République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université 20 Août 1955 - Skikda

Ref: D012124027D

Faculté de Technologie
Département de Génie Electrique
Laboratoire d'Automatique Skikda

THÈSE

En vue de l'obtention du diplôme de

Doctorat LMD

Domaine : **Science et Technologie**

Filière : **Génie Electrique**

Spécialité : **Automatique et Informatique Industrielle**

Présentée par

BENHARKOU Ibtihal

Thème

Contribution à la Conception de Lois de Commande Robustes Bio-Inspirées

Devant le jury composé de :

Président	Pr. LACHOURI Abderrazek	Professeur	Université de Skikda
Rapporteur	Pr.GHERBI Sofiane	Professeur	Université d'Annaba
Co-Rapporteur	Dr. MEHENNAOUI Lamine	MCA	Université de Skikda
Examinateurs	Pr. KIDOUCHE Madjid	Professeur	Université de Boumerdes
	Pr. GUECHI El-Hadi	Professeur	Université de Skikda

Année 2024

People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research
University 20 August 1955 – Skikda

Ref: D012124027D

Faculty of Technology
Department of Electrical Engineering
Automatic Laboratory of Skikda

Thesis

for the degree of

Doctor of Philosophy (LMD Doctorate)

Domain: Science and Technology
Section: Electrical Engineering
Specialty: Automatic and Industrial Computing

Presented by

BENHARKOU Ibtihal

Theme

Contribution to the Conception of Bio-Inspired Robust Control Laws

Dissertation committee:

President	Pr. LACHOURI Abderrazek	Professeur	Université de Skikda
Supervisor	Pr.GHERBI Sofiane	Professeur	Université d'Annaba
Co-Supervisor	Dr. MEHENNAOUI Lamine	Associate Professeur	Université de Skikda
Examinateurs	Pr. KIDOUCHE Madjid	Professeur	Université de Boumerdes
	Pr. GUECHI El-Hadi	Professeur	Université de Skikda

Year 2024

Acknowledgment

In the name of Allah, the most Merciful, the most Gracious. With the will of God this work has been achieved. Therefore, all the words can not express my thankfulness and gratitude to Allah who give me the chance to live, to learn, to attain the highest level of education and working on control techniques inspired from his mighty creation. Aiming to have his mercy and reward for following the first order in the first divine revelation to the prophet Muhammad peace be upon him which was "learn". Then,

I want to express my sincere gratitude to my respected supervisor, Pr Sofiane Gherbi, for his guidance, continuous support, valuable advises, his patience and his availability. I'll be so recognised to his honesty, dedication in work and to all what he did throughout the entire period of researches and my thesis redaction. Being under his supervision has made me strive to do more, thank you for believing on me.

I would like to express my deep gratitude to all of my dissertation committee who honored me for evaluating this work, Mr. Kidouche Abdelmadjid, Mr. Lachouri Abderrazek, Mr. Guechi El-Hadi and Mr. Mehennaoui lamine, thank you for accepting to read and review my work, your valuable feedback and constructive criticism helped me to refine and improve the quality of this research.

A dept of gratitude is owed to my parents for unwavering patience, their sacrifices and endless support throughout my personal and the academic life, providing me the adequate environment of work and the best of everything, giving me strength and invaluable motivation to chase my dreams and to see me a doctor.

I’m indebted to my husband for his invaluable contribution in this work, his patience, support and understanding during the completion of this thesis, I would like to extend my heartfelt appreciation to him, to be with me at every moment and to keep me going through the way after all the obstacles I faced.

Last and not the least, I would like to express my deepest gratitude to all the teachers who taught me in my curses, primary, middle, secondary schools and the university for their countless lessons, their kindness and guiding me on the path of greatness, with a special thank to who inspired me more and helped me build a strong foundation at the primary school "my mother".

BENHARKOU Ibtihal

Dedication

...

I would be honor to dedicate my dissertation work to:

My loving parents, I hope this achievement fulfill the dream you envisioned for me.

My parents in law, thank you for your love, supporting and praying for me all the time.

My husband Hamza, without your assistance the completion of this work would not have been possible.

My brother Mohamed dhiae eddine, my sisters Nour el Houda and roufeida, thank you for all the things you have done for me, I know you'll be proud of me as much as I am proud of you all.

My beloved son Adam Wael, my nephew Mohamed Redha, and my niece Amani, you mean the world for me.

A special dedication to my aunts, my cousins and specially my beloved supportive cousin Samira, my cousin Mohamed el Amine, and my dear cousin Boutheina.

My faithful sisters and brothers in law, my dear friend Lydia and all the families Benharkou and Kaddour.

Abstract

This thesis explores the robustification of bio-inspired control methods, focusing on two key contributions: the artificial immune feedback control system and neural networks. The primary goal is to enhance their robustness against uncertainties and disturbances, ultimately improving their performance in real-world applications, using well-known robust control theory techniques.

The robustification of the artificial immune feedback control system is achieved in the first contribution through the utilization of H∞ synthesis, embodied in a structured H∞ PID controller. The second contribution employs the linear quadratic regulator (LQR) state feedback control law to enhance the robustness of the artificial immune feedback control system. It is demonstrated that this approach enhances its robustness and adaptability, making it more suitable for dynamic and uncertain environments. The third contribution concerns the robustification of a backpropagation neural network-based PID by modifying its structure with a filter.

Throughout the thesis, theoretical frameworks and practical implementations are presented to demonstrate the effectiveness of the proposed robustification techniques. Simulation results showcase the improved performance, stability, and adaptability of the bio-inspired control systems in both artificial immune feedback control and neural networks.

Key words: Robust control, bio-inspired algorithms, optimization.

ملخص

تستكشف هذه الرسالة تقوية أساليب التحكم المستلهمة من البيولوجيا، مركزة على مساهمتين رئيسيتين: نظام التحكم الصناعي للردود المناعية الاصطناعية والشبكات العصبية. الهدف الرئيسي هو تعزيز متانتها ضد الأوضاع غير المؤكدة والاضطرابات، مما يؤدي في نهاية المطاف إلى تحسين أدائها في التطبيقات العملية، باستخدام تقنيات التحكم المستدامة المعروفة جيدًا.

تحقق تقوية نظام التحكم الصناعي لردود الفعل المناعية الاصطناعية في المساهمة الأولى من خلال استخدام توليف منظم، بنية تستخدم (H∞) المتجسد في متحكم (H∞PID). المساهمة الثانية تعتمد على قانون التحكم برد الحالة بواسطة المنظم الرباعي الخطي لتعزيز تقوية نظام التحكم الصناعي، لردود الفعل المناعية الاصطناعية. يظهر أن هذا النهج يعزز متانتها وقابليتها للتكيف، يجعلها أكثر ملاءمة للبيئات الديناميكية وغير المؤكدة. تتعلق المساهمة الثالثة بتقوية المنظم (PID) المستند إلى الشبكات العصبية التي تعتمد على انتشار الرجوع عن طريق تعديل هيكلها بفلتر.

طوال الرسالة، يتم تقديم الإطارات النظرية والتنفيذات العملية لإظهار فعالية تقنيات التقوية المقترحة. تُظهر النتائج التجريبية تحسين الأداء والاستقرار والقابلية للتكيف لأنظمة التحكم المستلهمة من البيولوجيا في كل من التحكم الصناعي لردود الفعل المناعية الاصطناعية والشبكات العصبية.

الكلمات الرئيسية: التحكم المستدام، الخوارزميات المستلهمة من البيولوجيا، التحسين.

Résumé

Cette thèse explore la robustification des méthodes de commande bio-inspirées, en se concentrant sur deux contributions clés : le système de commande à rétroaction immunitaire artificielle et les réseaux neuronaux. L'objectif principal est d'améliorer leur robustesse face aux incertitudes et aux perturbations, améliorant ainsi leurs performances dans les applications du monde réel, en utilisant des techniques bien connues de théorie de commande robuste.

La robustification du système de commande à rétroaction immunitaire artificielle est réalisée dans la première contribution grâce à l'utilisation de la synthèse H∞, incarnée dans un contrôleur PID H∞ structuré. La deuxième contribution utilise la loi de commande à retour d'état du régulateur quadratique linéaire (LQR) pour améliorer la robustesse du système de commande à rétroaction immunitaire artificielle. Il est démontré que cette approche améliore sa robustesse et son adaptabilité, la rendant ainsi plus adaptée aux environnements dynamiques et incertains. La troisième contribution concerne la robustification d'un PID basé sur un réseau neuronal à propagation arrière en modifiant sa structure avec un filtre.

Tout au long de la thèse, des bases théoriques et des mises en œuvre pratiques sont présentés pour démontrer l'efficacité des techniques de robustification proposées. Les résultats de simulation mettent en avant les performances améliorées, la stabilité et l'adaptabilité des systèmes de commande bio-inspirés à la fois dans la commande à rétroaction immunitaire artificielle et les réseaux neuronaux.

Mots clés : Commande robuste, algorithmes bio-inspirés, optimisation.

Introduction

The field of control systems has witnessed remarkable developments in the recent years, particularly in the realm of bio inspired control which draw inspiration from the adaptability and resilience found in the biological systems, these control strategies could promote the control challenge of complex and dynamic systems using the bio-inspired control algorithms offering promising solutions for diverse engineering applications[1]. However, in practical applications, control systems often encounter uncertainties, disturbances, and external factors that can affect their performance and stability. Robustness, the ability of a control system to maintain its desired performance in the face of such challenges, is of paramount importance[2].

The primary objective of this thesis is to delve into the realm of bio-inspired control methods, encompassing the following developed techniques, starting with the artificial immune systems which represents a computational model inspired by the functioning of the human immune system such the pattern recognition, memory and response behavior, the artificial immune system uses several mechanisms like clonal selection, negative selection, optimize control parameters, this lead the improvement of control performance and robustness in different domains including robotics, industrial automation and process control [3, 4]. Neural networks, which replicate the way the neurons in our brain works, have found widespread interest in controlling systems as it was employed successfully in many applications such as: the approximation of complex nonlinear function which relate the system input with its output, another application of neural network can be seen in adaptive control, it can adjust continuously the control parameters based on real time feedback, this is particularly helpful in situations where the system's dynamic change over time, in this case the neural network can adapt to these changes and optimize the suitable control action, furthermore, the neural network can be used in fault detection and diagnosis, predictive control, using the reinforcement learning an advanced machine learning

paradigm[5, 6]. Particle swarm optimization (PSO) is a methaheuristic optimization algorithm that is inspired by the social behavior of birds flocking or fish schooling, it was applied in wide range of discreet and continues optimization problems[7]. Furthermore, it's worth siting the genetic algorithms (GA), which represent an optimization algorithm inspired by the process of natural selection and evolution, they are used to find solution of complex optimization problems[8].

This research endeavors to refine and enhance the effectiveness of these bio-inspired control techniques, particularly in the face of uncertainties, disturbances, and variable operating conditions. In the course of our research, we have achieved promising results that highlight the potential of these robustified bio-inspired control methods that are not only inspired by nature but also capable of overcoming the challenges encountered in practical, real-world systems, in order to revolutionize the field of control systems. The following thesis is organized into several chapters as follow:

Chapter 1: Robust Control System: This chapter provides an essential foundation by exploring the fundamentals of robust control systems particularly, the H ∞ synthesis and the linear quadratic optimal control. It discusses the necessity for robust control in today's dynamic and uncertain environments, introducing the concept of robustness.

Chapter 2: Bio-Inspired Control: Bio-inspired control, is the subject of this chapter. It delves into the principles of the artificial immune feedback control and the neural network, emphasizing the mechanisms and inspirations drawn from the natural systems.

Chapter 3: H ∞ based PID Immune Feedback Controller Design: This chapter focuses on the application of an alternative to the H ∞ control theory known by the structured H ∞ control to enhance the robustness of the artificial immune feedback mechanism. It explores the integration of H ∞ control methods with bio-inspired principles. A comparative study between the PID immune feedback controller and the structured H ∞ PID immune controller is achieved in order to control a magnetic levitated system, with the exhibition and the discussion of the results found.

Chapter 4: LQR based Fuzzy Immune PID Controller: Chapter 4 delves into the combination of Linear Quadratic Regulator (LQR) and immune feedback control to design a new robust LQR based fuzzy immune PID controller. This hybrid approach leverages the strengths of LQR and fuzzy immune feedback control techniques to create a controller that exhibits robustness in front of the network disturbances comparing its results with

those found using the classical PID controller and the fuzzy immune PID controller.

Chapter 5: Neural Network PID Controller: In this chapter, we explore the utilization of neural networks for the design of PID controllers. The integration of artificial neural networks into control systems provides adaptability and learning capabilities, which can enhance the robustness of control methods. Our main contribution lies in using a filtered PI controller, where the neural network learning will be according to this new structure. To validate this approach, the standard PID controller was robustly synthesized, besides the neural network PID controller, these three controllers are used separately for the control of a system with infinite static gain and a pole at the origin, the simulation results are presented and discussed focusing on the performance robustness test.

The objective of this thesis is to contribute to the advancement of control systems by robustifying bio-inspired control techniques. By investigating and developing these innovative methodologies, we aim to create control systems that mimic nature's adaptability while withstanding the challenges posed by real-world applications. The subsequent chapters will provide in-depth insights into each aspect of this research, ultimately bringing us closer to achieving more resilient and adaptive control solutions for a wide range of engineering applications.

0.1 Contribution

The contributions of this thesis were the subject of the following works:

- Banharkou, I., Gherbi, S., Mehennaoui, L. A robust LQR-based fuzzy-immune PID applied for a greenhouse temperature networked control. European Journal of Electrical Engineering, Vol. 24, No. 3, pp. 133-138. (2022).
  https://doi.org/10.18280/ejee.240302

- Benharkou, I., Gherbi, S., Sedraoui, M. et al. A robust proportional filtered integral controller based on backpropagation neural network. Int. J. Transactions of the Institute of Measurement and Control, pp. 1–11. (2024).
  https://journals.sagepub.com/doi/abs/10.1177/01423312241266011 #tab-contributors.

- Ibtihal Banharkou, Sofiane Gherbi and Lamine Mehennaoui, Robust H ∞ Arti-

ficial Immune Feedback Controller, ICTAEE'18: Third International Conference On Technological Advances in Electrical Engineering, Skikda, Algeria, November 13-15, 2018.

- Ibtihal Banharkou, Sofiane Gherbi, Lamine Mehennaoui and Moussa Sedraoui, A New Robust Structured H ∞ PID With A Weighting Function Inspired From the Immune Feedback Mechanism, ICTAEE23: Fourth International Conference On Technological Advances in Electrical Engineering, Skikda, Algeria, May 23-24, 2023.

Table of Contents

List of Figures

List of Tables

Chapter 1

Robust Control System

1.1 Introduction

Classical feedback control systems have encountered limitations in recent years, primarily due to their inability to guarantee optimal performance in the presence of perturbations and uncertainties, particularly in poorly modeled systems. Indeed, it is worth noting that no mathematical model can precisely represent a physical system due to various factors. The precise measurement of system parameters can be challenging, and modeling nonlinearities is often a difficult task, which frequently results in their neglect. Consequently, this process yields a simplified and practical system that lacks certain characteristics and dynamics inherent in the actual system, referred to as plant uncertainty.

Robust control is a control theory technique that addresses the issue of uncertainty in control systems. It aims to design controllers that can perform satisfactorily in the presence of uncertainties, such as parameter variations, disturbances, and noise [18]. There are two main types of robustness [19]:

1. Stability robustness: Ensures the system's stability in the presence of uncertainties. A system is said to be robustly stable if it remains stable for all plants in the uncertainty set.

2. Performance robustness: Minimizes performance degradation under uncertain conditions. A system is said to have good performance 0 robustness if its performance specifications are met for all plants in the uncertainty set.

some of well known robust control methods are LQG/LQR, H∞ [20]

In control engineering, there is also a need to minimize a performance criterion, which leads to the development of optimization concepts. The optimal control problem can be described as the task of finding a control signal that not only satisfies to physical constraints, but also maximizes (or minimizes) a specified performance criterion. [21].
The following chapter gives an overview of the robust control theory, and especially the H∞ synthesis.

1.2 Plant uncertainty description

The mathematical model is designed for a physical plant complex and subtle, that's evaluated on how closely its response matches the one of the real plant. The discrepancy between the real system and the modeled one is called uncertainty, whatever the representation of uncertainty refers to the mechanism used to describe errors [10].
The controlled system is bounded by several uncertainties that can be divided in two classes: disturbance signal and perturbations in the model dynamics. Disturbances include the external stochastic inputs out of control, they can be from the signals linking the controlled loop elements as in the output signal acquisition by sensors, noised information by the actuator, or from the environment. Dynamic perturbations can be the result of the changes in system's parameters or unmodeled plant dynamics and particularly at high frequencies, as it can be due to the linearization of the system that has the risk of losing lot of data in the nonlinear parts [22].
These modeling uncertainties will affect with no doubt the stability and performance of the feedback control, the reason that makes the necessity of taking them into account in the system model design [23].

1.2.1 Non-parametric Uncertainty

Non-parametric uncertainties can be occurred in the aggregation of the nominal system to ease the calculations and to get rid from the complexity of the complete model that has the risk of missing lot of information of the modeled system. other reason of this kind of uncertainties is the parasitic in the actuators, transmitters or sensors that are often neglected [24]. The non-parametric uncertainties are modeled in one block , this block can be added or multiplied by the nominal system in different ways and give a perturbed dynamic plant. the additive uncertainty represents the absolute error between the nominal model and the actual dynamic, while

the multiplicative uncertainty represents the relative error[25].

1.2.2 Parametric Uncertainty

Parametric uncertainties or as they are known by the structured uncertainties, they represent the uncertainty in the physical system parameters estimated from physic laws and engineering, if the model parameters are obtained experimentally, all the disturbances that affect the system during the experiment increase the remaining uncertainties that is shown in the system parameters by the marge of tolerance, for example the electrical components that have always a range of tolerance we site the one of the standard resistor which is equal to 2 [22].

1.3 H∞ Control Theory

The H∞ problem was first introduced by its founder, Zames, at the IEEE CDC in 1976 and at the Allerton conference in 1979. It was formally posed in his paper in 1981 [26]. However, the origins of the H∞ problem date back to 1960, it was subsequently extended based on Zames' small gain theorem [27]. In 1989, Doyle, Glover, Khargonikar and Francis [28] argue that the H∞ problem can be solved by the solution of two Riccati equations. Later in 1994, Gahinet and Apkarian [29] provided a solution to the H∞ problem by reformulating it as a linear matrix inequality (LMI).

1.3.1 H∞ Norm Performance

The H∞ norm is a notation given to the peak value of a system's frequency response, this signifies the maximum energy gain over all the range of frequencies. The Bode diagram, known for representing the frequency response of a system through a combination of magnitude and phase plots, was initially designed for SISO systems. This concept was later extended to MIMO systems, giving rise to the singular values [30], which are a matrix decomposition playing a crucial role in the H∞ theorem. The following passage provides an overview of the singular values: The singular values of a matrix $M \in R^{m\times n}$, noted σ_i the non-negative square roots of the eigenvalues of $M^T M$ in the order $\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_n$, represented in a diagonal matrix $\Sigma \in R^{m\times n}$. Moreover, the decomposition of the matrix results two unitary matrices $U \in R^{m\times m}$ and $V \in R^{n\times n}$, such that:

$$M^T M = U\Sigma V^T = U \begin{pmatrix} \Sigma_r & 0 \\ 0 & 0 \end{pmatrix} V^T \tag{1.1}$$

$\Sigma_r = diag(\sigma_1, \sigma_2, ..., \sigma_r)$, this represents the method, Singular Value Decomposition (SVD) of the matrix M , one of the frequently used methods of algebra. The greatest singular value σ_1 ,is denoted $\overline{\sigma}(M)$ and the least singular value σ_r is denoted $\underline{\sigma}(M)$ [24]. The concept of Singular Value Decomposition is applied for calculating the H∞ norm as follow:

Consider a stable transfer function of SISO system $G(s) = \frac{Y(s)}{U(s)}$ where $U(s)$ and $Y(s)$ represent the input signal and the output signal in the frequency domain respectively.

The H∞ norm is defined as follow:

$$\|G(s)\|_\infty = Sup(\overline{\sigma}(G(j\omega))) \tag{1.2}$$

1.3.2 Properties of H∞

$$\forall F, G \in C^{p\times m}, \|F(s)G(s)\|_\infty \leq \|F(s)\|_\infty \|G(s)\|_\infty$$

$$\forall F, G \in C^{p\times m}, \left\| \begin{matrix} F(s) \\ G(s) \end{matrix} \right\|_\infty \geq \sup(\|F(s)\|_\infty, \|G(s)\|_\infty)$$

$$\forall F, G \in C^{p\times m}, \left\| F(s) \quad G(s) \right\|_\infty \geq (\|F(s)\|_\infty, \|G(s)\|_\infty)$$

The H∞ norm can be calculated according to an algorithm in iterative manner, by founding the smallest value of γ where the matrix $H(\gamma)$ has no eigenvalue in the imaginary axis. The maximum value of γ is approved by the following property:

$\|G(s)\|_\infty \prec \gamma$ if and only if the following Hamiltonian matrix has no eigenvalue in the imaginary axis:

$$H_\gamma = \begin{bmatrix} A - BR^{-1}D^TC & -\gamma BR^{-1}B^T \\ \gamma C^TS^{-1}C & -A^T + C^TDR^{-1}B^T \end{bmatrix} \tag{1.3}$$

With

$$\begin{cases} R = D^TD - \gamma^2 I \\ S = DD^T - \gamma^2 I \end{cases} \tag{1.4}$$

To find the H∞ of norm of a system, we can just look for the smallest value of γ where the matrix $H(\gamma)$. the following steps give a linear method to evaluate γ [30]:

1. we choose two values of γ , γ_{inf} and γ_{sup} with a marge of tolerance ε .

2. we test for $\gamma = 1/2\left(\gamma_{\text{sup}} + \gamma_{\text{inf}}\right)$, if H_γ has no pole in the imaginary axis, so $\gamma_{\text{sup}} = \gamma$ otherwise $\gamma_{\text{inf}} = \gamma$.

3. if $\gamma_{\text{sup}} - \gamma_{\text{inf}} \succ \varepsilon$ so go to the step number 2, otherwise $\|G(s)\|_\infty = 1/2\left(\gamma_{\text{sup}} + \gamma_{\text{inf}}\right)$.

1.3.3 The Standard H∞ Control Problem

The standard H∞ "sub-optimal" control problem can be illustrated by the Linear Fractional Transformation (LFT) scheme bellow (figure1.1).

w : represents the exogenous inputs (set points, perturbations... etc.)

u : is the control signal.

z : is the regulated output.

y : is the measured output.

P : is the augmented plant.

K : is the controller.

The state space of the P plant is the following:

$$P\begin{cases} \dot{x} = Ax + B_1w + B_2u \\ z = C_1x + D_{11}w + D_{12}u \\ y = C_2x + D_{21}w + D_{22}u \end{cases} \tag{1.5}$$

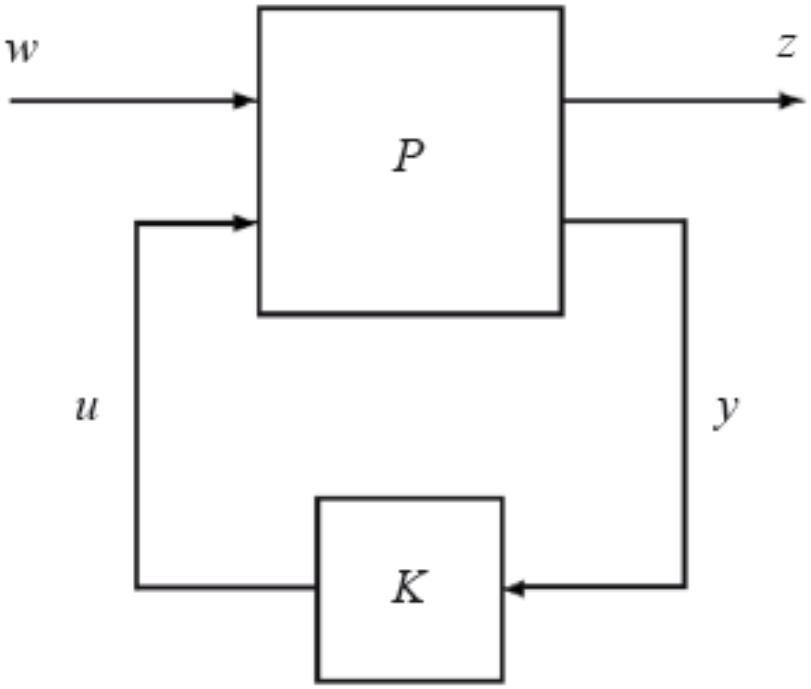

Figure 1.1: the standard closed loop Linear Fractional Transformation (LFT) model

From the figure 1.1 above we can write the following presentation:

$$\begin{bmatrix} z(s) \\ y(s) \end{bmatrix} = \begin{bmatrix} P_{11}(s) & P_{12}(s) \\ P_{21}(s) & P_{22}(s) \end{bmatrix} \begin{bmatrix} w(s) \\ u(s) \end{bmatrix} \tag{1.6}$$

$$u(s) = Ky(s) \tag{1.7}$$

$$z(s) = P_{11}w(s) + P_{12}Ky(s) \tag{1.8}$$

$$y(s) = (sI - P_{22}K(s))^{-1}P_{21}w(s) \tag{1.9}$$

From the equations (1.6), (1.7) and (1.8) we can conclude the transfer matrix T_{zw} between the exogenous inputs and the output to be controlled z.

$$T_{zw}(s) = F_l(P,K) = P_{11}(s) + P_{12}K(s)(sI - P_{22}K(s))^{-1}P_{21} \tag{1.10}$$

The notation $F_l(P,K)$ means linear fractional transformation.[9].

1.3.3.1 Optimal H∞ Control Problem The optimal H∞ problem is how to find an optimal controller capable of stabilizing the given plant and minimizing the H∞ norm of the closed loop transfer matrix T_{zw} . It should be noted that the H∞ controller is not exclusively designed for MIMO systems; however, its complexity has rendered it an impractical choice in many applications. Over time, a more practical suboptimal H∞ controller has been developed, maintaining essential properties, such as lower bandwidth [10].

1.3.3.2 Sub-optimal H∞ Control Problem For a given $\gamma > 0$, find all controllers guaranteed $\|T_{zw}\|_\infty < \gamma$, this characteristic will be used for finding the H∞ Controller [10].

We can summarize the H∞ Control Problem as follow:

For the stable LFT $F_l(P, K)$

find the controller K where $\|F_l(P, K)\|_\infty < \gamma$

1.3.4 Solution of the H∞ Control Problem

The state space of the augmented system represented in the figure 1, is taken to be in the following form:

$$\begin{bmatrix} \dot{x}(t) \\ z(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} A & B_1 & B_2 \\ C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{bmatrix} \begin{bmatrix} x(t) \\ w(t) \\ u(t) \end{bmatrix} \tag{1.11}$$

With: dim $x = n$, dim $w = m_1$, dim $u = m_2$, dim $z = p_1$, dim $y = p_2$

The following assumptions must be verified before applying the H∞ synthesis [30] [9]:

(A1) (A , B_2) is stabilizable and (C_2 , A) is detectable.

(A2) $D_{12} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ and $D_{21} = \begin{bmatrix} 0 & 1 \end{bmatrix}$.

(A3) $\begin{bmatrix} A - j\omega I & B_2 \\ C_1 & D_{12} \end{bmatrix}$ has full column rank for all ω .

(A4) $\begin{bmatrix} A - j\omega I & B_1 \\ C_2 & D_{21} \end{bmatrix}$ has full raw rank for all ω .

Assumption (A1) is a necessity for the existence of controllers capable to stabilize the system. The condition given in the assumption (A2) makes of the controller transfer matrix

proper. The assumptions (A3) and (A4) guaranteed respectively that the transfer $P_{z\to u}(P_{12})$ and $P_{y\to w}(P_{21})$ have no zero in the imaginary axis.

For simple expressions, we sit these complementary conditions:

$$\begin{cases} D_{11} = D_{22} = 0 \\ {D_{12}}^T(C_1 \quad D_{12}) = (0 \quad \mathrm{I}_{m2}) \\ \begin{pmatrix} B_1 \\ D_{21} \end{pmatrix} {D_{21}}^T = \begin{pmatrix} 0 \\ I_{p2} \end{pmatrix} \end{cases} \tag{1.12}$$

1.3.5 Solution of the H∞ Control Problem with Riccati equation

If the assumptions above are hold, the resolution of the H∞ control problem can be solved by Riccati equations. This method is known also by the algorithm of Glover-Doyle, it appeared for the first time in 1980 [30].

Riccati equation is given by the following expression:

$$XA + A^T X - XPX + Q = 0 \tag{1.13}$$

with: $P = P^T$ and $Q = Q^T$

The solution X, will be a symmetric matrix if it exists. Where is a stable matrix that the real part of their eigenvalues is negative, the solution is the following:

$$X = Ric\begin{pmatrix} A & -P \\ -Q & -A^T \end{pmatrix} \tag{1.14}$$

For the H∞ sub-optimal control problem the theorem below must be verified.

Theorem 1: after satisfying the assumptions A1-A4, the H∞ control problem has a solution, unless the five following equations are verified:

1. The matrix $H_\infty = \begin{pmatrix} A & \gamma^{-2}B_1{B_1}^T - B_2{B_2}^T \\ -{C_1}^T C_1 & -A^T \end{pmatrix}$ doesn't have eigenvalues in the imaginary axis.

2. It exists a symmetric matrix $X_\infty = Ric(H_\infty) \geq 0$ that means $A^T X + XA + X(\gamma^{-2}B_1 B_1^T - B_2 B_2^T X + C_1^T C_1 = 0$.

3. The matrix $J_\infty = \begin{pmatrix} A^T & \gamma^{-2}C_1^T C_1 - C_2^T C_2 \\ -B_1 B_1^T & -A \end{pmatrix}$ doesn't have any eigenvalue in the imaginary axis.

4. It exists a symmetric matrix that $Y_\infty = Ric(J_\infty) \geq 0$, that means $AY + YA^T + Y\left(\gamma^{-2}C_1^T C_1 - C_2^T C_2 + B_1 B_1^T\right) = 0$.

5. $\rho(X_\infty Y_\infty) < \gamma^2$

Theorem 2: after satisfying the conditions given by the theorem above the controller stabilizing the system is described by the following representation:

$$K = \begin{pmatrix} \hat{A}_\infty & Z_\infty Y_\infty C_2^T \\ -B_2^T X_\infty & 0 \end{pmatrix} \tag{1.15}$$

with:

$\hat{A}_\infty = A + \gamma^{-2}B_1 B_1^T X_\infty - B_2 B_2^T X_\infty - Z_\infty Y_\infty C_2^T C_2$.

$Z_\infty = \left(I_n - \gamma^{-2}Y_\infty X_\infty\right)^{-1}$.

Before applying this controller, we should pass by the first theorem to have an approximate value of γ [9].

1.3.6 Robust Stability

Consider the system with multiplicative uncertainty given in the figure below (figure 1.2) with multiplicative uncertainty of magnitude $|w_I(j\omega)|$.

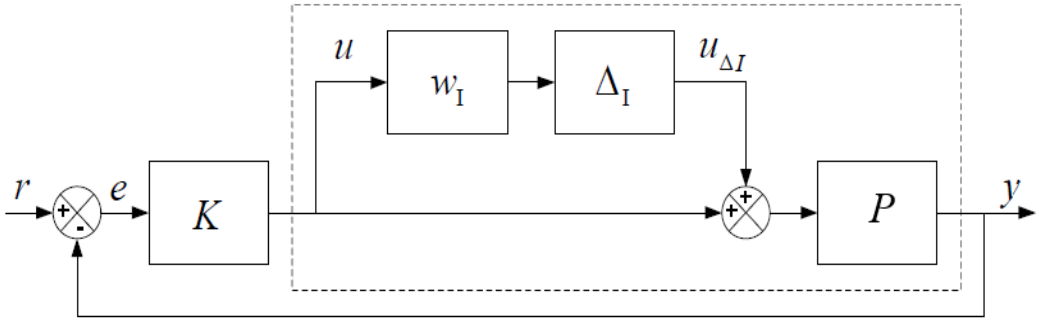

Figure 1.2: Feedback system with multiplicative uncertainty[9]

The transfer function of the system with uncertainty is:

$L_P = PK(1 + w_I\Delta_I) = L + Lw_I\Delta_I$ with $|\Delta_I(j\omega)| \leq 1$, $\forall\omega$.

In order to learn the robust stability, the figure below shows the the Nyquist plot of the system, and according to Nyquist concept the L_P should not encircle the point -1.

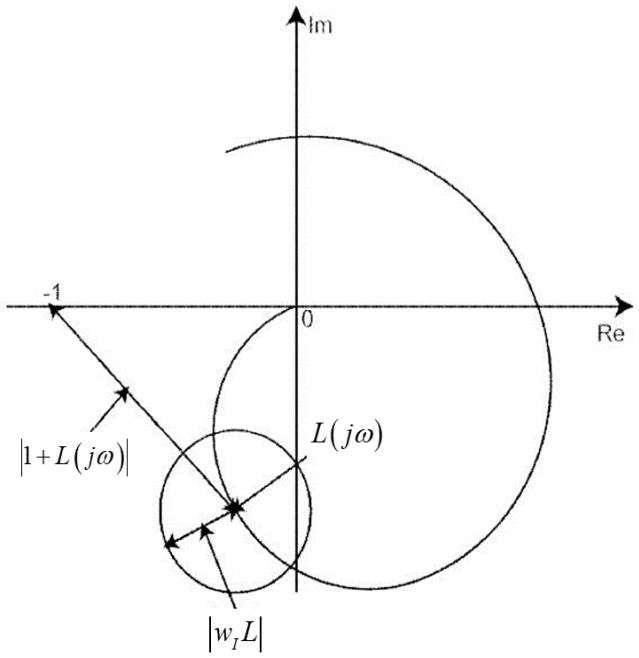

Figure 1.3: Nyquist plot of L_P for robust stability[9]

For robust stability, the circle represented L_P should not cover -1, consequently and $\forall\omega$:

$$RS \Leftrightarrow |w_I L| < |1 + L| \tag{1.16}$$

$$\left|\frac{w_I L}{1 + L}\right| < 1 \Leftrightarrow |w_I T| < 1 \tag{1.17}$$

$$\|w_I T\|_\infty < 1 \tag{1.18}$$

Where is the complementary T sensitivity function.

Hence the robust stability requirement is obtained for a system with multiplicative uncertainty [31].

$$RS \Leftrightarrow |T| < {}^1/_{w_I} \tag{1.19}$$

1.3.7 Robust Performance

To achieve robust performance, the nominal performance condition given by the equation below must be satisfied $\forall\omega$:

$$NP \Leftrightarrow \left|w_p S\right| < 1 \Leftrightarrow \left|w_p\right| < |1 + L| \tag{1.20}$$

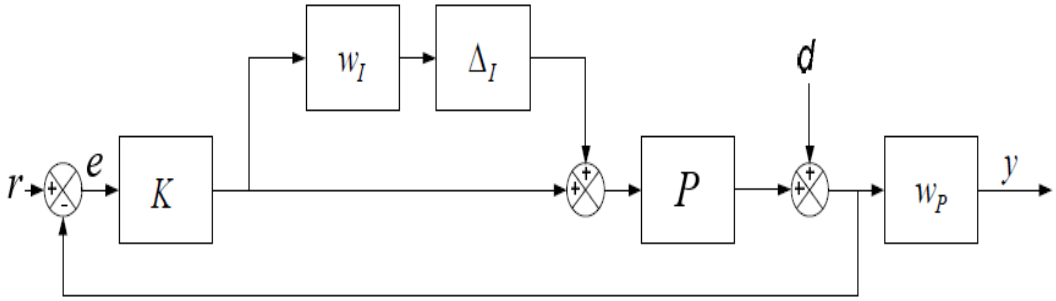

Figure 1.4: Control loop with multiplicative uncertainty[9]

$$RP \Leftrightarrow \left|w_p S_p\right| < 1 \Leftrightarrow \left|w_p\right| < \left|1 + L_p\right| \tag{1.21}$$

The Nyquist plot illustration is given by the following figure (figure 1.5): The robust

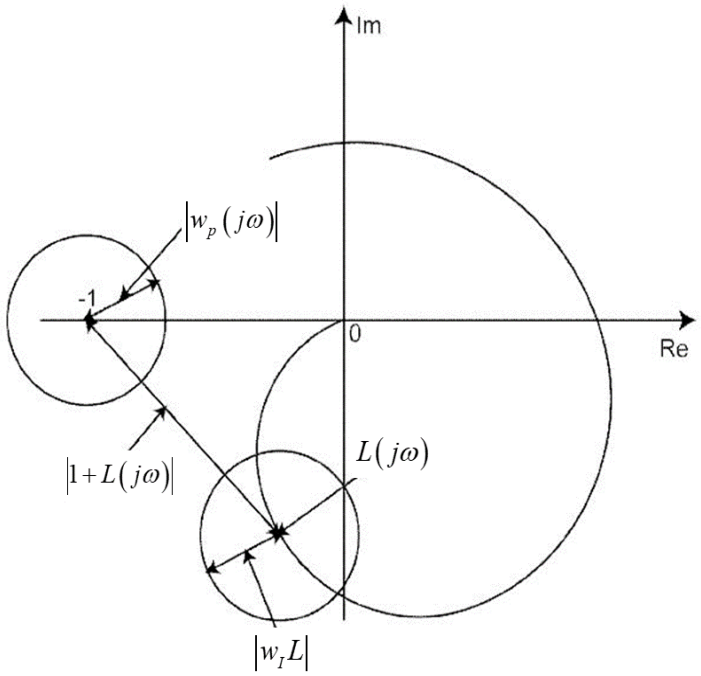

Figure 1.5: Nyquist plot illustration L_P for robust performance[9]

performance required that $L_p(j\omega)$ should not cross the circle with radius $\left|w_p(j\omega)\right|$ centred on -1.

The robust performance condition becomes:

$$RP \Leftrightarrow \left|w_p\right| + |w_I L| < |1 + L| \tag{1.22}$$

$$\Leftrightarrow \left|w_p(1 + L)^{-1}\right| + \left|w_I L(1 + L)^{-1}\right| < 1 \tag{1.23}$$

Or

$$RP \Leftrightarrow \max_{\omega} \left(\left| w_p S \right| + \left| w_I T \right| \right) < 1 \tag{1.24}$$

We summarize the robust conditions cited above by the following equations we get $\forall \omega$:

$$NP \Leftrightarrow \left| w_p S \right| < 1 \tag{1.25}$$

$$RS \Leftrightarrow \left| w_I T \right| < 1 \tag{1.26}$$

$$RP \Leftrightarrow \left| w_p S \right| + \left| w_I T \right| < 1 \tag{1.27}$$

It can be noted that the robust performance must satisfied the robust stability and nominal performance conditions[31].

1.3.8 Weighting functions

Consider the following feedback control loop in the presence of uncertainties:

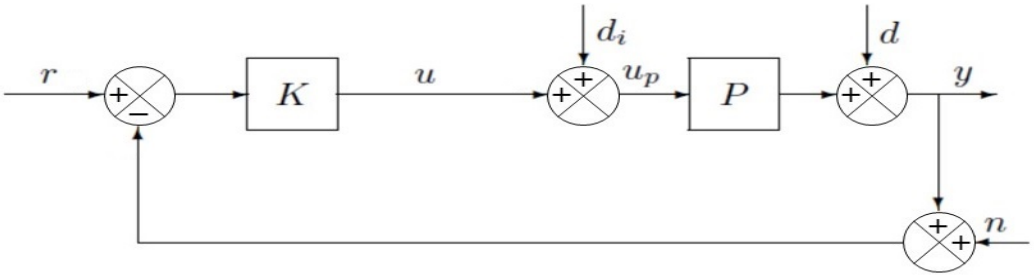

Figure 1.6: Standard Feedback Configuration

1.3.8.1 Performance Weighting function

To regulate the output, following the reference signal and rejecting the disturbance. And to improve the performance by minimizing the steady state error of the plant, the following weighting function was suggested (figure 1.7)[31], [10]:

$$W_e = \frac{{}^{s}/_{M_e} + \omega_e}{s + \varepsilon \omega_e} \tag{1.28}$$

Where:

ε : is the steady state error.

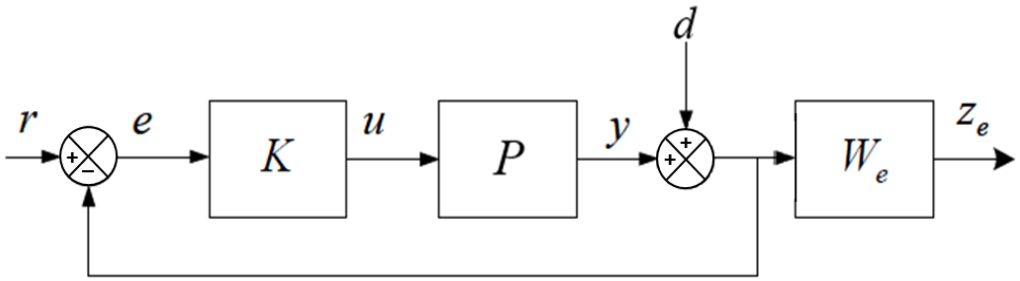

Figure 1.7: Feedback system with output sensitivity weighting function

M_e and ω_e parameters are shown in the figure below(figure1.8):

The choice of an appropriate weighting function W_e optimized the frequency response

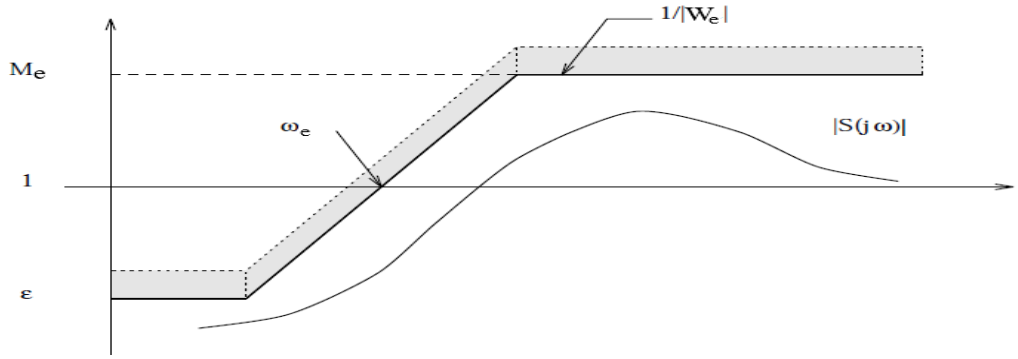

Figure 1.8: Performance weighting function Bode plot [10]

of the sensitivity function S and the performance of the plant, furthermore the condition $\|W_e S\|_\infty \leq 1$ is hold.

1.3.8.2 Control signal Weighting function From the standard closed loop system perturbed by the signal d_i on the controlled signal, the signal d on the output and the signal n on the measured signal. the control signal can be written as in the equation below:

$$u = KS(r - d - n) - Td_i \tag{1.29}$$

The magnitude of $|KS|$ in the low frequencies is limited by an acceptable cost of control effort and actuator saturation. To reduce the noises and disturbances at control signals in high frequencies which is limited by the controller bandwidth. A candidate weight is suggested W_u [31], [10]:

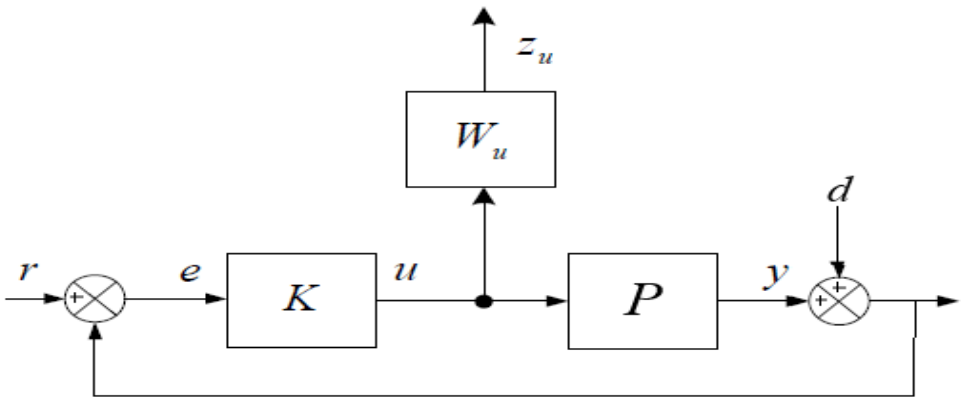

Figure 1.9: Feedback system with output sensitivity weighting function

$$W_u = \frac{s + {}^{\omega_{bc}}/_{M_u}}{\varepsilon_1 s + \omega_{bc}} \tag{1.30}$$

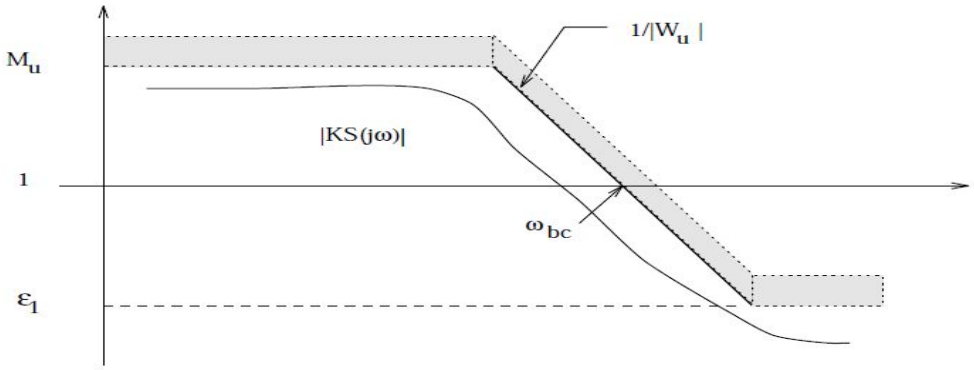

Figure 1.10: Control weighting function W_u Bode plot [10]

The good choice of W_u weighting function limits the magnitude of control signal and saves the energy of system.

1.3.8.3 Complementary sensitivity Weighting function The complementary sensitivity weighting function as it is illustrated in the figure(1.11) is designed to reduce the noise and disturbances, it is given by: following[31], [10]:

$$W_y = \frac{{}^{s}/_{M_y} + \omega_y}{s + \varepsilon_2 \omega_y} \tag{1.31}$$

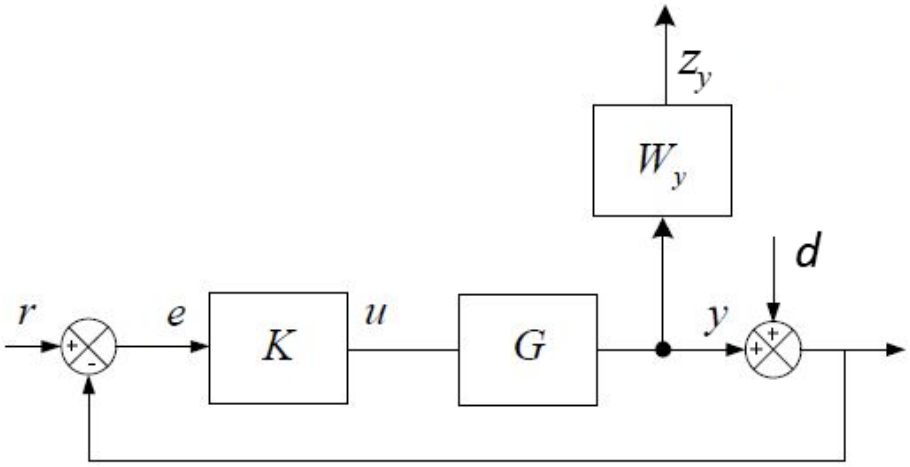

Figure 1.11: standard configuration of complementary sensitivity function

Example

Consider the nominal system of third order given by the following transfer function $P(s) = \frac{1}{(1+s)^3}$, we calculate the robust controller C that minimizes the H∞ norm of the closed loop feedback system for mixed sensitivity, that should satisfy the conditions below:

1. The gain crossover pulsation w_o range is between 1 and 10 [rad/s]
2. Steady state error is equal to 1/1000
3. Sensitivity pick is $\|S\|_\infty < 2$
4. The controller gain must be minimized when $w > 10w_o$

The conception of this controller is made using Matlab function "mixsyn"
The weighting functions are adjusted as the following:
Given: $w_o = 10$.
$W_e = \frac{s+12}{1.2s+0.012}$, $W_u = \frac{s}{0.9555s+9.555}$, $W_y = \frac{1}{W_e}$.
The controller H∞ is given by the transfer function:
$K = \frac{0.287s^3+278.4s^2+6010s+0.000321}{s^3+281s^2+2906s+29.03}$.

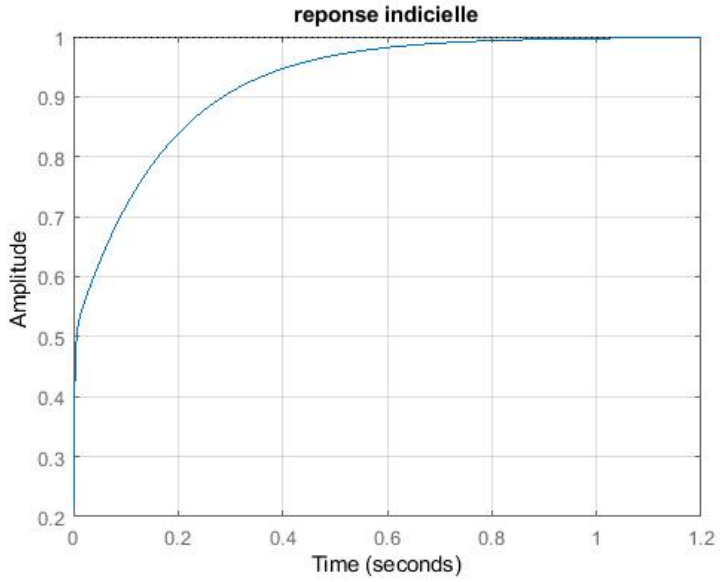

Figure 1.12: System response

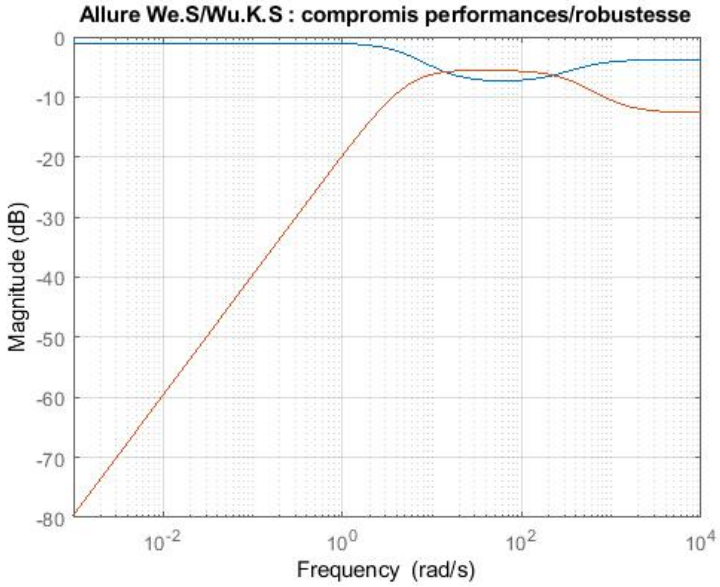

Figure 1.13: Stability and performance conditions singular values

it's clear that $\left\| \begin{matrix} W_e S \\ W_u KPS \end{matrix} \right\|_\infty < 1$ that means the controller calculated guarantees the mixed sensitivity concept.

1.4 Optimal Control Theory

The conception of optimization in control was first utilized in the 1960s, seeking the solution to convex, least squares optimization problems [32]. Optimal control aims to achieve a system with the best performance index or a designated criterion commonly denoted as J. This criterion replaces conventional design parameters such as gain, phase margins, damping ratio, and overshoot [33]. The Linear Quadratic Regulator (LQR) stands out as an optimal controller, requiring the system's state space, which describes the system's dynamics, to formulate and optimize the control law.

1.4.1 Optimal Control Theory Problem

Consider the system equation

$$\dot{x} = Ax + Bu \tag{1.32}$$

The block diagram giving the optimal configuration is in the figure below:

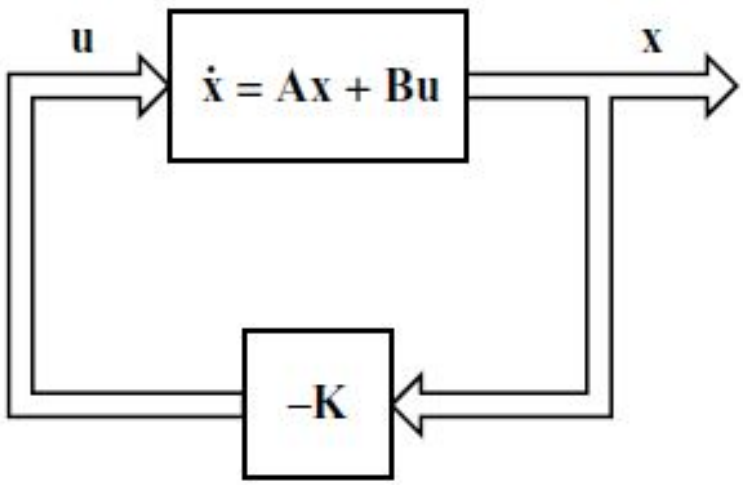

Figure 1.14: Quadratic Optimal Regulator systems

To determine the matrix K of the optimal control vector where:

$$u(t) = -Kx(t) \tag{1.33}$$

We should minimize the performance index[34]

$$J = \int_0^\infty \left(x^T Q x + u^T R u\right) dt \tag{1.34}$$

Where Q and R are positive-definite (or positive-semi-definite) hermitian or real symmetric matrices. The term $u^T Ru$ represents the energy of the control signal. The matrices Q and R represent the relative error and the expenditure of this energy.

1.4.2 Solution of the Optimal Control Theory Problem

From the equation (1.28) and (1.29), we can write:

$$\dot{x} = Ax - BKx = (A - BK)x \tag{1.35}$$

The eigenvalues of matrix $A - BK$ should have negative real parts (i.e. stable closed loop). Then, We can write:

$$J = \int_0^\infty \left(x^T Q x + x^T K^T R K x\right) dt \tag{1.36}$$

Hence

$$J = \int_0^\infty \left(x^T \left(Q + K^T R K\right) x\right) dt \tag{1.37}$$

we set:

$x^T \left(Q + K^T RK\right) x = -\frac{d}{dt}\left(x^T Px\right)$

Where P is a positive-definite Hermitian or real symmetric matrix. We obtain then:

$$x^T \left(Q + K^T RK\right) x = -\dot{x}^T P - x^T P\dot{x} = -x^T \left[(A - BK)^T P + P(A - BK)\right] x \tag{1.38}$$

From the equations(1.33) and (1.34) we can write

$$(A - BK)^T P + P(A - BK) = -\left(Q + K^T RK\right) \tag{1.39}$$

It was proved that if $(A - BK)$ is a stable matrix so a positive-definite matrix P that satisfies the equation (1.35) exists. The next procedure is to determine the elements of the matrix P from the matrix (1.35) and see if it's positive definite.

The performance index J can be evaluated as:

$$J = \int_0^\infty \left(x^T \left(Q + K^T R K\right) x\right) dt = -x^T P x\Big|_0^\infty = -x^T(\infty) P x(\infty) + x^T(0) P x(0) \tag{1.40}$$

As the matrix $(A - BK)$ has negative real parts, so $x(\infty) \to 0$, therefore:

$$J = x^T(0) P x(0) \tag{1.41}$$

Thus, the performance index J can be obtained from the initial condition of x and P .
In order to obtain the solution of the quadratic optimal control problem, we pursue as follow:
First, we put the positive-definite Hermitian or real symmetric matrix R in the form $R = T^T T$, where T is a non-singular matrix, than the equation (1.35) can be written as:

$$(A - BK)^T P + P(A - BK) + \left(Q + K^T T^T T K\right) = 0 \tag{1.42}$$

as it can be written as:

$$A^T P + PA + \left[TK - \left(T^T\right)^{-1} B^T P\right]^T \left[TK - \left(T^T\right)^{-1} B^T P\right] - PBR^{-1} B^T P + Q = 0 \tag{1.43}$$

So the minimization of the he performance index J requires the minimization of the term $x^T\left[TK - \left(T^T\right)^{-1} B^T P\right]^T \left[TK - \left(T^T\right)^{-1} B^T P\right] x$ .That can be reduced to zero when:
$TK = \left(T^T\right)^{-1} B^T P$.
Hence:

$$K = T^{-1}\left(T^T\right)^{-1} B^T P = R^{-1} B^T P \tag{1.44}$$

Therefor the optimal control law of the quadratic optimal control problem is the following[34]:

$$u(t) = -Kx(t) = -R^{-1} B^T P x(t) \tag{1.45}$$

We can summarize the solution of the quadratic optimal control problem in two steps:

1. we solve the Riccati equation to determine the matrix P .

2. we substitute the matrix P into the equation (1.40) to find the optimal matrix K , hence the optimal control law is found.

1.5 Conclusion

The aim of this chapter is to provide a comprehensive overview of robust control theory, with a particular focus on H∞ controller synthesis, as most of the control methods developed in this thesis are rooted in this approach. The discussion covers essential aspects of robust stability, robust performance, and the utilization of weighting functions. Moreover, a practical demonstration of H∞ synthesis is presented through a straightforward example.

In the latter part of the chapter, priority is given to introducing fundamental concepts of optimal control. The objective is to deepen the researcher's understanding of these control techniques, laying a strong groundwork for the subsequent exploration within the thesis.

Chapter 2

Bio-inspired Control

2.1 Introduction

The perfection in the nature creation, its discipline, its rules that organize, control all the organisms and ensure the suitable behavior of them. All of that make of nature a source of inspiration and provide a noteworthy artificial intelligence algorithms which can be used in finding the optimal path, recognition of patterns, invention of several robust and optimal control laws...etc, we sit the example of the packs of wolves that hunt together and the migration of birds in flocks together, this notion was inspired in designing virtual organisms to solve a problem together. The optimization inspired from ants colony, where the pheromone trails from ants facilitate to find the optimal path by leaving strong trails on it[35]. The biological immune system is a robust mechanism that uses several ways of defense to protect the body from foreign particles called antigens, it was inspired and it gave a computational intelligence paradigm which is known by the artificial immune system [36]. As well as the artificial neural network imitated from the biological neural and takes some of its characteristics such storing knowledge, transmitting the information from one neuron to another to achieve an action, besides its capacity of learning [37]. Nowadays, neural networks represents the most powerful technique in artificial intelligence.

The following chapter represents the Artificial Immune System and the Artificial neural network in details and their interest in controlling systems.

2.2 An Overview of the Biological Immune System

The immune system is a mechanism represents a complex of cells, molecules and organs that work in coordination capable to distinguish self and non-self elements, when a pathogen invade the body it will be detected and mobilized for the elimination. Furthermore, the immune system has the ability of remembering, so if the pathogen enters for the second time, the reaction will be faster and it won't waste any time to recognize the antigen and choose the appropriate way of defense [38], [39]. The immune response is divided into innate immunity and adaptive or acquired immunity which can be humoral response or cell mediated response.

2.2.1 Innate Immunity :

The innate immunity is considered the first line of defense, it includes the physical barriers such as the skin, the lysozymes in tears, nasal hair and mucus...etc. If the pathogen penetrates the first barriers, this stimulate a non specific immune response by macrophages and neutrophils, that ingest the antigen to get red from it and to present fragments of its protein to other immune cells.

2.2.2 Adaptive Immunity :

Adaptive immunity or as it called also acquired immunity, is characterized by the specificity in its reaction against the antigen, and the white blood cells (lymphocytes) the main element in this immunity response will be changed into memory cells after getting rid from the foreign pathogen. this characteristic helps to recognize rapidly this pathogen if it invades the body for another time. The two major types of lymphocytes are T cells and B cells, T cells enters in the mediated immunity response when the antigen invade a cell of the host. they can take the form of helper T cells which help the B cells and stimulate their production of antibodies, the other form is cytotoxic T cells which specialized in the recognition and the elimination of the infected cells by viruses or cancer. B cells are the basic of the humoral immunity, they occupy the function of producing the specific antibodies of the invaded antigen detected by their receptors [36, 40].

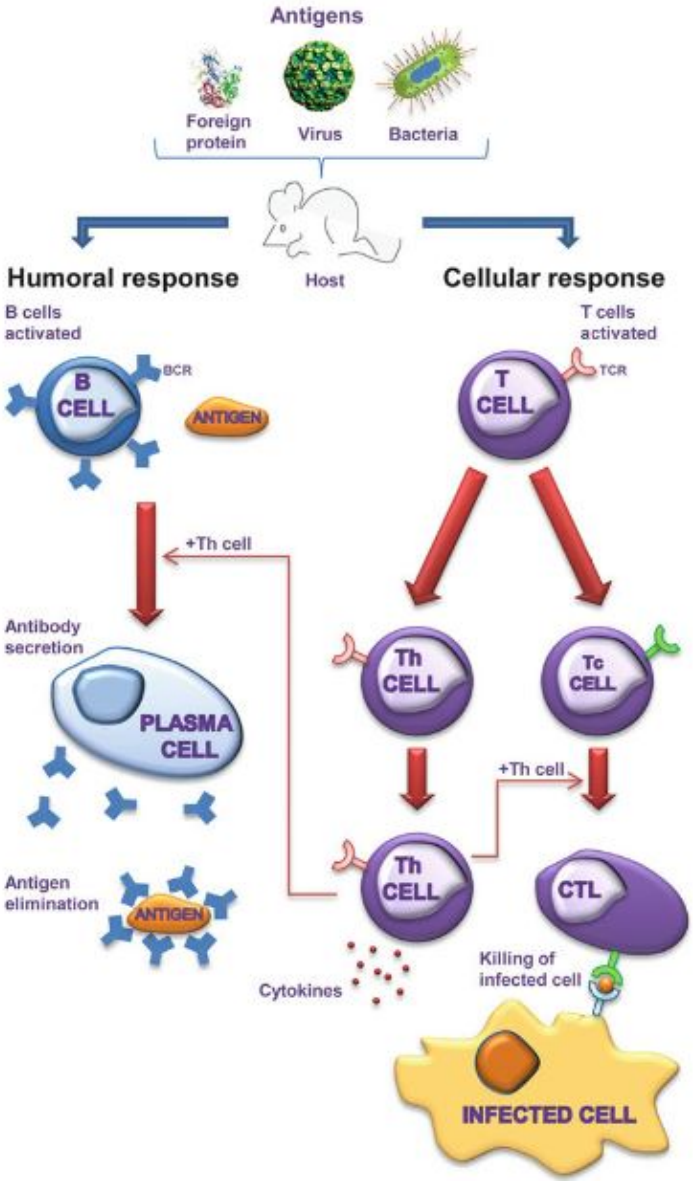

Figure 2.1: Adaptive immune system[11]

2.3 Artificial Immune System

Artificial Immune System (AIS) is a paradigm elaborated for manipulating data, classify and reasoning ideas as the natural immune system do. As it is defined as the artificial immune system is a computational model based upon metaphors of the biological immune one. The artificial immune system is intended to solve problems of optimization, controlling systems and robot navigation [36].

2.3.1 Immune Feedback control :

In order to enhance the quality of the classical control, An optimal and non linear con-

troller was designed according to one of the defense methods using by the immune system, where this last is formulated into equations that give in the end a robust PID controller.
As it is cited above, the regulation of the immune system in the case of an invasion by a foreign pathogen which transgresses the first line of defense and induces an humoral response, is assured by the lymphocytes T cells and B cells and according to the antigen amount in the organism. As long as the existence of the antigen, the helper T cells stimulate to produce more antibodies, but when the antigen amount is reduced, the helper T cells amount will be reduced also, and the suppressor T cells will increase to inhibit the B cells for more creation of antibodies. So the immune system will come back to the stability.

Basing on this concept of the biological immune defense method, in the generation k of

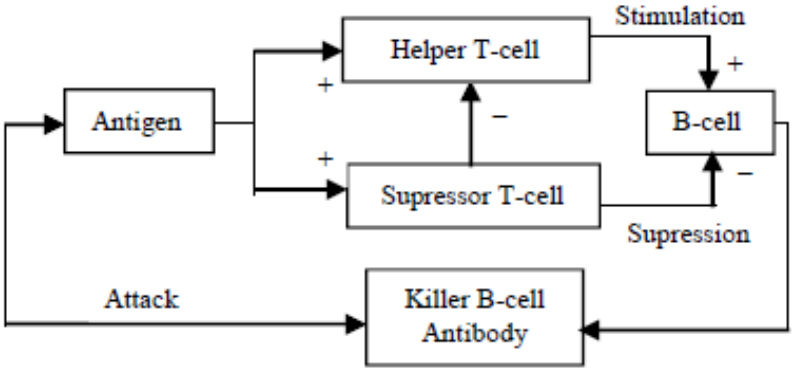

Figure 2.2: Immune regulating system[12]

an antigen, B cells receive the stimulation from the helper T cells (T_H), and the inhibition from the suppressor T cells (T_S), so we can write the equation below[17]:

$$S(k) = T_H(k) - T_S(k) \tag{2.1}$$

where $T_H(k)$ and $T_S(k)$ expressions are given by the following equations:

$$T_H(k) = k_1 \varepsilon(k) \tag{2.2}$$

$$T_S(k) = k_2 f(S(k), \Delta S(k)) \varepsilon(k) \tag{2.3}$$

$\varepsilon(k)$ is the antigen amount in a generation k .
k_1 , k_2 are respectively the stimulating and the inhibitory factors.
by replacing the equations (2.2) and (2.3) in (2.1) we will have the equation below:

$$S(k) = K\{1 - \eta f(S(k), \Delta S(k))\}\varepsilon(k) \tag{2.4}$$

$K = k_1$.

$\eta = k_2/k_1$ is a factor related to the stability of the system.

f is a nonlinear function which represents the inhibition ability to the external stimulation, it is based on simulation, according to the influence on antigen consistency on the antibodies under the effect of T cells regulating action. It's given by the equation below[41]:

$$f(x) = 1 - \frac{2}{1 + \exp(-ax)} \tag{2.5}$$

Figure(2.3) represents the behavior of this function according to the parameter a variation. This is similar to the classical control law

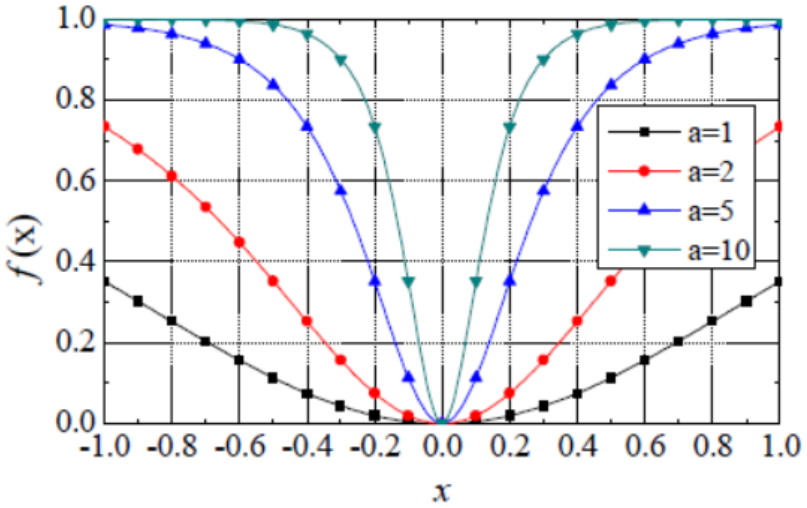

Figure 2.3: The effect of a factor on the behavior of the function f[13]

$$u(k) = K_p e(k) \tag{2.6}$$

where the gain is non linear $K_p = K\{1 - \eta f(S(k), \Delta S(k))\}$.

So we can write the immune feedback control law as the following [17]:

$$u(k) = K\{1 - \eta f(u(k), \Delta u(k))\}e(k) \tag{2.7}$$

B cells amount variation is the control signal and the antigen amount represents the error [42]. K and η are two important parameters that characterize the immune feedback

control, for having a fast response by increasing K parameter, η factor will reduce the overshoot of the system's output. So a reasonable adjustment of these two parameters will guarantee a faster response with a limited overshoot. [43].

Table 2.1: Comparison between the biological immune system and the immune feedback control[12]

Biological immune system	Artificial immune system
1. The antigens and antibodies reproduction Kth generation 2. $\varepsilon(k)$ the antigen amount in a generation K 3. $S(k)$ the antibodies concentration in a generation K	1. The Kth sampling time of discrete artificial immune system 2. $e(k)$ the error between the set value and the system's output 3. $u(k)$ the control law delivered by the controller in the generation K

2.4 Neural Networks

The story of the artificial intelligence began with the appearance of the neural networks concept, when McCulloch and Pitts developed a basic neural network according to their understanding of neurology in 1943 that can be used just for formal logic simulations. The researches was continued in this field until the years of 1950s, Rosenblatt could realize the perceptron model, which includes an input layer, an output layer, connected by weights that have to be adjusted for a desired output and this helped the perceptron to learn and recognize individuals. In the 1960s, Widrow and Hoff developed another learning model used for calculating mean-squares to adjust the weights of the network, as it can be implemented in a physical plant. In the year of 1969, Marvin Minsky and Seymour Papert wrote a book about perceptrons limited in single layer models, which was generalized to multi-layer perceptron, then Paul Werbos who emerged the back propagation algorithm [44]. Nowadays, neural networks field grants successful practical results and facilitates many operations in several sectors especially in controlling systems and optimization.

2.4.1 Biological Neuron and Artificial Neural Network :

As the artificial neural network is imitated from the biological nervous system, it worth

firstly identifying the biological neural network, its main parts and how it works.
The human brain consists of a 100 billion of neurons, a neuron is characterized by inputs called dendrites and outputs named axons. The incoming signals from the other neurons and throw the dendrites, will be treated and summed in the nucleus of the cell body, the generation of voltage impulse is well done only if the resulting signal between stimulating and inhibitory effects exceeds the threshold, in this case it will be transmitted via the axon to the other neurons. Neurons are joined by electro-chemical junctions known by synapses. The artificial neural network has the same architecture as the biological neuron. Inputs modeled from synapses, each one is multiplied by a weight. These signals are sent to a unit called threshold logic unit (TLU) similar in function to the cell body of the biological neuron. The resulting signals are compared to a given threshold, if they are larger than the threshold that leads to the activation, otherwise the output is 0 [45]. The figure below shows both of the biological neuron (a) and the artificial neural (b).

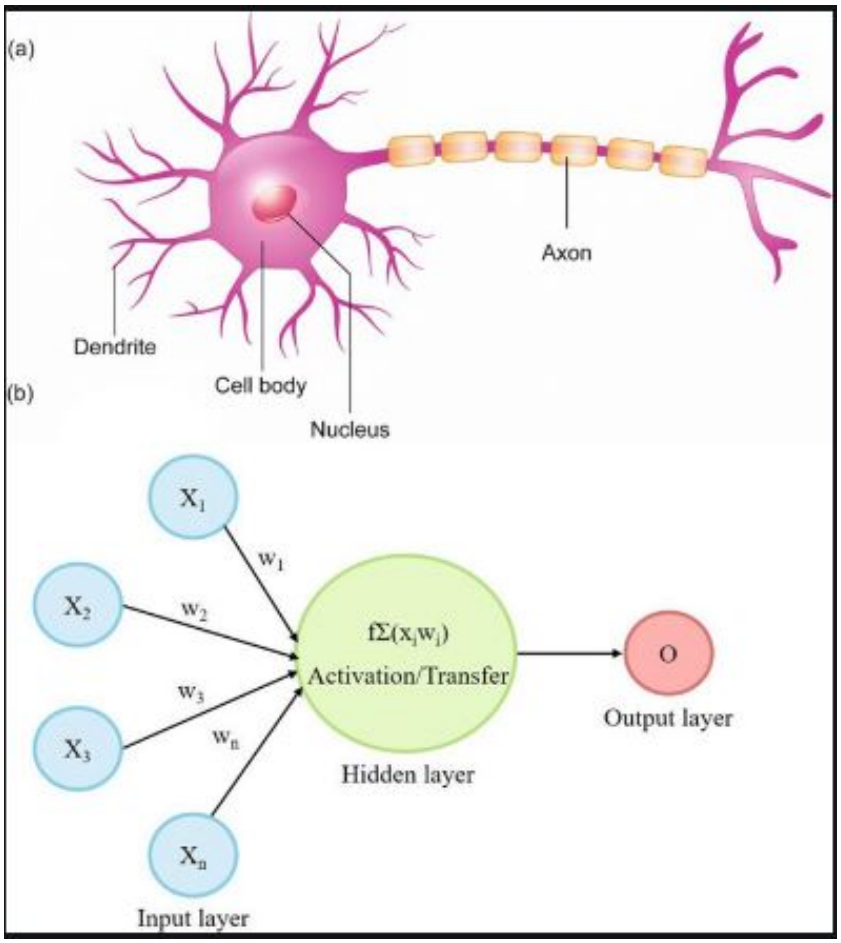

Figure 2.4: Biological neuron and artificial neuron[14]

The neural network is trained by adjusting the weights $(w_1, w_2, w_3, ...w_n)$ as in the figure

above until its output becomes similar to the chosen function. The nodes in the input are called the input layer, where the final output is called the output layer.

2.4.2 Neural Network Structure

The basic neural network is composed of only an input layer and an output layer, its called "single layer neural network", then it has been evolved to more and more complex architecture by adding hidden layers and formed " the multi layer neural network". The neural network which contains only one hidden layer takes the name "shallow neural network or vanilla neural network". However, the neural network which has two or more hidden layers is known by deep neural network, this last is the most used in practical applications. The activation function in the multi layer neural network must be different to the linear function, or the hidden layers will be ineffective and the neural network will take the form of the single layer neuron network [37].

2.4.3 Neural Network Learning :

Neural networks learn from the training occurred by the modification of synaptic strengths or weights, throw several kinds of learning rules, each one of them has its own functionality and model. All the weights are changed according to the same learning rule. Furthermore, neural networks have the ability of learning from examples, demonstrations...etc. Neural networks have also other noteworthy features:

- Robustness as they can be trained to compensate the damage or noise in hardware implementations, besides that neural networks maintain the desired response by increasing or decreasing the error rate when the noise increases.

- Flexibility : neural networks are flexible, they can be applied to a lot of problems. accept the problems which have an analytical solution.

- Generalization: the results obtained from the training of a given input patterns can be generalized to other new patterns that share some similarities with the training ones.

- Content-based retrieval: Neural networks restore memories by matching the existing data, they can even retrieve the uncompleted or corrupted input patterns effected by noise.

[39] Neural networks learning is acquired according to three major families: supervised learning, unsupervised learning and reinforcement learning.

2.4.3.1 Supervised Learning: This learning model aims to find a function yields the correct output for a given input pattern, otherwise, this kind of learning should pursuit target values. The training is made by calculating the outputs of the neural network and the expected one of the given data and comparing them. Weights are adjusted according to the error Supervised learning treats two major problems classification and regression. Examples of supervised learning: perceptron learning algorithm, back-propagation and least mean squares learning [36, 44].

2.4.3.2 Unsupervised Learning: This kind of learning is achieved without any associated target values, that means the training of data is occurred with no given answer. The clue used in the unsupervised learning is the analyze of data and trying to understand the similarities and the differences. Examples of the unsupervised learning: adaptive resonance theory, Hebbian learning and Kohonen self organizing maps [36, 44].

2.4.3.3 Reinforcement Learning: is designed to maximize a numerical reward signal for a defined action, this class of learning estimates the values estimated by agents explicitly or implicitly in order to choose the actions that maximize the positive reinforcement amount achieved by the agents during a period of time. So the reinforcement learning aims to find the actions which yield the most reword [46].

2.5 Conclusion

This chapter was devoted to introduce two bio-inspired control techniques the artificial immune feedback control and the artificial neural network, as these techniques were inspired from the biological immune systems and the human brain, the analogy between the biological systems and the artificial ones was presented, the main PID immune feedback control law is emerged, besides an overview of the the existed learning model used by the neural network and their application. In the next chapters we intend to robustify these control techniques with the conventional control techniques such the examples presented in the first chapter.

Chapter 3

H∞ based PID Immune Feedback Controller Design

3.1 Introduction

As it was described in the first chapter, the discrepancy between the real system and its mathematical model is unavoidable, and that will degrade the system performances including the robustness. For that, we propose a robust PID controller where its parameters are calculated using the H∞ synthesis and the immune feedback block is identified and implemented as a weighting function.

In the following chapter we will state the full method adopted in this work, and for its application, this technique will be applied for the control of a magnetic levitation system (Maglev) which has taken a crucial interest in these last years, this study takes into account the parametric uncertainties occurred while the identification of the system, and finally the performance robustness test is examined to analyze the adequacy of the controller proposed.

We will introduce the PID immune feedback controller law and the structured H∞ synthesis, the simulation results given by the PID immune feedback controller and the H∞ based PID immune feedback controller are compared and discussed.

3.2 Structured H infinity Controller

Despite the advantages of the H∞ synthesis, its engineering application faced many limitations, owing to the complexity of the controller adopted and its high order. A new structured controller was proposed by Apkarian in the last years, the design procedure is beginning with the controller structure which is fixed according to the control requirements, the selection of the appropriate weighting functions is necessary for an H∞ performance matrix with multi-dimensional performance output and the optimization of the controller parameters is achieved in the end[47, 48, 49].
The obtained structured H∞ controller based on non-smooth optimization was proven its effectiveness due to many reasons such as keeping the controller order low, using optimization algorithms to parameterize the parameters of the PID controller and to specify the structure of the controller[50].

3.3 Application

3.3.1 System description Magnetic levitation system is an advanced technology in which the magnetic field is generated to keep the levitation of an object without any support, in the presence of gravity force and parametric uncertainties[51].

The magnetic levitation system (or as it called Maglev) is widely applied in several disciplines, electricity, electronics and mechatronics, it was the main idea of friction-less bearing and the innovative high speed train, which currently used in the most of developed countries[52].
As the Maglav control challenge is to adjust the steel ball position, an infra-red sensor is used to measure the actual vertical ball position, this value is transferred to a computer (the control unit) which sends a voltage between $[-5V, 5V]$, converted into current by a driver embedded to the unit, this last passes throw a coil, hence the production of a magnetic field [53], the mechanical unit is shown in the figure below (Figure 3.1).

In general, electromagnetic systems are characterized by an unstable behavior and the nonlinear modeled equations. Therefore, most control approaches are made according to

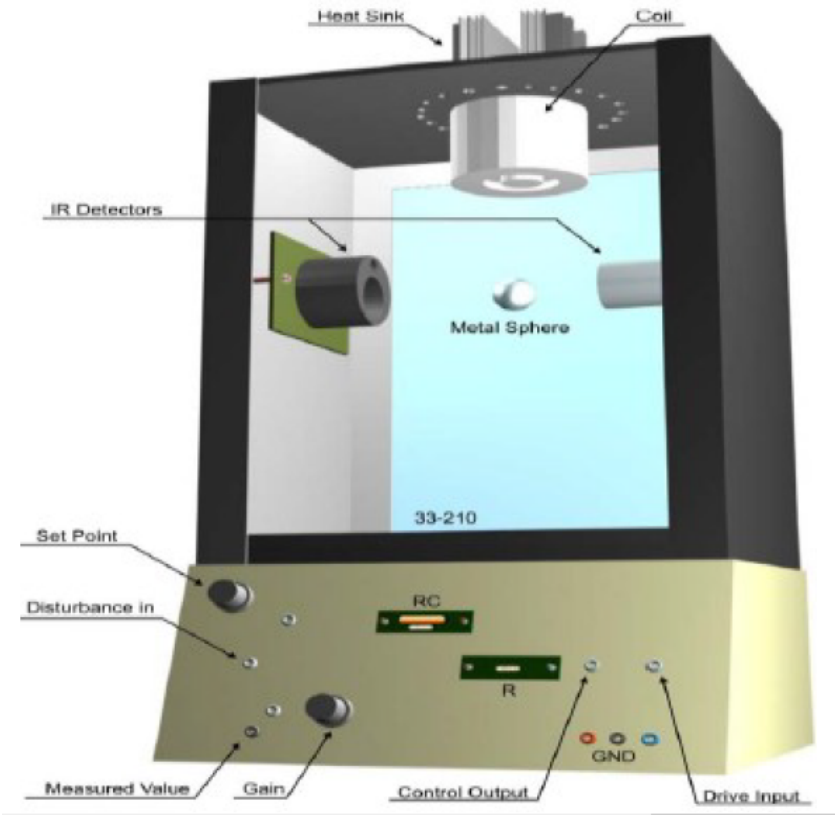

Figure 3.1: Maglev Mechanical Unit [15]

the linearized model around a given operating point[54].

From the circuit above, and using Newton and Faraday laws we can write:

$$\frac{d^2x}{dt^2} = mg - F_{em} \tag{3.1}$$

$$V = R_c i + L_c \frac{di}{dt} \tag{3.2}$$

Where:

$i =$ electric current [A]

$x =$ ball position [m]

$m =$ Mass of the ball [Kg] $g =$ gravity [m/s^2]

$R_c =$ coil resistance [Ω]

$L_c =$ coil inductance [H]

Using the theorem of the generalized force, the electromagnetic levitation force is written as:

$$F_{em} = -\left[\frac{\partial \varpi_m}{\partial x}\right] \tag{3.3}$$

where the magnetic energy is given by the equation below:

$$\varpi_m = \left[\frac{1}{2}Li^2\right] \tag{3.4}$$

Assuming that the relation between coil inductivity and the position of the steel ball is as follow:

$$L(x) = L_1 + \frac{2K}{x} \tag{3.5}$$

where:

$L =$ inductivity [H]

$x =$ the ball position [m]

$L_1 =$ inductivity when the ball is at the point $(x = 0)$

$K =$ coil constant [$N * m^2/A^2$]

From the equations (3.4),(3.5), F_{em} can be written as:

$$F_{em} = K\left[\frac{i^2}{x^2}\right] \tag{3.6}$$

At the equilibrium point ($x = x_0$ and ($i = i_0$), the current is constant so its derivative is becoming null.

So from the equation (3.2), i_0 becomes:

$$i_0 = \frac{V}{R} \tag{3.7}$$

As well, the acceleration is null, so the equation (3.1) becomes:

$$mg = K\left[\frac{{i_0}^2}{{x_0}^2}\right] \tag{3.8}$$

so:

$$x_0 = i_0\sqrt{\frac{K}{mg}} \tag{3.9}$$

To linearize the system, Taylor's series is applied, and by ignoring the higher order terms, we find:

$$\frac{d^2x}{dt^2} = \left|-\left(\frac{\partial f(i,x)}{\partial i}\right)\right|_{(i_0,x_0)\Delta i} - \left|-\left(\frac{\partial f(i,x)}{\partial x}\right)\right|_{(i_0,x_0)\Delta x} \tag{3.10}$$

The Laplace transformation of the equation (3.10) is given y the following equation:

$$s^2\Delta x = -\left(K_i\Delta i + K_x\Delta x\right) \tag{3.11}$$

$$\Delta x\left(s^2 + K_x\right) = -K_i\Delta i \tag{3.12}$$

$$frac\Delta x\Delta i = \frac{-K_i}{s^2 + K_x} \tag{3.13}$$

Which represents the transfer function of the system, where:

$$K_i = \frac{\partial f\left(i,x\right)}{\partial i} = \frac{K.2i}{\partial i} = \frac{2mg}{i_0} \tag{3.14}$$

$$K_x = \frac{\partial f\left(i,x\right)}{\partial x} = -\frac{K.2i^2}{mx^3} = -\frac{2mg}{x_0} \tag{3.15}$$

According to the measurements achieved in [15]

$R_c = 21.5\Omega$

$K = 1.477 \times 10^{-4}$ [$N * m^2/A^2$] [Given]

$m = 0.021$ kg [Given]

$L = 0.711$ H

$i_0 = 0.23$ A

$x_0 = 0.15 \times 10^{-2}$ m

$K_i = 0.5145$

$K_x = 0.274$

$$\frac{\Delta x}{\Delta i} = \frac{-0.5145}{s^2 + 0.274} \tag{3.16}$$

The block diagram of the controlled system is as follow (figure 3.2):

3.3.2 Controlling Maglev system with PID immune feedback controller

As it was descried above in chapter 2, the artificial immune feedback control aims to improve the control quality and inspire some of the characteristics of the biological immune system, strong robustness and self adaptability. The immune feedback controller which is modeled by a non linear proportional action, is merged by the classical PID controller to overcome its weakness in noise tracking and steady state error slaked, this involve a robust PID immune feedback controller illustrated in the figure be-

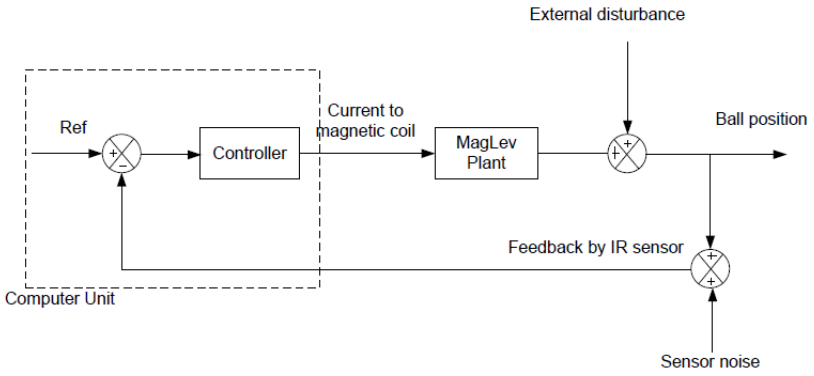

Figure 3.2: Closed loop response of the uncertain system[16]

low (figure 3.3):

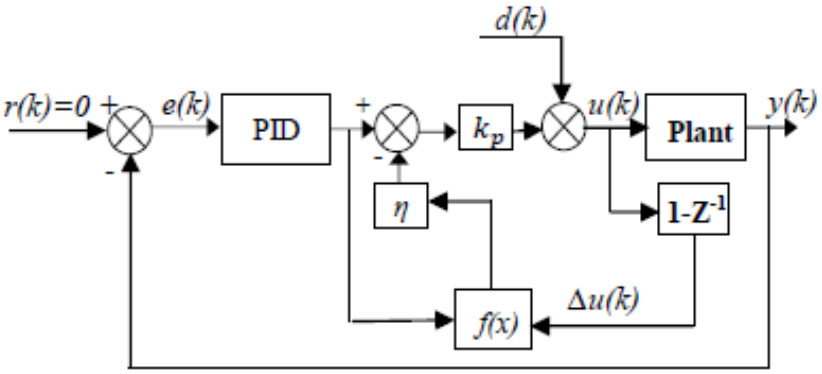

Figure 3.3: PID immune feedback control[17]

The PID immune feedback control law is given by the equation below:

$$u(k) = u(k-1) + K_p(e(k) - e(k-1)) + K_i e(k) + K_d(e(k) - 2e(k-1) + e(k-2)) \tag{3.17}$$

Where:

$K_p = K\{1 - \eta f(u(k), \Delta u(k))\}$.

K_i and K_d are the integral and differential coefficients.

Controlling Maglev system by the artificial immune feedback control is executed in

Matlab simulink where the parameters of the PID controller are adjusted using ITAE optimization criterion besides the two factors of the artificial immune bloc η and a. Where the Integral Time Absolute Error (ITAE) criterion is based on the minimization of the tracking error in a limited time, its mathematical expression is the following [55] :

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

Then, the optimized PID parameters K_p , K_i and K_d are expressed as follow:

$$MIN_{K_p,K_i,K_d} \int_0^\infty t\,|e(t)|dt \tag{3.19}$$

The control loop in Matlab/ Simulink is the following (figure 3.4):

After many simulations, the best response optimized using ITAE algorithm is in the figure

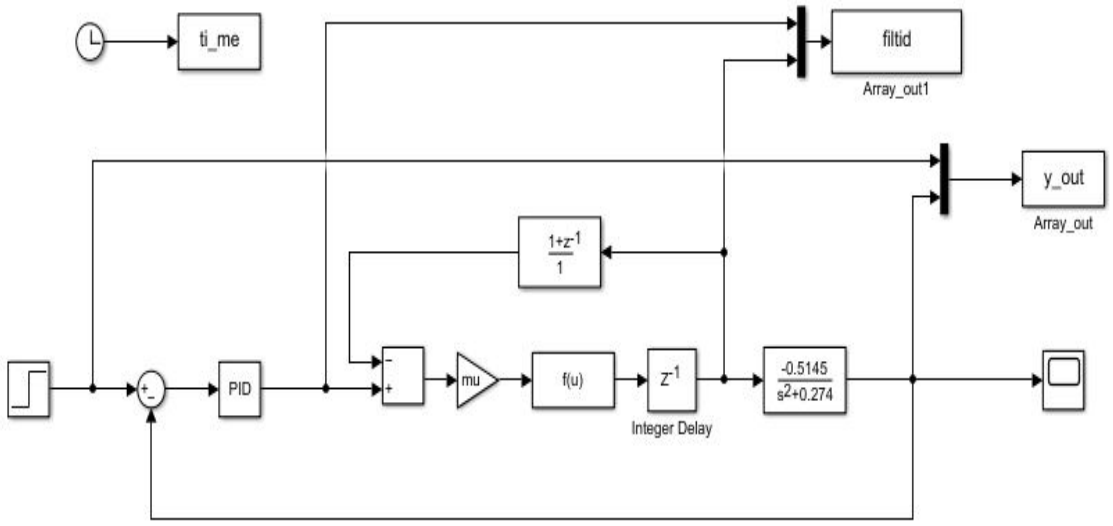

Figure 3.4: PID immune feedack control

(figure 3.5), Where, the final parameters of the PID controller and the optimized bloc are fixed in the end at : $\eta = 1.3352$, $a = 0.7507$, $K_p = 1.8823$, $K_i = 0.1780$, $K_d = 1.3411$.

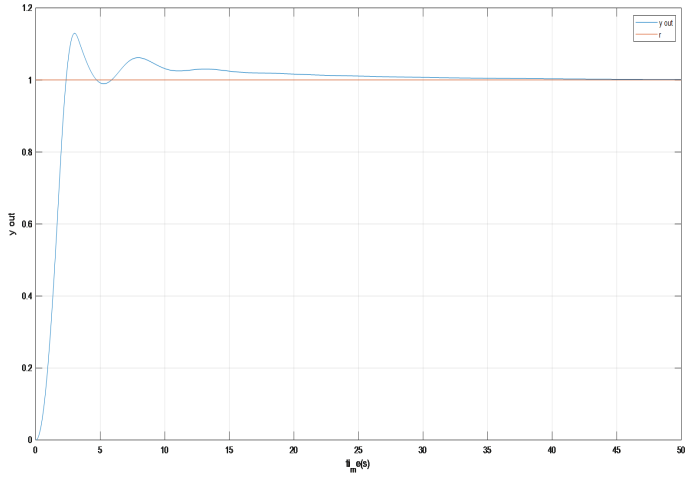

Figure 3.5: Closed loop response

3.3.3 Controlling Maglev system with structured PID immune controller

In the following section, the alternative method of the H∞ norm known by the H∞ structured control is adopted. However, the hinfstruct algorithm minimizes the H∞ norm throw multiple optimizations started from random generated points, by selecting the desired order and structure of the controller.

We suggest the PID configuration for the controller's structure with adjustable parameters given by the equation below:

$$C(s) = K_p + \frac{K_i}{s} + K_d \frac{s}{T_f.s + 1} \tag{3.20}$$

3.3.3.1 Weighting function selection

In order to enhance the performances of Maglev system obtained by the artificial immune feedback controller and in the presence of parameters uncertainties, the artificial immune block found in the previous section is identified using the function "ident" of Matlab, the behavior of the immune block and its identification is shown in the following figure (3.6):

The identified function is a low pass filter of the second order, to reduce the error sensitivity in the low frequencies for the best disturbance rejection, it is used as a performance weighting function appending to the nominal system to obtain the corresponding augmented system, its expression is given by the equation below:

$$W_t = \frac{-0.7659s^2 - 0.9978s - 0.01028}{s^2 + 0.6583s + 0.002325} \tag{3.21}$$

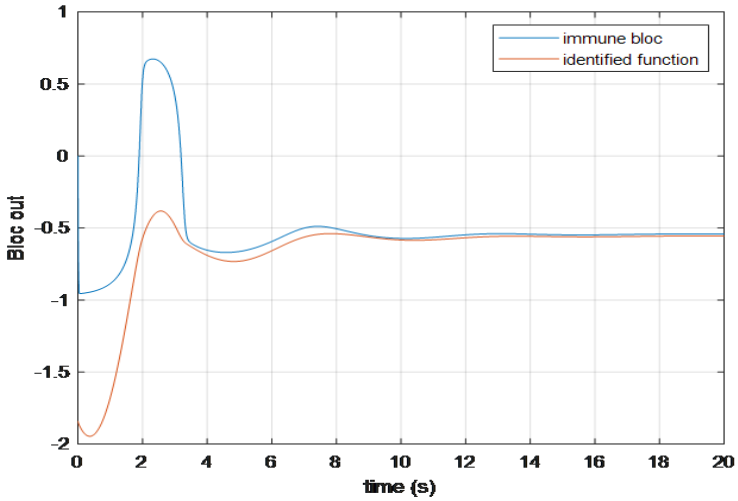

Figure 3.6: Immune bloc and its identification behavior

According to Matlab simulations, the best achieved H∞ norm is $gam = 0.7656$ and the closed loop system is shown in the following figure (3.7):

Figure (3.8) shows the frequency response of the sensitivity function S and the inverse of

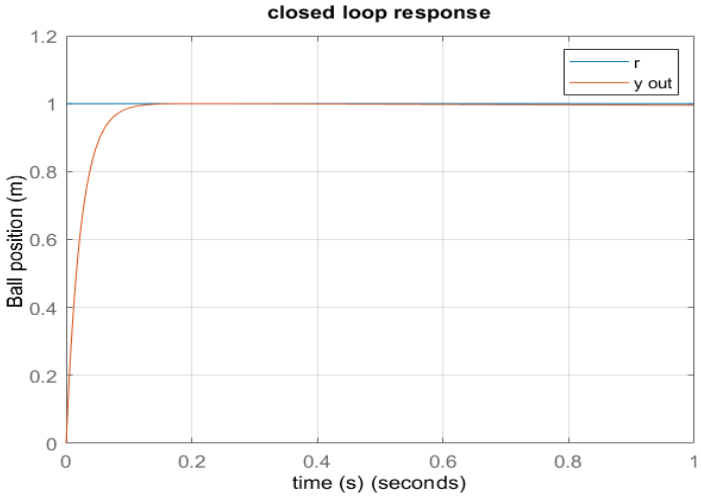

Figure 3.7: Closed loop response

the performance weighting function Wt, it is obvious that the magnitude of S is less than the magnitude of Wt^{-1} for all frequencies which means that the nominal performance has been achieved.

In order to demonstrate the control performance robustness and to investigate the impact of the parametric uncertainties on the identified model of Maglev system, an uncertainty of 10 % is added to the all system parameters, throw matlab plot, the system's output is varying as in the following figure (3.9):

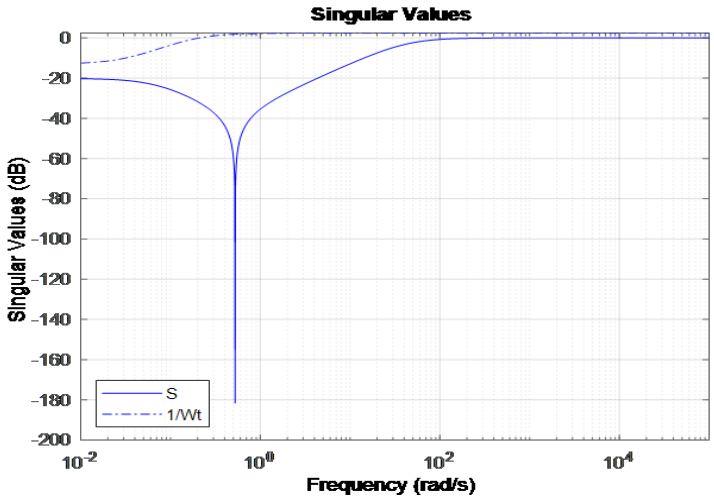

Figure 3.8: sensitivity function Bode diagram

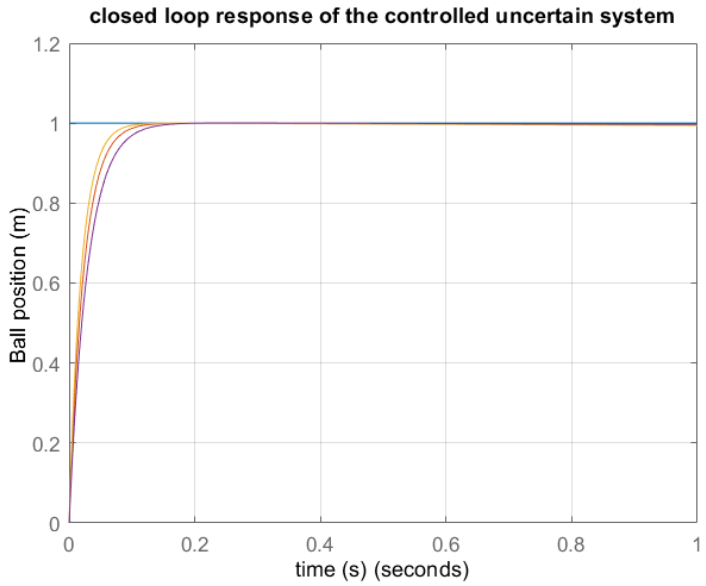

Figure 3.9: Closed loop response of the uncertain system

the following table is summering the characteristics of the two different responses, the first one given by the artificial immune feedback controller and the second one given by the structured h infinity controller, it can be seen that the settling time and the rise time are considerably reduced, moreover the proposed controller could eliminate totally the overshoot, and it could guarantee the best position of the levitated body, the steel ball. Furthermore, the structured PID controller could satisfy the h infinity norm with gamma factor less than 1 and ensure the performance robustness as it is shown in the figure (3.8)

It can be observed from the figure (3.9) that the proposed controller is rugged in front of the parametric uncertainties.

Table 3.1: Maglev setting parameters

Maglev System						
controllers	kp	ki	kd	O(%)	ts(s)	tr(s)
PID-Immune	1.8823	0.1780	1.3411	15	12.5	1.466
Structured PID	-4.92	-2.31e-05	-83.1	0	0.0904	0.0512

3.4 Conclusion

in this chapter the magnetic levitation system is controlled with the artificial immune feedback control, where the parameters of PID controller and the non liner block of the artificial immune system are optimized using ITAE criterion, the system's output missed lot of desirable characteristics especially for the chosen system, a slow response with an important overshoot. For that, the artificial immune block is identified and added to a structured PID controller as a weighting function, from the results presented above, it can be seen the amelioration of the system's response, besides the nominal performance guaranteed.

Chapter 4

LQR Based Fuzzy Immune PID Controller Design

4.1 Introduction

The following chapter discusses an innovative bio-inspired controller, where the parameters of the PID immune feedback controller are calculated according to the optimal linear quadratic algorithm, the proposed controller is applied to adjust the temperature parameter of a greenhouse system throw ethernet network. This controller will be compared with the classical PID and the fuzzy immune PID over Matlab/Simulink environment and using TrueTime toolbox.

4.2 LQR-based PID Controller

The linear quadratic regulator (LQR) as it is fully described in the first chapter, is characterized by its robust and optimal control law which could fulfill the desired performances in many control fields, it is based on the selection of feedback gain K that minimizes the cost function J [34].

In order to simplify the linear quadratic regulator structure and its implementation, its control law is converted to the three inherent parameters of the PID controller [56], these parameters are calculated as in the equations below:

$$\begin{bmatrix} K_p & K_d \end{bmatrix} = \overline{K_p}(\overline{C}^{-1}) \tag{4.1}$$

$$K_i = (I_m + K_d CB)\overline{K_i} \tag{4.2}$$

Where:

$$\overline{C} = \begin{pmatrix} C \\ CA - CB\overline{K_p} \end{pmatrix} \tag{4.3}$$

$$\overline{K_p} = (I_m + K_d CB)^{-1}(K_p C + K_d CA) \tag{4.4}$$

$$\overline{K_i} = (I_m + K_d CB)^{-1} K_i \tag{4.5}$$

Where (A, B, C) are the system state matrix.

4.3 The proposed LQR-based fuzzy-immune PID

An innovative structure of the fuzzy-immune PID controller is proposed in this work taking the advantages of the fuzzy-immune PID combined with those of LQR approach resulting in a new LQR-based fuzzy-immune PID structure, where the parameters of the PID are adjusted using the LQR-based PID algorithm explained in the previous section, combined with the artificial immune block.

4.4 Application

4.4.1 System Presentation Due to the numerous advantages of the greenhouses, their realization is becoming unavoidable in all over the world to extend the production season of plants and different crops which must grow in a favorable climate and in a good environmental conditions. Therefore, the chosen controller must overcome the external influences to guarantee the high quality of plantations with low cost and energy consumption.

The main parameters that should be controlled in a greenhouse are : air temperature, air moisture, CO_2 concentration, light density. For that, the greenhouse should be equipped by several controllers, sensors and actuators [57].

The model given in this work represents an experimental model which describes the dynamics of the greenhouse climate system (especially the temperature inside) given by [58]

, it is presented in a parametric ARX model form as follow[59]:

$$A\left(q^{-1}\right)y\left(t\right) = B\left(q^{-1}\right)u\left(t-nk\right) + e\left(t\right) \tag{4.6}$$

where:

$A\left(q^{-1}\right) = 1 + a_1q^{-1} + a_2q^{-2} + ...a_{na}q^{-na}$.

$B\left(q^{-1}\right) = b_0 + b_1q^{-1} + b_2q^{-2} + ...a_{nb}q^{-nb}$.

$A\left(q^{-1}\right)$ and $B\left(q^{-1}\right)$ represent the nominal dynamic of the system.

$u\left(t-nk\right)$ is the input signal.

$y\left(t\right)$ is the output signal.

nk is the number of input samples occurred before the input affects the output.

$e\left(t\right)$ is the disturbance signal.

q^{-1} is the delay operator.

The vector of the estimated parameters is the following:

$\theta = \begin{bmatrix} a_1 & a_2 & ... & a_{na} & b_1 & b_2 & ... & b_{nb} \end{bmatrix}$

Using experimental data and MATLAB identification toolbox, the authors in [58] obtained the following temperature control sampled transfer function :

$$H\left(q^{-1}\right) = \frac{r_1q^{-1} + r_2q^{-2}}{1 - l_1q^{-1} + l_2q^{-2}} = \frac{0.2215q^{-1} + 0.2215q^{-2}}{1 - 1.384q^{-1} + 0.3989q^{-2}} \tag{4.7}$$

In the frequency (Laplace) domain it can be written as follow:

$$H\left(s\right) = \frac{b_0s + b_1}{a_0s^2 + a_1s + a_2} = \frac{-0.5149s + 67.9}{s^2 + 9.205s + 2.192} \tag{4.8}$$

4.5 Greenhouse Networked Control Loop

4.5.1 Networked Control System One of the significant advancements in digital computation and communication technologies is the development of the Networked Control Systems (NCSs). These systems facilitate the seamless transmission of information among controllers, sensors, and actuators, thanks to their integrated network interfaces [60], besides the incredibly benefits added to the conventional control systems such as higher flexibility, ease of installation and maintenance, and the low cost. In contrast, the deal with data packet-based communication has unavoidable consequences including

packet losses, the time delays possibly occurred and other network bandwidth constraints that can all influence the system's performance and even bring it to the instability[61].

4.5.2 Networked Control of the Greenhouse Temperature

Due to the problems occurred during the transmission of the information using networks such as packets dropouts, time delay..., it is mandatory to use a robust control method to overcome them. The communication channel chosen for controlling the temperature parameter in the greenhouse is Ethernet, as it represents the most practical network solution in controlling distant systems owing to its easy configuration, low cost and its high communication rates [62], three nodes are used (the controller, the system and the network) the configuration of the Ethernet network is given by the table below: In the figure below (Figure

Table 4.1: Ethernet network parameters

Parameters Values		
Number of nodes	Data rate	Minimum frame size
3	80000 bits/s	80

4.1) the Simulink model of the networked temperature control system, the Ethernet network is simulated using TrueTime toolbox the most used network simulator thanks to its rich network components in its library [63]. In the following sections, three controllers

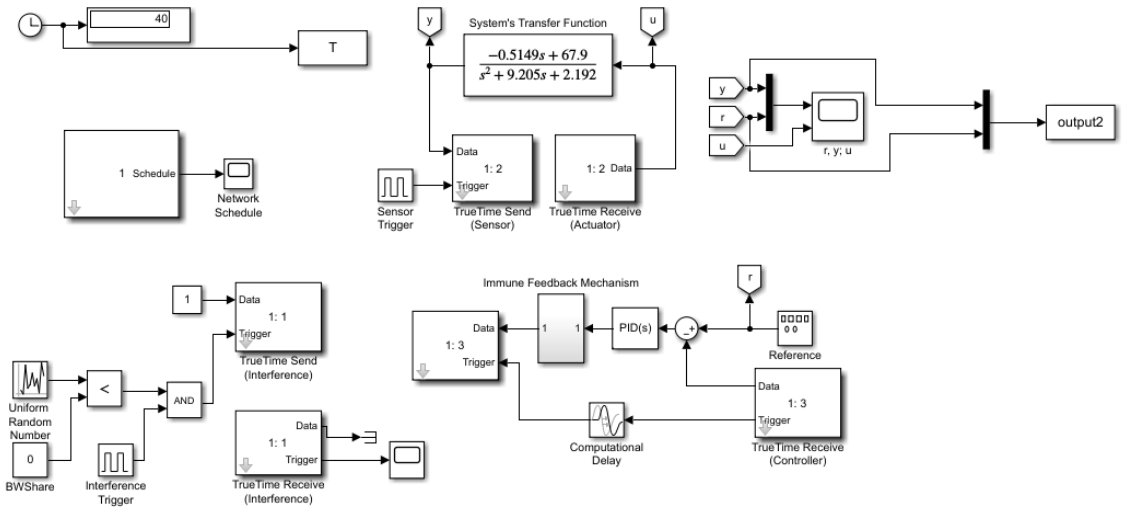

Figure 4.1: Matlab/Simulink model of the LQR-based fuzzy-immune PID structure

are adopted separately to control the networked greenhouse temperature system model: the classical PID, the fuzzy immune PID and the proposed LQR based fuzzy immune

controllers. Note that the artificial immune block is integrated only in the second and the third application.

4.5.2.1 Greenhouse temperature control using an optimally tuned classical PID

In the first application, our system is controlled by the classical PID structure where the parameters K_p , K_i and K_d are optimally adjusted using the Integral Time Absolute Error (ITAE) criterion to minimize the tracking error in a minimum time. This optimization problem can be solved by many software, in our case we used Matlab, the optimal PID parameters following : $K_p = 0.3893$, $K_i = -0.0022$, $K_d = 0.0007$.

4.5.3 Immune feedback law nonlinear function f fuzzy modeling

The non-linear function f plays an important role in the feedback immune PID controller structure, it shapes the system response speed according to the error tracked, as it has a non linear dynamic, in the two following applications,it will be approximated by a fuzzy model where the inputs represent, the controller output $u(k)$ and its variation $\Delta u(k)$.

The fuzzy sets are P (Positive) and N (Negative). The output variable function $f(u(k), \Delta u(k))$, after fuzzification takes the sets P (Positive), N (Negative) and Z (Zero).

The fuzzy rules are obtained according to the immune mechanism principle as follows [64]

1. If u is P and Δu is P then f is N
2. If u is P and Δu is N then f is Z
3. If u is N and Δu is P then f is Z
4. If u is N and Δu is N then f is P

The fuzzification is made according to Mamdani approach and the deffuzification is made by the centroid method.

the artificial immune mechanism gain is adjusted at the value 10 (the best value in terms of time performances and robustness after several tests) .

4.5.3.1 Greenhouse temperature control using the fuzzy-immune PID structure In the second application, the temperature parameter of the greenhouse is adjusted using the fuzzy-immune PID controller with $\eta = 10$
(the best value for high performance and robustness after several tests), here the PID parameters are also optimally tuned by resolving the ITAE criterion using Matlab software, we obtained the following values :
$K_p = 0.4371$, $K_i = 0.3107$, $K_d = 0.0779$.

4.5.3.2 Greenhouse temperature control using the proposed LQR-based fuzzy-immune PID For the third application, we applied our proposed approach for the greenhouse temperature control, the LQR-based fuzzy-immune PID controller uses the same structure as the fuzzy-immune one, however, the parameters of the PID are adjusted using the algorithm described in section 4.3.
Controlability: The controlled system is said to be controllable if the initial states can be transformed to the desired states by a control signal in a finite time[65].
Kalman's test: A system with state vector x of dimension n , is controllable if the controllability matrix has collomun rank [65].
The controllability matrix is the following: $Co = \begin{pmatrix} 1 & -9.2050 \\ 0 & 1 \end{pmatrix}$
We have:
$det(Co) = 1 \neq 0$.
$n = 2$ (the system has 2 states) and $rank\,(Co) = 2$ So the system is controllable.

$Q = 0.06 * I$ where I is the identity matrix, $R = 0.7$.
The LQR-based PID parameters are founded also using MATLAB software, they are adjusted as follow:
$K_p = 0.2812$, $K_i = 0.2881$, $K_d = 0.0310$.

4.5.4 Simulation, results and discussion

In order to determine the robustness of the above control methods, different network disturbances are added to each controller by making some variations in the rate of packets loss probability and in bandwidth sharing by changing the bandwidth occupied by the disturbing node, for many cases as follow:

1. Case (1): Packets loss probability=0, BW share=0

2. Case (2): Packets loss probability=0, BW share=0.3

3. Case (3): Packets loss probability=0, BW share=0.5

4. Case (4): Packets loss probability=0.3, BW share=0

5. Case (5): Packets loss probability=0.4, BW share=0

6. Case (6): Packets loss probability=0.3, BW share=0.3

7. Case (7): Packets loss probability=0.5, BW share=0.5

Figures 4.2 to 4.8 represent a square signal tracking of the greenhouse temperature networked control model with the three approaches.

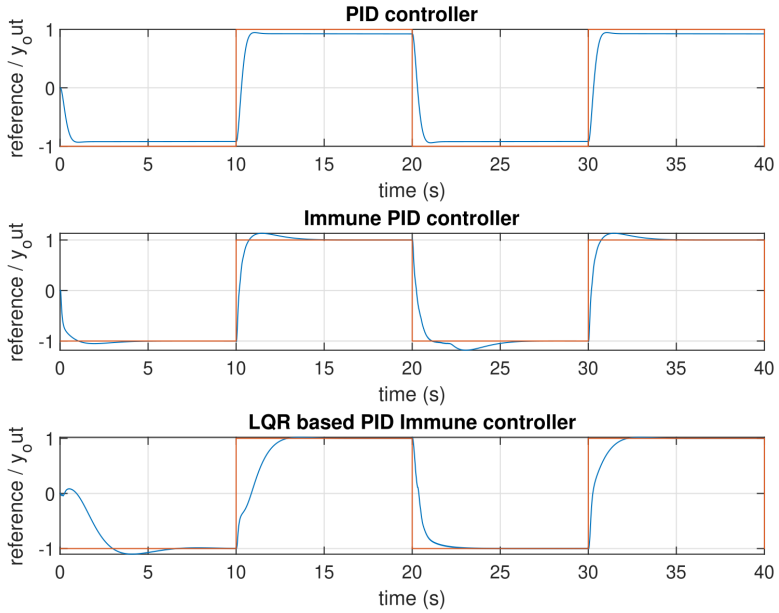

Figure 4.2: Square signal tracking Case (1)

It can be noticed that:

1. The PID controller keeps the system's robustness with a static error maintained in all scenarios.

2. The fuzzy immune PID controller maintain a good performances except with high packets loss percentage, the overshoot increases with 100 %.

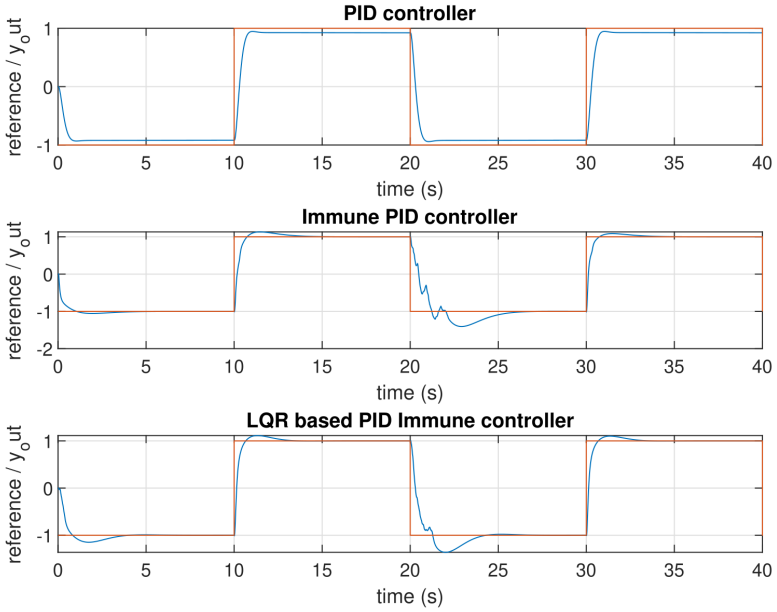

Figure 4.3: Square signal tracking Case (2)

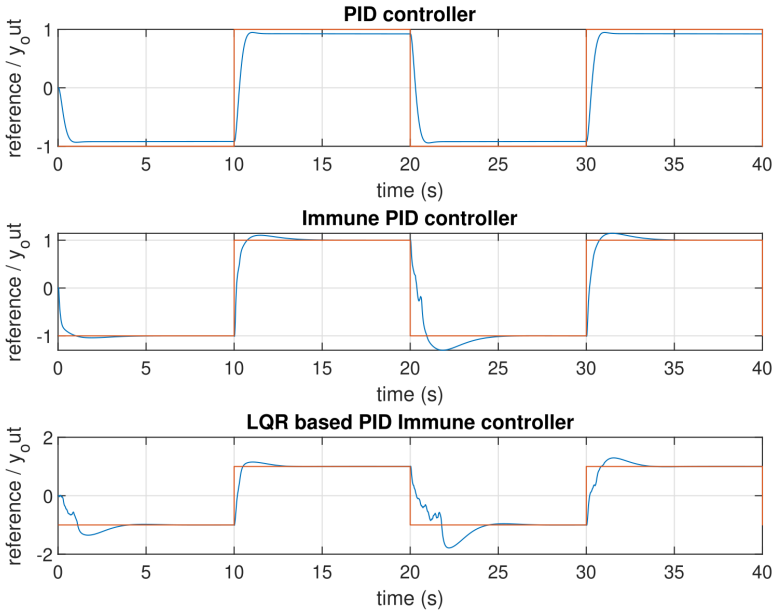

Figure 4.4: Square signal tracking Case (3)

3. The proposed LQR-based fuzzy immune PID controller could ameliorate considerably the system's output comparing with the results found using the the fuzzy immune PID controller, despite all the network disturbances.

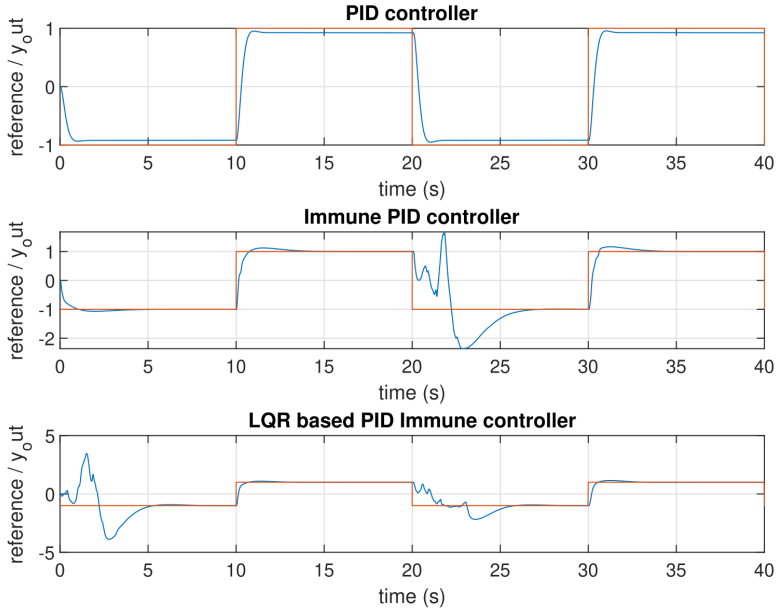

Figure 4.5: Square signal tracking Case (4)

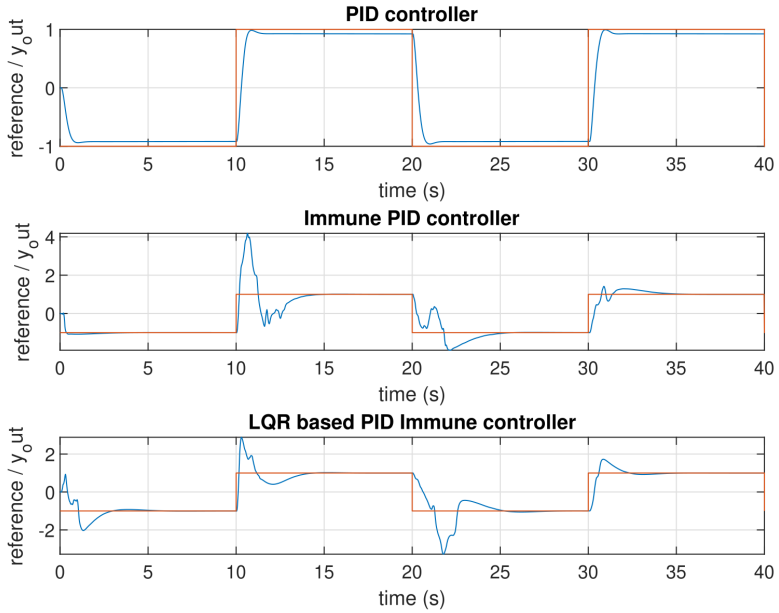

Figure 4.6: Square signal tracking in Case (5)

4.5.4.1 Robustness margins After evaluating the time performances from the simulation results, in this section, the robustness margins are calculated in order to analyze the system's robustness using the three control methods (see Table 4.2).

We can pick up the following notes :

- The optimal classical PID controller could attend the highest phase margin and a

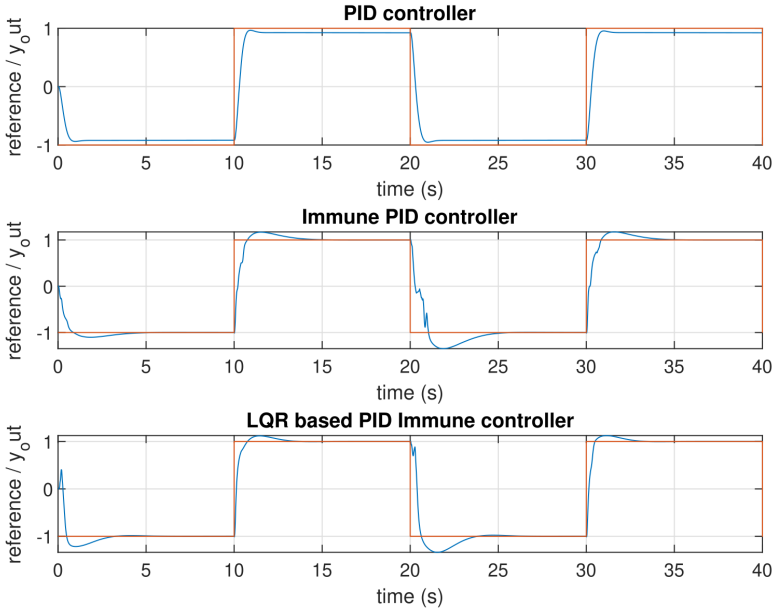

Figure 4.7: Square signal tracking Case (6)

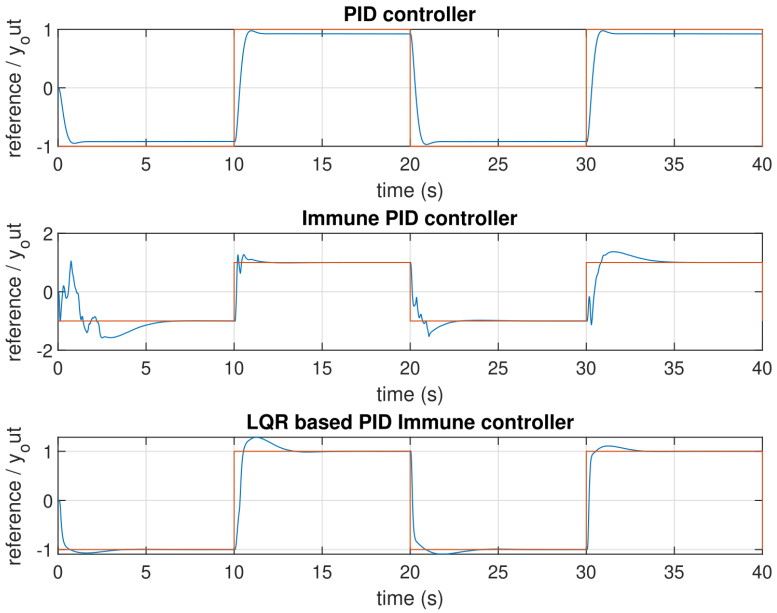

Figure 4.8: Square signal tracking Case (7)

Table 4.2: Robustness margins

controllers/margins	$M_g\,(dB)$	φ_g (deg)	Δ_d
Optimal PID	52.0685	76.7782	0.4775
Fuzzy-immune PID	81.8488	59.1217	2390.2
LQR-based fuzzy-immune PID	81.8502	49.6752	1377.8

good gain margin, however,the delay margin is too short, as it corresponds to the delay amount added to the open loop system before it becomes unstable.

- The fuzzy-immune PID and the LQR-based fuzzy-immune PID controllers have a nearby robustness margins.

From all these results, it is obvious that the LQR-based fuzzy-immune PID controller is the most robust and have the best time performances comparing with the classical PID and the fuzzy immune PID controllers.

4.6 Conclusion

A novel LQR-based fuzzy-immune PID structure is introduced in this chapter, using one of the bio-inspired control methods existed, the fuzzy immune feedback control law, where the parameters of the PID controller are adjusted using the linear quadratic optimization algorithm. the application on a networked greenhouse through Ethernet network shows a good time performances and a high robustness and even in the presence of the network uncertainties comparing with those found using the classical PID and the fuzzy immune PID controller. this confirms the superiority of the proposed approach.

Chapter 5

Robust Neural Network PID Controller Design

5.1 Introduction

Neural networks NN have been successfully applied to control various systems using different learning methods. The following chapter will shed light on a modified structure of an NN-PID controller namely the proportional and filtered integral action controller. In this controller, parameters are calculated using the backpropagation learning algorithm. It demonstrates advantages in controlling systems with high gain and integral perturbations by reducing high-frequency noise amplification and enhancing system robustness. Its application involves three major steps. First, a BPNN-PID controller must be designed to utilize its optimized parameters for calculating the constant time of a low-pass filter. This filter is then integrated into the integral action of a BPNN-P-FI controller structure. This modified structure was primarily designed to eliminate the disadvantage of the derivative term, which amplifies high-frequency noise. Simulation results prove its contribution to enhancing time-domain performance and closed-loop robustness compared to those provided by the optimally tuned PID controller and the BPNN-PID controller..

5.2 Back Propagation Based Neural Network PID Controller Design

5.2.1 Neural Network PID Controller The NN-PID controller combines the principle of a standard neural network with the classical structure of PID controllers resulting in a robust NN-PID controller with more efficient control laws. In fact, These control laws could guarantee a noteworthy results in controlling physical systems with complex dynamics and in the presence of modeling uncertainties and the effect of measurement noise. Therefore, the corresponding NN synthesis model must be designed according to the analysis of the historical data or from the simulated input signals and the desired output, the resulting PID has the characteristic of adapting its parameters with the parametric uncertainties, hence a good reference trajectory tracking with a safe robustness margin [66].

figure (5.1) presents a typical neural network PID.

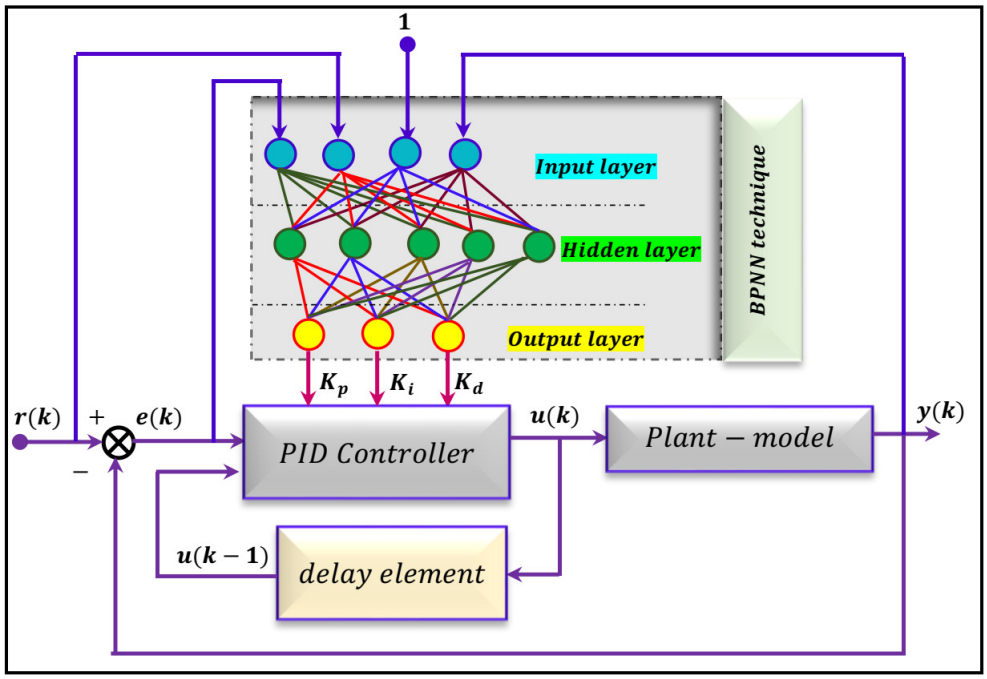

Figure 5.1: BPNN-PID Controlled Closed-loop System

Here, each node in the input layer is expressed by:

$$O_j^{(1)} = x(j), j = 1, 2, 3, 4 \tag{5.1}$$

The outputs of the first and the second layer are calculated with the same manner. For instance, the outputs of the first layer can be computed using the following two formulas, given as below

$$net_i^{(2)}(k) = \sum_{j=1}^{4} w_{ij}^{(2)} O_j^{(1)}, i = 1, 2, 3, 4, 5 \tag{5.2}$$

$$O_i^{(2)}(k) = f\left(net_i^{(2)}(k)\right) \tag{5.3}$$

The activation function to be used in this study presents the tangent hyperbolic function, expressed as below

$$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{5.4}$$

As the standard PID control law is often expressed by the equation (5.5):

$$u(t) = K_p\, e(t) + K_i \int_{\tau=0}^{\tau=t} e(\tau)d\tau + K_d \frac{d}{dt} e(t) \tag{5.5}$$

where K_p, K_i and K_d are the proportional, integral and derivative parameters respectively. Hence, the corresponding discrete-time PID control law is expressed by:

$$u(k) = u(k-1) + \gamma_1 e(k) - \gamma_2 e(k-1) + \gamma_3 e(k-2) \tag{5.6}$$

where γ_1, γ_2 and γ_3 are respectively defined by

$$\begin{cases} \gamma_1 = \dfrac{K_p}{T_s}\left[T_s\left(1 + K_i T_s\right) + K_d\right] \\ \gamma_2 = \dfrac{K_p}{T_s}\left[T_s + 2K_d\right] \\ \gamma_3 = \dfrac{K_p K_d}{T_s} \end{cases} \tag{5.7}$$

Where, T_s is the sampling time, previously chosen by the user. u is the control signal, generated from the synthesized standard PID controller, e is the control error, calculated from the subtraction of the reference input r from the model output y . It is defined, at each sampling time k, by equation (5.8), given as below

$$e(k) = r(k) - y(k) \tag{5.8}$$

This neural network structure will be employed in the next subsection using the back-

propagation learning approach to tune the PID controller-parameters in the presence of model-uncertainties and sensor noise effects.

5.2.2 Back Propagation Based PID Controller

Backpropagation is a paramount supervised learning tool in the field of systems control [67]. Many researchers have developed methods for tuning the PID controller using neural network-based backpropagation learning (BPNN-PID controller) [68, 69]. This machine learning based technique focuses on minimizing the tracking error rate at each sampling time. The weights of the neural network model are updated backward using an optimization approach known as the gradient descent method. At the end of the learning procedure, this algorithm finds the best combination of parameters for the PID controller, resulting in a robust controller with automatic online parameter tuning [66, 70]. The standard BPNN-PID controller design can be illustrated as shown in the figure below:

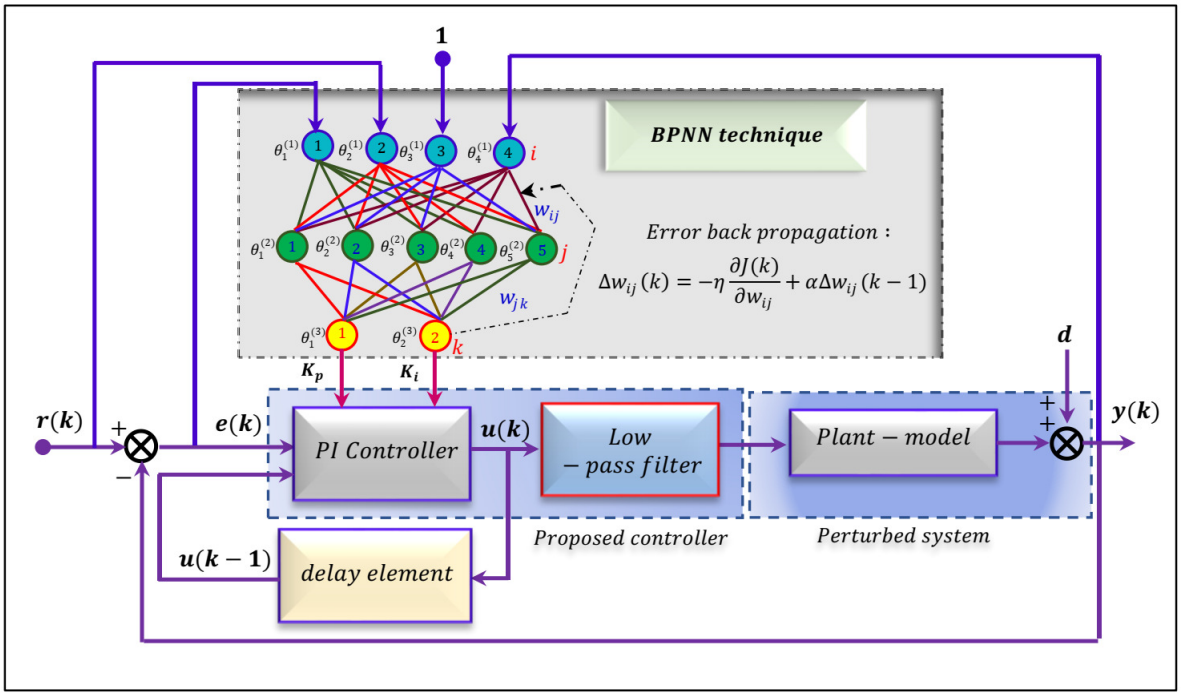

Figure 5.2: Closed-loop system based on the proposed BPNN-P-FI controller for uncertain system

The backpropagation process involves calculating the hidden layer inputs and outputs using equation(5.9) to equation (5.11), given as below

$$net_i^{(3)}(k) = \sum_{i=1}^{5} w_{li}^{(3)} O_i^{(2)} \tag{5.9}$$

$$O_l^{(3)}(k) = g\left(net_l^{(3)}(k)\right) \tag{5.10}$$

where $O_1^{(3)}(k)$, , $O_2^{(3)}(k)$ and $O_3^{(3)}(k)$ represents the three gains of the standard BPNN-PID controller K_p, K_i and K_p respectively. They are given by:

$$O_1^{(3)}(k) = K_p, O_2^{(3)}(k) = K_i, O_3^{(3)}(k) = K_d \tag{5.11}$$

Second, as the three gains of the standard BPNN-PID controller are assumed to be non-negative, the activation function is chosen according to equation(5.12), given by

$$g(x) = \frac{1}{2}(1 + \tanh(x)) = \frac{e^x}{e^x + e^{-x}} \tag{5.12}$$

The error back propagation which requires to be minimized J is formulated, at each sampling time k from the preceding control error denoted by e, equation (5.13), it is given by

$$J(k) = \frac{1}{2}(r(k) - y(k))^2 \tag{5.13}$$

Third, this training algorithm must be repeated until the squared error reaches its minimum value. Since the error minimization is operated according to the gradient descent method, each weight updating is guaranteed by satisfying equation(5.14), expressed as below

$$\Delta w_{li}^{(3)}(k) = -\eta \frac{\partial J(k)}{\partial w_{li}^{(3)}} \tag{5.14}$$

where η is the learning rate. Moreover, as the gradient descent-based minimization process is usually associated with the problem "local minima", this could lead to the failure of ensuring optimal tuning of the controller parameters. Therefore, the possibility of adding the momentum factor α in equation(5.14) is necessary during the error minimization in the back propagation learning. That means the weight change $\Delta w_{li}^{(3)}$ is affected by the previous data, equation (5.15), this expression is becoming as follow:

$$\Delta w_{li}^{(3)}(k) = -\eta \frac{\partial J(k)}{\partial w_{li}^{(3)}} + \alpha \Delta w_{li}^{(3)}(k-1) \tag{5.15}$$

The partial variation of the back propagation error is equation(5.16), expressed by

$$\frac{\partial J(k)}{\partial w_{li}^{(3)}} = \frac{\partial J(k)}{\partial y(k)} \cdot \frac{\partial y(k)}{\partial u(k)} \cdot \frac{\partial u(k)}{\partial O_l^{(3)}(k)} \cdot \frac{\partial O_l^{(3)}(k)}{\partial net_l^{(3)}(k)} \cdot \frac{\partial net_l^{(3)}(k)}{\partial w_{li}^{(3)}} \tag{5.16}$$

The derivative ratio, i.e. $\frac{\partial y(k)}{\partial u(k)}$ can be approximated by the alternative expression, given by equation (5.17) as below

$$\frac{\partial y(k)}{\partial u(k)} = \frac{y(k) - y(k-1)}{u(k) - u(k-1)} \tag{5.17}$$

Accordingly, the control law derivative can be written like follow

$$\frac{\partial u(k)}{\partial O_1^{(3)}(k)} = e(k) - e(k-1) \tag{5.18}$$

$$\frac{\partial u(k)}{\partial O_2^{(3)}(k)} = e(k) \tag{5.19}$$

$$\frac{\partial u(k)}{\partial O_3^{(3)}(k)} = e(k) - 2e(k-1) + e(k-2) \tag{5.20}$$

As a result, from equation (5.15), the update of $\Delta w_{li}^{(3)}(k)$ is performed by equation (5.21), expressed by

$$\Delta w_{li}^{(3)}(k) = \eta \delta_l^{(3)} O_i^{(2)} + \alpha \Delta w_{li}^{(3)}(k-1) \tag{5.21}$$

where the parameter $\delta_l^{(3)}$ is calculated by equation (5.22), expressed by

$$\delta_l^{(3)} = e(k) \frac{\partial y(k)}{\partial u(k)} \cdot \frac{\partial u(k)}{\partial O_l^{(3)}(k)} g'\left(net_l^{(3)}(k)\right) \tag{5.22}$$

Also, from equation(5.15), the update of $\Delta w_{ij}^{(2)}(k)$ is performed by equation (5.23), as follow

$$\Delta w_{ij}^{(2)}(k) = \eta \delta_l^{(2)} O_i^{(1)} + \alpha \Delta w_{li}^{(2)}(k-1) \tag{5.23}$$

where the updating of the hidden layer weights $\delta_i^{(2)}$ is computed by equation (5.24), given by

$$\delta_i^{(2)} = f'\left(net_i^{(2)}(k)\right) \sum_{i=1}^{3} \delta_l^{(3)} w_{li}^{(3)}(k) \tag{5.24}$$

Here, the derivative of the two activation functions $f'(x)$ and $g'(x)$ are respectively defined by equation (5.25), given as below

$$\begin{aligned} f^{'}(x) &= 1 - f(x)^2 \\ g^{'}(x) &= g(x)(1 - g(x)) \end{aligned} \tag{5.25}$$

5.3 The Proposed BPNN-P-FI Controller Synthesis

Binding a low-pass filter directly downstream to the integral part of the PID controller being synthesized provides a significant improvement in time performance, as well as robustness, as proven in numerous systems. This improvement is achieved by enhancing gain and phase margins, thus reducing overshoot. The derivative action amplifies high-frequency signals, which is not recommended for systems vulnerable to external noise, particularly in industrial processes controlled by mathematical models with high static gain and/or poles at the origin.

Indeed, the Proportional-Filtered Integral (P-FI) controller operates similarly to the standard PID controller but without the drawbacks associated with using only the integral (I) or derivative (D) terms. These drawbacks often lead to issues such as windup and amplification of sensor noise in closed-loop system responses [71, 72]. Therefore, a low-pass filter, expressed by equation (5.26), is adopted in the proposed controller synthesis.

$$F(s) = \frac{1}{T_f s + 1} \tag{5.26}$$

The resulting P-FI controller will have the following transfer function

$$C_{P-FI}(s) = K_p + \frac{K_i}{s}\left(\frac{1}{T_f s + 1}\right) \tag{5.27}$$

The proposed control structure unifies the standard PID controller and the low-pass filter. The main advantage of employing this structure lies, on the one hand, the satisfaction of $+20dB$ down-slope in the open-loop modulus frequency response of $|G_{BO}|_{dB} = |G_n(\omega)\, C_{P-FI}(\omega)|_{dB}$, plotted in semi-logarithmic scales in the Bode diagram. Where, according to the principle of loop shaping, more attenuation of the sensor noise effect was hold, which often occurs at the high frequencies. On the other hand, this modified structure guarantees a significant robust stability (RS) margin despite the unstructured additive model uncertainties that were added to the nominal model output. Furthermore, it allows

to improve the nominal performance (NP) at low frequencies by filtering the integral part where its effect becomes more dominant in this frequency range.

5.3.1 Low pass Filter Design

The idea of computing the low-pass filter constant T_f lies in one of the main control methods: Internal Model Control (IMC) or Two-stage Internal Model Control (TS-IMC), which is the latest advanced version of the former. TS-IMC successfully overcomes the main drawback of the IMC method, namely the sluggish response when subjected to a load disturbance input. This sluggishness often arises due to the presence of specific poles leading to inherent slow behavior in the actual dynamics of the process being controlled. Consequently, the adoption of a two-degree-of-freedom controller structure, i.e., $2-DOF$, is required in the synthesis of the desired controller [71]. Building upon these benefits, the controller synthesis used in this paper is based on the principles of the TS-IMC method [72] to compute the desired time constant as follows:

$$T_f = \sqrt{\frac{\alpha_2}{\alpha_0}} \tag{5.28}$$

where the two coefficients α_0 and α_2 are given from the *Maclaurin* series polynomial expansion of the PID controller law expressed by:

$$C(s) = \frac{1}{s}\left(\alpha_0 + \alpha_1 s + \alpha_2 s^2 + ... + \alpha_n s^n\right) \tag{5.29}$$

According to equation (5.5), we can deduce the following expression to express the transfer function of the standard PID controller:

$$C_{PID}(s) = \frac{1}{s}\left(K_i + K_p s + K_d s^2\right) \tag{5.30}$$

From equation (5.29), equation (5.30) and equation (5.31), the low-pass filter time constant T_f is computed by

$$T_f = \sqrt{\frac{K_d}{K_i}} \tag{5.31}$$

5.3.2 PBNN-P-FI Controller Parameters Learning Procedure

In the synthesis of the proposed P-FI BPNN controller, the same NN model, served to design the standard BPNN-PID controller, is adopted. However, the only difference lies in con-

sidering only the two inputs, i.e., K_p and K_i. Then, the two input and output layers, the hidden layer and the training algorithm can be updated using equation (5.9) to equation (5.24) taking care the switching-off the hidden-layer output $O_3^{(3)}(k) = K_d$ during the fitness minimization process. It is therefore appropriate to adopt the discrete-time BPNN P-FI control law instead of one based on the BPNN-PID. Where the new parameters of the discrete time P-FI control law are expressed as follow:

The P-FI control law given by the equation (5.27) can be rewritten as in the equation (5.32):

$$C_{P-FI}(s) = \left(\frac{K_p T_f s^2 + K_p s + K_i}{s + T_f s^2} \right) E(s) \tag{5.32}$$

where $E(s)$ is the Laplace transform of the control error, given previously in equation (5.8). Also, the discrete-time of equation (5.32) yields also equation (5.33), expressed as below:

$$\begin{aligned} \frac{u_{p-fi}(k) - u_{p-fi}(k-1)}{T_s} + \\ T_f \frac{u_{p-fi}(k) - 2u_{p-fi}(k-1) + u_{p-fi}(k-2)}{{T_s}^2} \\ = K_p T_f \frac{e(k) - 2e(k-1) + e(k-2)}{{T_s}^2} \\ + K_p \frac{e(k) - e(k-1)}{T_s} + K_i e(k) \end{aligned} \tag{5.33}$$

Then, the discrete-time of the control law u_{p-fi}, provided by the proposed P-FI controller, is given by the following equation:

$$\begin{aligned} u_{p-fi}(k) = a_1 u_{p-fi}(k-1) - a_2 u_{p-fi}(k-2) + b_1 e(k) - \\ b_2 e(k-1) + b_3 e(k-2) \end{aligned} \tag{5.34}$$

Where the coefficients a_1, a_2, b_1, b_2 and b_3 are respectively expressed by:

$$\begin{cases} a_1 = \dfrac{1+2\beta}{1+\beta} \\ a_2 = \dfrac{\beta}{1+\beta} \\ b_1 = K_p\left(1+\dfrac{K_i T_s}{1+\beta}\right) \\ b_2 = K_p\left(\dfrac{1+2\beta}{1+\beta}\right) \\ b_3 = K_p\left(\dfrac{\beta}{1+\beta}\right) \end{cases} \tag{5.35}$$

where the parameter $\beta = \frac{T_f}{T_s}$ depends heavily on selecting the sampling time T_s. The proposed method using the BPNN-P-FI controller can be summarized in the proposed flowchart, depicted in figure (5.3).

5.4 Application, results and discussion

In order to prove the effectiveness of the proposed robust BPNN-P-FI controller, a transfer function with a very high gain and a pole at the origin presented in [73]) is chosen as the typical model for this application, it is given by :

$$G(s) = \frac{523500}{s^3 + 87.35s^2 + 10470s} \tag{5.36}$$

The presence of a pole at the origin of the s-plane in $G(s)$ implies an infinite static gain, resulting in marginal stability for the closed-loop system.
First, the standard PID controller is tuned using *PidTuner*, the parameters setting in the frequency-domain are the following:

- Desired bandwidth $\omega_B = 100\ rad/s$.
- Desired phase margin $\varphi = 63°$.

After several trials, we obtained the best gains values of the standard PID controller as :

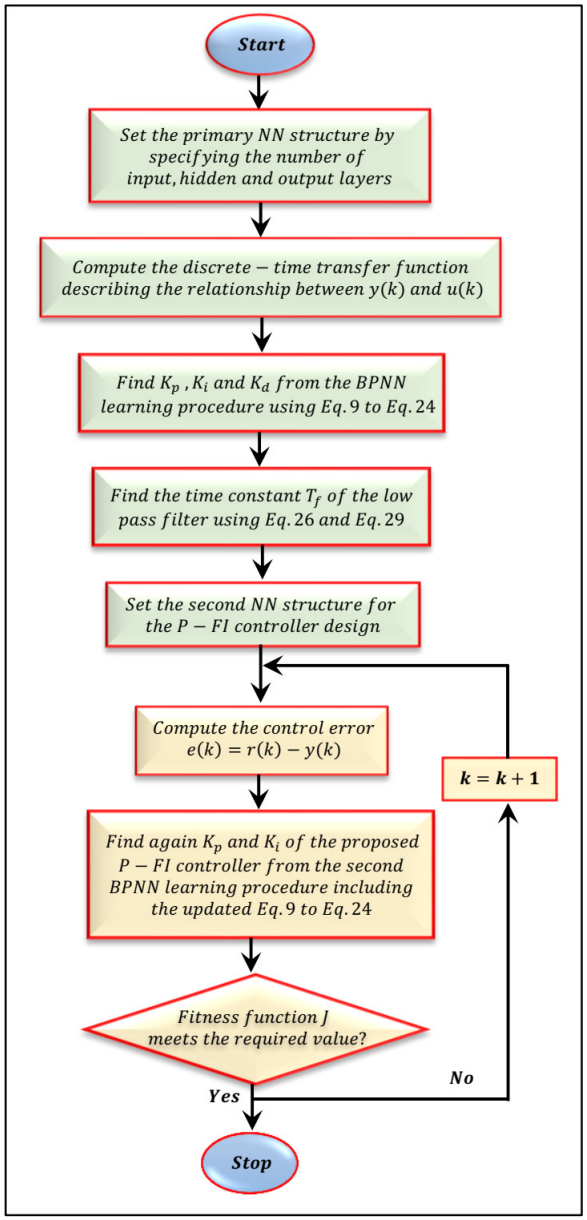

Figure 5.3: The flowchart describing the BPNN-P-FI synthesis procedure

$K_{p_0} = 0.83756234$, $K_{i_0} = 13.435$, $K_{d_0} = 0.01861$.

According to the results found using the PID controller, it is glaring that the gain associated to integral action is very important, the reason which will inevitably impacts the time response of the closed-loop system and especially during the transient phase, resulting an important rise time and a significant overshoot. To overcome this issue, it will be necessary to consider different bandwidth ω_B. As a matter of fact, choosing an alternative bandwidth will spur a considerable degradation of non-minimum phase (NP) properties owing to the presence of a high steady-state. To strike a balance between these conflicting aims of non-minimum phase and robust stability (NP/RS), we have selected ω_B equal to: $\omega_B = 100$rad/s as it offers the best trade-off between these two objectives.
Now, for the two remaining controllers based on BPNN, before starting the algorithm steps, it is necessary first to initialize the weights, this action will be achieved randomly using Matlab software, then, the discretization of equation (5.36) is performed using the sampling time: $Ts = 0.02sec$ resulting the following discrete-time transfer function :

$$G_d(z) = \frac{0.3969z^2 + 0.8507z + 0.1573}{z^3 - 0.7693z^2 - 0.05636z - 0.1743} \tag{5.37}$$

where the associated digital process output $y(k)$ is given by

$$\begin{aligned} y(k) = 0.7693y(k-1) + 0.05636y(k-2) + \\ 0.1743y(k-3) + 0.3969u(k-1) + 0.8507u(k-2) + \\ 0.1573u(k-3) \end{aligned} \tag{5.38}$$

5.4.1 BPNN-PID and BPNN-P-FI Synthesis

5.4.1.1 BPNN-PID controller synthesis The synthesis of the standard BPNN-PID controller is conducted as described above, utilizing the discrete output provided by Equation (5.38). The neural network employed here is trained using the backpropagation learning method with the gradient descent algorithm to minimize the backpropagation error. This iterative process is repeated multiple times until the error is significantly reduced, and then the final controller parameters are determined. This determination involves starting from several random weight initializations within a time simulation range

of $[0,\ t_{\max} = 10]$ sec.

The initial values assigned to the integral and derivative gains are set close to zero. However, the proportional gain is initially set to 0.5 to prevent null commands, as mentioned in Equation (5.5), which could hinder the minimization of the backpropagation error. The evolution of the three setting parameters is depicted in Figure (5.4), starting from the preset initial values until they converge to their final stationary values. The resulting stationary values of each gain are as follows:

$$K_{p_1} = 0.6234,\ K_{i_1} = 0.2624\ ,\ K_{d_1} = 0.2525.$$

Utilizing Equations (5.6) and (5.7), which describe the relationship between the optimal control law and the control error in discrete time, we can extract the following parameters during the entire time simulation:

$$\gamma_1 = 8.4815,\ \gamma_2 = 16.333,\ \gamma_3 = 7.8548.$$

5.4.1.2 The BPNN PI-F controller synthesis As mentioned before, the time constant of the low pass filter is calculated using equation (5.32) according to the results found using the NNPID controller. Hence, the resulting transfer function of the desired low-pass filter is given by

$$F(s) = \frac{1}{0.98s + 1} \tag{5.39}$$

The following step is to bind this filter $F(s)$ in cascade with the integral part of the standard BPNN-PID structure by omitting completely the derivative part. Then, the backpropagation learning will be achieved using this new control-loop generated using the second NN architecture, in order to adjust the proportional and integral gains of the proposed BPNN-P-FI controller following the same procedure as that of the standard BPNN-PID controller, leading to the following results: $K_{p_2} = 0.5035\ ,\ K_{i_2} = 0.007$.
Finally, using these values to set the parameters of the desired optimal control law with the control error (equation (5.34) and equation (5.35) gives:

$$a_1 = 1.9918,\ a_2 = 0.99181,\ b_1 = 0.50379,\ b_2 = 1.0029,\ b_3 = 0.49938.$$

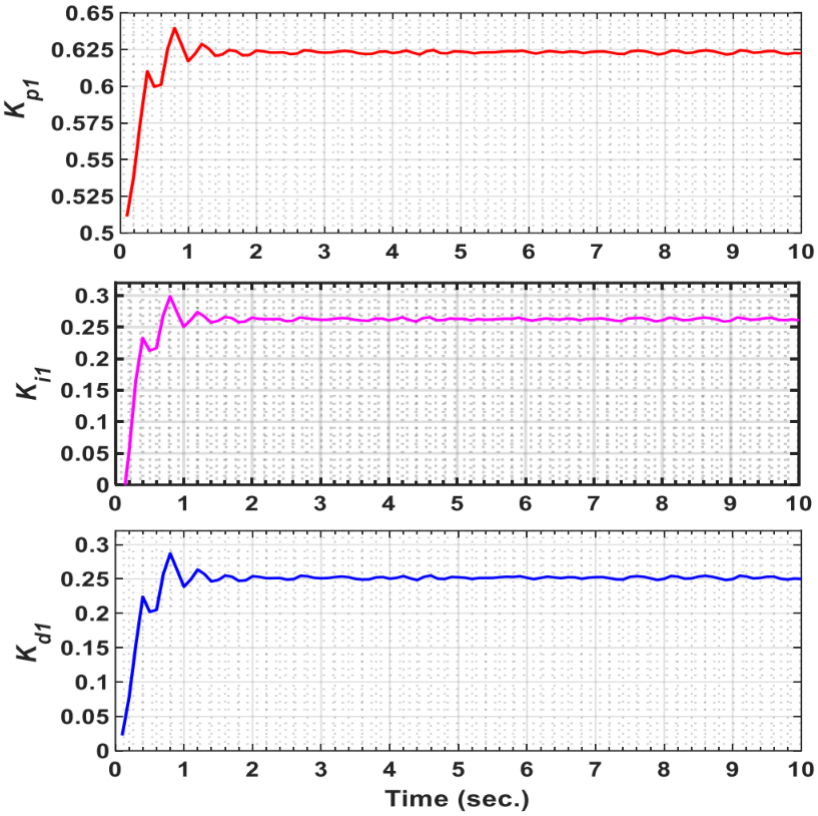

Figure 5.4: Evolution of three setting parameters of the standard BPNN-PID controller

5.4.2 Time-domain Responses Comparison In this section, we intend to compare the time-domain performances provided by the three different controllers presented in this work. First of all, the process outputs is computed at each sampling time using the discrete transfer function given by the equation equation (5.38), where the initialization sequence of the control and process-output signals are all set by zero. Also, it is important to note that the closed-loop system has a fast time response using the three controllers, where, the time simulation was reduced from $t_{\max} = 10sec.$ to $t_{\max} = 0.5sec.$ for more details of the transient-state performances. Now, it is worth to point out that the nominal performance properties (NP) generally includes key characteristics such as a small settling time, low overshoot, and minimal time required to mitigate tracking errors. Referring to figure (5.5), it is glaring that the proposed BPNN-P-FI controller outperforms the other two controllers in terms of NP. The system's output of the proposed controller could guarantee a faster settling time of $t_{s_{5\%}} = 0.05$ seconds, a lower overshoot of $D\% = 5.41\%$ at $t_{D\%} = 0.04$ seconds, and the shortest time required to minimize as much

as possible the steady-state tracking errors of $t_{\zeta_{5\%}} = 0.09$ seconds. Comparing to the two other controllers, the standard BPNN-PID controller occupies the second-best position, where the standard PID controller demonstrates the least favorable performance.

Table 5.11 summaries the performance of the three controllers throw the parameters

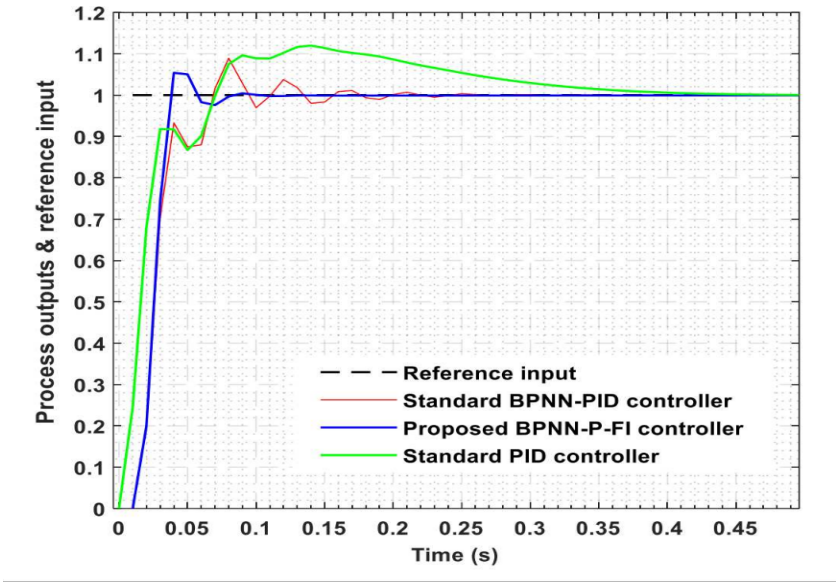

Figure 5.5: Reference tracking dynamics ensured in time domain by the three controllers

characterized reference tracking dynamics, where the superior parameter values are highlighted in bold. Moreover, Table 5.2 provides an overview of the enhancement percentage of each parameter, which is computed by comparing the time response of the proposed BPNN-P-FI controller with that of the other two controllers. This ratio is calculated according to the following rule:

$$\left| \frac{performance\ of\ the\ designed\ controller - performance\ of\ the\ BPNN-P-FI}{performance\ of\ the\ designed\ controller} \right| \%$$

Considering the NP/RS trade-off, the BPNN-P-FI controller proves to be the most favorable choice for controlling a physical process described by a plant model with an a high order gain and a pole located at the origin of the s-plane. It offers a superior balance between achieving excellent nominal performance properties and ensuring robust stability. The BPNN-P-FI controller demonstrates significant improvements in terms of settling time, overshoot, and tracking error compared to the standard PID and standard BPNN-PID controllers.

where t_r, t_s and $D_{\%}$ are the rise time, the settling time and the overshoot respectively.

Table 5.1: Step response characteristics of each controller

	t_r (sec.)	t_s (sec.)	$D_{\%}$
BPNN-P-FI	**0.05**	**0.09**	**5.41**
BPNN-PID	0.08	0.22	9.64
PID	0.26	0.41	14.36

Table 5.2: : Improvement percentage (in %) of each time response parameter that is ensured when the proposed BPNN-P-FI controller is applied

	t_r (sec.)	t_s (sec.)	$D_{\%}$
BPNN-P-FI compared to BPNN-PID	37.5%	59.09%	43.87%
BPNN-P-FI compared to PID	**80.76%**	**78.04%**	**62.32%**

5.5 Robustness analysis

This section aims to evaluate the robustness of the closed-loop system controlled by the three synthesized controllers in the presence of unknown modeling uncertainties. These modeling uncertainties are the relative deviation between the nominal model and all neighboring perturbed models. First, the examination of the sensor noise suppression dynamics, can provide valuable information about robust stability and especially the attenuation of the effect of sensor noise that commonly occurs at high frequencies. besides the closed-loop stability assessment in the presence of unstructured multiplicative uncertainties, as at high frequencies, the closed-loop system should exhibit minimal sensitivity to these uncertainties [74].

5.5.1 Frequency-domain Robustness Comparison

In the following part, figure (5.6) a noise is added to the sensor signal of the controlling loop, comparing the sensor noise suppression dynamics, applied in the frequency-domain by the three previously presented controllers. It represents the frequency response in terms of the maximal singular value plot of the complementary sensitivity function $\sigma_{\max}[S_c(\omega)] = \sigma_{\max}[G_o(\omega)/(1 + G_o(\omega))]$. Therefore, it provides valuable information regarding the robust stability (RS) properties including the attenuation of sensor noise effects, typically at the high frequencies, as well as the closed-loop stability in the presence of the unstructured multiplicative uncertainties. Here, $G_o(s) = G(s)C(s)$ represents the open-loop system, which depends on the synthesized controller.

On the other hand, it is well known that robust stability is ensured when $\sigma_{\max}[S_c(\omega)]$

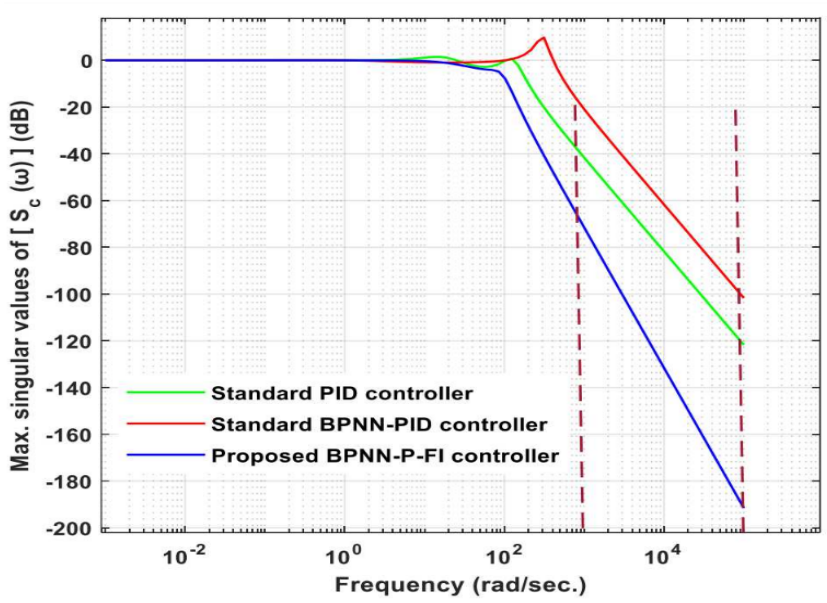

Figure 5.6: Sensor noise suppression dynamics ensured in frequency domain by the three controllers

decreases significantly at high frequencies and approaches unity (0_{dB}) at low frequencies. Regarding figure (5.6),the proposed BPNN-P-FI controller, $\sigma_{\max}[S_c(\omega)]$ remains below $+80_{dB}$ within the frequency range $\omega \in \left[10^3,\ 10^5\right]\ rad/\sec$.. In contrast, the standard PID controller and the standard BPNN-PID controller exhibit $\sigma_{\max}[S_c(\omega)]$ values below $+40_{dB}$ and $+20_{dB}$, respectively, within the same frequency range. Based on the observations cited above, the proposed BPNN-P-FI controller exhibits the best performance in terms of robust stability. It offers over twice time sensor noise diminution compared to the standard PID controller and more than four times compared to the standard BPNN-PID controller. Consequently, the proposed BPNN-P-FI controller achieves the highest level of robustness, followed by the standard PID controller in the second position and the standard BPNN-PID controller in the third position.

An other robustness test is applied by adding noise at the system's output. Therefore, a random values of the order of 1 % are added to the output in order to simulate the noise affect, the following figure (5.7) describes the system's behaviour using the three proposed controllers, PID, BPNN-PID and BPNN-P-FI : After applying a disturbance at the system's output, the BPNN-PID and the BPNN-P-FI controllers still keeping the required performance and the robustness of the system owing to the adaptive control, with a negligible overshoot given by the BPNN-P-FI, where it's glaring that the PID controller presents an important overshoot of 14 % besides the time taken to reach the steady state.

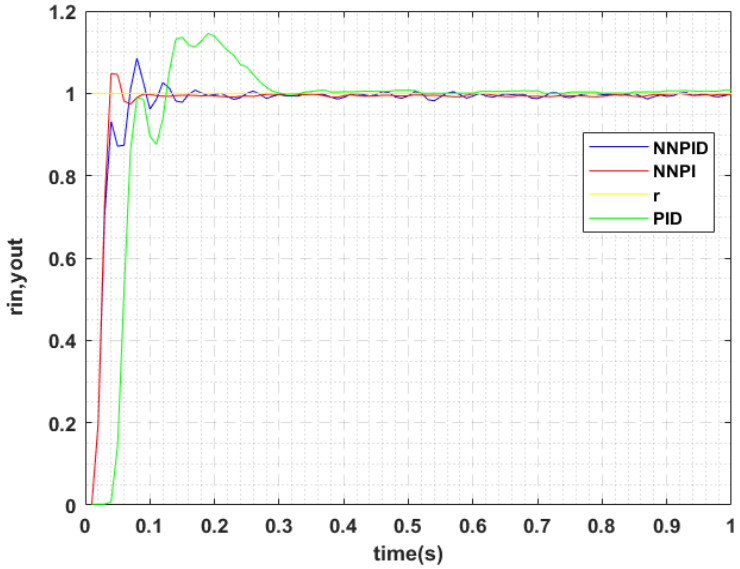

Figure 5.7: System's output after applying the noise at the system's output

Consequently the superiority of the proposed BPNN-P-FI controller.

5.5.2 Perturbation rejection

In the following part, another aspect of the system's robustness test regarding the perturbation rejection. We assume that perturbations are applied to the output of the nominal model to closely match it with the measured process output, the amplitude of this perturbation input is assumed to be equal to 25% of the the setpoint. It is applied exactly at the starting time $t = 0.5$ s. Additionally, the synthesized nominal model includes a high static gain, which has a significant impact on the trade-off between performance and robustness of the closed-loop system besides the location of one pole at the origin. Furthermore, this system is considered uncertain, where its gain value is deviating from the nominal value by a relative gap equal to 30%. This means that, since the nominal gain is equal to 523500, the corresponding uncertain gains have values fluctuating randomly between the two upper and lower limits, given respectively by 680550 and 366450. In this study, 200 uncertain static gains are generated, resulting in a total of 200 perturbed model. Accordingly, the best possible trade-off is realized when a good set point tracking dynamics is maintained with the best of these properties, such as rise time, settling time, overshoot, regardless the effect of the perturbation input. Simultaneously, good rejection dynamic is ensured when the effect of the model uncertainties is rapidly attenuated without impacting the performance of the closed-loop system. Taking all these key requirements into account, the nominal model controlled by the BPNN-P-

FI controller ensured the best performance-robustness trade-off with respect to all types of modeling uncertainties, ensuring a good mitigation of the perturbation effect, with a faster time response compared to that supplied by the two remaining controllers (see figure(5.8)).

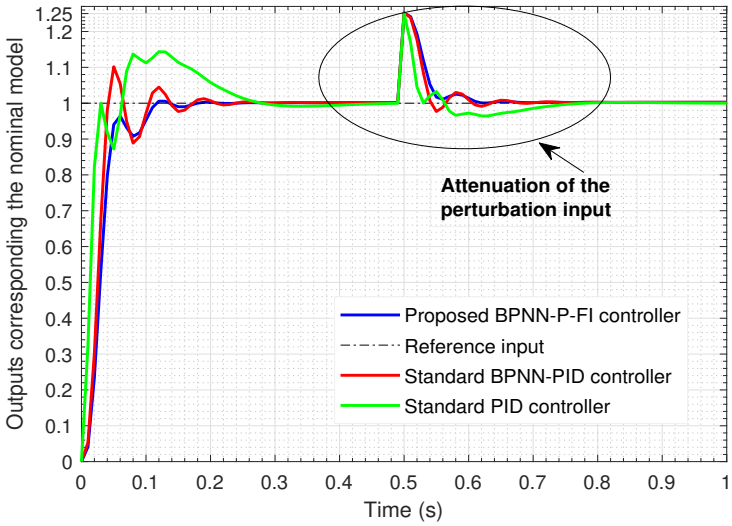

Figure 5.8: Perturbation rejection dynamics for the nominal model controlled by the three controllers

Eventually, in the figure (5.9), the output of the 200 closed loop uncertain model is presented by the dashed lines where each one of them encompasses the same perturbation effect, and using the three preceding controllers, it is glaring that the best trade-off is always achieved by the closed-loop system controlled by the proposed BPNN-P-FI controller.

Also, all resulting outputs are compared by the one provided for the nominal model case of the functioning system, which is mentioned using solid-line.

For further clarification, the transient-state part, as well as the one representing the disturbance rejection part of each previous controller, are zoomed in the two-time ranges comprised between $0 \leq t \leq 0.2$ sec. and $0.49 \leq t \leq 0.7$ sec, respectively. Accordingly, the time response related to the proposed BPNN-P-FI controller attains the desired output very quickly, while the overshoots are much more limited than those corresponding to the two remaining controllers. These positive requirements are reached not only in the case of nominal operating state but also in the case of all uncertain models. Simultaneously, the disturbance rejection is guaranteed in sufficiently fast time even in the presence of these

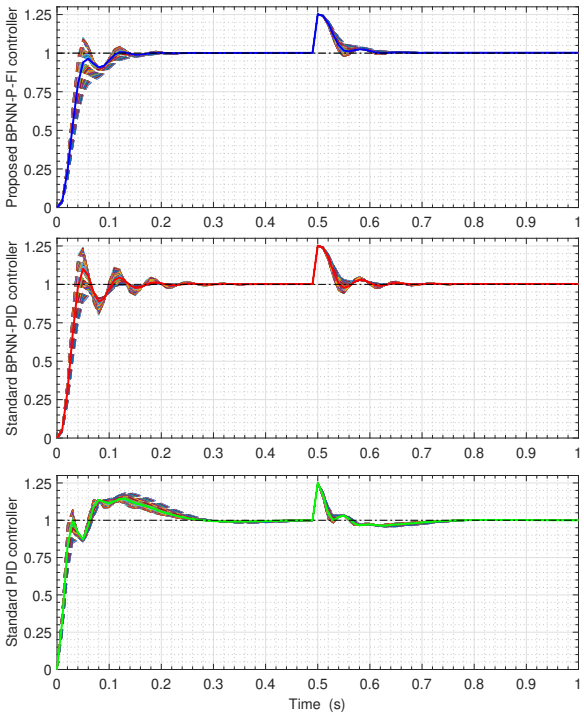

Figure 5.9: Perturbation rejection dynamics for 200 high-gain perturbed model controlled by the three controllers

uncertainties, where the tracking errors are completely eliminated in a time not exceeding t=0.1 sec. (see (figure(5.10)) As a result, the closed-loop system based on the proposed controller becomes much less sensitive to model uncertainties compared to the other two closed-loop systems based on the other two controllers.

5.6 Conclusion

This chapter addresses the robustness enhancement of the BPNN-PID controller by drawing inspiration from the filtered PI controller. The core concept of this approach involves integrating key characteristics of the filtered PI controller, which incorporates a low-pass filter in cascade with its integral part, while omitting the derivative action. This integration leads to significant performance improvements, especially in processes involving high-gain integral systems, as it effectively mitigates noise amplification in high frequencies.

The proposed BPNN-P-FI controller entails a two-step parameter tuning process. Initially, the three gains of the standard BPNN-PID controller are determined via backpropa-

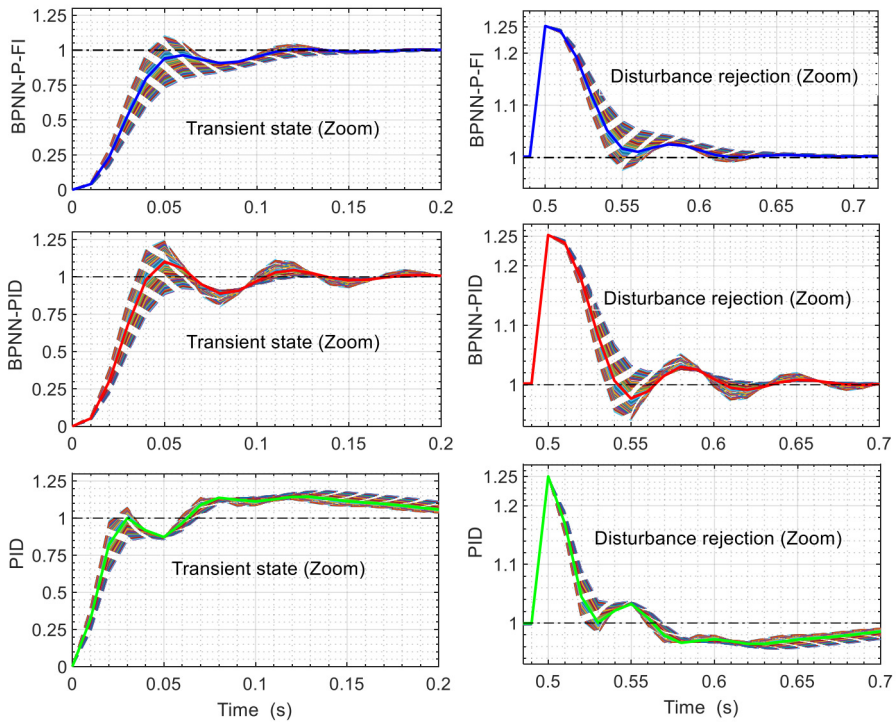

Figure 5.10: Highlighting the transient-state and the disturbance rejection zones for each controller

gation learning using a gradient descent minimization process. Subsequently, these gains are utilized to compute the coefficients of the low-pass filter, aimed at mitigating undesired effects stemming from the derivative action during the controller synthesis phase.

The proportional and integral gains of the robustified controller necessitate readjustment to accommodate the insertion of the low-pass filter in the control loop. The robust stability of the BPNN-P-FI controller is assessed in the frequency domain, while its nominal performance is analyzed in the time domain.

To validate this approach, the standard PID controller is robustly and optimally synthesized using the 'Pid Tuner' command in Matlab software. Simulation results are obtained by individually applying these three controllers to control an industrial process characterized by an infinite static gain and a pole at the origin of the s-plane, representing stability limits.

The findings underscore the superiority of the proposed BPNN-P-FI controller compared to the other two controllers, particularly in achieving a favorable balance between nominal performance and robust stability.

General Conclusion

Bio-inspired algorithms have proven to be efficient strategies across various scientific fields. Recently, much research has focused extensively on their development. In control systems, these algorithms are widely employed, particularly in the synthesis of innovative controllers for complex systems, enhancing the performance of closed-loop systems. However, despite their efficacy, there remains a significant gap in exploring the robustness aspect. Insufficient attention has been devoted to quantifying and enhancing the robustness of such controllers, especially concerning the diverse uncertainties affecting the closed-loop system, such as parametric and non-parametric variations, unknown uncertainties, sensor noise effects, and more. Addressing these robustness challenges is crucial for ensuring the reliability and effectiveness of bio-inspired controllers in real-world applications. In this thesis, we have explored and developed some bio-inspired control strategies, especially focusing on feedback immune control, fuzzy immune control, and neural networks. Our aim was to enhance their robustness, adaptability, and performance across a wide range of applications. The first section introduces the concept of robust control, which comprises advanced control methodologies used to maintain the stability of closed-loop systems against different perturbations, such as parametric and non-parametric uncertainties. The second chapter delves into the presentation of the considered bio-inspired algorithms in this thesis, which are : the artificial immune feedback control and neural networks, which mimic the adaptability and resilience observed in biological systems. This chapter is considered as a foundation for the subsequent chapters, wherein we delve deeper into their applications and potential enhancements.

In Chapter 3, our objective was to enhance the robustness of the Artificial Immune Feedback PID Controller. To achieve this, we designed a combined H∞ controller design method (a well-known and widely used robust control method), with the immune feedback PID controller. As application model, we choose the magnetic levitation system,

where the goal is to adjust the position of the magnetic ball and maintain it suspended without any support. This task presents a significant challenge due to gravitational forces, wind, disturbances, and inherent uncertainties during system identification. Therefore, the robust control is essential to ensure stability and accurate positioning of the levitated object. To address these challenges, we adopted two controllers : Initially, the PID immune feedback controller alone did not meet the required performance criteria, exhibiting a significant overshoot of 13.6 % and settling and rise time of 12.5 seconds and 1.466 seconds respectively. However, when we applied the structured H∞ PID controller after identifying and incorporating the nonlinear block of the immune system, the closed loop time response performances improved significantly. The overshoot was eliminated definitely where the settling and rise time were reduced to 0.0904 seconds and 0.0512 seconds respectively. The structured H∞ PID immune controller was able to guarantee robust performance, as demonstrated through simulation tests conducted in the presence of parametric uncertainties.

In Chapter 4, we used the robust Linear Quadratic Regulator (LQR) control law to enhance the robustness of the feedback Immune PID Controller. Additionally, we modeled the immune block using a Mamdani's fuzzy model, which provides more flexibility to address various uncertainties. As a result, we developed the LQR-based Fuzzy Immune PID controller. The key contribution lies in optimizing the parameters of the PID immune controller using the LQR algorithm. To demonstrate the effectiveness of our controller in the presence of uncertainties, we applied it to greenhouse temperature system controlled through an Ethernet network. Such system is subject of different network disturbances such as packet dropout and bandwidth sharing rates of the disturbing nodes. For comparison purpose, multiple perturbations scenarios were simulated to assess the robustness of the greenhouse temperature control system using three controllers : the classical PID, the PID immune feedback controller, and the LQR-based PID immune controller. The PID controller could present a good robustness with a permanent static gain in all scenarios. The fuzzy immune feedback controller maintain a good performance except with high packets loss probability, where the overshoot increased to 100%. The LQR-based fuzzy immune PID controller exhibited a good performance illustrated in the robustness margins, the highest gain margin of 81.8502 dB comparing to the two values obtained by using the two previous controllers 52.0685 and 81.8488. The simulation results pre-

sented proves the superiority of the proposed controller in terms of time performances and robustness against network uncertainties.

In the final part of our work, we investigated the Backpropagation Neural Network PID controller (BPNN-PID). Our primary objective was to enhance its robustness in controlling a class of complex systems, particularly the high-gain integral system, which are known for their sensitivity to sensor noise in the high-frequency range. As a contribution, we introduced the Filtered Backpropagation Neural Network PI controller (BPNN-P-FI). This controller inherits the advantages of the BPNN-PID while mitigating its disadvantages, particularly its sensitivity to sensor noise arising from the presence of derivative action. This enhancement is achieved by incorporating a low-pass filter, which enhances the robustness of the controlled system against sensor noise by eliminating the need of derivative action.

The effectiveness of these two controllers, in addition to the traditional PID controller, was compared in controlling a linearized industrial system drawn from literature, which was modeled as a high-gain integral system. This system is characterized by an infinite static gain and a pole at the origin. Robustness simulation tests were conducted in the frequency domain, evaluating the suppression of sensor noise, permanent noise at the system's output, and parametric variations. The proposed BPNN-P-FI controller demonstrated the highest level of robustness and time performance in the simulation tests, where we found improvements in step response characteristics compared to the classical PID controller as follows: 80.76% for the rise time, 78.09% for the settling time, and a 62.32% improvement in overshoot attenuation from 14.36 to 5.41. Additionally, the enhancement of these parameters comparing to the BPNN-PID controller was also achieved with improvements 37.5%, 59.09% and 43.87% respectively. According to the other robustness tests starting from the sensor noise suppression, the proposed BPNN-P-FI controller ensured the robust stability where $\sigma_{\max}[S_c(\omega)]$ was mitigated significantly at high frequencies and settled down below $+80_{dB}$ within the frequency range $\omega \in \left[10^3,\ 10^5\right]\ rad/\sec$. In contrast, the standard PID controller and the standard BPNN-PID controller exhibit $\sigma_{\max}[S_c(\omega)]$ values below $+40_{dB}$ and $+20_{dB}$, respectively, within the same frequency range. Moreover, the advantages guaranteed by the proposed controller can be seen through the other robustness tests performed in adding a permanent noise at the system's output where the system could maintain the stability of the system,

and when the perturbation was injected to the uncertain model, the error tracking was completely eliminated in a time not exceeding 0.1 seconds, and it presented much less sensitivity to model uncertainties compared to the other two closed loop systems based on the other two controllers adopted.

Future Work

As part of our future work, we intend to expand our research to include other recent bio-inspired algorithms, particularly those focused on meta-heuristic optimization. Our aim is to explore their application in various classes of systems, such as nonlinear continuous and discrete systems, as well as event-based systems. The primary objective is to investigate their robustness against different uncertainties. Additionally, we plan to enhance their robustness using various techniques. Another important goal is to apply our findings to real-world processes.

References

[1] N Arana-Daniel, C Lopez-Franco, and A.Y Alanis. *Bio-Inspired Algorithms for Engineering*. Butterworth-Heinemann, 2018.

[2] E Kayacan, H Ramon, and W Saeys. Robust trajectory tracking error model-based predective control for unmanned ground vehicles. *IEEE/ASME TransMechatron*, 21(2):806–814, 2016.

[3] L.N de Castro and J Timmis. Artificial immune systems: A new computational intelligence approach. *New York: Springer*, 2002.

[4] E Hart and J Timmis. Application areas of ais: The past, present and the future. *Journal of Applied Soft Computing*, 8(1):191–201, 1998.

[5] S Jaagannathan and F.L Lewis. Multilayer discrete-time neural-net controller with guaranteed performance. *IEEE Transactions on Neural Networks and Learning Systems*, 7(1):107–130, 1996.

[6] Y Jiang, C Yang, J Na, G Li, Y Li, and J Zhong. A brief review of neural networks based learning and control and their applications for robots. *Neural Network for Complex Systems: Theory and Applications*, 2017(1), 2017.

[7] J Kennedy and R Eberhart. Particle swarm optimization(pso). In *Proceeding of IEEE internatinal conference on neural network*, pages 1942–1948. Australia, 1995.

[8] J Halland. Genetic algorithms. *Sientific American*, 267(1):66–73, 1992.

[9] S Skogestad and I Poslethwaite. *Multivariable Feedback Control*. John, Wiley and Sons,Inc., River Street, Hoboken, NJ, United States, 2001.

[10] K Zhou and J.C Doyle. *ESSENTIALS OF ROBUST CONTROL*. 1999.

[11] *https://www.researchgate.net/figure/The-humoral-and-cell-mediated-branches-of-the-immune-response-The-humoral-branch$_f$ig*1$_2$37057095.

[12] R Bouchebbat and S Gherbi. Design and application of fuzzy immune pid adaptive control based on particle swarm optimization in thermal power plants. In *The 6th International Conference on Systems and Control*. IWAC, 2017.

[13] B Wang, Y Sun, J Cao, and G Zhang. Control and simulation of an underwater robot using single neuron pid control based on immune feedback mechanism. In *Proceedings of the 8th World Congress on Intelligent Control and Automation, Jinan, China*, page 2509, 2010.

[14] S.A Damiati. Digital pharmaceutical sciences. *American Association of Pharmaceutical Scientists PharmSciTech*, 21(206), 2020.

[15] M Debasis, A Bhattachaya, T.K Mondal, and D Indranil. Maglev system modeling and lqr controller design in real time simulation. IEEE WISPNET Conference, 2016.

[16] A Nayak. *Controller Design for Magnetic Levitation System*. National Institute of Technology, India, 2015.

[17] K Takahashi and T Yamada. Application of an immune feedback mechanism to control systems. *International Journal Series C Mechanical Systems, Machine Elements and Manufacturing*, 41(2):184–191, 1998.

[18] Michael Green and David JN Limebeer. *Linear Robust Control*. Courier Corporation, 2012.

[19] SP Bhattacharyya. Robust control under parametric uncertainty: An overview and recent results. *Annual Reviews in Control*, 44:45–77, 2017.

[20] Deepti Thapliyal and Deepak Punetha. Control line of sight (los) using robust control. *International Journal of Control and Automation*, 10(5):105–114, 2017.

[21] D.E Kirk. *Optimal Control Theory: An Introduction*. CRC Press, 2004.

[22] R Herzog and J Keller. *An Overview on Robust Control*. Swiss Society for Automatic Control, 2010.

[23] P.H Petkov and T.N Slavov. *Design of Embedded Robust Control Systems Using Matlab/ Simulink*. The Institution of Engineering and Technology Michael Faraday House, United Kingdom, 2018.

[24] W Haw. Optimal robust control systems design and analysis by statespace approches. *PhD Thesis*, 1995.

[25] D.W Gu, P.H Petkov, and M.M Konstantinov. *Robust Control Design with MATLAB*. Springer, 2013.

[26] G Zames. Feedback and optimal sensitivity: Model reference transformations, multiplicative seminorms, and approximate inverses. *IEEE Transactions on automatic control*, 26(2):301–320, 1981.

[27] G Zames. On the input-output stability of nonlinear time varying feedback systems. *IEEE Transactions on automatic control*, 11:228and465, 1966.

[28] J.C Doyle, K Glove, P.P Khargonekar, and B.A Francis. State-space solutions to standards h_2 and h ∞ control problems. *IEEE Transactions on automatic control*, 34:831–847, 1989.

[29] P Gahinet and P Apkarian. A linear matrix inequality approach to h ∞ control. *IEEE Transactions on automatic control*, pages 421–448, 1994.

[30] G Duc and S Font. *Commande H* ∞ *et* μ *Analyse*. Hermes Science publication, 1999.

[31] S Skogestad and I Poslethwaite. *Multivariable Feedback Control-Analysis and design*. 2nd edition, Wiley, 2005.

[32] W.S Levine. *Control System Advanced Methods*. CRC Press; 2nd edition, 2010.

[33] B.C Kuo. *Automatic Control Systems*. Prentice Hall, INC., Englewood cliffs, New Jersey, 1975.

[34] K Ogata. *Modern Control Engineering*. Prentice Hall; 5th edition, 2010.

[35] J Heaton. *Artificial Intelligence for Humans*, volume 2. Heaton Research, Inc., United States, 2014.

[36] D.N Sotiropoulos and G.A Tsihrintzis. *Machine Learning Paradigms: Artificial Immune Systems and their Applications in Software Personalization*, volume 118. Springer, 2017.

[37] P Kim. *Matlab Deep Learning*. Apress. ISBN: 978-1-4842-2845-6, 2017.

[38] Jr. C. A Janeway. How the immune system recognizes invaders. *Scientific American, a division of Nature America, Inc*, 269(3):72–79, 1993.

[39] D Floreano and C Mattiussi. *Bio-Inspired Artificial Intelligence: Theories, Methods and Technologies*. The Massachusetts Institute of Technologie Press, 2008.

[40] B Rensberger. *In Self Defence*. Oxford University Press, 1996.

[41] S.K Tiwari and G Kaur. Analysis of fuzzy pid and immune pid controller for three tank liquid level control. In *International Journal of Soft Computing and Engineering (IJSCE)*, volume 1, page 2231_2307, 2011.

[42] H Song, B Fang, and P Wang. Research and applications of immune pid adaptive controller in anti-skid braking system for aircraft. In *International Confererence on Information Engineering and Computer Science*. IEEE, 2009.

[43] H Song, B Fang, and P Wang. Application of fuzzy immune pid control based on pso in hydraulic agc press systemt. In *IEEE International Confererence on Intelligent Human-Machine Systems and Cybernitics*. IEEE, 2009.

[44] J Tim. *Artificial Intelligence: a Systems Approach*. Infinity Science Press LLC, New Delhi, 2008.

[45] K Gurney. *An Introduction to Neural Networks*. CRC Press, 1997.

[46] S.S Richard and G.B Andrew. *Reinforcement Leaning: An Introduction*. MIT Press, Cambredge, MA, 1997.

[47] P Apkarian, V Bompart, and D Noll. Non-smooth structured control design with application to pid loop-shaping of a process. *International Journal of Robust and Nonlinear Control*, 17:1320–1342, 2007.

[48] P Apkarian. Tuning controllers against multiple design requirements. In *American Control Confererence*. Washington, DC, USA, 2013.

[49] P Apkarian and P Gahinet. Structured h ∞ synthesis in matlab. *IFac Proc*, 44:1435–1440, 2011.

[50] A Marcos and M Sato. Robust model-matching controller design using matlab "hinfstruct"command. In *SICE International symposium on control systems, Okayama, japan*. SICE, 2017.

[51] *User Manual Magnetic Levitation Experiment manual*. Quanser Inc, Ontario, 2012.

[52] A Bobtsov, A Pyrkin, R Ortega, and A Vedyakov. A state observer for sensorless control of magnetic levitation systems. *Automatica*, 97:263–270, 2018.

[53] S Bouissou and Piriou. F. Study of 3d formulations to model electromagnetic devices. *Transactions on magnetics*, 30:3228–3231, 2002.

[54] A Elhajjaji and M Ouladsine. Modeling and nonlinear control of magnetic levitation systems. *IEEE Transactions on Industrial Electronics*, 48(4):831–838, 2001.

[55] F.G Martins. Martins pid controllers using itae criterion. *Int. J. Engng Ed*, 21(5):867–873, 2005.

[56] S Mukhopadhyay. P.i.d. equivalent of optimal regulator. *Electronics Letters*, 14(25):2–3, 1978.

[57] E.F Rodriguez, C Kubota, G.A Giacomelli, M.E Tignor, S.B Wilson, and M. Mcmahon. 'dynamic modeling and simulation of greenhouse environments under several scenarios: A web-based application'. *Computers and Electronics in Agriculture*, 70:105–116, 2010.

[58] I Cojuhari, B Izvoreanu, D Moraru, A Speian, and A Romanov. Greenhouse temperature control system. pages 33–36. International Conference on Development and Application Systems, 2012.

[59] I Dumitrache, I Dumitriu, and I Mihu. 'automatizari electronice'. *Bucarest editeur*, page 660, 1993.

[60] Y Tipsuwan and M.Y Chow. Control methodologies in networked control systems. *Control Engineering Practice*, 11(10):1099–1111, 2003.

[61] M Cloosterman, N Wouw, W Heemels, and H Nijmeijer. Stability of networked control systems with uncertain time varying delays. *IEEE Transactions on automatic control*, 54(7):1575–1580, 2009.

[62] J.D Decotignie. A prespective on ethernet tcp/ip as a fieldbus. In *Proc IFAC Conference on Fieldbus Systems*, pages 138–142, 2001.

[63] C Anton, H Dan, and O Martin. *TrueTime 2.0 bêta. Reference Manual*. Lund Institute of Technology, Sweeden, 2010.

[64] P wang, H Meng, and Q.Z Ji. Study on fuzzy immune pid controller for main steam pressure control system in marine steam power plant. *Applied Mathematics and materials*, 686:89–94, 2014.

[65] B Southall, B.F Buxton, and J.A Marchant. Controllability and observability: Tools for kalman filter design. In *British Machine Vision Conference*.

[66] V.N Siva Praneeth, V Bharath Kumar, D Sampath, Y.V Pavan Kumar, D John Pradeep, C Pradeep Reddy, and R Kannan. *Implementation of Neural Network-based PID Controller for Speed Control of an IC Engine*. Springer, 2022.

[67] R Hecht-Nielsen. Theory of the backpropagation neural network. In *Neural networks for perception*, pages 65–93. Elsevier, 1992.

[68] R Patel and V Kumar. Multilayer neuro pid controller based on backpropagation algorithm. *Procedia Computer Science*, 54:207–214, 2015.

[69] D.A Permatasari and D.A Maharani. Backpropagation neural network for tuning pid pan-tilt face tracking. In *2018 3rd International Conference on Information Technology, Information System and Electrical Engineering (ICITISEE)*, pages 357–361. IEEE, 2018.

[70] H Xu, J Lai, Z. Yu, and J. Liu. Based on neural network pid controller design and simulation. In *Proceedings of ICCIA-12*, 2012.

[71] S.W Sung and I.B Lee. Limitations and countermeasures of pid controllers. *Industrial and engineering chemistry research*, 35(8):2596–2610, 1996.

[72] J Lee and T.F Edgar. Improved pi controller with delayed or filtered integral mode. *AIChE*, 48:2844–2850, 2002.

[73] J.K Liu. *Advanced PID Control MATLAB Simulation*. Publishing House of Electronics Industry, Beijing, 2004.

[74] Karl Johan Åström and Richard M Murray. *Feedback systems: An Introduction for Scientists and Engineers*. Princeton university press, 2021.

[75] H Clarke. *Optimization and Nonsmooth Analysis*. Canadian Math.Soc. Series. John Wiley and Sons, New York, 1983.

[76] P Apkarian and D Noll. Nonsmooth h ∞ synthesis. *IEEE Trans*, 51(1):71–86, 2006a.

[77] P Apkarian and D Noll. Nonsmooth optimization for multidisk h ∞ synthesis. *IEEE Trans*, 12(3):229–244, 2006b.

Appendix A

Framework for tuning fixed control structures

Assuming the standard form of the structured H ∞ synthesis given by the figure below:

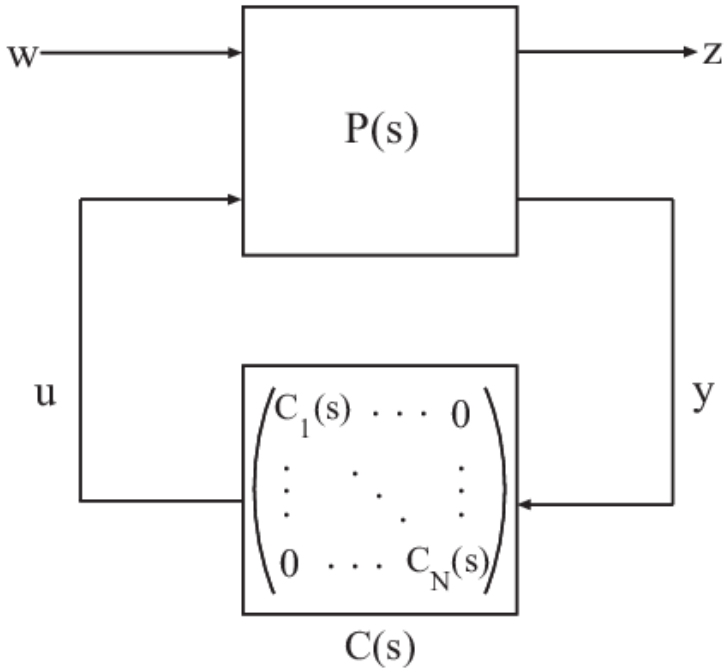

Figure A.1: Standard form of the structured H ∞ synthesis

The structured H ∞ synthesis involves solving semi-infinite, non-convex, and non-smooth algorithms under the form[49]:

$$\min_{C(s)} imize \ \|F_l P(s), C(s)\|_\infty \Leftrightarrow \min_{x \in R^K} imize \ \max_{w \in [0,\infty[} \bar{\sigma}\left(F_l\left(\hat{P}(j\omega), x\right)\right) \quad \text{(A.1)}$$

Where C(s) is the structured controller and the vector x gathers all low-level tunable parameters in C(s). This is a challenging mathematical programming problem, it is worth noting that the right hand side function of the equation (A.1) represents the composition of the convex but nonsmooth function $\max_{w} \bar{\sigma}(.)$ with the non-convex but differentiable mapping $x \rightarrow F_l\left(\hat{P}(j\omega), x\right)$. This composite function is the full description of the Clarke subdifferential is accessible [75]. This program can be rewritten as:

$$\min_{x \in R^K} imize \ f_\infty(x) \Leftrightarrow \max_{w \in [0,\infty[} f(\omega, x) \quad \text{(A.2)}$$

to solve the equation (A.2), it was constructed in a tangent model around the current iterate x which represents a "quadratic first-order" local approximation of the original problem. An adequate descent direction h is then computed by solving a convex quadratic program of the form:

$$\min_{h \in R^K} imize \ \widehat{f_\infty}(x+h) := \max_{\omega \in \Omega_f} f(\omega, x) - f_\infty(x) + \phi_\omega^T h + \frac{1}{2} h^T Q h \quad \text{(A.3)}$$

Where: Ω_f is some finite set of frequencies and $\phi_\omega \in \partial f(\omega, x)$ is a subgradient of $f(\omega, x)$ many possible implementations of this scheme are available in [76] . However, Ω_f has to contain the active frequencies ω_a achieving the peak value in equation (A.2). This minimal requirement is enough for the program to converge[49]. By adding a few well chosen frequencies, the tangent model given by equation (A.3) can be improved, where including frequencies bracketing, the corresponding pic can significantly accelerate the convergence [76, 77]

A.1 Software Tools

The solution of the structured H ∞ problem can be effectuated in Matlab using some software tools, these last are available in Robust control toolbox. The general structure of tunable transfer function provides by the software is given by the instruction below:
C=ltiblock.tf('C','number of zeros','number of poles')
However, it exists other instructions for pre-defined tunning for gains, PIDs, and state space. The tunable block of the PID controller is denoted by this expression:

C=ltiblock.pid('C','PID')

The structured H ∞ design achieved by the satisfaction of the following requirements for the feedback loop:

$$\|W_e S\|_\infty < 1, \|W_t T\|_\infty < 1 \quad \text{(A.4)}$$

Where: S is the sensitivity function and the complementary sensitivity function, and W_e,

W_t are the frequency-weighting functions.
The weighting functions are represented using the following instruction makeweight:

W = makeweight(dcgain,[freq,mag],hfgain)
W(0)=dcgain
W(Inf)=hfgain
W(jfreq)=mag
The aggregate transfer function $H(s) = Diag(W_eS, W_tT)$ is represented in Matlab by the variable H0 and it depends on the tunable PID block, it is given by the following expression:

H0=blkdiag(We*S,Wt*T)
Once the standard form is formulated, the minimization of the linear fractional transformation is achieved by the following command:

op=hinfstructOptions('RandomStart',number of random starts)

[H,gam]=hinfstruct(H0,op)
Hinfstruct can be configured to run multiple optimizations from random generated starting points.
The tuned PID controller has an access using the following command:

C1=getBlockValue(H,'C')

Appendix B

PID equivalent to the linear quadratic regulator

The following method proposed by Mukhopadhyay is designed specially to avoid high cost and complexity of optimal regulators, starting from the linear state space system:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned} \tag{B.1}$$

Assuming $y, \int ydt, \dot{y}$ as the inputs of the controlled input u which can be expressed as:

$$u = -K_p y - K_i \int ydt - K_d \dot{y} \tag{B.2}$$

for

$$u(0) = 0$$

By substituting the first equation in the second equation we can write:

$$\begin{aligned} u &= -K_p Cx - K_i \int ydt - K_d C(Ax + Bu) \\ &Or \\ u &= -\bar{K}_p Cx - \bar{K}_i \int ydt \end{aligned} \tag{B.3}$$

Where the new PID gains are defined as:

$$\bar{K}_p = (I_m + K_d CB)^{-1} \left(K_p C + K_d CA \right) \tag{B.4}$$

$$\bar{K}_i = (I_m + K_d CB)^{-1} K_i \tag{B.5}$$

Expressing the new variable:

$$z = \int_0^t y dt \tag{B.6}$$

for

$$z(0) = 0$$

Then we can write:

$$\dot{z} = y = Cx \tag{B.7}$$

The augmented states of the system will be defined as:

$$\bar{x} = [xz]^T \tag{B.8}$$

The new system's representation will be as follow:

$$\dot{\bar{x}} = \bar{A}\bar{x} + \bar{B}u \tag{B.9}$$

Where the augmented system matrices are expressed as:

$$\bar{A} = \begin{bmatrix} A & 0 \\ C & 0 \end{bmatrix}, \bar{B} = \begin{bmatrix} B \\ 0 \end{bmatrix}$$

Minimising the following performance index with $\bar{Q}$, R as augmented state and input weighting matrices:

$$J = \frac{1}{2} \int_0^\infty \left(\bar{x}^T \bar{Q} \bar{x} + u^T R u \right) dt \tag{B.10}$$

The desired optimal control law is given by the equation below:

$$u^* = -R^{-1} \bar{B}^T \bar{P} \bar{x} \tag{B.11}$$

Where $\bar{P}$ is the positive-definite solution of the nonlinear algebric Riccati equation:

$$\bar{P}\bar{A} + \bar{A}^T \bar{P} \bar{B} R^{-1} \bar{B}^T + \bar{Q} = 0 \tag{B.12}$$

Comparing the equation (B.3) and (B.11) we can extract the following expressions:

$$\bar{K}_p = R^{-1} B^T P_{11}, \bar{K}_i = R^{-1} B^T P_{12} \tag{B.13}$$

P_{11}, P_{12}, are the submatrices of the matrix $\bar{P}$:

$$\bar{P} = \begin{bmatrix} P_{11} & P_{12} \\ P_{12}^T & P_{22} \end{bmatrix}$$

(B.14)

From these new gains we can rewrite the PID gains in the forms:

$$\left[K_p K_d\right] = \bar{K}_p \bar{C}^{-1} \tag{B.15}$$

Where:

$$\bar{C} = \begin{bmatrix} C \\ CA - CB\bar{K}_p \end{bmatrix}$$

$$K_i = (I_m + K_d CB)\,\bar{K}_i \tag{B.16}$$

B.1 Software Tools

The augmented system is presented in Matlab as follow:
A=[As,zeros(ns,ls);Cs,zeros(ls,ls)];
B=[Bs;zeros(ls,ms)];
C=[Cs,zeros(ls,ls);zeros(ls,ns),eye(ls,ls)];
Cd=[Cs,zeros(ls,ls);zeros(ls,ns),zeros(ls,ls)];
Where: As, Bs, Cs are given from the initial state representation, ns,ls,ms are the matrices lengths.
For the controller design:
LQR-based PID output-feedback controller design for linear time invariant systems is given by the following function which based on S. Mukhopadhyay: P.I.D. equivalent of optimal regulator .

[F,P,E]= LQRPID(sys,Q,R,N)

It calculates the optimal (if l=n/2) or the sub-optimal (if l $\sim$ =n/2) output-feedback PID (Proportional-Integral-Derivative)
Where:
F, P, E are respectively: the PID parameters presented in a vector, the solution of Riccati equation, the closed loop system eigen values.
However, this function is effectuated after testing the matrices lengths and applying the following instructions:

$$[\sim, Pr, Er] = lqr(A, B, Q, R, N)$$

P11=Pr(1:ns,1:ns);

P12=Pr(1:ns,ns+1:ns+ls);

$\bar{K}_p$ and $\bar{K}_i$ are given by the two following instructions:

sKp=inv(R)*(Bs'*P11+N(1:ns,1:ms)');

sKi=inv(R)*(Bs'*P12+N(ns+1:ns+ls,1:ms)');

The matrix given by the equation (B.16) is defined by the the instruction: sC=[Cs;Cs*As-Cs*Bs*sKp];

Kpd=sKp*pinv(sC);

The final parameters of the PID controller (equations (B.15),(B,16)) are the following:

Kp=Kpd(1:ms,1:ls);

Kd=Kpd(1:ms,ls+1:ls+ls);

Ki=(eye(ms)+Kd*Cs*Bs)*sKi;

Fr=[Kp Ki Kd];