 \\ 0 & 0 & I_x - I_y \end{bmatrix}}_{\Theta^{\mathrm{T}}} \underbrace{\begin{bmatrix} \omega_y(t)\omega_z(t) \\ \omega_x(t)\omega_z(t) \\ \omega_x(t)\omega_y(t) \end{bmatrix}}_{\Phi(t,\boldsymbol{x}(t))} \right], \tag{3.31}$$

In this model, A is zero since the system is inherently non-linear and matrix B is an

identity matrix, denoting direct influence of control inputs. The matrices Θ, Φ, and Λ together define the entirety of the rotational dynamics

3.5.2.2 Implementing a Baseline Controller – Inner Loop

The baseline controller input for the inner loop is defined as

$$\begin{aligned}\tau_{\text{ideal}}(t) = -\bar{I}\Big[K_{P,\omega_{\text{ref}}}(\omega(t) - \omega_{\text{ref}}(t)) + K_{D,\omega_{\text{ref}}}(\dot{\omega}(t) - \dot{\omega}_{\text{ref}}(t))\Big] \\ + K_{I,\omega_{\text{ref}}}\int_{t_0}^{t}(\omega(\tau) - \omega_{\text{ref}}(\tau))\mathrm{d}\tau + \omega^{\times}(t)\bar{I}\omega(t), \quad t \geq t_0,\end{aligned} \tag{3.32}$$

where $K_{P,\omega_{\text{ref}}}$, $K_{I,\omega_{\text{ref}}}$ and $K_{D,\omega_{\text{ref}}}$ represent the proportional, integral, and derivative gain matrices, respectively, all positive-definite in $\mathbb{R}^{3\times3}$. The scalar $\bar{I}$ denotes the estimated inertia matrix of the UAV.

3.5.2.3 Inner Loop Reference Model

Similar to the outer loop, the inner loop reference model guides the trajectory of $\omega_{\text{ref}}(t)$, to ensure proper tracking and alignment of the command signal, $\omega_{\text{cmd}}(t)$. The accurate tracking of the inner loop is essential as the dynamics of the outer loop directly depend on the accuracy of the rotational position.

We must now generate the rotational reference model in the same form as Equation 3.6. To achieve this goal we first let

$$\dot{\omega}_{\text{ref}}(t) = -K_{P,\omega_{\text{ref}}}(\omega_{\text{ref}}(t) - \omega_{\text{cmd}}(t)) + \dot{\omega}_{\text{cmd}}(t), \quad \omega_{\text{ref}}(t_0) = \omega_{\text{ref},0}, \quad t \geq t_0, \tag{3.33}$$

which can be rewritten in the form

$$\dot{\omega}_{\text{ref}}(t) = \underbrace{-K_{P,\omega_{\text{ref}}}}_{A_{\text{ref}}}\underbrace{\omega_{\text{ref}}(t)}_{x_{\text{ref}}(t)} + \underbrace{1_3}_{B_{\text{ref}}} r_{\text{cmd}}, \quad \omega_{\text{ref}}(t_0) = \omega_{\text{ref},0}, \quad t \geq t_0, \tag{3.34}$$

where the command signal r_{cmd} is defined by the equation

$$r_{\text{cmd}}(t) = K_{P,\omega_{\text{ref}}}\omega_{\text{cmd}}(t) + \dot{\omega}_{\text{cmd}}(t), \quad t \geq t_0. \tag{3.35}$$

The rotational command inputs $\omega_{\text{cmd}}(t)$, and $\dot{\omega}_{\text{cmd}}(t)$ are described by

$$\omega_{\text{cmd}}(t) = J(\phi(t), \theta(t)) \left[-K_{P,\text{rot}}(\eta(t) - \eta_{\text{d}}(t)) + \dot{\eta}_{\text{d}}(t) \right], \quad t \geq t_0, \tag{3.36}$$

where $K_{P,\text{rot}} \in \mathbb{R}^{3\times 3}$ is positive-definite and user-defined. Here, $\omega_{\text{cmd}}(\cdot)$ represents the command angular velocity, which is a function of the UAV's current attitude, represented by $\eta(t)$, and the desired attitude, $\eta_{\text{d}}(t)$. The desired attitude, $\eta_{\text{d}}(t)$, is derived from the outer loop, ensuring the inner loop is aligned with the overall control objectives.

Both $\omega_{\text{cmd}}(t)$, and $\dot{\omega}_{\text{cmd}}(t)$ incorporate the Jacobian J as well as the time derivative of the Jacobian $\dot{J}$ to capture the proper frame of reference. These variables are defined by the equations

$$J(\phi(t), \theta(t)) = \begin{bmatrix} 1 & 0 & -\sin(\theta) \\ 0 & \cos(\phi) & \cos(\theta)\sin(\phi) \\ 0 & -\sin(\phi) & \cos(\phi)\cos(\theta) \end{bmatrix}. \tag{3.37}$$

3.6 Unmatched Uncertainties in MRAC for UAVs

In model reference adaptive control of UAVs, the presence of unmatched uncertainties requires a robust control strategy. These uncertainties can significantly influence both translational and rotational dynamics.

The dynamics of a UAV are subject to a variety of external influences that can disturb its intended path and stability. These unmatched uncertainties are disturbances that affect the system's behavior in ways not aligned with the input channel of the controller, making them challenging to predict and counteract directly.

For the translational motion, unmatched uncertainties can be forces arising from environmental factors such as unpredictable wind gusts or variations in air density, which are not uncommon in outdoor flight scenarios. These forces can introduce errors in position and velocity tracking and may be modeled as

$$m\dot{v}(t) = F_{\text{control}}(t) + \delta_{\text{trans}}(t), \quad v(t_0) = v_0, \quad t \geq t_0, \tag{3.38}$$

where $F_{\text{control}}(t)$ represents the control force generated by the propellers, and $\delta_{\text{trans}}(t)$ encapsulates all unmatched translational disturbances for which there is no characterization by means of a regressor vector.

The rotational dynamics of the UAV are similarly affected by unmatched uncertainties, which exhibit torques due to factors such as aerodynamic asymmetries, imprecise actuation, or external moments caused by environmental interactions. These dynamics can by modeled as

$$I\dot{\omega}(t) = \omega^{\times}(t)I\omega + \tau_{\text{control}}(t) + \delta_{\text{rot}}(t), \quad \omega(t_0) = \omega_0, \quad t \geq t_0, \tag{3.39}$$

where $\tau_{\text{control}}(t)$ represents the control torque, and $\delta_{\text{rot}}(t)$ the unmatched external torques for which there is no characterization by means of a regressor vector.

The challenges posed by these uncertainties are particularly pronounced in an quadcopter, where the interaction of multiple rotors with the environment can lead to complex dynamics. As such, the design of the control system must account for a broad spectrum of potential disturbances to ensure stability and performance. Robust control methodologies provide a framework for designing controllers that can withstand these disturbances, maintaining the UAV's operational integrity in the face of the unpredictable elements inherent in real-world flight conditions.

In this thesis, we will employ an external disturbance, in both the inner and outer loop, to challenge the controller and simulate extreme environmental factors UAVs could face.

3.7 Conclusion

This chapter has elaborated on the principles of MRAC and its application to managing the complexities of UAV dynamics, with a particular focus on the quadcopter. The discussion spanned from the foundational concepts of MRAC, addressing how this control strategy accommodates system uncertainties, to its practical implementation in enhancing the flight control systems of UAVs.

We began by distinguishing between matched and unmatched uncertainties, explaining how each interacts with the control system. Matched uncertainties, entering through the same channels as the control inputs, can often be counteracted directly by adaptive modifications to the control law. This adaptability is central to MRAC's strength, allowing it to adjust control parameters in real time to maintain desired system performance despite changes in system dynamics or external disturbances.

In contrast, unmatched uncertainties, or external disturbances not aligned with the control input paths, require more nuanced approaches. These disturbances are particularly challenging as they cannot be directly canceled by the control input and thus necessitate robust control strategies to maintain system stability and performance. We discussed various techniques to handle these uncertainties, including the implementation of dead zones within the control law to prevent excessive controller responses to minor disturbances, thus preserving system integrity and actuator longevity.

The application of MRAC to the quadcopter was detailed, highlighting the specific challenges and solutions relevant to UAVs. We showcased how MRAC could be tailored to meet the unique demands of aerial vehicles, which operate in highly dynamic environments and are subject to frequent disturbances. The discussion included the formulation of both the inner and outer loop controls, emphasizing the integration of MRAC to ensure robustness and adaptability in the UAV's control systems.

Finally, this chapter emphasized the necessity of a dual approach combining MRAC with robust control techniques as well as a baseline controller to effectively manage the uncertainties inherent in UAV operations along with potentially aggressive control behavior. By leveraging MRAC's adaptability and robust control strategies' resilience, the control system can ensure that the UAV maintains its desired trajectory and orientation, achieving high performance standards crucial for practical applications.

In the next chapter, we will review the theoretical foundations and practical implementations of reproducing kernel Hilbert spaces (RKHS) in the context of UAV control systems. We will explore how RKHS can be seamlessly integrated with existing MRAC strategies to enhance the adaptive capabilities of quadcopters. This integration aims to address the complexities of UAV dynamics more effectively, offering a robust framework

for handling the intricate environments UAVs operate in. The discussion will include detailed theoretical explanations as well as practical examples to demonstrate the application of RKHS in improving the precision and stability of quadcopter flight control.

Chapter 4

Adaptive Control on Reproducing Kernel Hilbert Spaces

In this chapter, we present an in-depth study of reproducing kernel Hilbert spaces (RKHS) and the design of adaptive control systems for functional uncertainties lying in these spaces. Thus, we will apply those results to quadcopter control. We begin by comparing parametric and non-parametric methods in control theory, highlighting the advantages and limitations of each. In this thesis, we refer to parametric method as those control systems that require some parameterization of the functional uncertainties by means of some regressor vector or an equivalent formulation. Non-parametric methods, such as the one proposed herein, do not require a parameterization of the functional matched uncertainty. The theoretical foundations of RKHS are then reviewed, explaining how this framework allows for sophisticated nonlinear modeling and the design of adaptable control systems. Practical implementations in MRAC for quadcopters are explored, detailing the algorithms and mathematical structures that enable robust and efficient flight control. Additionally, we analyze the benefits RKHS introduces to adaptive systems and address both the benefits and limitations that arise with its implementation. This chapter aims to bridge the gap between theoretical concepts of RKHS and their practical utility in improving adaptive control strategies for complex dynamic systems. The results from this chapter are primarily extracted from [12, 13].

4.1 Parametric vs. Non-Parametric Control in Adaptive Systems

The distinction between parametric and non-parametric control methods offers unique benefits suited to different types of control challenges. In the following, we discuss such distinction.

4.1.1 Parametric Control Methods

Parametric control assumes non-linear uncertainty $f(x)$ has the form $f(x) = \Theta^{\mathrm{T}}\Phi(t, x(t))$ where $\Theta \in \mathbb{R}^{N\times m}$ is the unknown constant coefficients and $\Phi(t, x(t)) \in \mathbb{R}^N$ denotes the vector of parametric functions. The primary advantage of parametric methods lies in their simplicity and efficiency. With a well-defined model structure, parametric controllers can be optimized to provide excellent performance under the assumption that the model accurately represents the system dynamics.

The efficacy of parametric methods, however, hinges heavily on the accuracy of the model. Error between the model and actual system behavior can lead to poor control performance and potentially unstable systems. Furthermore, parametric models often lack the flexibility to cope with structural changes in the system dynamics, making them less adaptable to systems with evolving or unknown dynamics.

4.1.2 Non-Parametric Control Methods

Non-parametric control methods, such as those based on RKHS, do not assume a predefined model structure for the nonlinear function $f(x)$. Instead, they assume $f(x)$ lies in a functional space and utilize data-driven approaches to approximate $f(x)$ directly from observed data. This approach allows non-parametric methods to adaptively learn and update the model as new data becomes available, providing a flexible framework that can handle complex, nonlinear, or unknown system dynamics effectively.

The strength of non-parametric methods lies in their versatility and ability to model dynamics from minimal assumptions about the underlying system. This makes them particularly valuable in applications where the system might be too complex for effective

modeling with parametric methods or where the system properties might change over time.

The flexibility of non-parametric methods, however, comes at the cost of higher computational demands. These methods typically require the processing of large datasets and the continuous updating of the model, which can be computationally intensive.

4.1.3 Integration in Adaptive Control Systems

In practice, the integration of parametric and non-parametric methods can often provide a balance between efficiency and flexibility. By combining the computational efficiency and predictability of parametric models with the adaptive capabilities of non-parametric methods, control systems can achieve robust performance even under varying and uncertain conditions.

For instance, a compounded approach might utilize a parametric model for the core system dynamics supplemented by a non-parametric model that captures deviations or anomalies from the predicted behavior. This combination allows for the handling of both typical operating conditions efficiently and adaptively responds to changes or unexpected disturbances.

In this thesis, we will explore the implementation of both parametric and non-parametric methods within the framework of RKHS to enhance the adaptive control of quadcopters. By leveraging the strengths of parametric control, such as its predictive efficiency and ease of implementation for well understood dynamics, alongside the robust adaptability of non-parametric methods through RKHS, This chapter will explore the theoretical foundation of RKHS, followed by practical implementation into a quadcopter control system.

4.2 Theoretical foundation of RKHS

4.2.1 Overview of RKHS

A Hilbert space provides a structured environment endowed with an inner product and is complete, that is, every Cauchy sequence converges to a point in the space itself [14, Ch. 3]. RKHSs are Hilbert spaces defined by closure of spans of kernel functions, located in the underlying subset of $\mathbb{R}^n$ at so-called 'centers'. There are many examples of Kernels. For instance, the Wendland kernel [15], employed in this thesis, distinguishes itself for its compact support and computational efficiency, making it an excellent choice for large scale applications such as control systems in UAVs. This kernel also provides a smooth approximation of functions.

Basis centers, or kernel centers, significantly influence how the space is modeled by concentrating the kernel's effect, allowing for a localized adaptation to data. In control systems, such as those managing the dynamics of a quadcopter, selecting appropriate basis centers can enhance the system's responsiveness to changes in dynamic behaviors, especially under varying operational conditions. However, as we will see in Section 5.2, even poor choices of basis centers can still significantly improve performance from the standard non-parametric approach.

Utilizing RKHS in the MRAC design for the quadcopter allows the adaptive control algorithm to efficiently adapt to changes, such as varying payloads or environmental conditions. The RKHS framework enables the control system to flexibly adjust to these dynamics without the need for explicit definitions of the nonlinearities involved. The kernel's ability to approximate complex functions means that it can intuitively grasp the underlying dynamics of the system based on observed data alone. This capability is fundamental to enhancing the adaptability and robustness of the control system, ensuring it performs well under a wide range of conditions. The choice of the Wendland kernel and its implementation through RKHS ensures that the control system remains robust, maintaining stability and effective response under unpredictable conditions.

4.2.2 Elements of RKHS

We begin by exploring the fundamental mathematical constructs used in the study of RKHSs within adaptive control systems, particularly applied to quadcopter dynamics. The subspace, denoted $\mathbb{X} \subset \mathbb{R}^n$, and the control space $\mathbb{U} \subset \mathbb{R}^m$, serve as the domains for the state vector and control inputs, respectively. These spaces are essential for modeling system interactions and the impact of control strategies. We expand these concepts to define $\mathcal{H}$, a RKHS of scalar-valued functions, and its vector-valued extension $\mathcal{H}^m$, which is formed by the Cartesian product of $\mathcal{H}$ m times. The kernel function $\mathfrak{K}(\cdot,\cdot)$ is central to RKHS applications. It assesses element similarity within the space, facilitating the elevation of data into a higher-dimensional space and controlling the smoothness and complexity of functions. The kernel function is defined as $\mathfrak{K} : \Omega \times \Omega \to \mathcal{H}$ such that

$$\mathfrak{K}(x, y) = \langle \Phi(x), \Phi(y) \rangle_{\mathcal{H}}, \tag{4.1}$$

where $\Phi : \mathbb{X} \to \mathcal{H}$ maps data from the input space $\mathbb{X}$ to the feature space $\mathcal{H}$. The kernel function is defined on the domain Ω, where $\Omega \subseteq \mathbb{X}$, ensuring that all evaluations are within the context of the state space. In this thesis, we consider kernel functions that are continuous over $\Omega \times \Omega$ and bounded on the diagonal, that is, there exists a constant $\bar{\kappa} > 0$ such that

$$\mathfrak{K}(\xi, \xi) = \|\mathfrak{K}_\xi\|_{\mathcal{H}}^2 \leq \bar{\kappa}^2, \quad \forall \xi \in \Omega, \tag{4.2}$$

where $\mathfrak{K}_\xi(\cdot) \triangleq \mathfrak{K}(\cdot, \xi)$. Consider the following RKHS-based function approximation used in MRAC systems

$$\hat{f}(x) = \sum_{i=1}^{N} \alpha_i \mathfrak{K}(x, x_i). \tag{4.3}$$

In this equation, α_i are coefficients adjusted through learning algorithms, which optimize the system's response by adapting to observed data. The kernel function $\mathfrak{K}$, such as the Gaussian or Wendland kernel, measures the similarity between the current state x and a set of predetermined state points x_i, allowing the system to interpolate or extrapolate the system dynamics efficiently. This modeling approach is especially valuable in areas where the dynamic model cannot be easily or fully parameterized, as it allows the controller to adapt based on real time data rather than relying solely on predefined models.

Finally, we introduce the evaluation operator E_ξ, crucial in this context, defined as such that $E_\xi : \mathcal{H} \to \mathbb{U}$

$$E_\xi f = f(\xi), \quad \forall \xi \in \Omega, f \in \mathcal{H}. \tag{4.4}$$

The evaluation operator plays an important role in enabling real time control adjustments by assessing function values at specific points in the space.

The native space $\mathcal{H}$ of the RKHS is characterized by the closure of the span of the operator kernel sections. It can be expressed as

$$\mathcal{H} \triangleq \overline{\text{span}\{\mathfrak{K}_\xi \alpha \mid \xi \in \Omega, \alpha \in \mathbb{U}\}}, \tag{4.5}$$

This structural characterization is fundamental for modeling and control implementations, as it provides a robust mathematical framework for handling complex data structures and dynamics within the control system. Due to the infinite-dimensional nature of $\mathcal{H}$, practical applications necessitate the construction of a finite-dimensional subspace, $\mathcal{H}_N$. This subspace, formed by kernel centers $\Xi_N \subset \Omega$, is essential for approximating $\mathcal{H}$ within computational limits

$$\mathcal{H}_N \triangleq \text{span}\{\mathfrak{K}(\cdot, \xi_i) \mid \xi_i \in \Xi_N\}. \tag{4.6}$$

These centers are critical as they determine the accuracy and efficiency of RKHS-based control algorithms. The Grammian matrix $\mathbb{K}(\Xi_N, \Xi_N)$, constructed from the inner products of kernel sections, facilitates the analysis and implementation of these algorithms.

We now explore the concept of the reproducing property, essential for evaluating functions within the RKHS framework. The reproducing property for vector-valued functions within this native space is defined by

$$\langle \mathfrak{K}_\xi \alpha, f \rangle_{\mathcal{H}} = \langle E_\xi^* \alpha, f \rangle_{\mathcal{H}} = \langle \alpha, E_\xi f \rangle_{\mathbb{U}} = \alpha^{\mathrm{T}} f(\xi), \tag{4.7}$$

for any $\alpha \in \mathbb{U}$ and $f \in \mathcal{H}$, where E_ξ^* denotes the adjunct operator of E_ξ [14, Definition 2.5]. This property connects the evaluation of functions at a point ξ with the inner product structure of the space, facilitating numerous theoretical results and practical

applications within RKHS.

To accommodate systems where control inputs or outputs are inherently multi-dimensional, we extend $\mathcal{H}$ to $\mathcal{H}^m$. This extension is realized by taking the Cartesian product of $\mathcal{H}$ multiple times, effectively forming a vector-valued RKHS

$$\mathcal{H}^m \triangleq \underbrace{\mathcal{H} \times \cdots \times \mathcal{H}}_{m \text{ times}}, \tag{4.8}$$

where m corresponds to the dimensionality of the control space $\mathbb{U}$. This construction is essential for systems requiring a framework that supports vector-valued function analysis and control strategy development.

Next, we examine the operator kernel and its section [16, 17], building on the foundation of the reproducing property. Operator kernels extend scalar kernels to matrix-valued functions, thus accommodating the manipulation of vector-valued functions within the RKHS defined in (4.8), we define

$$\mathcal{K}(\xi, \eta) \triangleq \mathfrak{K}(\xi, \eta) I, \tag{4.9}$$

$$\mathcal{K}_\xi(\cdot) \triangleq \mathfrak{K}(\xi, \cdot) I, \tag{4.10}$$

where $\mathcal{K}$ is the operator kernel and $\mathcal{K}_\xi$ is the operator kernel section, with I being the identity matrix on $\mathbb{R}^m$. The kernel section $\mathcal{K}_\xi$ represents the localized effect of the kernel at point ξ, facilitating precise control and adaptation within the RKHS framework.

Uncertainty classes within RKHS define the bounds of system uncertainties critical for robust control. The class C_R, as detailed in [12], encapsulates all functions in $\mathcal{H}$ whose norm does not exceed a specified threshold R

$$C_R \triangleq \{f \in \mathcal{H} \mid \|f\|_{\mathcal{H}^m} \leq R\}. \tag{4.11}$$

For practical implementations, a finite-dimensional subset $C_{R,\epsilon,N}$ is considered, which includes functions from C_R that adhere to a tolerance for projection error $\epsilon > 0$, and

involves a limited number of basis functions N

$$C_{R,\epsilon,N} = \{f \in C_R \mid \|(I - \Pi_N)f\|_{\mathcal{H}^m} \leq \epsilon\}, \tag{4.12}$$

where $\Pi_N : \mathcal{H}^m \to \mathcal{H}_N^m$ denotes the projection operator. The class $C_{R,\epsilon,N}$ is crucial for the design of controllers that must remain resilient within the scope of known uncertainties and computational constraints.

Finally, we address the power function, a fundamental concept in RKHS for assessing the approximation quality. The power function $P_N : \Omega \to \mathbb{R}$ measures how closely functions from a finite dimensional subspace $\mathcal{H}_N$ can approximate functions in the full space $\mathcal{H}$. Following [18, 19], we define the vector-valued power function of the subspace $\mathcal{H}_N \subseteq \mathcal{H}$ in the direction $\alpha \in \mathbb{U}$ as

$$P_N^\alpha(x) \triangleq \sqrt{\langle (K(x,x) - K_N(x,x))\alpha, \alpha \rangle_{\mathbb{U}}} \tag{4.13}$$

for all $x \in \mathbb{X}$. This function provides a worst-case scenario estimate of the deviation from the original function, crucial for the design of adaptive systems where control decisions rely on approximate models.

4.2.3 Conclusion

As we have seen, the theory of RKHS provides a robust mathematical foundation for analyzing complex data structures using kernel functions. This theory is particularly relevant to the field of control systems, where non-linearities and uncertainties are prevalent. By leveraging the power of RKHS, we can design adaptive control algorithms that are both robust and flexible, capable of responding to dynamic changes in system behavior.

The next section will focus on applying RKHS to adaptive control systems for quadcopters. It will illustrate how we use specific kernels, such as the Wendland kernel, to improve algorithm performance in real-world scenarios, demonstrating RKHS's effectiveness in practical control applications.

4.3 Application of RKHS in MRAC

4.3.1 Infinite Dimensional Application

In the integration of RKHS into MRAC, the objective is to encapsulate system uncertainties and nonlinearities within a functional framework, enhancing the adaptive control strategy. This section outlines the equations and the rationale behind their use in the RKHS-MRAC methodology. We will now present the standard equations outlining model reference adaptive control in reproducing RKHSs as seen in [20, 21, 12], beginning with a standard plant model described by the equation

$$\dot{x}(t) = Ax(t) + B\big(u(t) + E_{x(t)}f\big), \quad x(t_0) = x_0, \quad t \geq t_0, \tag{4.14}$$

where $x(t) \in \mathbb{X} \triangleq \mathbb{R}^n$ signifies the plant state evolving in the state space $\mathbb{X}$, and $u(t) \in \mathbb{U} \triangleq \mathbb{R}^m$ represents the control input within the control space $\mathbb{U}$. The matrix $A \in \mathbb{R}^{n\times n}$ is unknown while the matrix $B \in \mathbb{R}^{n\times m}$, is known and ensures the pair (A, B) is controllable. The nonlinear matched uncertainty in the system is f, and the term $E_{x(\cdot)}$ denotes the evaluation operator at the trajectory $x(t)$.

We also consider a reference model in the same form as standard MRAC defined in Section 3.2 by

$$\dot{x}_r(t) = A_r x_r(t) + B_r r(t), \quad x_r(t_0) = x_{r,0}, \quad t \geq t_0, \tag{4.15}$$

where $x_r(t) \in \mathbb{X}$ indicates the reference trajectory and $r(t) \in \mathbb{U}$ is the uniformly continuous and bounded reference command input. The matrices $A_r \in \mathbb{R}^{n\times n}$ and $B_r \in \mathbb{R}^{n\times m}$ are predefined such that A_r is Hurwitz, ensuring the system's stability. Considering the plant and reference model given by (4.14) and (4.15) assume that the matching conditions

$$A_r = A + B\alpha^{\mathrm{T}}, \tag{4.16}$$

$$B_r = B\beta^{\mathrm{T}}, \tag{4.17}$$

are satisfied by matrices $\alpha \in \mathbb{R}^{n\times m}$ and $\beta \in \mathbb{R}^{m\times m}$. To establish a systematic adaptive

law for trajectory tracking in the presence of system and functional uncertainties, we introduce a space $Z \triangleq \mathbb{R}^n \times \mathbb{R}^{n\times n} \times \mathbb{R}^{m\times m} \times \mathcal{H}$ and propose the following adaptive laws

$$\dot{\hat{\alpha}}(t) = -\Gamma_\alpha x(t) e^{\mathrm{T}}(t) P B, \quad \hat{\alpha}(t_0) = \alpha_0, \quad t \geq t_0, \tag{4.18a}$$

$$\dot{\hat{\beta}}(t) = -\Gamma_\beta r(t) e^{\mathrm{T}}(t) P B, \quad \hat{\beta}(t_0) = \beta_0, \tag{4.18b}$$

$$\frac{\partial \hat{f}}{\partial t}(t, \cdot) = \Gamma_f \mathcal{K}_{x(t)} B^{\mathrm{T}} P e(t), \quad \hat{f}(t_0, \cdot) = f_0, \tag{4.18c}$$

where $\dot{\hat{\alpha}} \colon [t_0, \infty) \to \mathbb{R}^{n\times m}$, $\dot{\hat{\beta}} \colon [t_0, \infty) \to \mathbb{R}^{m\times m}$, and $\hat{f} \colon [t_0, \infty) \times \mathcal{H}^m \to \mathbb{U}$ denote the adaptive gain matrices. The matrices $\Gamma_\alpha \in \mathbb{R}^{n\times n}$, $\Gamma_\beta \in \mathbb{R}^{m\times m}$, and $\Gamma_f \in \mathbb{R}^{m\times m}$ are the user-defined, symmetric, and positive-definite adaptive rate matrices, and $P \in \mathbb{R}^{n\times n}$ is the symmetric, positive-definite solution to the algebraic Lyapunov equation

$$-Q = A_r^{\mathrm{T}} P + P A_r, \tag{4.19}$$

where $Q \in \mathbb{R}^{n\times n}$ is a user-defined, symmetric, and positive-definite matrix that reflects desired performance characteristics such as convergence rates. Equation (4.19) ensures that the closed-loop system will have desirable stability properties. It is important to note that the adaptive law, which is defined by (4.18), embodies a partial differential equation (PDE), rather than an ordinary differential equation (ODE), due to the RKHS setting. The use of a PDE in this context is necessary to model the infinite dimensional aspect of functions within the RKHS, as it allows the adaptive law to handle variations across the entire spectrum of the function space, a requirement when dealing with the complexity of functional uncertainties in control systems.

The limiting control input is defined by

$$u(t) = \hat{\alpha}^{\mathrm{T}}(t) x(t) + \hat{\beta}^{\mathrm{T}}(t) r(t) - E_{x(t)} \hat{f}(t, \cdot), \quad t \geq t_0, \tag{4.20}$$

which combined the contributions of the adaptive gains $\hat{\alpha}(t)$, $\hat{\beta}(t)$, and $\hat{f}(t, \cdot)$ within the RKHS.

The system's performance, specifically how well the trajectory of the state $x(t)$ tracks the reference $x_r(t)$, is encapsulated in the trajectory tracking error dynamics

$$\begin{aligned} \dot{e}(t) = A_r e(t) - B\big(\hat{\alpha}^{\mathrm{T}}(t)x(t) + \hat{\beta}^{\mathrm{T}}(t)r(t) - E_{x(t)}\hat{f}(t,\cdot)\big), \\ e(t_0) = x_0 - x_{r,0}, \quad t \geq t_0, \end{aligned} \tag{4.21}$$

where the terms $\hat{\alpha}(t) \triangleq \alpha - \hat{\alpha}(t)$, $\hat{\beta}(t) \triangleq \beta - \hat{\beta}(t)$, and $\hat{f}(t,\cdot) \triangleq f(\cdot) - \hat{f}(t,\cdot)$ represent the discrepancies between the true values and their adaptive gains, which are adaptively updated to assure boundedness of $e(\cdot)$, $\hat{\alpha}(\cdot)$, $\hat{\beta}(\cdot)$, $\hat{f}(\cdot,\cdot)$ over time as well as asymptotic convergence of $e(\cdot)$ to zero..

Theorem 1 ([12])**.** Consider the nonlinear plant dynamics (4.14), the reference model (4.15), the control input (4.20), and the adaptive laws (4.18). Assume that the matching conditions (4.17) are verified, let $\Omega = \mathbb{X}$, and assume that the kernel $\mathcal{K}(\cdot,\cdot)$ that defines $\mathcal{H}$ is such that (4.2) is verified for all $\xi \in \mathbb{X}$. If the system given by (4.29) and (4.18) has a unique solution over the set $[t_0, \infty) \subseteq [0, \infty)$ for every initial condition, then the trajectories of (4.21) and (4.18) are uniformly bounded and the tracking error is convergent,

$$\|x(t) - x_r(t)\|_{\mathbb{X}} \to 0 \quad \text{as} \quad t \to \infty, \tag{4.22}$$

uniformly in $t_0 \geq 0$.

4.3.2 Finite-Dimensional Application

As outlined in Section 4.3.1, the control input prescribed by adaptive laws is inherently infinite dimensional due to the function $f(t,\cdot)$, with $t \geq t_0$, existing within an RKHS. The infinite dimensional nature makes it impossible to implement on real physical systems. To bridge the gap between theoretical models and implementable strategies, we derive finite dimensional approximations of the infinite dimensional entities.

The finite-dimensional approximation of the control input is predicated upon the finite-dimensional subspace $\mathcal{H}_N^m$. This subspace is constructed as a Cartesian product of m RKHS subspaces, each spanned by kernel functions centered at strategically chosen points in the state space. We define $\mathcal{H}_N$ as

$$\mathcal{H}_N^m \triangleq \mathcal{H}_1^m \times \cdots \times \mathcal{H}_N^m = \operatorname{span}\{\mathcal{K}(\cdot,\xi_i) | \xi_i \in \Xi_N, 1 \leq i \leq N, \alpha \in \mathbb{U}\}, \tag{4.23}$$

where $\mathcal{K}(\cdot, \xi_i)$ signifies the operator kernel function rooted at the point ξ_i.

The essential aspect of this methodology is the finite-dimensional representation of the uncertainty function $f(t, \cdot)$ by $\hat{f}_N(t, \cdot)$, serving as an approximation of the infinite dimensional function within the space $\mathcal{H}_N$. This approximation takes the form of a weighted sum of kernel functions, with the weights being time-varying coefficients $\hat{\Theta}_i(t)$ where

$$\hat{f}_N(t, \cdot) \triangleq \sum_{i=1}^{N} \mathcal{K}(\cdot, \xi_i)\hat{\Theta}_i(t), \quad t \geq t_0, \tag{4.24}$$

with each $\hat{\Theta}_i(t)$ residing within the set $\mathbb{U}$. These coefficients are adaptively tuned over time.

Subsequently, the infinite dimensional control law (4.18)is approximated by a finite dimensional controller applying $\hat{f}_N(t, \cdot)$

$$u_N(t) = \hat{\alpha}^{\mathrm{T}}(t)x_N(t) + \hat{\beta}^{\mathrm{T}}(t)r(t) - E_{x_N(t)}\hat{f}_N(t, \cdot), \tag{4.25}$$

where $u_N(t)$ denotes the implementable control input, and $E_{x_N(t)}$ is the evaluation functional applied to $\hat{f}_N(t, \cdot)$ at the state $x_N(t)$.

The finite-dimensional controller (4.25) is constructed to mimic the behavior of its infinite dimensional counterpart, aiming to minimize the trajectory tracking error within the physical constraints of the system. The detailed coordinate realizations of the adaptive laws in finite dimensions can be written as

$$\hat{f}_N(t, \cdot) = \begin{bmatrix} \sum_{i=1}^{N} \hat{\theta}^1_{N,i}(t)\mathfrak{K}_{\xi_{N,i}}(\cdot) \\ \sum_{i=1}^{N} \hat{\theta}^2_{N,i}(t)\mathfrak{K}_{\xi_{N,i}}(\cdot) \\ \vdots \\ \sum_{i=1}^{N} \hat{\theta}^m_{N,i}(t)\mathfrak{K}_{\xi_{N,i}}(\cdot) \end{bmatrix} = \sum_{i=1}^{N} \mathcal{K}(\cdot, \xi_{N,i})\hat{\Theta}_{N,i}(t), \tag{4.26}$$

illustrating the vectorized form of $\hat{f}_N(t, \cdot)$. Here, $\hat{\Theta}_{N,j}(t)$ are vectors composed of the coefficients $[\hat{\theta}^1_{N,j}(t), \ldots, \hat{\theta}^m_{N,j}(t)]^{\mathrm{T}}$, each corresponding to the kernel basis function $\mathfrak{K}_{\xi_{N,j}}(\cdot)$ centered at $\xi_{N,j}$.

With the finite dimensional subspace defined, we deploy the corresponding projection operator Π_N, as seen in [12]. To decompose the unknown vector nonlinearity into two orthogonal components, distinctly

$$E_{x(t)}f = E_{x(t)}(\Pi_N f + (I - \Pi_N)f), \quad t \geq t_0, \tag{4.27}$$

where $\Pi_N f$ represents the orthogonal projection of a generic function f from the infinite dimensional space $\mathcal{H}^m$ into $\mathcal{H}_N^m$. With the application of (4.25), the dynamics characterized by the plant (4.14) are reformatted as

$$\begin{aligned} \dot{x}_N(t) =& Ax_N(t) + B\left(u_N(t) + E_{x_N(t)}[\Pi_N f]\right) \\ &+ BE_{x_N(t)}[(I - \Pi_N)f], \quad x(t_0) = x_0, \quad t \geq t_0. \end{aligned} \tag{4.28}$$

Accordingly, we denote the trajectory tracking error dynamics as $e_N(t) \triangleq x_N(t) - x_r(t)$ for $t \geq t_0$. By defining $\hat{f}_N(t, \cdot) \triangleq f - \hat{f}_N(t, \cdot)$, the trajectory tracking error dynamics are defined as

$$\begin{aligned} \dot{e}_N(t) =& A_r e_N(t) - B\left(\hat{\alpha}_N^{\mathrm{T}}(t)x_N(t) + \hat{\beta}_N^{\mathrm{T}}(t)r(t) - E_{x_N(t)}[\Pi_N \hat{f}_N(t, \cdot)]\right) \\ &+ BE_{x_N(t)}[(I - \Pi_N)\hat{f}_N(t, \cdot)], \quad e_N(t_0) = x_0 - x_{r,0}. \end{aligned} \tag{4.29}$$

When examining (4.21) and (4.29), it becomes apparent that the trajectory tracking error dynamics, when described by a finite dimensional approximation within RKHS, incorporate an additional term, $BE_{x_N(t)}[(I - \Pi_N)f], t \geq t_0$, which can be interpreted as an unmatched disturbance. In the next section, we will introduce new adaptive laws to accommodate this term.

4.3.3 The Dead-Zone Modification

Similar to the dead-zone modification presented in Section 3.3.1, a new method is adopted here to mitigate the projection error described in (4.29), drawing from the

methodology outlined in [12]. We define the dead-zone width

$$d_{0,N} = \frac{2\|PB\|}{\lambda_{\min}(Q)} \sup_{\eta \in \Omega_0} P_N(\eta), \tag{4.30}$$

where $d_{0,N}$ is finite and $d \geq d_{0,N}$ is user-defined. Consider the adaptive laws

$$\dot{\hat{\alpha}}_N(t) = \begin{cases} -\Gamma_\alpha x_N(t) e_N^{\mathrm{T}}(t) PB, & \|e_N(t)\| \geq d, \\ 0, & \|e_N(t)\| < d, \end{cases} \tag{4.31a}$$

$$\dot{\hat{\beta}}_N(t) = \begin{cases} -\Gamma_\beta r(t) e_N^{\mathrm{T}}(t) PB, & \|e_N(t)\| \geq d, \\ 0, & \|e_N(t)\| < d, \end{cases} \tag{4.31b}$$

$$\dot{f}_N(t) = \begin{cases} \Gamma_N \Pi_N \mathcal{K}_{x_N(t)} B^{\mathrm{T}} P e_N(t), & \|e_N(t)\| \geq d, \\ 0, & \|e_N(t)\| < d, \end{cases} \tag{4.31c}$$

where $\dot{\hat{\alpha}}_N(t_0) = \dot{\hat{\alpha}}_0, \dot{\hat{\beta}}_N(t_0) = \dot{\hat{\beta}}_0, \dot{f}_N(t_0) = \dot{f}_0, \Gamma_\alpha \in \mathbb{R}^{n \times n}, \Gamma_\beta \in \mathbb{R}^{m \times m}$ and $\Gamma_N \in \mathbb{R}^{m \times m}$ are user-defined, symmetric, and positive-definite, and P denotes the symmetric, positive-definite solution of the Lyapunov equation (4.19).

4.4 Application of RKHS to the Quadcopter MRAC Design

We will now review the use of RKHSs to capture nonlinearities in the classical MRAC framework presented in Chapter 3. The use of RKHSs to capture nonlinearities complements the standard parametric MRAC strategy by providing non-parametric modeling capabilities. This integration allows for superior handling of uncertainties and model inaccuracies inherent in the nonlinear dynamics of aerial vehicles.

4.4.1 Integration of RKHS into MRAC Translational Dynamics

The restructuring of the translational dynamics under the MRAC framework equipped with RKHS begins with the virtual translational control input, as previously covered in Section 2.3.1. Consider the UAV's translational dynamics given by (**??**) and introduce

an unmatched external disturbance, $d : [t_0, \infty) \to \mathbb{R}^3$, as well as for a non-parameterized matched uncertainty $f : \mathbb{R}^6 \to \mathbb{R}^3$. This results in translational dynamics defined by

$$\dot{v}^{\mathbb{I}}(t) = m^{-1}\left(\mu(t) - \frac{1}{2}\rho_{\text{air}} S c_D R(\eta(t)) v(t) \|v(t)\| + f(x(t))\right) + d(t), \quad v^{\mathbb{I}}(t_0) = v_0^{\mathbb{I}}, \quad t \geq t_0. \tag{4.32}$$

Both the non-parameterized uncertainties and the unmatched uncertainties add a layer of complexity to the dynamics by challenging the controller's ability to maintain desired performance under additional, unpredictable conditions. The inclusion of these uncertainties test the robustness and adaptability of the control system, making the model more representative of real-world scenarios where such disturbances are common. By implementing these disturbances into the system dynamics, we can evaluate and enhance the controller's effectiveness under more challenging and realistic operating conditions.

As seen with classical MRAC, by defining $\Phi(t, v(t)) = -R(\eta(t))v(t)\|v(t)\|$ and $\Theta = \frac{1}{2}\rho_{\text{air}} S c_D$ the dynamics can be rewritten in the form

$$\underbrace{\begin{bmatrix} \dot{r}^{\mathbb{I}}(t) \\ \dot{v}^{\mathbb{I}}(t) \end{bmatrix}}_{\dot{x}(t)} = \underbrace{\begin{bmatrix} 0_{3\times3} & I_3 \\ 0_{3\times3} & 0_{3\times3} \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} r^{\mathbb{I}}(t) \\ v^{\mathbb{I}}(t) \end{bmatrix}}_{x(t)} + \underbrace{\begin{bmatrix} 0_{3\times3} \\ 1_3 \end{bmatrix}}_{B} \underbrace{\begin{bmatrix} m^{-1} & 0 & 0 \\ 0 & m^{-1} & 0 \\ 0 & 0 & m^{-1} \end{bmatrix}}_{\Lambda} \left(\mu(t) + \Theta^T \Phi(t, v(t)) + E_{x(t)} f\right) + \underbrace{\begin{bmatrix} d_x(t) \\ d_y(t) \\ d_z(t) \end{bmatrix}}_{d(t)} \tag{4.33}$$

where $d_x, d_y, d_z : [t_0, \infty) \to \mathbb{R}$ denotes the components of $d(\cdot)$. The reference model with the application of RKHS remains the same as classical MRAC and is defined by (3.28) as does the reference input defined by (3.29).

4.4.2 RKHS Enhanced Control Law

While the underlying dynamics remain constant, we must now incorporate the MRAC system outlined in Section 4.3 in the adaptive control strategy while employing kernel

functions that account for the nonlinearities that are not captured by traditional parametric methods. The control law is defined by

$$u(t) = \hat{K}_x^{\mathrm{T}}(t)x(t) + \hat{K}_r^{\mathrm{T}}(t)r_{\mathrm{cmd}}(t) - \hat{\Theta}_{\mathrm{MRAC}}^{\mathrm{T}}(t)\Phi(t, x(t)) - \begin{bmatrix} \hat{\Theta}_{N_x}^{\mathrm{T}}(t)\mathfrak{K}_{\Xi_N}(v_x(t)) \\ \hat{\Theta}_{N_y}^{\mathrm{T}}(t)\mathfrak{K}_{\Xi_N}(v_y(t)) \\ \hat{\Theta}_{N_z}^{\mathrm{T}}(t)\mathfrak{K}_{\Xi_N}(v_z(t)) \end{bmatrix}, \quad t \geq t_0. \tag{4.34}$$

Here, $\hat{\Theta}_{\mathrm{MRAC}} : [t_0, \infty) \to \mathbb{R}^{n \times m}$ denote the adaptive weights for the parametric component of the control law, while $\hat{\Theta}_{N_x}(t)$, $\hat{\Theta}_{N_y}(t)$, and $\hat{\Theta}_{N_z}(t) \in \mathbb{R}^N$, represent the adaptive gains for the RKHS-based non-parametric component, each corresponding to the three components in the control space, respectively. Although $\hat{\Theta}_{N_x}(t)$, $\hat{\Theta}_{N_y}(t)$, and $\hat{\Theta}_{N_z}(t)$ are configured in real space $\mathbb{R}^N$, they correspond to coefficients that adaptively weigh the RKHS contributions, effectively embedding the dynamic behavior within the RKHS despite their representation in $\mathbb{R}^N$. The vector $\mathfrak{K}_{\Xi_N}(\cdot) \triangleq [\mathfrak{K}_{\xi_1}(\cdot), \ldots, \mathfrak{K}_{\xi_N}(\cdot)]^{\mathrm{T}} \in \mathbb{R}^N$ captures the collection of kernel basis functions located at the centers in $\Xi_N \subset \mathbb{R}^3$, where $\mathfrak{K}_{\xi_i}(\mathbf{x})$ denotes a kernel function centered at ξ_i.

4.4.3 Basis Center Configuration

In this thesis, we define the basis centers, denoted as Ξ_N, within the velocity space $\mathbb{R}^3$ to mitigate the effects of unmodeled matched and unmatched uncertainties due to aerodynamic forces. The basis centers are arranged to forms lattices covering the relevant domain of the velocity space, which allows for a dense and uniform approximation of the functional uncertainties within the RKHS framework. Two lattices are formed, namely a cubic latex and a toroidal latex centered around the user defined trajectory. This strategic placement ensures that the RKHS-based component of the MRAC controller can effectively weigh and compensate for aerodynamic effects throughout the entire operating range.

For the cubic latex, the set of basis centers as

$$\Xi_N = \left\{ \mathbf{v}_{ijk} \mid \mathbf{v}_{ijk} = \begin{bmatrix} i\Delta v_x \\ j\Delta v_y \\ k\Delta v_z \end{bmatrix},\ i = 0, \ldots, N_x - 1,\ j = 0, \ldots, N_y - 1,\ k = 0, \ldots, N_z - 1 \right\} \tag{4.35}$$

where Δv_x, Δv_y, and Δv_z are the spacings between adjacent points in the lattice along the three components in the velocity space, respectively, and N_x, N_y, and N_z denote the number of points along each axis. The kernel functions centered at these basis centers work collaboratively to form an adaptive mesh that spans the entire velocity space, which is crucial for the kernel based adaptive control law expressed in (4.34). For the toroidal latex, the basis centers are defined by

$$\Xi_N \triangleq \bigcup_{\tau \in \mathcal{T}} \left\{ \xi \in \mathbb{R}^3 : \xi = R[cos\beta, \sin\beta, 0]^{\mathrm{T}} + v_{\mathrm{ref}}(\tau) \right\}, \tag{4.36}$$

where $\mathcal{T} \subset [t_0, \infty)$ is user-defined and $v_{\mathrm{ref}} : [t_0, \infty) \to \mathbb{R}^3$ denotes the reference velocity. The radius of the trajectory is represented by $R \in \mathbb{R}$ while $\beta \in \mathbb{R}$ denotes the angle at which a basis center will be placed, perpendicular to the trajectory.

4.4.4 RKHS-Enhanced Adaptive Law

While the classical adaptive laws for MRAC remain unchanged and are written in the form

$$\dot{\hat{K}}_x(t) = -\Gamma_x x(t) e^{\mathrm{T}}(t) PB, \qquad \hat{K}_x(t_0) = K_{x0}, \quad t \geq t_0, \tag{4.37a}$$

$$\dot{\hat{K}}_r(t) = -\Gamma_r r_{\mathrm{cmd}}(t) e^{\mathrm{T}}(t) PB, \qquad \hat{K}_r(t_0) = K_{r0}, \tag{4.37b}$$

$$\dot{\hat{\Theta}}_{\mathrm{MRAC}}(t) = \Gamma_\Theta \Phi(t, x(t)) e^{\mathrm{T}}(t) PB, \quad \hat{\Theta}_{\mathrm{MRAC}}(t_0) = \Theta_{\mathrm{MRAC0}}. \tag{4.37c}$$

Additional laws for the parametric components of the control law. Additional laws must now be added to adjust the weights corresponding to the RKHS component. These

adaptive laws are defined by

$$\dot{\hat{\Theta}}_{N_x}(t) = \gamma_x \mathbb{K}^{-1}(\Xi_N, \Xi_N)\mathfrak{K}_{\Xi_N}(v_x(t))\tilde{x}_x(t), \quad \hat{\Theta}_{N_x}(t_0) = \Theta_{N_x,0}, \quad t \geq t_0, \tag{4.38a}$$

$$\dot{\hat{\Theta}}_{N_y}(t) = \gamma_y \mathbb{K}^{-1}(\Xi_N, \Xi_N)\mathfrak{K}_{\Xi_N}(v_y(t))\tilde{x}_y(t), \quad \hat{\Theta}_{N_y}(t_0) = \Theta_{N_y,0}, \tag{4.38b}$$

$$\dot{\hat{\Theta}}_{N_z}(t) = \gamma_z \mathbb{K}^{-1}(\Xi_N, \Xi_N)\mathfrak{K}_{\Xi_N}(v_z(t))\tilde{x}_z(t), \quad \hat{\Theta}_{N_z}(t_0) = \Theta_{N_z,0}. \tag{4.38c}$$

where

$$\tilde{x}(t) = B^T P e(t) \in \mathbb{R}^3 \tag{4.39}$$

$\gamma_x, \gamma_y, \gamma_z \in \mathbb{R}^{N\times N}$ are user-defined symmetric, positive-definite adaptative rate matrix. The Grammian matrix, used within the adaptive law (4.38), is defined as

$$\mathbb{K}(\Xi_N, \Xi_N) = \begin{bmatrix} \mathfrak{K}(\xi_1, \xi_1) & \cdots & \mathfrak{K}(\xi_1, \xi_N) \\ \vdots & \ddots & \vdots \\ \mathfrak{K}(\xi_N, \xi_1) & \cdots & \mathfrak{K}(\xi_N, \xi_N) \end{bmatrix}, \tag{4.40}$$

where each entry $\mathfrak{K}(\xi_i, \xi_j)$ evaluates the kernel between the points ξ_i and ξ_j in the selected set of centers Ξ_N. This matrix is integral to computing the inverse kernel matrix $\mathbb{K}^{-1}(\Xi_N, \Xi_N)$, which directly influences the adaptive updates of $\hat{\Theta}_N(t)$.

4.4.5 Wendland Kernel

The Wendland kernel, employed in this thesis, is a type of radial basis function that is particularly favored for its computational properties. It is defined by the equation

$$\mathfrak{K}(r) \triangleq (1-r)_+^4(4r+1), \tag{4.41}$$

where $r = \frac{\|x-y\|}{l}$ represents a scaled radius between points x and y in the input space. The parameter $l > 0$ is a length scale that adjusts the sensitivity of the kernel to the distances between points, effectively controlling how rapidly the kernel value decreases with increasing distance between points. The positive part function $(x)_+$ is defined to be

$$(x)_+ \triangleq \begin{cases} x, & \text{if } x > 0, \\ 0, & \text{if } x \leq 0. \end{cases} \tag{4.42}$$

The Wendland kernel is chosen for its compact support and smoothness, key attributes for the RKHS framework in this thesis. Its compact support ensures that the kernel impacts only a limited range of nearby points, significantly enhancing computational efficiency. Furthermore, the smoothness of the Wendland kernel facilitates the creation of continuously differentiable functions. These properties optimize the RKHS application by minimizing computational demands and ensuring stability in the adaptive control process.

4.5 Benefits of RKHS-Embedded Adaptive Control Systems

The use of RKHS tools to improve the performance of MRAC systems brings significant advancements in the management and control of complex dynamic systems. This section covers some of these benefits.

One of the foundational benefits of RKHS-embedded adaptive control is its superior capability to counter functional uncertainties in nonlinear system dynamics. Traditional linear models often struggle to capture the complex behaviors of systems such as quadcopters, whose operational dynamics are not only nonlinear but also vary significantly under different flight conditions. RKHS offers an adaptive approach that can approximate these nonlinear functions with high accuracy, without requiring detailed prior modeling of the system's nonlinear characteristics.

RKHS-based control systems inherently exhibit enhanced robustness to disturbances and uncertainties, crucial for adaptive control applications like those in aerospace. The strength of RKHS methodologies lies in their non-parametric nature, allowing the control model to autonomously redefine and compute its dynamics, freeing it from the constraints of a fixed structure. This adaptability enables the system to continuously adjust to variable operational conditions, effectively compensating for model inaccuracies and external disturbances.

Moreover, the flexibility of RKHS to utilize diverse kernels, such as the Wendland kernel, provides additional benefits. These kernels are specifically chosen to optimize sys-

tem performance in particular environments, enhancing the system's capability to handle non-linear dynamics and multi-dimensional state spaces efficiently. By leveraging computational efficiencies and targeted feature extraction capabilities of these kernels, RKHS based MRAC systems maintain stability and accuracy through complex and evolving environmental challenges. Alternative kernels can be found in [15, Ch. 9].

Finally, integrating RKHS methodologies into existing MRAC systems primarily requires adjustments to the controller input, with minimal changes needed to the overall structure and framework. This relatively straightforward integration ensures that existing systems can be easily upgraded to include advanced adaptive control capabilities, thereby enhancing their functionality with minimal effort.

4.6 Challenges and Limitations

RKHS-embedded MRAC systems enhance adaptability and robustness against disturbances. However, this integration is accompanied by significant challenges and limitations, notably in computational complexity, model dependence, scalability, and practical implementation.

The primary challenge in the application of RKHS to adaptive control systems is the high computational complexity. This complexity originates from the extensive calculations required to handle kernel functions and the associated Grammian matrix, especially as the number of data points or the system dimensions increases. Each update of control parameters involves calculations over kernel functions, which scale quadratically with the number of points in the kernel's support. For dynamic systems that require real time control responses, the computational delays introduced can impede system performance and stability, particularly in sectors where rapid responses are crucial, such as aerospace industries.

The success of RKHS based control systems hinges significantly on the alignment of the kernel function with the system's true dynamics. Misalignment can cause poor performance and unexpected responses. The early stages of RKHS integration are particularly challenging due to the possible scarcity of initial data, which is critical for accurately

modeling system behaviors. Insufficient or low quality data can greatly reduce model effectiveness, leading to poor controller performance.

4.7 Conclusion

This chapter has detailed the integration of RKHS into MRAC systems, emphasizing its application in managing the dynamic complexities of quadcopter flight. We've explored the theoretical aspects of RKHS, emphasizing its non-parametric nature that allows for dynamic and flexible modeling. This feature is especially beneficial in aerospace applications, where adaptability to evolving conditions is crucial. The use of RKHS for non-parametric modeling complements traditional parametric MRAC methods by providing a more flexible and robust adaptation mechanism, which precisely models uncertainties and the system's dynamic behavior.

By employing the Wendland kernel within the RKHS, we leverage its compact support and smoothness to enhance computational efficiency and ensure continuous differentiability of the control functions. These properties are particularly valuable in real-time control environments, where quick adaptation to changing conditions is crucial. The strategic configuration of basis centers within the velocity space optimizes the kernel's effectiveness, ensuring that only nearby points are influenced, which reduces computational overhead and improves the system's response to real-world conditions.

In practical terms, integrating RKHS into existing MRAC frameworks requires minimal adjustments, primarily to the controller input, without overhauling the existing system architecture. This ensures that legacy systems can be efficiently upgraded, benefiting from advanced control capabilities. Furthermore, this adaptation not only tests the controller's robustness but also enhances its applicability to scenarios where unpredictability is common.

Looking ahead, the next chapter will present simulation results to demonstrate the enhanced performance of these RKHS-embedded control systems. We will evaluate the practical implications of these theoretical advancements by examining the performance of quadcopter controllers through simulations, providing a comprehensive assessment of

their effectiveness in typical and challenging scenarios.

Chapter 5

Simulation Results

This chapter analyzes the robust MRAC system for quadcopters presented in the previous chapters, assessing its performance through various configurations and enhancements. We begin with detailed evaluations of the baseline MRAC system's capability in tracking reference trajectories and adapting dynamically to external disturbances. Subsequent sections analyze the changes in performance resulting from integration of RKHS into the MRAC framework, providing a comparative analysis under identical conditions to underscore the enhancements in adaptability and precision. Additionally, this chapter discusses the computational costs associated with these implementations, offering insights into the trade-offs between system efficiency and computational demand. These analyses collectively aim to illustrate the efficacy and potential of advanced control strategies in managing complex dynamics in real-world scenarios through simulations.

5.1 Performance of Robust MRAC

This section examines the MRAC system's performance, presented in Chapters 3 and 4, particularly its response to complex, external disturbances applied as forces. These disturbances represent real-world challenges that feature abrupt and unpredictable changes, testing the MRAC's capacity to dynamically adapt and maintain control. The MRAC system aims to ensure the UAV follows its predetermined trajectory and maintains target velocities starting from a state of rest, despite these complexities.

We begin our analysis of the MRAC system's robustness with the parameters delin-

eated in Tables 5.1 and 5.2, which represent the foundational settings for the translational and rotational dynamics of the UAV.

Parameter	**Value [Units]**
Dead Zone Width	0.2
m	1 kg
S	0.07 m^2
ρ_{air}, $\rho_{\text{air,est}}$	1.225 $\mathrm{kg/m}^3$, 1.23 $\mathrm{kg/m}^3$
c_D, $c_{D,\text{est}}$	$\begin{bmatrix} 0.001 & 0 & 0 \\ 0 & 0.001 & 0 \\ 0 & 0 & 0.001 \end{bmatrix}$, $\begin{bmatrix} 0.0013 & 0 & 0 \\ 0 & 0.0008 & 0 \\ 0 & 0 & 0.0011 \end{bmatrix}$
$\Gamma_{x_{\text{tran}}}, \Gamma_{\theta_{\text{tran}}}$	$10^3 \cdot I_6$
$\Gamma_{r_{\text{tran}}}$	$3 \cdot 10^2 \cdot I_3$
Q_{tran}	$4 \cdot 10^{-3} \cdot \text{diag}([1, 1, 20, 1, 1, 20])$
$K_{P_{\text{tran}}}, K_{I_{\text{tran}}}, K_{D_{\text{tran}}}$	$8 \cdot I_3$
$A_{\text{ref, tran}}$	$\begin{bmatrix} 0_{3\times3} & I_3 \\ -8 \cdot I_3 & -8 \cdot I_3 \end{bmatrix}$
$B_{\text{ref, tran}}$	$\begin{bmatrix} 0_{3\times3} \\ m^{-1} \cdot I_3 \end{bmatrix}$

Table 5.1: Translational parameters for classical MRAC

Parameter	**Value (Units)**
I, I_{est}	$\begin{bmatrix} 2.27199 \cdot 10^{-2} & 0 & 0 \\ 0 & 2.20204 \cdot 10^{-2} & 0 \\ 0 & 0 & 1.61469 \cdot 10^{-2} \end{bmatrix}$ ($\mathrm{kg \cdot m}^2$), $\begin{bmatrix} 2.33 \cdot 10^{-2} & 0 & 0 \\ 0 & 2.25 \cdot 10^{-2} & 0 \\ 0 & 0 & 1.60 \cdot 10^{-2} \end{bmatrix}$ ($\mathrm{kg \cdot m}^2$)
$K_{P_{\text{rot}}}, K_{I_{\text{rot}}}, K_{D_{\text{rot}}}$	$10 \cdot I_3$
$A_{\text{ref, rot}}$	$-10 \cdot I_3$
$B_{\text{ref, rot}}$	I_3
Q_{rot}	I_3
$\Gamma_{x_{\text{rot}}}, \Gamma_{\Theta_{\text{rot}}}$	$5 \cdot 10^2 \cdot I_6$
$\Gamma_{r_{\text{rot}}}$	$10^3 \cdot I_3$

Table 5.2: Rotational parameters for classical MRAC

Armed with these parameters, the system's resilience against differentiable and non-differentiable disturbances is tested. Key performance indicators, such as trajectory tracking and velocity control, are scrutinized to quantify the efficacy of the MRAC, particularly with the integration of the dead-zone modification. Successively, we will review the results obtained under various disturbance conditions.

5.1.1 Performance with Continuous Differentiable Disturbance

The MRAC system is evaluated from an initial zero position and velocity, providing a baseline to assess the system's responsiveness and accuracy effectively. The robust MRAC's capability is evaluated under a continuous differentiable disturbance that simulates varying aerodynamic conditions. The disturbance is uniformly applied across the x, y, and z axes, characterized by a sinusoidal pattern oscillating between zero and 0.6 N. This shift ensures that comparisons with a discontinuous non-differentiable step function and a non-differentiable function is accurate.

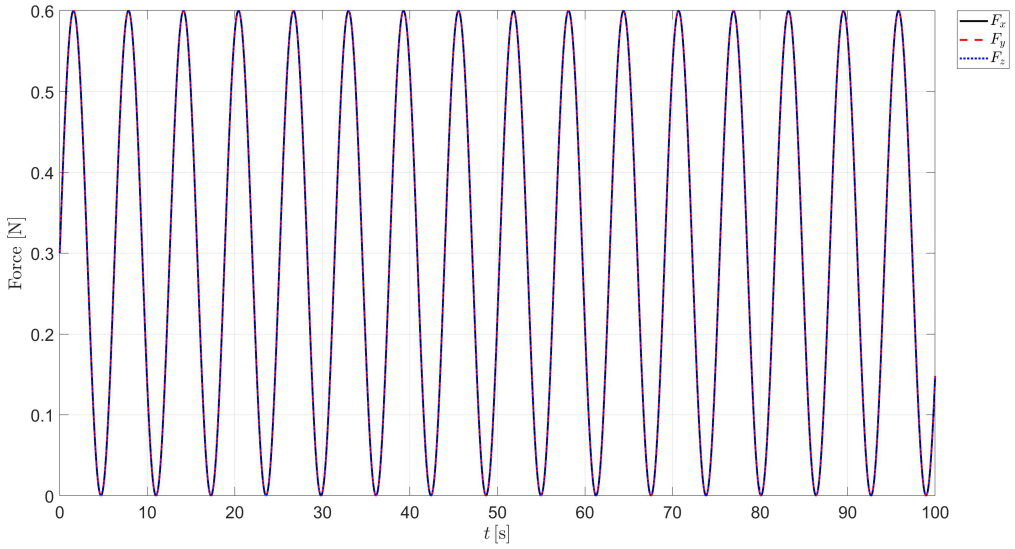

Figure 5.1: Applied continuous differentiable external disturbance

The disturbance applied to the system can be mathematically represented in each spatial direction by

$$F_x(t) = 0.3\sin(t) + 0.3, \qquad t \geq t_0, \tag{5.1a}$$

$$F_y(t) = 0.3\sin(t) + 0.3, \tag{5.1b}$$

$$F_z(t) = 0.3\sin(t) + 0.3. \tag{5.1c}$$

We now examine the velocity tracking performance of the MRAC system through both one-dimensional and three-dimensional perspectives.

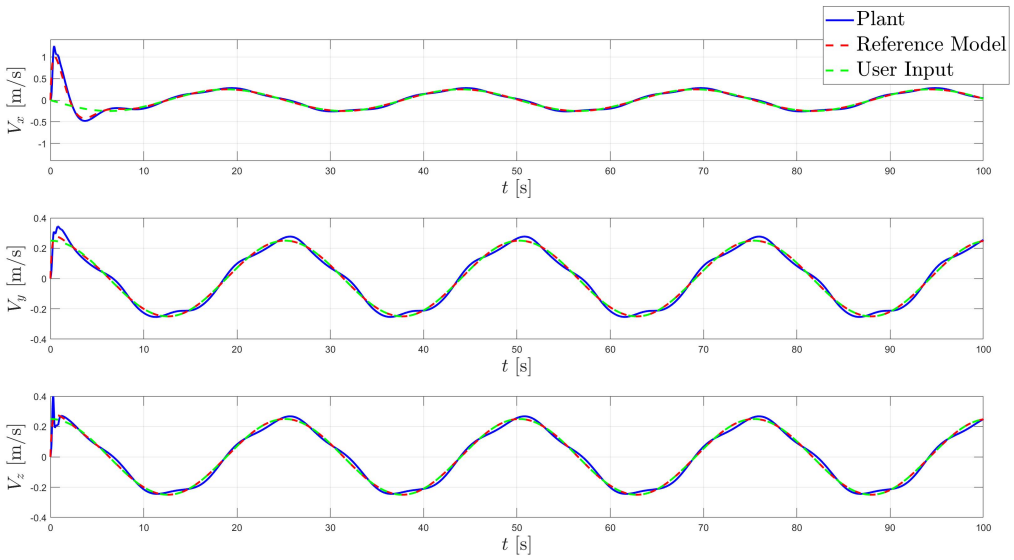

Figure 5.2: Velocity tracking of MRAC under continuous differentiable disturbance

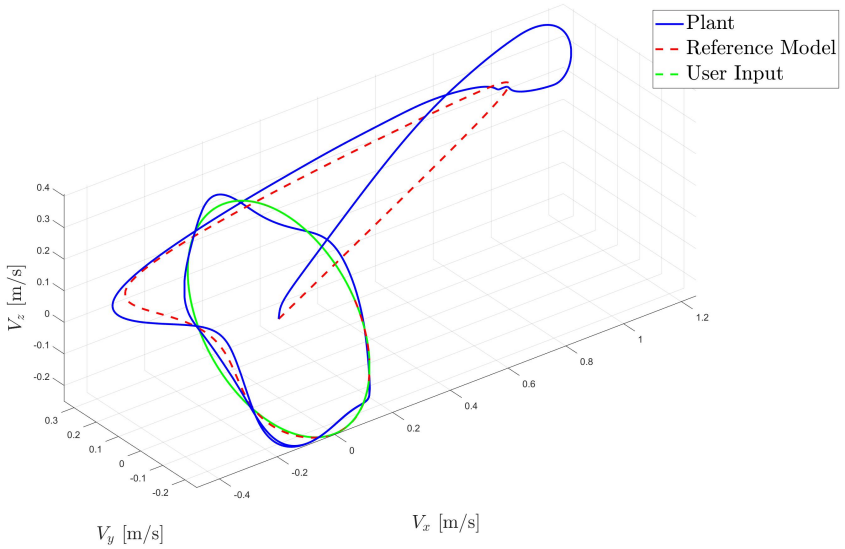

Figure 5.3: Three-dimensional velocity tracking of MRAC under continuous differentiable disturbance

Figures 5.2 and 5.3 illustrate the system's velocity tracking in response to a continuous and differentiable external disturbance. Initially, the system experiences significant deviations from the set trajectory, primarily due to starting conditions that are far removed from the target values. While the velocity tends to align towards the desired trajectory, continuous oscillations around the set path are observed due to the external disturbance pushing it off course. Figure 5.2 distinctly shows how the velocity components undergo constant fluctuations, failing to stabilize effectively over time. Figure 5.3

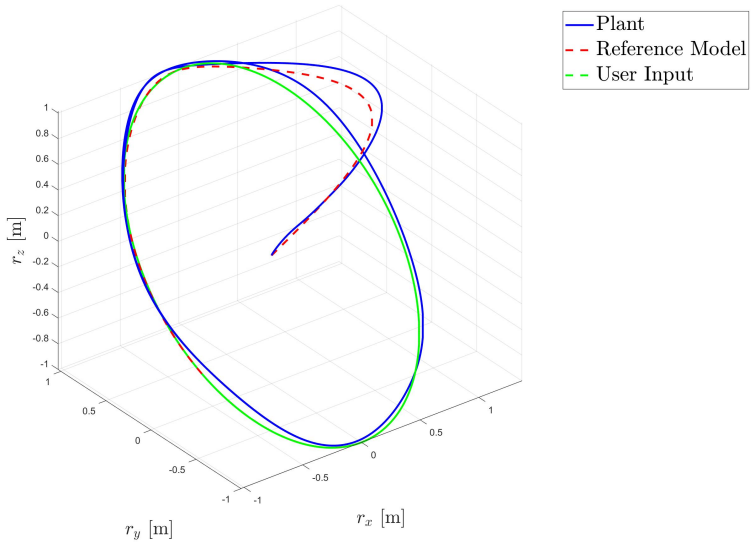

Figure 5.4: Three-dimensional position tracking of MRAC under continuous differentiable disturbance

adds perspective by indicating that these minor deviations persist, demonstrating the classical MRAC's struggle to maintain precise trajectory adherence under dynamic and challenging conditions.

Following the analysis of velocity tracking, we shift focus to position tracking. Although the controller manages to bring the trajectory closer to the desired path than seen with the velocity, slight but persistent fluctuations are evident.

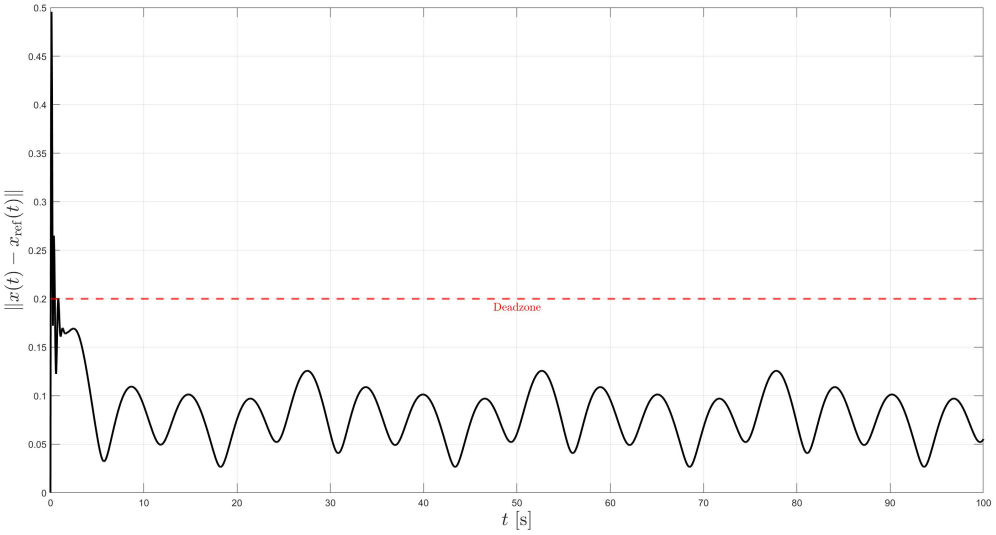

Figure 5.5: Error dynamics in MRAC under continuous differentiable disturbance

The MRAC's performance is further quantified by examining the error dynamics and the evolution of adaptive gains. The trajectory tracking error encompasses both translational position and velocity, which provides a holistic measure of the system's accuracy. Although the error dynamics, shown in Figure 5.5, indicate that the errors remain within the dead zone, they are still notably higher than desired. This underscores the significance of the disturbance and the robustness required from the control system.

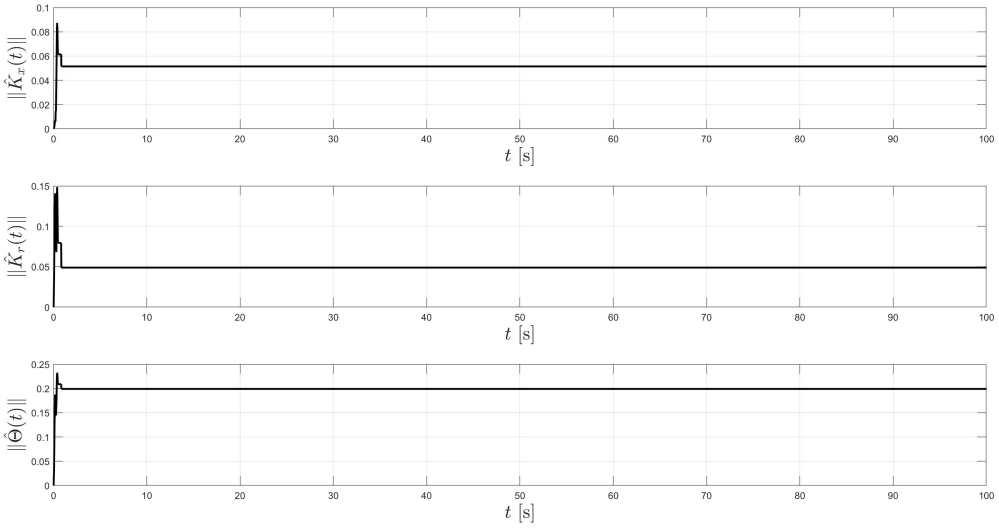

Figure 5.6: Adaptive gain adjustment in MRAC under continuous differentiable disturbance

Figure 5.6 shows that the adaptive gains initially surge as the MRAC system works to correct errors and stabilize performance. Once the system's error remains consistently within the dead zone, the gains level off, signifying that the controller has reached an effective operational state.

In the following section, we will review the controller's performance under a continuous non-differentiable function to further explore the system's adaptability and the effectiveness of the dead-zone modification.

5.1.2 Performance with Continuous Non-differentiable Disturbance

In this section, we evaluate the robust MRAC system under a continuous non-differentiable disturbance designed to introduce high-frequency variations in force. This type of disturbance is characterized by a piecewise linear function that increases and decreases over a defined period, making the problem fundamentally more challenging due to its non-differentiable nature.

The mathematical representation of the disturbance applied uniformly across all spatial directions is given by

$$F_x(t) = 0.6 \cdot w\left(t, \frac{8\pi}{5}\right)^{1/5}, \qquad t \geq t_0, \tag{5.2a}$$

$$F_y(t) = 0.6 \cdot w\left(t, \frac{8\pi}{5}\right)^{1/5}, \tag{5.2b}$$

$$F_z(t) = 0.6 \cdot w\left(t, \frac{8\pi}{5}\right)^{1/5}. \tag{5.2c}$$

where

$$w(t, \tau) = 2\left|\frac{t}{\tau} - \left\lfloor \frac{t}{\tau} + 0.5 \right\rfloor\right|, \quad t \geq t_0, \tag{5.3}$$

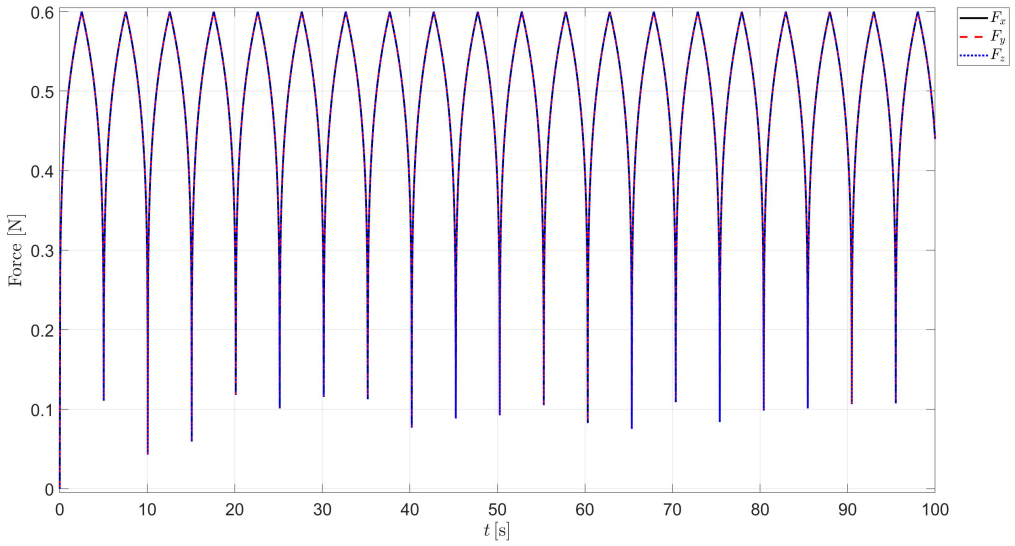

Figure 5.7: External continuous non-differentiable disturbance

Figure 5.7 illustrates the complex nature of the external disturbance. The high-frequency variations in force necessitate continuous adjustments in the MRAC's control strategies to mitigate immediate impacts on system stability and trajectory adherence.

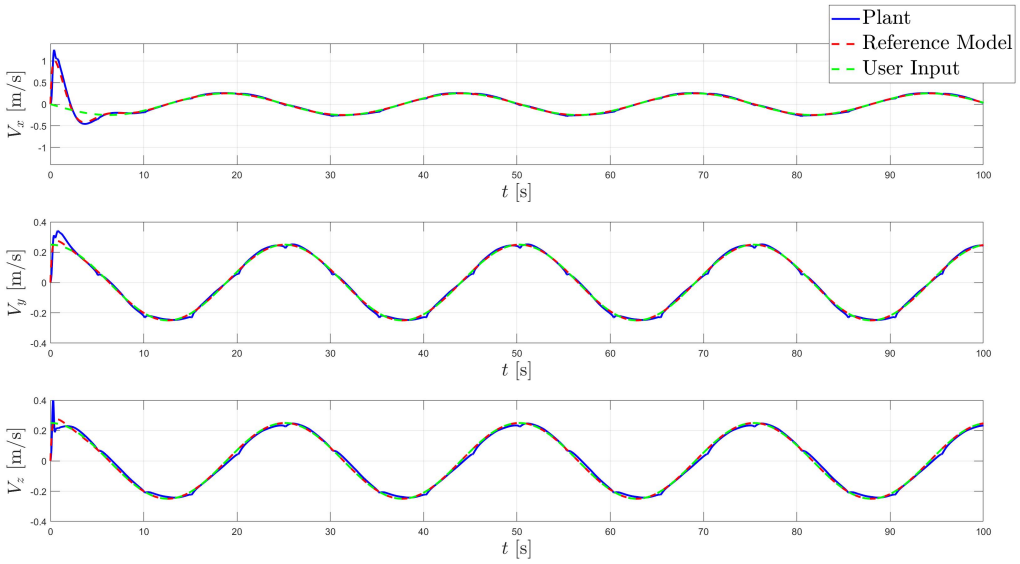

Figure 5.8: One-dimensional velocity tracking under continuous non-differentiable disturbance

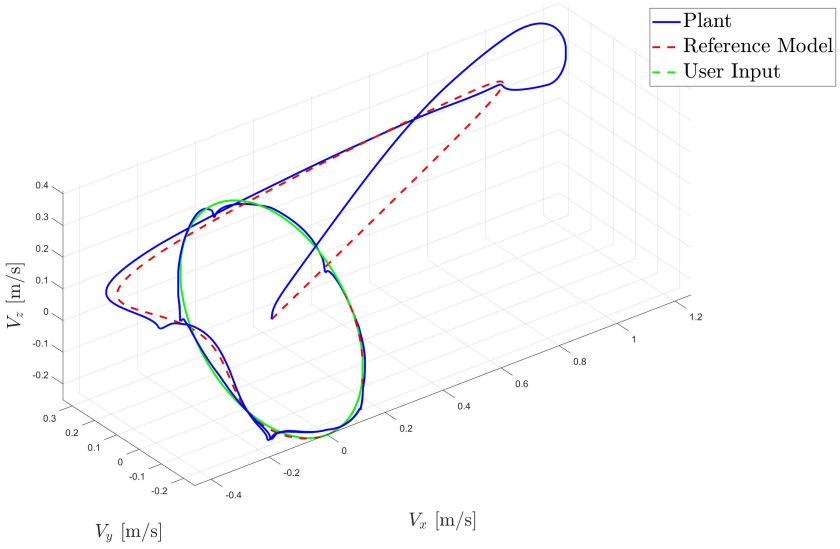

Figure 5.9: Three-dimensional velocity tracking under continuous non-differentiable Disturbance

The velocity tracking presented in Figures 5.8 and 5.9 shows notable deviations from the set trajectory, with sharper adjustments than those observed under the smooth differentiable disturbance. The irregular and abrupt changes illustrate the MRAC's struggle to maintain precise trajectory adherence under dynamic and challenging conditions.

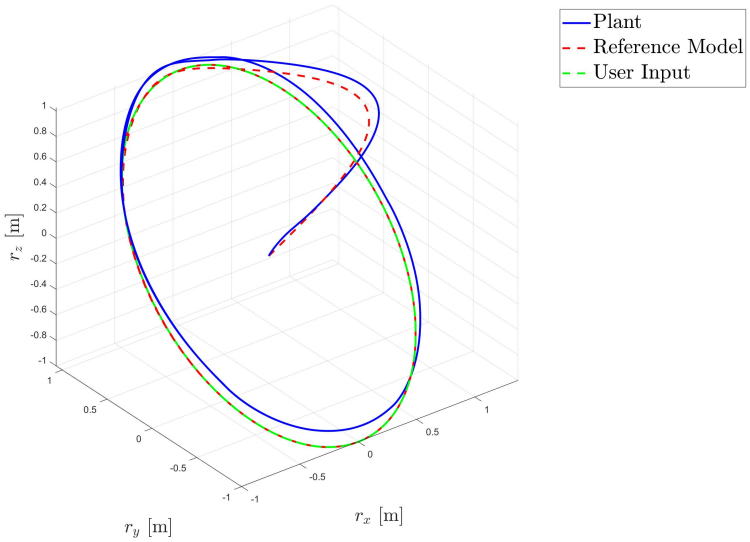

Figure 5.10: Three-dimensional position tracking under continuous non-differentiable disturbance

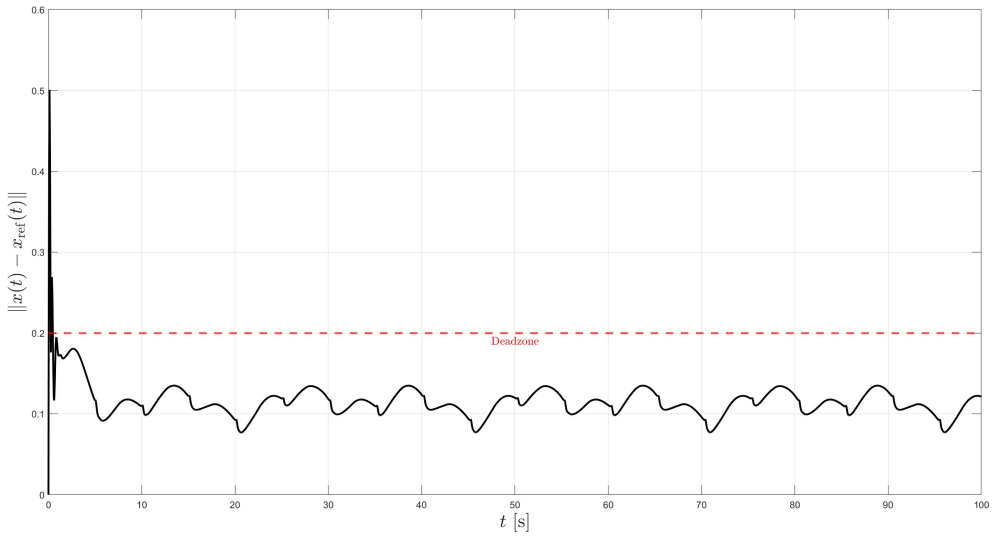

Figure 5.11: Error dynamics under continuous non-differentiable disturbance

Despite the volatile velocity tracking, the position tracking depicted in Figure 5.10 remains relatively smooth, but consistent deviation from the desired trajectory is evident. Figure 5.11 shows that while errors remain within the dead zone, they settle at a higher level compared to those observed with continuous differentiable disturbances, highlighting the increased difficulty in error minimization under non-differentiable conditions.

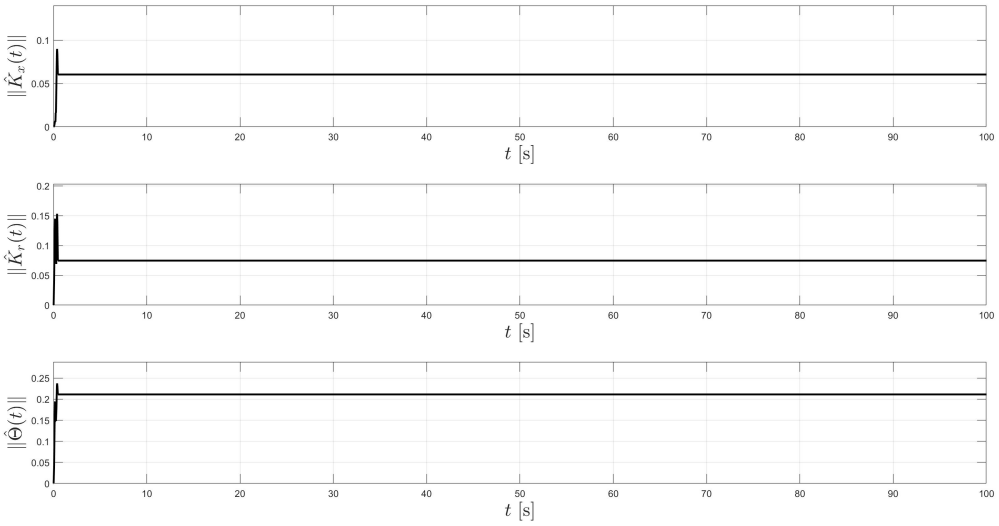

Figure 5.12: Adaptive gain adjustment under continuous non-differentiable disturbance

Adaptive gains, as seen in Figure 5.12, initially surge to combat the disturbance before stabilizing, similar to what was seen in with the continuous differentiable disturbance.

In the following section, we will review the controller's performance under a discontinuous non-differentiable function to further explore the system's adaptability and the effectiveness of the dead-zone modification.

5.1.3 Performance with Discontinuous Non-differentiable Disturbance

In this section, we assess the performance of the MRAC system under the influence of a discontinuous non-differentiable disturbance, modeled explicitly as a step function. This step function alternates between values of 0 and 0.6 at intervals of $\frac{\pi}{2}$ seconds, creating a highly dynamic and unpredictable control environment. The disturbance is designed to be particularly challenging, applying sudden forces intermittently, which rigorously tests the adaptive capabilities of the MRAC system.

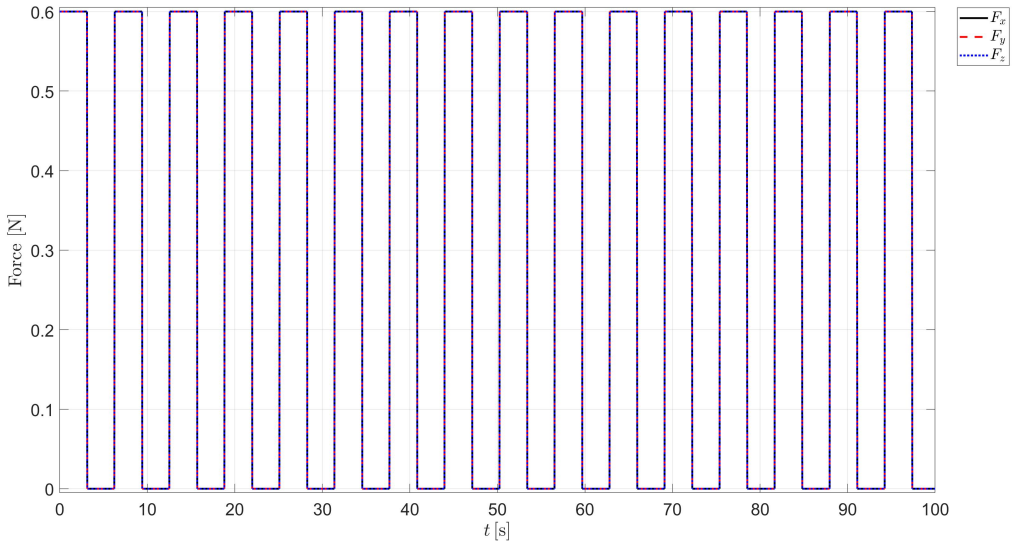

Figure 5.13: Applied discontinuous non-differentiable external disturbance

The mathematical model for this discontinuous disturbance applied across the x, y, and z axes is defined by

$$F_x(t) = 0.6 \cdot \text{step\_function}\left(t, \frac{\pi}{2}\right), \qquad t \geq t_0, \tag{5.4a}$$

$$F_y(t) = 0.6 \cdot \text{step\_function}\left(t, \frac{\pi}{2}\right), \tag{5.4b}$$

$$F_z(t) = 0.6 \cdot \text{step\_function}\left(t, \frac{\pi}{2}\right). \tag{5.4c}$$

where step_function(t, τ) toggles between 0 and 1 every $\tau > 0$ seconds. This setup challenges the MRAC by inducing sharp transitions that must be managed by the controller.

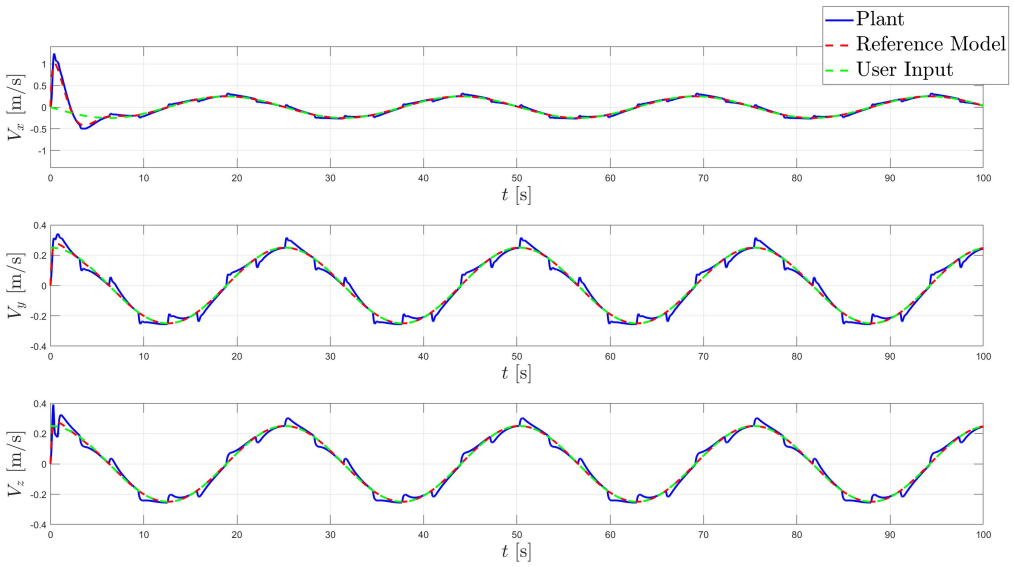

Figure 5.14: Velocity tracking of MRAC under discontinuous non-differentiable disturbance

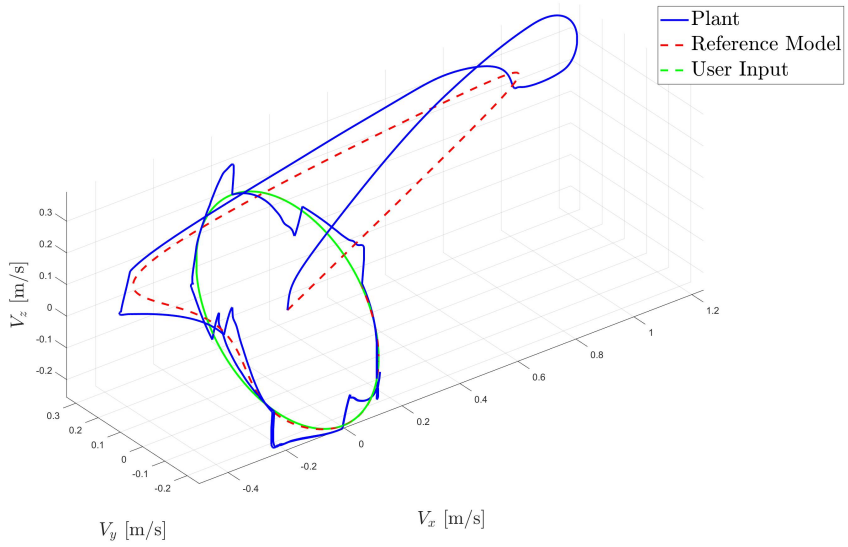

Figure 5.15: Three-dimensional velocity tracking of MRAC under discontinuous non-differentiable disturbance

Velocity tracking during this test exhibits very sharp and erratic fluctuations as shown in Figures 5.14 and 5.15. These fluctuations are illustrative of the difficulties in adjusting control inputs swiftly enough to effectively counteract the abrupt changes induced by the step function.

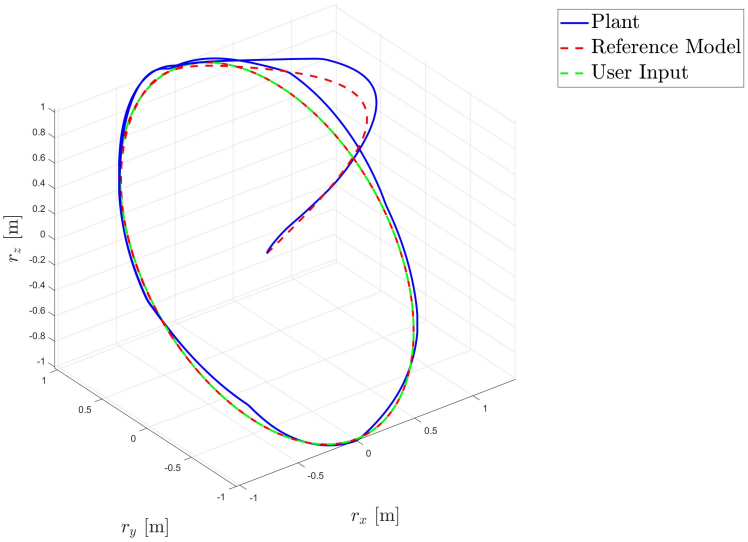

Figure 5.16: Three-dimensional position tracking of MRAC under discontinuous non-differentiable disturbance

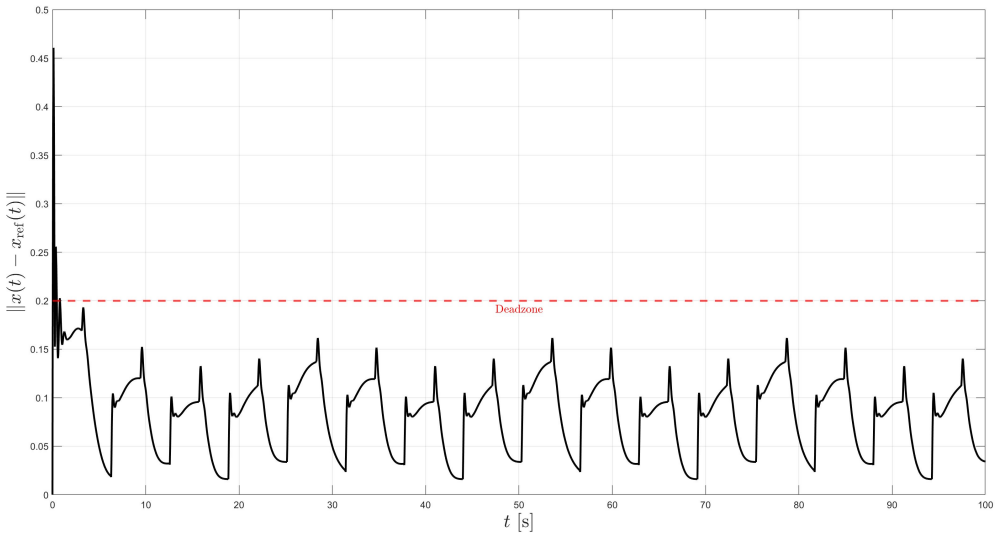

Figure 5.17: Error dynamics in MRAC under discontinuous non-differentiable disturbance

While the system aims to maintain a stable trajectory, the abrupt changes in velocity due to the step disturbances lead to notable deviations from the desired path, emphasizing the MRAC's struggle to maintain accurate course under severe disturbances.

As depicted in Figure 5.17, the error dynamics are far from smooth and exhibit significant fluctuations. The magnitude of the disturbance was selected to challenge the MRAC system allowing us to better distinguish the effects when applying RKHS

methods. The strategic choice of a dead zone width of 0.2 was made because the long-term error consistently stays below this threshold, highlighting the testing rigor and the efficacy of the dead zone setting.

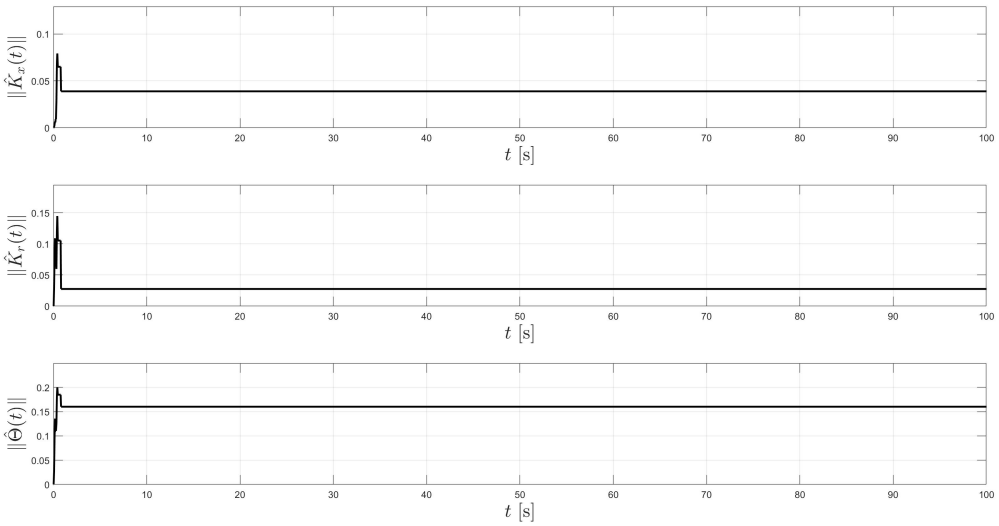

Figure 5.18: Adaptive gain adjustment in MRAC under discontinuous non-differentiable disturbance

The adaptive gains, shown in Figure 5.18, once again respond in a similar manner as with the continuous disturbances.

This rigorous examination not only demonstrates the MRAC's capability to adapt to a wide range of disturbances but also emphasizes the need for precise tuning and possibly more advanced adaptation strategies to further enhance the system's robustness and responsiveness. In the following section, we will compare these performance outcomes with those achieved using embedded RKHS techniques, where we anticipate observing significant improvements in handling such challenging disturbances.

5.2 Performance of RKHS-Embedded Robust MRAC

This section evaluates the enhancements obtained by integrating a RKHS into the robust MRAC framework. The same parameter settings used in Tables 5.1 and 5.2 for the robust MRAC approach are maintained here, as well as the same disturbances defined by (5.1), (5.2), and (5.4). This consistency ensures that the improvements observed are attributable solely to the RKHS implementation.

5.2.1 RKHS Results with Cube-Shaped Basis Center Distribution

Within the RKHS-enhanced MRAC, a cube-shaped basis center distribution is employed to optimize the spatial coverage of the RKHS functions, thus improving the system's response to disturbances. This section discusses the specific configurations and the resulting system performances.

In this RKHS implementation, the adaptive gain, γ_x, γ_y, and γ_z are set to 10^5. The choice of 27 basis centers at distance of 0.4 along the sides of the latex in the velocity space was strategically made to optimize computational efficiency and enhance system performance. This configuration was selected because it provided the best balance between computational demands and performance improvement, as additional centers beyond 27 offered only marginal performance gains.

5.2.1.1 Performance with Continuous Differentiable Disturbance

This section examines the system's response under the same continuous differentiable disturbance previously tested with the robust MRAC, referenced in (5.1).

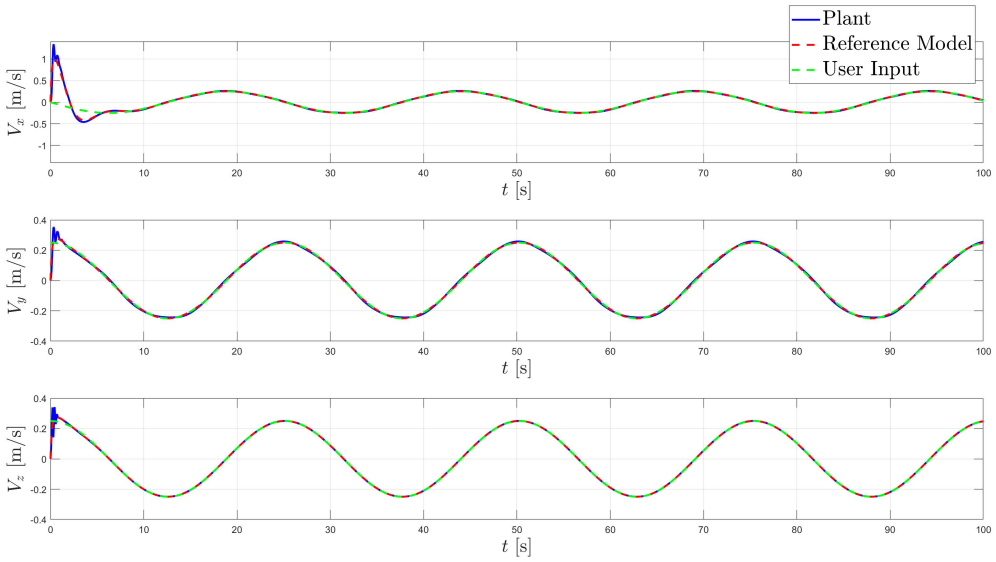

Figure 5.19: Velocity tracking of RKHS-embedded MRAC with cube-shaped basis center distribution under continuous differentiable disturbance

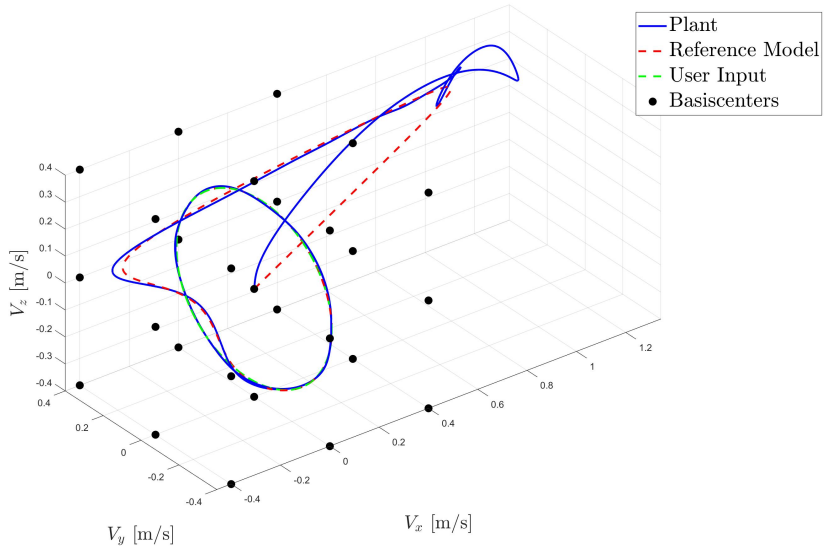

Figure 5.20: Three-dimensional velocity tracking of RKHS-embedded MRAC with cube-shaped basis center distribution under continuous differentiable disturbance

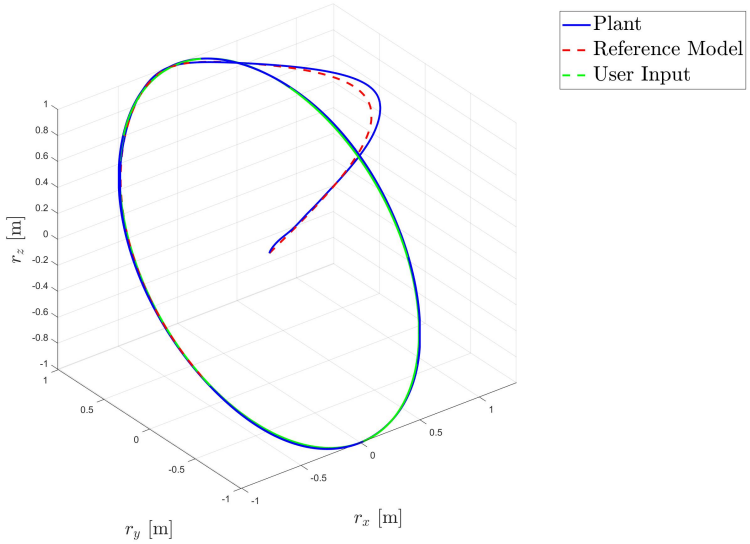

Figure 5.21: Three-dimensional position tracking of RKHS-embedded MRAC with cube-shaped basis center distribution under continuous differentiable disturbance

Initially, the RKHS-enhanced MRAC system displays somewhat erratic behavior due to the non-parametric nature of the RKHS learning phase. This is observable in Figure 5.19, where the tracking velocity oscillates rapidly early on. Over time, as the RKHS framework learns and optimizes the kernel functions, the system's ability to accurately track velocity significantly improves. The combination of RKHS with a baseline controller and standard MRAC components helps stabilize the system's response early on, allowing for effective learning and subsequent fine-tuning to dynamic changes. Figure 5.25 showcases the RKHS-enhanced MRAC system's improved tracking of complex three-dimensional trajectories. The RKHS integration smoothens the flight path and significantly reduces deviations, easily handling the continuous differentiable disturbance as seen in Figure 5.21.

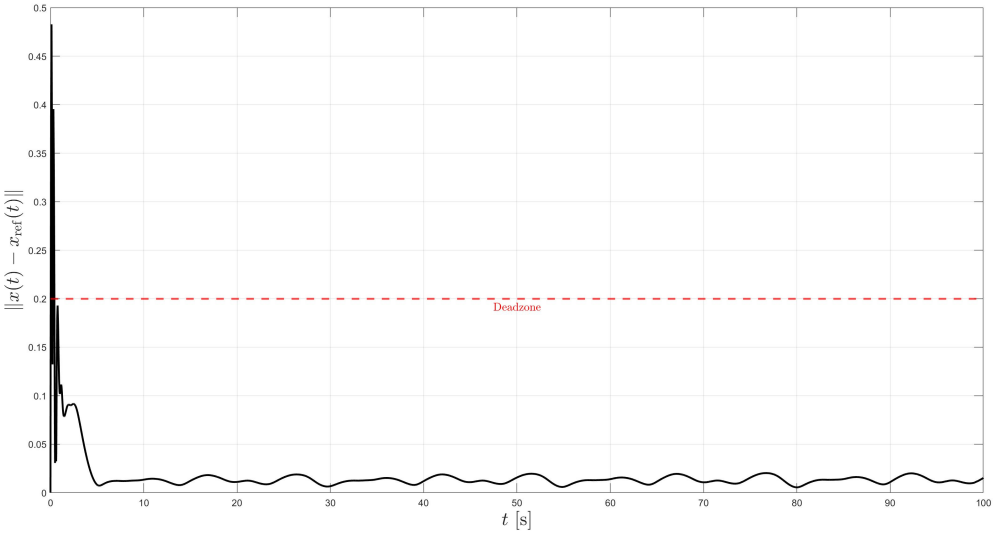

Figure 5.22: Error dynamics of RKHS-embedded MRAC with cube-shaped basis center distribution under continuous differentiable disturbance

Figure 5.22 presents the tracking error dynamics for the MRAC system enhanced by the integration of an RKHS. While initial error spike is higher in the RKHS-enhanced configuration due to its non-parametric learning phase, the integration of RKHS significantly improves the tracking error over time. This system achieves a lower steady state error and a smoother error profile compared to classical MRAC. The dual-controller setup, which merges the adaptive capabilities of RKHS with the stability of a baseline MRAC, effectively mitigates the erratic behavior typically seen in pure non-parametric controls. This combination not only quickens the convergence of error but also stabilizes it, minimizing overshoot and enhancing settling times.

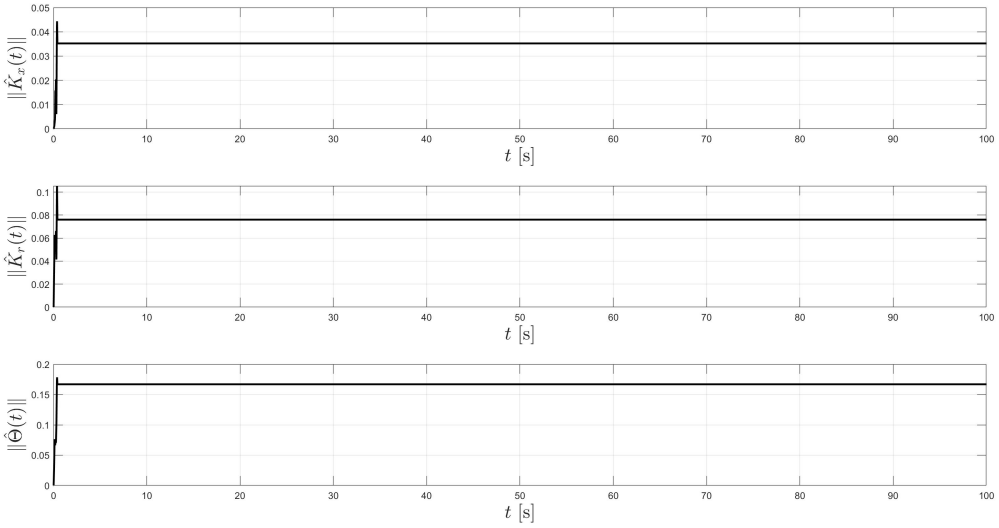

Figure 5.23: Adaptive gain adjustment of RKHS-embedded MRAC with cube-shaped basis center distribution under continuous differentiable disturbance

Figure 5.23 demonstrates the adaptive gains within the RKHS-enhanced MRAC system. The inclusion of RKHS provides supplementary support to the classical MRAC controller, leading to a noticeable reduction in the magnitude of the adaptive gains. This support allows the adaptive gains to increase at a slower pace, resulting in a more gradual and controlled adaptation process. The combined operation of RKHS with the classical MRAC components helps stabilize the system's performance from the outset, making the adaptation process more effective and reducing the system's initial response volatility.

In the subsequent sections, we will analyze the overall performance with an applied continuous non-differentiable disturbance.

5.2.1.2 Performance with Continuous Non-differentiable Disturbance

This section examines the RKHS-embedded MRAC system's performance under the same continuous non-differentiable disturbance applied in the classical MRAC simulation, as defined by (5.2). This analysis highlights the system's robustness in managing disturbances that lack smooth derivatives, emphasizing the adaptability and effectiveness of the RKHS methodology.

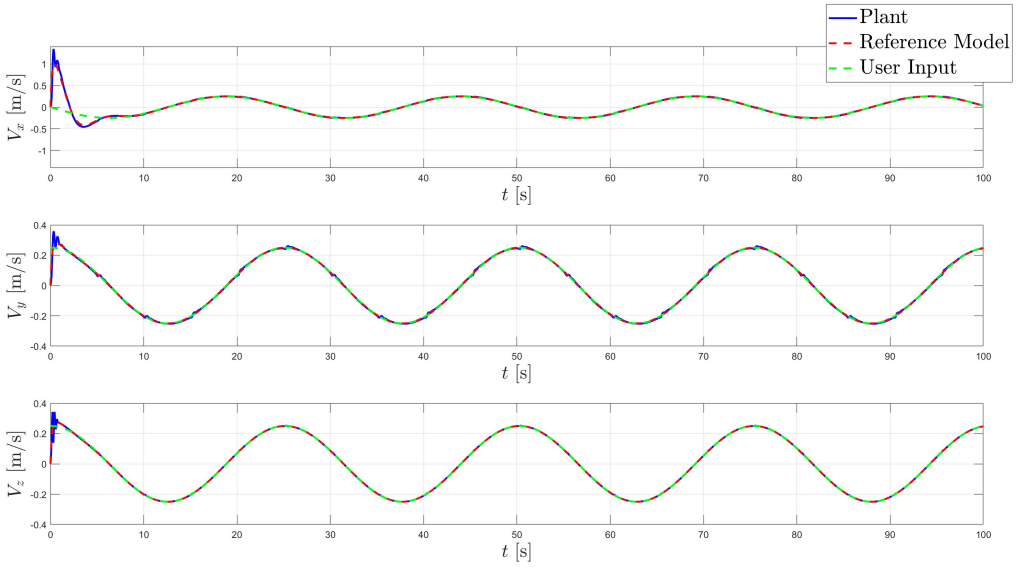

Figure 5.24: Velocity tracking of RKHS-embedded MRAC with cube-shaped basis center distribution under continuous non-differentiable disturbance

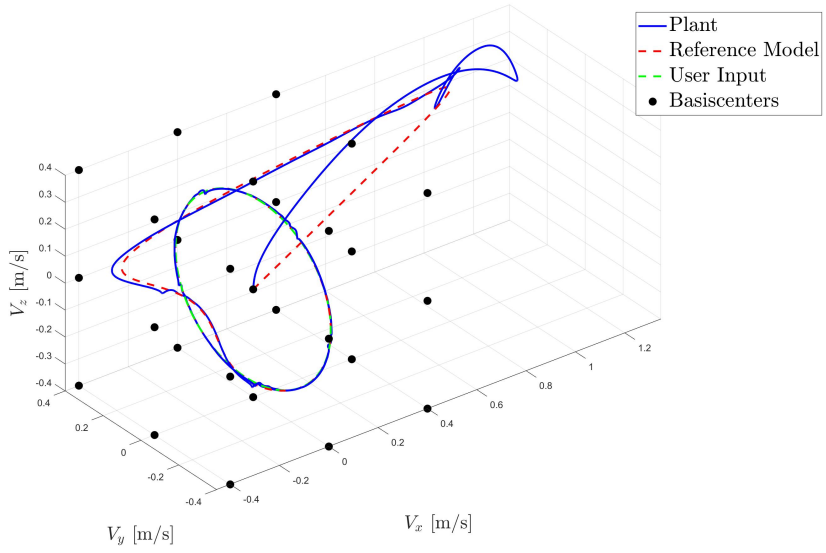

Figure 5.25: Three-dimensional velocity tracking of RKHS-embedded MRAC with cube-shaped basis center distribution under continuous non-differentiable disturbance

The RKHS-enhanced MRAC exhibits excellent trajectory tracking capabilities, similar to those observed with the smooth differentiable function. Despite the complexity of the non-differentiable disturbance, the system maintains high performance levels. Figures 5.24 and 5.25 illustrate the near-perfect adherence to the desired trajectories. This consistent performance underlines the RKHS's ability to effectively handle complex unmatched uncertainties.

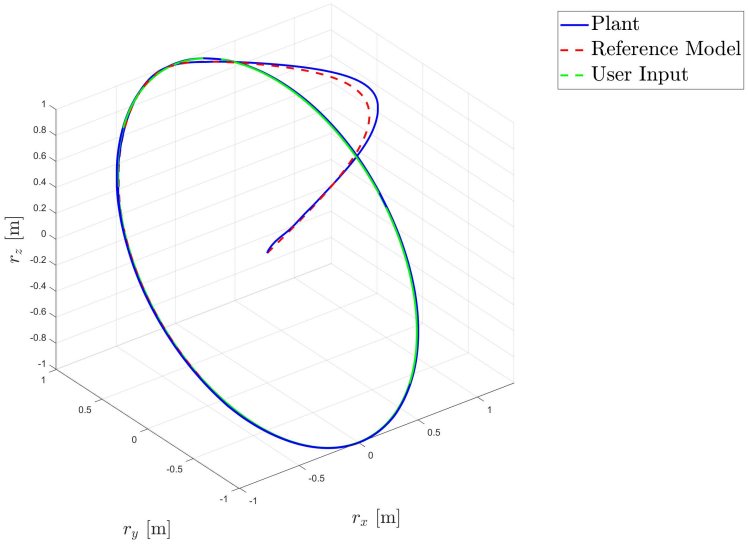

Figure 5.26: Three-dimensional MRAC Position Tracking with Implemented RKHS under Continuous Non-differentiable Disturbance

Figure 5.26 further confirms these results, showing the system's position tracking that closely follows the prescribed path with minimal deviations. This robust position tracking, even under challenging disturbance conditions, is indicative of the sophisticated control dynamics enabled by RKHS.

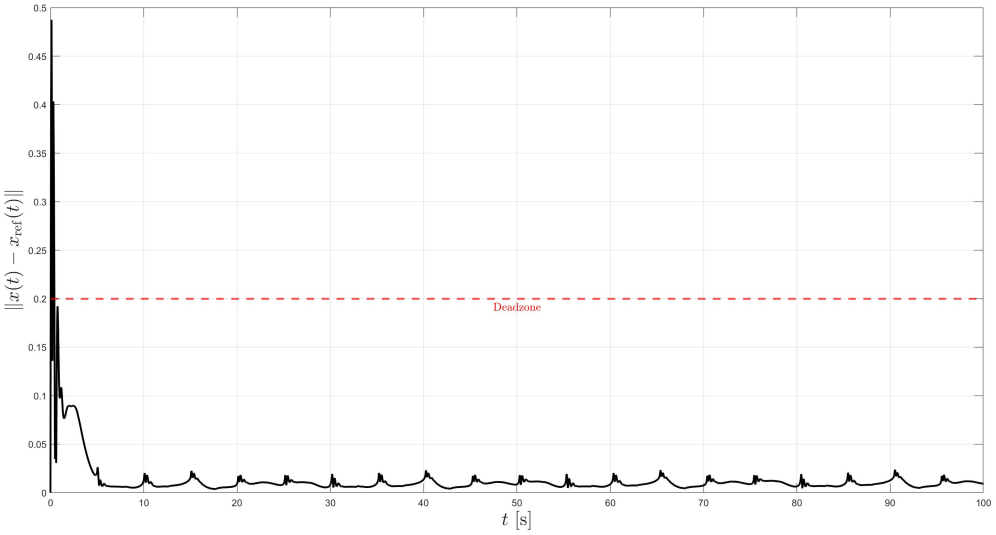

Figure 5.27: MRAC Tracking Error with Implemented RKHS under Continuous Non-differentiable Disturbance

The error dynamics, illustrated in Figure 5.27, while showcasing a lower error profile than the robust MRAC alone, reveal that the steady-state error is not as smooth as that observed under the continuous differentiable disturbance. However, the RKHS's capability to effectively manage these errors, even under non-differentiable conditions, is evident and reinforces the system's adaptability and precision.

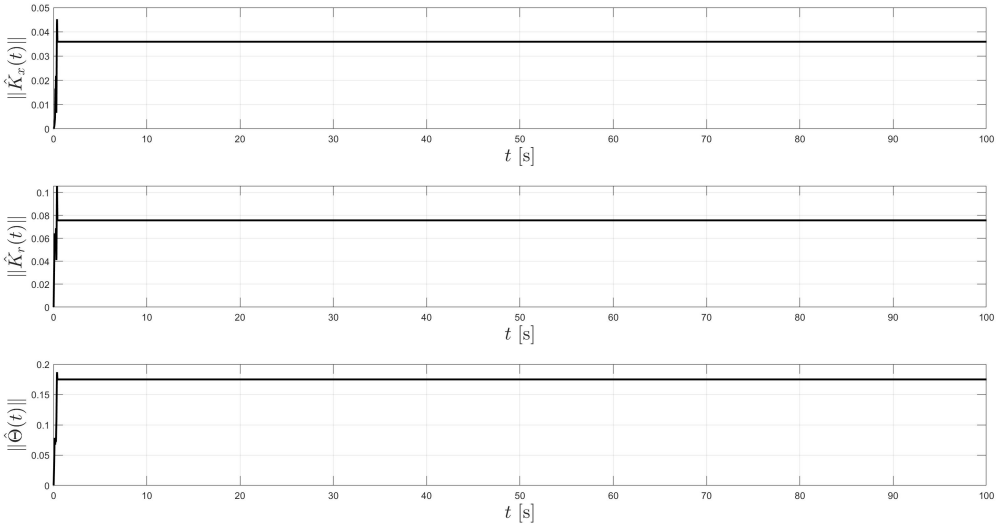

Figure 5.28: Adaptive gain adjustment in MRAC with implemented RKHS under continuous non-differentiable disturbance

The adaptive gains, as seen in Figure 5.28, adjust in a manner very similar to that observed under the RKHS-embedded continuous differentiable disturbance scenarios. These gains increase more gradually than what is seen under robust MRAC, benefiting from the RKHS's moderation, which not only stabilizes their growth but also enhances the overall stability and effectiveness of the adaptive control process.

These findings demonstrate the RKHS's profound capability to handle non-differentiable functions effectively, thereby contributing to enhanced system performance and robustness across varying disturbance types. Next, we will explore how the system performs under a discontinuous non-differentiable disturbance, further challenging the RKHS-enhanced MRAC with abrupt changes and severe disruptions.

5.2.1.3 Performance with Discontinuous Non-differentiable Disturbance

This section assesses the RKHS-embedded MRAC system's performance when subjected to a discontinuous non-differentiable disturbance defined by (5.4), a more challenging scenario that tests the limits of the RKHS's adaptability. This type of disturbance, as previously defined, diverges significantly from the typical operational scenarios anticipated within the native space of the RKHS, posing substantial challenges in system response and error correction.

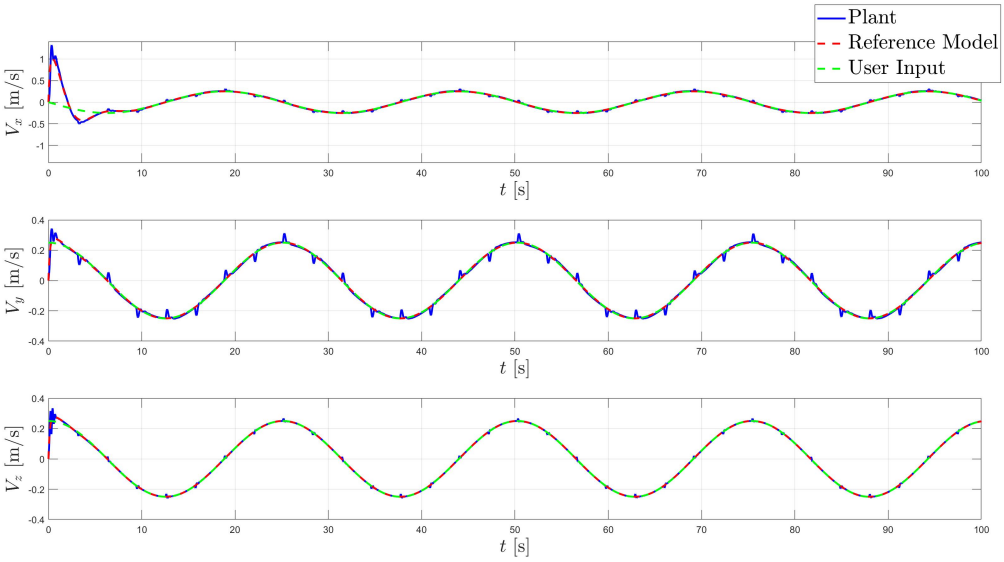

Figure 5.29: Velocity tracking of RKHS-embedded MRAC with cube-shaped basis center distribution under discontinuous non-differentiable disturbance

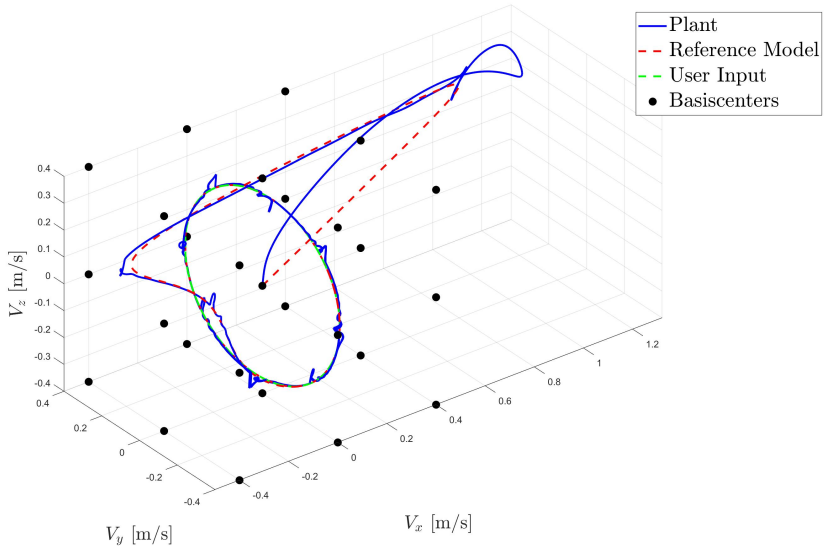

Figure 5.30: Three-dimensional velocity tracking of RKHS-embedded MRAC with cube-shaped basis center distribution under discontinuous non-differentiable disturbance

Figures 5.29 and 5.30 show clear deviations from the desired trajectories, which were not observed in previous RKHS-embedded test cases under smoother disturbances. This deviation indicates that the RKHS struggles with the discontinuous nature of the disturbance, likely because these conditions are far from the kernel's native space, making it challenging for RKHS to effectively handle.

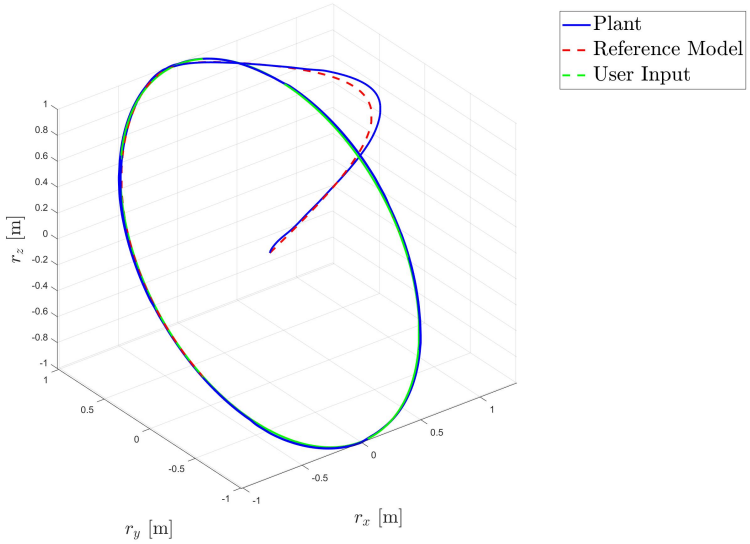

Figure 5.31: Three-dimensional position tracking of RKHS-embedded MRAC with cube-shaped basis center distribution under discontinuous non-differentiable disturbance

Despite these challenges, the RKHS-embedded system still shows substantial improvements over the classical robust MRAC setup. As illustrated in Figure 5.31, while the system exhibits some initial instability, it quickly regains its tracking accuracy, demonstrating the dynamic adaptability of RKHS.

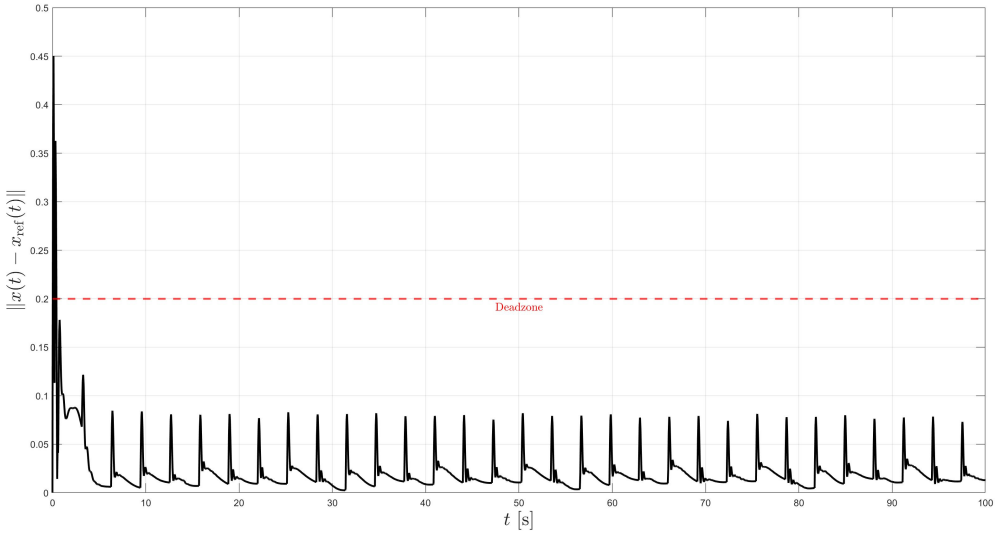

Figure 5.32: Error dynamics of RKHS-embedded MRAC with cube-shaped basis center distribution under discontinuous non-differentiable disturbance

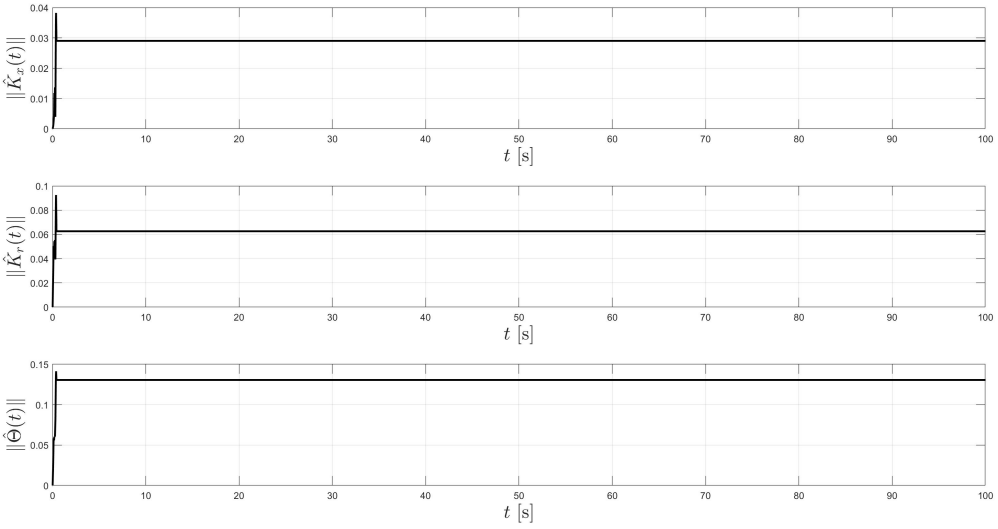

Figure 5.33: Adaptive gain adjustment of RKHS-embedded MRAC with cube-shaped basis center distribution under discontinuous non-differentiable disturbance

Figure 5.32 further highlights the error dynamics under this disturbance. The error profile, although spiking at the onset of each force, stabilizes much quicker compared to the robust MRAC alone. This faster stabilization emphasizes the RKHS's efficacy in quickly adapting to sudden changes, maintaining a consistently lower error level across the disturbance timeline.

As shown in Figure 5.33, the adaptive gains in this scenario are comparable to those observed under continuous disturbances with RKHS, reflecting the system's robustness in gain management despite the increased disturbance complexity.

These findings underscore the RKHS's potential to enhance MRAC systems even under severe and abrupt disturbance conditions. The next section will explore the performance implications of employing a torus basis center distribution instead of the cubic lattice, aiming to determine if different spatial arrangements of basis centers could further optimize the RKHS's performance under such demanding conditions.

5.2.2 RKHS Results with Torus-Shaped Basis Center Distribution

This subsection examines the performance of the RKHS-embedded MRAC system using a torus-shaped basis center distribution, maintaining the same parameter settings as used in Tables 5.1 and 5.2. The adaptive gains, γ_x, γ_y, and γ_z remain set at 10^5, consistent with the previous cubic lattice basis center distribution case, ensuring comparability. The number of basis centers is also maintained at 27, strategically arranged in nine rings in the velocity space at an angular distance of 120 deg and radius of 0.05 along the user-defined velocity, each containing three centers, to facilitate a direct comparison of different geometrical configurations under the same operational conditions. The disturbances applied are identical to those in previous sections, defined by (5.1), (5.2), and (5.4), to systematically assess the impact of basis center distribution on system performance.

5.2.2.1 Performance with Continuous Differentiable Disturbance

This section delves into the RKHS-enhanced MRAC's response to a continuous differentiable disturbance using a torus-shaped distribution of basis centers, investigating how this configuration influences system dynamics compared to the cubic lattice distribution.

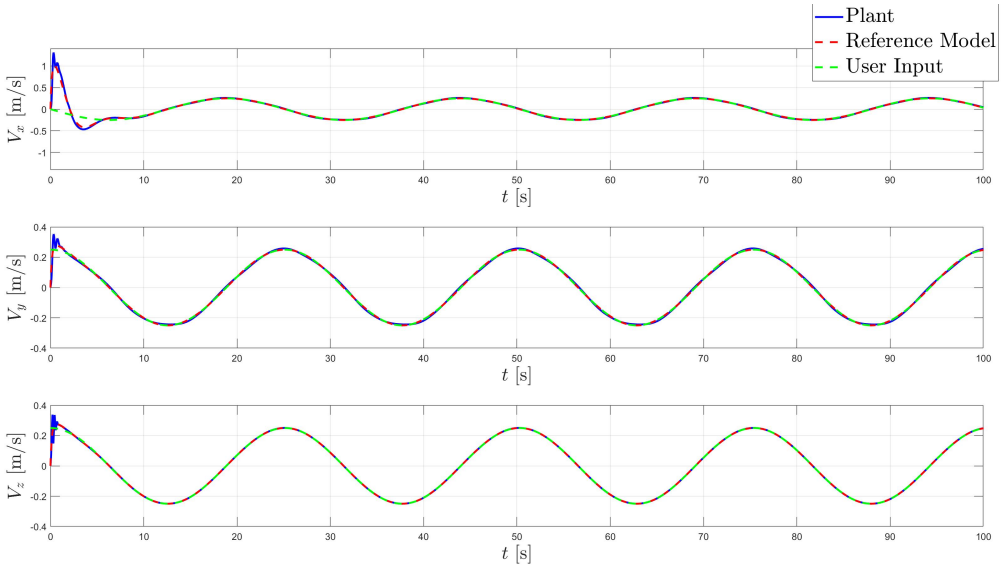

Figure 5.34: Velocity tracking of RKHS-embedded MRAC with torus-shaped basis center distribution under continuous differentiable disturbance

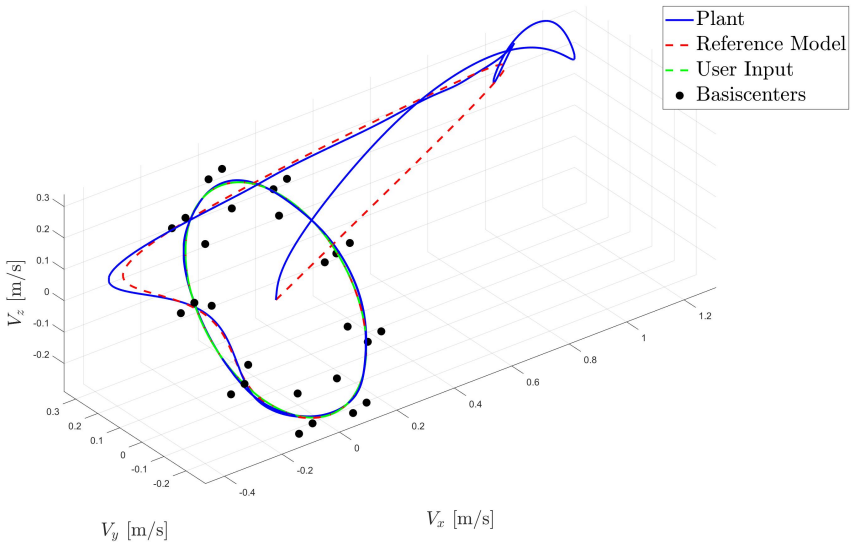

Figure 5.35: Three-dimensional velocity tracking of RKHS-embedded MRAC with torus-shaped basis center distribution under continuous differentiable disturbance

Figures 5.34 and 5.35 illustrate the system's velocity tracking capabilities. The performance metrics are nearly identical to those observed with the cubic basis center distribution, highlighting that the RKHS's efficacy in handling continuous differentiable disturbances is robust across different spatial arrangements of the basis centers.

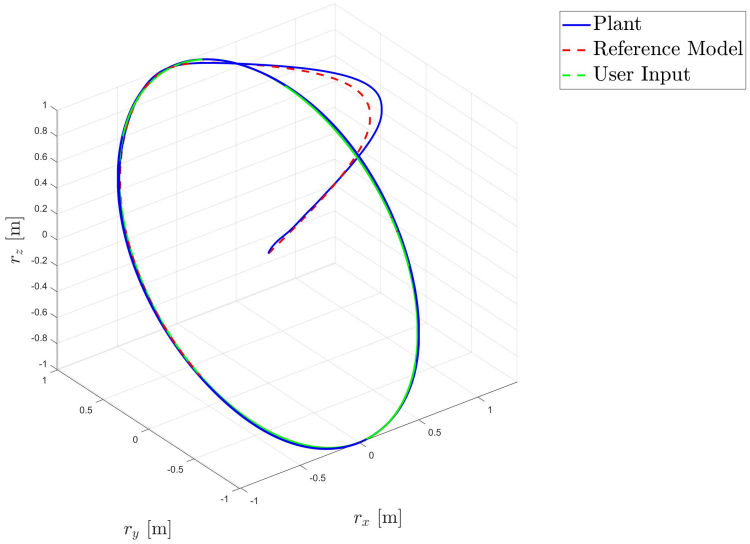

Figure 5.36: Three-dimensional position tracking of RKHS-embedded MRAC with torus-shaped basis center distribution under continuous differentiable disturbance

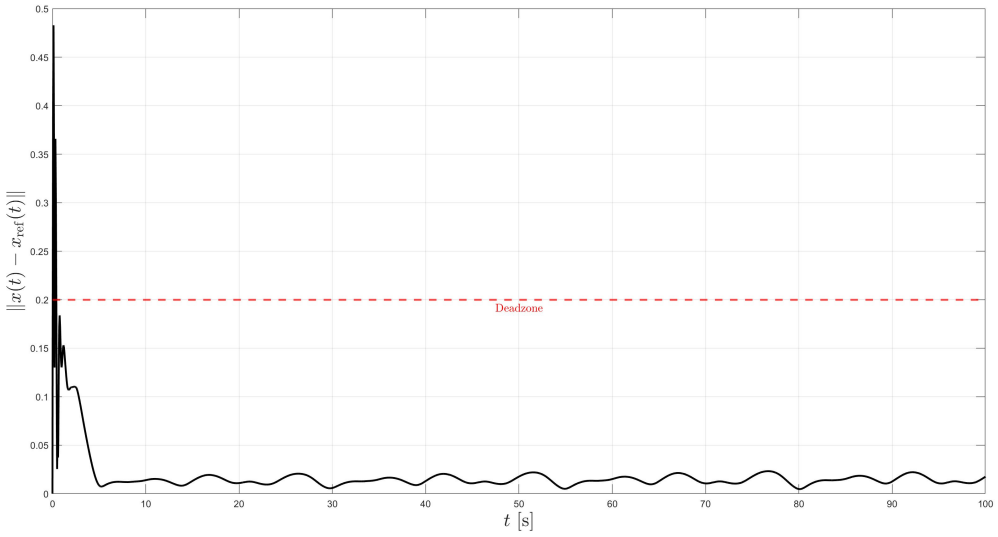

Figure 5.37: Error dynamics of RKHS-embedded MRAC with torus-shaped basis center distribution under continuous differentiable disturbance

Figure 5.36 confirms consistent performance in position tracking, with the system maintaining precise control over its trajectory, similar to the cubic configuration. However, as shown in Figure 5.37, while the error dynamics are significantly improved compared to the robust MRAC alone, the cubic lattice basis center distribution slightly outperforms the torus configuration in terms of minimizing steady-state error fluctuations. This slight variance might be attributed to the spatial alignment and density of

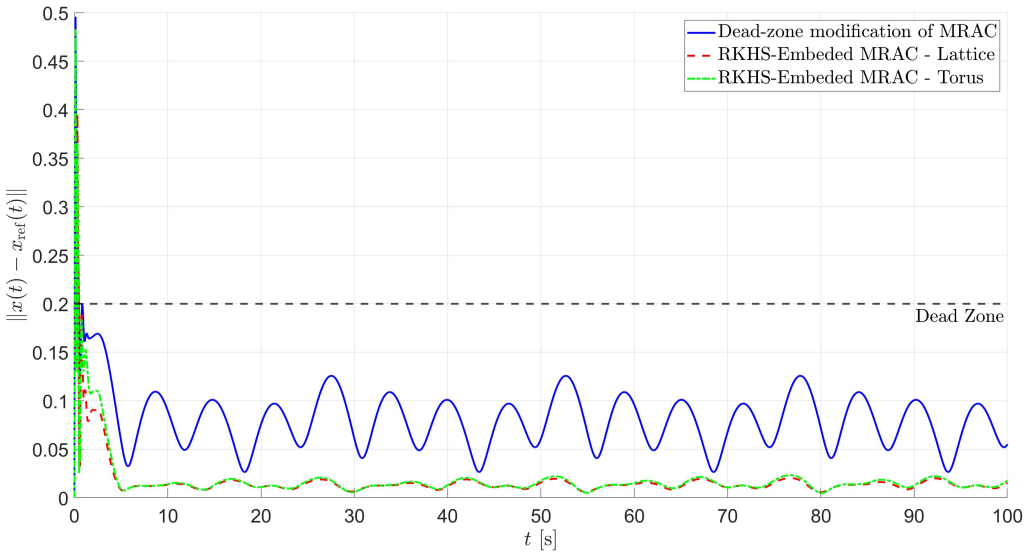

Figure 5.38: Comparative error dynamics across multiple control strategies under continuous differentiable disturbance

the basis centers in the cubic lattice potentially providing a more effective coverage and interaction within the RKHS's native space.

Figure 5.38 displays a comprehensive comparison of error dynamics across different control strategies. This graph underscores the substantial improvement offered by RKHS embedding in reducing tracking errors, regardless of the basis center distribution type, with the cubic distribution showing a marginally better performance.

In the next section, we will explore the system's performance with a torus basis center distribution under continuous non-differentiable disturbances, further challenging the RKHS-enhanced MRAC with abrupt changes and severe disruptions.

5.2.2.2 Performance with Continuous Non-differentiable Disturbance

This section continues our examination of the RKHS-embedded MRAC system, now focusing on its response to continuous non-differentiable disturbances as previously represented by (5.2). Similar to earlier assessments, this analysis aims to validate the robustness and adaptability of the RKHS implementation using a torus-shaped basis center distribution.

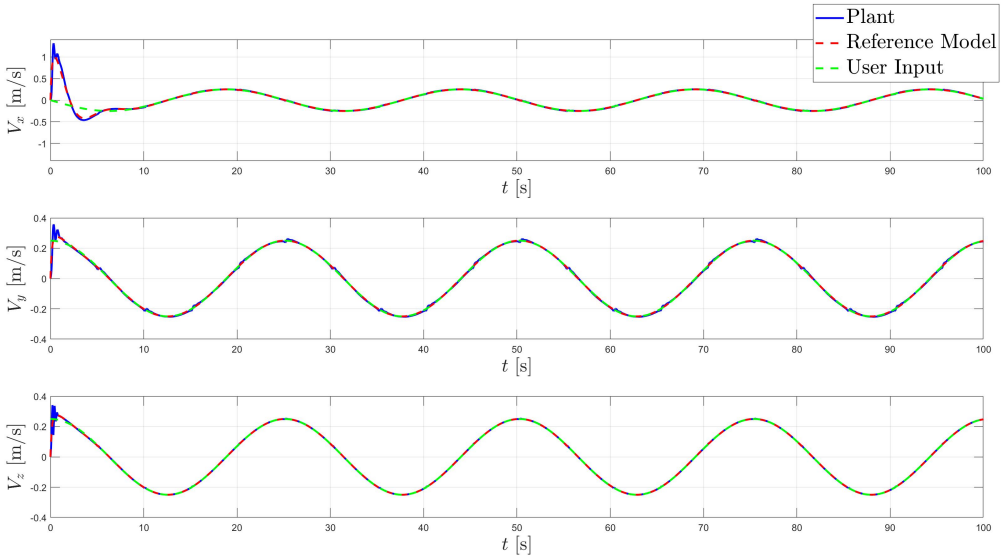

Figure 5.39: Velocity tracking of RKHS-embedded MRAC with torus-shaped basis center distribution under continuous non-differentiable disturbance

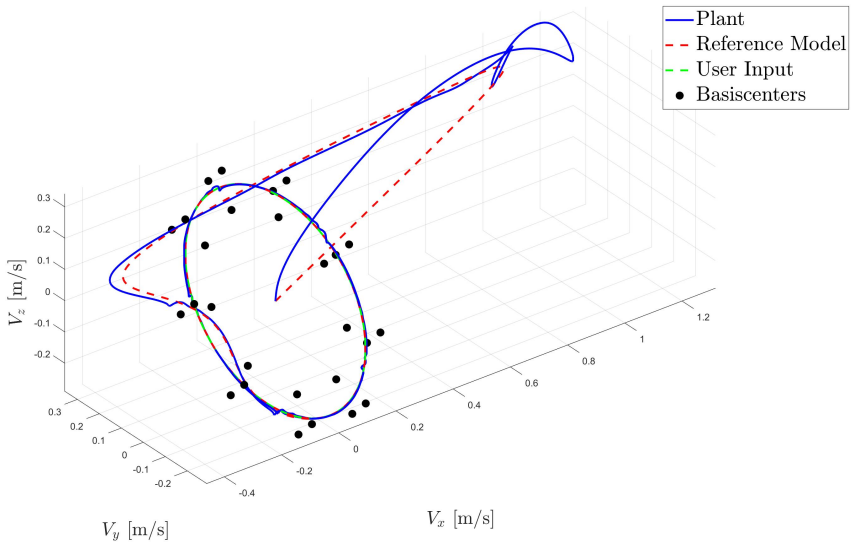

Figure 5.40: Three-dimensional velocity tracking of RKHS-embedded MRAC with torus-shaped basis center distribution under continuous non-differentiable disturbance

Figures 5.39 and 5.40 demonstrate the velocity tracking capabilities of the MRAC system when enhanced with RKHS using the torus distribution. The performance is observed to be very similar to that seen with the cubic lattice basis center distribution, with the system maintaining excellent trajectory adherence despite the complex nature of the disturbance.

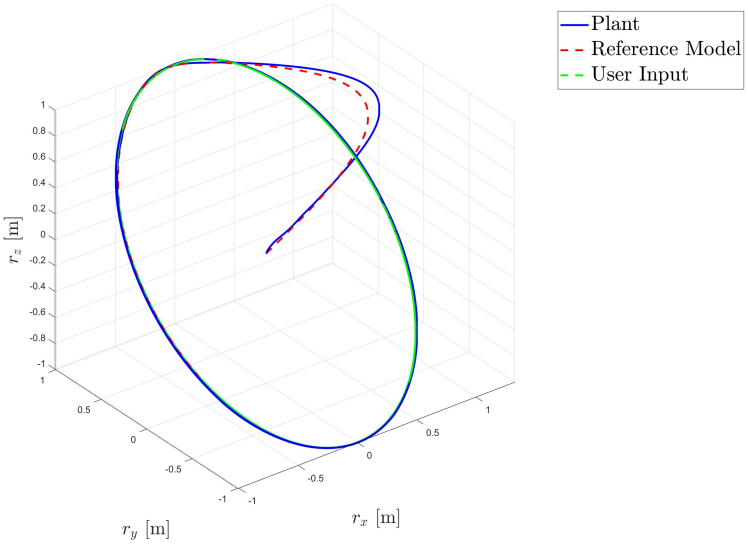

Figure 5.41: Three-dimensional position tracking of RKHS-embedded MRAC with torus-shaped basis center distribution under continuous non-differentiable disturbance

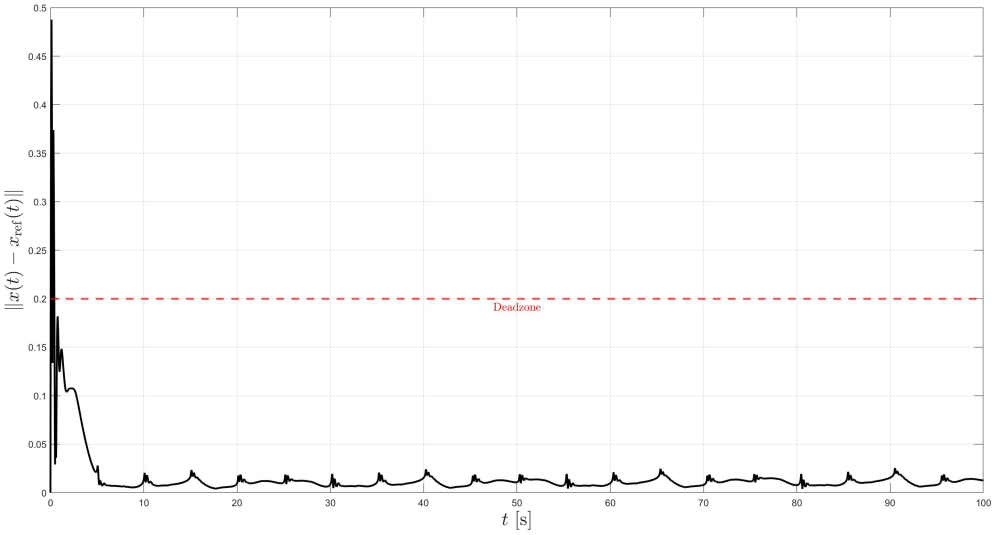

Figure 5.42: Error dynamics of RKHS-embedded MRAC with torus-shaped basis center distribution under continuous non-differentiable disturbance

Figure 5.41 confirms that the three-dimensional position tracking of the system aligns closely with the expected trajectory. This accuracy in positioning, even under challenging disturbance conditions, underscores the RKHS's ability to effectively manage dynamic changes and uncertainties.

The error dynamics, shown in Figure 5.42, illustrate that while the error profile is

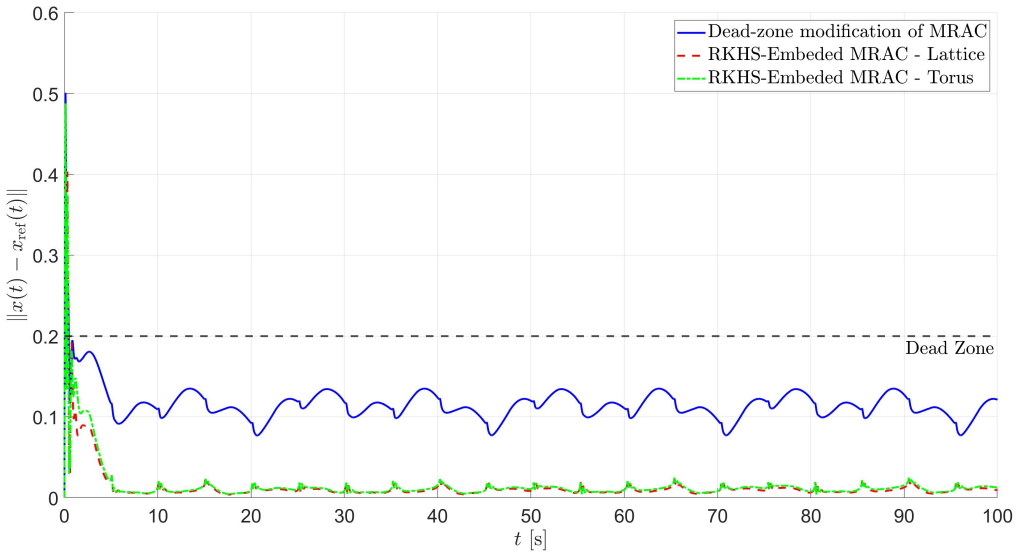

Figure 5.43: Comparative error dynamics across multiple control strategies under continuous non-differentiable disturbance

almost identical to that observed in the cubic lattice configuration, it is substantially lower than that seen with the robust MRAC approach alone. Although the error spikes at the onset of the disturbance, it quickly stabilizes, underscoring the quick corrective capabilities of the RKHS-enhanced system.

Figure 5.43 presents a comparative view of error dynamics across different control strategies. Despite minor variances in the performance, the RKHS embedding substantially enhances the error management capabilities of MRAC systems, regardless of the basis center distribution type. Interestingly, the cubic basis center distribution exhibits slightly better performance, potentially due to more effective spatial coverage and interaction within the RKHS's native space.

These results affirm the RKHS's profound capability to handle non-differentiable functions effectively, contributing to enhanced system performance and robustness across varying disturbance types. In the next section, we will review the system's performance under a discontinuous non-differentiable disturbance, further challenging the RKHS-enhanced MRAC with abrupt changes and severe disruptions.

5.2.2.3 Performance with Discontinuous Non-differentiable Disturbance

This section explores the RKHS-embedded MRAC system's response under discontinuous non-differentiable disturbances as defined by (5.4). Using a torus-shaped basis center distribution, we assess whether the geometric configuration impacts the system's ability to handle abrupt changes in the control environment.

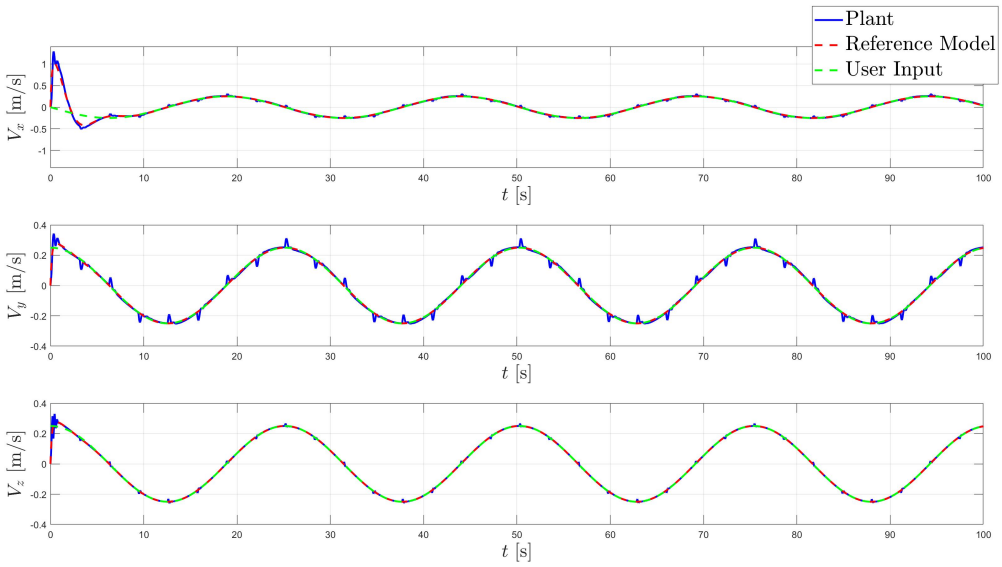

Figure 5.44: Velocity tracking of RKHS-embedded MRAC with torus-shaped basis center distribution under discontinuous non-differentiable disturbance

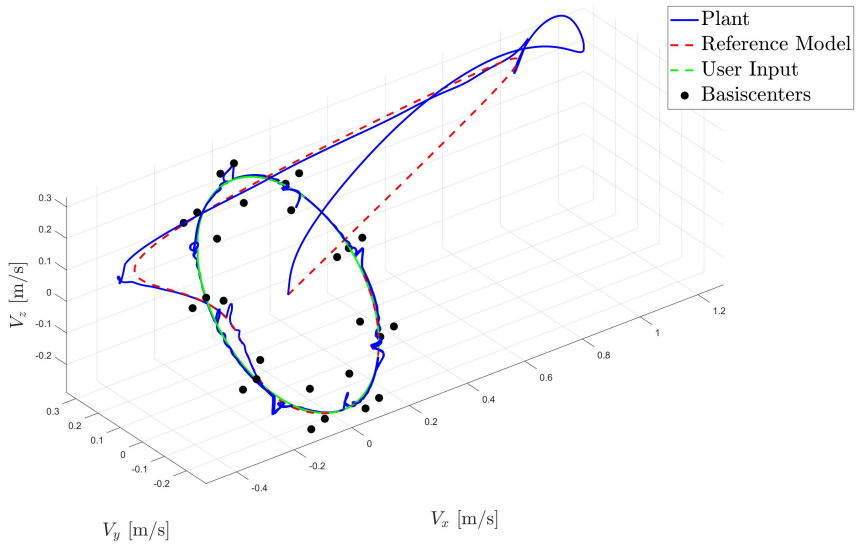

Figure 5.45: Three-dimensional velocity tracking of RKHS-embedded MRAC with torus-shaped basis center distribution under discontinuous non-differentiable disturbance

Figures 5.44 and 5.45 depict the system's velocity tracking, showing robust performance that mirrors the results seen with the cubic lattice basis center distribution. The system manages to maintain trajectory control despite the disruptive nature of the disturbance.

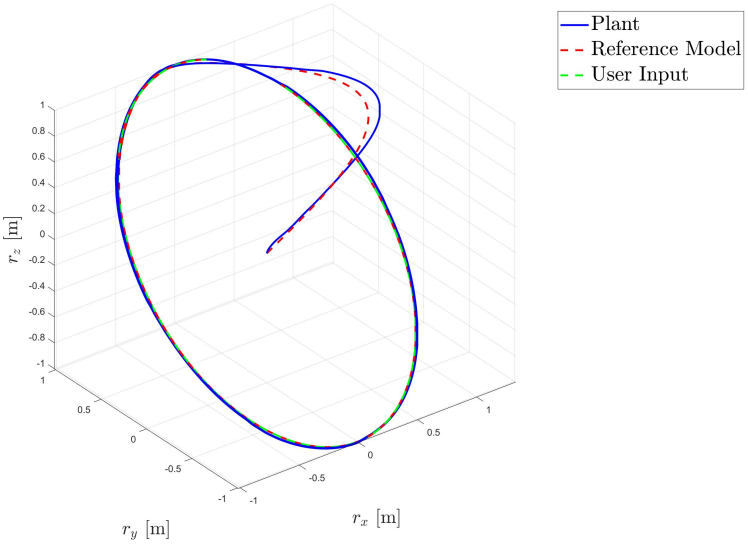

Figure 5.46: Three-dimensional position tracking of RKHS-embedded MRAC with torus-shaped basis center distribution under discontinuous non-differentiable disturbance

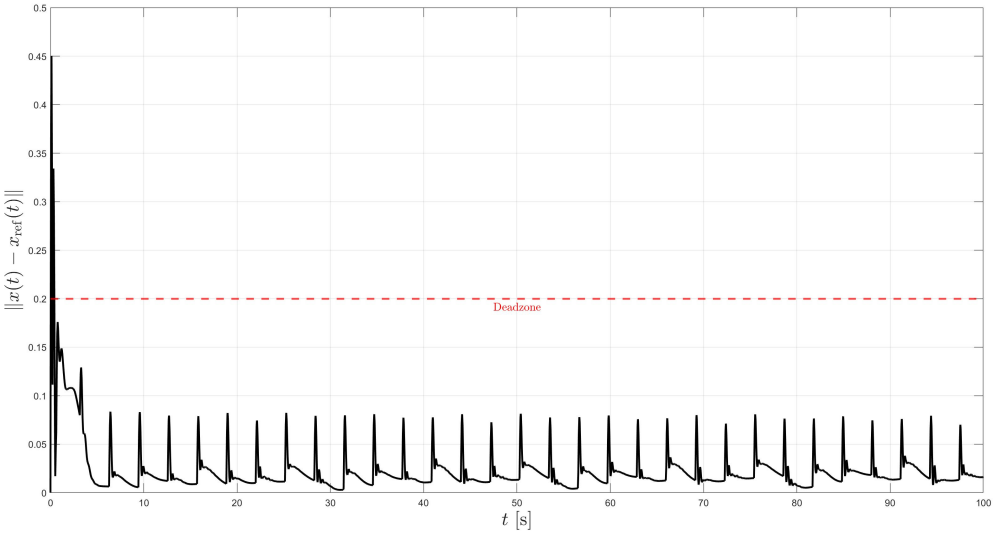

Figure 5.47: Error dynamics of RKHS-embedded MRAC with torus-shaped basis center distribution under discontinuous non-differentiable disturbance

Figure 5.46 further supports the conclusion that the RKHS's control dynamics are effectively adapted to the torus distribution, as the position tracking closely adheres to the desired path with minimal deviations.

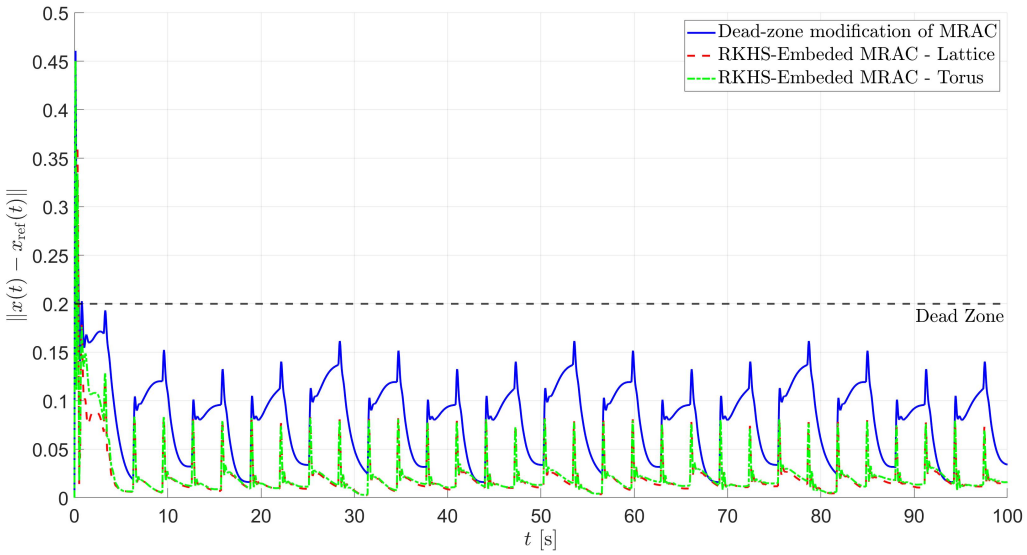

Figure 5.48: Comparative error dynamics across multiple control strategies under discontinuous non-differentiable disturbance

The error dynamics, presented in Figures 5.47 and 5.48, reveal that the torus basis center distribution shows a lower peak error each time the step function is applied, but also exhibits a slightly higher minimum error compared to the cubic lattice. This variance might suggest that while the torus distribution can quickly adapt to sudden disturbances, its configuration could slightly delay the error stabilization phase, potentially due to the spatial distribution and interaction of the basis centers within the RKHS's native function space.

These results not only confirm the effectiveness of RKHS embedding in enhancing MRAC systems under challenging conditions but also illustrate subtle differences influenced by the choice of basis center distribution. In the upcoming section, we will delve into the computational cost results and the impact of additional basis centers or adjustments to the fill distance on performance, exploring how these factors influence the overall efficacy and efficiency of the RKHS-enhanced MRAC system.

5.3 Computational Costs and Performance Metrics of RKHS

This section delves into the computational aspects of employing RKHS in robust MRAC systems. We discuss how the number of basis centers, the fill distance, and their effects influence computational efficiency, normalized run time, and the accuracy of the control system under various disturbances.

5.3.1 Run Time Analysis

Normalized run time, calculated as the ratio of the actual simulation completion time to the predefined simulation length, serves as a crucial metric to assess the computational load of the MRAC system enhanced with RKHS. This normalization facilitates a direct comparison of run times across different system configurations and simulation scales.

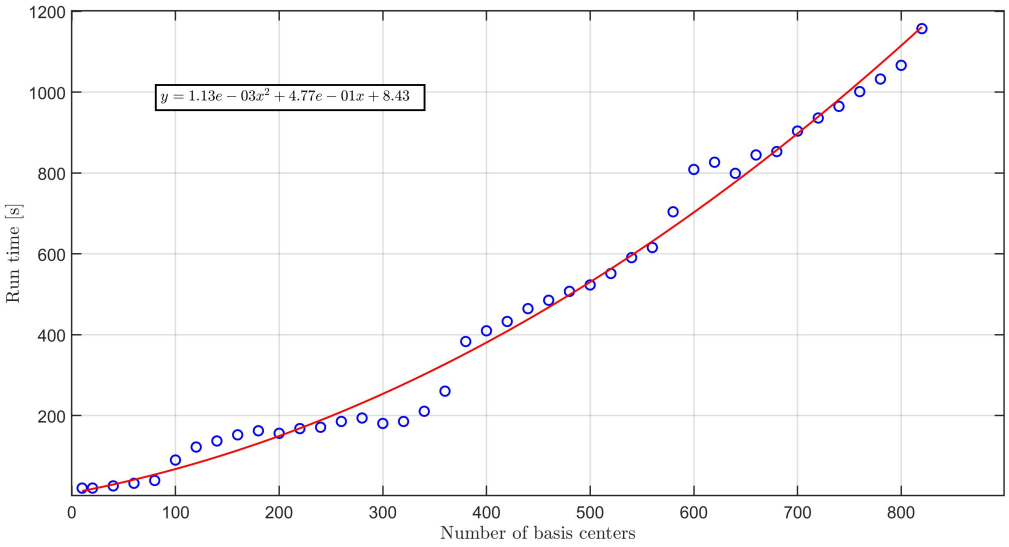

Figure 5.49: Run-time vs. number of basis centers

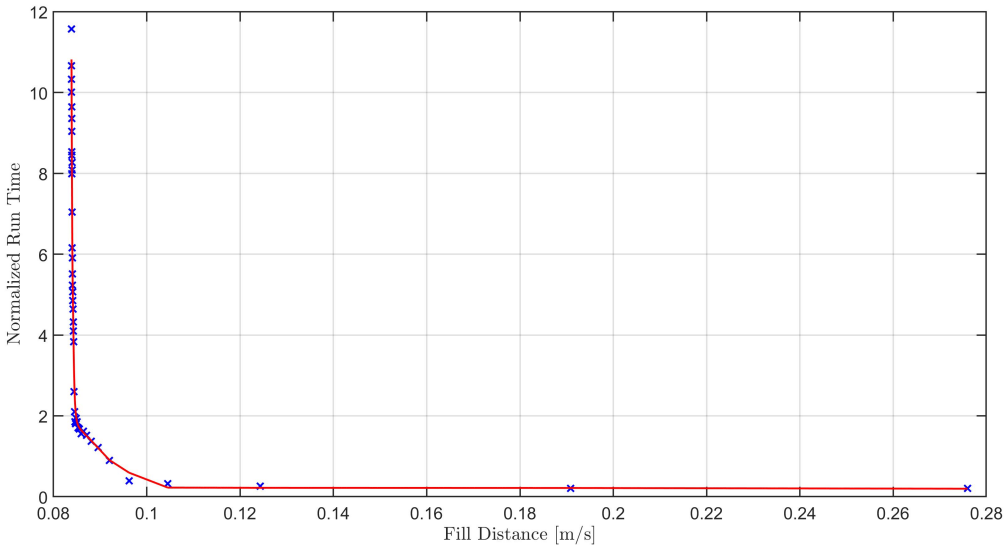

Figure 5.50: Impact of fill distance on normalized run time within the RKHS framework.

Figure 5.49 illustrates that the normalized run time increases exponentially as more basis centers are added to the RKHS configuration. This exponential increase stems from the augmented dimensionality of the Hilbert space required to be computed, significantly intensifying the processing load. The computational complexity grows because each additional basis center incrementally contributes to the RKHS's ability to approximate complex functions more accurately.

As depicted in Figure 5.50, decreasing the fill distance, which measures the maximal

distance between any domain point and its nearest basis center, results in a reduction in normalized run time. A shorter fill distance enhances the RKHS's approximation properties, improving computational efficiency and responsiveness of the control system. Nonetheless, achieving a smaller fill distance often requires a larger number of basis centers, thus increasing the overall computational demands.

5.3.2 Steady State Error Analysis

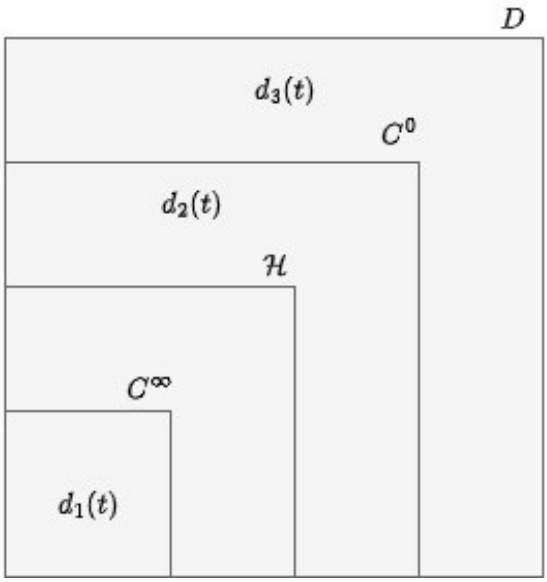

Figure 5.51: RKHS native space

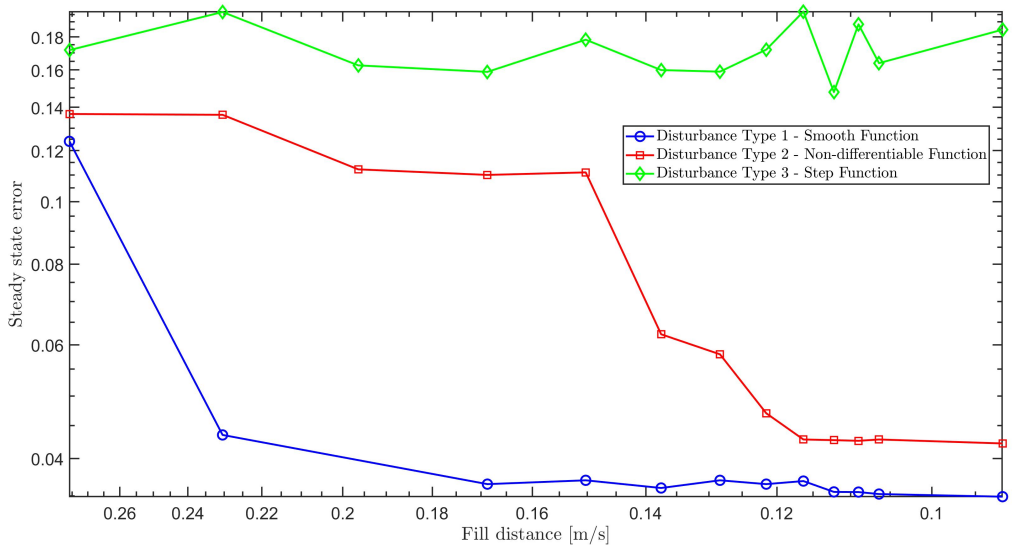

Figure 5.52: Normalized steady state error vs. fill distance

Figure 5.52 provides an in-depth analysis of steady state errors across different disturbance types including smooth differentiable functions, continuous but non-differentiable functions, and discontinuous step functions. These distinctions are crucial as they impact the RKHS controller's effectiveness based on the proximity of these functions to the RKHS's native space.

For smooth differentiable disturbances, which closely align with the RKHS's native space characteristics of smoothness and continuity, the addition of basis centers significantly enhances performance initially. This is because such disturbances can be effectively approximated within the RKHS, allowing rapid error reduction. However, the benefit of adding additional basis centers quickly diminishes, as minimal basis centers are needed to fully represent the function.

In the case of continuous but non-differentiable functions, the RKHS also manages to perform reasonably well, though these functions introduce more complexity compared to smooth differentiable disturbances. Such functions are seen to require a larger number of basis centers to achieve a similar level of error reduction, reflecting their increased complexity and the RKHS's need to accommodate their irregular characteristics.

The step function, characterized by its inherent discontinuities, presents a substantial challenge. This type of function is notably problematic for RKHS because it deviates significantly from the native space of RKHS, which is defined by functions exhibiting smooth and continuous properties. The abrupt transitions found in step functions lead to poor approximation capabilities within the RKHS framework. This discrepancy highlights a fundamental limitation of RKHS as it struggles to approximate functions that fall outside of its native space effectively, resulting in elevated steady state errors.

5.4 Conclusion

The exploration of robust MRAC systems embedded with RKHS for quadcopters has illuminated several key aspects of advanced control systems. The analyses performed have highlighted both the strengths and limitations of integrating RKHS into MRAC frameworks, offering valuable insights into their practical implementation.

Firstly, the results from various simulations have shown that embedding RKHS into MRAC systems significantly enhances their adaptability and precision in handling disturbances. This improvement is especially evident in scenarios involving continuous differentiable and non-differentiable disturbances, where the RKHS's ability to approximate complex dynamics in real-time has led to more accurate trajectory tracking and velocity

control, even under challenging conditions.

However, the system's performance under discontinuous non-differentiable disturbances revealed some limitations. While RKHS integration aids in quicker response and error correction, the inherent characteristics of RKHS, which favor smooth and continuous functions, make it less effective against abrupt changes that are highly non-linear and discontinuous. This highlights an area for further research and potential refinement in RKHS methodology or alternative strategies that could better handle such extremes.

Moreover, the computational cost analysis has provided crucial data on the trade-offs involved. While the inclusion of more basis centers generally results in higher accuracy and better disturbance handling, it also significantly increases the computational load. This necessitates a balanced approach to configuring RKHS in practical applications, where computational resources and real-time response requirements must be carefully considered.

Overall, the integration of RKHS into MRAC systems presents a promising advance in control systems for quadcopters and potentially other dynamic systems. Future work will benefit from exploring adaptive or hybrid RKHS structures that could dynamically adjust their complexity based on real-time performance feedback, optimizing both computational efficiency and control accuracy. Additionally, further studies into RKHS applications in control systems should consider the impact of different kernel functions and their ability to adapt to the specific characteristics of the control task and disturbances encountered.

Chapter 6

Conclusion and Future Development

This thesis has provided a detailed examination of RKHS-embedded MRAC systems to design control systems for quadcopters. Within the context of robotics, the use of adaptive control systems that leverage a framework able to counter uncertainties and disturbances lying in some unparameterized Hilbert space is unprecedented. This application-focused approach yielded significant advancements in adaptability and robustness under varying operational conditions as well as an improved understanding of how the disturbance functions proximity to the native space influences system performance.

Throughout this work, the integration of MRAC with RKHS methodologies was explored to enhance the control strategy's response to nonlinear disturbances and complex dynamic scenarios encountered in UAV flight. The simulations performed, demonstrated the capability of this advanced control strategy in maintaining accurate trajectory and velocity tracking, even under the influence of sophisticated external disturbances modeled by complex non differentiable and discontinuous functions.

A crucial aspect of this thesis was the comprehensive simulation environment developed to test and validate the control strategies. These simulations provided a robust platform for evaluating the RKHS-enhanced MRAC system's performance, offering insights into its operational capabilities and limitations. The computational cost analysis of the RKHS implementation highlighted the balance between system accuracy and computational load, a key consideration for real time control applications.

While unprecedented results were achieved, the scope of this research was constrained

by time and resources, resulting in areas for further development. Specifically, extending the RKHS framework from the outer to the inner loop could enhance rotational tracking capabilities. Additionally, implementing bounds on control output could lead to smoother flight behavior. Furthermore, while the simulation results were encouraging, testing these control strategies on an actual UAV would provide more conclusive evidence of their effectiveness. Practical implementation on a physical quadcopter would allow for the verification of performance characteristics in real-world conditions, including the management of challenging payload dynamics such as sloshing during liquid transport in applications like firefighting and agricultural spraying, where dynamic payload adjustments are essential.

In closing, this thesis lays the groundwork for future research that could extend these methodologies to broader applications within aerospace and other fields requiring precise, adaptive control systems. The integration of theoretical research with potential real-world applications sets a promising path forward for the development of more resilient, efficient, and adaptable UAV control systems.

Acknowledgement

I am deeply grateful to my committee, especially Dr. Andrea L'Afflitto, for the extensive and valuable feedback he provided throughout my research process. His detailed guidance has been crucial to my learning and success. Additionally, I express my gratitude to Dr. Andrew Kurdila for his insights, which were vital in refining the direction of my research and enhancing my understanding of complex topics. I am immensely grateful to both for their pivotal roles in helping me successfully complete my thesis under significant time constraints.

I am thankful to Mattia Gramuglia, whose prior research provided a solid foundation for my work. Mattia's contributions have greatly clarified many of the complexities of my thesis topic, enriching my understanding and influencing the direction of my study.

Lastly, I would like to thank Haoran Wang for his readiness to aid my research endeavors. Haoran's generosity in sharing his time and expertise has profoundly deepened my understanding of our shared research interests.

Bibliography

[1] R. B. Anderson, J. A. Marshall, and A. L'Afflitto, "Constrained robust model reference adaptive control of a tilt-rotor quadcopter pulling an unmodeled cart," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 57, no. 1, pp. 39–54, 2021.

[2] J. A. Marshall, W. Sun, and A. L'Afflitto, "A survey of guidance, navigation, and control systems for autonomous multi-rotor small unmanned aerial systems," *Annual Reviews in Control*, vol. 52, pp. 390–427, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1367578821000882

[3] E. Lavretsky and K. A. Wise, *Robust and Adaptive Control With Aerospace Applications*. London: Springer, 2012.

[4] M. Mehrandezh and R. Paranjape, "Nonlinear adaptive control using neural networks: Stability analysis via lyapunov's direct method," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 7, pp. 3065–3076, 2018.

[5] S. Li, S. Khadka, S. Majumdar, and K. Tumer, "Evolutionary reinforcement learning for sample-efficient multiagent coordination," *Journal of Robotics and Autonomous Systems*, vol. 139, p. 103755, 2021.

[6] A. L'Afflitto, R. B. Anderson, and K. Mohammadi, "An introduction to nonlinear robust control for unmanned quadrotor aircraft: How to design control algorithms for quadrotors using sliding mode control and adaptive control techniques," *IEEE Control Systems Magazine*, vol. 38, no. 3, pp. 102–121, 2018.

[7] M. Gramuglia, "Design and high-fidelity simulations of an adaptive control system for x8-copters employed for payload delivery," Master's thesis, Politecnico di Milano, 2022-23, advisor: Prof. Francesco Braghin; Co-advisors: Prof. Andrea L'Afflitto.

[8] P. Bibik, J. Narkiewicz, M. Zasuwa, and M. Żugaj, "Quadrotor dynamics and control for precise handling," in *Springer International Publishing*. Cham: Springer, 2016, pp. 335–351.

[9] G. Hoffmann, H. Huang, S. Waslander, and C. Tomlin, "Quadrotor helicopter flight dynamics and control: Theory and experiment," 2007.

[10] E. A. A. Retha, *Novel Concepts in Multi-rotor VTOL UAV Dynamics and Stability*, 2017, ch. 20, pp. 667–694.

[11] A. L'Afflitto and K. Mohammadi, "Equations of motion of rotary-wing unmanned aerial system with time-varying inertial properties," *Journal of Guidance, Control, and Dynamics*, vol. 41, no. 2, pp. 559–564, 2018.

[12] H. Wang, A. Kurdila, A. L'Afflitto, D. Oesterheld, and D. Stilwell, "Robust model reference adaptive control based on reproducing kernel hilbert spaces," *Journal of Dynamic Systems, Measurement, and Control*, 2024.

[13] A. J. Kurdila, A. L'Afflitto, and J. A. Burns, *Robust and Adaptive Control of Nonlinear Systems over Native Spaces*. London, UK: Springer, accepted, publication pending.

[14] E. Kreyszig, *Introductory Functional Analysis with Applications*, 1st ed. John Wiley & Sons, 1978.

[15] H. Wendland, *Scattered data approximation*. Cambridge, UK: Cambridge University Press, 2004.

[16] V. I. Paulsen and M. Raghupathi, *An Introduction to the Theory of Reproducing Kernel Hilbert Spaces*. Cambridge, UK: Cambridge University Press, 2016, vol. 152.

[17] T. Evgeniou, C. A. Micchelli, and M. Pontil, "Learning multiple tasks with kernel methods," *Journal of Machine Learning Research*, vol. 6, no. Apr, pp. 615–637, 2005.

[18] D. Wittwar, G. Santin, and B. Haasdonk, "Interpolation with uncoupled separable matrix-valued kernels," *Seminari Padovani di Analisi Numerica*, vol. 11, pp. 23–39, 2018.

[19] H. Wang, B. Scurlock, N. Powell, A. L'Afflitto, and A. J. Kurdila, "Mrac with adaptive uncertainty bounds via operator-valued reproducing kernels," *IEEE Control Systems Letters*, vol. 7, pp. 3771–3776, 2023.

[20] H. Wang, N. Powell, A. L'Afflitto, A. J. Kurdila, and J. Burns, "The power function for adaptive control in native space embedding," *Journal of Control Theory and Applications*, 2023.

[21] A. Kurdila and Y. Lei, "Adaptive control via embedding in reproducing kernel Hilbert spaces," in *American Control Conference*. IEEE, 2013, pp. 3384–3389.